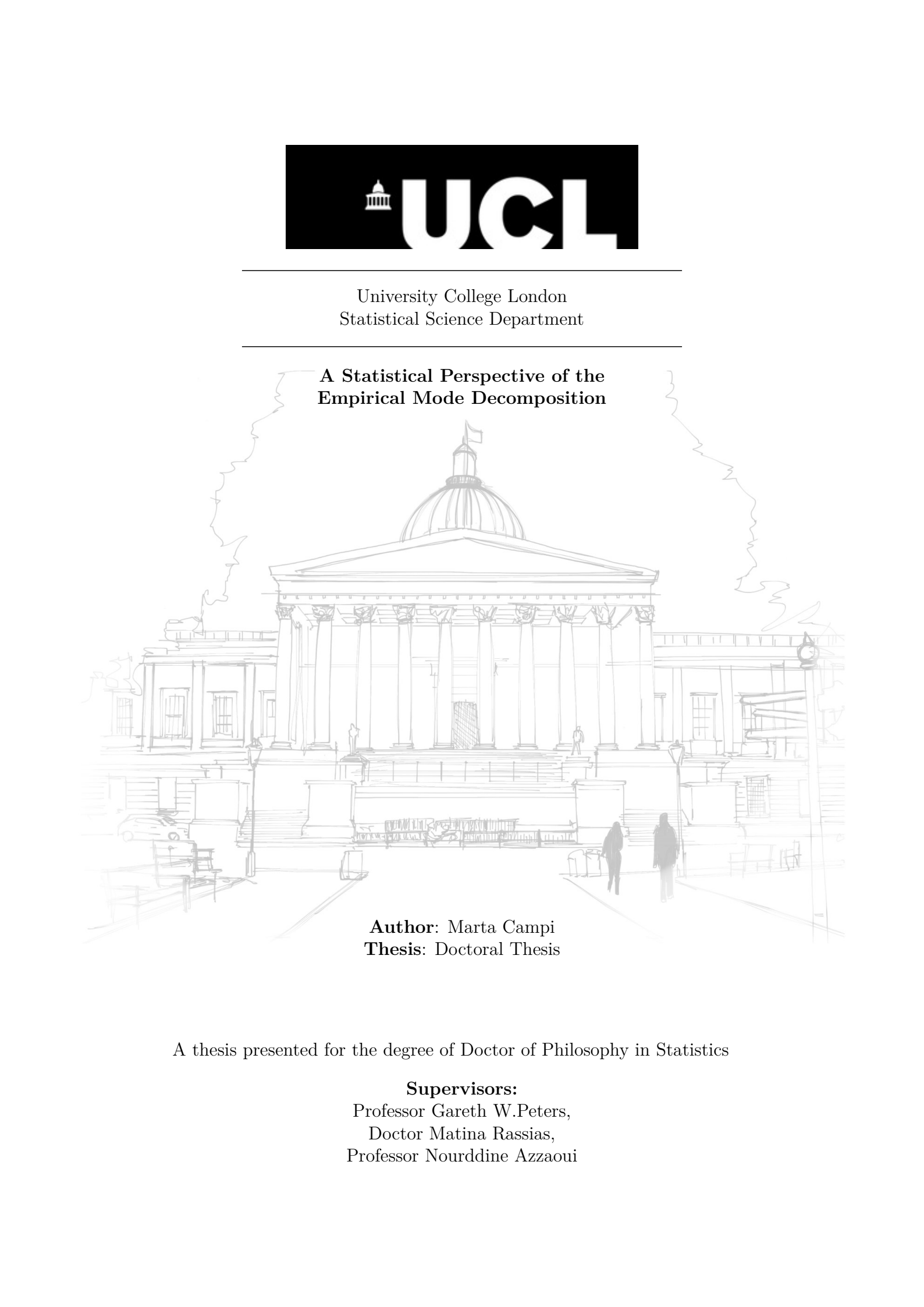


University College London
Statistical Science Department

A Statistical Perspective of the Empirical Mode Decomposition

Author: Marta Campi
Thesis: Doctoral Thesis

A thesis presented for the degree of Doctor of Philosophy in Statistics

Supervisors:
Professor Gareth W.Peters,
Doctor Matina Rassias,
Professor Nourddine Azzaoui

Declaration of authorship

I, Marta Campi, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

Date: 1 December 2021

Signed:

Acknowledgements

I have the suspect that being a researcher, like any other totalising experience involving pure devotion, is a journey, and this is not a finishing line. Quite the opposite, instead, it appears to be "the" starting point. Understanding but mostly appreciating this concept is a luxury that I have been the pleasure to share with some people along this way, and who must be individually acknowledged since they made a peculiar difference in one or multiple ways. So, here it is.

First and foremost, to my primary supervisor, Gareth W. Peters. His guidance, support, constant belief in the person that I am and my skills, for having offered me this incredible opportunity, for sharing with me his way of life and work code, for never compromising on the quality of the outcome, thank you. The possibility of working together is one of the most precious things, and I will never be thankful enough for it.

To Matina Rassias, my second supervisor, for her fantastic way of being, her knowledge, her passion for teaching, her kindness and her precision of work at the same time. You had strongly developed the person that I am and hugely enriched me along this way. Thank you.

To my collaborators. To Dorota Toczydlowska, for her strong personality and kindness, patience, professional way of working, friendship, her bits of advice. To Ioannis Chalkiadakis, for his friendship, presence, understanding and sharing my exact perspective, for the pleasure of working and being friends at the same time. To Kylie-Anne Richards, for her sincere way of being, professionalism, guidance, constant support, and belief in the person I am, her friendship. To Tomoko Matsui, for her incredible and unique way of being, her positive attitude, code of work, and sense of humour. To Dimitris Christopoulos, for his support, his belief in my skills, his genuine way of sharing, the possibility that he offered me to develop whom I would like to become. To Nourddine Azzaoui, for his kindness and sharing with me his pleasure of working in probability theory.

To my family. To my father. For his constant support, strong way of life, and intuitions about me and the person I have become. Thank you. I would have never got to this point without you. To my mother. Her integrity, solidness, support and guidance have profoundly shaped my character. Thank you, mum. To my brother, for being the only one who can truly feel and understand who I am, beyond any kind of limit, space or time. Thank you, Jino.

To Paolo. His constant presence from day zero of this experience, support, belief

that I can always conquer whatever and whoever, this is undeniably yours.

To O., no words are required here.

To La Judy e Katanga. For speaking my language since the very first day we met, for the person I have become and the people you have become, for having evolved together, but always being present, in this timeless research of ourselves that does not foresee any compromise. Thank you.

To Sara. For being my best friend. For having been there, on my side, constantly, with no doubts at all, for not judging me but advising me if worried about me, for taming me, and for letting me taming you, as real friends should always do.

To Fabri. For always having believed that I am something more, for sharing with me the absolute pleasure of doing research and the importance of pursuing this life, for always being there no matter what. Thank you.

To my flatmate, Andre. For sharing with me the doubts and the happy moments of this choice. I believe that you know that the support that you gave me through these years was incredibly meaningful. Thank you.

To the guys at the Stats Department at UCL. To Fede, Marco, Marta, Rui, Xinyi, Albe, Rodrigo, Anna, Hannah, Theo, for being my friends and always accepting me, for having fun together, for teaching together, for going out, for sharing the idea of doing of PhD and being a researcher. Thank you all, guys.

To my volleyball team and my coaches. To Karol, Koleva, Gaia, Vicky, Irene, Yoli, Aga, Cecile, Angie, Gabi, Mark, Alex, Luis. For being my second family in London, always supporting me and for being on my side. Thank you, guys.

Last but not least, to Jacques. For having reminded me of the real meaning of being a dreamer and why this is undoubtedly the path I should follow.

Journal Papers

Published

1. "Machine Learning Mitigants for Biometric Cyber Risk"
   Marta Campi, Gareth W. Peters, Nourddine Azzaoui.
   IEEE Access.

In submission

1. "Stochastic Embedding Approaches for EmpiricalMode Decomposition"
   Marta Campi, Dorota Toczydlowska, Gareth W. Peters.
   IEEE Signal Processing Magazine

In preparation

1. "Spectral Representation Learning For Real-Valued Signals by Complex Gaussian Process"
   Dorota Toczydlowska, Marta Campi, Gareth W. Peters.

2. "Green Bond Performance and Risk Indicators"
   Marta Campi, Kylie-Anne Richards, Gareth W. Peters.

International Visits

1. Institute of Statistical Mathematics (ISM), Tokyo, Japan.
   February 2018

2. Laboratoire de Mathématiques Blaise Pascal, Université Clermont Auvergne, Clermont-Ferrand, France
   July 2017

3. 47-th Probability Summer School Saint-Flour, Saint-Flour, France
   July 2017

Abstract

This research focuses on non-stationary basis decompositions methods in time-frequency analysis. Classical methodologies in this field such as Fourier Analysis and Wavelet Transforms rely on strong assumptions of the underlying moment generating process, which, may not be valid in real data scenarios or modern applications of machine learning. The literature on non-stationary methods is still in its infancy, and the research contained in this thesis aims to address challenges arising in this area. Among several alternatives, this work is based on the method known as the Empirical Mode Decomposition (EMD). The EMD is a non-parametric time-series decomposition technique that produces a set of time-series functions denoted as Intrinsic Mode Functions (IMFs), which carry specific statistical properties. The main focus is providing a general and flexible family of basis extraction methods with minimal requirements compared to those within the Fourier or Wavelet techniques. This is highly important for two main reasons: first, more universal applications can be taken into account; secondly, the EMD has very little a priori knowledge of the process required to apply it, and as such, it can have greater generalisation properties in statistical applications across a wide array of applications and data types.

The contributions of this work deal with several aspects of the decomposition. The first set regards the construction of an IMF from several perspectives: (1) achieving a semi-parametric representation of each basis; (2) extracting such semi-parametric functional forms in a computationally efficient and statistically robust framework. The EMD belongs to the class of path-based decompositions and, therefore, they are often not treated as a stochastic representation. (3) A major contribution involves the embedding of the deterministic pathwise decomposition framework into a formal stochastic process setting. One of the assumptions proper of the EMD construction is the requirement for a continuous function to apply the decomposition. In general, this may not be the case within many applications. (4) Various multi-kernel Gaussian Process formulations of the EMD will be proposed through the introduced stochastic embedding. Particularly, two different models will be proposed: one modelling the temporal mode of oscillations of the EMD and the other one capturing instantaneous frequencies location in specific frequency regions or bandwidths. (5) The construction of the second stochastic embedding will be achieved with an optimisation method called the cross-entropy method. Two formulations will be provided and explored in this regard. Application on speech time-series are explored to study such methodological extensions given that they are non-stationary.

Impact Statement

The research presented in this thesis provides benefits both inside and outside academia and contributes to methodological and application-related developments which address essential questions of four main research areas: non-stationary time-series, machine learning (ML) for Gaussian Processes(GP) and multi-kernel methods (MKL), automatic speaker verification (ASV) problems and health diagnostic.

In the era of big data, one relevant aspect that should be taken into account when doing data analysis is the non-stationary and non-linear nature characterising the data system of the studied phenomenon. This thesis explores the discrimination power for classification problems of a time-series and time-frequency method known as the Empirical Mode Decomposition (EMD) and introduces it to a statistical context. Different components of the method are explored, enhanced and exploited for feature extraction tasks that provide robust performances in multiple classification settings.

A further relevant point that should be considered in analysing data is that the observed signal of the studied phenomenon might be comprised of multiple, time-varying basis components whose structural behaviour cannot be easily detected given their intrinsic composite nature using standard ML procedures. In the framework of Gaussian Processes, a solution offered to this problem is enclosed by MKL methods. These techniques employ complex kernel structures for GP to reproduce the underlying signal structure. This thesis proposes an alternative way to construct an MKL by using the time-series method above introduced, the EMD, whose statistical interpretation is highly beneficial in the context of multi-component signals. As a result, a more powerful tool for GP development is constructed and could be applied.

These core methodological developments could impact the public and private sectors in the data analysis process since a powerful non-stationary technique is proposed in combination with a refined version of the standard GP-MKL framework and, therefore, complex data system setting could be challenged.

At an application level, the proposed method deal with two relevant speech analysis problems. The first one is the one of ASV system. Particularly in the private sector, and within several contexts as financial services, call centres, mass-market of human-computer interfaces, ASV technologies are nowadays facing the challenge of spoofing attacks mimicking a target speaker's voice in person or remotely

via artificial tools. The EMD extracts speech signal features to assess their discrimination power in classifying natural and synthetic voices. Results provide robust performances within multiple ASV scenarios, such as in the presence or absence of background noise. The second application involves the health diagnostic of Parkinson's disease through speech sample voices. This topic has become of central interest for two critical reasons, i.e. to facilitate the daily routine of a patient affected by recurrent visits to the clinic and remove the subjectiveness of the disease assessment, which strictly links to a set of questions posed by the doctor. The proposed methodological framework aims to detect the presence or absence of the disease through speech samples in order to promote telemonitoring for such a disease. Results provide high performance in this sense, overcoming gold standard existing methodologies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The importance of data analysis has become increasingly critical in fields such as finance, biology, environmental studies, economics, engineering, amongst others. This has been driven by the necessity of statistical modelling to facilitate a deeper understanding of real-world phenomena to predict the future according to the past. One of the most essential approaches in statistical science considers observations which are sequentially recorded over time and produce a time-series. The way statisticians typically analyse a time-series is by viewing it as a realisation of a stochastic process, such that future realisations will depend on past observations in a stochastic manner. The goal of time-series analysis may vary across applications and can include trend forecasting, summarising features dynamically over time or producing explanatory summaries from smoothed, filtered or predictive stochastic models. However, the central task is always to specify the probability law of the underlying process in order to capture its dynamics, provide an accurate description and compute reliable forecasts of future outcomes. Different perspectives can be considered in the analysis, and this is reflected in the methodologies, which exploit different sets of assumptions.

The data generating mechanism of a stochastic process may be either discrete or continuous in its inherent nature. As a result, the choice of modelling links accordingly to the class of analysed processes. One possibility of the continuous class of models is working with diffusions respecting mathematical properties, such as, for example, the existence of the stochastic differential equation (SDE) form. Another option is a time-series model, which has specific attributes such as linearity or dynamic volatility, etc. Such models are applicable in the case of continuous processes partially observed over time. In the case of a partially observed discrete time process, a time-series may offer a more efficient modelling solution. Within these classes of models, several assumptions have to be fulfilled. For example, to characterise solutions to the diffusion equation, restrictions on drift or volatility may be necessary, Whilst in the case of a time-series model, constraints on stationarity often apply. These kinds of features outline the characterisation of the family which describes the data. One of the statisticians' tasks is to consider such a dataset under the assumptions of the possible families characterising it and estimate a lower-dimensional representation. Statisticians

refer to such a reduced data representation as a model, and it could one of two basic forms, i.e., non-parametric or parametric.

If the probability law belongs to a family specified according to finite-dimensional parameters, then the adopted model is named a parametric model. They usually consist of model identification, parameter estimation, model checking and forecasting. Parametric models offer a well-structured framework, allowing a straightforward interpretation and a known likelihood. This allows for the estimation of parameters to be more tractable in general, guaranteeing the statistical properties, such as efficiency, absence of bias, consistency and asymptotic normality of the resulting parameter estimates. However, model-misspecification frequently occurring in non-stationary contexts affects the performance of such stationary parametric models. Furthermore, adjusting classical parametric model families to capture non-stationarity can result in models with excessively high dimensional parameter spaces, producing biases of the parameter estimates or requiring a large number of parameters to detect the essential dynamics of the original process. Moreover, testing the assumptions that have to be satisfied by the time-series to embed the parametric specification may be considerably challenging. These issues may be overcome by considering a non-parametric model.

Non-parametric models provide a great deal of flexibility to capture features of the data; yet, they may not generalise as well in out-of-sample analysis. Their significant advantages correspond to higher flexibility, together with fewer assumptions for the model specification. Drawbacks of these methods are enclosed in their definition: they can be considered “black boxes", which become problematic to interpret or compare, since they usually lack mathematical definition in closed form. As a consequence, several algorithms are taken into account to estimate the model. In addition, the model may not be fitted on the original raw data but rather summaries of the data. One method often applied is given by two stages: feature extraction and data modelling.

A fundamental approach employed to fully comprehend the underlying structure of a process or a time-series is representing it in terms of core components (functions or processes), which depend on two perspectives: pathwise decomposition and stochastic process decomposition. In the classical approach, one can think of a decomposition of a process, say of a time series, as representing the time series as n-decomposable into weighted combinations of other simpler processes or random variables. The classical linear process decomposition has to be attributed to Wald, where a time series is decomposed into a linear form (Note: linear in the coefficients of the basis). In terms of the model specification, this is equivalent, for a given time-series S_t to the following:

$$S_t = \sum_i w_i \phi_i(S_t) \tag{1.1}$$

where w_i represents a certain coefficient defining a linear combining rule, while, $\phi_i(S_t)$ is the basis taken into account, which is non-linear. The majority of the time-series models fall into this class, whether parametric or non-parametric.

The reason for considering this approach as a meaningful representation is the power of the basis, which can better capture features of the original stochastic process and then reconstruct it efficiently. A relevant aspect in this context is the definition of the basis; regardless of the domain taken into account, one of the most significant challenges affecting the time-series community is the problem of a priori or a posteriori basis decomposition methodologies. A priori models typically rely on stationarity and linearity (or both) of the given data generating process and often produce a decomposition that requires an infinite number of bases. When it comes to the a posteriori basis, a data-driven modelling procedure can be formulated. Following this concept, all the mathematical constraints given by an a priori basis are, therefore, avoided. The main issue of an a posteriori set of functions considered as the basis is the lack of a theoretical definition. As a consequence, sets of non-unique bases are obtained, since they are strictly defined at an empirical level. The lack of uniqueness of the basis function is a challenge that is encountered with the method of focus in this thesis, the Empirical Mode Decomposition (EMD).

The need for stochastic modelling is of great relevance at this point. Observed time-series are one of the sets of results of a stochastic process, which indeed carries a random component. Therefore, uncertainty in the set of observations has to be controlled by the given method. The common practice of a simple case as the regression model is as follows: the method is deterministic, conditional on the observed time-series; by considering a set of assumptions, the model is posed, and an objective function (or risk functional) with associated loss function defining the class of estimator is minimised or solved for the unknown parameters, in terms of x and y, given values. A deterministic system of equations is then obtained, which, for instance in linear regression modelling, corresponds to the Least Squares minimisation (which in this case is attainable in a closed-form system of equations). Then, the solution to these estimating equations produces, in statistics, the resulting estimators of the model parameters, or basis coefficient weights in the case of decomposition (1.1), which will naturally be realisations of random variables themselves, as they are comprised of functions of the input observation data. Typically, S is treated as known or could be generated from a random process or given from experimental design, while y is a random outcome. Therefore, the estimator although deterministic from the way it was obtained (optimising a loss function minimisation), is, by definition, random since it is a function of the data, which is a realisation of a random process.

The time variation of a time-series is fundamental to describe its evolution. However, there are several applications in which the analysis of a stochastic process on the time domain does not fully describe it and, consequently, it is advantageous to consider another aspect of characterisation of a process or time series, namely the frequency domain perspective. The frequency-domain perspective provides an alternative characterisation of information in the process that may be more readily interpreted and can shed additional insight on structures in the process or time series under study. Analysing a time-series through its spectral

representation can often reveal superimposition of different waves, which can be interpreted more straightforwardly and hence offer more efficient summaries of the original process, as well as insight into appropriate choices for basis decompositions, such as presented in 1.1.

This thesis finds its location in the above discussions by considering a time-frequency, non-parametric basis decomposition technique, commonly used in signal processing and communication engineering, called the Hilbert Spectrum, which is defined by the Empirical Mode Decomposition (EMD), along with the Huang Hilbert Transform (HHT) (Huang et al., 1998). It offers a representation of the original process as a summary expressed by the EMD basis, called Intrinsic Mode Functions (IMFs). The Properties of this a posteriori extraction method are deeply investigated, since it accommodates non-stationarity and non-linearity of the given process. Each basis represents an estimator of a certain locally time-adapted oscillatory mode characterising the original time-series and is fully expressed as a function of the data. Moreover, through the HHT, it is possible to observe each IMF within its frequency domain. The main contribution will be studying the statistical properties of these bases.

An essential aspect being considered is that EMD is a deterministic pathwise technique. Therefore, the given decomposition would not accommodate a stochastic model formulation without further statistical assumptions being made. It is the intention of this thesis to embed this pathwise deterministic characterisation into a more general stochastic model framework that brings it into the realm of time-series models and allows one to move from pathwise extrapolation to actually performing statistical forecasting in a meaningful time-series sense. The purpose is to characterise the distribution of each underlying stochastic process carrying a specific mode of oscillation whose observed realisation corresponds to the decomposition basis function. To accomplish this goal, the estimator characterisation of the basis representation has to be consistently expressed as a natural consequence of the chosen stochastic model family of the original signal. Consequently, the first question to be addressed is the identification of a suitable family of stochastic models. The required assumption to proceed asserts that the convolution of the stochastic processes of the pathwise realisation of the bases provides the stochastic process describing the observed pathwise realisation of the original signal. Several statistical tools have to be studied. First and foremost, the choice of the stochastic model falls into the class of Gaussian Processes. These are stochastic models offering relevant properties, such as flexibility or smoothness, suitable for real-world applications. Moreover, the convolution of each Gaussian process for the basis estimators will combine and produce the Gaussian process for the original signal, so that the desired assumption will be fulfilled.

The primary interest of this thesis is to study several aspects of this stochastic embedding. A significant step will be the estimation stage of the Gaussian processes. This class of stochastic processes also allows great versatility in the covariance function. Several solutions have been proposed in the literature cov-

ering such a task and will be reviewed in detail within this work. Different perspectives in the estimation approach used to compute the covariance function could reveal the unknown data structure more efficiently. For example, the choice of stationary or non-stationary covariance functions might increase or decrease the performance of the tasks of interest, such as classification or forecasting, especially in real-world application studies. Given the extensive use of Gaussian processes in the machine learning community, various enhancements have been introduced in the literature on kernel methods that directly link to the study of covariance functions. The equivalence between covariance matrices and Mercer's kernel gram matrices given by Mercer's theorem and its conditions will be presented and discussed in this work. This thesis reviews several approaches, both stationary and non-stationary, to construct kernel functions to investigate the given embedding properties.

The structure of the stochastic embedding for the EMD that is considered in this thesis involves a convolution of multiple Gaussian processes, one for each extracted basis characterising one of the oscillation modes of the original signal. In this regard, each mode, and therefore each basis function, carries a specific covariance structure that can be modelled according to a unique kernel, differing one from another. The development of these settings became of particular interest since it captures the idea behind the proposed methodology, a recent method used in many machine learning techniques, known as multiple kernel learning. Instead of considering a unique kernel, this approach relies on different kernel functions, one for each extracted feature representation of the original data. Afterwards, it merges the predefined kernels into a unique expression, either in a linear or nonlinear fashion, according to a weighting combination rule function. The benefit in the developed setting of this framework is that it allows modelling each decomposition basis function stochastic process according to a specific kernel and then merging them by following such a multi-kernel technique.

The challenge addressed by this research stems from the lack of a statistical or probabilistic background of the EMD method. It aims to shed light on methodological research questions, introducing statistical properties and behaviours of the technique. The targeted issues that are considered examine and extend this adaptive decomposition method and correspond to different classification problems. The purpose is to disclose statistical properties of the EMD by extracting a set of features characterising the decomposition procedure and then investigating their discriminatory power.

The applications selected to develop this statistical background concern speech signals within different scenarios. By being strongly non-linear and non-stationary, speech time-series embody an exemplary candidate for such a challenge.

The first application, relevant within several contexts, such as financial services, call centres, mass-market or human-computer interfaces, involves Automatic Speaker Verification (ASV) technologies, which are subject to spoofing attacks mimicking a target speaker's voice, either in person or remotely via ar-

tificial tools. The EMD is employed to extract features from speech signals to assess their discrimination power in classifying natural and synthetic voices.

The second application aims to exploit the EMD to detect Parkinson's disease (PD) within a set of male and female speakers at different stages. One of the symptoms affecting PD patients consists of cerebellar dysfunction resulting in impaired coordination, or "ataxia". The speech disturbance that results from cerebellar dysfunction is referred to as 'ataxic speech'. The EMD bases are employed to provide an objective assessment of the dysfunction by quantifying disturbances in the acoustic equivalences of disturbance in displacement, direction and rate (velocity). The IMFs provide a powerful tool to capture such time-dependent speech attributes.

1.1 Motivation

A core tenet of statistical modelling is to form a parsimonious characterisation of data to be used in statistical exercises, such as regression, classification or interpolation, to have summary statistics or summary characterisations of a time-series. In this regard, one of the most discussed scenarios corresponds to stationary versus non-stationary settings. The natural question to ask is which kind of functional representation provides a parsimonious view of the observed time-series to generalise modelling. The primary motivation behind this work is to explore the category of non-stationary methods and to compare them to stationary methods for different tasks. Within such a perspective, time-frequency analysis is the focus of this thesis. Such an area strongly requires the definition of a functional representation dealing with non-stationarity. A feature often affecting non-stationary data is non-linearity. Most of the time-frequency literature provides methods which can deal either with one or the other, even if non-linearity and non-stationarity are usually linked. Therefore, the first part of this thesis investigates the existing time-frequency analysis methods and compares the functional representation basis employed to summarise the data.

Time-frequency analysis methods have been widely investigated algorithmically, given their ability to separate the variability of a stochastic process into contributions related to oscillations. The second motivation for this work is to explore the statistical interpretation of such methods to characterise them from a mathematical perspective. Time and frequency domains reflect statistical traits of a time-series that reveal the structure of the data-generating process.

The second part of this thesis focuses on embedding the selected time-frequency analysis method into a stochastic framework. An essential differentiation consists of a deterministic basis function decomposition based on spectral analysis versus a stochastic spectral decomposition. Within the former, the decomposition relies on the basis representation of a deterministic signal, while, from a stochastic perspective, the method relies on processes' characteristic functions. It is central to this thesis to define the connection between the deterministic decompositions

on a sample path and their extensions to a stochastic embedding, allowing us to investigate them at a probabilistic level. Motivations to achieve such a stochastic embedding are to use each basis significantly for regression, forecasting and classification. The ultimate purpose is to quantify factors such as predictive uncertainty and make a model selection through inference procedures to formally define performance rates. The primary challenge of this area is to structure a kernel function dealing with non-stationarity, generated by the superimposition of different dynamics intrinsic to the basis functions. Another relevant movitation of this work is to study the stability of the selected kernel functions for the EMD bases and find an efficient technique defining the kernel hyperparameters. Gaussian Processes are the machine learning method employed to develop such a framework, since they allow kernel functions with a non-stationary structure highly suitable for tackling mode-mixing affecting the EMD.

One recent method that could be compared to the EMD is the decomposition method presented in subsection 1.2.3, known as Singular Spectrum Decomposition (SSD), which represents a generalisation of the existing SSA. This technique relies on the definition of a trajectory matrix, which finds its roots in studying the evolution of a dynamic system (Packard et al. (1980), Ruelle (1980)). This method aims to efficiently identify time scales associated with the original signal and uses the SVD to define elementary matrices to capture its structural variation. The problem of "isolating" the various time scales characterising a given signal has been a challenging issue, greatly affecting different time-frequency methods. One motivation at the base of the proposed stochastic embedding is tackling this problem by characterising the stochastic distribution of each specific intrinsic time scale through a Gaussian Process modelled according to a distinct kernel function.

Another relevant point in statistics is forecasting models that differentiate between the high-frequency content of their underlying stochastic process and the low-frequency content. The EMD suits this framework by providing basis functions ordered according to their frequency contents. Therefore, given the forecasting techniques using the EMD presented in subsection 1.2.2, an essential aspect is proposing a reliable statistical forecasting model adapted to the non-stationary context, which considers only low-frequency or high-frequency content separately. Such a challenge is one of the motivations of this work and will be solved in this thesis.

One of the most discussed issues affecting traditional time frequency methods arises from their time-resolution. The problem encountered with these methods is referred to as the "uncertainty principle" in signal processing, in analogy to Heisenberg's uncertainty principle. This principle states that it is impossible to determine a given particle's position and velocity simultaneously with arbitrary accuracy. One way to understand this concept in time-frequency is by considering the Short Time Fourier Transform. This chops the time domain data into pieces and then tapers each piece with an appropriate window function. Finally, it estimates the power spectrum for each window segment. The problem associated

with this process is the choice of the window size. If too large, there will be a good frequency resolution but a reduced time localisation. Conversely, if chosen too narrow, the window provides a good localisation in time but a poor one in the frequency domain. Hence, there is a trade-off that needs to be accounted for, and this difficulty affects most of the existing time-frequency transforms. The EMD with the Hilbert transform provides a natural solution to this problem by being a fully data-adaptive method. However, this method often fails in the most complex settings (highly non-stationary), due to the mode-mixing of the sifting procedure. In subsection 1.2.5, a stochastic embedding is proposed that aims to tackle this issue by constructing an adaptive time-frequency resolution partitioning the frequency domain through an optimisation technique known as cross-entropy. This will allow more flexibility and a much better resolution in generating new bases, called band-limited IMFs, derived from the existing ones but achieving a much better resolution of the time-frequency domain.

A further motivation connects to the considered application concerning speech analysis. Speech signals are inherently non-stationary, and the speech community is actively looking for techniques to handle such a feature. Furthermore, speech is considered the result of the superimposition of different frequencies known as "formants", which correspond to concentrations of acoustic energy around a particular frequency in the speech wave. Specifically, each format corresponds to a resonance mode of the vocal tract. The ranges at which such frequencies lie are unknown a priori and depend directly on the vocal tract length, making their estimation a highly biometric task. The first application develops a method dealing with speaker verification, i.e., the detection of real speaker voices versus synthetic or spoofed ones. The central motivation is to provide robust characterisations of speech time-series which address several problems, such as different sources of disturbance, like background noise or the different pitch of different speakers, or different algorithms generating the synthetic voice. The second application foresees the study of the proposed stochastic embedding in detecting the presence or absence of Parkinson's disease by using voice signals. The central motivation is to characterise the two families of utterances by using different kernel structure.

1.2 Background and related work

This section aims to provide a background of the different works considered in this thesis to develop a statistical perspective and analysis of the EMD. Multiple literature reviews have been required, considering various perspectives of three disciplines: nonparametric statistics, machine learning and signal processing. Therefore, a review of the main components employed to tackle the set of research questions is given. The first part introduces the main history of the EMD, an explanation of why it is relevant within different research areas is the first objective of this section. Furthermore, the history of the notion of instantaneous frequency will also be presented, since this is of high priority. The second part

will cover the different forecasting techniques that have been developed in the literature, making use of the EMD. This is highly central to understanding the need for the proposed stochastic embedding models. Afterwards, a recent method introduced in the signal processing community is presented. This method is presented and discussed in this thesis since, using a different perspective, it tackles a similar problem of defining the time scales of a given signal to capture its frequency variations over time. The author believes this is of relevance to understanding the motivation behind this work better. The following subsections describe machine learning techniques, such as Gaussian Processes and Multi kernel learning, employed in the development of this thesis. After that, the cross-entropy method is presented. This method comes from the literature of rare event simulations and will be used to propose a stochastic embedding model. Last, an introduction to speech signals is presented, with an explanation of why this topic is relevant to the constructed methodology .

1.2.1 History on the Empirical Mode Decomposition

A key focus of this thesis will be to consider decompositions of the form given in Equation 1.1 that are adapted to non-stationary and non-linear contexts. The core method studied will be based on the Empirical Mode Decomposition, which was introduced by Huang et al. (1998). In this framework, the basis function in Equation (1.1) will not be a priori specified in a parametric functional form; rather it will be a non-linear, time-varying basis that will only be specified by some characterising mathematical properties. This makes the extraction, and functional representation of such basis functions, a very interesting statistical modelling challenge. Before addressing these challenges, the mathematical properties and justification for the EMD basis representation framework are first explained.

In their work, Huang et al. (1998) highlight that the ideal basis for expanding the original time-series should embed four main features, named as locality, adaptivity, completeness and orthogonality. Locality and adaptivity detect non-linear structures in the data and non-stationarity properties, respectively. Locality is the most critical feature able to deal with non-stationarity since there is no constant time scale of the data. Hence, all events are strictly dependent on their occurrences and must be defined by such time points. In practice, this means that both the amplitude of the signal and its frequency should be functions of time. Such an assumption is opposed to the case of a periodic data system, where the underlying signal can be described according to repeated or periodic time scales. Adaptivity is required to adjust the bases to local variations of the original signal due to the non-stationarity and non-linearity of the system. Hence, they will not just fulfil the mathematical definition requirements but will also capture the underlying physics of the process. However, non-linearity is of significant interest for this basis feature: in the case of Fourier analysis, such a property manifests as harmonic distortion, and the strength of non-linearity generates the distortion. Completeness is required to achieve a certain degree of

precision with respect to the expansion of the data; orthogonality provides positivity of energy and excludes leakage of it across spurious bases. Consequently, even with infinite numbers of such bases, predetermined bases will not fit all the existing phenomena with a universal, constant representation. Indeed, generating an efficient mechanism to extract bases derived from the data provides the most efficient solution.

An essential attribute of the EMD basis expansion approach pertains to the notion of instantaneous frequency (IF), which is a feature that has generated significant discussion in the literature; see Cohen (1995), Gabor (1946), Boashash (1992*a*). It requires particular attention, since it could carry great discriminatory power within different statistical tasks. Overall, the concept of frequency takes its definition from mechanics when a vibratory motion comes into play and is associated with a vibrating body. The vibrating body fulfils a complete oscillation by moving from the equilibrium position to one end of the path, then to the other end of the path and back to the equilibrium position. Through this model, the frequency can be defined for any vibratory motion. Amongst others, the harmonic motion is of particular interest given its widespread use in several applications (solid bodies, atmosphere, etc.) due to its ability to describe the motion of a particle at any fixed point. For example, the frequency ω of a wave motion is defined as the number of waves that pass by any fixed point per unit time. Equivalently, the frequency ω of electric current in a circuit corresponds to the number of cycles per unit time. Therefore, if the intention is to compute spectral decomposition and the signal $s(t)$ is a weighted sum of harmonic vibrations, then the Fourier Transform (FT) would achieve this task. The spectrum $S(\omega)$ obtained can be computed at any time t, and it will be meaningful if the underlying signal $s(t)$ is stationary. Indeed, any stationary signal can be represented as the weighted sum of sine and cosine waves with specific frequencies, amplitudes and phases, constant in time. Therefore, the concept of frequency per se is unambiguous. Nevertheless, since it defines the number of cycles during one unit of time of a body in periodic motion, there seems to be an apparent paradox in associating the word "frequency" with the word 'instantaneous'.

There are two main difficulties in defining the IF. First, its definition relies on sine or cosine basis functions assuming constant amplitude, phase, and period over the whole data set, accounting for global frequency content concepts. In its standard form, it cannot be adaptive to capture local phase and period variations often encountered with non-stationary signals. To achieve this with Fourier bases, one would need to accommodate a time-varying coefficient model. This can be problematic, as one would need a potentially infinite number of functional coefficients for the potentially infinite number of bases required for non-stationary and non-linear signals. This poses a significant challenge when developing a model, as it becomes infeasible to work with a large, possibly infinite number, of infinite-dimensional functional coefficients. The second issue is the lack of a unique definition. Cohen (1995) describes the central paradoxes associated with the instantaneous frequency. These are essential difficulties inherent to such

a notion when defined as the derivative of the phase function and are partly resolved through the Hilbert Transform. The five paradoxes are given as follows: (1) the instantaneous frequency may not be one of the frequencies present in the spectrum. (2) If the spectrum consists of a line spectrum characterised by only a few sharp frequencies, then the instantaneous frequency may be continuous and range over an infinite number of values. (3) Even if the spectrum of the analytic signal is zero for negative frequencies, then the instantaneous frequency may be negative. (4) Even in the presence of a band-limited signal, the instantaneous frequency may go outside the band. (5) If the instantaneous frequency is an index of the frequencies which exist at time t, the direct intuition for this is that the only information concerning it is the one at present, hence at time t. However, to calculate the analytic signal at time t, the signal has to be known at all times. Therefore, even with its local nature, it may be essential to recognise the non-locality of this concept and find a different way to define it. The reader might refer to Boashash (1992*a*), Boashash (1992*b*) for a review of the definition of such a concept. In this thesis, the IF is employed to investigate its statistical properties in solving classification problems.

Several solutions have been proposed in the literature studying a generalisation of the IF suitable to non-stationarity (see Gabor (1946), Boashash (1992*a*)). A central and relevant step was the one given by Gabor (1946), who proposed a method for generating a unique complex signal from the real one. This method first finds the FT of the real signal $s(t)$ and then suppresses the amplitudes belonging to negative frequencies and multiplies the amplitudes of positive frequencies by two. By doing so, most of the issues introduced above can be alleviated. This method is equivalent to the following time-frequency procedure:

$$z(t) = s(t) + \jmath\, \mathcal{H}\left[s(t)\right] = a(t)e^{\jmath\theta(t)} \tag{1.2}$$

where $z(t)$ is usually referred to as the Gabor's complex signal or analytical signal, $s(t)$ is the real signal, $a(t)$ and $\theta(t)$ are the instantaneous amplitude and the instantaneous phase respectively and given as

$$\begin{aligned} a(t) &= \left[s(t)^2 + \mathcal{H}\left[s(t)\right]^2\right]^{1/2} \\ \theta(t) &= \arctan\left(\frac{\mathcal{H}\left[s(t)\right]}{s(t)}\right) \end{aligned} \tag{1.3}$$

and $\mathcal{H}[\cdot]$ is the Hilbert Transform (HT) defined as

$$\mathcal{H}\left[s(t)\right] = p.v. \int_{-\infty}^{+\infty} \frac{s(t-\tau)}{\pi\tau} d\tau \tag{1.4}$$

where p.v. denotes the Cauchy Principal value of the integral. The definition for the complex signal $z(t)$ allowed Gabor to define a one-to-one correspondence between $s(t)$ and $z(t)$, whose modulus and argument are given by the pair $[a(t), \theta(t)]$. Proof of such a fact is provided in Boashash (2015). Furthermore, the central moments of the frequency of the signal through the complex spectrum

$Z(\omega)$ can be defined as

$$\mathbb{E}\left[\omega^n\right] = \frac{\int_{-\infty}^{+\infty} f^n \left|Z(\omega)\right|^2 df}{\int_{-\infty}^{+\infty} \left|Z(\omega)\right|^2 df} \tag{1.5}$$

If the spectrum of the real signal, $S(\omega)$, was used instead in the above equation, all odd moments would be zero, since $|S(\omega)|^2$ is even and this would not fit the physical reality. This is because the obtained frequencies will only be negative, and such a quantity, in order to carry physical meaning describing a real world phenomenon, is required to be positive. Hence, utilising the complex spectrum instead allows suppression of the negative frequency part and retention of only the positive ones. Following on from Gabor's work, Ville (1958) defined the IF of a signal given as $s(t) = a(t)\cos\theta(t)$ whereby,

$$\omega_i(t) = \frac{1}{2\pi}\frac{d}{dt}\left[\arg z(t)\right] = \frac{1}{2\pi}\frac{d\theta(t)}{dt} \tag{1.6}$$

He also showed that the average frequency in a spectrum of such a signal is equal to the time average of the IF:

$$\mathbb{E}\left[\omega\right] = \mathbb{E}\left[\omega_i\right] \tag{1.7}$$

where

$$\mathbb{E}\left[\omega\right] = \frac{\int_{-\infty}^{+\infty} \omega \left|Z(\omega)\right|^2 d\omega}{\int_{-\infty}^{+\infty} \left|Z(\omega)\right|^2 d\omega} \tag{1.8}$$

and

$$\mathbb{E}\left[\omega_i\right] = \frac{\int_{-\infty}^{+\infty} \omega_i(t) \left|z(t)\right|^2 dt}{\int_{-\infty}^{+\infty} \left|z(t)\right|^2 dt} \tag{1.9}$$

Through these results, Ville then formulated a distribution of the signal in time-frequency which is nowadays commonly referred to as the Wigner-Ville Distribution (WVD) and is given as

$$\mathcal{W}[t,\omega] = \int_{-\infty}^{+\infty} z(t+\tau/2)\, z^\star(t-\tau/2) e^{-\jmath 2\pi\omega\tau} d\tau \tag{1.10}$$

where $z^\star$ represents the complex conjugate of z and $\mathcal{W}[t,\omega]$ practically corresponds to the FT of the product $z(t+\tau/2)\, z^\star(t-\tau/2)$ with respect to τ and is evaluated through Fast Fourier Transform algorithms. Ville (1958) showed that the first moment of the WVD with respect to the frequency leads to the IF

$$\omega_i(t) = \frac{\int_{-\infty}^{+\infty} \omega\, \mathcal{W}[t,\omega] d\omega}{\int_{-\infty}^{+\infty} \mathcal{W}[t,\omega] d\omega} \tag{1.11}$$

Therefore, Ville and Gabor showed that, with the introduction of the Hilbert transform, the following could be obtained: (1) the definition of a complex signal $z(t)$ whose spectrum is identical to that of the real signal, $s(t)$, for positive

frequencies and zero for the negative frequencies; (2) the definition of both amplitude and phase of a signal can be derived unambiguously, allowing the derivation of an expression for the instantaneous frequency.

Gabor defined, through the Hilbert transform, his complex signal as $z(t) = s(t) + \jmath\mathcal{H}[s(t)]$, known as Gabor's analytical signal, where $s(t)$ corresponds to the real part and $\mathcal{H}[s(t)]$ to the imaginary part. The two signals $s(t)$ and $\mathcal{H}[s(t)]$ are said to be in quadrature, since they are out of phase of $\pi/2$. In practice and under certain conditions, it does not always generate a signal plus its imaginary quadrature component (see Gabor (1946)). This is because the HT operation preserves the positive frequency domain of the spectrum and inverts the sign of the spectrum in the negative frequency domain. Thus, it does not simply transform the cosine term into a sine term. If there is any significant leakage of the positive spectral components into the negative region, then the HT will not yield to the quadrature component of $s(t)$. The requirement in this framework, so that the HT signal $\mathcal{H}[s(t)]$ will be the quadrature of the input $s(t)$, is enclosed by Bedrosian's product theorem (BPT) (Boashash (1992*a*)). In practice, this theorem states that, if there is a modulated signal of the form $a(t)\cos\theta(t)$, where physical meaning is attached to the amplitude $a(t)$ and the phase $\theta(t)$, and if the spectra of $a(t)$ and $\theta(t)$ are not separated in frequency, then the HT will be a result of overlapping and phase-distorted functions. Hence, the desired situation is when the spectra of $a(t)$ and $\cos\theta(t)$ are separated in frequency, providing then that the amplitude $a(t)$ and phase $\theta(t)$ are considered independently. Such a fact is manifests when a signal approaches a "narrow band" condition and, in that case, the Hilbert Transformed signal approximates the quadrature signal.

Furthermore, in several areas, a signal is often referred to as a "multicomponent signal", meaning a signal that is characterised by many intrinsic frequency components, each carrying its specific instantaneous frequency. Nevertheless, equation (1.6) can only express a signal IF value at a given time, rather than a collection of them, which is what is required to capture the notion of a multicomponent signal. Hence, there will be one frequency value at any given time. This led Cohen (1995) to introduce the term "monocomponent function". There is no clear definition of such a term, and the adoption of a narrow band signal provided by Schwartz et al. (1996) is employed as a limitation for the data so that the instantaneous frequency makes sense. Beyond the definition of a narrow band signal, what is also required at this point is the separation of these many intrinsic frequencies to obtain a meaningful, well-defined instantaneous frequency. One solution to this problem is making use of the Empirical Mode Decomposition, which decomposes the signal into narrow band components by empirically defining the physical time scales intrinsic to the data.

There are two definitions of bandwidth. One is related to the probability properties of signals and waves, in which processes are often assumed as Gaussian and stationary and can be defined in terms of spectral moments, while the second refers to the moments of the spectrum. To introduce the former definition, let us first define $N_0 = \frac{1}{\pi}\left(\frac{m_2}{m_0}\right)^{1/2}$ as the expected number of zero-crossings per

unit time and $N_1 = \frac{1}{\pi}\left(\frac{m_4}{m_2}\right)^{1/2}$ as the expected number of extrema per unit time, with m_i corresponding to the ith moment of the spectrum. A classic bandwidth measure (see Rice (1944), Rice (1945)) is often given by the parameter ν, which is given as

$$N_1^2 - N_0^2 = \frac{1}{\pi^2}\frac{m_4 m_0 - m_2^2}{m_2 m_0} = \frac{1}{\pi^2}\nu^2$$

For a narrow band signal $\nu = 0$, since the expected numbers of extrema and zero crossings have to be equal. The second definition of narrow band is again related to the moments of the spectrum but in a more general way. Consider a complex valued function in polar coordinates as $z(t) = a(t)e^{\jmath\theta(t)}$ with both $a(t)$ and $\theta(t)$ being functions of time. If this function has a spectrum, $S(\omega)$, the mean frequency is given as $\mathbb{E}[\omega] = \int \omega |S(\omega)|^2 d\omega$, which can also be expressed as

$$\begin{aligned}\mathbb{E}[\omega] &= \int z^*(t)\frac{1}{\jmath}\frac{d}{dt}z(t)dt \\ &= \int \left(\frac{d\theta(t)}{dt} - \jmath\frac{da(t)/dt}{a(t)}\right)a^2(t)dt \\ &= \int \frac{d\theta(t)}{dt}a^2(t)dt\end{aligned}$$

By following Cohen (1995), the definition of instantaneous frequency can be enclosed by $\frac{d\theta(t)}{dt}$. According to these definitions, the bandwidth is given as

$$\begin{aligned}\nu^2 &= \frac{(\omega - \mathbb{E}[\omega])^2}{\mathbb{E}[\omega]^2} = \frac{1}{\mathbb{E}[\omega]^2}\int (\omega - \mathbb{E}[\omega])^2 |S(\omega)|^2 d\omega \\ &= \frac{1}{\mathbb{E}[\omega]^2}\int z^*(t)\left(\frac{1}{\jmath}\frac{d}{dt} - \mathbb{E}[\omega]\right)^2 z(t)dt \\ &= \frac{1}{\mathbb{E}[\omega]^2}\left[\int \left(\frac{da(t)}{dt}\right)^2 dt + \int \left(\frac{d\theta(t)}{dt} - \mathbb{E}[\omega]\right)^2 a^2(t)dt\right]\end{aligned}$$

In order to obtain a narrow band signal, the above quantity needs to be small, and therefore $a(t)$ and $\theta(t)$ have to be gradually varying functions. Both definitions of the bandwidth are still given in a global sense, while, to capture the concept of instantaneous frequency, a local definition is required instead. Other solutions have been considered to tackle such a problem, as in Melville (1983), who made use of filtering. However, in non-stationary and non-linear systems, spurious harmonics are present, and filtering cannot work. In practice, for any function having a meaningful instantaneous frequency, the real part of its Fourier transform has to have only positive frequencies (see Gabor (1946), Titchmarsh (1948), Bedrosian (1963), Boashash (1992*a*)). Such a restriction is still global and has to be translated to a local equivalent. Huang et al. (1998) show in their work, with a basic example of a simple sine signal, that the instantaneous frequency can be defined if the given function is restricted to be symmetric locally with respect to its zero mean level. This led Huang et al. (1998) to define a new class

of functions named Intrinsic Mode Functions (IMFs), carrying precisely such a local property, and for which the IF could be derived. What is more, Huang et al. (1998) introduced a decomposition technique, i.e., the Empirical Mode Decomposition, which decomposes the data into components whose instantaneous frequencies can be derived. The obtained functions are the basis functions and are, indeed, the IMFs.

Once the necessary conditions for the instantaneous frequency to exist are taken into account, the definition of the basis can be given. Formally, the proposed class of functions, the Intrinsic Mode Functions, are defined by Huang et al. (1998) to meet the following conditions: (1) the number of extrema and the number of zero crossings must either equal or differ at most by one within the whole data set; (2) the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero at any point. The importance of the two conditions can be interpreted as follows: while the first one is related to the classical assumption of narrow band for a stationary Gaussian Process, the second one has been made so as to reduce to a local restriction necessary to determine the instantaneous frequency and avoid asymmetric waves affecting it. By doing so, the local mean forcing local symmetry replaces the local time scale that cannot be defined in the case of non-stationary data. The name intrinsic mode functions was given by Huang et al. (1998) to represent the oscillation modes characterising the data. Within each cycle provided by the zero-crossings, a basis represents one single mode of oscillation. It can, indeed, be a non-stationary basis. Once the Hilbert transform is applied, each IMF can be written in the form of an analytic signal $z(t)$. By performing a Fourier transform on $z(t)$ as follows

$$\mathcal{F}[z(t)] = \int_{-\infty}^{\infty} a(t) e^{\jmath(\theta(t)-\omega t)} dt$$

then, by the stationary phase method (see Copson and Copson (2004)), the maximum contribution to $\mathcal{F}[z(t)]$ is given by the frequency satisfying $\frac{d}{dt}(\theta(t) - \omega t)$ from which it follows $\omega = \frac{d\theta(t)}{dt}$. Such definition agrees with the definition of frequency for the classic wave theory (Whitham (2011)) and, more importantly, with the best-fit sinusoidal function locally.

Beyond the definitions of instantaneous frequency and a basis for determining it, the decomposition of the data has to be constructed. The decomposition is achieved with three main assumptions highlighted in Huang et al. (1998): (1) the time-series has to have at least two extrema, one maximum and one minimum. (2) The time-lapse between extrema defines the time scale. (3) If there were no extrema present in the data, then it can be differentiated to reveal them. The central point of the EMD is, indeed, extracting oscillatory modes of a time-series according to its intrinsic time scales by exploiting the a posteriori basis. The choice in identifying the different scales falls on extrema, rather than on inflexions, since a better resolution of the oscillatory modes can be obtained that can, moreover, also be applied to non-zero mean data. The process to extract the IMFs is called a sifting procedure.

The EMD provides a set of basis functions and a residue which represents the single convexity tendency function, akin to a trend or a constant characterising the original time-series. The only ingredient required by the technique is the location of the extrema. The zero-mean reference is obtained through the sifting process and this is one of the advantages of the EMD, since it removes the issue of the mean values produced by a significant DC component of the data with non-zero mean. The critical point is that the basis usually carries a physical meaning representative of the scales intrinsic in the original process. However, this is not always the case, since there are certain phenomena which are intermittent by nature and, therefore, the decomposition would be affected: a single IMF would detect two different scales and, to interpret the decomposition, the whole set of bases would be necessary. Nevertheless, the gain of this technique versus other basis expansions, such as the Fourier one, is the physical interpretation implied by the EMD, usually lacking in the other approaches.

As previously mentioned, two main properties needed for the basis are completeness and orthogonality. The former is satisfied both theoretically and practically by the EMD; a time-series $s(t)$ can be completely represented according to such decomposition as $s(t) = \sum_{i=i}^{L} \gamma_i(t) + r(t)$, where $\gamma_i(t)$ is an IMF basis for a finite number of K IMF basis functions indexed by $i = 1, ..., L$ and $r(t)$ a residual tendency term. Each basis is ordered in the number of oscillations compared to the previous. Regarding orthogonality, the EMD provides it only at a practical level. Huang et al. (1998) defined an index of orthogonality to detect it a posteriori. However, even by using such a criterion, some small leakage can be found. This is true also in the case of pure sinusoidal components carrying different frequencies, given observed data of finite length. In general, orthogonality is a property which can be associated only with linear decomposition systems and does not make physical sense for a nonlinear decomposition, as with the EMD.

Having extracted all the IMFs of the original signal allows computing the whole set of instantaneous frequencies through the application of the Hilbert transform on each basis. As a result, the observed time-series can be represented as the sum of different functions expressed as a generalised Fourier expansion since amplitude and frequency are time-varying. This set of components denoting the frequencies as a function of time is defined as the Hilbert spectrum and given as $H(f, \omega)$. It provides a three-dimensional plot where the x-axis is the time, the y-axis is the frequency and the z-axis the energy. The interpretation of this spectrum is entirely different to the one of classical Fourier analysis: in the latter one, the presence of energy at a certain frequency ω means that a sine or cosine function basis is persistent within the whole set of data. In the former one, instead, the presence of energy at a frequency ω represents that such specific wave has a higher likelihood to have appeared locally. Therefore, the EMD and the Hilbert spectrum together provide a probabilistic spectrum for non-stationary data.

1.2.2 Forecasting Techniques with the EMD

Forecasting time-series represents a core activity in scientific research. Different methodologies associated with various aspects can be considered, depending on factors like the area of interest, the size of the dataset or the data features. In multiple applications, such as wind speed, financial time-series, earthquake data, geophysical research, and many others, the original time-series behaviour results from multiple underlying time-series whose information content relates to different time scales. For example, the financial share price can often be decomposed into high and low frequency contents deriving from discontinuities or ruptures, versus long-term trends in the original data. Time-frequency methods and, among these, also the EMD, have been widely employed to solve this kind of task. Awajan et al. (2019) proposed an extensive review of the EMD forecasting methods developed within many different applications. An equivalent table to the one provided by Awajan et al. (2019) is represented below:

Cite	Year	Method	Data Category
Li and Wang (2008)	2008	EMD-ARIMA	Wind Speed
Lin et al. (2012)	2012	EMD-LSSVR	Exchange Rate
Zheng et al. (2013)	2012	EMD-RBFNN	Wind power
An et al. (2013)	2013	EMD, FFNN	Electricity demand
Okolobah and Ismail (2013)	2013	EMD, ANFIS	Peak Load
Abadan and Shabri (2014)	2014	EMD-ARIMA	Prices of rice
Kisi et al. (2014)	2014	EMD-ANN	River stage
Abadan et al. (2015)	2015	EMD-ARIMA	Exchange rates
Duan et al. (2015)	2015	AR-EMD-SVR	Ship motion
Ismail et al. (2015)	2015	EMD, LSSVM	River Flow
Duan et al. (2016)	2016	EMD-AR	Ocean waves
Yang and Lin (2016)	2016	EMD, SVR, ARIMA	Stock market
Zhu et al. (2017)	2017	EMD, LS-SVM	Carbon Price
Yahya et al. (2017)	2017	EMD, ANN	Tourism
Ismail and Shabri (2017)	2017	EMD-SVM	River flow
Tao et al. (2017)	2017	iEEMD, ARIMA, ELM, PF	Hog price
Zhao et al. (2017)	2017	EEMD-ARIMA	Occupancy of hotels
Bedi and Toshniwal (2018)	2018	EMD-based deep learning	Electricity Demand
Sun and Wang (2018)	2018	FFEMD	Wind Speed
Büyükşahin and Ertekin (2019)	2019	EMD-ARIMA-ANN	Sunspot
Zhang and Hong (2019)	2019	CEEMDAN-SVRQDA	Electric load
Xia and Wang (2020)	2020	EMD-PSOLSSVM	Energy consumption structure
Dai and Zhu (2020)	2020	SOPEEMD	Stock market returns

Table 1.1: Forecasting method using EMD or its variations in combination with other techniques. Note that the abbreviations for the proposed model methods are provided. The reader might refer to the actual references for further details.

The most relevant papers were selected. As shown, several studies and different advancements have been made to exploit EMD, or variations of it, to achieve more powerful forecasting performances. The ordinary procedure of most of the above-given forecasting techniques involving EMD goes as follows: the time-series is decomposed, and the IMFs are extracted; the forecasting method of interest is applied to each IMF; finally, the forecasted IMFs (or their combinations) are combined according to different techniques. At this stage, it is essential to comprehend that EMD fully applies to the deterministic pathwise realisation of a time-series. Therefore, its combination with any other procedure would still

carry this feature without accounting for any stochasticity or randomness unavoidably induced by the original time-series process. The ideal solution should first tackle the unique characterisation of each stochastic process of the IMFs. Afterwards, the convolution of the stochastic processes of the IMFs needs to be derived and examined. Through this approach, a valid and more powerful forecasting technique is achieved.

1.2.3 Alternative Adaptive Decomposition Techniques

In general, spectral analysis has the final aim of estimating the global power-frequency spectrum distribution of a given random process. Many applications require such a tool, which must not rely on classical-method assumptions such as stationarity of the underlying signal. The need is set for a time-frequency representation indicating how the power spectrum changes over time. The use of the concept of instantaneous frequency is central in this discussion. However, the definition of a meaningful instantaneous frequency is a difficult task, strongly affecting the research community.

A standard solution commonly adopted to handle the above problem has been to perform bandpass filtering of the signal and then apply the Hilber transform to extract the instantaneous frequency for each passband of interest (Freeman (2004), Liang et al. (2005)). However, the choice of passbands is heuristic and, therefore, the instantaneous frequencies obtained are difficult to interpret.

The alternative decomposition method known as the singular spectrum analysis (SSA or "Caterpillar" SSA) corresponds to a principal component analysis based on nonparametric spectral estimation (Vautard and Ghil (1989), Vautard et al. (1992), Ghil et al. (2002)). SSA windows the time-series and stores the windows within the columns of a matrix, commonly referred to as a trajectory matrix. The second step performs the singular value decomposition (SVD) of the trajectory matrix and represents it as a sum of rank-one bi-orthogonal elementary matrices. Afterwards, the elementary matrices are split into several groups and summed up together within each group. Lastly, the diagonal averaging of the new aggregated matrices is undertaken, producing a set of time-series, equivalent to additive components of the initial time-series. The derived SSA components are data-adaptive. As a result, in contrast with classical Fourier components, they can capture non-harmonic oscillations of the underlying time-series highly prevalent in non-linear and non-stationary series. However, several drawbacks are intrinsic to the procedure itself. First, the window length, also known as embedding dimension, must be chosen appropriately, since the reconstructed components strongly depend on it. Second, as in most applications employing the SVD, the required principal components used in reconstructing a specific SSA component time-series must be meticulously chosen to carry physical meaning associated with the studied phenomenon. This is achieved when the frequency content of the component is narrow-banded.

Reasons for introducing the SSA lie in a recent method called Singular Spectrum

Decomposition (SSD) (Bonizzi et al. (2012), Bonizzi et al. (2014)) that builds upon the SSA and provides a solution to the introduced problem. The advantages of SSD compared to the standard SSA are: (1) the entirely data-driven choice of the embedding dimension set to obtain a well-defined frequency band in the spectrum of the original signal. (2) The automated selection of the principal components to reconstruct a specific component signal that minimises the generation of spurious components. Like the EMD, the SSD corresponds to a fully data-driven decomposition technique based on the extraction of the energy associated with different intrinsic time scales. This result is attained by defining a new trajectory matrix used in the SSD method, guaranteeing a decrease of energy characterising the residual within an iterative approach.

The main objective of these decomposition techniques is to separate the multiple intrinsic frequencies proper of the original signal. This is done within both the SSD and the EMD by using an iteration scheme that, according to different properties of the signal, extracts monocomponent functions carrying energy in a decreasing fashion. The SSD was born as an alternative method to the SSA, addressing some of the main drawbacks of the EMD. In practice, and often in cases such as signal intermittency, the EMD components can be affected by mode mixing. This means that a single EMD basis carries different time scales, reflecting different intrinsic frequencies of the original signal, making physical meaning unclear. Hence, by transforming the SSA into an automated data-driven decomposition method, SSD should provide an interesting alternative to EMD.

Another procedure that should be taken into account in this context is the work of Daubechies et al. (2011) (and reference within). The authors propose the application of the wavelet synchrosqueezed transform to the EMD basis functions. This time-frequency transform reassigns signal energy in frequency, compensating for the spreading effects caused by the mother wavelet. Unlike other time-frequency reassignment methods (Hainsworth and Macleod (2003)), synchrosqueezing only allows reassignment in the frequency direction by preserving the time resolution. As a result, the inverse synchrosqueezing algorithm can reconstruct an accurate representation of the original signal. The central motivation for introducing such a method is that, like the direction of this thesis, Daubechies et al. (2011) aims to find a more robust representation of an IMF basis function by employing a particular time-frequency reassignment method. However, the reader should bear in mind that the application of synchrosqueezing requires the given signal to be an intrinsic mode type (IMT) function, which is formally introduced as a continuous function (for further details, see Daubechies et al. (2011)) carrying a unique frequency component. The IMT functions are defined as the set of IMT bases well-separated in the frequency domain up to a certain level of accuracy. Hence, it corresponds to a subspace of $L^{\infty}(\mathbb{R})$ and does not correspond to a vector space. As a result, and similarly to the EMD, the obtained IMT basis functions might not be well separated, and this approach would not work efficiently in the presence of highly non-stationary signals.

This alternative method further highlights the need for a complete data-adaptive decomposition technique dealing with the non-stationary data characteristics expressed in both domains, the time and the frequency. Hence, this further supports the proposed stochastic models, representing the EMD as a novel stochastic decomposition method.

1.2.4 Gaussian Processes, Kernel Methods and Multi-kernel Techniques

Over the past decade, Gaussian processes (GPs) have become the leading research technique for both classification and regression tasks in machine learning. Such a tool was initially proposed under the name kriging in the geostatistical literature (Cressie (2015)) and represents a non-linear nonparametric technique that has been used in multiple applications and whose literature has been widely developed (Rasmussen and Williams (2005), MacKay (1997), Ripley (2007)). In a nutshell, a Gaussian process generalises the Gaussian probability distribution and, while a probability distribution describes a random variable, a stochastic process controls functions instead. The great advantage of this inference technique lies in the wide range of data properties that can be easily reproduced, like smoothness, periodicity, scalability, etc., that are entirely controlled by the positive definite covariance kernel function. This operator determines the similarity between pair of points in the domain of the random function. Given this role, the covariance structure significantly affects the performance of a Gaussian process on the task of interest, and extra care must be taken when this is chosen, in order to unveil hidden data patterns. A common assumption made in this setting is to consider GPs that carry a zero-mean function. In such a case, the learning problem to infer a Gaussian process reduces to the learning problem of its kernel function hyperparameters.

Several choices could be considered for the covariance function structure, coming from different kernel methods branches. It could have a traditional stationary structure (Rasmussen and Williams (2005)) or might be derived from its spectral representation (Wilson and Adams (2013)) or could be a parametric or nonparametric kernel (Abbasnejad et al. (2012)). In this thesis, a review of kernel methods is provided in Chapter 4. A recent approach that became highly popular in the machine learning community is one of multiple kernel learning (Gönen and Alpaydın (2011*a*)). The idea behind such a notion is that more sophisticated covariance structures can be achieved by composing together a few standard kernel functions. This thesis applies this intuition and combines it with the Gaussian process framework to propose a stochastic embedding of the IMF basis functions. The proposed model foresees each IMF distributed according to a Gaussian process, whose convolution will reproduce the Gaussian process of the original signal. Specifically, both the original signal stochastic process mean and kernel functions will be equivalent to the sum of the mean functions and the sum of the kernel functions of the IMFs' stochastic processes, respectively. The

idea of such a model relies on the fact that the convolution of Gaussian processes will produce a Gaussian process and an additive structure for both mean and kernel functions of the stochastic process of the original signal.

This intuition will be further extended to define an alternative stochastic embedding of the EMD bases, which proposes the construction of new "band-limited" bases derived from the instantaneous frequencies of the original IMFs. The critical issue tackled with this model is the phenomenon known as mode-mixing, often encountered when the EMD sifting procedure is applied to a signal and does not identify basis functions that carry a unique frequency mode. Hence, a single IMF might carry multiple frequency components. The idea developed with this second stochastic embedding aims to determine an adaptive, optimal partition of the obtained Hilbert spectrum and regroup the IMFs' sample points according to the location of their instantaneous frequencies' sample points. Hence, if mode-mixing is present, the IMFs whose IFs belong to the same frequency partition will be aggregated, and a new band-limited IMF will be defined.
The central component required to achieve such stochastic embedding is a partition of the Hilbert spectrum that is a priori unknown. This is constructed through an optimisation method known as the cross-entropy method and presented in the subsection below.

1.2.5 The Cross-Entropy Method

The construction of the second stochastic embedding proposed in this thesis requires the definition of an optimal partition of the instantaneous frequencies domain, which is achieved through the optimisation method known as the cross-entropy method. The idea is that IF sample points derived from the original IMF bases functions might fall within the same frequency regions and hence should be modelled accordingly, since they capture an equivalent oscillating mode of the original signal. As a result, a new set of Quasi-IMF (QIMF) bases will be defined, called "band-limited" IMFs, by aggregating the original IMFs according to the location of their IFs within the regions of the computed partition. The intuition behind such a partition model is to characterise particular adaptive local bandwidths of the IMFs' frequency domain with different kernel classes (stationary, non-stationary, etc.) in a Gaussian process setting, rather than try to formally define the stochastic process of the instantaneous frequencies, which may be much more involved. The formulation of this model is presented in Chapter 6. The significant aspect to consider is that the desired partition of the IFs domain is unknown a priori, and for its formulation and derivation, a stochastic optimisation method is employed in this work. These two are presented in Chapter 7.

The formulation of the problem requires that a partitioning rule of the instantaneous frequency domain derived from the IFs sample points and an irregular sampling that best captures each frequency region must be jointly chosen. The algorithm selected to solve such a problem relies on a cross-entropy solution and

is presented in Chapter 7. The final goal is to identify a partition as close as possible to a uniform dispersal of signal energy per IMF to the sets of spectral partitions over time and frequency. Note that this is only one possible framework; different criteria could be considered, instead. This would imply a well-behaved obtained EMD decomposition, capturing all the existing oscillating modes of the original signal. The constructed procedure provides the partition by relying on the entropy measure; specifically, the Kullback–Leibler divergence will be used. Further explanations are given in Chapter 7.

Several everyday tasks in operations require the use of optimisation problems. Examples could be the travelling salesman problem, quadratic assignment problem, and max-cut problem; all these correspond to combinatorial optimisation problems (COPs), which are entirely known and static. One solution tackling these kinds of settings is discrete event simulation so as to estimate an unknown objective function. The cross-entropy method was first introduced by Rubinstein and Kroese (2004) through an adaptive algorithm for estimating probabilities of rare events in complex stochastic networks. It was soon revealed that it was also highly efficient in solving hard COPs (Rubinstein (2001), Rubinstein (1999)). This is achieved via a translation of the deterministic optimisation problem into a related stochastic optimisation one and then using a rare event simulation technique, as shown in Rubinstein (1999). The procedure foresees two main stages: (i) generate a random data sample according to a specific mechanism; (ii) update the parameters of the random mechanism based on the data to produce a "better" sample in the next iteration. Its important gain is the definition of a precise mathematical framework for deriving fast and optimal learning rules based on advanced simulation theory. Alternatives could be simulated annealing (Aarts and Korst (1989)), tabu search (Glover and Laguna (1997)) or genetic algorithms (Goldberg and Holland (1988)).

The cross-entropy method has been successful when applied to both deterministic and stochastic COPs. In the second class, the objective function is random or needs to be estimated via simulation. There is an increasing interest in the cross-entropy method, with spacing of applications to buffer allocation (Alon et al. (2005)), or static simulation models (Homem-de Mello and Rubinstein (2002)), or neural computation (Dubin (2002)), or DNA sequence alignment (Keith and Kroese (2002)), vehicle routing (Chepuri and Homem-De-Mello (2005)) and many others. For further reference, the reader might use Rubinstein and Kroese (2004) and De Boer et al. (2005).

In this thesis, the cross-entropy method is employed to develop two different algorithms to construct the partition required to formulate the band-limited IMF basis functions. The two algorithms consider two sampling distributions; one continuous and one discrete, to compare alternative approaches of this novel technique constructing new bases derived from Empirical Mode Decomposition.

1.3 Research Questions and Outline of the Thesis

Based on the motivations and context previously discussed in Section 1.2 and Section 1.1, the following research objectives and questions have been developed and characterise the thesis research contributions. These will be presented from a macro through to a micro-level of detail as follows:

To fully exploit all the facets of Empirical Mode Decomposition framework, the first step that has to be taken corresponds to a statistical analysis of it with the final aim of understanding how to incorporate this method in a statistical modelling context. Such an investigation lacks in this literature and will be beneficial for the next set of research questions strictly related to how to construct the EMD basis or IMFs efficiently:

- Traditional time-frequency methods rely on the assumption that the underlying signal is linear and stationary. Furthermore, classical Fourier-like methods that cannot deal with non-stationarity often provide an infinite number of basis functions that cannot be used in practical applications. Moreover, the time-frequency resolution of these methods is subject to a trade-off for which either time or frequency cannot be efficiently specified without the associated cost of losing information in the other domain. Hence, the need for a method that is fully data-driven is highly required.

- How to robustly estimate and extract EMD functions to obtain a reliable basis if the underlying signal is subject to high levels of non-stationarity and non-linearity. Furthermore, hoow to formally define the EMD in a formal mathematica expression.

- In their work, Huang et al. (1998) highlighted that one area of future research needs to focus on the ideal class of functions used to represent the general IMF basis. Several choices are available, and one of the research questions of this thesis is showing the different representations available in the literature and how they might influence the decomposition. A further question is how to estimate the parameters defining this class of functions effectively and concisely. The algorithm that is used to extract each basis sifts the original signal several times until a certain stopping criterion is satisfied. The selected functional representation for the IMFs will optimise the number of sifting, and the number of IMFs identified since this is not unique.

- Another critical point in the extraction of the IMFs is the stopping criterion selected within the sifting procedure. Several options have been proposed, each affecting the final decomposition and, hence, the algorithm's convergence in finding an accurate IMF. The question concerns the selection of the best stopping criterion for the sifting procedure to produce the best IMF set. This means that the sifting procedure will not sift too many

times and lose some of the physical meaning of an IMF basis, and, contemporarily, it will not sift too few times without identifying a real IMF. A study of how the different stopping criteria influence the decomposition will be, therefore, provided.

- As highlighted by Huang (2014), the EMD is a "Reynolds type" decomposition, which means that it is used to extract variations from the data by separating its mean (expectation value), in this case, the local mean, from the fluctuations by spline fits. Therefore, a system of nested equations can be built, and each IMF is represented as a linear combination of the spline coefficients of the original signal. One of the resolved research questions in this thesis presents such a system. This further justifies the importance of the chosen representation of the original signal since it will be the one also expressing each IMF.

  A relevant point related to the instantaneous frequency and how to compute it in closed form. After having a selected a functional form for the representation of an IMF basis function, the Hilbert Transform of that representation is calculated. Challenges may arise in this case since the HT integral may not converge to a finite quantity and, consequently, the resulting instantaneous frequency will not exist and cannot be expressed in closed form. This work has been achieved in the literature by el Malek and Hanna (2020). One of the point raised in this thesis is to provide a stastical interpretation for it and how to use it within machine learning classification tasks.

The first part of the thesis addresses the above research questions and contributes to introducing the EMD to a statistical framework developed in the second part. Furthermore, it will provide a deep understanding of the EMD to then formulate an EMD feature library for non-stationary and non-linear signals whose insights will be precious to tackle different classification tasks.

The second set of research questions developed in the second part of the thesis tackles the problem that the EMD is a pathwise deterministic decomposition method often used to solve problems that are instead stochastic. The aim is to produce a stochastic embedding representation that deals with non-stationary processes. To attain such a goal, a Gaussian Process embedding of the EMD representation of the signal is performed by proposing a new set of models dealing with multiple assumptions in terms of the derived distributions for the IMF basis functions and the residual tendency. In this respect, the proposed method will offer a natural construction of a multi-kernel Gaussian Process for the stochastic process of the original signal with a covariance function corresponding to the sum of the covariance matrices of the individual IMFs. Chapter 4 will present kernel methods in order to study kernel functions available that could be used in this setting. This framework is developed in the second part of the thesis. The stochastic embedding will be presented in Chapter 6.

Chapter 5 will instead introduce a machine learning method called the Support Vector Machine. One of the objectives of this thesis is to test the produced EMD feature library above mentioned to solve different classification tasks affected by non-stationary signals and time-varying properties. In this way, these features can be interpreted at a statistical level and provide valuable insights for the application of interest.

Part of this second set of research questions will construct a second class of EMD stochastic embedding whose aim is to characterise the stochastic process of specific frequency bandwidths of the original signal by using the observed instantaneous frequencies. The formulation of such a model will be achieved by solving an optimisation method by computing an optimal partition of the time-frequency plane, which will compute the distance between an ideal distribution assumed for the frequency bandwidths and the observed instantaneous frequencies. The optimisation method employed is called the cross-entropy method and will be presented in Chapter 7.

The third part of this thesis deals with a set of research questions linked to the application: speech signals. The first application deals with Automatic Speaker Verification technologies and aims to differentiate utterances of a real voice versus a synthetic one. The EMD feature library derived will be used in this setting to solve such a classification task. Furthermore, the advantage of using such a non-stationary technique will provide interpretable features for the classification problem. The use of the SVM will allow for a multi-kernel framework empowering the classifier in dealing with non-stationary speech signals. Results show that the EMD combined with traditional speech analysis methods outperforms standard techniques. This part is presented in Chapter 8.

Chapter 9 deals instead with the problem of detecting Parkinson's disease through speech signals. This Chapter will test the developed stochastic embedding and combine it with a novelty use of the Fisher kernel, which will provide a data-adaptive fitting and testing procedure. Results show that the stochastic embedding strongly outperforms standard speech methods.

1.4 Glossary and notation

The following notation is used throughout:

$\mathbb{I}$ is the indicator function;
$s(t)$ discrete-path realisation signal; $\tilde{s}(t)$ continuous analog signal representation;
$S(t)$ the stochastic process of the signal $s(t)$;
$\tilde{S}(t)$ the stochastic process of the approximated signal $\tilde{s}(t)$;
$\mathbb{E}[\cdot]$the expectation operator;
$\mathbb{C}\text{ov}[\cdot,\cdot]$the covariance operator;
$\rho[\cdot,\cdot]$ is the autocorrelation function operator;
l corresponds to a realization of one IMF

L is the total number of convexity changes of the original signal;
$|\{ \quad \cdot \quad \}|$ represents the cardinality set;
$\gamma(t)$ represents the l-th Intrinsic Mode Function (IMF) of the EMD basis functions;
$r(t)$ residual obtained with the EMD;
$\omega_l(t)$ is the instantaneous frequency of the k-th IMF;
$\jmath$ complex unit;
$\tilde{\gamma}_l(t)$ analytic extension of $\gamma_l(t)$
$\mathcal{F}[\cdot]$ is the Fourier transform;
$S(f) = \mathcal{F}[s(t)]$ is the Fourier transform of the signal $s(t)$;
$\mathcal{HT}[\cdot]$ is the Hilbert transform;
$\mathrm{CWT}[a, b]$ is the continuous Wavelet transform;
$\psi(t)$ is the basic wavelet;
$\mathcal{W}[t, \omega]$ is the Wigner-Ville distribution;
$\mathrm{Re}(\cdot)$ and $\mathrm{Im}(\cdot)$ the real and the imaginary opeartor respectively;
f is the raw data sampling frequency (radians)
ω is the raw data sampling frequency (Hz)
$\breve{\omega}$ is the sampling frequency of the constructed IMFs (Hz) (according to Nyquist rule)
$\mathcal{H}$ represents the Hilbert space of transformed features; ϕ is the Mel-scale frequency
$H(\cdot)$ represents the Shannon entropy; Fourier basis or harmonic function or harmonic will be used with the same meaning;
$m(l)$ is a Mel Frequency Cepstral Coefficient;
$k(x_i, x_j)$ represents a kernel function;
$\mathbb{K}$ represents the Gram Matrix associated to a kernel function; $\langle \cdot, \cdot \rangle$ is the inner product;
$M(\theta, \tau)$ is the characteristic function;
$A(\theta, \tau)$ is the ambiguity function;
$z(t)$ is the Gabor's complex signal or analytical signal;
$a(t)$ is the instantaneous amplitude;
$\theta(t)$ are the instantaneous phase;
z^* is the complex conjugate;
$H(\omega, t)$ is the Hilbert Spectrum;

Part I

Time-Frequency Analysis Methods

Chapter 2

Time-Frequency Analysis

Data analysis plays a central role in both theoretical and applied research. Several methodologies have been developed in statistics to perform such a task within various contexts and application domains. The chosen method that is deemed most appropriate for a given context will depend on data features and assumptions the data science is willing to assume when developing a model. Many perspectives exist, and a large variety of benchmark models have been exploited. However, a specific area is looking at data that forms the focus of this research work and offers significant benefits in the time-series and stochastic process context: time-frequency analysis.

The time representation of a signal is fundamental to detect variations of the process over time, but to deeply understand its features, another representation also plays a central role: the frequency content. A rich literature that studies the characteristics and relationship between time-domain analysis and frequency-domain analysis of signals has been established over the last fifty years of research in statistics, signal processing and applied probability. More recently, there has been a renaissance of these ideas appearing in new contexts in the machine learning literature. As a result, the relationship between time and frequency of a signal has become central and provided many advantages. Different tools to switch from one domain to the other have been introduced and investigated.

Within his book, (Cohen, 1995) showed why time-frequency analysis is relevant by offering a list of examples of particular interest. Either domain by itself does not provide enough information about the existence of frequencies and the time at which they happened. This is true for the spectrum representation, where it is possible to know which frequencies existed, but the time location cannot be detected. Spectra change over time for two main reasons. Firstly, frequencies come from physical phenomena that may differ over time. Secondly, the propagation of waves within a medium depends on the signal frequencies. As a result, the detection of when they existed is particularly relevant for researchers and scientists. Thus, a joint time-frequency distribution is crucial to determine what is happening with respect to the analysed phenomenon: locating the frequencies, how long they last and their intensities.

The first one introducing the concept of frequency representation was Fourier, with the final aim of analysing the heat equation. The superimposition of a frequency over another, resulting in a signal expressed as a sum of sinusoidal waves called harmonics, is likely one of the most critical mathematical definitions (1807). Several variations were built (Fast Fourier Transform, Short Fourier Transform, the spectrogram, etc.), and similar methods were presented, such as the Wigner distribution (later become the Wigner-Ville distribution), the Wavelet analysis, the Evolutionary Spectrum. The idea beyond all these transforms is to identify a basis that is an a priori basis able to deal with intrinsic data properties (stationary, non-stationary, linear, nonlinear, etc.).

The idea of this Chapter is to introduce the fundamental techniques of time-frequency and their power developed in data analysis through their definitions, properties and primary use. The first part of the Chapter reviews some of the statistical data properties and definitions usually sought (or checked) when real datasets come into play. Afterwards, the time-frequency methods are classified into main categories: stationary and non-stationary. In the stationary ones, the one presented is the Fourier Analysis. Several mathematical properties are introduced since the reader might benefit in understanding the central concepts behind such a technique and how this differs from the EMD introduced later in Chapter 3. The non-stationary time-frequency methods presented are the Short Time Fourier Transform, the Wavelet Transform and the Wigner-Ville distribution. Last, a discussion on the time-frequency resolutions of the discussed transforms is provided. This is highly central to motivating the need for the research work conducted in this thesis.

2.1 Statistical Data Properties

This section reviews some core definitions of attributes that characterise phenomena often observed in real datasets. The purpose is to explain the main concepts required to understand and compare the reviewed time-frequency methods in this thesis. Most of the proposed methodologies can only deal with non-stationarity or non-linearity of the data system, but not with both contemporaneously. Therefore, the concept of stationarity, linearity and non-stationarity are presented. The assumption of stationarity is a common assumption of many statistical tools, and therefore even non-stationary processes tend to be transformed into stationary ones. Of course, this assumes the type of non-stationarity is sufficiently simple that pre-determined simple transformations may remove it. By being a property of stochastic processes, lets first consider the definition of a stochastic process:

Definition 2.1.1 (Stochastic process)**.** *A stochastic process* $\{..., S(1), S(2), ..., S(t), S(t+1), ...\} = \{S(t)\}_{t=-\infty}^{\infty}$ *is a sequence of random variables indexed by time t.*

In most applications the time index is a regularly spaced index representing calendar time. Within time series analysis, the ordering imposed by this index

is relevant to capture possible temporal relationships. The time index can either be discrete ($t \in \mathbb{N}$) or continuous ($t \in \mathbb{R}$). In the latter case the sequence corresponds to an uncountable infinite number of random variables. A realisation of a stochastic process with T observations in the sequence of observed data will be denoted by:

$$\{S(1) = s(1), S(2) = s(2), ..., S(T) = s(T)\} = \{s(t)\}_{t=1}^{T} \tag{2.1}$$

Next, the concept of stationarity of a time series is discussed. Roughly speaking, stationarity implies time-invariance of the joint probability distribution of the data generating process (strict stationarity), or just of its first two moments (weak stationarity). Other "levels" of stationarity could be considered up to an order m, where m represents the first m moments of the joint probability distribution. The definitions of both strict and weak stationarity are provided:

Definition 2.1.2 (Strict stationarity)**.** *A random process* $\{S(t)\}$ [1] *is said to be strictly stationary if for all* $k \in \mathbb{N}$, $\tau \in \mathbb{Z}$ *and* $(t_1, t_2, ..., t_k) \in \mathbb{Z}^k$,

$$(S(t_1), S(t_2), ..., S(t_k),) \overset{\text{d}}{=} (S(t_{1+\tau}), S(t_{2+\tau}), ..., S(t_{k+\tau}),) \tag{2.2}$$

where $\overset{\text{d}}{=}$ *defines equality in distribution.*

This means that all the distributions have the same mean, variance, etc, assuming that these quantities exist. However, strict stationarity does not make any assumption about the correlation between $S(t), S(t_1), ..., S(t_r)$, other than the correlation between $S(t)$ and $S(t_r)$ only depends on $t - t_r$ and not on t. That is, strict stationarity allows for general dependence between the random variables in the stochastic process. There might be processes that instead of being defined through this first definition, they rely instead on the following:

Definition 2.1.3 (Weakly stationarity)**.** *A random process* $\{S(t)\}$ *is said to be weakly stationary (or covariance stationary) if for all* $\tau, t \in \mathbb{Z}$,

$$\mathbb{E}\left[S(t)\right] = \mu \tag{2.3}$$

$$\mathbb{C}ov[S(t), S(t-\tau)] = \mathbb{C}ov(\tau) \tag{2.4}$$

with $\mathbb{C}ov(0) < \infty$.

Strictly stationarity implies weakly stationarity, but not the converse. In the above defintion, eqn. 2.4 represents the autocovariance function of the random process $\{S(t)\}$. In practice, the covariance between two instances $S(t_1)$ and $S(t_2)$ at two different times t_1 and t_2 of a stochastic process is called autocovariance. A formal definition is given as follows

Definition 2.1.4 (Autocovariance Function)**.** *The autocovariance function of a random process* $\{S(t)\}$ *is the covariance between* $S(t)$ *and* $S(t-\tau)$ *defined as*

$$\mathbb{C}ov[S(t), S(t-\tau)] = \mathbb{E}\left[S(t) - \mathbb{E}\left[S(t)\right]\right] \mathbb{E}\left[S(t-\tau) - \mathbb{E}\left[S(t-\tau)\right]\right] \tag{2.5}$$

[1] This notation is equivalent to $\{S(t)\}_{t=\infty}^{\infty}$.

As shown in eqn. 2.4, for stochastic processes which are stationary at least up to order $m = 2$, the autocovariance function only depends on the difference $\tau = t - (t - \tau)$ and can therefore be defined as $\mathbb{C}\text{ov}(\tau)$. A further quantity often used in the study of a stochastic processes is the autocorrelation function.

Definition 2.1.5 (Autocorrelation Function)**.** *The autocorrelation function of a random process* $\{S_t\}$ *is defined as*

$$\rho\left[S(t), S(t-\tau)\right] = \frac{\mathbb{C}ov\left[S(t), S(t-\tau)\right]}{\sqrt{\mathbb{V}ar\left[S(t)\right]}\sqrt{\mathbb{V}ar\left[S(t-\tau)\right]}} \tag{2.6}$$

As for the autocovariance function, if the stochastic process is stationary at least up to order $m = 2$, then the autocorrelation function is a function of $\tau = t-(t-\tau)$ and can therefore be defined as $\rho(\tau)$.

Another assumption that most data analysis methods rely on is the concept of linearity. They take into account datasets by supposing that they come from a linear system. Mathematically, a stochastic process $\{S(t)\}$ is defined as linear according to the following definition.

Definition 2.1.6 (Linear Stochastic Process)**.** *A linear stochastic process* $\{S(t)\}$ *is a process that can be written on the form*

$$S(t) = \epsilon(t) + \sum_{i=1}^{\infty} \psi_i\, \epsilon(t-i) = \left(1 + \sum_{i=1}^{\infty} \psi_i B^i\right) \epsilon(t) = \psi(B)\epsilon(t) \tag{2.7}$$

where $\epsilon(t)$ *is i.i.d. with* $\mathbb{E}\left[\epsilon(t)\right] = 0$, $\mathbb{E}\left[\epsilon(t)\right]^2 < \infty$ *and* $\sum_0^\infty \psi_i < \infty$.

Note that B represents the backward shift operator such that $BS(t) = S(t-1)$, $B^iS(t) = S(t-i)$ and $\psi(B)\epsilon(t)$ is called the transfer function. The above processes are often referred to Moving-Average (MA) Processes. For comparison, the Wold's Theorem is also provided in the following definition

Definition 2.1.7 (Wold's Theorem)**.** *Any covariance-stationary process* $S(t)$ *has a unique representation as the sum of a purely deterministic component and an infinite sum of white noise terms, given as*

$$S(t) = \delta_t + \sum_{i=1}^{\infty} \psi_i \epsilon(t-i)$$

with $\psi_0 = 1$, $\sum_{i=0}^\infty \psi_i^2 < \infty$ *and the terms* ϵ_t *defined as the linear innovations* $S(t) - \mathbb{E}[S(t)|\mathcal{H}_{t-1}]$ *where* $\mathbb{E}[\cdot]$ *denotes the linear expectation or projection on the space* $\mathcal{H}_{t-1}$ *that is generated by the observations* $S(s)$, $s \leq t-1$

Hence, a linear stochastic process is a process that satisfy the above representation and Wold's theorem. This means that it can be re-expressed uniquely as the sum of purely deterministic component and a non-deterministic component given as the infinite sum of withe noise terms with a linear representation coefficients. To identify a non-linear process instead, one should consider any

process that cannot be expressed in the above form uniquely. There are many covariance-stationary processes that are not linear since the innovations are not independent (though white noise) or the absolute coefficients do not converge. There is no definite definition for nonlinearity provided. In Potter (1999), the three main model dealing with nonlinearity are described. The reader might refer to that for further details.

The concepts of a stationarity and linearity of a stochastic process have been introduced. What happens in practice is that these assumptions on a process are idealised and do not hold in real-world applications. It is indeed likely that the investigated stochastic process happens to be strongly non-linear ad non-stationary. A stochastic process is said to be non-stationary if its mean and autocorrelation functions averaged over the sample functions (and not over time) change with respect to a general time t. Therefore, it is highly challenging to detect properties of such processes since they are depending on time and changing over its domain. Given these issues in defining a general technique to analyse non-stationary data, several solutions have been considered. One example is that methodologies of a specific class of stochastic processes have been developed. Another solution is considering piecewise stationary data which is a method that splits non-stationary data in intervals that are approximately locally stationary so that they can be analysed approximately under the stationarity assumptions just defined above. Otherwise, it is possible to factorise the sample functions of some non-stationary processes in one of the following ways:[2]

$$
\begin{aligned}
s(t) &= a(t) + u(t) \\
s(t) &= a(t)\, u(t) \\
s(t) &= u(t^n)
\end{aligned}
$$

where $u(t)$ is a sample function coming from a stationary random process $\{u(t)\}$ and $a(t)$ is a deterministic function repeating over each function. The three equations correspond to: (1) decomposition of trend into non-stationary and stationary components, (2) decomposition of volatility into non-stationary and stationary components and (3) non-stationarity from a time dilation or time-scaling effect. It is possible to combine together the above functions and fit different kind of nonstationary data. The idea of this section is providing a general overview of different situations that can be found within data analysis. Three main properties have been described that may lead to the employment of different methodologies that are going to be presented in the next section.

A critical remark has to be taken into account when reading the following sections. The concepts of a stochastic process and stationarity, linearity, and its autocovariance function have been presented. This is required to understand and describe the data properties derived from such stochastic processes. In practice, scientists directly deal with realisations of stochastic processes, which could

[2]However, it has been provided lot of evidence proving that working with non-stationary data creates several issues in terms of estimation and forecasting. Since physical phenomena tend to be non-stationary, a well-defined technique able to deal with them is highly required.

be considered deterministic if the random component and the distribution of the process are instead ignored. Therefore, the presented time-frequency methods are all introduced for a deterministic and continuous signal $s(t)$. This is highly relevant since, in Chapter 3, the primary method of this thesis, i.e. the Empirical Mode Decomposition, is also introduced with this perspective. The stochastic embedding taking into account the randomness of the signal, will then be introduced in later Chapters.

2.2 Stationary Time-Frequency Methods

One of the most common goals of data analysis is synthesizing a general set of data through the use of an elementary basis. This procedure can be considered in mathematics for deterministic functions as to expressing a general function f as a linear combination of elementary basis functions. To provide an example, the power functions 1, s, s^2.. can be exploited to summarise an arbitrary function $f(s)$ as a linear combination of the power series as follows:

$$f(s) = a_0 + a_1 x + a_2 s^2 + ... \tag{2.8}$$

where each coefficient a_k is by the Maclaurin formula:

$$a_k = \frac{f^{(k)}(0)}{k!}, k = 0, 1, 2, 3.. \tag{2.9}$$

By observing the general equation for the coefficients, it is evident that it can only work if f, f', f''.. are defined at $s = 0$. Therefore, even if a large number of functions are part of this set, not every function can be defined through the use of this specific basis. By looking at this simple example, it has to be noted that defining a basis and then trying to synthesize an arbitrary function is a process that may raise several issues (depending on the basis itself, the arbitrary function, how the basis fits the function, assumptions made on both, etc.). The extension of these concepts to stochastic processes or sample paths/ observed trajectories of such processes i.e. time-series brings additional levels of complexity. By focusing on time-frequency analysis as a framework, several basis have been introduced. As discussed previously, the Fourier method was the first approach to shed light on the concept of frequency analysis and hence the possibility of switching the domain. The main problem of this transform, however, is created by strong required assumptions. As a consequence, further methods have been developed within the literature for non-stationary data such as the spectogram, the Wavelet analysis or the Wigner-Ville distribution. These techniques overtake the Fourier analysis in most of the applications since they might be considered as variations of the original Fourier spectral analysis. Even if they provide better performances, data often present both non-stationary and non-linearity; thus, several issues are still present and need to be solved. An alternative to all these methodologies is the Huang-Hilbert Transform. The innovation of this procedure is the main approach in defining a basis for the analysis of data: it is an adaptive basis strictly

depending on data without the need of strong mathematical assumptions. This procedure is the main focus of this thesis and will be introduced in Chapter 3.

The section is organised as follows: the first part is entirely dedicated to the Fourier transform and the explanation of its main difficulties. Then, non-stationary methods are presented[3] by underlying lacks in their possible data applications.

2.2.1 The Fourier Analysis

The basic idea of the Fourier transform is the decomposition of a signal into the sum of sinusoids characterised by different frequencies. Therefore, the original signal can be interpreted as the sum of waveforms distinguished by different frequencies (and hence their amplitudes). The mathematical relationship of this concept can be expressed as follows:

$$S(\omega) = \mathcal{F}[s(t)] = \int_{-\infty}^{\infty} s(t)\, e^{-\jmath 2\pi\omega t} dt \tag{2.10}$$

where $s(t)$ is the signal that has to be decomposed into the sum of sinusoids, $\mathcal{F}[s(t)]$ is the Fourier transform of $s(t)$ and $\jmath = \sqrt{-1}$. Note that throughout the thesis, the frequency units will be interchangeably denoted in radians by f and in hertz by ω where the conversion between units is given by $f = 2\pi\omega$. It is common practice to define the above integral with $f(x)$ or $f(t)$ instead of $s(t)$. In this thesis, the considered signal will always be a function of time and therefore defined as $s(t)$.

Usually, periodic functions are associated with the term Fourier series instead of Fourier transforms. However, it has been shown that the Fourier series is a special case of the Fourier transform. While, if the original signal is not periodic, then the Fourier transform is a continuous function of frequency. This results in synthesising $s(t)$ as the weighted infinite sum of an infinite number of bases terms, one for each frequency in the infinite continuum of frequencies required in this commonly arising case. This is the formalisation of the concept previously defined when talking about deformed signals in section 1.2. The point is that the Fourier transform expresses a signal within the domain of frequency. It should carry precisely the same information as the original waveform with the advantage of looking at it from a different perspective. Several applications have been exploited as a problem-solving tool in analysing data by assuming another point of view.[4] In the following subsections, the Fourier integral is firstly introduced for a continuous signal $s(t)$. Then, its inverse transform is presented to obtain the original signal $s(t)$ again from the frequency domain. Afterwards, the conditions for the existence of the Fourier integral are discussed to explain which signal $s(t)$ will be suitable for such a transform. Finally, the main Fourier transform properties are introduced.

[3]The wavelet analysis, the Wigner-Ville distribution and the spectrogram were chosen as the most used in time-frequency analysis for non-stationary signals.

[4]Some examples of applications are: linear systems, antennas, optics, random process, probability, quantum physics, boundary-value problems, etc.

The Fourier Integral

Given a function of the real variable t the following integral can be formed:

$$S(\omega) = \mathcal{F}[s(t)] = \int_{-\infty}^{\infty} s(t)\, e^{-j2\pi\omega t} dt \tag{2.11}$$

If this exists for every real value of the parameter f, then it defines a function $\mathcal{F}[s(t)]$ known as the Fourier transform or Fourier integral of $s(t)$.[5] The Fourier transform can be considered as a complex quantity given by:

$$S(\omega) = \mathrm{Re}(\omega) + \jmath\, \mathrm{Im}(\omega) = |S(\omega)| e^{\jmath\theta(\omega)} \tag{2.12}$$

where $\mathrm{Re}(\omega)$ and $\mathrm{Im}(\omega)$ define the real and the imaginary part of the Fourier transform respectively. $|S(\omega)|$ represents the amplitude of the Fourier spectrum of $s(t)$ and is given by $\sqrt{\mathrm{Re}(\omega)^2 + \mathrm{Im}(\omega)^2}$. Finally, $\theta(\omega)$ represents the phase angle of the Fourier transform and is given by $\tanh^{-1}\left[\frac{\mathrm{Im}(\omega)}{\mathrm{Re}(\omega)}\right]$.

Fourier Transform Properties.

Having defined when the Fourier Integral is well defined, it is now worth to consider the basic properties that such an integral transform respects. The Fourier transform implies the respect of some basic properties that are listed within this section. As before, Brigham and Morrow (1967) is followed.

- Linearity
  If $s(t)$ and $r(t)$ have Fourier transforms defined respectively as $S(\omega)$ and $R(\omega)$ then the sum $s(t) + r(t)$ has Fourier transform equal to the sum of their related Fourier transforms $S(\omega) + R(\omega)$.

$$\begin{aligned}\int_{-\infty}^{\infty} [s(t) + r(t)]\, e^{-j2\pi\omega t} dt &= \int_{-\infty}^{\infty} [s(t)]\, e^{-j2\pi\omega t} dt + \int_{-\infty}^{\infty} [r(t)]\, e^{-j2\pi\omega t} dt = \\ &= S(\omega) + R(\omega)\end{aligned} \tag{2.13}$$

- Symmetry
  If $s(t)$ and $S(\omega)$ are a Fourier transform pair, then this relationship is establishes as:

$$s(-t) = \int_{-\infty}^{\infty} S(\omega)\, e^{-j2\pi\omega t} d\omega \tag{2.14}$$

  and by interchanging the parameter t and f as follows:

$$s(-\omega) = \int_{-\infty}^{\infty} S(t)\, e^{-j2\pi\omega t} dt \tag{2.15}$$

[5] Remark that t defines time and f is the index indicating the frequency. Hence, $\mathcal{F}[s(t)]$ is function of the frequency while $s(t)$ is determined as a function of time.

- Time Scaling
  If $S(\omega)$ is the Fourier transform of $s(t)$, then by scaling the function of a real parameter $k > 0$, i.e. $s(kt)$, its Fourier transform is given by:

$$\int_{-\infty}^{\infty} s(kt)\, e^{-j2\pi\omega t} dt = \int_{-\infty}^{\infty} s(t')\, e^{-j2\pi t'\left(\frac{\omega}{k}\right)} \frac{dt'}{k} = \frac{1}{k} S\left(\frac{\omega}{k}\right) \tag{2.16}$$

  where $t' = kt$. If $k < 0$, then the result is $\frac{1}{|k|} S\left(\frac{\omega}{k}\right)$ given a change of the sign. If impulses are taken into account, then the result is:

$$\delta(at) = \frac{1}{|a|} \delta(t) \tag{2.17}$$

- Frequency-scaling
  If $S(\omega)$ is the inverse Fourier transform of $s(t)$, then the inverse Fourier transform of $\mathcal{F}(k\omega)$ (with k real constant) is given by:

$$\int_{-\infty}^{\infty} S(k\omega)\, e^{-j2\pi\omega t} df = \int_{-\infty}^{\infty} s(\omega')\, e^{-j2\pi\omega'\left(\frac{t}{k}\right)} \frac{d\omega'}{k} \tag{2.18}$$

$$= \frac{1}{|k|} c\left(\frac{t}{k}\right) \tag{2.19}$$

  where $\omega' = kf$. If impulses are the functions of interest, then the result is:

$$\delta(a\omega) = \frac{1}{|a|} \delta(\omega) \tag{2.20}$$

- Time-shifting
  By shifting $s(t)$ of a constant t_0, the Fourier transform is:

$$\int_{-\infty}^{\infty} s(t - t_0)\, e^{-j2\pi} dt = \int_{-\infty}^{\infty} s(s)\, e^{-j2\pi\omega(s+s_0)} ds \tag{2.21}$$

$$= e^{-j2\pi\omega t_0} \int_{-\infty}^{\infty} s(s)\, e^{-j2\pi\omega(s)} ds \tag{2.22}$$

$$= e^{-j2\pi f t_0} S(\omega) \tag{2.23}$$

  where $s = t - t_0$.

- Frequency-shifting
  By shifting $S(t)$ of a constant ω_0, the inverse Fourier transform is:

$$\int_{-\infty}^{\infty} S(\omega - \omega_0)\, e^{-j2\pi\omega t} dt = \int_{-\infty}^{\infty} S(s)\, e^{-j2\pi t(s+\omega_0)} ds \tag{2.24}$$

$$= e^{-j2\pi t\omega_0} \int_{-\infty}^{\infty} S(s)\, e^{-j2\pi t s} ds \tag{2.25}$$

$$= e^{-j2\pi t\omega_0} s(t) \tag{2.26}$$

  where $s = \omega - \omega_0$.

Some comments need to be pointed out after introducing the Fourier analysis, its existing conditions and properties. Given its expressive power and simplicity, the Fourier transform has been highly exploited in time-frequency analysis. Offering another perspective in analyzing a signal as the sum of sinusoids has provided several advantages within data analysis. The primary assumption to make the decomposition valid is stationarity of data; an overview of the concept of stationarity has been provided. Nevertheless, data obtained by observing real phenomena (especially physical phenomena) can often come from non-stationary processes. Therefore, the application of the Fourier transform generates unreliable results: the decomposed signal does not synthesize the original one, and extra harmonics are often needed. These challenges make the Fourier transform inefficient for non-stationary data.

Fourier transform-based alternative methods have been constructed to apply it to non-stationary data as the Fast Fourier Transform or the Short Fourier Transform. However, issues have still been found. Another developed option may be considering piecewise stationary processes; the problem here is that the time interval taken into account often seems too short due to the need for approximations.

The second strong assumption that makes the Fourier analysis of limited use is linearity. Non-stationary data often derive from a non-linear system; thus, they usually deviate their classical wave-profile of sine or cosine. Given the concept of the superimposition of trigonometric functions as a base for the Fourier transform, if any deformation is present because of non-linear data, additional spurious harmonics are again needed.

To sum up, the Fourier analysis has been popular for its simplicity. However, new methodologies able to deal with non-stationary data and non-linear systems are highly needed. As an answer to this requirement, the following section has been introduced.

2.3 Non-stationary Time-Frequency Methods

The following methods have been developed within the literature in order to deal with non-stationary data. Many of them depend on the Fourier analysis or are based on similar principles. Therefore, the non-linearity and non-stationarity of the data will still represent a challenge. Such a fact encloses the main reason for the need for an adaptive basis decomposition (as the Huang-Hilbert Transform later introduced). Several methodologies could be considered. The following are the selected methods for this thesis since widely used within the literature: the Short Time Fourier transform, also known as the spectrogram, the Wavelet analysis and the Wigner-Ville distribution.

2.3.1 The Short Time Fourier Transform

The reasoning behind this transformation is taking the Fourier transform and making it better suited to the analysis of non-stationary data. The basic concept is building a piecewise stationary process and then considering limited-width Fourier spectral analysis over time. The method consists of moving this window through the whole domain of the signal and then get a time-frequency distribution. A graphical display of the Short Time Fourier Transform (STFT) magnitude is called the spectrogram of the signal.

The main challenges of this methodology find their roots in, firstly, choosing the window size combined with the time scale of the signal. Secondly, the length of the signal may also arise issues since it could be too short to embedding the authentic features of the analysed physical phenomenon. A further critical point is a trade-off given by the localisation of changes over time: the window should be narrow enough to detect an event precisely; conversely, synthesising the frequency requires a bigger time span. An effect of narrowing the window too highly is the identification of meaningless spectra of the signals obtained from the decomposition of the original one. Furthermore, as in the Fourier transform, this method uses an a priori basis, which does not consider data-driven characteristics.

To define a short duration time signal, a window function $u(t)$ centered at time t is exploited; the idea is multiplying it times the original signal by assigning more weight to that part and less to the rest of it. The modified signal can be determined as:

$$s_t(\tau) = s(\tau)\, u(t-\tau) \tag{2.27}$$

The new signal is a function of two different times: window index and time index, given by:

$$s_t(\tau) = \begin{cases} s(\tau) & \text{for } \tau \text{ near t} \\ 0 & \text{for } \tau \text{ far away from t} \end{cases} \tag{2.28}$$

By looking at the above definition, it can be observed that the decomposed signal will be the same as the original around t, while 0 for the times far from it. As a result, the word " window " should be clarified since the concept looks at one piece by time of the original signal and analyse its properties by making it stationary. Since the modified signal emphasises the signal around the time t, the Fourier transform will reflect the distribution of frequency around that time:

$$S(\omega) = \frac{1}{\sqrt{2\pi}} \int e^{-\jmath\omega\tau} s_t(\tau)\, d\tau \tag{2.29}$$

Note that, compared to the original definition in eqn. 2.11, the Fourier transform is defined by taking 2π in front of the integral. By substituting 2.27 within it:

$$S(\omega) = \frac{1}{\sqrt{2\pi}} \int e^{-\jmath\omega\tau} s(\tau)\, u(t-\tau)\, d\tau \tag{2.30}$$

Therefore, for each time t there is an energy density spectrum given by

$$P(t,\omega) = |S(\omega)|^2 = \left| \frac{1}{\sqrt{2\pi}} \int e^{-\jmath\omega\tau} s(\tau)\, u(t-\tau)\, d\tau \right|^2 \tag{2.31}$$

The mathematics community refers to the above as the short time Fourier transform or STFT. By looking at the spectra as a whole, then the time-frequency distribution of the original signal is given. This may be named differently but is generally indicated as the " spectrogram ". The final purpose of the STFT is to study frequencies that may be of particular interest and provide further insights on properties of the signal. However, the same reasoning could be done from a time perspective: by employing a window $U(\omega)$ and moving it through the spectrum, the time transform is applied. This is indeed the inverse of the Fourier transform and usually referred to as the short-frequency Fourier transform. If the time window is short, then the frequency window $U(\omega)$ is big and vice versa. In the former case, the spectrogram is called a broadband spectrogram, while in the latter is a narrowband spectrogram. General properties of the spectrogram are given as follows:

- **Characteristic function**
  The characteristic function of the spectrogram is obtained as follows:

$$M(\theta,\tau) = \int\int |S(f)|^2 e^{\jmath\theta t+\jmath\tau\omega} dt d\omega \tag{2.32}$$

$$= A_s(\theta,\tau)\, A_u(-\theta,\tau) \tag{2.33}$$

  where $A_s(\theta,\tau)$ represents the ambiguity function of the signal and is given by:

$$A_s(\theta,\tau) = \int s^*\left(t-\frac{1}{2}\tau\right) s\left(t+\frac{1}{2}\right) e^{j\theta t} dt \tag{2.34}$$

  and $A_u(-\theta,\tau)$ is the ambiguity function of the window defined as the above by replacing $s(t)$ with $u(t)$.

- **Total energy**
  The total energy of the spectrogram is computed by evaluating the characteristic function at zero. Hence, it is given by:[6]

$$E = \int\int P(t,\omega)\, dt d\omega = M(0,0) \tag{2.35}$$

$$= A_s(0,0)\, A_\omega(0,0) \tag{2.36}$$

$$= \int |s(t)|^2 dt \times \int |u(t)|^2 dt \tag{2.37}$$

  It is worth noting that if the energy of the window is resulting to be one (by choosing an appropriate window) then the energy of the spectrogram equals the energy of the signal.

- **Marginals**

[6]The total energy results from integrating over all time and frequency. This computation is a general rule to obtain it.

The marginal of time is computed by integrating over the frequency:

$$\begin{aligned} P(t) &= \int |S(f)|^2 d\omega \\ &= \frac{1}{2\pi} \int s(\tau)\, u(\tau - t)\, s^*(\tau')\, u^*(\tau' - t)\, e^{-\jmath f(\tau - \tau')} d\tau d\tau' df \\ &= \int x(\tau)\, u(\tau - t)\, s^*(\tau')\, u^*(\tau' - t)\, \delta(\tau - \tau')\, d\tau d\tau' \\ &= \int |s(\tau)|^2 |u(\tau - t)|^2 d\tau \\ &= \int A_s^2(\tau)\, A_u^2(\tau - t)\, d\tau \end{aligned} \tag{2.38}$$

It is possible to deal with the same procedure for the marginal frequency and obtain the marginal frequency as:

$$P(\omega) = \int B^2(\omega')\, B_H^2(\omega - \omega')\, d\omega' \tag{2.39}$$

where $B(\omega)$ represents the spectral amplitude. It can be observed that the marginal of the spectrogram do not generally respect the correct marginals, namely $|s(t)|^2$ and $|S(\omega)|^2$,

$$P(t) \neq A^2(t) = |s(t)|^2 \tag{2.40}$$

$$P(\omega) \neq B^2(\omega) = |S(\omega)|^2 \tag{2.41}$$

This results from the fact that the spectrogram spreads and blurs the energy distributions of the window with the ones of the original signal. As a result, the time-frequency representation tends to be unreliable. It it important to point out that the time marginal only depends on the magnitude of the window and the signal and not on the phase. Likewise, the frequency marginal is only connected to the amplitudes of the Fourier transforms. The next property is caused by this last feature of the marginals.

- **Averages of Time and Frequency Functions**
  Since both time and frequency marginals do not respect the marginals as per the statements in Equations (2.40) and (2.41), then any average measure of time and frequency will always be biased. This can be proven as follows:

$$\mathbb{E}[g_1(t) + g_2(\omega)] = \int \int \{g_1(t) + g_2(\omega)\} P(t, \omega)\, d\omega dt \tag{2.42}$$

$$\neq \int g_1(t) |s(t)|^2 dt + \int g_2(\omega) |S(\omega)|^2 d\omega \tag{2.43}$$

- **Localization trade-off**
  As previously introduced, one of the main problems related to this method is the localization of changes over time. A narrow time window is required to pick a specific event. On the other hand, a refined frequency detection is given by a narrow frequency window. However, they cannot be both narrow simultaneously. Therefore, the need of a trade-off in the choice of the windows is suggested. It is a procedure that depends on the signal itself and on the window as well.

- **Number of windows**
  It has been proven that the consideration of more windows has brought many advantages in detecting specific times and frequencies. Hence, depending on the studied problem, there might be a different solution related to the number of the chosen windows.

The short-time Fourier transform has been discussed within this section since it is often exploited as a solution for working with non-stationary processes. Several disadvantages have been found. The localisation trade-off with the lack of a mathematical solution regarding the sliding window setting are some of them and profoundly affect such a method. It is a heuristic procedure, rigorously depending on the dataset. The most relevant issue is represented by the fact that this is not an a posteriori method. The importance of this last point will be further explained in the followings Chapters.

2.3.2 The Wavelet Transform

The Wavelet transform is an alternative transform to the STFT that tries to resolve issues with the uncertainty associated with window selection in either time or frequency domains. It has a high resolution in both time and frequency domains. Hence, it does provide both information about the frequencies present in a signal and at which time these frequencies have occurred. As presented below, this is accomplished through the employment of different scales. There is an essential difference between the STFT and the Wavelet transform. In the STFT and the Wavelet transform, the frequency resolution is directly proportional to the window size. However, in the Wavelet transform, a centre frequency shift is required along with a window size change (time scaling).

A fundamental assumption made on the signal $s(t)$ is that it has to be square integrable as follows:

$$\int_{-\infty}^{\infty} s^2(t)\, dt < \infty \tag{2.44}$$

which is equivalent to writing $s(t) \in L^2(\mathbb{R})$.[7]

The Wavelet transform, similarly to the STFT, maps a time function into a two-dimensional function of a parameter a, named the scale (or dilation factor),

[7] "Note that a dc signal is not an $L^2(\mathbb{R})$ function, neither a pure sinusoid. However, all functions of finite magnitudes and magnitudes and compact support are." (Chan 1995).)

and a factor τ, called the translation. The parameter a scales the function by compressing it or stretching it, while the parameter τ controls the function along the time axis. The continuous wavelet transform of a signal $s(t)$ is defined as:

$$CWT(a,\tau) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} s(t)\, \psi\left(\frac{t-\tau}{a}\right) dt \tag{2.45}$$

where $\psi(t)$ is the basic or mother wavelet function and $\psi((t-\tau)/a)/\sqrt{a}$ the wavelet basis function, often called baby wavelets. By considering the following change of variable $at' = t$:

$$CWT(a,\tau) = \sqrt{a} \int_{-\infty}^{\infty} s(at')\, \psi\left(\frac{t'-\tau}{a}\right) dt' \tag{2.46}$$

it is possible to observe the equivalence in scaling $\psi(t)$ or $s(t)$ in 2.45 or 2.46 respectively to get the wavelet transform.

The basic wavelet $\psi(t)$ may be both real and complex. Therefore, the Wavelet transform could be real or complex. If $\psi(t)$ is complex, then its complex conjugate is employed in both equations 2.45 and 2.46. The difference is often related to the requirements of the application of interest. Examples of $\psi(t)$ are

- Modulated Gaussian (Morlet)

$$\psi(t) = e^{\jmath f t} e^{-\frac{t^2}{2}} \tag{2.47}$$

- Second derivative of a Gaussian

$$\psi(t) = \left(1 - t^2\right) e^{-\frac{t^2}{2}} \tag{2.48}$$

- Haar

$$\psi(t) = \begin{cases} 1, & 0 \le t \le 1/2 \\ -1, & 1/2 \le t < 1 \\ 0, & \text{otherwise} \end{cases} \tag{2.49}$$

- Shannon

$$\psi(t) = \frac{\sin(\pi t/2)}{\pi t/2} \cos\left(\frac{3\pi t}{2}\right) \tag{2.50}$$

Note that, similarly to the spectrogram for the STFT, the scalogram of the wavelet transform is defined as

$$|CWT(a,\tau)|^2 \tag{2.51}$$

Given the above figures and the definition of the wavelet basis functions, the following properties can be deduced:

- $\int_{-\infty}^{\infty} \psi(t)\, dt = 0$ i.e. they have zero dc components. Note that the dc component stand for the "direct current" component which derives from electrical engineering. In practice, the dc component, also referred to as dc offset, corresponds to the zero-frequency component of a signal. This is in contrast to the sinusoids, which "alternate".
- They are bandpass signals.
- They decay rapidly towards zero with time (the original French word for this property corresponds to "Ondelette").

The first property directly derives from the admissibility condition of a wavelet, which ensures the wavelet transform has an inverse. Consider $s(t) \in L^2(\mathbb{R})$ and its the continuous wavelet transform given as in eq. 2.45. If $\psi(t)$ is such that this transform is invertible, then

$$s(t) = \frac{1}{c_\psi} \int_{-\infty}^{+\infty} \int_{a>0}^{+\infty} CWT(a,\tau) \frac{1}{\sqrt{a}} \psi\left(\frac{t-\tau}{a}\right) \frac{1}{a^2} da d\tau \tag{2.52}$$

where c_ψ is a constant that depends only on $\psi(t)$ and a is positive. The constant has value

$$c_\psi = \int_0^\infty \frac{|\Psi(f)|}{f} df < \infty \tag{2.53}$$

where $\Psi(\omega)$ is the Fourier transform of $\psi(t)$. The above equation imposes an admissibility condition on $\psi(t)$. For $c_\psi < \infty$, $\psi(t)$ must be such that

$$|\Psi(\omega)| < \infty, \quad \text{for any} \quad f \tag{2.54}$$

and $\Psi(0) = 0$ which implies

$$\int_{-\infty}^{+\infty} \psi(t) = 0 \tag{2.55}$$

As presented in Chan (1994), the introduced examples for $\psi(t)$, they all satisfy the admissibility condition. For the modulated Gaussian, $\Psi(0)$ is not exactly zero, although by selecting ω_0 sufficiently large then $\Psi(0)$ is approximately equal to zero. Note that, within Appendix A of Chan (1994), the proof of the invertibility of the CWT is provided. The proof exploits the resolution of the identity theorem which states that the transformation of a one-dimensional signal $s(t)$ into the two-dimensional wavelet domain via 2.52 is invertible if the transformation is an isometry up to a constant factor c_ψ as in eqn. 2.53.
The second property follows from the first one. Regarding the third property, the rapid decay of $\psi(t)$ is not a requirement for $\psi(t)$ to be a wavelet. In practice, $\psi(t)$ should have a compact support in order to have a good time localisation.

By comparing eqn. 2.31 with eqn 2.45, there is a similarity between $\psi(t)$ of the CWT and $u(t)e^{-\jmath\omega\tau}$. As highlighted in Chan (1994), the integral in 2.45 could be seen in four different ways. First, it computes the inner product, i.e. the cross-correlation, of $s(t)$ with $\psi(t/a)\sqrt{a}$ at shit τ/a. Therefore, it computes the

similarity between these two. Second, it is the output of a bandpass filter of impulse response $\psi(-t/a)/\sqrt{a}$, of input $s(t)$ at the instant τ/a. Third, since 2.46 is identical to 2.45, it also provides the similarity or the inner product of a scaled signal $s(at)$ and $\sqrt{a}\psi(t)$ at shift τ/a. Lastly, from 2.46, it follows that the CWT is also the output of a bandpass filter of impulse response $\sqrt{a}\psi(-t)$, of input $s(at)$ at the instant τ/a.

The above different interpretations of the 2.45 give rise to different forms of this transform. It might depend either on the available algorithm to compute it or to the application of interest. Hence, one could consider the continuous wavelet transform, the discrete parameter wavelet transform, the discrete time wavelet transform or the discrete wavelet transform. These are well-presented in Chan (1994).

2.3.3 The Wigner-Ville Distribution

There are two main categories used in the literature to describe a signal's frequency content: a linear representation such as the Fourier transform or a quadratic representation such as the power spectrum, corresponding to the square of the Fourier transform. The most used counterpart to the power spectrum is a quadratic, or bilinear, joint time-frequency representation. Note that a linear joint-time frequency representation corresponds to the STFT above described. Several quadratic joint time-frequency representations have been introduced in the past. However, the Wigner distribution is presented in this paragraph since it is one of the most popular for its characteristics in being both very straightforward and powerful. The section firstly explains the motivations for this distribution and then its central properties. The uniqueness of the Wigner distribution lies in its description of a signal's time-varying nature more effectively compared to many other existing methods such as the STFT. Furthermore, its properties are highly beneficial for signal analysis in general. The Wigner distribution's main drawback is cross-term interference and limits such a transform in several applications. Such an issue is described below (see Cohen (1995), Papoulis (1977)).

Time-Dependent Power Spectrum

As highlighted above, the power spectrum is the Fourier transform's square and describes the signal's energy distribution in the frequency domain. While the Fourier transform is linear, the power spectrum is a quadratic function of frequencies. An alternative way to detect such information is to employ the STFT to describe the signal's energy distribution in a joint time-frequency domain. The aspect taken into account at this stage regards the Wiener-Khinchin theorem (Papoulis (1977)), which states that the autocorrelation function of a wide-sense-stationary random process has a spectral decomposition given by the power spectrum of that process. Hence, the power spectrum can also be considered as the Fourier Transform of the auto-correlation function $\rho(\tau)$ (as defined in defini-

tion 2.1.5). Note that $\rho(\tau)$ also corresponds to

$$\rho(\tau) = \int s(t)\ s^*(t-\tau)\ d\tau \tag{2.56}$$

where s^* denotes the complex conjugate of s. The power spectrum can then be expressed as

$$P(t,\omega) = \ |S(\omega)|^2 = \int \rho(\tau)\ e^{-\jmath f\tau}\ d\tau \tag{2.57}$$

The above equation is not a function of time and, therefore, indicates how much energy is present in frequency f over the entire time period of the signal. Hence, it is not possible to deduce if the signal's power spectrum changes over time. One possible way to tackle this issue is to make the autocorrelation function time-dependent $\rho(t,\tau)$. The resulting Fourier transform with respect to the variable τ becomes a function of time as

$$P(t,\omega) = \int\ \rho(t,\tau)\ e^{-\jmath\omega\tau}\ d\tau \tag{2.58}$$

where $P(t,\omega)$ is now a time-dependent power spectrum (Qian and Chen (1996)). The choice of $\rho(t,\tau)$ is not arbitrary. For example the following should hold

$$\int\ P(t,\omega)\ dt = |S(\omega)|^2 \tag{2.59}$$

which is traditionally known as the frequency marginal condition, meaning that by adding all the instantaneous-time power spectrum $P(t_0,\omega)$ should yield to the total power spectrum $|S(\omega)|^2$. Furthermore, the integration along the frequency axis should be equal to the instantaneous energy, i.e.

$$\frac{1}{2\pi}\int P(t,\omega)\ d\omega = |s(t)|^2 \tag{2.60}$$

which is commonly known as the time marginal condition (see Qian and Chen (1996)). If $P(t,\omega)$ represents the signal energy distribution in the joint time-frequency domain, then it is real valued, i.e. $P(t,\omega) = P^*(t,\omega)$. A further desired property from the traditional energy concept is that the time-dependent spectrum would be non-negative.

The Wigner Distribution

The Wigner distribution was developed in quantum mechanics in by Wigner in Wigner (1997) and then introduced in signal analysis by Ville in Ville (1948). It has been applied in several areas such as seismology, geography, electrical engineering, speech analysis, EEG analysis. It uses a variation of the classical autocorrelation function which is called instantaneous autocorrelation, which omits the integration step. As a result, time remains in the main result. The instantaneous autocorrelation function is therefore a two dimensional function depending on t and the lag τ given as

$$\rho(t,\tau) = \ s^*(t-\frac{1}{2}\tau)\ s(t+\frac{1}{2}\tau) \tag{2.61}$$

Substituting the above equation into 2.58 yields to (Qian and Chen (1996), Papoulis (1977))

$$\mathcal{W}(t,\omega) = \frac{1}{2\pi}\int s^*(t-\frac{1}{2}\tau)\, s(t+\frac{1}{2}\tau)e^{-\jmath\omega\tau}\, d\tau \tag{2.62}$$

$$= \frac{1}{2\pi}\int S^*(\omega+\frac{1}{2}\theta)\, S(f-\frac{1}{2}\theta)e^{-\jmath\omega\theta}\, d\theta \tag{2.63}$$

Hence, the Wigner distribution calculates the frequency content for each time t by taking the Fourier transform of the instantaneous autocorrelation across the axis of the lag variable τ. The final result is real-valued. Such a calculation is allowed since the Fourier spectrum of a signal equals the Fourier transform of its autocorrelation function.
The Wigner distribution is said to be bilinear in the signal because the signal enters the transformation twice. At a time t, this transform corresponds to the product of pieces of the signal evaluated at past times with pieces of the signal evaluated at future times, which are added up. The shifting time in the past and the future is equivalent. Therefore, in order to determine properties of the Wigner distribution at a time t, the left part of the signal is folded to the right to see if there is any overlap. If there is, then tose properties will be present at time t. Furthermore, the Wigner distribution weighs the far away times equally to the near times, which means that is highly non-local. Eqn 2.63 is often referred to as auto-Wigner distribution. Likewise, the cross-Wigner distribution is given as

$$\mathcal{W}(t,\omega) = \frac{1}{2\pi}\int s(t+\frac{1}{2}\tau)\, g^*(t-\frac{1}{2}\tau)e^{-\jmath\omega\tau}\, d\tau \tag{2.64}$$

In the following paragraphs, different properties of this distribution are presented.

Range of the Wigner Distribution

The Wigner distribution satisfies the finite support properties in time and frequency (see Qian and Chen (1996))

$$\begin{aligned} &\mathcal{W}(t,\omega) = 0 \quad \text{for } t \text{ ouside } (t_1,t_2) \quad \text{if } s(t) \text{ is zero outside } (t_1,t_2) \\ &\mathcal{W}(t,\omega) = 0 \quad \text{for } \omega \text{ ouside } (\omega_1,\omega_2) \quad \text{if } S(\omega) \text{ is zero outside } (\omega_1,\omega_2) \end{aligned} \tag{2.65}$$

The time and frequency support of the Wigner distribution are claimed desirable properties. If the $s(t)$ is non-zero in a certain range (t_1,t_2) and zero elsewhere, then the Wigner distribution is also non-zero in this range (t_1,t_2) and zero elsewhere. The time support property might be misleading. It should not be inferred that any zero-valued region in the signal has a corresponding zero-valued region in the Wigner distribution. This is true only if the zero-filled region extends to $\pm\infty$.
Further distributions have been proposed in the literature to tackle this challenge such as the Cone-Kernal (Zhao et al. (1990)) distribution as well as the smoothed Pseudo Wigner Distribution (Hlawatsch et al. (1992)); however, a second problem often occurs in this cases since the marginal properties of the Wigner distribution are not respected.

The Characteristic Function of the Wigner Distribution

As highlighted in Cohen (1995) and Thayaparan (2000), the characteristic function of the Wigner distribution is given as

$$\begin{aligned} M(\theta,\tau) &= \int\int e^{\jmath\theta t+\jmath\tau\omega}\mathcal{W}(t,\tau)\ dt\ d\omega \\ &= \frac{1}{2\pi}\int\int\int e^{\jmath\theta t+\jmath\tau\omega}\ s^*(t-\frac{1}{2}\tau')s(t+\frac{1}{2}\tau')\ e^{-\jmath\tau'\omega}\ d\tau'\ dt\ d\omega \\ &= \int\int e^{\jmath\theta t}\ \delta(\tau-\tau')\ s^*(t-\frac{1}{2}\tau')s(t+\frac{1}{2}h')\ d\tau'\ dt \\ &= \int s^*(t-\frac{1}{2}\tau)s(t+\frac{1}{2}\tau)\ e^{\jmath\theta t}\ dt \\ &= A(\theta,\tau) \end{aligned} \tag{2.66}$$

where $A(\theta,\tau)$ corresponds to the symmetric ambiguity function defined in eqn. 2.34. Note that the characteristic function of the spectrogram was also introduced with respect to this function. Reasons to consider such a function lie in the fact that in viewing results of the Wigner distribution or the ambiguity function, the latter one would isolate the auto-terms from the cross-terms providing a better interpretation. The reader should refer to Sandsten (2016) for further explanation. In terms of the spectrum, the characteristic functions is given instead as

$$M(\theta,h) = \int\ S^*(\omega+\frac{1}{2}\theta)S(\omega-\frac{1}{2}\theta)\ e^{\jmath h\omega}\ d\omega \tag{2.67}$$

General Properties

In this section, the general properties of the Wigner distribution are introduced. The reader might refer toCohen (1995) and Thayaparan (2000) for further details.

- **Nonpositivity**
  As highlighted in Cohen (1995) and Thayaparan (2000), a bilinear distribution satisfying the marginals cannot be positive throughout the entire time-frequency plane, i.e. it must be negative in some regions. The Wigner distribution, however, satisfies the marginals and therefore it is expected to be negative in certain regions. However, there is an exception presented in Cohen (1995) which provides that the Wigner distribution is not really bilinear and belongs to the class of positive distributions which are not bilinear.

- **Time and Frequency Shift Invariance**
  If the signal $s(t)$ is time-shifted by t_0 and/or spectrum-shifted by ω_0, then the Wigner distribution is shifted accordingly

$$\text{if}\quad s(t)\ \rightarrow e^{\jmath\omega_0 t}\ s(t-t_0)\quad\text{then}\quad \mathcal{W}(t,\omega)\ \rightarrow \mathcal{W}(t-t_0,\omega-\omega_0) \tag{2.68}$$

If the signal in the Wigner distribution is replaced by $e^{\jmath\omega_0 t}\ s(t-t_0)$ and $\mathcal{W}_{s\tau}$ represents the shifted distribution then

$$\begin{aligned}\mathcal{W}_{s\tau} &= \frac{1}{2\pi}\int e^{-\jmath\omega_0(t-\tau/2)}\ s^*(t-t_0-\frac{1}{2}\tau)\ \times\ e^{-\jmath\omega_0\ (t+\tau/2)}s(t-t_0+\frac{1}{2}\tau)e^{-\jmath\tau\omega}\ d\tau \\ &= \frac{1}{2\pi}\int s^*(t-t_0-\frac{1}{2}h)s(t-t_0+\frac{1}{2}\tau)e^{-\jmath\tau(\omega-\omega_0)}\ d\tau \\ &= \mathcal{W}(t-t_0,\omega-\omega_0)\end{aligned} \tag{2.69}$$

- **Reality**
  The Wigner distribution is always real, even if the signal is complex. As shown in Cohen (1995), by considering the complex conjugate of $W(t,f)$

$$\begin{aligned}\mathcal{W}^*(t,\omega) &= \frac{1}{2\pi}\int s(t-\frac{1}{2}\tau)s^*(t+\frac{1}{2}\tau)\ e^{\jmath\tau\omega}\ dh \\ &= \frac{1}{2\pi}\int_{\infty}^{-\infty} s(t+\frac{1}{2}h)s^*(t-\frac{1}{2}\tau)\ e^{-\jmath\tau\omega}\ dh \\ &= \frac{1}{2\pi}\int_{-\infty}^{\infty} s(t+\frac{1}{2}\tau)s^*(t-\frac{1}{2}\tau)\ e^{-\jmath\tau\omega}\ d\tau \\ &= \mathcal{W}(t,\omega)\end{aligned} \tag{2.70}$$

  This fact can also be proved through the characteristic function. Recall that $M^*(-\theta,-\tau) = M(\theta,\tau)$ is the condition for a distribution to be real. But the characteristic function of the Wigner distribution corresponds to the ambiguity function which satisfies this property.

- **Symmetry**
  By substituting $-\omega$ for ω into the Wigner distribution, it is possible to observe that the identical form is obtained if the signal is real. Furthermore, real signals have symmetric spectra. Hence, for symmetric spectra, the Wigner distribution is symmetrical in the frequency domain. Similarly, for real spectra the time waveform is symmetrical an, therefore, the Wigner distribution is symmetric in the time domain. Therefore:

$$\begin{aligned}&\mathcal{W}(t,\omega) = \mathcal{W}(t,-\omega) \quad \text{for real signals } \equiv \text{ symmetrical spectra, } S(\omega) = S(-\omega) \\ &\mathcal{W}(t,\omega) = \mathcal{W}(-t,\omega) \quad \text{for real spectra } \equiv \text{ symmetrical signals, } s(t) = s(-t)\end{aligned} \tag{2.71}$$

- **Time and Frequency Marginals**
  The Wigner distribution satisfies the time-frequency marginals

$$\int \mathcal{W}(t,\omega)\ d\omega = |s(t)|^2 \tag{2.72}$$

$$\int \mathcal{W}(t,\omega)\ dt = |S(f)|^2 \tag{2.73}$$

Both this equations can be verified by considering $M(\theta, 0)$ and $M(0, \tau)$ as follows

$$M(\theta, 0) = \int |s(t)|^2 e^{j\theta t} dt \; ; \quad M(0, \tau) = \int |S(\omega)|^2 e^{j\tau f} d\omega \tag{2.74}$$

Note that these are the characteristic functions of the marginals and hence the marginals are satisfied. Since the marginals are satisfied, the total energy condition is also automatically satisfied.

- **Instantaneous Frequency and Group Delay**
  Consider $s(t) = a(t)e^{j\theta(t)}$ where $a(t)$ and $\theta(t)$ are amplitude and phase function respectively, both real valued. Then

$$\mathbb{E}[\omega] = \frac{\int \omega \; \mathcal{W}(t, \omega) d\omega}{\int \mathcal{W}(t, \omega) \; d\omega} = \frac{1}{|a(t)|^2} \int \omega \; \mathcal{W}(t, \omega) \; d\omega = \theta'(t) \tag{2.75}$$

  which states that, at a specific time t, the mean instantaneous frequency is equal to the mean instantaneous frequency of the given signal (see Cohen (1995)). Assume now that the Fourier transform of the signal $s(t)$ is $S(\omega) = B(\omega)e^{j\psi(\omega)}$. Then the first derivative of the phase $\psi'(\omega)$ is called the group delay function. For the Wigner distribution, the following holds:

$$\mathbb{E}[\omega]_s = \frac{\int t \; \mathcal{W}(t, \omega) dt}{\int W(t, \omega) \; dt} = \frac{1}{|S(\omega)|^2} \int t \; \mathcal{W}(t, \omega) \; dt = -\psi'(\omega) \tag{2.76}$$

  which states that the conditional mean time of the Wigner distribution is equal to the group delay (Cohen (1995)). The above results are quite relevant since they are always true for any given signal. This is not true in the case of the STFT.

- **Local Spread**
  According to the obtained result, the instantaneous frequency is the conditional average for a particular time. Now, the spread of that average is taken into account, corresponding to the conditional standard deviation. Consider the second conditional moment in frequency

$$\begin{aligned} \mathbb{E}[\omega^2]_t &= \frac{1}{|s(t)|^2} \int f^2 \; \mathcal{W}(t, \omega) \; d\omega \\ &= \frac{1}{2} \left[\left(\frac{a'(t)}{a(t)} \right)^2 - \left(\frac{a''(t)}{a(t)} \right) \right] + \theta'^2(t) \end{aligned} \tag{2.77}$$

  where $a(t)$ is the amplitude of the signal. The conditional spread frequency is

$$\begin{aligned} \sigma^2_{\omega|t} &= \mathbb{E}[\omega^2]_t \; - \; \mathbb{E}[\omega]_t^2 \\ &= \frac{1}{2} \left[\left(\frac{da(t)/dt}{a(t)} \right)^2 - \frac{d^2a(t)/dt}{a(t)} \right] \end{aligned} \tag{2.78}$$

Such an expression could be negative and hence cannot be properly interpreted. Therefore, while the Wigner distribution gives an excellent result for the average conditional frequency, it gives a very poor one for the spread of those frequencies.

- **Cross-Terms**
  The Wigner distribution produces cross-terms representing significant oscillating terms located in the middle between the signal components. Furthermore, they can be twice as large as the different signal components regardless of how far apart are all the signal components. This makes the Wigner distribution highly not suitable for non-toy signals. Consider a two-component signal defined as $s(t) = s_1(t) + s_2(t)$ for which the Wigner distribution is

$$\mathcal{W}_s(t,\omega) = \mathcal{W}_{s_1}(t,\omega) + \mathcal{W}_{s_2}(t,\omega) + 2\Re\left[W_{s_1,s_2}(t,\omega)\right] \tag{2.79}$$

  where $\mathcal{W}_{s_1}(t,\omega)$ and $\mathcal{W}_{s_2}(t,\omega)$ are called auto-terms and corresponds to the Wigner distributions of $s_1(t)$ and $s_2(t)$ respectively. The term

$$2\Re\left[\mathcal{W}_{s_1,s_2}(t,f)\right] = 2\Re\left[\mathcal{F}\left[s_1(t+\frac{1}{2}\tau)s_2^*(t-\frac{1}{2}\tau)\right]\right] \tag{2.80}$$

  is called cross-term. This term will always be present, located midway between the two auto-terms and oscillating proportionally to the distance between the auto-terms. The direction of the oscillation is orthogonal to the line connecting the auto-terms. Therefore, the Wigner distribution will always produce a cross-term between each pair of component. Furthermore, they might also adopt negative values which could also be misleading in the interpretation.

Definitions and properties of the Wigner distribution have been presented in this subsection. The main argument in favour of this transform over the spectrogram is that no window has to be chosen. However, it is essential to highlight that the spectrogram is not one distribution. Instead, it is an infinite class of distributions whose disadvantage is represented by selecting the "right" window, which is not a non-trivial task. Therefore, the Wigner distribution is, in these respects, better than any spectrogram. Furthermore, the Wigner distribution gives a consistent picture of both the instantaneous frequency and the group delay. This is never the case for the spectrogram, even if good approximations could be achieved. Besides, the Wigner distribution satisfies the marginals and always gives the correct answers for averages of functions of frequency or time and always satisfies the uncertainty principle of the signal. The spectrogram, instead, never gives correct answers for these averages and never satisfies the signal's uncertainty principle. The major drawback of the Wigner distribution is identified as the cross-term interference. At each time, if there is more than one frequency existing, then the Wigner distribution might generate undesired terms. Nevertheless, these terms are localized and hence occur in the midway

of the pair of corresponding auto-terms. On the other hand, the spectrogram resolves the components in some instances and is also very easy to interpret.
In the following subsection, the introduced time-frequency methods are compared in terms of their time-frequency resolution to better capture their different approaches and understand that many of the issues will be resolved through the Empirical Mode Decomposition presented in the following Chapter.

2.4 The Time-Frequency Resolutions of the Different Transforms

If measured in time, a signal $s(t)$ would describe its amplitude changes over the domain of the variable time t. When the frequency becomes the variable of interest instead, the Fourier transform can be applied. The obtained spectrum would then contain complete information in the frequency domain in terms of magnitudes and phases of the frequency component at any given frequency. However, the issue is that no explicit information would be available in the spectrum regarding temporal characteristics of the signal, i.e. when a specific frequency has occurred.

One of the objectives of the introduced transforms is tackling this problem and achieving a partition for the time-frequency plane that will most effectively capture the properties of the original signal in both domains, hence proposing an optimal time-frequency resolution. What happens in practice is that it is impossible to increase both temporal and frequency resolutions. When one is improved, the other must suffer. This phenomenon, previously discussed in this thesis, is known as the uncertainty principle and comes from the Heisenberg Uncertainty Principle encountered in quantum physics. It states that it is impossible to precisely measure both the position and the momentum of a microscopic particle simultaneously.

In Figure 2.1 taken from Scholl (2021), the time-frequency resolutions of the different transforms are presented. The resolutions can be controlled by the window length of the transform of interest. In general, a short window would capture a short period and, therefore, has a precise time resolution. However, the frequency resolution would be poor since the considered signal would contain few samples corresponding to only a few frequency bins and would not provide enough information in this respect.

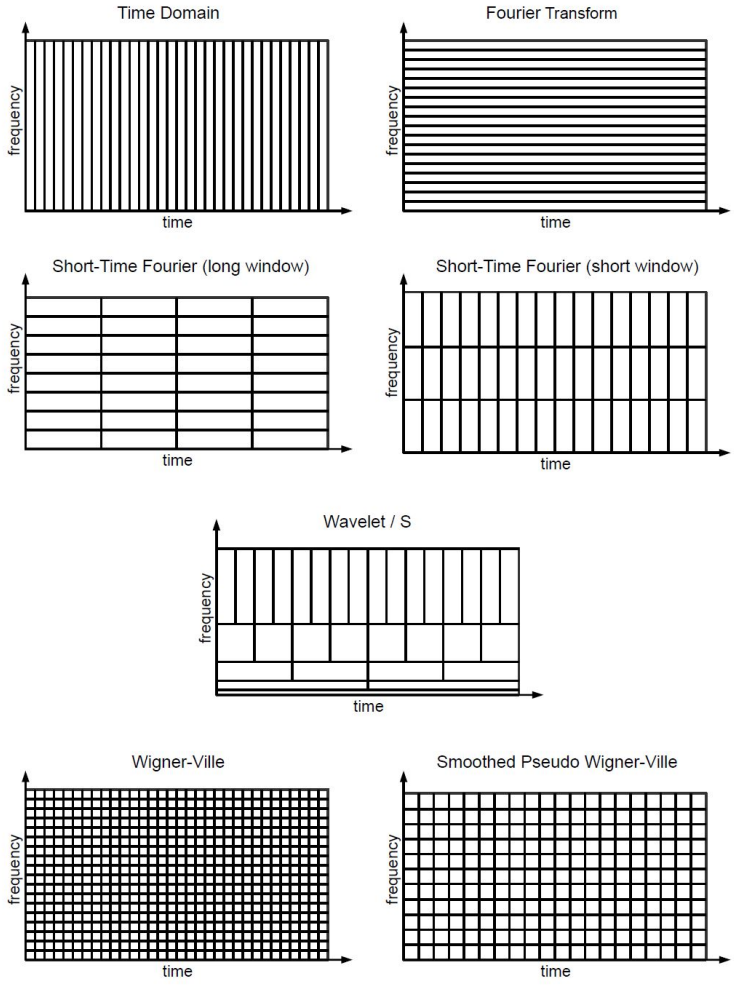


Figure 2.1: Figure showing a schematic overview of the time and frequency resolutions of the different transforms introduced in comparison with an original time-series dataset. The figure is taken from Scholl (2021). For the Smoothed Pseudo Wigner-Ville distribution the reader might refer to this paper.

On the contrary, a long window would provide a poor time resolution but offers precise frequency information due to the large number of samples that can be captured. This corresponds indeed to the uncertainty principle stating that the product of the resolution in time and frequency is limited such that $BT \geq 1/4$ (with B being the bandwidth of one frequency bin). The size and orientation of the blocks indicate how the windows of the transforms describe the time and frequency domains. The original time-series has a high resolution in the time-domain but zero resolution in the frequency one and the Fourier transform, which instead has a high resolution in the frequency domain and zero resolution in the frequency domain time-domain. By looking at the STFT, the trade-off between the two resolution using either a long or small window is presented. The continuous wavelet transform has for small frequency values a high resolution in the frequency domain and a low resolution in the time domain. For large frequency values instead has a low resolution in the frequency domain and high

resolution in the time domain. Hence the CWT makes a trade-off overcoming the STFT drawbacks. The idea behind that is that a great majority of the real-world signals have slowly oscillating content occurring on long scales, while high-frequency content tend to happen on a short scale. As an example, the human auditory system works this way. However, if there were natural phenomena for which the high-frequency events were long, then the CWT would not be an appropriate choice.
The Wigner distribution instead overcomes the limited resolutions of the STFT and the wavelet transform by offering a very high resolution in both time and frequency domains. The problem is given by its cross-terms representing artifacts occurring in the presence of multicomponent signals resulting from the Wigner distribution being a quadratic transform.

The Wigner distribution appears to provide a finer time-frequency resolution amongst the ones proposed in this Chapter. Nevertheless, if the analysed signal is multi-component, cross-terms might mislead its interpretation and present unreliable results. As a response, several existing time-frequency methods have been proposed to suppress the cross-terms and maintain the concentration of the auto-terms. Hence, measurement criteria for concentration are required. One way is to consider the definition of instantaneous frequency and instantaneous phase for the analytic signal or any mono-component complex-valued signal. Different solutions can be considered in this respect, exploring different principles as the reassignment principle (Kodera et al. (1976)) or synchrosqueezing (Daubechies et al. (2011)). However, one of the most relevant and modern methods dealing with the computation of the instantaneous frequency is represented by the Empirical Mode Decomposition (Huang et al. (1998)). At this stage, it is essential to highlight that this technique provides an adaptive time-frequency resolution strongly depending on the underlying data. This is the primary reason for selecting such a methodology as the based method of this thesis. In the following Chapters, statistical aspects of the EMD are studied along with a stochastic embedding, further developing a more refined framework partitioning the time-frequency plane through a novel methodology.

Chapter 3

Methodology: The Empirical Mode Decomposition

One of the common characteristics of all the methodologies mentioned so far is using bases that are a priori defined. Most of them are built to analyse non-stationary but linear data or non-linear but stationary data. The employment of such decomposition methods highly often provides components that, even by carrying a mathematical interpretation, lack physical meaning with the added issue of harmonic distortions proper of the Fourier transform.

An alternative approach developed in data analysis by Huang et al. (1998) is the Hilbert-Huang transform. It is an a posteriori, data-driven and adaptive basis decomposition time-frequency methodology applied to many areas, such as engineering, biomedical, financial or geophysical datasets. This transform was initially introduced to analyse water surface wave evolution in Huang et al. (1998); specifically, it has been employed to observe distorted waves and their variations occurring over time. Therefore, the need for a time-frequency spectrum dealing with both non-stationary and non-linear data was highly fundamental.

The EMD reproduces a signal along with its physical meaning by considering the concept of instantaneous frequency. The main issues related to the definition of such a notion have been introduced in section 1.2. By making data analytic, the Hilbert transform is an efficient tool to compute the instantaneous frequency. The idea behind such a transform is emphasizing local properties of a general signal $s(t)$ through its convolution with $1/t$. However, as explained in section 1.2, some controversies linked to the concept of multi-component signals are still found. The significant contribution of the Hilbert transform started to be central only after the introduction of the Empirical Mode Decomposition Huang et al. (1998). Along with the Hilbert transform, they were named Hilbert-Huang Transform.

The EMD is a form of basis decomposition typically considered for spatial or temporal signals, which has several advantages compared to the traditional Fourier and wavelet decompositions. Firstly, it is more appropriate in the presence of non-linear and non-stationary systems. Secondly, the specification of the basis

functions does not require any a priori parametric formulation. Indeed, the fluctuations are automatically and adaptively extracted from the signal, leading to recursive resolution of the basis functions. This is both advantageous in that basis functions are naturally adapted to a given signal but also disadvantageous as the basis functions must be non-parametrically specified in a functional form. Fortunately, there are numerous ways to achieve such representations based on statistical penalised spline representations.

The essential principle of EMD is to decompose signals into a sum of certain suitable oscillatory functions called intrinsic modes functions (IMFs). With comparison to wavelets or Fourier analysis, an IMF represents a simple oscillatory mode, like the simple harmonic function. However, it is more general: instead of constant amplitude and frequency, as in a simple harmonic component, the IMF can have a time-varying amplitude and frequency. The significance of this representation, therefore, lies in the ability to perform a locally adapted class of basis functions that will be suitable even in the context of non-stationary signals. It is important at this stage to distinguish between the concept of the EMD basis decomposition and the ability to construct such a decomposition.

The EMD method is based on the simple assumption that any time-series signal may have many different coexisting modes of oscillations that, when superimposed together, combine to reconstruct the original signal exactly. The particular form of modes of oscillation deconstructed in an EMD decomposition aims to produce the oscillatory functions called IMFs. Each IMF has the property that it contains the same number of extrema as zero-crossings. What is more, for each IMF, the oscillation will also be symmetric with respect to the " local mean". In other words, the classical definition of an IMF corresponds to any function verifying the following properties:

1. The number of extrema and the number of zero-crossings must either equal or differ at most by one.

2. The mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero.

It is also important to realize that the EMD decomposition produces a deterministic decomposition of the signal and not a stochastic decomposition typically considered in a time-series structure. This clearly has implications on the ability to forecast and construct statistically meaningful confidence intervals for such a decomposition.

The obtained IMFs have the property that they are ordered in their oscillatory index. Therefore, the EMD algorithm aims to produce IMFs by recursively reducing oscillation indices. The bases extraction procedure is an algorithm named sifting. One of the main issue related to it is that it may never converge to a proper IMF function at a given decomposition stage. This has led to a range of heuristic rules being developed to specify algorithms for terminating the search

for an IMF basis at a given stage, based on different forms of approximation. An overview of these heuristic rules is provided in this Chapter.

No matter the type of sifting algorithm adopted, they all have two common goals: eliminating riding waves and making the wave profiles (oscillations) more symmetric. While the first purpose serves the Hilbert transform to give a meaningful instantaneous frequency, the second makes the neighbouring wave amplitudes have a symmetric aspect, reducing the Hilbert transform envelope oscillation. That is why the sifting process must be repeated as many times as possible to reduce the extracted signal to an IMF.

Using the mean of envelopes is an alternative to calculating the signal's local mean, which is particularly suited to capture the structure of non-stationary signals. In this last case, the local mean involves a local time scale which is impossible to define a priori. Huang et al. (1998) suggested the use of the local mean of the envelopes defined by the local maxima and the local minima to force the local symmetry. This technique is empirical, but as it was highlighted by Huang et al. (1999), Huang et al. (1998), it should always produce a consistent instantaneous frequency.

The primary purpose of this Chapter is to present a formal mathematical definition of the EMD and the Hilbert transform when a cubic spline representation is used as parametric interpolation of the discrete path $s(t)$. The sifting procedure comprises several steps at which different choices developed in the literature could be taken. It is central to this Chapter to consider these proposed solutions and discuss how they affect the obtained decomposition.

This Chapter is organised as follows: firstly, a formal definition of the EMD is proposed. Afterwards, the extraction of the IMFs is described, where a proposition for the expression of the k-th identified IMF will be given with respect to the original signal. The concept of the instantaneous frequency is then presented, and the Hilbert transform of the IMF representation is formally introduced. A section on how to interpret the EMD basis decomposition and the instantaneous frequency is then provided to better understand the power of these basis functions. A section on some unexpected situations that might occur during the sifting procedure is then discussed. To this end, different steps of the sifting are then examined, i.e. the envelope boundary construction, the spline considered for the interpolation of the signal, the basis functions and the envelopes, the stopping criteria and the extrema detection.

3.1 EMD Formal Definition

Assume a continuous non-stationary signal is partially or discretely observed in time. The signal $s(t)$ is observed at $0 = t_1 < \cdots < t_N = T$. For the EMD to exists, the partially observed discrete signal $s(t)$ needs is converted into a continuous representation; therefore, the discrete signal $s(t)$ is converted back into a continuous analog signal denoted $\tilde{s}(t)$; in this case a semi-parametric

model known as a natural cubic spline is used, given in equation (3.1). As a consequence, the EMD bases denoted as $\{\gamma_l(t)\}_{l=1}^{L}$ will also be expressed as natural cubic splines, derived from representation $\tilde{s}(t)$.

Definition 3.1.1. *Given a set of l knots $a = \tau_1 < \tau_2 < \cdots < \tau_l = b$, a function $\tilde{s} : [a, b] \to \mathbb{R}$ is called a cubic polynomial spline if:*

- *$\tilde{s}(\cdot)$ is a polynomial of degree 3 on each interval (τ_j, τ_{j+1}) $(j = 1, \ldots, l-1)$*
- *$\tilde{s}(\cdot)$ is twice continuously differentiable*

It is then a natural cubic spline when $\tilde{s}''(a) = \tilde{s}''(b) = 0$.

Hence, the signal representation $\tilde{s}(t)$ is expressed in the class of truncated power basis, where the knot points are placed at the sampling times $(\tau_i = t_i)$

$$\tilde{s}(t) = a_0 + a_1 t + a_2 t^2 + a_3 \left(t - \tau_1\right)_+^3 + \cdots + a_{3+l-2} \left(t - \tau_{l-1}\right)_+^3 . \tag{3.1}$$

The coefficients are estimated by standard penalised least squares

$$\sum_{i=1}^{N-1} \left(s(t_i) - \tilde{s}(t_i)\right)^2 + \lambda \int_{t_i}^{t_{i+1}} \tilde{s}''(t)^2 dt \tag{3.2}$$

with natural cubic spline constraints $\tilde{s}''(0) = \tilde{s}''(t_N) = 0$ and where $\lambda > 0$ controls smoothness of the representation. In this case, the number of total convexity changes (oscillations) of the analog signal $\tilde{s}(t)$ within the time domain $[0, t_N]$ is denoted by to $L \in \mathbb{N}$. One may now define the EMD decomposition of a speech signal $\tilde{s}(t)$ as follows.

Definition 3.1.2. *The Empirical Mode Decomposition of signal $\tilde{s}(t)$ is represented by the finite number of non-stationary basis functions known as Intrinsic Mode Functions (IMFs), denoted by $\{\gamma_l(t)\}$, such that*

$$\tilde{s}(t) = \sum_{l=1}^{L} \gamma_l \left(t\right) + r \left(t\right) \tag{3.3}$$

where $r\left(t\right)$ represents the final residual (or final tendency) extracted, which has only a single convexity. In general the γ_l basis will have l-convexity changes throughout the domain (t_1, t_N) and each IMF satisfies:

- ***Oscillation*** *The number of extrema and zero-crossing must either equal or differ at most by one:*

$$abs\left(\left|\left\{\frac{d\gamma_l(t)}{dt} = 0 : \quad t \in (t_1, t_N)\right\}\right| - |\{\gamma_l(t) = 0 : \quad t \in (t_1, t_N)\}|\right) \in \{0, 1\} \tag{3.4}$$

- ***Local Symmetry*** *The local mean value of the envelope defined by a spline through the local maxima denoted $\tilde{s}^{U_l}(t)$ and the envelope defined by a spline through the local minima denoted by $\tilde{s}^{L_l}(t)$ is equal to zero pointwise i.e.*

$$m_l(t) = \left(\frac{\tilde{s}^{U_l}(t) + \tilde{s}^{L_l}(t)}{2} \right) I\left(t \in [t_1, t_N]\right) = 0 \tag{3.5}$$

The minimum requirements of the upper and lower envelopes are:

$$\begin{aligned}
&\tilde{s}^{U_l}(t) = \gamma_l(t), \quad if \quad \frac{d\gamma_l(t)}{dt} = 0 \quad \& \quad \frac{d^2\gamma_l(t)}{dt^2} < 0, \\
&\tilde{s}^{U_l}(t) \geq \gamma_l(t) \quad \forall t \in (t_1, t_N) \\
&\tilde{s}^{L_l}(t) = \gamma_l(t), \quad if \quad \frac{d\gamma_l(t)}{dt} = 0 \quad \& \quad \frac{d^2\gamma_k(t)}{dt^2} > 0, \\
&\tilde{s}^{L_l}(t) \leq \gamma_l(t) \quad \forall t \in (t_1, t_N).
\end{aligned} \tag{3.6}$$

Note that each IMF carries a unique number of convexity changes that can occur on any time spacings and not cyclically with a definite period as in a Fourier basis unless that signal is indeed stationary. Typically, the times of convexity change are irregularly spaced and reflect non-stationarity in a local bandwidth of the frequencies, that characterise the signal at that time instant. As a result of this property, one can still order the basis IMFs naturally according to the unique number of total convexity changes they produce in (t_1, t_N). As underlined in Huang et al. (1998), the construction of an IMF basis is directly linked to the concept of local symmetry, required to handle non-stationary data. This notion is enclosed by the mean envelope that captures a local time scale and the definition of a local averaging time scale is hence bypassed. Such a requirement is fundamental to avoid asymmetric waves affecting the concept of instantaneous frequency, of which, we mathematically formalise the definition for below. Further, note that, in the above representation, $\gamma_l(t)$ is not explicitly expressed in a functional form, as opposed to classical stationary methods where a parametric family of basis functions are stated, such as a cosine basis or a wavelet basis function. Here, the basis can take any functional form so long as it satisfies the decomposition relationship and the properties stated for an IMF. A natural way to proceed to represent an IMF is to utilise a smooth, flexible characterisation that can adapt to local non-stationary time structures; the one selected in this work to represent $\gamma_l(t)$ is the cubic spline.

3.2 Extraction of EMD Basis Functions (IMFs)

Given a mathematical representation for the IMFs, the process applied to extract recursively the IMF spline representations is now outlined. This procedure is known as *sifting*. The first step consists of computing extrema of $\tilde{s}(t)$; this can be done based on observations or on the interpolated signal, $\tilde{s}(t)$. Note, if there is noise in the signal it may also be advantageous to apply a penalised spline

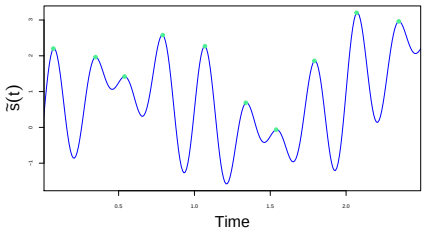
to obtain $\tilde{s}(t)$. In terms of determining the maximum and minimum convexity changes of $s(t)$, the use of $\tilde{s}(t)$ is advocated. Using $\tilde{s}(t)$, the roots of the first derivative $\tilde{s}'(t)$ produce the sequence of time points for successive maxima and minima:

$$\left\{t_j^*\right\}_{l=1}^{L} = \left\{t \in [t_1, t_N] a_1 + 2a_2 t + 3 \sum_{i=3}^{3+l-2} a_i \left(t - \tau_1\right)_+^2 = 0\right\}. \tag{3.7}$$

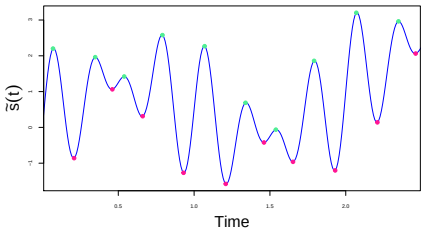
Without loss of generality, the maxima occur at odd intervals, i.e. t_{2j+1}^*, and minima occur at even intervals, i.e. t_{2j}^*. The second step of sifting builds an upper ($\tilde{s}^{U_l}(t)$) and lower ($\tilde{s}^{L_l}(t)$) envelope of $\tilde{s}(t)$ using two natural cubic splines through the sequence of maxima and the sequence of minima respectively:

$$\begin{aligned} \tilde{s}^{U_l}(t) &= a_0^{U_l} + a_1^{U_l} t + a_2^{U_k} t^2 + \sum_{i=0}^{\lfloor L/2 \rfloor} a_{i+3}^{U_k} \left(t - t_{2i+1}^*\right)_+^3, \\ \tilde{s}^{L_l}(t) &= a_0^{L_l} + a_1^{L_l} t + a_2^{L_l} t^2 + \sum_{i=0}^{\lfloor L/2 \rfloor} a_{i+3}^{L_l} \left(t - t_{2i}^*\right)_+^3, \end{aligned} \tag{3.8}$$

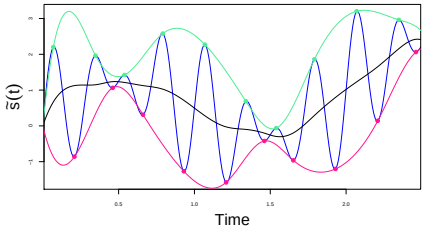
such that $\tilde{s}^{U_l}(t_{2j+1}^*) = \tilde{s}(t_{2j+1}^*)$ for all odd t_j^* and $\tilde{s}^{U_l}(t) \geq \tilde{s}(t)$ and equivalently $\tilde{s}^{L_l}(t_{2j}^*) = \tilde{s}(t_{2j}^*)$ for all even t_j^* and $\tilde{s}^{L_l} \leq \tilde{s}(t)$. One then utilises these envelopes to construct the mean signal denoted by $m_l(t)$ given in equation (3.5), which will then be used to compensate the original speech signal $\tilde{s}(t)$ in a recursive fashion, until an IMF is obtained. These bases are recursively extracted, this means that, once the l-th IMF is computed, it is subtracted from the main signal and the sifting procedure is applied to the residual signal to obtain the next IMF which will have one less convexity changes than the previously extracted IMF on (t_1, t_N). The procedure is detailed in the following sections of this Chapters along with the stopping criteria (details are also discussed in Dalpiaz, Rubini, D'Elia, Cocconcelli, Chaari, Zimroz, Bartelmus, Haddar et al. (2013)). Next, an illustration of the sifting process for IMF basis extraction is given in Figure 3.1.



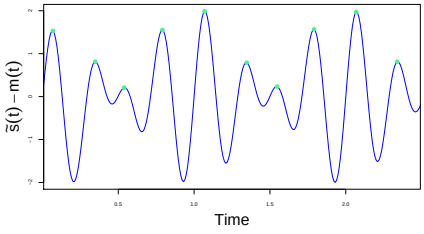
(a) Computing maxima of $\tilde{s}(t)$.



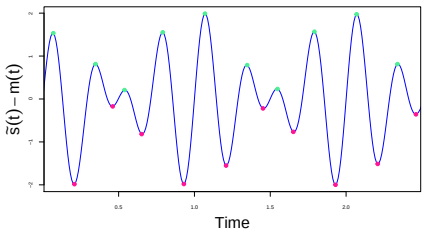
(b) Computing minima of $\tilde{s}(t)$.



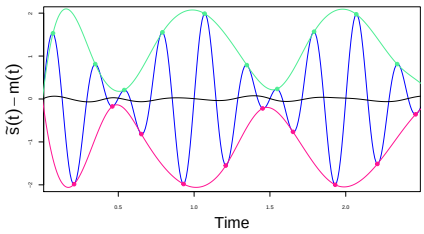
(c) Computing $\tilde{s}^{U_l}(t)$,$\tilde{s}^{L_l}(t)$, $m_l(t)$



(d) Computing $\tilde{s}(t) \leftarrow \tilde{s}(t) - m(t)$. Computing maxima of $\tilde{s}(t)$.



(e) Computing minima of $\tilde{s}(t)$.



(f) Computing $\tilde{s}^{U_l}(t)$,$\tilde{s}^{L_l}(t)$, $m_k(t)$.

Figure 3.1: Initial steps of the sifting procedure. This procedure continues until an IMF $\gamma_l(t)$ is identified.

It is often the case that such an algorithm does not reach a mean level equal to 0, which would indicate the termination of sifting for a given IMF. Multiple solutions in the literature have been proposed as stopping criteria of the sifting procedure (Dalpiaz, Rubini, D'Elia, Cocconcelli, Chaari, Zimroz, Bartelmus, Haddar et al., 2013) dealing with this computational aspect. From the sifting process, it is clear that these bases are recursively extracted; this means that, once the l-th IMF is computed, it is subtracted from the main signal, and the sifting procedure is applied to the residual signal to obtain the subsequent IMF which will have one less convexity change than the previously extracted IMF on (t_1, t_N). Hence, it is useful to develop recursive parameter estimation approaches to understand the linking relationship between the coefficients of two successive extracted IMFs. By exploiting the definition of cubic spline used in the representation of the analog speech signal $\tilde{s}(t)$ and the IMF basis functions, one can obtain a mathematical connection between the coefficients of $\tilde{s}(t)$ and the coefficients of $\gamma_l(t)$ detailed as follows:

Proposition 3.2.1. *The l-th extracted IMF denoted as $\gamma_l(t)$ can be expressed as a cubic spline whose coefficients are a linear combination of the spline coefficients of $\tilde{s}(t)$ and the coefficients of the $l-1$ IMFs extracted until such point of the sifting procedure and the coefficients of its mean envelopes, i.e.*

$$\gamma_l(t) = \tilde{s}(t) - \sum_{j=1}^{l-1} \gamma_j(t) - m_l(t) = \sum_{i=1}^{N-1} \left(a_i^l t^3 + b_i^l t^2 + c_i^l t + d_i^l \right) \mathbb{1}\left(t \in [t_{i-1}, t_i]\right) \quad (3.9)$$

where the spline coefficients are given as follows:

$$a_i^l = a_i - \textstyle\sum_{j=1}^{l-1} a_i^j - \frac{1}{2}(a_i^{U_l} + a_i^{L_l}) \qquad c_i^l = c_i - \textstyle\sum_{j=1}^{l-1} c_i^j - \frac{1}{2}(c_i^{U_l} + c_i^{L_l})$$

$$b_i^l = b_i - \textstyle\sum_{j=1}^{l-1} b_i^j - \frac{1}{2}(b_i^{U_l} + b_i^{L_l}) \qquad d_i^l = d_i - \textstyle\sum_{j=1}^{l-1} d_i^j - \frac{1}{2}(d_i^{U_l} + d_i^{L_l})$$

Such a proposition expresses the EMD construction of an IMF by considering the outer loop steps of the described algorithm. This means that, by looking at Algorithm 8, the proposition considers steps 1-3 to prove the statement. The proof is given in Appendix A. Remark: it is important to highlight at this stage that in the above representation N points of segmentation ha been considered to evaluate each $\gamma_l(t)$; however, the segmentation considered for the envelopes is given by the number of oscillations points L ($L << N$), which reduces by one for each IMF. Such segmentation is a subset of the t sequence of points $\{t_i\}_{i=1:N}$ considered to construct each IMF (which correspond to the t of the original spline $\tilde{s}(t)$) and will be denoted as $\{\tau_i\}_{i=1:L}$ so that it will be directly related to each IMF. Note that $t_1 = \tau_1$ and $t_N = \tau_L$.

3.3 Instantaneous Frequency

This section provides an understanding of the concept of instantaneous frequency related to each of the IMF's obtained from the EMD methodology. Classical Fourier methods require stationarity, where the frequencies of basis components are pure harmonics that are static over time Huang et al. (1998). Nevertheless, signals are often non-stationary and non-linear and, therefore, carry time-varying frequency components, which will be reflected in their IMFs. Though it is possible to have time-varying coefficients Fourier methods (Cohen, 1995) which tend to capture non-stationarity with fix basis, the EMD provides more flexibility. By being a data-driven, a posteriori method, its basis, i.e. the IMFs, are indeed more flexible in their ability to capture both non-stationary amplitudes and frequencies. Hence, IMFs will admit a time-varying frequency structure that can be characterized by instantaneous frequencies (IFs).
The IF of a given IMF basis is extracted in the following stages. First one takes the Hilbert Transform of each $\gamma_l(t)$, so that an analytic extension of the given IMF can be constructed. The Hilbert Transform can be computed in closed form readily if $\gamma_l(t)$ respects the restrictions defined in (3.6). Define $z_l(t) = \gamma_l(t) + \jmath\check{\gamma}_l(t) = a_l(t)e^{\jmath\theta_l(t)}$ the analytic extension of $\gamma_l(t)$ with time varying amplitude

$a_l(t) = \sqrt{\gamma_l^2(t) + \check{\gamma}_l^2(t)}$ and time varying phase $\theta_l(t) = \arctan \frac{\tilde{\gamma}_l(t)}{\gamma_l(t)}$. Then $\check{\gamma}_l(t)$ is obtained via Hilbert Transform as follows:

$$\check{\gamma}_l(t) = \frac{1}{\pi} \lim_{\epsilon \to \infty} \int_{-\epsilon}^{+\epsilon} \frac{\gamma_l(\tau)}{t - \tau} d\tau \tag{3.10}$$

The instantaneous frequency $\omega_l(t)$ for IMF l is then found from $z_l(t)$:

$$\omega_l(t) = \frac{1}{2\pi} \frac{d\theta_l(t)}{dt} = \frac{1}{2\pi} \frac{\check{\gamma}_l'(t)\gamma_l(t) - \check{\gamma}_l(t)\gamma_l'(t)}{\gamma_l^2(t) + \check{\gamma}_l^2(t)}. \tag{3.11}$$

We see that Huang et al. (1998) imposed the conditions (3.6) characterizing the IMFs properties to then ensure that the instantaneous frequency remains positive and therefore admits a meaningful physical interpretation. It will be advantageous to obtain the Hilbert transform of the l-th IMF by considering the natural cubic spline representation per knot segmentation as a local cubic polynomial for $t \in [\tau_{i-1}, \tau_i]$. Then the Hilbert transform is constructed as the following sum of local cubic polynomial transforms, see for details el Malek and Hanna (2020):

$$\check{\gamma}_l(t) = \mathcal{HT}\left[\gamma_l(\tau)\right] = \frac{1}{\pi} \sum_{i=1}^{N-1} \check{\gamma}_{l_i}(t) \quad \tau_{i-1} < t \leq \tau_i \tag{3.12}$$

where $\triangle_i = \tau_i - \tau_{i-1}$ and $\check{\gamma}_{l_i}(t)$ is the Hilbert transform of the i-th polynomial:

$$\begin{aligned} \check{\gamma}_{l_i}(t) &= \left(a_{l_i} t^3 + b_{l_i} t^2 + c_{l_i} t + d_{l_i}\right) \log\left(\frac{t}{t - \triangle_i}\right) \\ &+ a_{l_i}\left(\frac{\triangle_i^2 t}{2} - \triangle_i t^2 - \frac{\triangle_i^3}{3}\right) + b_{l_i}\left(-\triangle_i t - \frac{\triangle_i^2}{2}\right) - c_{l_i}\triangle_i. \end{aligned} \tag{3.13}$$

The instantaneous frequency is performed per IMF so that it is possible to understand the local frequency and how it varies over time with each basis. To provide such concept in the context of non-stationary signals, Huang et al. (1998) needed to detect local structures of the data by assuming eqns. 3.6. If such conditions of the IMFs are not satisfied, the instantaneous frequency often assumes negative values which lack physical meaning. As in the Fourier methods, where a natural ordering of the static frequency (phase) for each basis exists, in this case, although the frequencies are time-varying, the extraction of the IMFs and property of the IMFs will still preserve the ordering in time of the instantaneous frequency, where the ordering is obtained from the number of oscillations (convexity changes) in each IMF basis begin decreasing with IMF index. In the case of a periodic signal, this would be analogous to strict ordering on frequency of the basis components, however, interestingly here we observe that IMFs may have some time intervals in $(t1, t_N)$ where a high order IMF may have lower instantaneous frequency than a lower order IMF. Assume that the interpolated signal $\tilde{s}(t)$ can be decomposed into components respecting eqns. 3.6. After the EMD

and the HHT of the IMFs are computed, $\tilde{s}(t)$ can be expressed in a "Fourier-like" expansion as:

$$\begin{aligned}\tilde{s}(t) &= Re\left\{\sum_{l=1}^{L+1} a_l(t)\exp\{\jmath\,\theta_l(t)\}\right\} \\ &= Re\left\{\sum_{l=1}^{L+1} a_l(t)\exp\{\jmath\int_{t_1}^{t_N} 2\pi\omega_l(t)dt\}\right\}\end{aligned} \tag{3.14}$$

in which the residual $r(t)$ is included $(L+1)$. The index l refers to each IMF and $Re\{.\}$ denotes the real part of a complex quantity. This expansion was proposed in Huang et al. (1998). Note that the differences with the classical Fourier expansion are the amplitude a_l and the frequency ω_l which are time-varying. Under the classical Fourier expansion the signal representation takes the form for a truncated infinite series as in equation (3.15) given by:

$$\tilde{s}(t) = Re\left\{\sum_{l=1}^{L+1} a_l\exp\{\jmath\int_{t_1}^{t_N} 2\pi\omega_l dt\}\right\} \tag{3.15}$$

The figures below show the difference between the Argand diagrams of the HHT of first IMF of a simulated signal and the the 10-th harmonic of the Fourier series of the same signal (capturing the equivalent frequency). It is possible to observe how the IMF captures amplitudes and frequencies over time while, in the case of the classical Fourier Transform, this non-stationary variation is not detected.

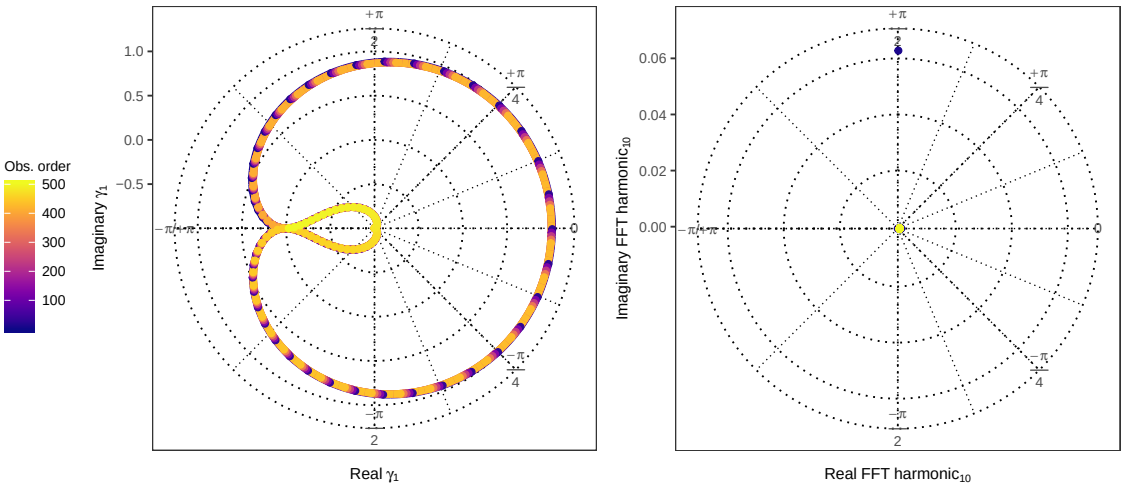


Figure 3.2: Argand diagrams of $\gamma_1(t)$ (left) and the 10-th harmonic (right) of the of the signal $y(t) = \cos(2\pi t)$ with $t \in (0, 10)$.

3.4 Interpreting EMD Basis Decomposition

In the case that the target signal is made of a finite number of pure stationary harmonics, the IMF decomposition will match the finite collection of Fourier bases as shown in the example in Figure 3.3. When the signal is not comprised of a finite number of pure harmonics or is non-stationary then the instantaneous frequencies for the IMF bases are not pure harmonics. However, the IMF bases that are extracted from EMD sifting decomposition can still be naturally ordered, but in a different manner to classical notions of frequency orders in Fourier analysis. They are ordered by oscillation count (total convexity changes), rather than frequency, this is not equivalent as the IMF bases are not in general strictly periodic. Due to this interesting difference, one may observe that IMFs may have some time intervals in (t_1, t_N) where a high order IMF may have lower instantaneous frequency than a lower order IMF, so long as over the entire interval it has greater number of convexity changes. Figure 3.4 presents an example of such a fact.

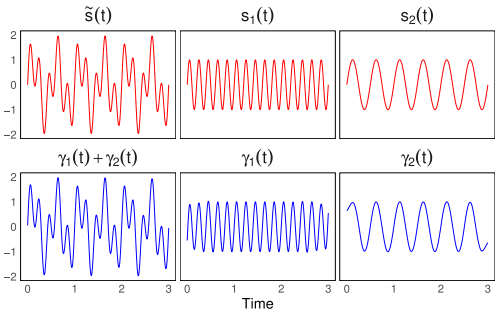


Figure 3.3: Top panels panels represent $\tilde{s}(t) = \sin(4\pi t) + \sin(10\pi t)$. Bottom panels provide the two IMFs basis functions.

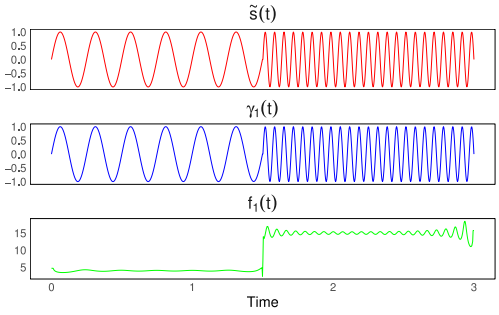


Figure 3.4: Top panel: signal $\tilde{s}(t) = \sin(4\pi t)\mathbb{I}[t \leq t_1] + \sin(15\pi t)\mathbb{I}[t > t_1]$. Middle panel: IMF extracted to represent $s(t)$ and Bottom panel: instantaneous frequency for IMF.

3.5 Some unexpected situations

Even though the EMD is more suitable in several applications than classical time-frequency analysis techniques, there are situations inherently linked to the

sifting procedure that have to be studied at a statistical level. By doing so, the stability of the decomposition should be more reliably achieved. This section aims to present some of these challenges and further motivate the following sections dealing with different aspects of the sifting procedure.

The first step of the sifting process is computing the envelopes through the extrema of the signal. Since the endpoints cannot be classified as maxima or minima, the resulting envelopes will present some distortions. This problem affecting the decomposition is well-known as the end effects. The following figure shows what happens to the borders when this problem comes into play.

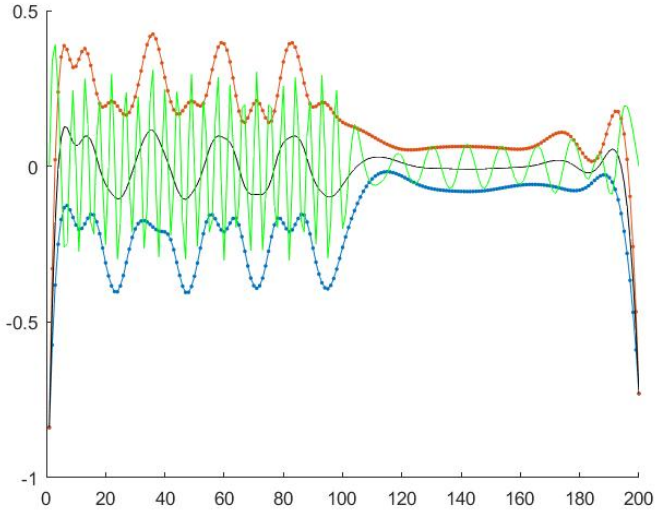


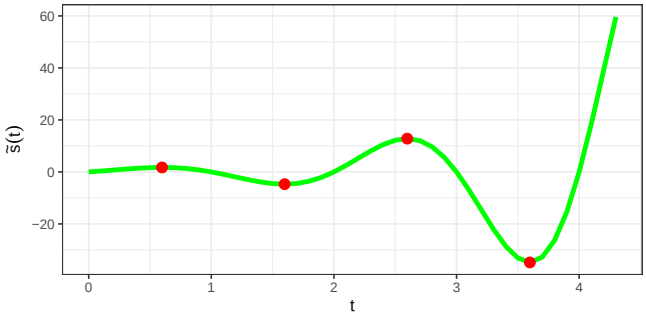
Figure 3.5: Example of the end effect problem.

What happens in practice is that the first and last samples have to be categorised as maxima or minima. They could be considered as such simultaneously or evaluated according to the nearest extremum to guarantee alternation. A third option would be leaving them as free (Huang et al., 1998). Several solutions have been implemented in the literature corresponding to signal expansion techniques based on symmetry or linear approximation that improves this issue (Massouleh and Kordkheili, 2019). The most common adopted are Rilling's mirror method (Rilling et al., 2003), Coughlin's method (Coughlin and Tung, 2004) and slope based method (Dätig and Schlurmann, 2004).

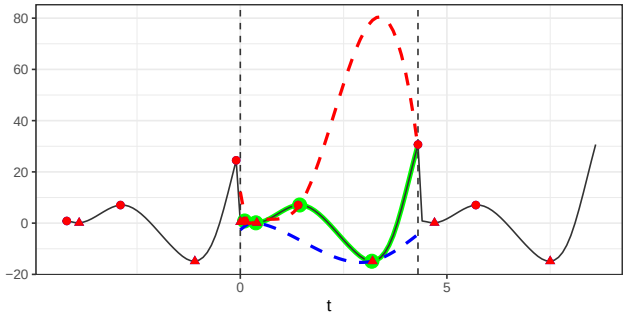
The mirror method adds a certain number of extrema before and after the border samples by mirroring the signal with respect to them. This technique is widely employed and maintain envelopes ends from divergence. Coughlin's method adds two sinusoids at the beginning and the end of the signal, whose amplitude and period correspond to the transverse difference and twice the longitudinal distance of the two neighbourhood extrema of each border, respectively. In the slope based method, two positive and negative line slopes between the first three extrema are computed (and the last three extrema equivalently). The method relies on the

fact that the distance between a new maximum and the first maximum equals the distance between the first two maxima, and the same reasoning applies to the minima. However, this method often leads to an error during the signal decomposition and border data loss (see Xiong et al. (2014) for further details).

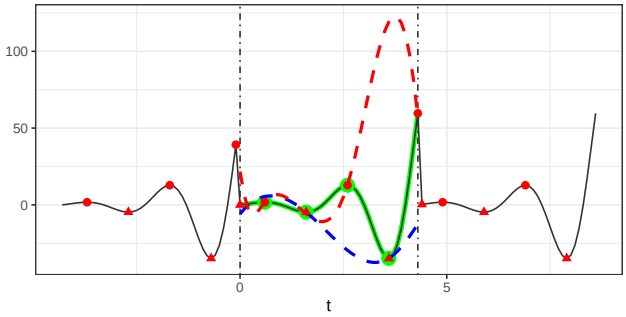
The experiments conducted in this thesis relies on the EMD R package (Kim and Oh, 2009). This package has multiple possibilities for the boundary conditions affecting the envelope constructions and hence the final decomposition. Particularly, it considers the idea proposed by Huang et al. (1998) extending the original signal by adding artificial waves repeatedly on both sides of the boundaries. These waves are usually referred to as "characteristic waves" and are obtained by repeating the implicit mode formed from extreme values nearest to the boundary. The three solutions considered in this thesis refer to the arguments named "wave", "periodic" and "symmetric" of the emd function of such a package. The first argument constructs a wave defined by two consecutive extrema at either boundary and adds four waves at either end. The periodic and the symmetric ones instead extend both boundaries periodically or symmetrically, respectively. Figure 3.6 shows the interpolated signal $\tilde{s}(t)$ in subfigure a) and then, subfigures b), c) and d) provides the envelopes obtained through these three solutions. Furthermore, Figure 3.7 presents the steps of the sifting procedure extracting the first IMF from the given signal. While the periodic solution seems to fail and provide undershoots and overshoots of the envelopes, the wave and symmetric boundary conditions perform efficiently in constructing the envelopes and extracting the first IMF. Given similar performances, the wave boundary condition has been selected for all the decomposition extracted in part III for all the experiments.



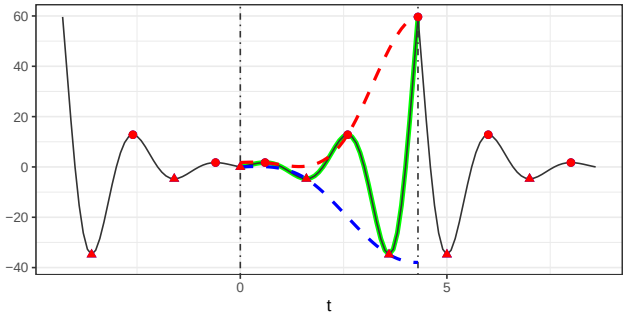
(a) Natural cubic spline $\tilde{s}(t)$ interpolating $s(t) = \exp(t)\sin(\pi t)$ with $t \in [0, 4]$. In red the extrema.



(b) Method "wave" for the boundary condition of the EMD R package.

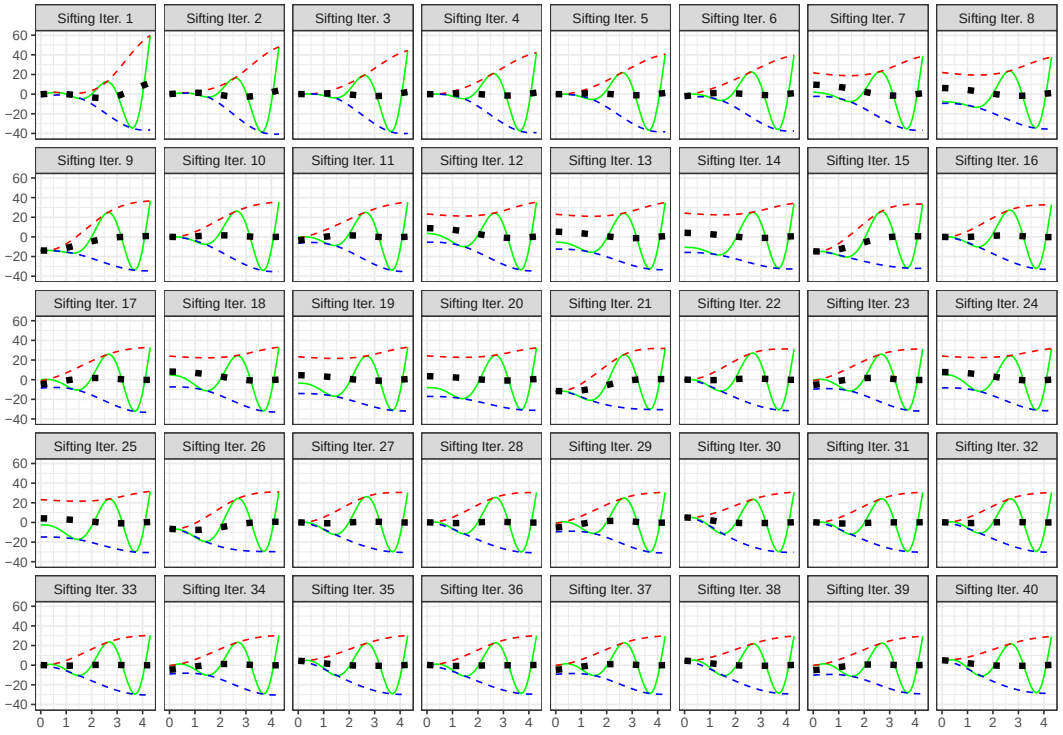


(c) Method "periodic" for the boundary condition of the EMD R package.

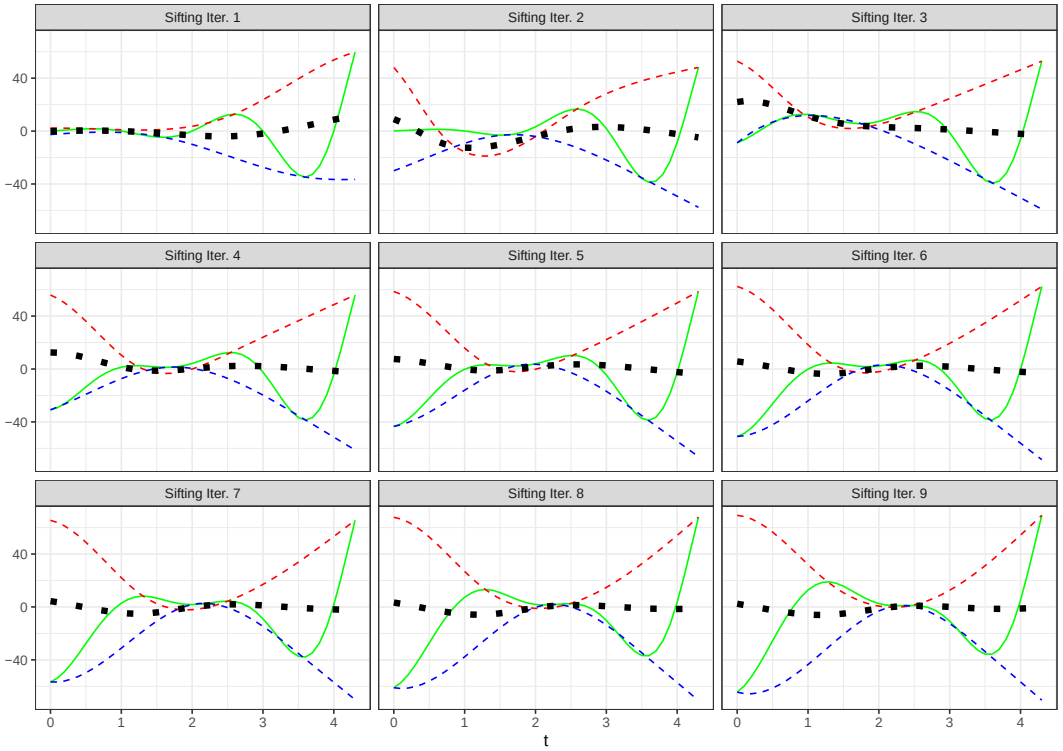


(d) Method "symmetric" for the boundary condition of the EMD R package.

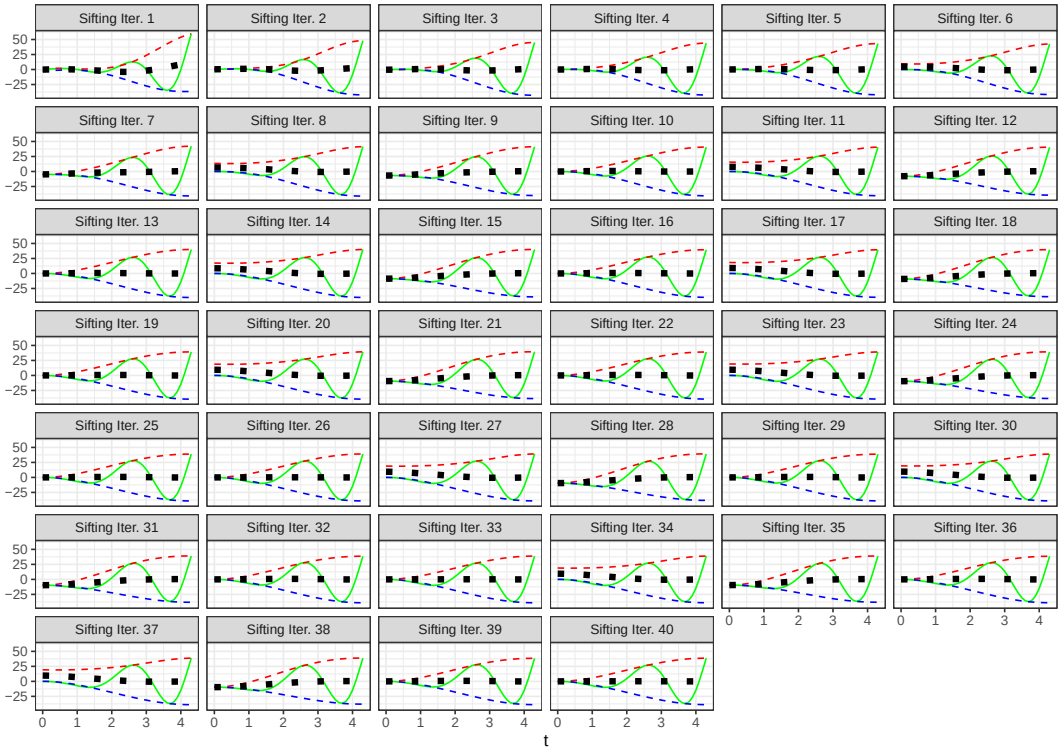
Figure 3.6: Different methods used for the boundary conditions affecting the end effects and the decomposition.



(a) Method "wave" for the boundary condition of the EMD R package.



(b) Method "periodic" for the boundary condition of the EMD R package.



(c) Method "symmetric" for the boundary condition of the EMD R package.

Figure 3.7: First steps of the sifting procedure applied to the signal provided in Figure 3.6 with the three solutions considered by the EMD R package for the boundary conditions named as wave, periodic and symmetric. Each subplots represents a sifting iteration done to extract the first IMF of the given signal.

The second main difficulty is the cubic spline fitting; its drawback is the production of overshoot or undershoot phenomena evidencing the incompleteness of the envelopes and so the unreliability of the IMFs. The following figure presents an example of this issue; it is possible to observe that the interpolated envelopes miss some part of the underlying signal at some points.

This issue also relates to the parametric representation selected to interpolate the original discrete samples $s(t)$ used to define $\tilde{s}(t)$. Section 3.7 presents different solutions which have been provided in the main literature for the EMD. In this thesis, the natural cubic spline is the one taken into account since optimal. Further discussion will be given below.

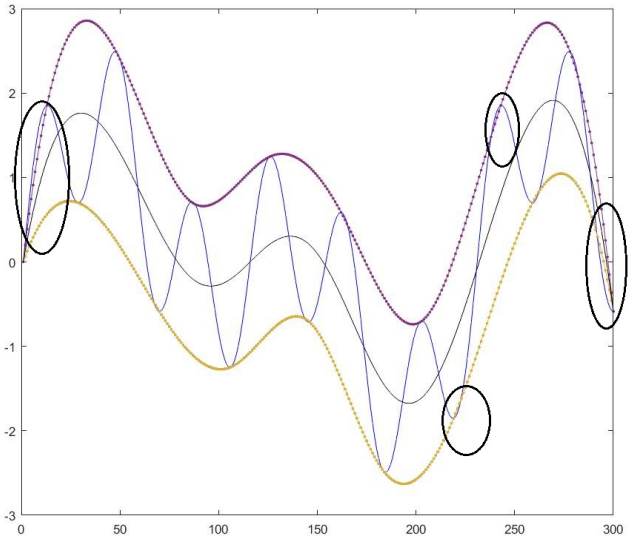
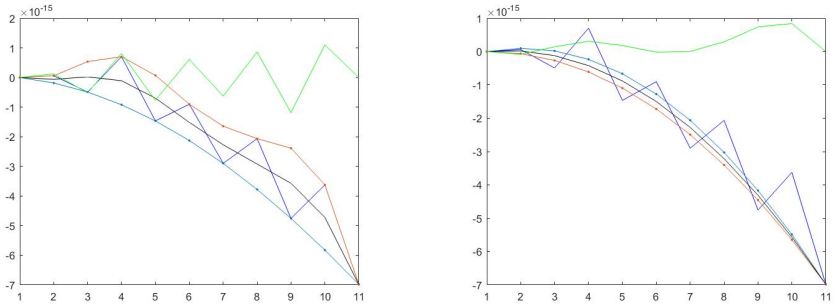


Figure 3.8: Undershoot and overshoot phenomena are shown by circles.

A further issue provided by the sifting technique is the following: after the averaging operation of the extrema envelopes, the resulting function may not be an IMF; this mainly depends on the fact that, according to the slope of the curve, some negative maxima and positive minima can appear after sifting. These unwanted phenomena are attenuated by repeating the procedure until having a satisfactory IMF.

As it was discussed in Huang et al. (1999), the time spacing of the extrema offers a better measure of time scale because it measures wide-band data with multiple riding waves. However, by examining data more closely, it can be observed that even the spacing of the extrema can miss some subtle time-scale variations, given weak oscillations that can cause a local change in curvature but cannot create a local extremum; this phenomenon is known as hidden scales.

Another critical issue is that the EMD only detects functions oscillating around the zero mean axis but fails with signals oscillating around a given shape. Figure 3.9 presents this problem: after the first extraction, the envelopes cannot embed the first IMF with the result that the sifting procedure stops and identifies a non-real residual. This kind of behaviour can be encountered in many natural phenomena, especially with non-stationary data.



(a) First IMF extraction. **(b)** Second IMF extraction.

Figure 3.9: Figure presenting an example of a function that does not oscillate around the zero mean, given in black. In green, the first two extracted IMFs are plotted. Note that the envelopes for the last iteration of the sifting procedure are also represented in each subplot.

After having introduced the central problems related to the sifting procedure of the Empirical Mode decomposition and having shown its main drawbacks, the following sections describe some alternatives affecting the procedure and the choice adopted within this thesis.

3.6 Stopping Criteria

The EMD decomposes a signal into the sum of IMFs. It is the sifting process that extracts the basis, and, by definition, it " sifts" many times to obtain a basis function. Its two primary purposes are eliminating riding waves and making the wave profile more symmetric with respect to zero. The mean of the upper and lower envelopes has to be zero by default. However, the average of the IMFs envelopes separated by the signal cannot be zero. Therefore, the more repeated the sifting process is, the closer to zero the average will be. At the same time, too many steps will make IMFs constant amplitude frequency-modulated functions. In that case, they would not embed any physical meaning, and, most likely, they would mix different frequencies.

Several authors proposed different solutions in the literature based on different principles. However, the obtained results end up failing, given the lack of a mathematical formulation for all of them. The main issue of these criteria is that they all use heuristic rules, and they are not related to the IMF definition. Therefore, it might stop too early with some frequencies that could be missed or too late, producing IMFs without any physical meaning.

The goals of this section are, firstly, describing some of the most used stopping criteria of the literature; secondly, showing how they affect the stability of the algorithm and, therefore, the final extraction of the IMFs.

The following stopping criteria are presented in Dalpiaz, Rubini, D'Elia, Cocconcelli, Chaari, Zimroz, Bartelmus, Haddar et al. (2013).

3.6.1 Cauchy-Type Convergence (SD)

The first stopping criterion that has been proposed is given by Huang et al. (1996). The idea is limiting amplitude and frequency modulations to obtain meaningful IMFs through a certain threshold for the standard deviation of two consecutive sifting results.

$$SD = \sum_{t=0}^{T} \left[\frac{\left| \left(\gamma_{l(h-1)}(t) - \gamma_{lh}(t) \right) \right|^2}{\gamma_{l(h-1)}^2(t)} \right] \tag{3.16}$$

where h indicates the number of times that the sifting procedure has been repeated, $\gamma_{lh} = \gamma_l$ is an IMF. Huang et al. (1998) provide 0.2 and 0.3 as suitable thresholds for this criterion; however, this method does not take into account the idea of IMF as a monocomponent function: its produced IMFs usually tend not to have such a property, and so mix different time scales.

3.6.2 Mean Fluctuations Threshold

To improve the above, the following criterion based on three different thresholds aimed to provide small fluctuations of the mean and, at the same time, considering locally large variations is taken into account. Three thresholds are defined : α, θ_1 and θ_2. For $(1-\alpha)$ data, the criterion keeps sifting if $\sigma(t) < \theta_1$, while for the remaining fraction $\sigma(t) < \theta_2$. The definition of $\sigma(t)$ is the "evaluation function" and is given by:

$$\sigma(t) := \frac{a(t)}{m(t)} \tag{3.17}$$

where $a(t)$ is the mode amplitude defined as:

$$a(t) := \frac{\left(\tilde{s}^{U_l}(t) - \tilde{s}^{L_l}(t) \right)}{2} \tag{3.18}$$

and $\tilde{s}^{U_l}(t)$ and $\tilde{s}^{L_l}(t)$ represents the upper and lower envelopes respectively. The usual number for these thresholds are $\theta_1 = 0.05$, $\theta_2 = 10\theta_1$ and $\alpha = 0.05$. The main issue in this case is that the thresholds do not adapt the signal, i.e. it is an heuristic rule that can or cannot fit the signal depending on its own features.

3.6.3 Energy Difference Tracking

This criterion assumes that the Empirical Mode Decomposition provides IMFs and residue mutually orthogonal. By considering a non-stationary signal $\tilde{s}(t)$ comprised of mutually irrelevant components as follows:

$$\tilde{s}(t) = \tilde{s}_1(t) + s_2(t) + ... + \tilde{s}_L(t) = \sum_{i=1}^{L} \tilde{s}_i(t) \tag{3.19}$$

By taking into account the total energy of the signal computed as

$$E_{\tilde{s}} = \int_{-\infty}^{\infty} \tilde{s}^2(t)\, dt = \int_{-\infty}^{\infty} \sum_{i=1}^{L} \tilde{s}_i(t)^2\, dt \tag{3.20}$$

By assuming orthogonality between them:

$$\int_{-\infty}^{\infty} \tilde{s}_i(t)\, \tilde{s}_j(t)\, dt = 0 \tag{3.21}$$

where i and j represent two different components. Therefore the total energy of the signal becomes:

$$\begin{aligned} E_{\tilde{s}} &= \int_{-\infty}^{\infty} \sum_{i=1}^{L} \tilde{s}_i(t)^2\, dt \\ &= \int_{-\infty}^{\infty} \tilde{s}_1^2(t)\, dt + \int_{-\infty}^{\infty} \tilde{s}_2^2(t)\, dt + ... + \int_{-\infty}^{\infty} \tilde{s}_L^2(t)\, dt \\ &= E_1 + E_2 + ... + E_L \end{aligned} \tag{3.22}$$

where E_i refers to the energy of the i-th components of the signal $\tilde{s}(t)$. By recalling that the EMD decomposition is given by:

$$\tilde{s}(t) = \sum_{l=1}^{L} \gamma_k(t) + r(t) \tag{3.23}$$

where $\gamma_l(t)$ is an IMF and $r(t)$ the residue or the mean trend of $\tilde{s}(t)$. Different IMFs $\gamma_1, \gamma_2, \gamma_3 ... \gamma_L$ incorporate different frequencies from high to low. If the EMD considers mutually orthogonal components, after having removed the first one, the energy of the residual signal is given by:

$$E_{2,..,L} = \int_{-\infty}^{\infty} \left[\sum_{i=2}^{L} \tilde{s}_i(t) \right]^2 dt \tag{3.24}$$

It is then possible to observe the following:

$$E_{tot} = E_1 + E_{2,..,L} = E_s \tag{3.25}$$

This is the sum of the energy of the first component separated by the signal together with the energy of the residual signal. Within its work, Junsheng et al. (2006) considers the case where a component $\gamma_1(t)$ (or IMF) is not orthogonal to the others. Then, when it is separated by the signal, the sum of its energy along with the residual signal will be given by:

$$\begin{aligned} E_{tot} &= \int_{-\infty}^{\infty} \gamma_1^2(t)\, dt + \int_{-\infty}^{\infty} \left(\tilde{s}(t) - \gamma_1(t)\right)^2 dt \\ &= E_{\gamma_1} + \int_{-\infty}^{\infty} \left(\tilde{s}^2(t) - 2\tilde{s}(t)\, \gamma_1(t) + \gamma_1^2(t)\right)^2 dt \\ &= 2E_{\gamma_1} + E_{\tilde{s}} - 2\int_{-\infty}^{\infty} \tilde{s}(t)\, \gamma_1(t)\, dt \end{aligned} \tag{3.26}$$

By supposing the following:

$$\gamma_1(t) = A\tilde{s}_i(t) + e(t) \tag{3.27}$$

where A is constant and $e(t)$ has been defined as the error component of $\gamma_1(t)$, then:

$$E_{tot} = E_x + 2\left(A^2 - A\right)E_i + 2E_e \neq E_x \tag{3.28}$$

where

$$E_e = \int_{-\infty}^{\infty} e^2(t)\,dt \tag{3.29}$$

Therefore, it is worth noting that if the decomposed signal is comprised of orthogonal components, then the energy of the original signal ($E_{\tilde{s}}$) equals the sum of the energy of the components (E_{tot}). In the case of a component that is not orthogonal there is going to be an error term defined as:

$$E_{err} = E_{tot} - E_{\tilde{s}} = 2\left(A^2 - A\right)E_i + 2E_e \tag{3.30}$$

Within their paper, Junsheng et al. (2006) explain that the sifting procedure will stop when $|E_{err}|$ reaches a certain minimum together with the mean value of the envelops that has to be small enough. However, the main issue of this criterion is given by the fact that IMFs generated by nonstationary and nonlinear signals are not orthogonal. Therefore, the identification of a threshold for $|E_{err}|$ is heuristic and strictly depending on the signal.

3.6.4 Orthogonality Criterion

Another method that exploits the concept that IMFs should be mutually orthogonal has been proposed by Lin and Hongbing (2009). They underline the fact that generally an IMF should satisfy the following;

$$\sum_{t=1}^{N} \gamma_l(t)\left(\tilde{s}(t) - \gamma_l(t)\right) = 0 \tag{3.31}$$

which states the orthogonality of the IMFs. As stopping criterion they determine the following index and the sifting procedure will stop once that it reaches a certain-value:

$$OC = \left|\sum_{t=1}^{N} \frac{m_l(t)\,\tilde{s}(t)}{m_l(t)\left(\tilde{s}(t) - m_l(t)\right)}\right| \tag{3.32}$$

where $\tilde{s}(t)$ is the original signal and $m_l(t)$ is the mean envelop. As for the previous methods, the main problem is that the value to stop the sifting procedure is pre-defined and is highly related to the signal features. A general threshold for this criterion is stopping the sifting procedure when OC > 1.05. Moreover, non-stationary and non-linear data do not produce orthogonal IMFs.

3.6.5 A Simple Example

These stopping criteria are the most used within the literature. They shared the problem of being heuristic without a mathematical formulation that determines a general rule for every kind of signal. The last two methods are based on the theoretical concept that IMFs are mutually orthogonal. It has been proved that IMFs of non-stationary and non-linear signals are not orthogonal, leading to energy leakage. On relevant alternative is proposed in Huang et al. (2008) who determined the orthogonal empirical mode decomposition or OEMD based on the Gram-Schmidt orthogonalization. However, they demonstrate that even this decomposition presents some issues, i.e. mixing higher frequency components with lower frequency one; as a result, some instantaneous frequencies turn to be negative and so lacking any physical meaning. Moreover, the residue seems to be not orthogonal to any other components, and therefore, there is always some energy leakage.

Several stopping criteria have been introduced within the literature of Empirical Mode Decomposition. The problem of the identification of the right number of IMFs without any scale mixing problems or the convergence of the algorithm are still issues that have to be solved. Within this section a toy example that supports the above evidence is being included.

In order to demonstrate what has been stated above, the following signal has been taken into account and then interpolated with a natural cubic spline:

$$s(t) = \sin(\pi t) + \sin(6\pi t) + \sin(8\pi t) + 0.5t, \quad \text{for} \quad t \in [0, 2.6] \tag{3.33}$$

Within this section, four main stopping criteria have been discussed: the Cauchy-Type Convergence (SD) method, the Mean Fluctuations Threshold method, the Energy Difference Tracking method and finally the Orthogonality Criterion. Each of them has been implemented on R and applied to the above signal. The results are aimed to show that every criterion provides a different number of IMFs based on heuristic rules strictly depending on the studied signal. Therefore, every method should be carefully used over different kind of signals since the obtained IMFs may be not a meaningful representation of original one. Figure 3.10 presents the identified IMFs by using these four criteria.

The top panel shows the interpolated signal $\tilde{s}(t)$. The four subpanels show instead the decompositions obtained through the different stopping criteria. The top-left panel is the result provided by the Cauchy-Type Convergence criterion; it utilises a threshold equal to 0.3 of the standard deviation of two consecutive sifting steps. In this specific case, it extracts five different IMFs and a residual component. The top-right panel shows the result of the EMD exploiting the criterion exploiting the mean fluctuations threshold. The idea behind it is indeed providing small fluctuations of the mean and emphasising locally large variations. It identifies five IMFs and a residue. The bottom-left panel presents the result of the Energy Difference Tracking criterion. Note that, when applied, it was done in combination with the Mean Fluctuations Threshold one. It identifies two IMFs only. This reflects the problem of heuristic rules that have been

previously discussed. Lastly, the bottom-right panel shows the orthogonality criterion applied to the considered signal. As for the former case, it has been used in combination with the Mean Fluctuation Threshold criterion. It extracts two IMFs since another threshold, the index of orthogonality, has been taken into account within the algorithm.

By observing this simple example, it is possible to notice that the first two stopping criteria extract more IMFs than the third and the fourth ones. However, in the latter ones, the first IMF seems to capture most of the frequency content compared to the second and the third IMFs identified. Furthermore, in the decompositions relying on the SD and the MFD stopping criteria, the IMFs show a more regular profile without riding waves or mixing frequencies between the basis functions. The studied signal includes three different sinusoids carrying three distinct frequencies and a trend component. The EMD using the first two stopping criteria identifies 5 IMFs and a residual. Hence, two extra components are identified. While the EMD applied with the last two stopping criteria (the energy and the OC ones) extracts two IMFs and a residual only. Hence, the first two are more sensitive to riding waves and over-extract extra components; however, the last two tend to concentrate most of the frequency content within the highest basis function causing the phenomenon known as mode-mixing since it also detects more than just one frequency.

Given this evidence, the SD and the MFD stopping criteria are the ones that provide more stable results and, therefore, are the ones adopted for the decomposition obtained in the experiments in part III. Within these experiments, more challenging tasks are required, and the MFD stopping criterion will be selected. The fact that multiple thresholds are used instead of only one of the SD stopping criterion will provide more stable results.

3.7 Spline Interpolation and Alternative Envelope Algorithms

The parametric representation interpolating the maxima and minima envelopes plays a central role within the sifting procedure. Many solutions have been proposed in the literature. Some of them are presented within this section to provide a general overview. They represent alternative solutions to the classical EMD algorithm and tackle the main issues affecting the sifting above introduced, i.e. end effects, hidden scale, mode mixing and boundary conditions.

The experiments conducted in this thesis in part III will rely on the natural cubic spline since its combination with the "wave" boundary condition above described offer the most stable solution in terms of the sifting procedure.

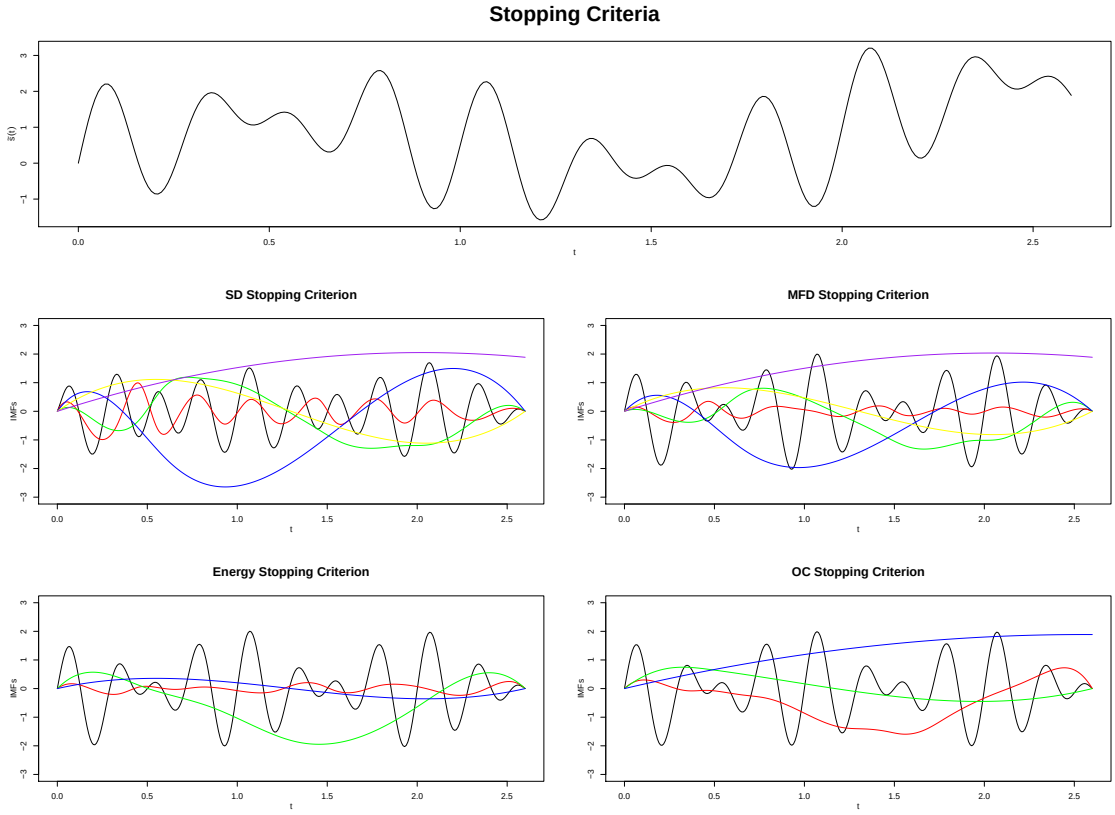


Figure 3.10: Stopping Criteria. The top panel shows the original signal $\tilde{s}(t)$ derived by interpolation using a natural cubic spline of the discrete samples of $s(t) = \sin(\pi t) + \sin(6\pi t) + \sin(8\pi t) + 0.5t$ for $t \in [0, 2.6]$. The other four sub-figures present the decompositions obtained through the different stopping criteria. It is possible to observe that different number of IMFs are found as well as they present different shapes.

In the following subsections, firstly, the concept of cubic spline is introduced with the primary motivation to use such a specific class of fitting curves. This corresponds to the most common approach in constructing the EMD envelopes. Afterwards, the alternative methods used in the literature to solve such a task are presented. Particularly, they are the B-spline combined with an alternative technique exploiting the binomial operator; the natural and the clamped cubic splines; the Akima spline, which is the basis for a segment power function method introduced in the EMD literature.

3.7.1 Basis Functions: Cubic splines

The first step of the EMD foresees fitting a spline trough the discrete path-realization $s(t)$. After that, the same spline will be used for the envelope construction. One solution is offered by smoothing splines. Consider the discrete path-realization $s(t)$. The following linear relationship is assumed:

$$s(t_i) = g(t_i) + \epsilon_i \quad \text{for} \quad i = 1, \dots, N \tag{3.34}$$

where g is a smooth function equal to the conditional mean of s_i given t_i and ϵ_i are independent, mean zero errors with constant variance σ^2. Instead of relying on the classical Least Squares estimator, O' Sullivan (see O'Sullivan (1986)) proposed the notion of smooth function by introducing a roughness penalty over the usual sum of squares, i.e. penalized sum of squares, as follows:

$$\sum_{i=1}^{n}(s_i - \sum_{j=1}^{m+k+1} \beta_j \hat{g}\left(t_i\right))^2 + \lambda \int \sum_{j=1}^{m+k+1} \beta_j(\hat{g}^{''}\left(t_i\right))dt \tag{3.35}$$

Integrating the squared second order derivative represents the penalty and is controlled by the tuning parameter λ. The main issue featuring splines is the identification of the optimal number of knots along with their locations. Few knots may generate underfitting, while many of them could cause overfitting. Here, knots are placed at data points and overfitting is highly monitored. There is only one solution to the minimization problem 3.35 given by:

$$\underset{\hat{g}}{\operatorname{argmin}}\, L(\hat{g}, \lambda) \tag{3.36}$$

which is a function of $t_0 < t_1 < \cdots < t_N$ called *smoothing spline function*. This function will be chosen as sensing basis function within the EMD section. A specific class of smoothing spline, named *cubic spline*, will be selected. The reasoning behind such choice can be justified through the following two statements.

Definition 3.7.1 (Smoothness)**.** *The smoothness measure of a function f on interval $[a, b]$ is given by its integrated second derivative over that interval:*

$$\int_a^b (f''(t))^2 dx. \tag{3.37}$$

It is possible to show that the following theorem applies, under such a measure of smoothness.

Theorm 3.7.2 (Optimality of Natural Cubic Spline)**.** *Given interpolation data $\{(t_i, s_i)\}_{i=1}^n$, then among all functions $f \in \mathbb{C}^2[a, b]$ which interoplate (go exactly through the observed points), the natural cubic spline is the smoothest as quantified through measure in Equation 3.37.*

Such theorem justifies the employment of natural cubic splines approximating our IMF basis function $\tilde{s}(t)$. It will also be exploited to approximate the starting signal denoted by $s(t)$. These statements are shown in section 3.1. This is highly central from an EMD perspective: cubic splines are very popular given their smoothness. Within the EMD section, each basis function is defined as a cubic spline and denoted as $\tilde{s}(t)$. In the following subsections, the different alternatives that have been studied in thel literature with respect to the EMD are introduced. Note that an algorithm to compute each spline is provided in Appendix B.

3.7.2 B-Splines and the Binomial Operator

One solution often adopted when interpolation is the task of interest is represented by B-splines. This class of parametric interpolators has been introduced by de Boor (2001) and is below presented. The critical reason for adopting such a spline is its efficiency in terms of computational cost. Furthermore, their recursive formulation allows for great flexibility.

In an EMD context, B-splines have been used by Chen et al. (2006), who proposed an alternative EMD by deriving recursive formulation of the Hilbert transform. The reader should refer to such work for further open mathematical discussion related to the EMD. A general framework for B-splines is now introduced.

Let $\tau := (\tau_i)$ be a non-decreasing sequence of m scalars. The $i-th$ B-spline of order k for the knots sequence τ is denoted by $B_{i,k,\tau}$ and defined as:

$$B_{i,k,\tau}(t) := (\tau_{i+k} - \tau_i)\mathcal{D}[\tau_{i+k} - \tau_i](. - t)_+^{k-1}, t \in \mathbb{R} \tag{3.38}$$

where the operator $\mathcal{D}\left[\tau_1, \tau_2, \ldots, \tau_n\right]\{f\}$ applied to any function f represents the k-th order divided difference of function f at values $\tau_1, \tau_2, \ldots, \tau_n$. The argument $\left\{(. - t)_+^{k-1}\right\}$ is zero if $\tau < t$ and equals $(\tau - t)^{k-1}$ if $\tau \geq t$. They form a basis for the space of splines of order k with knots $\tau_i, i \in \mathbb{Z}$. According to the recurrence relation (see de Boor (2001)), after $k-1$ applications of such property, $B_{i,k,\tau}$ assumes the following form:

$$B_{i,k,\tau} = \sum_{\tau=i}^{i+k-1} \alpha_{\tau,k} B_{\tau,1} \tag{3.39}$$

This property is defined as the DeBoor-Cox recursion formula and offers more flexibility compared to other configuration of splines.

A precise mathematical representation for the envelopes still lacks within the literature. The definition of such element could be a keystone in the sifting procedure. By exploiting the B-spline approach introduced, Chen et al. (2006) construct a function, named the compensating function, which replaces the mean value. Consider the B-splines definition. Within a classical EMD context, Chen et al. (2006) defines the set of knots $t := \{t_j : j \in \mathbb{Z}\}$ as the extreme points of $\tilde{s}(t)$. Inside the support of $B_{j,k,t}$, the next linear functional is defined:

$$\lambda_{j,k,t} : \tilde{s} \longmapsto \frac{1}{2^{k-2}} \sum_{l=1}^{k-1} \binom{k-1}{l} \tilde{s}(t_{j+l}) \tag{3.40}$$

which is a binomial average of the extrema within the support of $B_{j,k,t}$. This representation is needed in order to define the operator that replaces the mean envelope of the classic EMD as:

$$V_{t^h,k} x := \sum_{j \in \mathbb{Z}} \lambda_{j,k,t}(x) B_{j,k,t} \tag{3.41}$$

The good reason in employing such approach is given by a natural convergence of the sifting algorithm; it results from the variation diminishing property of

B-spline series. However, one of its drawbacks that still have to be investigated is the uniqueness of the provided IMFs.

3.7.3 Akima Splines and The Segment Power Function

Akima splines are first-order smooth splines introduced by Salomon (2011). The idea is computing the slope of a point (t_i, s_i) according to its two predecessors $((t_{i-1}, s_{i-1}), (t_{i-2}, s_{i-2}))$ and its two successors $((t_{i+1}, s_{i+1}), (t_{i+2}, s_{i+2}))$ as follows:

$$\tilde{s}'(t_i) = \frac{|m_{i+2} - m_{i+1}|(m_{i-1}) + |m_{i-1} - m_{i-2}|(m_{i+1})}{|m_{i+2} - m_{i+1}| + |m_{i-1} - m_{i-2}|} \tag{3.42}$$

where $m_{i-2} = \frac{s_{i-1}-s_{i-2}}{t_{i-1}-t_{i-2}}$. For each splines, endpoints are estimated through the employment of two quadratic polynomials and by assuming that $t_4 - t_2 = t_3 - t_1 - t_2 - t_0$ and $t_N - t_{N-2} = t_{N-1} - t_{N-3} = t_{N-2} - t_{N-4}$. It is then possible to compute $m_0 = 2m_1 - m_2$, $m_1 = 2m_2 - m_3$, $m_{N-1} = 2m_{N-2} - m_{N-2}$ and $m_N = 2m_{N-1} - m_{N-2}$. By considering the last equations, the whole dataset is then covered.

If cubic spline provides a too smooth interpolation, the Akima one offers a too flexible approximation, meaning that its interpolations is not smooth enough. Qin and Zhong (2006) proposed the segment power function algorithm to address this issue. The idea involves the use of a power function method to interpolate adjacent points P_{i-1}, P_i, P_{i+1} and P_i, P_{i+1}, P_{i+2} and then splice the two curves. For instance, consider the generic interpolation points denoted P_1, P_2, P_3 and P_2, P_3, P_4. The functional curve for this splicing takes the form:

$$\tilde{s}(t) = \frac{t_3 - t}{t_3 - t_2}\tilde{s}_2(t) + \frac{t - t_2}{t_3 - t_2}\tilde{s}_3(t) \tag{3.43}$$

where $\tilde{s}_2(t), \tilde{s}_3(t)$ are single valued smooth curves (first order continuous and differentiable) interpolated over points P_1, P_2, P_3 and P_2, P_3, P_4 respectively. They considered the use of a power functional form, given for the interpolation of three generic points in the (t, s)-plane denoted by $P_1(t_1, x_1), P_2(t_2, x_2), P_3(t_3, x_3)$ according to:

$$\tilde{s}_2(t) = \begin{cases} \left(\frac{t-t_2}{t_1-t_2}\right)^\beta \left[\frac{(t_3-t_2)s_1-(t_2-t_1)s_3}{t_3-t_1}\right] + \frac{s_3-s_1}{t_3-t_1}(t-t_2) + x_2, & t \leq t_2, \\ \left(\frac{t-t_2}{t_3-t_2}\right)^\beta \left[\frac{(t_3-t_2)x_1-(t_2-t_1)s_3}{t_3-t_1}\right] + \frac{x_3-s_1}{t_3-t_1}(t-t_2) + s_2, & t \geq t_2, \end{cases} \tag{3.44}$$

The value of $\beta \in \mathbb{R}$ should be considered carefully. These authors recommended a value of around $\beta = 2.5$ by declaring it robust for their applications.

Part II

Machine Learning Techniques and Extensions

Chapter 4

Characterisation of Time-Frequency Domain

One of the purposes of this thesis is solving classifications tasks testing different EMD based features. A statistical approach often considered to develop such a framework requires the definition of a decision function, often referred to as a classifier. The problem can then be formulated as a learning procedure aimed to identify the optimal classifier. Two classification methods will be used in later Chapters (i.e. Chapters 5, 6 respectively) utilising the Support Vector Machine and Gaussian Processes within a decision-theoretic framework of Generalised Likelihood Ratio testing. Details and extensions related to these statistical techniques will be then presented.

A common issue encountered when analysing real-world data-sets is the nonlinear and nonstationary properties of the of the data. As a result, traditional linear methods cannot be applied, and nonlinear procedures are required instead. A technique that has become highly popular within both machine learning and statistical signal processing to overcome such a challenge is represented by kernel methods. The idea is to map the existing data-set in the input space, into a new space known as the features space where linear algorithms apply again. The mapping is usually unknown explicitly, and the so-called kernel trick comes into play. In such a way, a family of kernel functions can then be employed to synthesise information regarding the unknown mapping. These methods rely on the notion that a nonlinear data transformation into a higher dimensional feature space increases the probability of the linear separability of the transformed data. This is accomplished by exploiting properties of dot products in the high or infinite-dimensional feature space in terms of kernel functions of the input space. Therefore, formally introducing such spaces and their attributes is imperative.

In this work, kernel methods are required to characterise EMD based features used to carry the classification tasks of interest. Such features result from the spectral decomposition of observed path realisations of stochastic processes or deterministic functions. The selected space for the tasks mentioned above is the space of continuous functions in $\mathbb{R}$ with d derivatives denoted as $C^d(\mathbb{R})$,

where d is some integer. This class of functions will admit classical calculus expansions such as the Taylor series or Maclaurin series. The ability to derive monomial basis representations comes from the restriction on the regularity of the underlying function, which could be, for example, C^{∞}. In this case, the expansion coefficients can be obtained if the function is differentiable in an infinite number of times at some points. If these assumptions do not hold and attributes such as non-stationarity and non-linearity come into play, ad hoc spaces are required.

The following steps of this Chapter are firstly to review kernel learning procedures. In this respect, several aspects are considered as the feature vectors employed to summarise the data, the kernel proximity and the kernel choice. After, a review of the kernel families employed in part III for different experiments relying on the Support Vector Machine framework introduced in Chapter 5 is provided. In this regard, multi-kernel learning techniques will also be taken into account. In Chapter 6, other kernel representations within a Gaussian Process setting will be considered. Therefore, a section describing such kernels is also provided.

4.1 Kernel Learning

The assumption made in the presented settings of this Chapter consists of considering data which are vectors, i.e. $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^D$. Denote $\boldsymbol{X} \in \mathbb{R}^{N \times D}$ the matrix whose i-th row is $\boldsymbol{x}_i$. Consider the input data $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^D$ and the associated matrix $\boldsymbol{X} \in \mathbb{R}^{N \times D}$ such that

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1 \\ \vdots \\ \boldsymbol{x}_N \end{bmatrix}_{N \times D} = \begin{bmatrix} x_{1,1} & \cdots & x_{1,D} \\ \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,D} \end{bmatrix}_{N \times D} \tag{4.1}$$

What happens when kernel methods come into play is that a set of "features" can be chosen and define a space $\mathcal{H}$. Such features are "performed" (implicitly or explicitly) in the hope that relevant structure will be revealed by the mapping of the data to a much higher dimensional space. The data $\boldsymbol{X}$ are therefore mapped to the feature space $\mathcal{H}$ using a mapping

$$\varphi : \mathbb{R}^D \to \mathcal{H} \tag{4.2}$$

and then the task of interest, i.e. classification or regression or clustering, is performed in $\mathcal{H}$ using ad hoc methods belonging to supervised or unsupervised learning, for example. Note that the feature space is directly denoted as $\mathcal{H}$, hence a dot product space. Consider the feature space $\mathcal{H} \subset \mathbb{R}^P$, with $P >> D$. For each data point $\boldsymbol{x}_i$, with $i = 1, \ldots, N$ the following is applied

$$\varphi(\boldsymbol{x}_i) = [\varphi_1(\boldsymbol{x}_i), \ldots, \varphi_P(\boldsymbol{x}_i)]_{1 \times P} \tag{4.3}$$

and, therefore, $\varphi(\boldsymbol{x}_i) \in \mathcal{H} \subset \mathbb{R}^P$ represents the i-th input vector projected into the feature space of higher dimension P. In matrix form this is given as

$$\boldsymbol{\Phi} = \varphi(\boldsymbol{X}) = \begin{bmatrix} \varphi(\boldsymbol{x}_1) \\ \vdots \\ \varphi(\boldsymbol{x}_N) \end{bmatrix}_{N \times P} = \begin{bmatrix} \varphi_1(\boldsymbol{x}_1) & \cdots & \varphi_P(\boldsymbol{x}_1) \\ \vdots & \ddots & \vdots \\ \varphi_1(\boldsymbol{x}_N) & \cdots & \varphi_P(\boldsymbol{x}_N) \end{bmatrix}_{N \times P} \tag{4.4}$$

Consider now the covariance matrix of $\boldsymbol{\Phi}$ given as

$$\boldsymbol{C}_{P \times P} = \boldsymbol{\Phi}^\intercal \boldsymbol{\Phi} = \begin{bmatrix} \varphi_1(\boldsymbol{x}_1) & \cdots & \varphi_1(\boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ \varphi_p(\boldsymbol{x}_1) & \cdots & \varphi_P(\boldsymbol{x}_N) \end{bmatrix}_{P \times N} \begin{bmatrix} \varphi_1(\boldsymbol{x}_1) & \cdots & \varphi_P(\boldsymbol{x}_1) \\ \vdots & \ddots & \vdots \\ \varphi_1(\boldsymbol{x}_N) & \cdots & \varphi_P(\boldsymbol{x}_N) \end{bmatrix}_{N \times P} \tag{4.5}$$

whose element $C_{n,m}$ is given as

$$C_{n,m} = [\varphi_n(\boldsymbol{x}_1), \ldots, \varphi_n(\boldsymbol{x}_N)] \begin{bmatrix} \varphi_m(\boldsymbol{x}_1) \\ \vdots \\ \varphi_m(\boldsymbol{x}_N) \end{bmatrix} = \sum_{i=1}^{N} \varphi_n(\boldsymbol{x}_i)\varphi_m(\boldsymbol{x}_i) = \mathbb{C}\text{ov}(\varphi_n(\boldsymbol{X}), \varphi_m(\boldsymbol{X})) \tag{4.6}$$

and which represents the covariance between the n-th feature function φ_n and the m-th feature function φ_m in the feature space $\mathcal{H} \subset \mathbb{R}^P$ with $n, m = 1, \ldots, P$ across the given input data samples.

Having chosen a dot product space for the feature space, i.e. $\mathcal{H} \subset \mathbb{R}^P$, associated with it is a kernel along with a kernel matrix, also known as Gram Matrix, $\boldsymbol{K} \in \mathbb{R}^{N \times N}$ according to elements:

$$\begin{aligned} K_{i,j} &= k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \varphi(\boldsymbol{x}_i), \varphi(\boldsymbol{x}_j) \rangle \\ &= \varphi(\boldsymbol{x}_i)\varphi(\boldsymbol{x}_j)^\intercal = [\varphi_1(\boldsymbol{x}_i), \ldots, \varphi_P(\boldsymbol{x}_i)] \begin{bmatrix} \varphi_1(\boldsymbol{x}_j) \\ \vdots \\ \varphi_P(\boldsymbol{x}_j) \end{bmatrix} = \sum_{p=1}^{P} \varphi_p(\boldsymbol{x}_i)\varphi_p(\boldsymbol{x}_j) \end{aligned} \tag{4.7}$$

for $i, j = 1, \ldots, N$. This corresponds to an inner product between the samples projected in the feature space since it is possible to write $\varphi(\boldsymbol{x}_i) = \varphi_i \in \mathcal{H} \subset \mathbb{R}^P$ which is a P dimensional vector in the feature space. In matrix form, this is given as

$$\boldsymbol{K}_{N \times N} = \varphi(\boldsymbol{X})\varphi(\boldsymbol{X})^\intercal = \begin{bmatrix} \varphi_1(\boldsymbol{x}_1) & \cdots \varphi_P(\boldsymbol{x}_1) & \\ \vdots & \ddots & \vdots \\ \varphi_1(\boldsymbol{x}_N) & \cdots \varphi_p(\boldsymbol{x}_N) & \end{bmatrix}_{N \times P} \begin{bmatrix} \varphi_1(\boldsymbol{x}_1) & \cdots \varphi_1(\boldsymbol{x}_N) & \\ \vdots & \ddots & \vdots \\ \varphi_P(\boldsymbol{x}_1) & \cdots \varphi_P(\boldsymbol{x}_N) & \end{bmatrix}_{P \times N} \tag{4.8}$$

Then any algorithm whose operations can be expressed in the input space in terms of dot products can be generalised to an algorithm which operates in the feature space by substituting a kernel function for the inner product.

In practice, the study of the Gram matrix $\boldsymbol{K}$ is in place of the input covariance matrix $\boldsymbol{X}^T\boldsymbol{X}$. Therefore, the choice of kernel for whichever selected task can be

highly influential on the outcome achieved. There are three main considerations that might affect the solution:

- The choice of the functional kernel (Mercer kernel, see examples in Schölkopf et al. (2002), Zhang et al. (2007), and Nguyen and Ho (2007).
- The choice of the feature vector used to summarise the data, for a detailed review see Guyon and Elisseeff (2006).
- The hyperparameters settings of the kernel utilized, hence the procedure to estimate and select them.

The aim of this section, and the following subsections, is to deal with these three aspects by first presenting the main components introduced in the prior section. Subsequently, the solutions adopted in this work will be shown. The procedure of interest at this stage is often referred to as Kernel Learning and foresees the learning of the kernel function structure, which is unknown a priori. The selected choice for the class of functional kernels is presented.

The following theorem introduces a class of kernel called the Mercer kernels

Theorm 4.1.1 (Characterization of Kernels)**.** *A function $k : X \times X \to \mathbb{R}$ which is either continuous or has a finite domain, can be decomposed as*

$$k(\boldsymbol{x}, \boldsymbol{x}') = \langle \varphi(\boldsymbol{x}), \varphi(\boldsymbol{x}') \rangle_{\mathcal{H}} \tag{4.9}$$

into a feature map φ into a Hilbert space $\mathcal{H}$ applied to both its arguments, followed by an inner product in $\mathcal{H}$ if and only if (iff) it is finitely positive semi-definite. A Mercer kernel.

For a detailed proof see Shawe-Taylor et al. (2004). Mercer kernels are often employed since satisfying the following rules. Consider a space X of sample and two Mercer kernels $k_1(\cdot, \cdot)$ and $k_2(\cdot, \cdot)$ over $X \times X$. Then $k(\cdot, \cdot)$ is also a Mercer kernel under the following constructions:

- Addition: $k(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}, \boldsymbol{x}') + k_2(\boldsymbol{x}, \boldsymbol{x}')$
- Positive scaling: $k(\boldsymbol{x}, \boldsymbol{x}') = \alpha k_1(\boldsymbol{x}, \boldsymbol{x}')$ for $\alpha > 0$
- Multiplication: $k(\boldsymbol{x}, \boldsymbol{x}') = k_1(\boldsymbol{x}, \boldsymbol{x}') k_2(\boldsymbol{x}, \boldsymbol{x}')$
- Dot product (inner product): $k(\boldsymbol{x}, \boldsymbol{x}') = \varphi(\boldsymbol{x}) \cdot \varphi(\boldsymbol{x}')$ for any function $\varphi(\boldsymbol{x})$ on $\boldsymbol{x} \in X$.
- Standardization: $k(\boldsymbol{x}, \boldsymbol{x}') = \frac{k_1(\boldsymbol{x}, \boldsymbol{x}')}{\sqrt{k_1(\boldsymbol{x}, \boldsymbol{x}') k_1(\boldsymbol{x}, \boldsymbol{x}')}}$

It is then advantageous to adopt this class of kernels given these properties and the presented framework of this Chapter.

Formally, once the kernel is selected, the Gram Matrix can be defined as follows.

Definition 4.1.2 (Gram Matrix or Kernel Matrix)**.** *The Gram Matrix is a positive semi-definite matrix constructed for a given kernel function* $k(\boldsymbol{x}, \boldsymbol{x}')$ *for data* $\boldsymbol{x}_i \in X$ *with* $i \in \{1, \ldots, N\}$ *given by*

$$\begin{aligned} \boldsymbol{K} &= \begin{bmatrix} k_{11} & k_{12} & k_{13} & \cdots & k_{1N} \\ k_{21} & k_{22} & k_{23} & \cdots & k_{2N} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ k_{N1} & k_{N2} & k_{N3} & \cdots & k_{NN} \end{bmatrix} \\ &= \begin{bmatrix} k(\boldsymbol{x}_1, \boldsymbol{x}_1) & k(\boldsymbol{x}_1, \boldsymbol{x}_2) & k(\boldsymbol{x}_1, \boldsymbol{x}_3) & \cdots & k(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ k(\boldsymbol{x}_2, \boldsymbol{x}_1) & k(\boldsymbol{x}_2, \boldsymbol{x}_2) & k(\boldsymbol{x}_2, \boldsymbol{x}_3) & \cdots & k(\boldsymbol{x}_2, \boldsymbol{x}_N) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ k(\boldsymbol{x}_N, \boldsymbol{x}_1) & k(\boldsymbol{x}_N, \boldsymbol{x}_2) & k(\boldsymbol{x}_N, \boldsymbol{x}_3) & \cdots & k(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{bmatrix} \end{aligned} \tag{4.10}$$

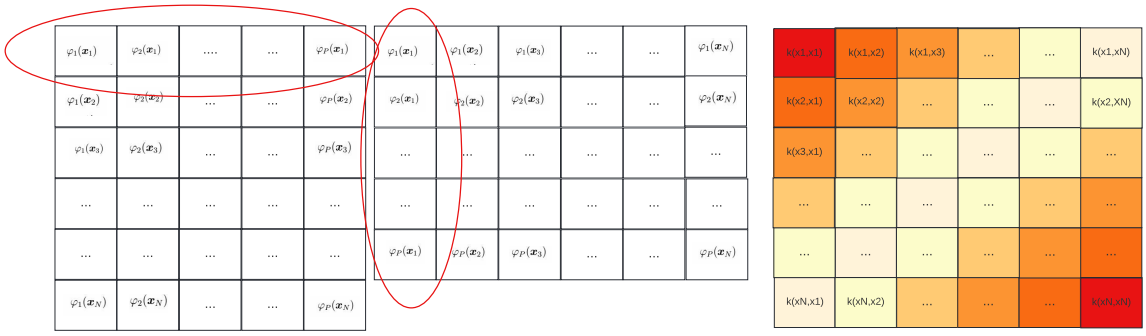


Figure 4.1: The two matrices $\varphi(\boldsymbol{x})$ and $\varphi(\boldsymbol{x})^\intercal$ are shown in white. On the right, the Gram Matrix resulting from the inner product is presented. Note that the Gram Matrix colour provides symmetry, and on each cell, the resulting entry is printed. Furthermore, for the first cell, $k(\boldsymbol{x}_1, \boldsymbol{x}_1)$, the two vectors used to obtained such result are highlighted.

The structure of the Gram matrix will determine the association between the pair of data points and model hidden information required for the classification tasks presented in later Chapters.

The rest of this section deals with the remaining two aspects affecting the kernel choice that comes into play in this work. Firstly, the feature vectors used to summarise the data in part III are introduced. Keeping in mind that the central goal is to describe EMD based features behaviour and, therefore, the kernel choice would offer a way to characterise their structural time-varying properties. The last aspect corresponding to the methods chosen for the hyperparameters setting and learning algorithms are described.

4.1.1 Feature Vector to Summarise the Data: EMD Features

In this subsection, the EMD based features employed to represent the original data signals are presented. The EMD extracts basis functions carrying ordered spectral frequency content information of the interpolated signal $\tilde{s}(\boldsymbol{t})$, as introduce in equation 3.1 of Chapter 3. The classification task considering the IMFs or features engineered on them, aims to characterise such information through the use of kernels detecting structural changes related to differences in their spectral content. In the time domain this is given as the number of oscillations, whilst in the frequency domain this will be captured by the calculated instantaneous frequencies obtained with the Hilbert transform. Such ordering should also be reflected in the other extracted features based on the IMFs and would affect the performances of the classification task. High-frequency features tend to capture most of the spectral content of the original signal and, therefore, are expected to provide better performances. Furthermore, these are also carrying most of the non-stationary traits of $\tilde{s}(\boldsymbol{t})$, which, if efficiently identified and properly handled through an ad hoc kernel structure, would provide a great deal of discrimination power.

At this stage, it is essential to highlight that within this thesis a method of partitioning the time-frequency plane with a posteriori basis functions is utilized, since the location of the spectral information cannot be known a priori. Once this step is achieved, the main focus is construction of a classification framework relying on kernel learning procedures that describe the regions of the partitioned time-frequency plane through different hyperparameter structural sets. In such a way, a more effective discrimination power can be produced. To achieve this, it is critical to formally define the EMD based features that will be used to characterise such frequency content information within different feature space domains, since this can highly affect the kernel choice selection process.

The following table provides a summary of the multiple representations considered for each IMF basis function. The time mesh defined to summarise the features is denoted by $t_i^{'}$ such that $t_i^{'} \in \{0 = t_1^{'}, \ldots, t_N^{'} = N\}$. Part III presents several experiments, synthetic and speech applied, where the EMD is performed either over the interpolated synthetic samples or the interpolated voice samples, and five IMFs will are stored: the first three with the highest frequency; the lowest; and the residual. Afterwards, both instantaneous frequencies and coefficients of the cubic spline of each IMF are calculated. Classical statistics are also extracted by a sliding window of fixed length over an IMF, such that, $W\left[\tau_1, \tau_{j+1}\right] = W\left[\tau_{j+1}, \tau_{j+2}\right] = \cdots = W\left[\tau_{j+N-1}, \tau_{j+N}\right]$, where $\tau_j \in \{0 = \tau_1, \ldots, \tau_V = N\}$. The considered classical statistics are, in order from top to bottom, are: mean, variance, minimum, maximum, kurtosis, skewness and root mean square (RMS). Note that the residual $r(t_i^{'})$ is included in the decomposition and denoted as $\gamma_{k+1}(t_i^{'})$.

EMD Feature	Label	Window
IMFs	$\gamma_1(t_i^{'})$, $\gamma_2(t_i^{'})$, $\gamma_3(t_i^{'})$, $\gamma_k(t_i^{'})$, $\gamma_{k+1}(t_i^{'})$	NA
Instantaneous Frequencies	$f_1(t_i^{'})$, $f_2(t_i^{'})$, $f_3(t_i^{'})$, $f_k(t_i^{'})$, $f_{k+1}(t_i^{'})$	NA
Cubic Spline Coefficients	$\mathbf{b^1}(t_i^{'})$, $\mathbf{b^2}(t_i^{'})$, $\mathbf{b^3}(t_i^{'})$, $\mathbf{b^k}(t_i^{'})$, $\mathbf{b^{k+1}}(t_i^{'})$	NA
	$\mathbf{c^1}(t_i^{'})$, $\mathbf{c^2}(t_i^{'})$, $\mathbf{c^3}(t_i^{'})$, $\mathbf{c^k}(t_i^{'})$, $\mathbf{c^{k+1}}(t_i^{'})$	NA
	$\mathbf{d^1}(t_i^{'})$, $\mathbf{d^2}(t_i^{'})$, $\mathbf{d^3}(t_i^{'})$, $\mathbf{d^k}(t_i^{'})$, $\mathbf{d^{k+1}}(t_i^{'})$	NA
Classical Statistics	$\hat{\mu}_1$, $\hat{\mu}_2$, $\hat{\mu}_3$, $\hat{\mu}_k$, $\hat{\mu}_{k+1}$	$W[\tau_j, \tau_{j+1}]$
	$\hat{\sigma}_1^2$, $\hat{\sigma}_2^2$, $\hat{\sigma}_3^2$, $\hat{\sigma}_k^2$, $\hat{\sigma}_{k+1}^2$	$W[\tau_j, \tau_{j+1}]$
	$\tilde{c}_1$, $\tilde{c}_2$, $\tilde{c}_3$, $\tilde{c}_k$, $\tilde{c}_{k+1}$	
	c_1^*, c_2^*, c_3^*, c_k^*, c_{k+1}^*	$W[\tau_j, \tau_{j+1}]$
	$\hat{\beta}_{2_1}$, $\hat{\beta}_{2_2}$, $\hat{\beta}_{2_3}$, $\hat{\beta}_{2_k}$, $\hat{\beta}_{2_{k+1}}$	$W[\tau_j, \tau_{j+1}]$
	$\hat{k}_1$, $\hat{k}_2$, $\hat{k}_3$, $\hat{k}_k$, $\hat{k}_{k+1}$	$W[\tau_j, \tau_{j+1}]$
	RMS_1, RMS_2, RMS_3, RMS_k, RMS_{k+1}	$W[\tau_j, \tau_{j+1}]$

Table 4.1: Table describing the extracted EMD based features used within part III for the synthetic and the speech experiments. The IMFs are firstly extracted and then the IFs, the Spline Coefficients and the Classical Statistics were extracted for each of the considered basis functions (i.e. the first five IMFs). Note that $\tilde{c}_i = \min[\tau_i, \tau_{i+1})$, $c_i^* = \max[\tau_i, \tau_{i+1})$

This set of features is used throughout the synthetic and speech experiments. One of this thesis's main goals is to explore their discrimination power in different classification tasks. Results, along with details of their use, will be given in part III, Chapters 8, 9, respectively.

The third aspect influencing the kernel choice learning process corresponds to the method employed for the hyperparameter selection. The techniques considered in this thesis are the Kernel Target Alignment and cross-validation methods for the classifier Support Vector Machine. In the following subsection, an overview of these methods is presented.

4.2 Families of Kernel for Support Vector Machine

In this subsection, the kernel function families considered in the experiments related to the framework developed in Chapter 5 are presented.

Six kernels have been explored: the radial basis; the Laplace radial basis; the polynomial; the sigmoid; the Bessel and the linear functions. Each of them has specific parameters that need to be optimized to obtain accurate performances of the SVM algorithm.

Kernel	**Formula**	**Optimisation Parameter**
RBF	$k(x_i, x_j) = \exp(-\gamma\|x_i - x_j\|^2)$	γ
Laplace RBF	$k(x_i, x_j) = \exp(-\gamma\|x_i - x_j\|)$	γ
Polynomial	$k(x_i, x_j) = (\gamma\langle x_i, x_j\rangle + r)^d$	γ, r and d
Sigmoid	$k(x_i, x_j) = \tanh(\gamma\langle x_i, x_j\rangle + r)$	γ and r
Bessel	$k(x_i, x_j) = \frac{\text{Bessel}^d_{\nu+1}(\gamma\|x_i - x_j\|)}{(\|x_i - x_j\|)^{(-d(\nu+1))}}$	γ, ν and d
Linear	$k(x_i, x_j) = \langle x_i, x_j\rangle$	–

Table 4.2: Kernel functions employed in Chapter 5. Note: γ gamma or scale; r offset; d degree and ν order. The optimisation of this kernel functions is conducted within a Support Vector Machine framework, hence, there will also be a C cost optimisation parameter which is introduced in Chapter 5.

SVMs are strictly dependent on the selected hyperparameters of the kernel functions. Optimal selections can be made for performance measurements evaluated through a cross-validation score of the training set. Several methods are available for the search of optimal hyperparameters. The selected grid-search method is the most numerically stable and easy to implement. In the SVM settings, the hyperparameters regions are set as follows: $C \in \{2^{-2}, 2^{-1}, \ldots, 2^6\}$; $r \in \{2^{-5}, 2^{-4}, \ldots, 2^{-2}\}$; $d \in \{1, 2, 3\}$; and $\nu \in \{1, 2\}$. Note that C corresponds to the cost parameter introduced in Chapter 5. Regarding the grid for γ, the kernlab package approach for R, which uses the sigest function to obtain the grid range for this parameter, was adopted. The selected values for γ corresponds to a trimmed mean of its grid. The experiments related to this framework are implemented with 2-fold cross-validation of the training set to tune the hyperparameters for classification.

Figure 4.2 presents a set of constructed Gram matrices for the kernel functions presented in Table 4.2. The chosen hyperparameters are presented in the caption of the Figure. The linear kernel is repeated across the six subplots of its column since it is evaluated on the same data grid. Different structures of the underlying data can be captured through these kernels. The linear and the polynomial kernel, also presented in Rasmussen and Williams (2005), are not stationary kernels and, therefore, might provide more powerful insights if the underlying dataset does not carry such a property.

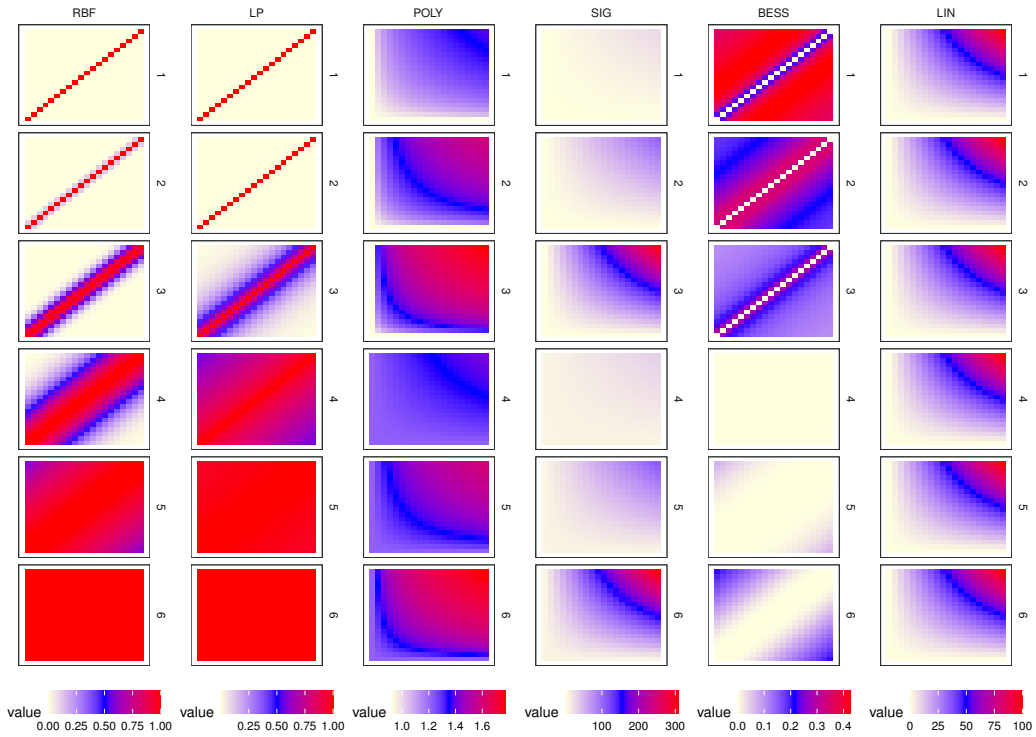


Figure 4.2: Figure presenting the Gram Matrices for the presented kernel in table 4.2. The selected grid of hyperparameters follows: for the radial basis function kernel $\gamma = [0.01, 0.1, 0.5, 1, 4, 10]$; for the laplace kernel $\gamma = [0.01, 0.1, 0.5, 1, 4, 10]$; for the polynomial kernel $\gamma = [0.5, 1]$, $r = [0.5, 7]$, $d = 0.1$; for the sigmoid kernel $\gamma = [0.5, 1, 2]$ and $r = [0.5, 7]$; and for the Bessel kernel $\gamma = [0.5, 1]$, $\nu = [0.5, 7]$ and $d = 0.1$.

4.3 Multi-Kernel Learning Combining

The framework presented involves classical kernel-based learning algorithms based on a single kernel to define similarity between pairs of points. A more recent approach that is progressively growing within various literature, particularly within the machine learning community, corresponds to multiple kernel learning (MKL). Gönen and Alpaydın (2011*b*) provides a detailed review of this concept. The reasoning behind such approaches involves two central aspects: first, different kernels correspond to different types of similarities across the data and, instead of choosing the one that works best, a combination of those allows for an added level of flexibility and a more efficient solution. Secondly, typical learning problems often involve multiple, heterogeneous data sources that carry diverse representations and require different kernels. In this case, multiple information sources can be achieved.

A second review of these approaches regarding multiple kernel learning algorithms is given in Peters (2017). It is noted that several strategies could affect different actions of the learning algorithms, i.e. one-stage versus two-stage kernel combining rules, kernel combination objectives and optimal solutions, boosted

kernel learners, and ensemble learners of multiple kernels. Furthermore, MKL methods could combine the kernel functions in several manners (Gönen and Alpaydın (2011*b*)), which can be both linear or non-linear functions. Two main approaches are usually found, one which only learns the combination function, or combination weights, for fixed kernels. The second option would instead learn both the kernel hyperparameters along with the optimal combination, i.e. the optimal weights.

The resulting multiple kernel combination is aligned with theorem 4.1.1 in the sense that a kernel function can be constructed from an inner product of a feature map and will still be a kernel as follows

$$k_\eta(\boldsymbol{x}_i, \boldsymbol{x}_j) = k_\eta\left(\langle \tilde{\varphi}_m(\boldsymbol{x}_i), \tilde{\varphi}_m(\boldsymbol{x}_j)\rangle\right) = f_\eta\left(\left\{k_n\left(\boldsymbol{x}_i^m, \boldsymbol{x}_j^m\right)\right\}_{m=1}^{M}\right) \tag{4.11}$$

where $\tilde{\varphi}_m(\boldsymbol{x}_i) = (\varphi_m \circ \varphi_{m-1} \circ \cdots \circ \varphi_1)(\boldsymbol{x}_i)$ or, analogously, combinations of kernels and $f_\eta : \mathbb{R}^M \to \mathbb{R}$ represents the combination function and could be linear or non-linear (multiplication, power, exponentiation). Therefore, the formulation of different kernels families characterising the discrimination boundary in the data (or state-space) can be easily obtained. The desired outcome consists of achieving linear discrimination through the kernel space embedding, with the selection of either features and kernels being critical to the performance of this method. In this work, a variety of such choices that involve different aspects of the EMD is explored. Particularly, combinations of different EMD features embedded through different kernel functions will be explored in Chapter 5 within the SVM framework, and in Chapter 6 within the stochastic embedding setting. Results will be given in part III, Chapters 8 and 9 for multiple speech experiments.

In this work, the use of a convex weighted combining rule will be employed and is defined as follows

$$k_\eta(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{m=1}^{M} \eta_m k_m(\boldsymbol{x}_i^m, \boldsymbol{x}_j^m) \tag{4.12}$$

where the kernel weights may be selected in different ways. Within such construction, each $k_m(\boldsymbol{x}_i^m, \boldsymbol{x}_j^m)$ characterizes a distinct sub-set of features of the data. It is then possible to interpret the contribution of each individual component to the learning process. The η coefficients can be interpreted to understand which features are more relevant for discrimination. In order to estimate such η weights, the approach of Tanabe et al. (2008) is adopted in this thesis. This uses the performance obtained by each kernel separately, by selecting

$$\eta_m = \frac{\pi_m - \delta}{\sum_{m=1}^{M}(\pi_m - \delta)} \tag{4.13}$$

where π_m is the accuracy of k_m used individually and δ is the threshold that should be less than or equal to the minimum of the accuracies obtained from single-kernel learners.

The above framework will be exploited in Chapter 5 to construct a more powerful SVM framework. The same approach will also be exploited in the case of the stochastic embedding presented in Chapter 6. In preparation, a review of MKL procedure in the context of Gaussian processes will be further discussed below in subsection 4.4.

4.4 Families of Kernel for Gaussian Processes

This section will present the kernel functions often used as a default choice related to the Gaussian process framework. This work provides the necessary building blocks required for Chapter 6, which proposes a stochastic embedding of the EMD basis functions, achieved through the use of Gaussian processes. The choice of the kernel strongly affects the inference performance of the Gaussian process on the given task. This section will present the properties characterising such a stochastic process, given and controlled by their covariance operator, corresponding to a definite positive covariance kernel function.
Gaussian processes were introduced in the machine learning community as an alternative inference method when Neal (2012) observed that Bayesian neural networks became Gaussian processes if the number of hidden units approached infinity. A Gaussian process constructs a prior over functions rather than over parameters (Wilson and Adams, 2013, Rasmussen and Williams, 2005). A more detailed discussion about Gaussian processes will be provided in Chapter 6.

The first set of kernel functions corresponds to the standard stationary kernels often encountered in the Gaussian process literature and reviewed in Rasmussen and Williams (2005, Chapter 4). These correspond to kernel functions which are a function of the distance between points of the input domain $\mathcal{X}$, but not of the points themselves, hence not of $\boldsymbol{x}$. The ones considered in this work are the square exponential, the rational quadratic, the periodic and the locally periodic. Gaussian processes are often used as a tool for pattern discovery, with parametric kernels employed by default. As highlighted in Wilson and Adams (2013), even the square exponential kernel, which is the most used in practice, merely acts as an effective smoothing interpolator that cannot easily reproduce the non-stationarity and non-linearity properties of the underlying system. Therefore, these kernels partly weaken the Gaussian process framework if applied to such challenging data structure settings. Various approaches have been proposed in the literature to resolve this issue.

To detect such complex, hidden data structures, one of the proposed solutions is combining Gaussian processes with different types of Bayesian neural networks and shape an alternative way to construct more expressive covariance kernel functions. See amongst other Salakhutdinov and Hinton (2007), Wilson et al. (2011), Damianou and Lawrence (2013). These methods present limitations as they are usually defined to tackle a specific kind of structure, i.e. input-dependent correlations between different tasks, and rely on a combination of simple kernels. Furthermore, the interpretation of the obtained results is often challenging, and

the computational cost associated with the implemented inference procedures is highly demanding.

The second procedure proposed in the literature pursues the idea that more refined kernel structures can be achieved by adding, multiplying or composing a few standard existing kernels and still obtain other positive definite kernel functions. These kernel designs set up the basis for the concept of multiple kernel learning (MKL) which was introduced above (see Gönen and Alpaydın (2011*a*) for a good review). The aim is to obtain a richer representation of the data with the combination of predefined kernels. This can be achieved through several strategies, for example, hierarchical kernel learning (HKL) (Bach, 2008, Jawanpuria et al., 2015), which learns from a set of base kernels assumed to be embedded on a directed acyclic graph. Bach et al. (2004) based on Lanckriet et al. (2004) proposed a dual formulation of the quadratically constrained quadratic program associated with the learning optimisation procedure of the coefficients of the kernel combination as a second-order cone programming problem. Archambeau and Bach (2011) selected a convex combination of kernel matrices with sparsity of the kernel weights by exploiting a hierarchical Bayesian approach. Durrande et al. (2016) utilized an additive GPs with additivity within the kernel function through a parsimonious numerical method for data-driven parameter estimation. The critical point of these methods is imposing diverse kinds of restrictions, making them less flexible or general for various applications. Without these constraints, however, the proposed kernel function constructions of these methodologies would easily lead to overfitting and, therefore, cannot be relaxed. As highlighted in Wilson and Adams (2013), some combinations have direct interpretable results while others do not. Hence, the task of constructing an effective inductive bias for kernel compositions leading to the discovery of the statistical structure of a Gaussian process is indeed arduous. In general, any stochastic process covariance function study faces a challenge, if no further assumptions or restrictions are made.

Another line of multiple kernel learning approaches considers the assumption that it is possible to model Gaussian processes covariance function by acting on their power spectral density and then convert it back with the inverse Fourier transform. This concept relies on Bochner's theorem (later introduced) and on the fact that it is much easier to reach the positivity requirement of the power spectral density, rather than the positive definiteness of the covariance kernel function. These ideas have been introduced by Wilson and Adams (2013) and refined or exploited in Remes et al. (2017), Samo and Roberts (2015), Tobar et al. (2015), Lázaro-Gredilla et al. (2010) and Tompkins and Ramos (2018) They have been shown to achieve complex kernel compositions carrying a more expressive way of modelling. Some of these approaches have been used for the classification tasks developed in later Chapters and are presented below.

Another class of kernels for Gaussian processes can be considered in this settings is the one proposed by Sauer et al. (2021) Volodina and Williamson (2020) Ming et al. (2021). These approaches rely on the use deep learning procedures that

can be used to produce Gaussian Processes emulators. Volodina and Williamson (2020) proposed to fit nonstationary GP emulators in response to the restrictions imposed by weak stationary GP by specifying finite mixtures of region-specific covariance kernels. In general, the method firstly fits a stationary GP. If nonstationarity is detected through traditional diagnostics, those diagnostics are then used to fit suitable mixing functions capturing such a property. Another approach dealing with nonstationarity of the underlying signal is the one given in Sauer et al. (2021). Deep Gaussian Processes (DGPs) are employed in this work so to reproduce abrupt regime changes in training data. The approach of this work relies on active learning (AL) strategies that distribute runs non-uniformly in the input space, which is something that a standard GP would not achieve. Even though this kind of approach appears to be successful to handle nonstationarity in general, DGPs are actually affected by the problem that variational distributions could be poor representations of the true posterior distributions, especially when multi-modality is present in the dataset. Ming et al. (2021) proposed a novel DGP inference method using stochastic imputation. This introduces a simple while efficient DGP training procedure which considers optimisations of conventional stationary GPs. These approaches represent alternative methods to fit nonstationary GPs efficiently and are relevant in this work, given the interest in nonstationary signals applications.

An alternative way to construct a kernel falls into the category of data-driven kernel functions. As highlighted in this review, Abbasnejad et al. (2012), there are three types of kernel learning procedures: parametric, non-parametric and data-driven. The presented framework and the methods considered in this work include parametric approaches, the most used methodology in practice within kernel algorithms. This kind of procedure uses an optimisation process that seeks the parameters of a predefined model. Following what was introduced above, there are two main categories of parametric methods: single base kernel and multiple base kernel. This thesis investigates both approaches in the context of Gaussian processes.

The second type of kernel learning algorithm is known as the non-parametric kernel learning, (Hoi and Jin, 2008, Hoi et al., 2007, Raina et al., 2007) and considers no prior model for the kernel. Consequently, the algorithm objective function is formulated with criteria to find an optimal kernel. The main shortcoming of this class of methods is that the optimal kernel has to be built from the training and test data during testing, since there is no further model to be used for testing.

The third family of methods that could be considered in kernel learning is the data-driven or data-dependent approach. Examples of these kernels are the Fisher kernel (Jaakkola, Haussler et al., 1999) and the marginalisation kernel (Herbster et al., 2005). The advantage of these methods is that they carry parameters directly defined on the underlying data rather than a priori given by the kernel method. The significance of these methods lies in the fact that their optimisation step is related to estimating the data-driven model parameters

through relatively simple solutions. In this work, the fisher kernel, as proposed in Jaakkola, Haussler et al. (1999), is exploited in the Gaussian process stochastic embedding proposed in Chapter 6 and below presented and discussed.

This section begins with a review of the traditional stationary kernel functions, presenting their corresponding Gram Matrices. A review of Bochner's theorem and the spectral mixture kernels relying on this theorem are discussed. Finally, the Fisher Kernel and its construction will be described. Note that for consistency within the subsections, the input variable for the kernel functions is denoted by $\boldsymbol{x}$. However, the real experiments are applied to speech signals and therefore, will be a function of time instead with the input variable corresponding to $\boldsymbol{t}$.

4.4.1 Stationary Kernels

Stationary kernels are an important and highly studied subset of kernels. A stationary kernel is a kernel whose value is a function of $\boldsymbol{x}-\boldsymbol{x}'$, i.e. it is invariant to translation of the inputs:

$$\begin{aligned} k(\boldsymbol{x},\boldsymbol{x}') &= k(\boldsymbol{x}-\boldsymbol{x}') \\ k(\boldsymbol{x}+\boldsymbol{z},\boldsymbol{x}'+\boldsymbol{z}') &= k(\boldsymbol{x},\boldsymbol{x}') \end{aligned} \tag{4.14}$$

In this work, the following stationary kernels, whose utilisation with GPs is fully described in Rasmussen and Williams (2005), are employed for classification purposes.

Kernel	**Formula**	$\boldsymbol{\theta}_k$
Square Exponential (SE)	$k_{SE}(\boldsymbol{x},\boldsymbol{x}') = \exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{x}')^2}{2l^2}\right)$	l
Rational Quadratic (RQ)	$k_{RQ}(\boldsymbol{x},\boldsymbol{x}') = \left(1+\frac{(\boldsymbol{x}-\boldsymbol{x}')^2}{2\alpha l^2}\right)^{-\alpha}$	α, l
Periodic (PR)	$k_{PR}(\boldsymbol{x},\boldsymbol{x}') = \exp\left(-\frac{2\sin^2(\pi\lvert\boldsymbol{x}-\boldsymbol{t}'\rvert/p)}{l^2}\right)$	l, p
Locally Periodic (locPR)	$k_{locPR}(\boldsymbol{x},\boldsymbol{x}') = k_{SE}(\boldsymbol{x},\boldsymbol{x}')k_{PR}(\boldsymbol{x},\boldsymbol{x}')$	l_1, l_2, p

Table 4.3: Description of the stationariy kernel functions employed in in Chapter 6 to study a stochastic embedding of the EMD. Note that $\boldsymbol{\theta}_k$ represents the set of hyperparameters used in the formula (given in the table). l represents the length scale; α represents the relative weighting of scale variations; p represents the period within both the periodic and locally periodic kernel; l_1 and l_2 are the two different length scales of the locally periodic kernel. Further details about the hyperparameters are provided in the text below.

This class of kernels are parametric functions specified by a set of hyperparameters controlling the structural power of the kernel on the underlying data. Each function carries specific hyperparameters, and each hyperparameter covers a particular role in shaping the kernel. Note that l represents the length scale and

determines the length of the oscillations of the underlying signal. α within the rational quadratic kernel, corresponds to a relative weighting of large-scale and small-scale length variations. This kernel is equivalent to adding together many SE kernels. p within the periodic and locally periodic kernels represents the period parameter and determines the distance between repetitions of the underlying signal. l_1 and l_2 are parameters for the two different length scales of the locally periodic kernel, where l_1 corresponds to the length scale of the exponentiated quadratic, while l_2 corresponds to the length of the periodic function.

Given its flexibility, the square exponential kernel (also known as the radial basis function kernel) is the most used in practice. It has the property of being a universal kernel (Micchelli et al., 2006) and it can be integrated against most functions. Furthermore, Rasmussen and Williams (2005) show that it is infinitely differentiable, which means that a Gaussian process using this kernel structure for its covariance function will have mean square derivatives of all orders and, therefore, will be highly smooth. The rational quadratic represents an alternative to the square exponential since it corresponds to the superimposition of many square exponential kernels. When a Gaussian process comes into play, one often employs these two kernels as the first choice. However, as underlined in Wilson and Adams (2013), these are no more than smooth interpolators. If the underlying signal has discontinuities, is discontinuous in its first derivative or shows a high level of non-stationarity or non-linearity, the length scale of these kernels will be usually learnt according to the shortest oscillation of the signal. As a result, it would fail to fit different time-varying data regions. On the other hand, one could use the periodic or the locally periodic kernel functions. These provide additional flexibility to model periodic signals over time, or signals which are periodic at a local level by multiplying a periodic kernel with a square exponential one.

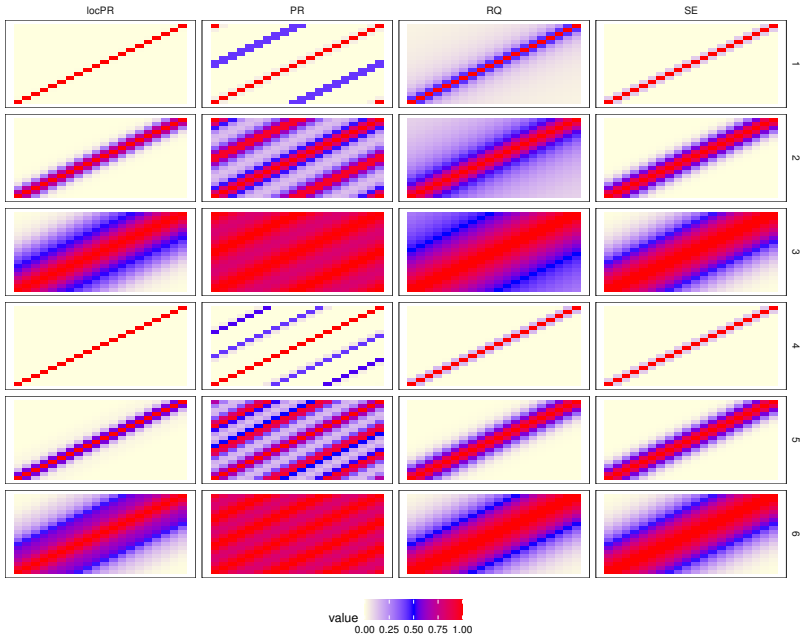


Figure 4.3: The different Gram Matrices for the presented kernel in table 4.3. Note that the selected grid of hyperparameters is given as follows: for the square exponential, $l = [0.25, 1, 3]$. For the rational quadratic $l = [0.25, 1, 3]$ and $\alpha = [0.5, 7]$. For the periodic kernel $l = [0.25, 1, 3]$ and $p = [0.5, 7]$. For the locally periodic kernels $l_1 = l_2 = [0.25, 1, 3]$ and $p = [0.5, 7]$.

Figure 4.3 shows different Gram matrices for the introduced kernels. Note that for the square exponential, the three Gram matrices repeats since only three cases are taken into account, i.e. $l = [0.25, 1, 3]$. These plots show how the selected hyperparameters strongly change the shape of the Gram matrix and, therefore, greatly affect the task of interest. This demonstrates the importance of the chosen kernel as a keystone in the Gaussian process literature.

Next, Bochner's theorem is introduced. Such theorem is presented to equip this thesis framework in the development of the spectral mixture kernel given in Wilson and Adams (2013). In this work, the authors design the power spectral density of a stationary scalar-valued Gaussian process by a mixture of square exponential functions. By then exploiting Bochner's theorem (Bochner, 1953, Bochner et al., 1959), the spectral mixture kernel can be computed via the inverse Fourier transform of the obtained power spectral density.

4.4.2 Bochner's theorem

It is standard practice to represent a stationary kernel as function of a single argument given as $k(\boldsymbol{\tau})$, where $\boldsymbol{\tau} = \boldsymbol{x} - \boldsymbol{x}'$.

The covariance function of a stationary process, or simply, a stationary kernel can be fully characterised by its spectral representation derived by Bochner (1953). Bochner's theorem defines a one-to-one mapping from stationary kernels to finite

measures via Fourier transform.

Theorm 4.4.1 (Bochner)**.** *A complex-valued function k on $\mathbb{R}^P$ is the covariance function of a weakly stationary mean square continuous complex-valued random process on $\mathbb{R}^P$ if and only if it can be represented as*

$$k(\boldsymbol{\tau}) = \int_{\mathbb{R}^D} e^{2\jmath\pi \boldsymbol{s}^\intercal \boldsymbol{\tau}} \psi_k(d\boldsymbol{s}) \tag{4.15}$$

where ψ_k is a positive finite measure.

If ψ_k has a density $S(\boldsymbol{s})$, then S is called the spectral density or power spectrum of k, and k and S are Fourier duals :

$$\begin{aligned} k(\boldsymbol{\tau}) &= \int S(\boldsymbol{s}) e^{2\jmath\pi \boldsymbol{s}^\intercal \boldsymbol{\tau}} d\boldsymbol{s} \\ S(\boldsymbol{s}) &= \int k(\boldsymbol{\tau}) e^{-2\jmath\pi \boldsymbol{s}^\intercal \boldsymbol{\tau}} d\boldsymbol{\tau} \end{aligned} \tag{4.16}$$

For a proof see Yaglom (2012).

Therefore, the properties of a stationary kernel can be entirely described by its spectral density. This theorem, and the duality implied by it, are employed in the Gaussian process literature to circumvent the difficulty often affecting this community in defining positive-definite functions to design stationary covariance kernels. The central motivation is that the positivity requirement of the power spectral density might be more easily achieved than the positive definiteness requirement of the covariance kernel.

In the following subsection, the work proposed in Wilson and Adams (2013) exploiting Bochner's theorem is presented.

4.4.3 Spectral Mixture Kernels

The above duality has been exploited in Sinha and Duchi (2016), Yang et al. (2015) to design stationary kernel representations. Lázaro-Gredilla et al. (2010) utlized the duality to learn the spectral density as a mixture of Dirac delta functions leading to a sparse spectrum kernel. It has also being used for large-scale inference in Rahimi et al. (2007). The interest in such duality in this work arises when considering the approach proposed in Wilson and Adams (2013), who derived a stationary spectral mixture (SM) kernel by formulating any stationary Gaussian process covariance kernel function with spectral densities corresponding to scale-location mixture of Gaussians. Such a construction is achieved through two main facts:

1. The square exponential kernels and a mixture of those provides Gaussian spectral densities are centered around the origin. Therefore, having non-zero mean Gaussians would offer much more flexible spectral densities.

2. Mixtures of Gaussians are dense in the set of all distribution functions.

The formulation for the obtained $S_{SM}(\boldsymbol{s})$ and $k_{SM}(\boldsymbol{\tau})$ are derived and given as follows:

$$\begin{aligned} k_{SM}(\boldsymbol{\tau}) &= \sum_{q=1}^{Q} w_q \prod_{p=1}^{P} \exp(-2\pi^2 \boldsymbol{\tau}_p^2 v_q^{(p)}) \cos(2\pi \boldsymbol{\tau}_p \mu_q^{(p)}) \\ S_{SM}(\boldsymbol{s}) &= \sum_{q=1}^{Q} w_q \left[\mathcal{N}(\boldsymbol{s}|\boldsymbol{\mu}_q, \boldsymbol{M}_q) + (-\boldsymbol{s}|\boldsymbol{\mu}_q, \boldsymbol{M}_q)\right]/2 \end{aligned} \tag{4.17}$$

where $S_{SM}(\boldsymbol{s})$ is a mixture of Q Gaussians on $\mathbb{R}^P$, where the q^{th} component has mean vector $\boldsymbol{\mu}_q = (\mu_q^{(1)}, \ldots, \mu_q^{(P)})$ and covariance matrix $\boldsymbol{M} = \text{diag}(v_q^{(1)}, \ldots, v_q^{(P)})$ and $\boldsymbol{\tau}_p$ is the p^{th} component of the P dimensional vector $\boldsymbol{\tau} = \boldsymbol{x} - \boldsymbol{x}'$. Note that the spectral densities are symmetric.

This class of kernel has been exploited in Parra and Tobar (2017), who constructed a spectral mixture kernel for multi-output Gaussian processes and it was extended to large scale multidimensional pattern extrapolation in Wilson et al. (2014). In general, spectral kernels work highly efficiently in expressing kernels with long-range, non-monotonic or periodic structures (see for example Tompkins and Ramos (2018)). There have also been extensions to handle non-stationarity as given in Remes et al. (2017) and Wilson (2014), and Samo and Roberts (2015). These are beyond the main scope of this thesis and will not be considered.

For simplicity and without loss of generality, $Q = 1$ is chosen, and, therefore the kernel will be given as follows:

$$k(\boldsymbol{\tau}) = k(\boldsymbol{x}, \boldsymbol{x}') = \exp(-2\pi^2(\boldsymbol{x} - \boldsymbol{x}')^2\sigma^2)\cos(2\pi(\boldsymbol{x} - \boldsymbol{x}')\mu) \tag{4.18}$$

Figure 4.4 represents the Gram matrices for the spectral kernel presented in equation 4.18, with hyperparameters $\mu = 10, \sigma^2 = 0.5$ for the left panel and $\mu = 30, \sigma^2 = 00.5$ for the right panel of the top row. In the bottom row the Gram matrices are generated with $\mu = 100, \sigma^2 = 1$ and $\mu = 2, \sigma^2 = 0.0005$ in the left and right panels, respectively. It is possible to observe that even with a unique Gaussian component, such Gram matrices express a great deal more of structural changes compared to the ones provided by the stationary kernels.

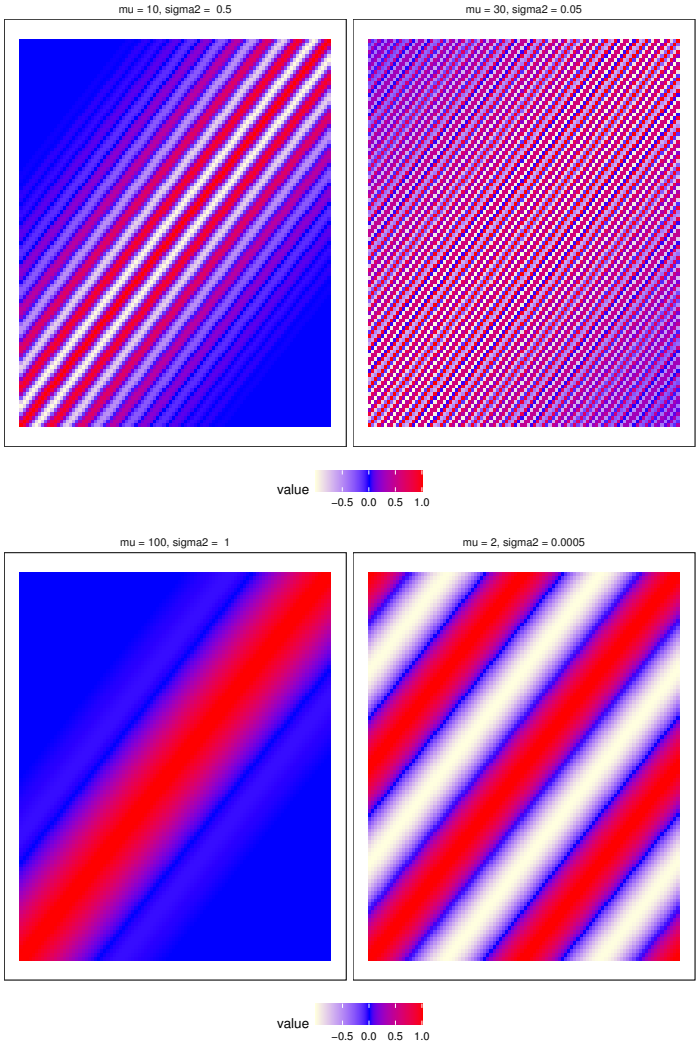


Figure 4.4: Figure presenting the Gram Matrices for the presented kernel in equation 4.18. Note that the hyperparameters values for μ and σ^2 are given above the plots.

4.4.4 The Fisher Kernel

following kernel structure considered in the speech application experiments is the Fisher kernel. As introduced above, this kernel function falls into the class of data-driven kernel functions and has been widely used since fully data adaptive. This means that it relies completely on the data rather than adopting any parametric form or nonparametric solution.

To understand the idea behind such kernel function, the two main approaches when dealing with classifications problems of a complex dataset will be discussed. The first is data engineering into sequences of variable-sized arrays combined with a generative model, and then the classification is carried with a Bayes Rule. The second employs a discriminative method, directly estimating either the posterior probability or the discriminant function for the class label and

appears to offer better solutions in the classification task. Jaakkola, Haussler et al. (1999) proposed the Fisher kernel as a link between these two approaches to obtain a more robust classifier and map a variable length sequence onto a new fixed dimension feature vector space. The mapping is determined by the gradient of the log-likelihood of the parameters of an underlying generative model and defines a new feature space called the Fisher score space. It describes how that parameter contributes to the process of generating a particular input data. The gradient maintains all the structural assumptions that the model encodes about the generation process.

Its primary aim is to provide a generic mechanism incorporating generative probability models into discriminative classifiers. It has been further extended by Jaakkola and Haussler (1999*b*) who proposed a general class of probabilistic regression models and a parameter estimation technique that uses arbitrary kernel functions. The main reason for introducing such a kernel function is to classify complex data types such as speech, text or bio sequence data. The examples that follow demonstrate the successful application of this method. Jaakkola, Diekhans and Haussler (1999) applied the Fisher kernel method to detect remote protein homologies, which performed well in classifying protein domains by SCOP (Structural Classification of Proteins) superfamily. Jaakkola et al. (2000) found that using the Fisher kernel significantly improved previous methods for the classification of protein domains based on remote homologies. Moreno and Rifkin (2000) used the Fisher kernel method for large scale Web audio classification. Smith and Niranjan (2000) presented experimental justification for the Fisher kernel, explaining that it limits the dimension of the feature space by giving some beneficial regularisation, especially when the two classes are inseparable. Vinokourov and Girolami (2001) successfully employed the Fisher kernel for document classification. Fine et al. (2001) used SVMs for speaker verification and identification tasks. The reader might refer to Sewell (2011) for a well-presented summary of the kernel.

One of the applications of this thesis concerns the classification task involving patients affected by Parkinson's disease. The stochastic embedding proposed in Chapter 6 will be utlized in this application in later Chapters, with motivations and reasonings provided in part III.

The Fisher kernel has been successfully employed within speech verification, and recognition tasks by Fine et al. (2001) and Smith and Gales (2001). The selected role for this work consists of detecting voice disturbances in displacement, direction and velocity to differentiate voices of healthy patients and voices of ill subjects affected by Parkinson's disease. Differences in the trend or magnitude of the scoring vectors should reflect different features of the underlying generative model and hence provide the desired discrimination power. The main strength of this method is given by the kernel rather than by the employed time-series model. Particularly, a voice affected by Parkinson's' disease versus one of a healthy subject should provide variations in terms of the considered scoring method since the two generative models are intrinsically different.

To introduce such a kernel function, note that a change of the input variable is in place. Hence, while the rest of the Chapter ha been introduced by using $\boldsymbol{x}$, this time the input variable becomes $\boldsymbol{t}$. The reasons to do so is to avoid confusion when the applications Chapters will be introduced. The derivation of the Fisher kernel will be done in the setting of speech signals where one usually works with a function of time, hence $\boldsymbol{t}$. Therefore, the author decided to stick to the notation of the applied section rather than using $\boldsymbol{x}$.

Consider the signal $\tilde{s}(\boldsymbol{t})$ as given in equation 3.1, in Chapter 3. Consider now a probability model for $\tilde{s}(\boldsymbol{t})$ whose probability density function is denoted as $f(\tilde{s}(\boldsymbol{t})|\boldsymbol{\theta}_k)$, where $\boldsymbol{\theta}_k$ is a vector of the model parameters. Define $\nabla\boldsymbol{\theta}_k$ as the gradient operator with respect to $\boldsymbol{\theta}_k$ and $\log_e f(\tilde{s}(\boldsymbol{t})|\boldsymbol{\theta}_k)$ is the log-likelihood with respect to the model with a given set of parameters $\boldsymbol{\theta}_k$. Then the Fisher score, $U_{\boldsymbol{\theta}_k}(\mathbf{t})$, is the gradient of the log-likelihood with respect to the model with a given set of parameters $\boldsymbol{\theta}_k$.

$$U_{\boldsymbol{\theta}_k}(\mathbf{t}) = \nabla\boldsymbol{\theta}_k \log_e f(\tilde{s}(\boldsymbol{t})|\boldsymbol{\theta}_k) \tag{4.19}$$

It provides an embedding into the feature space. The Fisher kernel refers to the inner product in this space, and is defined as

$$k(\mathbf{t}, \mathbf{t}') = U_{\boldsymbol{\theta}_k}(\mathbf{t})^\intercal \; I^{-1} \; U_{\boldsymbol{\theta}_k}(\mathbf{t}') \tag{4.20}$$

where I is the Fisher Information Matrix and is defined as $I = \mathbb{E}\left[U_{\boldsymbol{\theta}_k}(\mathbf{t}) \; U_{\boldsymbol{\theta}_k}(\mathbf{t})^\intercal\right]$. Remark that the Fisher Information Matrix measures the amount of information that $S(\mathbf{t})$, i.e. the random process whose realisation corresponds to $\tilde{s}(\mathbf{t})$, carries about $\boldsymbol{\theta}_k$. In practice, the Fisher score $U_{\boldsymbol{\theta}_k}(\mathbf{t})$ maps $\tilde{s}(\boldsymbol{t})$ into a feature vector that is a point in the gradient space of the manifold $M_{\boldsymbol{\Theta}_k}$. This mapping is referred to as Fisher score mapping (Jaakkola, Haussler et al., 1999). The gradient $U_{\boldsymbol{\theta}_k}(\mathbf{t})$ can be used to define the direction δ which maximizes $\log_e f(\tilde{s}(\boldsymbol{t})|\boldsymbol{\theta}_k)$ while traversing the minimum distance in the manifold, as defined by $D(\boldsymbol{\theta}_k, \boldsymbol{\theta}_k + \delta)$, where $D(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$. This latter gradient is usually known as natural gradient and is obtained from the ordinary gradient via $\phi_{\boldsymbol{\theta}_k}(\boldsymbol{t}) = I^{-1}U_{\boldsymbol{\theta}_k}(\mathbf{t})$. Hence, the mapping $\tilde{s}(\boldsymbol{t}) \rightarrow \phi_{\boldsymbol{\theta}_k}(\boldsymbol{t})$ is called the natural mapping and the natural kernel associated to it corresponds to the inner product between these feature vectors relative to the local Riemannian metric as given in Equation. 4.20. Note that the information matrix is asymptotically immaterial and that the simpler kernel that can be taken into account and it will be the one employed in this work is

$$k(\mathbf{t}, \mathbf{t}') = U_{\boldsymbol{\theta}_k}(\mathbf{t})^\intercal \; U_{\boldsymbol{\theta}_k}(\mathbf{t}') \tag{4.21}$$

This kernel will be employed in Chapter 9 to develop an ad hoc methodology for the given classification task.

Chapter 5

SVM Classifier and Statistical Interpretation

The first step in classification tasks is to summarise the underlying signals through feature extraction. The EMD captures several non-stationary attributes of the considered time-series through different spaces such as parameter space, basis space, and instantaneous frequencies. These representations were reviewed in Chapter 4, subsection 4.1.1, as they correspond to a relevant and important step for kernel learning procedures. Table 4.1 shows the EMD based features that will be used in the synthetic and speech experiments of part III. When working with kernel methods and large hyperplane classifiers to ensure that the resulting numerical procedures are robust, a normalisation step is applied before every classification exercise to each feature.

The framework of this Chapter relies on the Support Vector Machine as the preferred classifier. Such a technique corresponds to a supervised machine learning method widely employed to solve both classification or regression problems. In classification tasks, the problem solved is the identification of a hyperplane able to separate the given data points into two (the problem is usually a binary classification problem) classes. Intuitively, a good hyperplane maximises the distance between the nearest training data point of any class. Further discussion will be provided in the sections below. It is essential to highlight that such a machine learning technique is often far from the statistical interpretation required when solving data analysis. It is, therefore, of the primary aim of this Chapter to provide a statistical interpretation of such a classifier by relying on two critical approaches introduced in the literature. The discussion is below provided.

One issue often affecting the machine learning community is to visualise high-dimensional datasets through an efficient and computationally fast technique. Several solutions have been proposed in the last few decades (the reader might refer to Maaten and Hinton (2008) and references within for further details) as Chernoff faces, pixel-based techniques and techniques that represent the dimensions in the data as vertices in a graph. The main drawback of these methods is that they represent tools that can display multiple data dimensions and leave

the interpretation to the human eye. This represents a hazardous procedure and, indeed, a subjective one. Instead, dimensionality reduction techniques could represent the alternative approach, providing a low dimensional data set that can be plotted in a scatterplot. A dimensionality reduction method aims to retain as much information as possible of the high dimensional data by mapping it into a low dimensional one. Traditional examples are represented by the Principal Component Analysis (Hotelling, 1933) or the classical multidimensional scaling (Torgerson, 1952) corresponding to linear methodologies aiming to obtain a low dimensional representation of dissimilar data points far apart. The central issue with such methods is their linearity assumption, producing unreliable results when non-stationary and non-linear real data plays the central role. As a response to such an issue, numerous non-linear solutions have been implemented (Demartines and Hérault, 1997, Sammon, 1969, Hinton and Roweis, 2002). In this thesis, the t-SNE proposed by Maaten and Hinton (2008) is employed. This technique will not only provide a good solution for data visualisation in Chapter 8, but it also provides a tool for the interpretation of the optimal hyperplane often found into a high dimensional space obtained through a kernel function. Further discussion is provided in the sections below.

5.1 Classification framework: EMD-Support Vector Machine

Support Vector Machine (SVM) is a method of supervised machine learning which allows for classification and regression based on structural risk minimization (see Cortes and Vapnik (1995)). The goal is determining a hyperplane of separation with the maximum distance to the closest points of the identified classes. These points are called *Support Vectors*. By considering a training set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$, a feature vector $\boldsymbol{x}_i \in \mathbb{R}^D$ and class labels $y_i \in \{-1, +1\}$, the hyperplane of separation can be defined as $d(\boldsymbol{x}_i, \boldsymbol{w}, b) = \boldsymbol{w}^\intercal \cdot \boldsymbol{x}_i + b = 0$, where $\boldsymbol{w} \in \mathbb{R}^D$ represents the weight vector and b a scalar. The optimal hyperplane that separates data into two classes is the one that minimises the following objective function:

$$\left[\frac{1}{n}\sum_{i=1}^{n}\max\left(0, 1 - y_i(w \cdot x_i - b)\right)\right] + \lambda\|w\|^2 \tag{5.1}$$

This corresponds to a quadratic optimisation problem (Manjula et al., 2011) and can be solved in the parameter space with respect to $\boldsymbol{w}$ and b. There are several solutions to solve this problem as the sub-gradient descend and coordinate descend. In this work, we adopt the quadratic optimisation problem which can be given by its primal form. For all $i \in \{1, \ldots, n\}$, the so called slack variable $\xi_i = \max\left(0, 1 - y_i(w \cdot x_i - b)\right)$ are firstly introduced, measuring the distances ξ_i of the points crossing their margin and incorporated in the optimisation; each ξ_i is the smallest non-negative number satisfying $y_i(w \cdot x_i - b) \geq 1 - \xi_i$. The

optimization problem is then given by:

$$\text{minimize } \frac{1}{n}C\sum_{i=1}^{n}\xi_i + \lambda\|w\|^2 \tag{5.2}$$

subject to $y_i(w\cdot x_i - b) \geq 1-\xi_i$ and $\xi_i \geq 0$, for all i. Note that C is the trade-off factor, compromising between the maximization of the margin and the minimization of the misclassification error. The primal problem is typically reformulated to as a dual problem through a Lagrangian and the solution is guaranteed if the Karush-Kuhn-Tucker conditions (Boyd and Vandenberghe, 2004) are verified. By solving for the Lagrangian dual, the problem then becomes:

$$\text{maximize } f(\alpha_1 \dots \alpha_n) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i\alpha_i(x_i\cdot x_j)y_j\alpha_j \tag{5.3}$$

subject to $\sum_{i=1}^{n}\alpha_i y_i = 0$, and $0 \leq \alpha_i \leq \frac{1}{2n\lambda}$ for all i. Since the dual maximization problem is a quadratic function of the α_i subject to linear constraints, it is efficiently solvable by quadratic programming algorithms. Here, the variables α_i are defined such that $\mathbf{w} = \sum_{i=1}^{n}\alpha_i y_i \mathbf{x}_i$. Moreover, $\alpha_i = 0$ exactly when $\mathbf{x}_i$ lies on the correct side of the margin, and $0 < \alpha_i < (2n\lambda)^{-1}$ when $\mathbf{x}_i$ lies on the margin's boundary. It follows that $\mathbf{w}$ can be written as a linear combination of the support vectors. The offset, b, can be recovered by finding an $\mathbf{x}_i$ on the margin's boundary and solving $y_i(\mathbf{w}\cdot\mathbf{x}_i - b) = 1 \iff b = \mathbf{w}\cdot\mathbf{x}_i - y_i$. (Note that $y_i^{-1} = y_i$ since $y_i = \pm 1$.)
The presented framework provide a linear classifier assuming linear separability of the data which is, in practice, rare to observe. The solution tackling this problem is known as kernel trick and extends such method to non-linear settings by projecting the data into the so called feature space through a non-linear map $\phi(\mathbf{x}_i)$. $\phi(\boldsymbol{x}, \boldsymbol{\Psi})$, parametrised by $\boldsymbol{\Psi} \in \mathbb{R}^d$, where data is linearly separable again. What is needed in the feature space is the definition of the inner product operation. A traditional way to do so is considering kernel function defined as $k(x_i, x_j) = \langle\phi(\boldsymbol{x_i}, \boldsymbol{\Psi}), \phi(\boldsymbol{x_j}, \boldsymbol{\Psi})\rangle$. In general, the definition of a kernel function is given as in Chapter 4 with theorem 4.1.1.
The classification vector $\mathbf{w}$ in the transformed space satisfies $\mathbf{w} = \sum_{i=1}^{n}\alpha_i y_i \phi(\mathbf{x}_i)$ where, the α_i are obtained by solving the optimization problem:

$$\begin{aligned}\text{maximize} f(\alpha_1 \dots \alpha_n) &= \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i\alpha_i\langle\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle y_j\alpha_j \\ &= \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i\alpha_i k(\mathbf{x}_i, \mathbf{x}_j) y_j\alpha_j\end{aligned} \tag{5.4}$$

subject to $\sum_{i=1}^{n}\alpha_i y_i = 0$, and $0 \leq \alpha_i \leq \frac{1}{2n\lambda}$ for all i. The coefficients α_i can be solved for using quadratic programming, as before. Again, we can find some index i such that $0 < \alpha_i < (2n\lambda)^{-1}$, so that $\phi(\mathbf{x}_i)$ lies on the boundary of the

margin in the transformed space, and then solve

$$\begin{aligned} b = \mathbf{w} \cdot \phi(\mathbf{x}_i) - y_i &= \left[\sum_{j=1}^{n} \alpha_j y_j \langle \phi(\mathbf{x}_j), \phi(\mathbf{x}_i) \rangle \right] - y_i \\ &= \left[\sum_{j=1}^{n} \alpha_j y_j k(\mathbf{x}_j, \mathbf{x}_i) \right] - y_i \end{aligned} \tag{5.5}$$

Finally, the optimal decision function of a classifier is $\mathbf{z} \mapsto \operatorname{sgn}(\mathbf{w} \cdot \phi(\mathbf{z}) - b) = \operatorname{sgn}\left([\sum_{i=1}^{n} \alpha_i y_i k(\mathbf{x}_i, \mathbf{z})] - b\right)$, producing non-linear classification decision boundaries dependent on the kernel choice.

5.2 Interpreting the kernel space linear-decision boundary in sub-spaces of the state space

In the SVM, the separating linear hyperplane has direct interpretation in the kernel space but finding its functional form in the state space or feature space is a highly non-linear inverse problem in general. Hence, interpreting the decision boundary in the data state-space for the SVM classification problem can be a challenging task. One way to assess and interpret the accuracy of the discrimination is by employing a projection method. In this work, the dimensionality reduction technique proposed by Maaten and Hinton (2008) and known as the t-Distributed Stochastic Neighbor Embedding is considered. It which converts a high-dimensional data set $\mathcal{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ into a two or three-dimensional data set $\mathcal{Y} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n\}$ which is easier to observe. Compared to other methods, t-SNE can detect both global and local structures in the data. It models the Euclidean distance between two high-dimensional data points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ according to the joint probabilities p_{ij}; such probabilities measure the pairwise similarity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ by symmetrizing two conditional probabilities as:

$$p_{j|i} = \frac{\exp\left(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\|\boldsymbol{x}_i - \boldsymbol{x}_k\|^2 / 2\sigma_i^2\right)} \qquad p_{i|i} = 0 \tag{5.6}$$

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2N} \tag{5.7}$$

The variance σ_i of the Gaussian centered over $\boldsymbol{x}_i$ is set such that the perplexity of the conditional distribution P_i equals a given perplexity $Perp(Pi)$, where the perplexity is an entropy measure defined as $Perp(Pi) = H(P_i)$ and $H(P_i)$ represents the Shannon entropy of P_i, i.e. $H(P_i) = -\sum_j p_{j|i} \log_2 p_{j|i}$, and can be interpreted as a smooth measure of the number of neighbours. The optimal value σ_i differs for each $\boldsymbol{x}_i$ and is provided by a binary search. To measure the similarity between two corresponding points $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ in the low-dimensional space $\mathcal{Y}$, a heavy-tailed distribution is considered:

$$q_{ij} = \frac{\left(1 + \|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \|\boldsymbol{y}_k - \boldsymbol{y}_l\|^2\right)^{-1}} \quad q_{ii} = 0. \tag{5.8}$$

A t-Student distribution with one degree of freedom (Cauchy distribution) instead of a Gaussian is used in the low dimension so that difference in volume between high- and low-dimensional spaces is considered. The identification of the points in the low dimension $\mathcal{Y}$ is given by minimizing the Kullback-Leibler divergence between the two joint distribution P and Q:

$$C = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{5.9}$$

Such cost function is minimized by applying the gradient-descent technique. The gradient is given by:

$$\frac{\partial C}{\partial \mathbf{y}_i} = 4 \sum_j (p_{ij} - q_{ij}) \left(1 + \|\boldsymbol{y}_i - \boldsymbol{y}_j\|^2\right)^{-1} (\boldsymbol{y}_i - \boldsymbol{y}_j) \tag{5.10}$$

As for the classic Stochastic Neighbor Embedding Hinton and Roweis (2003), the gradient descent is initialized at sampling map points coming from an isotropic Gaussian centered around zero with small variance. To speed up the optimization and also avoid poor local minima identification, the gradient is updated by a momentum term given by:

$$\mathcal{Y}^{(t)} = \mathcal{Y}^{(t-1)} + \eta \frac{\partial C}{\partial \mathcal{Y}} + \alpha(t)(\mathcal{Y}^{(t-1)} - \mathcal{Y}^{(t-2)}) \tag{5.11}$$

where $\mathcal{Y}^{(t)}$ represents the solution at iteration t, η is the learning rate and $\alpha(t)$ is the momentum at iteration t. For the t-Distributed Stochastic Neighbor Embedding, such momentum term works if it is small until the map points have become moderately well-organized. Another way used to improve the optimization is called "early exaggeration" Maaten and Hinton (2008), which consists of multiplying all the p_{ij}'s by a factor (i.e. 4 or 12) during the first steps of the optimization so that clusters present in the original dataset $\mathcal{X}$ will tightly group in the map $\mathcal{Y}$. All the experiments are run with a perplexity set equal to 20, an early exaggeration factor of 12 for the first 250 iterations, the maximum number of iteration T is 1000 and the momentum term $\alpha^{(t)} = 0.5$ for $t < 250$ and $\alpha^{(t)} = 0.8$ for $t \geq 250$. The learning rate η is set initially equal to 200 and then updated according to the scheme described in Jacobs (1988). In all the experiments using t-SNE to represent some of the considered features, used a first PCA step is used to reduce the dimensionality of the dataset; this is due to faster computation of the pairwise distances.

Chapter 6

A Stochastic Embedding For The EMD

This Chapter presents a stochastic embedding for the Empirical Mode Decomposition. As introduced in Chapter 1, the EMD has been employed within several fields and to solve various tasks. The most valuable property is that unlike many other time-frequency decomposition methods mapping the original signal onto a space spanned by a priori selected bases, the EMD defines the set of IMFs entirely derived from the data and deals with both non-stationary and non-linear systems. However, the essential, often unnoticed, assumption in applying this method is that the considered underlying signal is deterministic, foreseeing a pathwise decomposition method. If granted, this fact weakens the method per se and does not account for the natural randomness associated with the studied phenomenon. The first action embodied by this Chapter is to consider this comment and propose an embedding that preserves the EMD inherent construction but accounts for the stochastic nature of the underlying data system.

The goal is to take the sample path realisation of a process and achieve a stochastic representation of the EMD, hence turning it into an adequate stochastic model. The extraction procedure produces a set of IMFs whose sum will reproduce the original realised signal path. Therefore the interest is to seek a stochastic embedding that preserves the additive nature of the EMD. This will be achieved in the sense of infinite divisibility of the representation. Accordingly, the employment of convolutional stochastic representations that will preserve this additive property in a distributional and process sense is considered.

The concept of infinite-divisible distributions has been widely discussed and can be appointed to several works developed by Levy, Gnedenko-Kolmogorov, Feller, Sat, Steutel and van Harn. However, the pioneer is considered to be Bruno De Finetti, an Italian probabilist statistician and actuary whose work on infinite-divisible distributions was published in 1929. A review of these works is given in Mainardi and Rogosin (2008) and the references within. For more technical texts, the reader might refer to Steutel et al. (1979), Steutel and Van Harn (2003), Domínguez-Molina and Rocha-Arteaga (2007). The interest in this con-

cept within this work is represented by the need for an infinite-divisible stochastic process that can be decomposed into a finite number of stochastic processes carrying the same distribution as the original one. In this regard, several choices could be taken into account since different distributions carry the property of infinite divisibility. The other property that has to be carried by the convolutional decomposition obtained in this proposed stochastic embedding is that the resulting stochastic process has to be a second-order process having finite first and second moments that will fully characterise it.

The proposed embedding will then consider one GP for every IMF. The overall signal path will be characterised by the sum of GPs corresponding to the model that arises since one sums the IMFs to get the signal. As a result, one convolves the GPs for each IMF to obtain the signal stochastic embedding. Such an approach is equivalent to a multi-kernel representation of the signal where the number of kernel components is selected by the EMD sifting extraction procedure and is therefore signal adaptive. Furthermore, since each IMF has its GP representation, the kernel is tailored (to this local frequency component); this is akin to a multi-kernel representation that is frequency adaptive in non-stationary signals. Standard practices for a multi-kernel GP framework have been reviewed in Chapter 4, such as Duvenaud et al. (2011), Durrande et al. (2012), Van der Wilk et al. (2017) that propose either an additive kernel function or a convolutional kernel construction procedure, or Wilson and Adams (2013) and Lázaro-Gredilla et al. (2010) who suggested a multi-kernel frequency approach acting on the power spectral density of the kernel. These methods differ from the one explained as, this time, an additive kernel will sum over every input dimension (the IMFs) to derive a final representation of the original inferred function (the given signal). It is more advanced since fully data-adaptive by considering the non-stationarity of each sample at multiple frequencies with the number of kernel components linked to the obtained IMFs, rather than the use of a heuristic rule.

There will be two solutions for constructing the desired stochastic embedding. The first one, above discussed, considers one GP for each IMF. The second one will instead be based on a different intuition, i.e. on the idea that there should be one stochastic model for each frequency band rather than for every IMF. The covariance function operator considered in this case will provide information on the frequency content of the different frequency regions or, more formally, frequency bandwidths characterising the stochastic process of the original signal. This model will be achieved by constructing an optimal partition of the instantaneous frequency samples based on the cross-entropy optimisation method (introduced in Chapter 7). Remark that the instantaneous frequencies should be ordered according to the IMF index, given that the highest IMFs carry the highest frequencies and the lowest IMFs carry the lowest frequencies. In practice, the sifting procedure might produce mixed frequency IMFs, and the IFs will reflect such a fact by carrying multiple frequency contents. This phenomenon, usually called mode-mixing, might affect the EMD sifting procedure and provide unreliable results. However, suppose the IFs plane is partitioned according to a

newly defined grid (whose definition will be later introduced) which is optimal in the sense that it efficiently separates the instantaneous frequency values within frequency bandwidths that are highly concentrated. In that case, the existing IMF values can be aggregated according to the location of their correspondent IFs within specific frequency bandwidths. A stochastic embedding over these newly aggregate quasi-IMFs (QIMFs) will be proposed. The power of this model is that it provides a new redefined EMD decomposition method that is fully data-adaptive, also from a frequency domain perspective.

There are several points for which the stochastic embedding of the EMD is relevant. Firstly, this multi-kernel GP stochastic embedding allows one to impose a stochastic ordering between the IMF embeddings. The IMFs already preserve the notional oscillatory ordering captured during their construction, given that each IMF has one less convexity change than the previous. A stochastic ordering can be achieved by imposing a common covariance operator, such as a universal square exponential radial basis function kernel, across the IMFs. Then, for each successive IMF, ensure that the hyperparameters are strictly ordered relative to those of the previous IMFs to guarantee that the representation achieves either first order or second order stochastic dominance. Such an approach is superior to the one proposed in the literature since the added feature of stochastic ordering in a multi-kernel GP setting will better capture multiple frequencies superimposed in the original signal.

Secondly, as discussed in Chapter 2, traditional methods as the Fourier transform or the Wavelet transform are affected by the so-called Uncertainty principle and need to compromise in time-frequency resolutions. This stochastic embedding allows for a fully adaptive solution that implements an optimal frequency domain partition and provides a more refined time-frequency resolution.

As discussed in Chapter 1, the Hilbert transform requires the underlying signal to be a narrow-band signal (i.e. an IMF) to compute a meaningful instantaneous frequency. If this is not the case, the obtained values of the IF can often be negative, and a lack of physical meaning will be in place. Wahlberg and Schreier (2010) proposed a method tackling this problem and suggested modelling the stochastic process associated with the instantaneous frequency; however, this is an arduous problem to solve since the IF is highly non-stationary, lies in the complex domain of the real part, i.e. the IMF, and the distribution of its stochastic process is highly intractable and difficult to derive in closed form. The second line of approaches in this field defines the complex analytic extension of the real part by exploiting the setting of kernel methods and Gaussian Processes. These works, provided by Bouboulis and Theodoridis (2010). Ambrogioni and Maris (2019), rely on the use of the complex Gaussian kernel and the general technique of kernel complexification used to obtain a complex-valued kernel from an arbitrary real-valued one. Other alternatives suggesting different approaches relying on the Hilbert transform instead are given by Girolami and Vakman (2002), Le Van Quyen et al. (2001), Turner and Sahani (2011). The second stochastic embedding proposed in this thesis tries to tackle this issue from a different

perspective and aims to solve it in a more straightforward and interpretable way.

Following the above discussion, this Chapter introduces three models that will be used within a speech application scenario to show evidence of their endowment. The first one will represent a benchmark comparison and model the stochastic process characterising the original signal $\tilde{s}(t)$. The second model acts on the IMFs and will propose a multi-kernel combining solution defining the stochastic process of the original signal as the sum of the stochastic processes defined over the IMFs and will capture the temporal modes determined by the IMFs. The last model aims to achieve more representative time basis functions and provides a more sophisticated solution. Instead of imposing a model characterising the IFs stochastic processes or following a kernel complexification approach, the novelty lies in the definition of a stochastic process for the original signal given as the sum of stochastic processes defined over a new set of time basis functions. These new basis functions will be called band-limited IMFs and are obtained according to the location of their instantaneous frequency values within a pre-computed grid of the frequency domain.

This Chapter is organised as follows: firstly, a review of the Gaussian Processes is provided. Secondly, the EMD stochastic representation developed within a Gaussian Process framework is presented. Afterwards, the stochastic embedding models are described. The last section introduces the Generelasid Likelihood Ratio Test exploited in this context and employed for the classification task performed in part III.

6.1 Introduction to Gaussian Processes

The origin of Gaussian processes are partly linked to neural networks (Bishop et al. (1995)), widely used in regression or classification problems which are these days the go to method when seeking to represent a flexible procedure to model a large variety of functions, regardless of the application. However, the flexibility of neural networks is accompanied by a very high computational cost required to identify many parameters determined from the data and can often produce significant overfitting. As a result, the statistical problem of weight regularisation comes into play with the difficulty of selecting the weight regularisation parameters. Such an issue can be tackled through a Bayesian approach, which specifies a hierarchical model with a prior distribution over the hyperparameters of the weights and then provides the prior distribution of the weights relative to the hyperparameters through an observations model. Inducing a posterior distribution over the weights and the hyperparameters is the final step of such a procedure. In the case of neural networks, a prior distribution over the weights of the network induces a prior distribution over functions. This prior over functions has a complex form which is often analytically intractable and whose implementations makes use of approximations (MacKay (1992)), or Monte Carlo approaches to evaluating integrals (Neal (1993)).

Neal (1996) observed that in real-world applications, neural networks should not be limited to nets containing only a small number of hidden units. Indeed, he showed that good predictions could be achieved if a net where the number of units tends to infinity is taken into account along with the Bayesian machinery. Furthermore, he proved that several classes of neural network models become Gaussian processes over functions as the number of hidden units approached infinity. Hence, the equivalence between the Bayesian approach to neural networks and Gaussian processes lies in the fact that in the former case, a prior on the network weights induces a prior over functions, while, in the latter, the alternative is putting Gaussian processes prior over functions. Note that a related idea has been used in spatial statistics under the name "kriging". Gaussian processes as a tool for regression or classification provides a much simpler inference technique with a more straightforward interpretation. Hence, they became the cornerstone of supervised machine learning techniques for inference procedures applied in non-linear regressions or classification problems (Rasmussen and Williams (2005)).

As highlighted in Rasmussen and Williams (2005), there are multiple ways to interpret Gaussian processes, and one can be found more straightforward than the other. Two main perspective can be considered, i.e. the weight-space view or the function-space view. In this work, the former one is the one taken into account. In practice, a Gaussian process is a nonparametric probabilistic inference tool that can be deemed a distribution over functions whose inference takes place within the function space. The properties of likely functions under a GP, e.g. periodicity, smoothness, tractability, non-linearity, robustness to overfitting, scalability, etc. (see Rasmussen and Williams (2005)), are controlled by the positive definite covariance function often referred to as kernel. A kernel function shapes the architecture of these properties by controlling the similarity between pairs of points in the random function domain. Therefore, selecting the most desirable kernel is a hardly delicate challenge primarily discussed in the literature. A review of the employed kernel for our application is given in Chapter 4. Formally, a Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution, which is entirely described by its mean and kernel covariance function. A more rigorous definition is introduced.

Definition 6.1.1 (Gaussian Process (GP))**.** *Denote by $f(x) : \mathcal{X} \to \mathbb{R}$ a stochastic process, parametrised with state-space $\{x\} \in \mathcal{X}$, where $\mathcal{X} \subseteq \mathbb{R}^d$. The random function $f(x)$ is a Gaussian Process if all finite dimensional distributions are Gaussian, where for any $n \in \mathbb{N}$, the random vector $(f(x_1), f(x_2), \ldots, f(x_n))$ is jointly normally distributed. A GP can be therefore interpreted formally as defined by the following class of random functions:*

$$f := \{f(\cdot) : \mathcal{X} \to \mathbb{R} : f(\cdot) \sim \mathcal{GP}\left(\mu(\cdot, \boldsymbol{\theta}_\mu), k(\cdot, \boldsymbol{\theta}_k)\right)\} \tag{6.1}$$

with $\mu(\cdot, \boldsymbol{\theta}_\mu) : \mathcal{X} \to \mathbb{R}$, $k(\cdot, \boldsymbol{\theta}_k) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$,

$$\begin{aligned} \mu(\cdot, \boldsymbol{\theta}_\mu) &= \mathbb{E}\left[f(\cdot)\right] \\ k(\cdot, \boldsymbol{\theta}_k) &= \mathbb{E}\left[\left(f(\cdot) - \mu(\cdot, \boldsymbol{\theta}_\mu)\right)\left(f(\cdot) - \mu(\cdot, \boldsymbol{\theta}_\mu)\right)\right] \end{aligned} \tag{6.2}$$

In most applications, lack of information about the mean function $m(x)$ is a challenge. For simplicity and without loss of generality, since Gaussian processes are a linear combination of Normal distributed random variables by definition, the mean function is commonly assumed to be zero (Bishop et al. (1995)). A more general case in which $m(x)$ is instead modelled according to some data function can be easily considered. The covariance function $k(x, x')$ can be, in general, any function that takes two arguments, i.e. x, x', such that $k(x, x')$ generates a nonnegative $n \times n$ covariance matrix $\mathbf{K}$ over a set of values in the state space $x_i \in \{x_1, \ldots, x_n\}$. The covariance kernel function is characterised by hyperparameters unknown a priori, and a learning procedure is required to identify them. The output of the Gaussian process model is a normal distribution, expressed in terms of the mean and variance. The mean value presents the most likely output, and the variance represents a measure of its confidence.

In this work, Gaussian processes are exploited to define the distribution of the basis functions of the Empirical Mode Decomposition and hence construct the desired stochastic embedding. Particularly, in the speech experiments presented in part III of this thesis, Gaussian processes will be exploited in the context of time-series and, therefore, the scenario of interest will be the one of Gaussian process regression. The following subsection provides a review of this setting, and a brief section on the hyperparameters learning problem is presented. Note that procedure for kernel learning have been already widely discussed in Chapter 4, in section 4.1.

6.1.1 Prediction with Gaussian Processes

In this subsection, a review of the prediction problem, which is often the one encountered in a time-series setting, is presented. Consider a set of observation $\{y_1, y_2, \ldots, y_n\}$ representing the dependent variable subject to noise at certain time instant points $\{t_1, t_2, \ldots, t_N\}$, hence one has $y_i = y(t_i)$ for $i = 1, \ldots, N$. The question of interest in prediction settings is to identify the estimate of the dependent variable at a new time instant $t_\star$. In a Gaussian process regression framework, $\boldsymbol{t} = t_1, t_2, \ldots, t_N$ represents the input vector and the final goal is to predict $y(\boldsymbol{t})$ at the new value input value $t_\star$, i.e. $y(t_\star)$ given the observations set.

Consider the observations being the sum of a function of the input vector $\boldsymbol{t}$ plus an additive Gaussian noise as follows

$$y(t) = f(t) + \epsilon \tag{6.3}$$

Hence, one has a set of observations y_i, $i = 1, \ldots, N$ on which each element is a sample from a Gaussian distribution representing the real value of the observation affected by some independent Gaussian noise ϵ with variance σ_n^2. A Gaussian Process is a set of random variables modelled by a multivariate Gaussian as

$$p(\boldsymbol{f}|\boldsymbol{t}) = \mathcal{N}(\boldsymbol{f}|\boldsymbol{\mu}, \mathbf{K}) \tag{6.4}$$

where $\boldsymbol{f} = f(\boldsymbol{t}) = (f(t_1), \ldots, f(t_N))$, $\boldsymbol{\mu} = \mu(\boldsymbol{t}) = (\mu(t_1), \ldots, \mu(t_N))$ and $\mathbf{K}_{ij} = k(t_i, t_j)$. The mean function is often assumed as $\boldsymbol{\mu} = \mu(\boldsymbol{t}) = 0$. Hence, the

resulting Gaussian process is then a distribution over these function whose shape is defined by $\mathbf{K}$. If two points t_i and t_j are are considered to be similar by the kernel function taken into account, then $f(t_i)$ and $f(t_j)$ can be expected to be similar. At this stage, there are a set of input observations $\boldsymbol{t}$ and one has estimated functions $\boldsymbol{f}$ with these observations. Assume a new instant point, or test input, denoted as $t_\star$ comes into play. The objective is to predict $f_\star$, expected value given $t_\star$. The joint distribution of $\boldsymbol{f}$ and $\boldsymbol{f}_\star$ can be modelled as

$$\begin{bmatrix} \boldsymbol{f} \\ f_\star \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \mathbf{K} & \mathbf{K}_\star \\ \mathbf{K}_\star^\intercal & \mathbf{K}_{\star,\star} \end{bmatrix}\right) \tag{6.5}$$

where $\mathbf{K} = k(\boldsymbol{t}, \boldsymbol{t})$, $\mathbf{K}_\star = k(\boldsymbol{t}, t_\star)$ and $\mathbf{K}_{\star,\star} = k(t_\star, t_\star)$ and $k(\cdot, \cdot)$ is a pre-selected kernel function able to reproduce a covariance function. This is modelling the joint distribution $p(\boldsymbol{f}, f_\star|\boldsymbol{t}, t_\star)$ but the goal is to identify the conditional distribution over $f_\star$ only, which is denote as $p(f_\star|\boldsymbol{f}, \boldsymbol{t}, t_\star)$. The derivation of this quantity uses the Marginal and Conditional distributions of a Multivariate Normal (Tong (2012)) given as follows

Theorm 6.1.2 (Marginals and conditionals of an MVN)**.** *Suppose $X = (\boldsymbol{x}_1, \boldsymbol{x}_2)$ is jointly Gaussian with parameters*

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \quad \boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix}, \quad \boldsymbol{\Lambda} = \boldsymbol{\Sigma}^{-1} = \begin{bmatrix} \boldsymbol{\Lambda}_{11} & \boldsymbol{\Lambda}_{12} \\ \boldsymbol{\Lambda}_{21} & \boldsymbol{\Lambda}_{22} \end{bmatrix} \tag{6.6}$$

Then the marginals are given by

$$\begin{aligned} p(\boldsymbol{x}_1) &= \mathcal{N}\left(\boldsymbol{x}_1|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11}\right) \\ p(\boldsymbol{x}_2) &= \mathcal{N}\left(\boldsymbol{x}_2|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_{22}\right) \end{aligned} \tag{6.7}$$

and the posterior condition is given by

$$\begin{aligned} p(\boldsymbol{x}_1|\boldsymbol{x}_2) &= \mathcal{N}\left(\boldsymbol{x}_1|\boldsymbol{\mu}_{1|2}, \boldsymbol{\Sigma}_{1|2}\right) \\ \boldsymbol{\mu}_{1|2} &= \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\left(\boldsymbol{x}_2 - \boldsymbol{\mu}_2\right) \\ &= \boldsymbol{\mu}_1 - \boldsymbol{\Lambda}_{11}^{-1}\boldsymbol{\Lambda}_{12}\left(\boldsymbol{x}_2 - \boldsymbol{\mu}_2\right) \\ &= \boldsymbol{\Sigma}_{1|2}\left(\boldsymbol{\Lambda}_{11}\boldsymbol{\mu}_1 - \boldsymbol{\Lambda}_{12}\left(\boldsymbol{x}_2 - \boldsymbol{\mu}_2\right)\right) \\ \boldsymbol{\Sigma}_{1|2} &= \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21} = \boldsymbol{\Lambda}_{11}^{-1} \end{aligned} \tag{6.8}$$

By exploiting the above it is then possible to obtain:

$$f_\star|\boldsymbol{f}, \boldsymbol{t}, t_\star \sim \mathcal{N}\left(\mathbf{K}_\star^\intercal\mathbf{K}\boldsymbol{f}, \quad \mathbf{K}_{\star\star} - \mathbf{K}_\star^\intercal\mathbf{K}^{-1}\mathbf{K}_\star\right) \tag{6.9}$$

However, given the assumed model in (6.3) and assuming additive independent identically distributed Gaussian noise with variance σ_n^2, the prior on the noisy observations becomes

$$\mathbb{C}\text{ov}(y) = \mathbf{K} + \sigma_n^2\mathbb{I} \tag{6.10}$$

Therefore, the prediction step corresponds to estimate the mean value and the variance for $f_\star$. Considering equation 6.5, then the joint distribution of the

observed target values and the function value at the test location under the prior is (Rasmussen and Williams (2005))

$$\begin{bmatrix} \boldsymbol{y} \\ f_\star \end{bmatrix} \sim \left(0, \begin{bmatrix} \mathbf{K} + \sigma_n^2 \mathbb{I} & \mathbf{K}_\star \\ \mathbf{K}_\star^\intercal & \mathbf{K}_{\star\star} \end{bmatrix}\right) \tag{6.11}$$

It becomes therefore obvious that the desired quantity is the conditional distribution of $f_\star$ given $\boldsymbol{y}$. By deriving the conditional distribution as done in equation (6.9) it is then possible to obtain the predictive equations for Gaussian process regression as

$$\bar{f}_\star | \boldsymbol{t}, \boldsymbol{y}, t_\star \sim \mathcal{N}\left(\bar{f}_\star, \mathbb{C}\mathrm{ov}(f_\star)\right) \tag{6.12}$$

where

$$\begin{aligned} \bar{f}_\star &:= \mathbb{E}\left[\bar{f}_\star | \boldsymbol{t}, \boldsymbol{y}, \boldsymbol{t}_\star\right] = \mathbf{K}_\star^\intercal \left[\mathbf{K} + \sigma_y^2 \mathbb{I}\right]^{-1} \boldsymbol{y} \\ \mathbb{C}\mathrm{ov}(f_\star) &= \mathbf{K}_{\star\star} - \mathbf{K}_\star^\intercal \left[\mathbf{K} + \sigma_y^2 \mathbb{I}\right] \mathbf{K}_\star \end{aligned} \tag{6.13}$$

Note that equation (6.12) might also be referred to as the distribution of $y_\star | \boldsymbol{y}$ given the considered model in equation (6.3). The mean value $\bar{f}_\star$ is also known as the matrix of regression coefficients and provides the best estimate for $y_\star$. The variance $\mathbb{C}\mathrm{ov}(f_\star)$ is also known as the Schur complement and provides a measure of uncertainty regarding the computed estimation. It is important to highlight that the mean function $\bar{f}_\star$ is a linear combination of the observations $\boldsymbol{y}$ and that the variance $\mathbb{C}\mathrm{ov}(f_\star)$ does not depend on the observations $\boldsymbol{y}$ but only on the input $\boldsymbol{t}$.

A quantity that must be introduced since then studied in the following paragraph for the hyperparameter learning is the marginal likelihood denoted as $p(\boldsymbol{y}|\boldsymbol{t})$. The marginalisation can be interpreted as an integral over the function values $\boldsymbol{f}$. The marginal likelihood is hence the integral of the likelihood times the prior given as

$$p(\boldsymbol{y}|\boldsymbol{t}) = \int p(\boldsymbol{y}|\boldsymbol{f}, \boldsymbol{t}) p(\boldsymbol{f}|\boldsymbol{t}) d\boldsymbol{f} \tag{6.14}$$

Under the Gaussian process model, the prior is Gaussian, i.e. $\boldsymbol{f}|\boldsymbol{t} \sim \mathcal{N}(0, \mathbf{K})$ and the likelihood is also Gaussian, $\boldsymbol{y}|\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{f}, \sigma_n^2 \mathbb{I})$. Using the logarithmic to simplify the calculation, it is common practice to consider the log marginal likelihood given as

$$\log p(\boldsymbol{y}|\boldsymbol{t}) = -\frac{1}{2} \boldsymbol{y}^\intercal \boldsymbol{C}^{-1} \boldsymbol{y} - \frac{1}{2} \log |\boldsymbol{C}| - \frac{n}{2} \log 2\pi \tag{6.15}$$

This inference is exact and can be derived in closed form since both the prior and the posterior are Gaussian, otherwise the shape of the likelihood would be analytically intractable. Each term of the log-likelihood provide a specific information for the model. The first one, which involves the observations $\boldsymbol{y}$ corresponds to the data-fit term. The second depends only on the covariance matrix $\boldsymbol{C}$ and works as the regularisation term in standard linear regression, adding a penalty as the complexity of the data increases. The third term corresponds to

a normalising constant. Williams (1998) offers a careful analysis of the effects of the hyperparameters in the log-marginal likelihood. There are multiple computational aspects that need to be considered when this quantity is optimised to derive the optimal set of hyperparameters. For example, the inversion of the covariance matrix is not always possible or if it is ill-conditioned then the inversion cannot be achieved or it is not trivial. Details about this issues can be found in Williams (1998), Gibbs and MacKay (1997) and Huhle et al. (2010). A brief review of these concept is also provided in this tutorial Melo (2012).

6.2 The EMD Stochastic Representation by Gaussian Processes

As presented in Chapter 3, within section 3.1, a signal $s(t)$ for $t \in [0, \infty]$ is a continuous true signal which is observed on discrete grid of points in the interval $[0, T]$, $t = (t_1 < \cdots < t_N) = \{t_i\}_{i=1:N}$, where the subscripts represent the sampling index times. The observed values of the true signal $s(t)$ might be exact or be perturbed. Note that noisy observations are not uncommon situation in real-world application. The perturbation of the true signal can be either deterministic (i.e. given by a chaotic system, not stable) or stochastic. If the realisations of the signal $s(t)$ are corrupted with some stochastic error term, the observed process is represented as follows

$$y(t) = s(t) + \epsilon, \text{ with } \epsilon \sim \mathcal{N}(0, \sigma^2). \tag{6.16}$$

Therefore, the observation set consists of pairs $\left\{t_n, y_n\right\}$ where $y_i = y(t_i)$ for $t_i \in [0, T]$. The first step is to find the EMD decomposition of the signal $s(t)$. Remark that, for the EMD to exists, the input signal needs to be approximated by a continuous representation. Hence, the discrete signal $s(t)$ is converted back into a continuous analog signal as presented in section 3.1 of Chapter 3 and denoted as $\tilde{s}(t)$ given in Equation (3.1). Afterwards, the EMD is applied and the set of basis functions $\gamma_l(t)$ with $l = 1, \ldots, L$ along with the residual $r(t)$ is obtained as in Equation (3.3) given in Chapter 3. The final goal is to obtain a stochastic representation of the continuous signal $\tilde{s}(t)$ by exploiting the Gaussian Process framework along with the Empirical Mode Decomposition. In the following subsections, the process assumption for the IMFs are firstly taken into account and then the one for the residual is provided. Afterwards, the multi-kernel formulation for a stochastic embedding of the EMD is presented.

6.2.1 The IMFs as Gaussian Processes

To achieve the desired embedding, consider first the stochastic process associated with the deterministic path $s(t)$ of the continuous signal $\tilde{s}(t)$ and denote that as $S(t)$. When the EMD is applied to the approximating signal $\tilde{s}(t)$, and the set of basis functions are extracted, each IMF $\gamma_l(t)$ can be considered as the realised path of the stochastic process denoted as $\Gamma_l(t)$ and the one for the residual $r(t)$

denoted as $R(t)$. The obtained reconstructed stochastic process will then be the one of the continuous signal $\tilde{s}(t)$ denoted as $\widetilde{S}(t)$. The two processes, $S(t)$ and $\widetilde{S}(t)$, are equal in the interval $[0, T]$ at the knot points of the interpolated continuous signal $\tilde{s}(t)$; however, at all the the other points, this will not be the case and, therefore, this will result into a residual error $\epsilon(t)$, that could be interpreted as a regression error, and that is given as

$$S(t) \stackrel{d}{=} \widetilde{S}(t) + \epsilon(t) \tag{6.17}$$

At an observed process level, this corresponds to $s(t) = \tilde{s}(t) + e(t)$, where $e(t)$ represents the observed error at $t \in [0, T]$ and t is not a knot point. This strictly relates to the interpolation representation selected; in the case of a penalised spline, for example, this will not be the case and an observed error $e(t)$ will be present at every $t \in [0, T]$. Note that the further assumption that is considered in the above Equation is the equality in distribution, with

$$\widetilde{S}(t) \stackrel{d}{=} \sum_{l=1}^{L} \Gamma_l(t) + R(t) \tag{6.18}$$

where $\Gamma_l(t)$ represents the GP for IMF l and there are $l = 1, \ldots, L$ of them. $R(t)$ represents instead the GP on the residual tendency component. Hence, in the remaining sections, the stochastic process will be constructed for the stochastic process of the approximated signal given as $\widetilde{S}(t)$. If $\widetilde{S}(t)$ was a stationary process and comprised of L underlying stationary processes, then its distribution could be given as a Gaussian Process defined as

$$\tilde{S}(t) \sim \mathcal{GP}\left(\mu(t; \boldsymbol{\theta}_\mu); k(t, t'; \boldsymbol{\theta}_k)\right) \tag{6.19}$$

where $\mu(t; \boldsymbol{\theta}_\mu)$ and $k(t, t'; \boldsymbol{\theta}_k)$ represent the mean and kernel functions respectively and are stationary over time. In this setting, $\widetilde{S}(t)$ could be decomposed into a set of L stochastic processes characterising the L harmonics of the observed $\tilde{s}(t)$. Hence, the mean and the kernel functions will be given as $\mu(t; \boldsymbol{\theta}_\mu) = \sum_{l=1}^{L} \mu(t; \boldsymbol{\theta}_{\mu_l})$ and $k(t, t'; \boldsymbol{\theta}_k) = \sum_{l=1}^{L} k_l(t, t'; \boldsymbol{\theta}_l)$. However, the assumption made in this work is that $\widetilde{S}(t)$ is highly non-stationary, and,can be decomposed into the sum of a finite number L non-stationary basis functions which are the IMFs. This is equivalent to say that $\widetilde{S}(t)$ has a formulation given as

$$\tilde{S}(t) \sim \mathcal{GP}\left(\mu(t; \boldsymbol{\theta}_\mu(t)); k(t, t'; \boldsymbol{\theta}_k(t))\right) \tag{6.20}$$

where $\mu(t; \boldsymbol{\theta}_\mu(t))$ and $k(t, t'; \boldsymbol{\theta}_k(t))$ are both non-stationary hence depending on t. The structure of these time-varying functions is unknown a priori, and the approach of this thesis is to model them as the sum of the individual L equivalent static functions (i.e. the mean and the covariance ones) of the IMFs obtained by decomposing the original observed approximated signal. At this point, the intuition for the requirement of the multi-kernel representation comes into play and shows its relevance. The novel contribution of this work will be indeed modelling $\widetilde{S}(t)$ as the sum of multiple stochastic processes whose structure can

be derived with the EMD rather than trying to model the time-varying mean and kernel functions through a complex and difficult o interpret multi-kernel representation. Since equality in distribution is assumed in Equation (6.18), then the next assumption required to define the desired stochastic embedding is that each process $\Gamma_l(t)$ will also be represented as a Gaussian process such that,

$$\Gamma_l(t) \sim \mathcal{GP}\Big(\mu(t, \boldsymbol{\theta}_{\mu_l}); k_l(t, t'; \boldsymbol{\theta}_l)\Big), \tag{6.21}$$

where $\mu(t, \boldsymbol{\theta}_{\mu_l})$ represents the mean function parametrised by $\boldsymbol{\theta}_{\mu_l}$ and $k_l(t, t'; \boldsymbol{\theta}_l)$ is a positive definite covariance kernel which is parametrized by a set of parameters $\boldsymbol{\theta}_l$. In the following paragraph, the prediction distribution of the stochastic process $\Gamma_l(t)$ distributed as a Gaussian Process given in Equation (6.21) is provided. Then, the assumption for the distribution of the residual tendency stochastic process $R(t)$ is derived.

Prediction with IMFs as Gaussian Processes

Assume that both the processes $\widetilde{S}(t)$ and $\Gamma_l(t)$ for $l = 1, \ldots, L$ are observed at the N time points $t_1 < \ldots < t_N$. Denote by $\mathbf{t}$ the vector of points t_n for $n = 1, \ldots, N$. In a Gaussian process regression setting, given the observations $\gamma_l(\mathbf{t}) = \Big[\gamma_l(t_1), \ldots, \gamma_l(t_N)\Big]$, the goal is to predict the values of $\gamma_l(t)$ at the new input argument u, i.e. $\gamma_l(u)$, given the collected information in the observation set. Since $\Gamma_l(t)$ is a Gaussian Process, the random variable $\Gamma_l(u)|\Gamma_l(\mathbf{t})$ is a Gaussian Process with the conditional mean

$$\mu_l(u) := \mathbb{E}_{\gamma_l(t)|\gamma_l(\mathbf{t})}\Big[\gamma_l(u)\Big] = \mathbf{k}_l\Big(u, \mathbf{t}\Big)\mathbf{K}_l\Big(\mathbf{t}, \mathbf{t}\Big)^{-1}\gamma_l(\mathbf{t})$$

and the conditional covariance matrix given by

$$\begin{aligned}\tilde{k}_l(u, u') &:= \mathbb{E}_{\gamma_l(t)|\gamma_l(\mathbf{t})}\Big[(\gamma_l(u) - \mu_l(u))(\gamma_l(u') - \mu_l(u'))\Big] \\ &= k_l\Big(u, u'\Big) - \mathbf{k}_l\Big(u, \mathbf{t}\Big)\mathbf{K}_l\Big(\mathbf{t}, \mathbf{t}\Big)^{-1}\mathbf{k}_l\Big(\mathbf{t}, u'\Big)^T\end{aligned}$$

where

$$\mathbf{K}_l(\mathbf{t}, \mathbf{t}) := \begin{bmatrix} k_l(t_1, t_1) & k_l(t_1, t_2) & \cdots & k_l(t_1, t_N) \\ k_l(t_2, t_1) & k_l(t_2, t_2) & \cdots & k_l(t_2, t_N) \\ \vdots & \vdots & \ddots & \vdots \\ k_l(t_N^{(i)}, t_1) & k_l(t_N, t_2) & \cdots & k_l(t_N, t_N) \end{bmatrix}_{N \times N}$$

and

$$\mathbf{k}_l(u, \mathbf{t}) := \begin{bmatrix} k_l(u, t_1) & k_l(u, t_2) & \cdots & k_l(u, t_N) \end{bmatrix}_{1 \times N}.$$

In practice, it can be advantageous to regularise the Gram matrix for a GP in order to improve the numerical properties of the inverse of this matrix, often encountered when working with such a model. To do so, the mean function and

kernel of the conditional distribution would be adjusted by a ridge regularisation as follows:

$$\mu_l(s) := \mathbb{E}_{\gamma_l(t)|\gamma_l(\mathbf{t}),\mathbf{t}}\Big[\gamma_l(u)\Big] = \mathbf{k}_l\Big(u,\mathbf{t}\Big)\Big(\mathbf{K}_l\Big(\mathbf{t},\mathbf{t}\Big) + \sigma_k^2\Big)^{-1}\gamma_l(\mathbf{t})$$

and the conditional covariance matrix given by

$$\begin{aligned}\tilde{k}_l(u,u') &:= \mathbb{E}_{\gamma_l(t)|\gamma_l(\mathbf{t}),\mathbf{t}}\Big[(\gamma_l(u)-\mu_l(u))(\gamma_l(u')-\mu_l(u'))\Big] \\ &= k_l\Big(u,u'\Big) - \mathbf{k}_l\Big(u,\mathbf{t}\Big)\Big(\mathbf{K}_l\Big(\mathbf{t},\mathbf{t}\Big) + \sigma_k^2\Big)^{-1}\mathbf{k}_l\Big(\mathbf{t},u'\Big)^T\end{aligned}$$

that is equivalent to the accounting for the artificial noise component in the model in Equation (6.21). The following subsection provides the stochastic process assumption made for the residual $r(t)$.

6.2.2 The Assumption for the Residual Tendency $r(t)$

The last component extracted by the EMD corresponds to the residual or tendency component $r(t)$. By definition, this last component has only one convexity within the domain $[0,T]$. A Gaussian Process would not achieve such a restriction and, therefore, two solutions could be considered. The first considers a monotonic Gaussian Process, as suggested in Lin and Dunson (2014), which represents an isotonic Gaussian Process. Such representation is, in practice, hard to construct since a shape constrained estimation procedure is required, and it will be computationally involving to construct.

Alternatively, to reduce the computational cost associated with the construction of the EMD, it is common practice to apply it on non-overlapping windows of the approximated signal $\tilde{s}(t)$. Then, the obtained segmented IMFs will be concatenated back to the original dimension and used to perform the task of interest. Once concatenated back into its original shape, the residual $r(t)$ will no longer carry a unique convexity change since multiple local tendencies would be joined into the same vector. Therefore, the Gaussian Process framework developed so far will apply again. Then, as for the IMF case, given the equality in distribution assumed in Equation (6.18), one can model the stochastic process of $r(t)$ as a Gaussian Process itself, providing

$$R(t) \sim \ \mathcal{GP}\Big(\mu(t;\boldsymbol{\theta}_{\mu_{l+1}}); k_{l+1}(t,t';\boldsymbol{\theta}_{l+1})\Big) \tag{6.22}$$

where $\boldsymbol{\theta}_{\mu_{l+1}}$ and $\boldsymbol{\theta}_{l+1}$ represent the set of hyperparameters of the mean function $\mu(t;\boldsymbol{\theta}_{\mu_{l+1}})$ and the kernel function $k_{l+1}(t,t';\boldsymbol{\theta}_{l+1})$ respectively.

Once the assumption on the stochastic process distribution of the residual is made, the following step correspond to the assumption on the distribution of the stochastic process of the reconstructed signal, i.e. $\widetilde{S}(t)$. To achieve that, the multi-kernel representation for the EMD is firstly introduced.

6.2.3 A Multi-Kernel Representation for the EMD

The goal of this section is to introduce the framework for a novel multi-kernel representation of the non-stationary stochastic process $\widetilde{S}(t)$ given in Equation (6.20). As before presented, multiple approaches could be considered covering this task (see Chapter 4 for further references). Regardless of the selected method, this could be, in practice, challenging to achieve since the level of non-stationarity affecting the approximated signal $\tilde{s}(t)$ is unknown a priori and identifying both a suitable kernel function representation and a set of hyperparameters capturing such a feature is a difficult task. The solution of this Chapter relies on the EMD basis functions capturing non-stationarity of the underlying signal and aims to formulate a stochastic embedding able to reproduce a multi-kernel representation which is more reliable than existing machine learning multi-kernel methods solving such a task. The derived stochastic process representation for $\widetilde{S}(t)$ could carry a distribution that is either unconditional or conditional. The statements required for these definitions are presented in these sections and show different statistical perspectives that could be considered in this setting.

So far, a model for every component required to develop the EMD stochastic emebedding have been proposed. Each IMFs and the residual will be a Gaussian Process as given in Equations (6.19), (6.21) and (6.22). To model the time-varying mean and kernel functions of the non-stationary stochastic process $\widetilde{S}(t)$ given in Equation (6.20), the formulation of an additive GP multi-kernel representation is given as follows:

$$\widetilde{S}(t) \sim \ \mathcal{GP}\left(\sum_{l=1}^{L+1} \mu_l(t, \boldsymbol{\theta}_{\mu_l}); \sum_{l=1}^{L+1} k_l(t, t'; \boldsymbol{\theta}_{k_l}) \right) \tag{6.23}$$

where the mean and the kernel functions will be given as the sum of the L mean and kernel functions of the stochastic processes modelling the IMFs and the residual tendency. Multiple settings could be considered depending on the distributional assumptions made on the stochastic processes of $\Gamma_l(t)$ for $l = 1, \dots L$ and $R(t)$ producing an unconditional or a conditional distributions for $\widetilde{S}(t)$. These settings are below presented.

The Unconditional Distribution of $\widetilde{S}(t)$

The most straightforward case considers a multi-kernel representation of $\widetilde{S}(t)$ with uncodintional distribution since both IMFs and the residual stochastic processes will be modelled individually and is given as

$$\widetilde{S}(t) \sim \ \mathcal{GP}\left(\sum_{l=1}^{L+1} \mu(t; \boldsymbol{\theta}_{\mu_l}); \sum_{l=1}^{L+1} k_l(t, t'; \boldsymbol{\theta}_l) \right) \tag{6.24}$$

L is the number of extracted IMFs and, therefore, the stochastic process of $\widetilde{S}(t)$ corresponds to a stochastic process that has mean equal to the sum of the means of the IMFs and the residual stochastic processes and an additive structure for

the kernel corresponding to the sum of the individual kernels of the IMFs and residual stochastic processes. The models for the stochastic process of the IMFs and the residual given as

$$\begin{aligned} \Gamma_l(t) &\sim \ \mathcal{GP}\Big(\mu(t,\boldsymbol{\theta}_{\mu_l}); k_l(t,t';\boldsymbol{\theta}_l)\Big) \ \text{ for } \ l=1,\ldots,L \\ R(t) &\sim \ \mathcal{GP}\Big(\mu(t;\boldsymbol{\theta}_{\mu_{l+1}}); k_{l+1}(t,t';\boldsymbol{\theta}_{l+1})\Big) \end{aligned} \tag{6.25}$$

In practice, in this work, centered Gaussian processes will always be taken into account, and, therefore, the proposed model for $\widetilde{S}(t)$ will consider an unconditional stochastic representation which will have zero mean as

$$\widetilde{S}(t) \sim \ \mathcal{GP}\Big(0; \sum_{l=1}^{L+1} k_l(t,t';\boldsymbol{\theta}_l)\Big) \tag{6.26}$$

The same reasoning applies to the unconditional distributions of the stochastic processes $\Gamma_l(t)$, for $l=1,\ldots,L-1$ and $R(t)$, given as

$$\begin{aligned} \Gamma_l(t) &\sim \ \mathcal{GP}\Big(0; k_l(t,t';\boldsymbol{\theta}_l)\Big) \ \text{ for } \ l=1,\ldots,L \\ R(t) &\sim \ \mathcal{GP}\Big(0; k_{l+1}(t,t';\boldsymbol{\theta}_{l+1})\Big) \end{aligned} \tag{6.27}$$

Furthermore, the stochastic processes between the IMFs and the residual will be considered independent, and no correlation structure amongst them is studied. This is the most statistical flexible solution and the one developed in the construction of the proposed stochastic embedding models below presented, while, the following two paragraphs presents alternative constructions for a multi-kernel stochastic EMD.

The First Conditional Distribution of $\widetilde{S}(t)$

In these two paragraphs, the assumption of centered Gaussian Processes above presented will be considered for the formulated results. Hence, the unconditional distributions will be the one with zero means introduced in Equations (6.26) and (6.27). The second solution that could be considered in these settings proposes a conditional distribution for $\widetilde{S}(t)$ and reviews Equation (6.24) as follows

$$\widetilde{S}(t)|R(t)=r(t) \sim \ \mathcal{GP}\Big(r(t); \sum_{l=1}^{L} k_l(t,t';\boldsymbol{\theta}_l)\Big) \tag{6.28}$$

where this time a conditional distribution for $\widetilde{S}(t)$ is assumed, conditioned on the stochastic process of the tendency $R(t)$. The observed $r(t)$ is incorporated into the mean of this newly constructed multi-kernel representation. Specifically, the conditional distribution comes from the fact that the stochastic process of the

last IMF denoted as $\Gamma_L(t)$ is the one conditioned on $R(t)$. Hence, the multi-kernel formulation will consider the following distributions for the IMFs

$$\begin{aligned} \Gamma_l(t) &\sim \mathcal{GP}\Big(0; k_l(t,t';\boldsymbol{\theta}_l)\Big) \text{ for } l=1,\ldots,L-1 \\ \Gamma_L(t)|R(t) &\sim \mathcal{GP}\Big(r(t); k_L(t,t';\boldsymbol{\theta}_L)\Big) \end{aligned} \tag{6.29}$$

Hence, the distribution of the stochastic process of the last IMF is the one that is conditioned on the stochastic process of the residual and this will be reflected within the multi-kernel EMD as given in Equation (6.30).

The Second Conditional Distribution of $\widetilde{S}(t)$

The second conditional distribution that could be considered in formulating a multi-kernel representation of the stochastic EMD focus on individual stochastic processes of the IMFs and condition them all (individually) on the stochastic process of the residual $R(t)$. Hence, when aggregated to form the representation of the original approximated signal stochastic process this will be given as follows

$$\widetilde{S}(t)|R(t)=r(t) \sim \mathcal{GP}\Big(\frac{1}{L}\sum_{l=1}^{L} r(t); \sum_{l=1}^{L} k_l(t,t';\boldsymbol{\theta}_l))\Big) \tag{6.30}$$

where the mean function is given as the sum of $r(t)$ an L number of times given that each IMF stochastic process $\Gamma_l(t)$ will be conditioned on $R(t)$ and therefore will have mean equal to $r(t)$. This formulation will then consider the following distributions for the IMFs

$$\Gamma_l(t)|R(t)=r(t) \sim \mathcal{GP}\Big(r(t); k_l(t,t';\boldsymbol{\theta}_l)\Big) \text{ for } l=1,\ldots,L \tag{6.31}$$

The different formulations for a multi-kernel stochastic EMD representation have been presented. These are not the only solutions, and other alternatives could be considered. The first solution considering an unconditional distribution of $\tilde{S}(t)$ will be the one adopted in the proposed models in the section below. Note that the stochastic embeddings have been presented for the first type of stochastic embedding developed, which models the IMFs directly. The second embedding will construct an alternative model defining a new set of quasi-IMFs based on the location of their instantaneous frequencies. The above formulations still apply. The proposed stochastic model embeddings studied in part III are now formally introduced and presented.

6.3 Construction of Stochastic Embedding for the EMD

In this section, the three core models that act as different stochastic embeddings of the signal $s(t)$ characterised by approximation $\widetilde{s}(t)$ used to extract the EMD basis decomposition. The first model embeds the approximated signal $\tilde{s}(t)$ and works as a reference guide or benchmark model since the most straightforward proposal within this context. In this regard, the advantage of modelling each spectral component characterising the signal rather than the signal itself is the sought objective since time-variation features are phenomena often heavily pronounced in real-data applications. Furthermore, it could be possible to apply a multi-kernel approach directly on $\tilde{s}(t)$ as the great deal of methods reviewed in the introduction of this Chapter and the one reviewed in Chapter 4 (see section 4.4). However, these methods often require stationarity of the underlying data system and this is not the case in various applications in practice and would result in poor performances.

To tackle these issues, the second model proposes the embedding of the IMFs basis functions $\gamma_l(t)$ with $l = 1, \ldots, L$. Each component is characterised by a specific number of convexity changes that detect a temporal mode of the original signal. This embedding aims to capture this concept to model the underlying basis through an ad hoc kernel function and then reproduce the kernel of the original signal as the sum of the ones individually modelled. Remark that the stochastic ordering captured by the IMFs will be preserved through the additive kernel structure. Furthermore, the relevance of the GP as the stochastic process will provide a representation of the IMFs through a smooth function.

The third stochastic embedding propose a more involved solution and foresees the definition of new quasi-IMFs denoted as $\gamma_m^{(BL)}$, where the upper script stays for "band-limited" IMFs. This is because the model aims to capture specific frequency bandwidths information. These new bases are obtained by aggregating existing IMFs whose instantaneous frequencies lie in the same frequency bandwidths. Such newly engineered bases are then summed up together to obtain $\tilde{s}(t)$. This embedding aims to refine the solution proposed with the second model, characterise IMFs with instantaneous frequency functions belonging to the same frequency bands, and, therefore, model them according to a unique kernel function. The accomplishment of such construction requires the estimation of an optimal grid, or optimal partition, of the instantaneous frequency domain in the sense that highly populated bandwidths should be defined. Such a partition will be identified through the cross-entropy method presented in Chapter 7 and denoted as $\hat{\Pi}$. For this Chapter, it is important to comprehend that an optimal partition will be found, and through that, the model will be derived. Then the following Chapter will present the methodology employed. Note that this system model is conditional on estimating this partition and differs in this sense from the first two above proposed.

6.3.1 System Model 1: Gaussian Process on $\tilde{s}(t)$

The first model assumes that the original observed signal $\tilde{s}(t)$ is obtained from a stochastic process $\tilde{S}(t)$ which is a GP given as follows:

$$\tilde{S}(t) \sim \mathcal{GP}\left(\mu(t;\boldsymbol{\theta}_\mu); k(t,t';\boldsymbol{\theta}_k)\right) \tag{6.32}$$

where $\mu(t;\boldsymbol{\theta}_\mu)$ and $k(t,t';\boldsymbol{\theta}_k)$ represent the mean and kernel functions respectively, $\boldsymbol{\theta}_\mu$ and $\boldsymbol{\theta}_k$ are the sets of hyperparameters of the mean and the kernel respectively. Note that the zero mean assumption is considered and, therefore, one has $\mu(t;\boldsymbol{\theta}_\mu) = 0$.

6.3.2 System Model 2: Time Domain Stochastic Embedding of the IMFs

Consider $\tilde{s}(t)$ decomposed into IMFs basis functions denoted as $\{\gamma_l(t)\}_{l=1}^L$ (introduced in section 3.2). The second model assumes that each $\gamma_l(t)$ is observed from a Gaussian Process $\Gamma_l(t)$ and then reconstructs the original $\tilde{s}(t)$ by summing up the IMFs. Each process $\Gamma_i(t)$ is assumed to be independent of all the other GPs $\Gamma_j(t)$ for $i \neq j$. A diagram of the model is presented below:

$$\tilde{s}(t) \begin{cases} \gamma_1(t) & \Gamma_1(t) \sim \mathcal{GP}(\mu_1(t;\boldsymbol{\theta}_{\mu_1}); k_1(t,t';\boldsymbol{\theta}_1)) \\ \dots & \\ \gamma_L(t) & \Gamma_L(t) \sim \mathcal{GP}(\mu_L(t;\boldsymbol{\theta}_{\mu_L}); k_L(t,t';\boldsymbol{\theta}_L)) \end{cases}$$

Given the above construction, the original GP $\widetilde{S}(t)$ can be expressed as a sum of GPs, one for each observed IMF $\gamma_l(t)$. The tool to achieve such a reconstruction is a multi-kernel representation of $\widetilde{S}(t)$ given by

$$\widetilde{S}(t) \overset{d}{=} \sum_{l=1}^{L} \Gamma_l(t) \sim \mathcal{GP}\left(\mu(t;\boldsymbol{\theta}_\mu), k(t,t';\boldsymbol{\theta}_k)\right), \tag{6.33}$$

where $\mu(t;\boldsymbol{\theta}_\mu) = \sum_{l=1}^L \mu_l(t;\boldsymbol{\theta}_{\mu_l})$ and $k(t,t';\boldsymbol{\theta}_k) = \sum_{l=1}^L k_l(t,t';\boldsymbol{\theta}_l)$.

6.3.3 System Model 3: Frequency Domain Stochastic Embedding via Band-limited Mixture IMF-IF Model

Consider the extracted instantaneous frequencies $\{\omega_l\}_{l=1}^L$. Ideally, these functions are ordered, in a decreasing fashion, according to the oscillation index of their IMFs, i.e. $\omega_1(t) > \omega_2(t) > \cdots > \omega_L(t)$. The reason to develop such a model is that, in practice, one may wish to have a stochastic representation of an EMD signal decomposition that is guaranteed to be characteristic of a particular frequency band. This third system model is formulated based on the idea of aggregating the IMFs samples whose IFs lie within the same frequency band.

Such IMFs are then modelled according to the same GP. To define the model, one needs first to introduce a partition rule which identifies different local frequency bandwidths. Such a rule is introduced in Chapter 7 and exploits the Cross-Entropy method. System Model 3 is now formally introduced. The diagram provides the main idea behind it.

$$\begin{array}{ccccccc} & \gamma_1(t) & \to & \omega_1(t) & & \Gamma_1^{(BL)}(t)|\Pi=\hat{\Pi} \sim \mathcal{GP}(\mu_1^{BL}(t;\boldsymbol{\theta}_{\mu_1^{BL}}), k_1^{BL}(t,t';\boldsymbol{\theta}_{k_1^{BL}})) \\ \nearrow & & & & & & \\ \tilde{s}(t) & \dots & & \longrightarrow & \hat{\Pi} \longrightarrow & \dots & \dots \\ \searrow & & & & & & \\ & \gamma_L(t) & \to & \omega_L(t) & & \Gamma_M^{(BL)}(t)|\Pi=\hat{\Pi} \sim \mathcal{GP}(\mu_M^{BL}(t;\boldsymbol{\theta}_{\mu_M^{BL}}), k_M^{BL}(t,t';\boldsymbol{\theta}_{k_M^{BL}})) \end{array}$$

System Model 3 can be constructed once the partition $\hat{\Pi}$ is obtained. The model is given by the IF being in a certain interval frequency band. The partition $\hat{\Pi}$ is estimated by a realisation of the stochastic process $\widetilde{S}(t)$ and, therefore, is conditioned upon it. This is stated in the distributional assumption of the different processes $\Gamma_m^{(BL)}$ which are indeed given as $\Gamma_m^{(BL)}|\Pi=\hat{\Pi}$.

Formally, by considering $\{\gamma_l(t)\}_{l=1}^L$, $\{\omega_l(t)\}_{l=1}^L$ and the partition Π, we will obtain the following set of aggregated IMFs:

$$\begin{cases} \gamma_1^{(BL)}(t) = \gamma_1(t)\mathbb{1}_{\left\{\omega_1(t)\in\bigcup_{d=1}^D \Pi_{1,d}\right\}} + \ldots + \gamma_K(t)\mathbb{1}_{\left\{\omega_L(t)\in\bigcup_{d=1}^D \Pi_{1,d}\right\}} \\ \gamma_2^{(BL)}(t) = \gamma_1(t)\mathbb{1}_{\left\{\omega_1(t)\in\bigcup_{d=1}^D \Pi_{2,d}\right\}} + \ldots + \gamma_K(t)\mathbb{1}_{\left\{\omega_L(t)\in\bigcup_{d=1}^D \Pi_{2,d}\right\}} \\ \vdots \\ \gamma_M^{(BL)}(t) = \gamma_1(t)\mathbb{1}_{\left\{\omega_1(t)\in\bigcup_{d=1}^D \Pi_{M,d}\right\}} + \ldots + \gamma_K(t)\mathbb{1}_{\left\{\omega_L(t)\in\bigcup_{d=1}^D \Pi_{M,d}\right\}} \end{cases}$$

which results in construction of $\gamma_m^{(BL)}(t)$ for $m=1,\ldots,M$ such that

$$\forall_m\ \gamma_m^{(BL)}(t) = \sum_{l=1}^{L} \gamma_l(t)\mathbb{1}_{\left\{\omega_l(t)\in\bigcup_{d=1}^D \Pi_{m,d}\right\}} \tag{6.34}$$

The idea is to reconstruct the original signal $\tilde{s}(t)$ through the above model so that no information is dispersed or lost since it groups the original IMFs in an alternative way. Hence, it is equivalent to the original IMFs decomposition. Each $\gamma_m^{(BL)}(t)$ will be embedded within a Gaussian Process, Γ_m^{BL}. Given the concept of stochastic ordering, we will firstly model $\Gamma_m^{(BL)}(t)$ according to the same kernel family. Therefore, the original signal will then correspond to

$$\tilde{s}(t) = \sum_{m=1}^{M-1} \gamma_m^{(BL)}(t) = \sum_{l=1}^{L} \gamma_l(t) \tag{6.35}$$

and the stochastic process $\widetilde{S}(t)$ is represented via multi-kernel representation exploiting $\Gamma_m^{BL}(t)$, that is

$$\widetilde{S}(t) \stackrel{d}{=} \sum_{m=1}^{M} \Gamma_m^{BL}(t) \sim \mathcal{GP}\left(\mu_s(t;\boldsymbol{\theta}_{\mu_s}), k_s(t,t';\boldsymbol{\theta}_{k_s})\right), \tag{6.36}$$

where $\mu_s(t;\boldsymbol{\theta}_{\mu_s}) = \sum_{m=1}^{M} \mu_m^{BL}(t)$ and $k_s(t,t';\boldsymbol{\theta}_{k_s}) = \sum_{m=1}^{M} k_m^{BL}(t,t';\boldsymbol{\theta}_{k_M^{BL}})$.

The following figure compares the original IMFs extracted on the speech signal shown in Figure 6.2 and the obtained IMFs Band Limited.

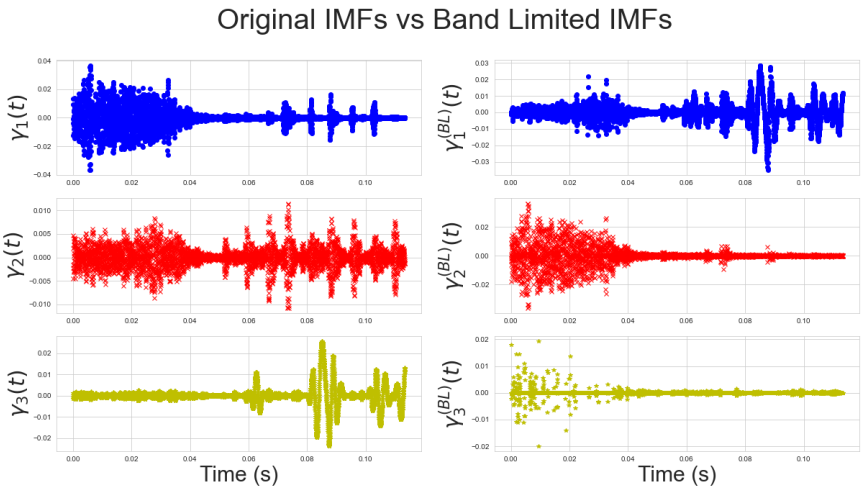


Figure 6.1: Comparison of the original extracted IMFs and the obtained Band Limited IMFs..

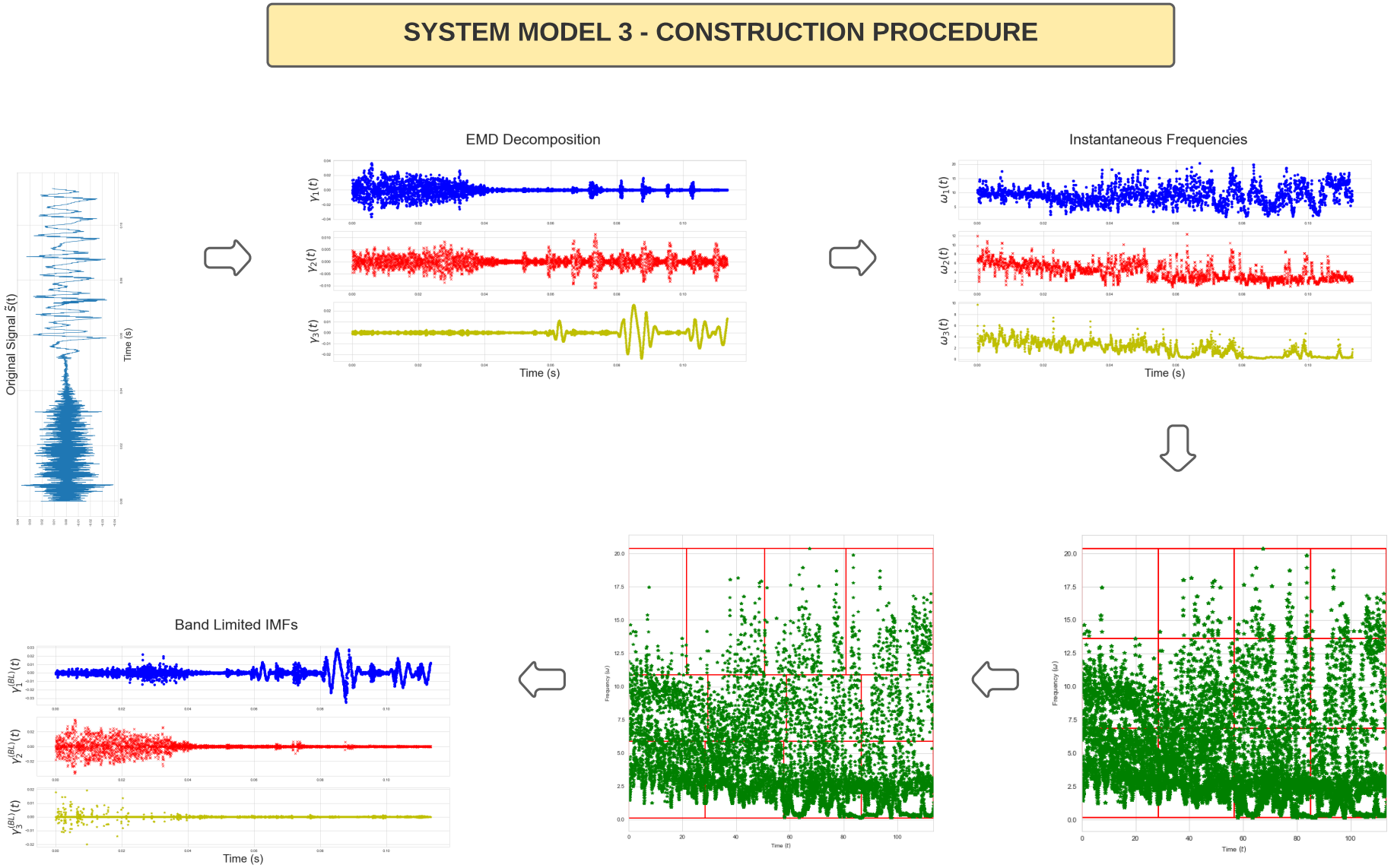


Figure 6.2: Figure presenting the steps required for the implementation of System Model 3. Note that, the fourth step represent the initial partition Π^0 used to initialised the cross-entropy procedure, while the fifth step is instead the estimated $\hat{\Pi}$.

6.4 Model Validation with the Generalised Likelihood Ratio Test

In this section, the model validation framework for the presented Gaussian processes is presented. Note that this setting will be developed for Chapter 9 where the conducted experiments will refer to speech voice signals. Particularly, the tested models will aim to differentiate between two populations of utterances, one affected by Parkinson's disease and the second describing healthy subject. The complete set of experiments and the attached results are fully provided in Chapter 9. At this stage, the focus is on the validation of the Gaussian process whose hyperparameters are identified in the estimation section.

Denote two family of Gaussian process that will be tested, namely $\widehat{S}(t)_0$ and $\widehat{S}(t)_1$. Then the two models that will be compared are:

$$\begin{aligned} \text{Model}_0 :\ & S(t)_0 \sim \mathcal{GP}\left(0, k_0(t,t')\right) \quad \forall t \in [t_1, t_N] \\ \text{Model}_1 :\ & S(t)_0 \sim \mathcal{GP}\left(0, k_1(t,t')\right) \quad \forall t \in [t_1, t_N] \end{aligned} \tag{6.37}$$

These are, by definition, Gaussian Processes with null mean function and covariance function constructed with one of the presented kernel function given in Chapter 4. The test comparing the the models given in 6.37 is a test comparing two distributions. This can be translated as

$$\begin{aligned} H_0 :\ & \mathcal{GP}\left(0, k_0(t,t')\right) = \mathcal{GP}\left(0, k_1(t,t')\right) \quad \forall t \in [t_1, t_N] \\ H_1 :\ & \mathcal{GP}\left(0, k_0(t,t')\right) \neq \mathcal{GP}\left(0, k_1(t,t')\right) \quad \forall t \in [t_1, t_N] \end{aligned} \tag{6.38}$$

Since a GP is also specified by its sufficient mean and covariance functions, testing for equality of distributions will be equivalent to testing for equality of the mean functions and the covariance functions. Hence, the distributional statements about population quantities in the null and alternative hypothesis are equivalent to the following population statements on the covariance functions only (note that the part related to mean function is omitted since the considered GPs will be all zero-mean GPs):

$$\begin{aligned} H_0 :\ & k_0(t,t') = k_1(t,t') \quad \forall t \in [t_1, t_N] \\ H_1 :\ & k_0(t,t') \neq k_1(t,t') \quad \forall t \in [t_1, t_N] \end{aligned} \tag{6.39}$$

Note that both covariance functions needs to be in the class of the Mercel Kernels (see Zaremba and Peters (2020), Garthwaite et al. (2002)). If the classes of covariance functions are restricted so that the Model_0 is nested in the Model_1, then the above hypotheses can be tested with the Generalised Likelihood Ratio Test (GLRT). Remark that the GLRT (see Garthwaite et al. (2002)) is a composite hypothesis test that can be used if the parameters are unknown and need to be estimated. It uses asymptotic distribution of the test statistic but it requires that the hypotheses are nested, that can be expressed in terms of restriction on mean and covariance formulations. For a review of the GLRT see Garthwaite et al. (2002). Under the test given in Eqn. 6.39, the null hypothesis is that there

is no difference between the model Model_0 and the model Model_1. This means that the employed GPs structure will not differentiate the two families. Hence, one wishes to test $H_0 : \boldsymbol{\theta} \in \Theta_0$ and $H_1 : \boldsymbol{\theta} \in \Theta - \Theta_0$. For any $S(t)$ distributed according to a GP with zero mean and covariance function $k(t, t'; \boldsymbol{\theta}_k)$, it is possible to write:

$$\log f(\tilde{s}(t)|\, \boldsymbol{\theta}_k) = -\frac{1}{2}\,(\tilde{s}(t) - 0)^{\intercal}\ \left(k(t, t'; \boldsymbol{\theta}_k)\right)^{-1}\ (\tilde{s}(t) - 0) \tag{6.40}$$

$$-\frac{1}{2}\ \log\ |k(t, t'; \boldsymbol{\theta}_k)| - \frac{n}{2}\log(2\pi) \quad \forall t \in [t_1, t_N] \tag{6.41}$$

where a constant variance of 1 is assumed for the error term. This leads to the definition of the test statistic corresponding to

$$L = \max_{\boldsymbol{\theta}_0}\Big[\log f(\tilde{s}(t)|\boldsymbol{\theta}_0)\Big] - \max_{\boldsymbol{\theta}_1}\Big[\log f(\tilde{s}(t)|\boldsymbol{\theta}_1)\Big] \tag{6.42}$$

For some constant A the test statistic makes use of critical region $L \leq A$. By defining d the difference in dimensionality of H_0 and $H_0 \cup H_1$, then one has that under the null hypothesis the asymptotic distribution of the test statistic is given according to

$$-2\log L \sim \mathcal{X}^2_d \tag{6.43}$$

By using the vectorial notation and a set of signals selected for the validation procedure and denoted as $\tilde{s}(\boldsymbol{t})^{\text{ts}}$, then the test statistic is given by:

$$\begin{aligned}\hat{L}_1 = &-(\tilde{s}(\boldsymbol{t})^{\text{ts}})^{\intercal}\left(\hat{\mathbf{K}}_0\right)^{-1}(\tilde{s}(\boldsymbol{t})^{\text{ts}}) - \log\left(\det\left[\hat{\mathbf{K}}_0\right]\right)\\ &+(\tilde{s}(\boldsymbol{t})^{\text{ts}})^{\intercal}\left(\hat{\mathbf{K}}_1\right)^{-1}(\tilde{s}(\boldsymbol{t})^{\text{ts}}) + \log\left(\det\left[\hat{\mathbf{K}}_1\right]\right)\end{aligned} \tag{6.44}$$

The above test is defined for the original signal $\tilde{s}(\boldsymbol{t})$. Given the introduced system models in section 6.3, it refers to system model one, hence the subscript $\hat{L}_1$. An equivalent test will be also conducted for the Gaussian process estimated for system model two and for system model three. Therefore, by considering a set of selected IMFs for the validation procedure and denoted as $\gamma_l(\boldsymbol{t})^{\text{ts}}$ for system model two and $\gamma_m(\boldsymbol{t})^{(BL),\text{ts}}$ for system model three, the following test will be defined and conducted

$$\begin{aligned}\hat{L}_2 = &-(\gamma_l(\boldsymbol{t})^{\text{ts}})^{\intercal}\left(\hat{\mathbf{K}}_{0,l}\right)^{-1}(\gamma_l(\boldsymbol{t})^{\text{ts}}) - \log\left(\det\left[\hat{\mathbf{K}}_{0,l}\right]\right)\\ &+(\gamma_l(\boldsymbol{t})^{\text{ts}})^{\intercal}\left(\hat{\mathbf{K}}_{1,l}\right)^{-1}(\gamma_l(\boldsymbol{t})^{\text{ts}}) + \log\left(\det\left[\hat{\mathbf{K}}_{1,l}\right]\right) \quad \forall l = 1, \ldots, L\\ \hat{L}_3 = &-(\gamma_m(\boldsymbol{t})^{(BL),\text{ts}})^{\intercal}\left(\hat{\mathbf{K}}_0, m\right)^{-1}(\gamma_m(\boldsymbol{t})^{(BL),\text{ts}}) - \log\left(\det\left[\hat{\mathbf{K}}_{0,m}\right]\right)\\ &+(\gamma_m(\boldsymbol{t})^{(BL),\text{ts}})^{\intercal}\left(\hat{\mathbf{K}}_{1,m}\right)^{-1}(\gamma_m(\boldsymbol{t})^{(BL),\text{ts}}) + \log\left(\det\left[\hat{\mathbf{K}}_{1,m}\right]\right) \quad \forall m = 1, \ldots, M\end{aligned}$$

The above tests will be carried to identify the discrimination power associated with the different IMFs stochastic embedding proposed. In this way, each embedded IMF and band limited IMFs will be individually tested.

Chapter 7

The Cross-Entropy Method

This Chapter provides an overview of the cross-Entropy method employed to solve a combinatorial search defining the partition problem required for the construction of system model 3 given in Chapter 6. The combinatorial search looks for suitable frequency partitions of the IFs samples, optimal for constructing a set of bases for EMD stochastic embedding based on band-limited representations. Remark that the proposed model aims to reassign the IFs samples within pre-selected frequency bandwidths and then construct this new set of basis functions based on such relocated IFs. At each specific time t_i, a new IMF, called band-limited IMF, is defined as the sum of the samples of the original IMFs whose IFs fell into the common identified frequency bandwidth (see model in subsection 6.3.3 in Chapter 6). Hence, the model has been ideally presented as one had achieved the notion of optimal frequency partition separating the existing IF samples. The term optimality in this regard has to be interpreted with the understanding of the cross-entropy method. Such a term is related to the idea of isolating core partitions in the frequency domain to produce a family of frequency band-limited bases that allow the user to focus on localised frequency contents. These can then be used to either reconstruct the entire stochastic signal representation or represent the non-stationary signal on a fixed time-scale related to the selected frequency band.

If the underlying process were comprised of stationary components, then classical Fourier methods would efficiently identify the frequency band-limited representations over time. However, the challenge faced in the developed settings of this thesis is that the considered signal is a realisation of a non-stationary stochastic process. Hence, the requirement of a method to produce an efficient partition of the time-frequency domain dealing with such a setting is of high priority. What is more, the frequency representation of the IMFs is the instantaneous frequency. Such a representation does not behave as standard stationary harmonics derived from Fourier decompositions. Therefore, it is a much more challenging task to identify what partitions of the frequency plane efficiently capture the time locations of energy concentration present in the signal.

In forming a band-limited representation, different criteria could be chosen. The

criterion of interest in this work is that one would have a finite fixed number of frequency partitions and, for each of these partitions, the energy content is equal. The optimality concept addressed above relates to the consideration of this criterion in the definition of the band-limited basis functions. Hence, the obtained time-frequency partition will be optimal because the derived band-limited basis functions will be defined as close as possible to a set of bases carrying equal frequency content.

According to such a criterion of optimality, a reference frequency partition for the time-frequency plane is given. In practice, one has an empirical representation obtained from the instantaneous frequencies derived from the existing IMFs. For both partitions, it is possible to compute a distribution describing it, where a uniform distribution (for each frequency bandwidth) will be defined for the reference frequency partition, and an empirical distribution will be given for the one obtained from the IFs. To derive the optimal partition for the given IFs sample, the discrepancy between the two distributions will be measured with the Kullback-Leibler (KL) distance. By considering such a measure, it is natural to consider the cross-entropy method to define the optimal partition and reformulate this optimisation problem as a cross-entropy problem.

One could think about the problem of choosing partitions being equivalent to choosing points on a mesh of the 2-d time-frequency plane that will form the desired partition. Mathematically, this is equivalent to a quantisation problem of the 2-d time-frequency plane that one has to set up to define the optimal partition through the minimisation problem between the empirical distribution obtained over the IFs in the 2-d time-frequency plane and the target distribution which has been selected to be uniform within each frequency band in time. This problem then becomes a combinatorial search problem that cannot be easily solved, given that there are too many possible choices. Consequently, the cross-entropy is employed since it offers a stochastic optimal combinatorial solution

The cross-entropy method (CEM) is a stochastic optimisation technique that was first presented by Rubinstein in 1999 (see Rubinstein (1999) Rubinstein (1997), Kroese et al. (2011), Rubinstein and Kroese (2004), De Boer et al. (2005)) employed for solving estimation and optimisation problems in general. It represents an efficient solution to solve NP-hard combinatorial optimisation problems by translating the deterministic search problem into a stochastic optimisation procedure. A core component of the CEM is that it exploits an Importance Sampling (IS) framework (see Homem-de Mello and Rubinstein (2002), Asmussen et al. (2005)) to approximate the optimal solution and will be later introduced. Furthermore, note that, in the main literature of CEM minimising the Kullback–Leibler (KL) divergence, the distributions are commonly referred to as the target (true) distribution treated as an ideal model for the data (in this case, a uniform distribution) and an empirical distribution (an approximation of the true distribution), which refers to the given sample observations (the IFs samples in this case).

The sought problem is the construction of a fully data-adaptive partition, since relying on the observed instantaneous frequencies, providing band-limited frequency components whose location is unknown a priori. The first element required to develop such a procedure is the definition of a partitioning rule for the frequency domain able to separate the sample points contained in it optimally. Afterwards, the optimisation problem that needs to be solved to find such a partition will be introduced. The first part of the chapter will present such components. The following step is represented by formally introducing the CEM for the previously formulated optimisation problem, and two different solutions will be provided. It is essential to highlight that certain zones of the frequency domain might present empty parts during the optimisation procedure, i.e. with no sample points; this could cause an issue and produce misleading results. A kernel density estimator will be employed to ensure an efficient optimisation procedure to avoid such an issue. Then the two formulations of the CEM are presented. The first one will consider the optimisation problem as a continuous one, while the second formulation will instead consider a discrete distribution.

7.1 Optimal Partition by Cross Entropy Method for Frequency and Time Domains

This section introduces the random search optimal partition rule describing the frequency domain and the optimisation problem formulation used to find such a partition.

7.1.1 Defining Partitioning Rule

To define the model, assume having NL two dimensional points $\mathbf{p}_{l,n} = (t_n, \omega_l(t_n))$ for $l = 1, \ldots, L$ and $n = 1, \ldots, N$. The points are given by a point-wise evaluation of the original instantaneous frequencies corresponding to L IMFs on N ordered times $t_0 < t_1 < \ldots < t_n < \ldots < t_{N-1} < t_N$ when IMFs values are observed.

Denote $\mathcal{T} = [t_0, t_N]$ the time interval and $\mathcal{I} = [\omega_0, \omega_M]$ the frequency interval, where $\omega_0 = \min_{n,l} \omega_l(t_n)$ and $\omega_M = \max_{n,l} \omega_l(t_n)$. Hence t_N ω_M will be given and obtained from the data, where t_N represents the length of the given signal and ω_M the maximum frequency achieved by the set of instantaneous frequencies. Define the two-dimensional rectangle $\Pi = \mathcal{I} \times \mathcal{T}$, which total area is given as follows:

$$|\Pi| := |\mathcal{I}| \times |\mathcal{T}| = (\omega_M - \omega_0)(t_N - t_0). \tag{7.1}$$

Recall that $\mathbf{p}_{nl} = (t_n, \omega_l(t_n)) \in \Pi$. The final interest is in representing the area $|\Pi|$ via an optimal partition Π^* defined through a discretised representation over a grid of $M \times D$ smaller rectangles. Particularly, assume that the frequency domain is partitioned into M subintervals, $\mathcal{I}_m := [\omega_{m-1}, \omega_m]$ for $m = 1, \ldots, M$

such that

$$\mathcal{I} = \bigcup_{m=1}^{M} \mathcal{I}_m, \text{ s.t. } \bigcap_{m=1}^{M} \mathcal{I}_m = \emptyset \text{ and } |\mathcal{I}| = \sum_{m=1}^{M} |\mathcal{I}_m|. \tag{7.2}$$

The rectangle $\mathcal{I}_m \times [t_0, t_N]$ is further divided into D smaller rectangles that have the same width, by partitioning the time interval $\mathcal{T} = [t_0, t_N]$ into D intervals $\mathcal{T}_{m,d} = [s_{m,d-1}, s_{m,d}]$ for $d = 1, \ldots, D$ such that $t_0 = s_0 < s_{m,d-1} < s_{m,d} \leq s_{m,D} = t_N$

$$\mathcal{T} = \bigcup_{d=1}^{D} \mathcal{T}_{m,d}, \text{ s.t. } \bigcap_{d=1}^{D} \mathcal{T}_{m,d} = \emptyset \text{ and } |\mathcal{T}| = \sum_{d=1}^{D} |\mathcal{T}_{m,d}| \tag{7.3}$$

Remark that it is not necessary that $|\mathcal{T}_{m,d}| = |\mathcal{T}_{m',d}|$ for $m \neq m'$ and $m, m' = 1, \ldots, M$.

Given the above, the main rectangle Π is partitioned by defining MD rectangles $\Pi_{m,d} = \mathcal{I}_m \times \mathcal{T}_{m,d}$ for $m = 1, \ldots, M$ and $d = 1, \ldots, D$. The rectangles that are defined by this partition are assumed to not overlap and as a result they satisfy

$$\Pi = \bigcup_{m,d}^{D} \Pi_{m,d}, \text{ s.t. } \bigcap_{m,d} \Pi_{m,d} = \emptyset \text{ and } |\Pi| = \sum_{m,d} |\Pi_{m,d}| \tag{7.4}$$

Remark that by this construction, the rectangles $\Pi_{m,d}$ that have the same index m share the same subinterval of $\mathcal{I}$ on the frequency axis, $\mathcal{I}_m$. However, the rectangles $\Pi_{m,d}$ and $\Pi_{m',d}$, when $m \neq m'$, $m, m' = 1, \ldots, M$, that have the same index d do not share the same subintervals on the time axis since $\mathcal{T}_{m,d} \neq \mathcal{T}_{m',d}$.

Assume one has a collection of L distinct IFs for the L IMFs obtained from the EMD procedure applied to signal $\widetilde{s}(t)$. Now assume that one has the discrete outputs of the HHT for these IFs represented by the point set $\mathcal{P} = \bigcup \mathcal{P}_{m,d}$, where $\mathcal{P}_{m,d}$ are defined as follows. Then one can consider the subset of such points per partition region which will be denoted by $\mathcal{P}_{m,d} = \left\{ \mathbf{p}_{l,n} : \mathbf{p}_{l,n} \in \Pi_{m,d} \right\}$ the set of points $\mathbf{p}_{l,n}$ that are located in a partition $\Pi_{m,d}$. The cardinality of this set is denoted by $|\mathcal{P}_{m,d}|$. Note, the total number of samples representing the IFs will be controlled by the user when approximating the Hilbert transform for the IMFs. As discussed previously, in some cases, the Hilbert transform is known in closed form for some IMF representations. Here, it is presented in a general setting when one may utilise an IMF representation of any form and as such would require a discrete approximation of the Cauchy Principal Value integral for the Hilbert transform.

Hence, the challenge becomes how to define a notion of optimal partition, in what sense it will be optimal and then how to solve for the optimal solution. The notion of optimality is selected here to reflect a concept of equi-energy partition in the time-frequency plane, which when one has a non-stationary signal comprised not of a finite collection of constant in time pure harmonics, one will have an IMF finite collection of basis functions which for the IFs will be time-varying signals in the time-frequency plane. The samples obtained of these signals from the transform will produce a 2-d histogram, and therefore to achieve

an equi-energy partition rule, the problem becomes equivalent to solving a density optimisation problem to produce an empirical ECDF as close as possible to a uniform distribution in 2d for a given number of partitions in time and frequency. This can be reposed in a discrete quantised framework as a combinatorial search problem.

7.1.2 Formulation of the Optimisation Problem for the Random Partition

The optimal partition Π^* is specified for the available sample set $\mathbf{p}_{l,n}$ given the problem statement described below. Suppose one takes the area Π and partition it according to $M-1$ horizontal partitions for the frequency axis and D vertical partitions for the time axis. Remark that, for the framework introduced in the above section, the horizontal frequency partitions are assumed constant over time, while, the vertical time partitions might vary within each band. Therefore, the problem will be restricted to only consider horizontal partitions of the frequency plane which are fixed across time in order to maintain interpretation of the band-limited basis function representation obtained. On the contrary, adaptivity on the time domain on the 2-d plane will be considered. Hence,the partition is parametrised according to increasing sequence of the frequency parameters $\omega_1, \ldots, \omega_{M-1}$, defining subintervals of $\mathcal{I}$, and D increasing sequences of time parameters $s_{m,1}, \ldots, s_{m,D-1}$, which defines subintervals of $\mathcal{T}$ for different m. The set of parameters that are to be estimated is denoted by:

$$\boldsymbol{\psi} = [\omega_1, \ldots, \omega_{M-1}, s_{1,1}, \ldots, s_{1,D-1}, \ldots, s_{m,1}, \ldots, s_{m,D-1}, \ldots, s_{M,1}, \ldots, s_{M,D-1}]\,. \tag{7.5}$$

The objective is to learn an optimal partition Π^* that will produce an empirical distribution function for the IFs in each sub-rectangle $\Pi_{m,d}$ which is as close to uniform distribution across the domain area Π as possible. To achieve this, the CEM relies on Importance Sampling and the definition of its distribution called the Importance distribution. To introduce such a distribution, assume one has a discrete random variable X that would represent the boundaries defining the sought, optimal partition Π^*, hence the tuples (m, d). This random variable X is characterised by the Importance distribution.

Hence, to estimate the elements of the vector $\boldsymbol{\psi}$ that define the optimal partition Π^*, consider such a discrete random variable X defined on the the indexes of the sub-rectangles $\Pi_{m,d}$, therefore, on the tuples (m, d) with corresponding probabilities that sum to 1. As a result, the set of possible values taken by X consists of DM tuples (m, d), for $m = 1, \ldots, M$ and $d = 1, \ldots, D$. Therefore, X controls assignments of a points to sub-rectangles $\Pi_{m,d}$.

Ideally, the target distribution of X will be uniform and characterized by a probability density function $\pi(x)$ defined as

$$\pi(x) = \prod_{m,d} \pi_{m,d}^{\mathbf{1}_{\{x=(m,d)\}}} \text{ for } \pi_{m,d} = \mathbb{P}\Big(X = (m,d)\Big) = \frac{|\Pi_{m,d}|}{|\Pi|}. \tag{7.6}$$

Therefore, the target distribution associates the probability of drawing tuple (m, d) to the proportion of the area of rectangle $\Pi_{m,d}$ to the overall area of Π.

However, what one actually has is an empirical distribution for X, as it directly depends on the points $p_{k,n}$, and is denoted as $\hat{\pi}(x)$. It is associated with the measure $\hat{\mathbb{P}}$ on the sample $p_{n,k}$ defined as

$$\hat{\pi}(x) = \prod_{m,d} \hat{\pi}_{m,d}^{\mathbf{1}_{\{x=(m,d)\}}} \text{ for } \hat{\pi}_{m,d} = \hat{\mathbb{P}}\Big(X = (m,d)\Big) = \frac{|\mathcal{P}_{m,d}|}{LN}, \tag{7.7}$$

Therefore, the probability of drawing tuple (m, d) reflects the proportion of the number of points $p_{n,l}$ that lay within the rectangle $\Pi_{m,d}$ to the overall sample size. Remark that both $\pi_{m,d}$ and $\hat{\pi}_{m,d}$ satisfy

$$\sum_{m,d} \pi_{m,d} = 1 \text{ and } \sum_{m,d} \hat{\pi}_{m,d} = 1. \tag{7.8}$$

The final goal is to select the support of X in such a way that the Kullback-Leibler divergence, measuring similarity between the two proposed distributions, is minimised. The Kullback-Leibler divergence defined as

$$KL(\pi, \hat{\pi}) = \int_{x \in \mathcal{X}} \pi(x) \log\left(\frac{\pi(x)}{\hat{\pi}(x)}\right) dx. \tag{7.9}$$

Since X is a discrete random variable that ranges of values is countable and takes values in the set of tuples $\mathcal{X} = \{(m, d)\}_{m,d}$, the integration problem in (7.9) can

be rewritten as a sum over the elements of the set $\mathcal{X}$, that is

$$\begin{aligned}
KL(\pi, \hat{\pi}; \psi) &= \sum_{m=1}^{M} \sum_{d=1}^{d} \pi\left(x = (m,d)\right) \log\left(\frac{\pi\left(x = (m,d)\right)}{\hat{\pi}\left(x = (m,d)\right)}\right) \\
&= \sum_{m=1}^{M} \sum_{d=1}^{d} \left\{ \pi\left(x = (m,d)\right) \Big(\log \pi\left(x = (m,d)\right) - \log \hat{\pi}\left(x = (m,d)\right) \Big) \right\} \\
&= \sum_{m=1}^{M} \sum_{d=1}^{d} \left\{ \frac{|\Pi_{m,d}|}{|\Pi|} \Big(\log \frac{|\Pi_{m,d}|}{|\Pi|} - \log \frac{|\mathcal{P}_{m,d}|}{KN} \Big) \right\} \\
&= \frac{1}{|\Pi|} \sum_{m=1}^{M} \sum_{d=1}^{d} \left\{ |\Pi_{m,d}| \Big(\log |\Pi_{m,d}| - \log |\Pi| - \log |\mathcal{P}_{m,d}| + \log KN \Big) \right\} \\
&= \frac{1}{|\Pi|} \sum_{m=1}^{M} \sum_{d=1}^{d} \left\{ |\Pi_{m,d}| \Big(\log |\Pi_{m,d}| - \log |\mathcal{P}_{m,d}| \Big) \right\} \\
&+ \frac{1}{|\Pi|} \sum_{m=1}^{M} \sum_{d=1}^{d} \left\{ |\Pi_{m,d}| \Big(\log KN - \log |\Pi| \Big) \right\} \\
&= \frac{1}{|\Pi|} \sum_{m=1}^{M} \sum_{d=1}^{d} \left\{ |\Pi_{m,d}| \Big(\log |\Pi_{m,d}| - \log |\mathcal{P}_{m,d}| \Big) \right\} \\
&+ \Big(\log KN - \log |\Pi| \Big) \frac{1}{|\Pi|} \sum_{m=1}^{M} \sum_{d=1}^{d} |\Pi_{m,d}| \\
&= \log KN - \log |\Pi| + \frac{1}{|\Pi|} \sum_{m=1}^{M} \sum_{d=1}^{d} \left\{ |\Pi_{m,d}| \Big(\log |\Pi_{m,d}| - \log |\mathcal{P}_{m,d}| \Big) \right\}
\end{aligned} \tag{7.10}$$

The vector of parameters belong to the multidimensional parameters space Ψ defined by the following constraints on its elements

$$\Psi = \begin{cases}
\omega_1, \ldots, \omega_{M-1} \in (\omega_0, \omega_M) \text{ such that } \omega_0 < \omega_1 < \ldots < \omega_{M-1} < \omega_M, \\
s_{1,1}, \ldots, s_{1,N_1-1} \in (t_0, t_N) \text{ such that } t_0 < s_{1,1} < \ldots < s_{1,D-1} < t_N, \\
\vdots \\
s_{m,1}, \ldots, s_{m,N_m-1} \in (t_0, t_N) \text{ such that } t_0 < s_{m,1} < \ldots < s_{m,D-1} < t_N, \\
\vdots \\
s_{M,1}, \ldots, s_{M,N_M-1} \in (t_0, t_N) \text{ such that } t_0 < s_{M,1} < \ldots < s_{M,D-1} < t_N.
\end{cases} \tag{7.11}$$

Therefore, the objective function of the constrained optimisation problem that finds optimal partitioning of Π that minimizes distance between the empirical and target distributions is specified by

$$\psi^* = \underset{\psi \in \Psi}{\operatorname{argmin}}\, KL(\pi, \hat{\pi}; \psi) = \underset{\psi \in \Psi}{\operatorname{argmax}} -KL(\pi, \hat{\pi}; \psi) \tag{7.12}$$

7.2 The Cross-Entropy Method for Maximising Equation (7.12)

The objective function of the random partitioning problem defined in (7.12) employs $KL(\cdot)$ as a similarity measure between two distributions, the empirical and the target ones. The objective function is optimised with respect to the vector of parameters ψ that belongs to the parameter space Ψ and, consequently, is the domain of the objective function. Hence, the optimal partition Π^* will be treated as a parameter that have to be learnt in the optimal Importance Distribution denoted by ψ.

Next, the goal is to present how to estimate ψ by employing the cross-entropy method. For convenience, the notation for the KL divergence from now on will be $KL(\pi, \hat{\pi}; \psi) = KL(\psi)$. The optimisation problem is solved by considering the level sets of the objective function $\{\psi : KL(\psi) \geq \gamma\}$ for $\gamma \in \mathbb{R}$. When $\gamma = \widehat{KL} = \text{argmax}_{\psi \in \Psi} KL(\psi)$, then $\{\psi : KL(\psi) \geq \gamma\} = \{\psi^\star\}$. Next, define a family of probability measure $\{\mathbb{P}_{\varphi'} : \varphi' \in \Phi\}$ on Ψ with densities $\{f_{\varphi'} : \varphi' \in \Phi\}$ that are parametrised by $\varphi' \in \Phi$. Let $\mathbb{E}_{\varphi'}$ denote the expectation taken with respect to $\mathbb{P}_{\varphi'}$. Fix φ' and γ and define a rare event probability problem:

$$\mathbb{P}_{\varphi'}\left[KL(\psi) \geq \gamma\right] = \mathbb{E}_{\varphi'}\left[\mathbb{I}_{\{KL(\psi)\leq\gamma\}}\right] = \int_\Psi \mathbb{I}_{\{KL(\psi)\leq\gamma\}} f_{\varphi'}(\psi) d\psi \tag{7.13}$$

Instead of approximating this probability naively by sampling from $f_{\varphi'}$, the importance sampling method is used. Let $g_{\varphi''}$ denote the importance sampler, where $\varphi'' \in \Phi$. Importance sampling approximates the rare event probability by

$$\begin{aligned}\mathbb{P}_{\varphi'}\left[KL(\psi) \geq \gamma\right] &= \int_\Psi \mathbb{I}_{\{KL(\psi)\leq\gamma\}} f_{\varphi'}(\psi) d\psi = \int_\Psi \mathbb{I}_{\{KL(\psi)\leq\gamma\}} \frac{f_{\varphi'}(\psi)}{g_{\varphi''}(\psi)} g_{\varphi''}(\psi) d\psi \\ &= \mathbb{E}_{\varphi''}\left[\mathbb{I}_{\{KL(\psi)\leq\gamma\}} \frac{f_{\varphi'}(\psi)}{g_{\varphi''}(\psi)}\right] \approx \frac{1}{S}\sum_{i=1}^{S}\left\{\mathbb{I}_{\{KL(\psi^i)\leq\gamma\}} \frac{f_{\varphi'}(\psi^i)}{g_{\varphi''}(\psi^i)}\right\}\end{aligned} \tag{7.14}$$

where vectors ψ^i for $i = 1, \ldots, S$ are iid samples generated from $g_{\varphi''}(\psi)$. The optimal importance sampler $g_{\varphi''}$ is selected through the cross-entropy criterion:

$$\begin{aligned}\varphi^\star &= \underset{\varphi'' \in \Phi}{\text{argmax}} \int_\Psi \mathbb{I}_{\{KL(\psi)\leq\gamma\}} f_{\varphi'}(\psi) \log \frac{f_{\varphi'}(\psi)}{g_{\varphi''}(\psi)} d\psi \\ &\approx \underset{\varphi'' \in \Phi}{\text{argmax}} \frac{1}{S}\sum_{i=1}^{S} \mathbb{I}_{\{KL(\psi)\leq\gamma\}} \log g_{\varphi''}(\psi^i)\end{aligned} \tag{7.15}$$

where vectors ψ^i for $i = 1, \ldots, S$ are iid samples generated from $f_{\varphi'(\psi)}$. Notice that the last line of 7.15 corresponds to the maximum likelihood estimation (MLE) of φ'' when the samples are $\{\psi^i : KL(\psi^i) \geq \gamma\}$. The CEM starts from an initial sampling distribution $g_{\varphi_0^\star}$ and iteratively updates the threshold $\hat{\gamma}$ and the sampling distribution $g_{\varphi''}$. For further details on the cross-entropy, the reader should refer to De Boer et al. (2005).

7.2.1 Utilising Kernel Density Estimator in Kullback-Leibler Divergence of the Partitioning Problem

It can happen that during the estimation process specifying the optimal partition Π, certain sub-rectangles $\Pi_{m,d}$ do not contain any of the sample points $\mathbf{p}_{l,n} = (t_n, \omega_l(t_n)) \in \Pi$ for $\Pi = \mathcal{I} \times \mathcal{T}$ for $n = 1, \dots N$ and $l = 1, \dots, L$. As a result, the corresponding set $\mathcal{P}_{m,d}$ is empty, that is $\mathcal{P}_{m,d} = \emptyset$. Consequently, the probabilities $\pi^e_{m,d}(x) = \frac{|\mathcal{P}_{m,d}|}{LN}$ equal zero and their logarithms used to calculate $KL(\pi, \hat{\pi}; \psi)$ in (7.10) tend to infinity. To avoid these numerical difficulties $\pi^e_{m,d}(x)$ is approximated by a kernel density estimator $\hat{\pi}^e_{m,d}(x; k, h)$ parametrised by kernel $k : \Pi \times \Pi \to \mathbb{R}$ and bandwidth $h > 0$ such that

$$\hat{\pi}^e_{m,d}(x; k, h) = \int_{\Pi_{m,d}} \hat{\pi}(\mathbf{p}; k; h) d\mathbf{p} = \int_{\omega_{m-1}}^{\omega_m} \int_{s_{m,d-1}}^{s_{m,d}} \hat{\pi}(\mathbf{p}; k; h) d\mathbf{p},$$

where $\hat{\pi}(\mathbf{p}; k; h) : \Pi \to [0, 1]$ is a kernel density estimator of points $\mathbf{p} = (t, \omega(t)) \in \Pi$ specified on a sample set $\mathbf{p}_{l,n}$

$$\hat{\pi}(\mathbf{p}; k; h) = \frac{1}{Nh} \prod_{n=1}^{N} \prod_{k=1}^{K} k\left(\frac{\mathbf{p} - \mathbf{p}_{n,k}}{h}\right) \quad \text{such that} \quad \int_{\Pi} \hat{\pi}(\mathbf{p}; k, h) d\mathbf{p} = 1.$$

The objective function of the partitioning problem in (7.9) or (7.10) is reformulated to be the Kullback-Leibler divergence between $\pi(x)$ and

$$\hat{\pi}^e(x; k, h) = \prod_{m,d} \left(\hat{\pi}^e_{m,d}(x; k, h)\right)^{\mathbf{1}_{\{x=(m,d)\}}}, \tag{7.16}$$

that is

$$\begin{aligned}
KL(\pi, \hat{\pi}^e; \psi) &= \int_{x \in \mathcal{X}} \pi(x) \log\left(\frac{\pi(x)}{\hat{\pi}^e(x; k, h)}\right) dx \\
&= \sum_{m=1}^{M} \sum_{d=1}^{d} \pi(x = (m, d)) \log\left(\frac{\pi(x = (m, d))}{\hat{\pi}^e(x = (m, d); k, h)}\right) \\
&= \sum_{m=1}^{M} \sum_{d=1}^{d} \left\{ \frac{|\Pi_{m,d}|}{|\Pi|} \Big(\log|\Pi_{m,d}| - \log|\Pi| - \log \hat{\pi}^e(x = (m, d); k, h) \Big) \right\} \\
&= \frac{1}{|\Pi|} \sum_{m=1}^{M} \sum_{d=1}^{d} \left\{ |\Pi_{m,d}| \Big(\log|\Pi_{m,d}| - \log \hat{\pi}^e(x = (m, d); k, h) \Big) \right\} \\
&\quad - \frac{\log|\Pi|}{|\Pi|} \sum_{m=1}^{M} \sum_{d=1}^{D} |\Pi_{m,d}| \\
&= -\log|\Pi| + \frac{1}{|\Pi|} \sum_{m=1}^{M} \sum_{d=1}^{d} |\Pi_{m,d}| \Big(\log|\Pi_{m,d}| - \log \hat{\pi}^e(x = (m, d); k, h) \Big)
\end{aligned} \tag{7.17}$$

and with a numerical trick, for $C > 0$ being a very small number, i.e. $C = 10^{-100}$

$$\begin{aligned}
KL(\hat{\pi}^e, \pi; \psi) &= \int_{x\in\mathcal{X}} \pi(x) \log\left(\frac{\pi(x)}{\hat{\pi}^e\left(x; k, h\right)}\right) dx \\
&= \sum_{m=1}^{M}\sum_{d=1}^{d} \pi\left(x = (m,d)\right) \log\left(\frac{\pi\left(x=(m,d)\right)}{\hat{\pi}^e(x=(m,d); k, h)}\right) \\
&= \sum_{m=1}^{M}\sum_{d=1}^{d} \left\{\frac{|\Pi_{m,d}|}{|\Pi|}\Big(\log|\Pi_{m,d}| - \log|\Pi| - \log C\frac{\hat{\pi}^e(x=(m,d); k, h)}{C}\Big)\right\} \\
&= \sum_{m=1}^{M}\sum_{d=1}^{d} \left\{\frac{|\Pi_{m,d}|}{|\Pi|}\Big(\log|\Pi_{m,d}| - \log|\Pi| - \log C - \log\frac{\hat{\pi}^e(x=(m,d); k, h)}{C}\Big)\right\} \\
&= \frac{1}{|\Pi|}\sum_{m=1}^{M}\sum_{d=1}^{d} \left\{|\Pi_{m,d}|\Big(\log|\Pi_{m,d}| - \log\frac{\hat{\pi}^e(x=(m,d); k, h)}{C}\Big)\right\} \\
&- \frac{\log|\Pi| + \log C}{|\Pi|}\sum_{m=1}^{M}\sum_{d=1}^{D}|\Pi_{m,d}| \\
&= -\log|\Pi| - \log C + \frac{1}{|\Pi|}\sum_{m=1}^{M}\sum_{d=1}^{d}|\Pi_{m,d}|\Big(\log|\Pi_{m,d}| - \log\frac{\hat{\pi}^e(x=(m,d); k, h)}{C}\Big)
\end{aligned} \tag{7.18}$$

In the below subsections, the importance distribution to solve the optimisation problems are introduced. Note that they are introduced for optimising Equation (7.15) but could be easily adapted to the case considering the KL divergence derived in this subsection.

7.2.2 Cross-Entropy Method Selection of Importance Distribution: Continuous Case via Truncated Normal

The optimisation problem in (7.15) can be approached as a continuous optimisation problem. The vectors ψ^i for $i = 1, \ldots, S$ are iid realisations from $g(\psi; \varphi)$. The elements of the random vector ψ are assumed to be independent random variables such that their joint distribution can be factorised as

$$g(\psi; \varphi) = \prod_{m=1}^{M}\left\{g_{\omega_m}(\omega_m; \varphi_m)\prod_{d=1}^{D} g_{s_{m,d}}(s_{m,d}; \varphi_{m,d})\right\} \tag{7.19}$$

Assume that each probability distribution $g_x(x; \varphi_x)$ corresponds a truncated univariate normal distribution parametrised by mean μ_x and standard deviation σ_x^2. Hence, $\varphi_x = [\mu_x, \sigma_x^2]$, the truncation for each ω_m corresponds to the fixed ends ω_0 and ω_M and the truncation for each $S_{m,d}$ corresponds to the fixed ends t_0 and t_N that are specified by the available sample sets in frequency and time domains, respectively. Therefore, the following distributions are proposed

$$W_m \sim \mathcal{N}_{[\omega_0,\omega_M]}\Big(\mu_m, \sigma_m^2\Big) \text{ and } S_{m,d} \sim \mathcal{N}_{[t_0,t_N]}\Big(\mu_{m,d}, \sigma_{m,d}^2\Big) \tag{7.20}$$

where the probability density function of $X \sim \mathcal{N}_{[a,b]}\left(\mu, \sigma^2\right)$ such that $a \leq X \leq b$ is given as follows

$$g_x(x; \mu, \sigma^2, a, b) = \frac{1}{\sigma} \frac{\phi\left(\frac{x-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)},$$

with $\phi(x)$ being the probability density function of a standard normal random variable and $\Phi(x)$ is the corresponding cumulative distribution function, given as

$$\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-0.5x^2}, \ \Phi(x) = \frac{1}{2}\left(1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right), \ \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \quad (7.21)$$

The described importance sampling distributions of W_m and $S_{m,d}$ with the truncation ensure that the realisation of W_m and $S_{m,d}$ are admissible and stay in the constrained space of the optimisation problem. Consequently, the estimates μ_m and $\mu_{m,d}$ satisfy the conditions of the feasible set Ψ. On the other hand, non-linearities are introduced to the estimation of μ_m and $\mu_{m,d}$ due to the presence of the cumulative distribution function $\text{erf}(x)$. These non-linearities in the estimation equations for μ_m, $\mu_{m,d}$, σ_m^2 and $\sigma_{m,d}^2$ may lead to numerical instability of the CEM algorithm. Given such a practical consideration, the estimation of μ_m and $\mu_{m,d}$ are proposed to be obtained in a less optimal way. By removing the truncation and assuming a fixed value for σ_m^2 and $\sigma_{m,d}^2$, the updated rule of the parameters is given as

$$\begin{aligned}
\hat{\mu}_{m,d} &= \min\left\{\max\left\{t_0 + a\sigma_{m,d}^2; \frac{\sum_{i=1}^S \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)}) \leq \gamma\right\}} s_{m,d}^{(s)}}{\sum_{i=1}^S \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)}) \leq \gamma\right\}}}\right\}; t_N - a\sigma_{m,d}^2\right\} \\
\hat{\mu}_{m} &= \min\left\{\max\left\{\omega_0 + a\sigma_{m}^2; \frac{\sum_{i=1}^S \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)}) \leq \gamma\right\}} \omega_{m}^{(s)}}{\sum_{i=1}^S \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)}) \leq \gamma\right\}}}\right\}; \omega_M - a\sigma_{m}^2\right\} \quad (7.22)
\end{aligned}$$

which correspond to a truncated MLE estimator of the expected value under the assumption of a normal distribution of both $S_{m,d}$ and W_m, respectively, derived in Appendix H. The truncated estimators always satisfy the conditions of the feasible set Ψ. The scalars $a > 0$, σ_m^2 and $\sigma_{m,d}^2$ are chosen in such a way that the values $a\sigma_m^2$ and $a\sigma_{m,d}^2$ are much smaller than $|\mathcal{I}_m|$ and $|\mathcal{I}_{m,d}|$, respectively. Consequently, the location of first and last partitions close to the boundaries are controlled. Remark that for sufficiently small scalars σ_m^2 and $\sigma_{m,d}^2$, the truncation may not be needed.

The following window proposed the algorithm implemented to develop the continuous CEM generating the desired random partition for the frequency domain.

Algorithm 1: Random Partition via CEM for Continuous Optimisation

Input: Set $M, D, S > 0$
Input: Set hyperparameters: $\sigma_1^2, \ldots, \sigma_M^2, \sigma_{1,1}^2, \ldots, \sigma_{M,D}^2 > 0$, $a > 0$, $\rho > 0$,$\beta > 0$,
Input: Set initial parameters $\omega_0 + a\sigma_m^2 < \mu_m^{[0]} < \omega_M - a\sigma_m^2$ and $t_0 + a\sigma_{m,d}^2 < \mu_{m,d}^{[0]} < t_N - a\sigma_{m,d}^2$
for $i > 0$ **do**

1. Generate S sets of realisations $\psi^{(s)[i]} = \left[W_1^{(s)[i]}, \ldots, W_{M-1}^{(s)[i]}], S_{1,1}^{(s)[i]}, \ldots, S_{M,D-1}^{(s)[i]}\right]$ $W_m^{(s)[i]} \sim \mathcal{N}\left(\mu_m^{[i-1]}; \sigma_m^2\right)$, $S_{m,d}^{(s)[i]} \sim \mathcal{N}\left(\mu_{m,d}^{[i-1]}; \sigma_{m,d}^2\right)$;
2. Set $\omega_m^{(s)[k]} = \min\left\{\max\left\{\omega_0 + a\sigma_m^2; \omega_m^{(s)[k]}\right\}; \omega_M - a\sigma_m^2\right\}$, $s_{m,d}^{(s)[k]} = \min\left\{\max\left\{t_0 + a\sigma_{m,d}^2; s_{m,d}^{(s)[k]}\right\}; t_N - a\sigma_{m,d}^2\right\}$;
3. Calculate $KL(\hat{\pi}, \pi; \psi^{(s)[i]})$ for $s = 1, \ldots, S$ and specify $\gamma^{[i]}$ being $1 - \rho$ empirical quantile of their values;
4. Calculate

$$\hat{\mu}_m = \frac{\sum_{s=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi}, \pi; \psi^{(s)[i]}) \leq \gamma^{[i]}\right\}} w_m^{(s)[i]}}{\sum_{s=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi}, \pi; \psi^{(s)[i]}) \leq \gamma^{[i]}\right\}}}, \quad \hat{\mu}_{m,d} = \frac{\sum_{s=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi}, \pi; \psi^{(s)[i]}) \leq \gamma^{[i]}\right\}} s_{m,d}^{(s)[i]}}{\sum_{s=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi}, \pi; \psi^{(s)[i]}) \leq \gamma^{[i]}\right\}}}$$

5. Smooth update of the parameters

$$\mu_m^{[i]} = \beta\mu_m^{[i-1]} + (1-\beta)\hat{\mu}_m, \quad \mu_{m,d}^{[i]} = \beta\mu_{m,d}^{[i-1]} + (1-\beta)\hat{\mu}_{m,d}$$

$i = i + 1$

until a convergence criterion is satisfied

After convergence, specify points of partition $\omega_m = \mu_m^{[i]}$ and $s_{m,t} = \mu_{m,d}^{[i]}$.

7.2.3 Cross-Entropy Method Selection of Importance Distribution: Discrete Case via Multinomial Distribution

The optimisation problem in (7.15) is also solved through a discretisation of the intervals $\mathcal{I}$ and $\mathcal{T}$. In such a way, a CEM method with an IS distribution reflecting the distribution of discrete random variables that determine the partitioning of the rectangle Π is taken into account. Consider regular dense grids of $\mathcal{I}$ and $\mathcal{T}$ constructed as follows:

1. Partition $\mathcal{I}$ into small N_ω intervals of size $\Delta_\omega = \frac{\omega_M - \omega_0}{N_\omega}$, and define $\mathcal{I}_{n_\omega}^{grid} = \omega_0 + [n_\omega - 1, n_\omega]\Delta_\omega$ for $n_\omega = 1, \ldots, N_\omega$, therefore $|\mathcal{I}_a^{grid}| = \Delta_\omega$;
2. Partition $\mathcal{T}$ into small N_τ intervals of size $\Delta_\tau = \frac{t_N - t_0}{N_\tau}$, and define $\mathcal{T}_{n_\tau}^{grid} = \omega_0 + [n_\tau - 1, n_\tau]\Delta_\tau$ for $n_\tau = 1, \ldots, N_\tau$, therefore, $|\mathcal{T}_\tau^{grid}| = \Delta_\tau$.

Then, define a probabilistic model to partition $\mathcal{I}$ into M subintervals, $\mathcal{I}_m$ for $m = 1, \ldots, M$. Define (M)-dimensional multinomial random vector $\mathbf{X}$ whose entries X_m on the support of $\{0, \ldots, N_\omega\}$ indicates how many subsequent grids

$\mathcal{I}_{n_\omega}^{grid}$ are connected to construct partitions $\mathcal{I}_m$ and corresponding break points $\omega_{m-1}, \omega_m \in \mathcal{I}$. Therefore, the multinomial random vector $\mathbf{X}$ models the number of grid points out of N_ω that belong to each of M intervals with probabilities of being in an interval being $0 \leq p_1, \ldots, p_M \leq 1$ for $\sum_{m=1}^{M} p_m = 1$. The distribution function of $\mathbf{X}$ is formulated as

$$\pi(\mathbf{x}; \mathbf{p}) = \pi(x_1, \ldots, x_M; p_1, \ldots, p_M) = \frac{N_\omega!}{\prod_{m=1}^{M} x_m!} \prod_{m=1}^{M} p_m^{x_m}. \tag{7.23}$$

for $\mathbf{p} = [p_1, \ldots, p_M]$. Recall that $\sum_{m=1}^{M} X_m = N_\omega$ since $\mathbf{X}$ divides N_ω points into M subsets. For instance, for realisations of $X_1,\ X_2$ such that $x_1 = 2$ and $x_2 = 5$, the partitions $\mathcal{I}_1 = [\omega_0, \omega_1]$ and $\mathcal{I}_2 = [\omega_1, \omega_2]$ are given by

$$\omega_1 = \omega_0 + \Delta_\omega x_1 \text{ and } \omega_1 = \omega_1 + \Delta_\omega x_2 = \omega_0 + \Delta_\omega (x_1 + x_2)$$

This example gives an intuition for the general rule

$$\omega_m = \omega_0 + \Delta_\omega \sum_{m'=1}^{m} x_{m'} \text{ for } m = 1, \ldots M - 1.$$

and defines the approach to sample $W_1, \ldots, W_{M-1}$ via change of variables such that $W_m = \omega_0 + \Delta_\omega \sum_{m'=1}^{m} X_{m'}$ for $m = 1, \ldots M - 1$. The realisation of $W_1, \ldots, W_{M-1}$, denoted by $\omega_1, \ldots, \omega_{M-1}$, represent the break points defining partitions $\mathcal{I}_1, \ldots, \mathcal{I}_M$. Also, recall that ω_0 and $W_M = \omega_M$ are fixed.

Model M independent not identical partitions of the time-domain interval $\mathcal{T}$ into D subintervals by following the same steps as before. Define M independent multinomial random variables that are D-dimensional, each, denoted by $\mathbf{X}'_m$ for $m = 1, \ldots, M$, whose entries $X'_{m,d}$ on the support of $\{0, \ldots, N_\tau\}$, for $d = 1, \ldots, D$, specify how many subsequent grids $\mathcal{T}_{n_\tau}^{grid}$ are connected to construct partitions $\mathcal{T}_{m,d}$ of $\mathcal{T}$ and determine break points $s_{m,d-1}, s_{m,d} \in \mathcal{T}$. Denote their distributions by $\pi(\mathbf{x}'_m; \mathbf{p}'_m)$ for $\mathbf{p}'_m = [p'_{m,1}, \ldots, p'_{m,D}]$ such that $\sum_{d=1}^{D} p'_{m,d} = 1$. For every $m = 1, \ldots, M$ this construction satisfies $\sum_{d=1}^{D} X'_{m,d} = N_\tau$ and

$$s_{m,d} = t_0 + \Delta_\tau \sum_{d'=1}^{d} x'_{m,d'} \text{ for } d = 1, \ldots, D-1,\ m = 1, \ldots M.$$

where $x'_{m,d}$ is a realisation of $X'_{m,d}$. Therefore, the random variables $S_{m,1}, \ldots, S_{m,D-1}$ for $m = 1, \ldots, M$ are defined via change of variables such that $S_{m,d} = t_0 + \Delta_\tau \sum_{d'=1}^{d} X'_{m,d'}$ for $d = 1, \ldots D-1$ with realisations $s_{m,1}, \ldots, s_{m,D-1}$ representing the break points of the partitions $\mathcal{T}_{m,1}, \ldots, \mathcal{T}_{m,D}$. Again, recall that t_0 and $S_{m,D} = t_N$ are fixed for every $m = 1, \ldots, M$.

Given this model, the joint distribution of $\Psi = [W_1, \ldots, W_{M-1}, S_{1,1}, \ldots, S_{M,D-1}$ can we written as

$$g(\psi; \varphi) = C\ \pi(\mathbf{x}_m; \mathbf{p}) \prod_{m=1}^{M} \pi(\mathbf{x}'_m; \mathbf{p}'_m), \tag{7.24}$$

and

$$\log g(\psi;\varphi) = \log C + \log(N_\omega!) + \sum_{m=1}^{M} \{\log(x_m!) + x_m \log(p_m)\} + M\log(N_\omega!) + \sum_{m=1}^{M}\sum_{d=1}^{D} \left\{\log(x'_{m,d}!) + x'_{m,d}\log(p'_{m,d})\right\}.$$

The objective function of the estimation problem with constraint imposed on $\mathbf{P} = [\mathbf{p}, \mathbf{p}'_1, \ldots, \mathbf{p}'_M] \in [0,1]$ is then formulated as

$$\begin{aligned}\Lambda(\mathbf{P},\lambda) = &\sum_{s=1}^{S}\left\{\mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\le\gamma\right\}}\left(\log C + \log(N_\omega!) + \sum_{m=1}^{M}\left\{\log(x_m^{(s)}!) + x_m^{(s)}\log(p_m)\right\}\right.\right.\\ &\left.\left. + M\log(N_\omega!) + \sum_{m=1}^{M}\sum_{d=1}^{D}\left\{\log(x_{m,d}^{'(s)}!) + x_{m,d}^{'(s)}\log(p'_{m,d})\right\}.\right)\right\}\\ &+ \lambda\left(1 - \sum_{m=1}^{M} p_m\right) + \sum_{m=1}^{M}\lambda_m\left(1 - \sum_{d=1}^{D} p'_{m,d}\right). \end{aligned} \tag{7.25}$$

Consequently

$$\begin{cases} \frac{\partial\Lambda(\mathbf{P},\lambda)}{\partial p_1} = \sum_{s=1}^{S}\left\{\mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\le\gamma\right\}}\frac{x_1^{(s)}}{p_1}\right\} - \lambda = 0\\ \vdots\\ \frac{\partial\Lambda(\mathbf{P},\lambda)}{\partial p_M} = \sum_{s=1}^{S}\left\{\mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\le\gamma\right\}}\frac{x_M^{(s)}}{p_M}\right\} - \lambda = 0\\ 1 - \sum_{m=1}^{M} p_m = 0 \end{cases}$$

$$\Rightarrow \begin{cases} p_1^* = \frac{1}{\lambda}\sum_{s=1}^{S}\left\{\mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\le\gamma\right\}}x_1^{(s)}\right\}\\ \vdots\\ p_M^* = \frac{1}{\lambda}\sum_{s=1}^{S}\left\{\mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\le\gamma\right\}}x_M^{(s)}\right\}\\ \sum_{m=1}^{M} p_m = 1. \end{cases}$$

Since $\sum_{m=1}^{M} p_m = 1$ and $\sum_{m=1}^{M} x_m^{(s)} = N_\omega$

$$\frac{1}{\lambda}\sum_{s=1}^{S}\left\{\mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\le\gamma\right\}}\sum_{m=1}^{M} x_m^{(s)}\right\} = 1 \Rightarrow \lambda = N_\omega\sum_{s=1}^{S}\mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\le\gamma\right\}}$$

and finally

$$\hat{p}_m = \frac{\sum_{s=1}^{S}\left\{\mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\le\gamma\right\}}\frac{x_m^{(s)}}{N_\omega}\right\}}{\sum_{s=1}^{S}\mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\le\gamma\right\}}} \tag{7.26}$$

Following the same steps, then

$$\hat{p}'_{m,d} = \frac{\sum_{s=1}^{S} \left\{ \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)}) \leq \gamma\right\}} \frac{x'^{(s)}_{m,d}}{N_\tau} \right\}}{\sum_{s=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)}) \leq \gamma\right\}}} \tag{7.27}$$

As for the continuous case, the algorithm used to implement the CEM method with a discrete IS distribution is presented.

Algorithm 2: Random Partition via CEM for Discrete Optimisation

Input: Set $M, D, S > 0,\ N_\omega \geq M,\ N_\tau \geq D$;
Input: Set hyperparameters: $\rho > 0, \beta > 0$,
Input: Set initial parameters $\mathbf{p}^{[0]}, \mathbf{p}'^{[0]}_1, \ldots, \mathbf{p}'^{[0]}_M$
for $i > 0$ **do**

1. Generate S sets of realisations $[\mathbf{x}^{(s)[i]}, \mathbf{x}'^{(s)[i]}_1, \ldots, \mathbf{x}'^{(s)[i]}_M]$, $\mathbf{x}^{(s)[i]} \sim \pi\left((\mathbf{x}, \mathbf{p}^{[i]}\right),\ \mathbf{x}'^{(s)[i]}_m \sim \pi\left(\mathbf{x}'_m, \mathbf{p}'^{[i-1]}\right),$;
2. Calculate $\psi^{(s)[i]} = \left[\omega^{(s)[i]}_1, \ldots, \omega^{(s)[i]}_{M-1}, s^{(s)[i]}_{1,1}, \ldots, s^{(s)[i]}_{M,D-1}\right]$ $\omega^{(s)[i]}_m = \omega_0 + \Delta_\omega \sum_{m'=1}^{m} x^{(s)[i]}_{m'},\ s^{(s)[i]}_{m,d} = t_0 + \Delta_\tau \sum_{d'=1}^{d} x'^{(s)[i]}_{m,d'}$;
3. Calculate $KL(\hat{\pi}, \pi; \psi^{(s)[i]})$ for $s = 1, \ldots, S$ and specify $\gamma^{[i]}$ being $1-\rho$ empirical quantile of their values;
4. Calculate

$$\hat{p}_m = \frac{\sum_{s=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)[i]}) \leq \gamma^{[i]}\right\}} \frac{x^{(s)[i]}_m}{N_\omega}}{\sum_{s=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)[i]}) \leq \gamma^{[i]}\right\}}},\ \hat{p}'_{m,d} = \frac{\sum_{s=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)[i]}) \leq \gamma^{[i]}\right\}} \frac{x'^{(s)[i]}_{m,d}}{N_\tau}}{\sum_{s=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)[i]}) \leq \gamma^{[i]}\right\}}}$$

5. Smooth update of the parameters

$$p^{[i]}_m = \beta p^{[i-1]}_m + (1-\beta)\hat{p}_m,\ p'^{[i]}_{m,d} = \beta p'^{[i-1]}_{m,d} + (1-\beta)\hat{p}'_{m,d};$$

$i = i + 1$

until a convergence criterion is satisfied
After convergence, specify points of partition
$\omega_m = \omega_0 + |\mathcal{I}| \sum_{m'=1}^{m} p^{[i]}_{m'},\ s_{m,d} = t_0 + |\mathcal{T}| \sum_{d'=1}^{d} p'^{[i]}_{m,d'}$;.

It is important to note at this stage that, the optimisation problem solved in this thesis it is a constrained optimisation problem since the parameters that have to be estimated to define the optimal partition Π^* are constrained according to the restrictions given in (7.11). Hence, any solution which does not respect such a condition will be discarded. If one wanted to solve the uncostrained optimisation problem then the support of the random variables introduced includes zero, which may lead to the situation that some partitions are of zero length. If that happens, the breakpoints $\omega_1, \ldots, \omega_M$ and $s_{1,1}, \ldots, s_{M,D-1}$ may not form an increasing sequence. Consequently, they would not belong to the feasible set Ψ. To address this difficulty, two procedures could be considered

1. sample directly from the conditional distribution
$$X_1, \dots, X_M | X_1 \neq 0, \dots, X_M \neq 0$$
$$X'_{1,1}, \dots, X'_{M,D} | X'_{1,1} \neq 0, \dots, X'_{M,D} \neq 0.$$
2. sampling from the multinomial distribution and and force non zero realisation by removing any realisations that contain 0 entry to meet the conditions of the feasible set.

7.2.4 Some Toy Examples

In this section, two toy example are proposed to observe the performances of the Cross-Entropy technique. In each toy example, four instantaneous frequency functions are simulated, i.e. one will have $\omega_1(t), \omega_2(t), \omega_3(t), \omega_4(t)$. Two scenarios are proposed: the first one considers four constant instantaneous frequency which do not vary over time. The second will instead take into account frequency variation over time. Both examples consider a time domain of 10 seconds and will employ the case of the Multinomial distribution for the importance sampling distribution. The first case consider IFs which cover a frequency domain betwnee 0Hz and 30 Hz. The second example instead will be only limited between 0Hz and 10Hz.

By taking into account the first example, Figure 7.1 describes the simulated instantaneous frequencies $\omega_1(t)$, $\omega_2(t)$, $\omega_3(t)$, $\omega_4(t)$ in the first scenario. These represent four constant functions over time, located at different frequencies being 3Hz, 7Hz, 10Hz and 30Hz. The idea is to observe how the CEM will perform in the case of constant frequency functions and if it will be able to identify such a scenario. Remark that the final goal is being able to fit a GP on specific frequency bandwidths, which will be the ones relevant for solving the given task within the application of interest. Therefore, the CEM has to "isolate" and identify those frequency regions that are populated by IFs. The challenge in this example will be to identify that the functions are not time-varying and for the CEM to be efficient a high number of initial bandwidths must be selected. This means that the initial M should be a high number with respect to the frequency range 0-30Hz.

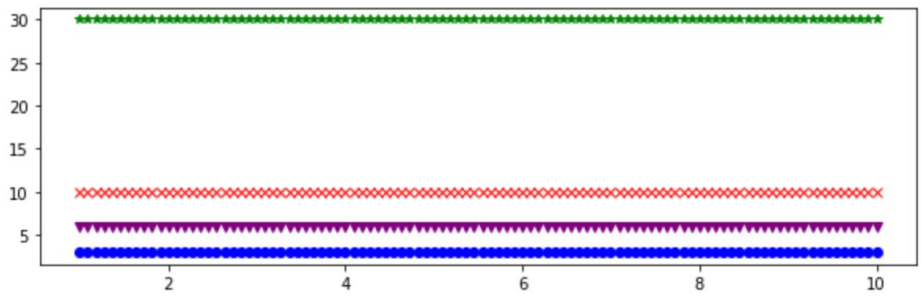


Figure 7.1: Simulated Instantaneous Frequencies $\omega_1(t), \omega_2(t), \omega_3(t), \omega_4(t)$ for the first scenario. The x-axis represents the time and the y-axis the frequency.

Afterwards, a grid for the initialisation of the cross-entropy is selected. In this case, $M = 10$ for the frequency axis (the y-axis) and $D = 10$ for the time axis (the x-axis). The choice of the initial number of rectangles that should be considered for the CEM is selected by the user. This is usually linked to the problem or the application of interest. In here, this example is shown only. However, multiple initialisations have been taken into account so for this example to work properly. Reasons behind that were that if M was small, then the CEM would not efficiently identify the different functions. With M big the computational cost could become too heavy. In this respect, this example appears to work efficiently and will be below presented.

The next step consists of implementing a kernel density estimator avoiding the problem of the estimated probability of some empirical partitions being infinity when there are no points present in those rectangle partitions. Figure 7.3 provides a display of the results of such estimator. It is possible to see that the provided estimates lie around the given frequencies and well perform.

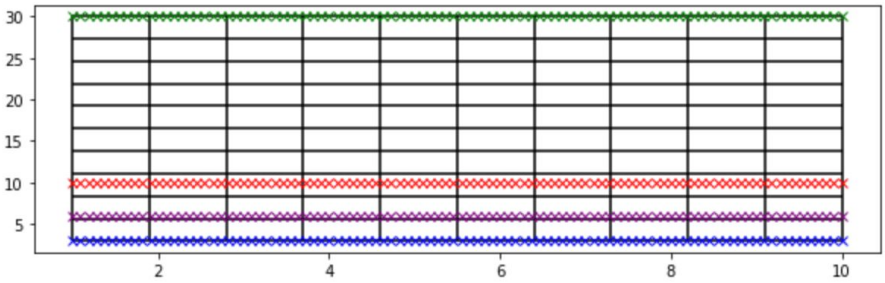


Figure 7.2: Initial partition Π for the first scenario. Note that $M = 10$ and $D = 10$.

Hence, the following step is to perform the cross-entropy method. The final resulting partition is provided in Figure 7.4. Note that such a final result has been identified in 7 steps of th CEM optimisazion technique.

The obtained partition shows that the four different instantaneous frequencies are well separated. Hence, the relevant partitions will be the horizontal ones. In this particular scenario where the challenge is having constant instantaneous frequencies, the objective of cross-entropy is to identify narrow bandwidths around such constant functions. Figure 7.4 show the obtained result. It is indeed possible to observe how the the cross-entropy method separates and achieves the desired results. One could argue that the bandwidths in between are empty and that the number of optimisation steps could be further reduced since the number of estimated parameters would be smaller. The main issue with that would be not being able to efficiently identify an optimal partition separating the different bandwidths since the number of pre-selected ones would be enough. However, more research is required in this regard with a computation of the optimal number of M and D with respect to the computational cost of the algorithm.

Figure 7.5 shows the second case of simulated instantaneous frequencies. This time, two constant functions are considered over time along with two time-

varying ones. Depending on the number of bandwidths pre-selected by the user, then instantaneous frequencies lying in the same bands will be considered as generated from to the same stochastic process .

The same procedure is followed as for the above toy example. The selected parameter for the number of required rectangles are $M = 4$ and $D = 4$ as given in Figure 7.6. In this case, the frequency axis varies between 0Hz and 10Hz, hence a smaller range compared to the one of the above example. Furthermore, the simulated number of IFs are indeed 4. Multiple solutions have been tried. The successful one is the one reported in this thesis. The initial selected number of bandwidths is $M = 4$ since the modelled IF are indeed four different functions. Refining the index D, i.e. increasing the number of rectangles with respect to time did not improve the final obtained partitions and, therefore, this has been kept with $D = 4$. Reducing it, however, provided poorer performance of the CEM. Further research is needed in this direction since the performances will be highly affected by the duration of the underlying IFs.

Figure 7.7 presents the kernel density estimator, which as for the previous case, provides a good representation of the underlying empirical instantaneous frequencies. This scenario is much more interesting since the variation of the underlying instantaneous frequencies is multiple and the kernel density estimator appear to work well for each of the frequency function.

Figure 7.8 presents the final optimal partition required to then construct the third system model. Note that this has been identified in 13 steps of the CEM. As shown, the cross-entropy algorithm captures the instantaneous frequencies falling in the same frequency bandwidth in a data-adaptive fashion. Indeed, close IFs or, in a better way, the ones that fall close to each other belong to the same frequency bandwidth. This is indeed the final goal of this methodology.

In the case of the speech applications, the CEM will perform well as shown by the resulting performances of the system model 3. It is important to highlight that, in this case, the CEM will consider the first three IMFs basis functions only. Reasons behind that is that, in speech analysis, the first three IMFs capture the majority of the formant frequencies, i.e. the frequencies at which the vocal folds vibrate, and identify biometric features required for the taks of interest (speech recognition, speech verification, etc.). Therefore, the CEM requires to aggregates IFs living in similar bandwidths which might be computed from different IMFs and so will efficiently compute the new Quasi-IMFs of the third system model.

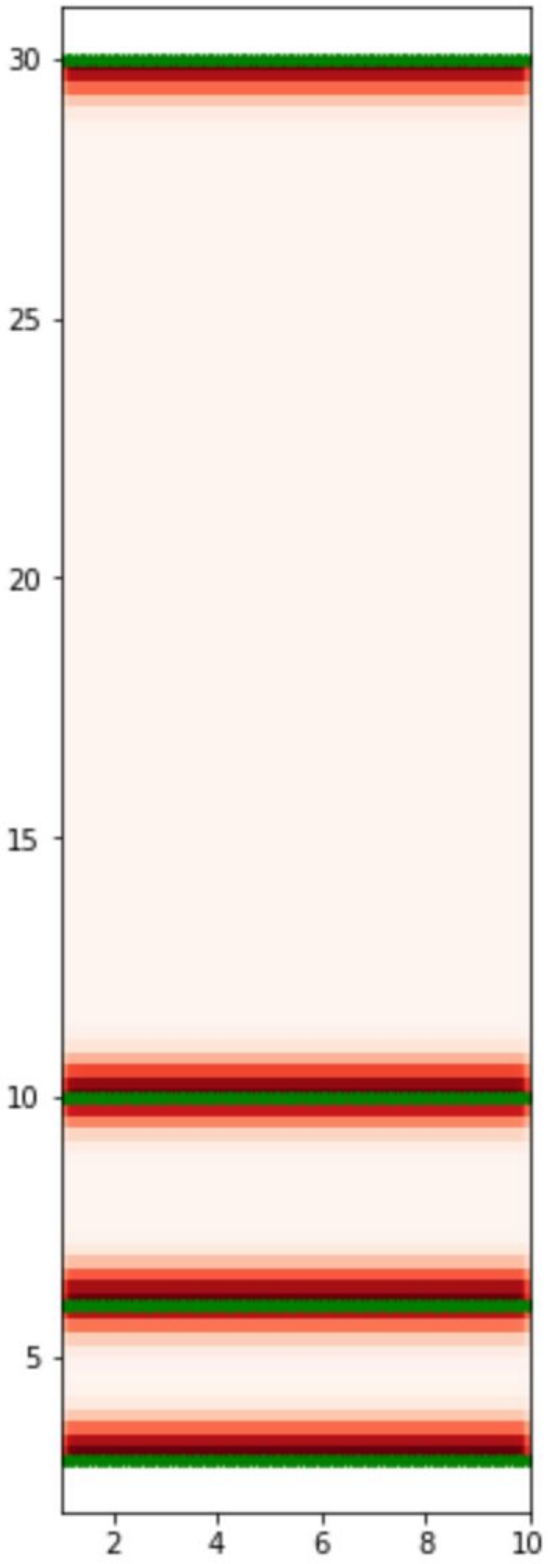


Figure 7.3: Kernel density estimator for the first scenario.

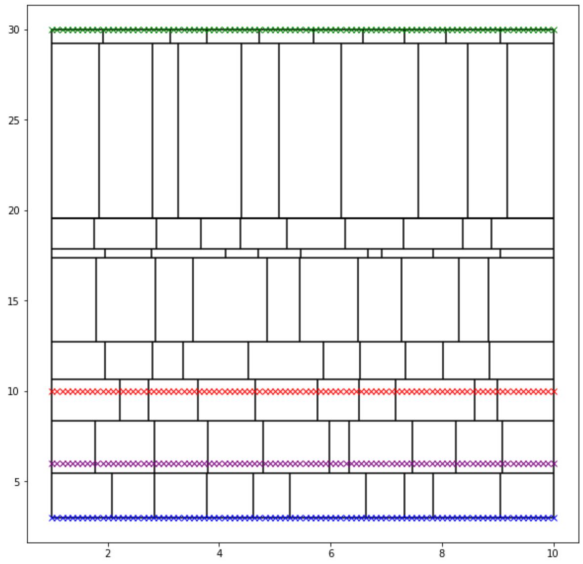


Figure 7.4: Optimal, final partition Π^* for the first scenario.

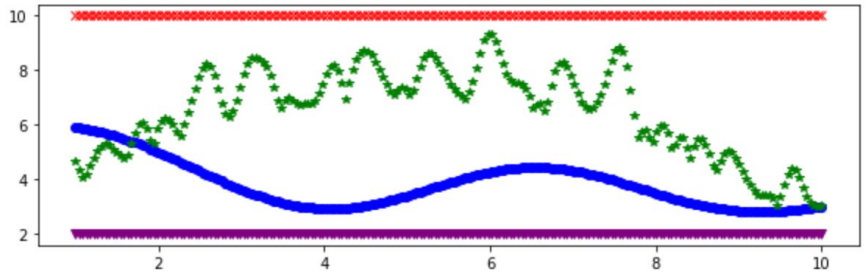


Figure 7.5: Simulated Instantaneous Frequencies $\omega_1(t), \omega_2(t), \omega_3(t), \omega_4(t)$ for the second scenario. The x-axis represents the time and the y-axis the frequency.

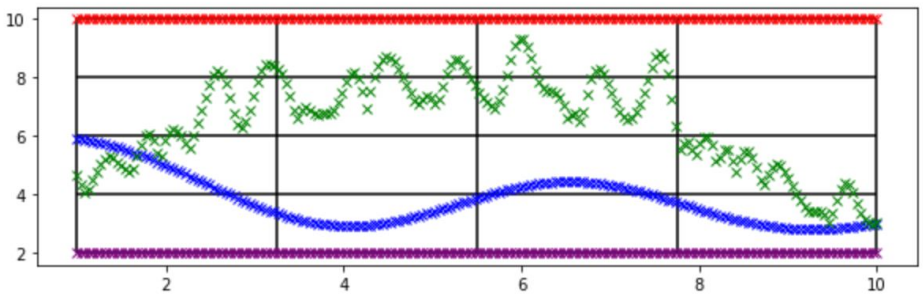


Figure 7.6: Initial partition Π for the second scenario. Note that $M = 4$ and $D = 4$

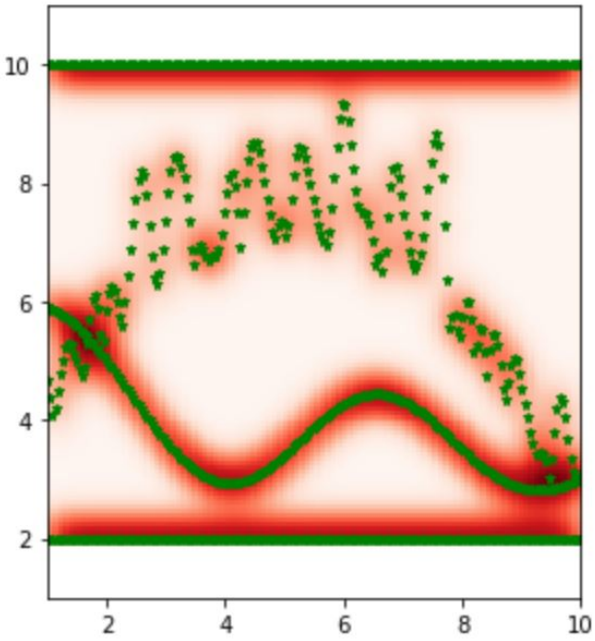


Figure 7.7: Kernel density estimator for the second scenario.

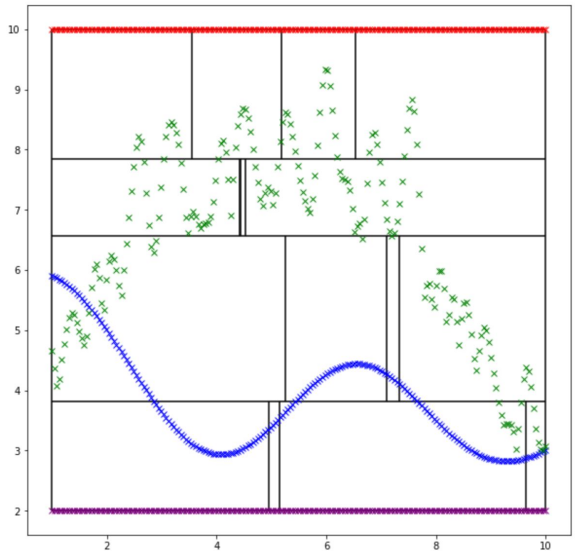


Figure 7.8: Optimal, final partition Π^* for the second scenario.

Part III

Speech applications

Chapter 8

A Cyber Security Application for Automatic Speech Verification

The prevalence of biometric authentication systems is increasing in many data access points in smart devices and remote data access settings. This has led to a new machine learning based approach to address the resulting biometric challenge of Automatic Speaker Verification (ASV). Modern machine learning approaches are recently tackling the study of ASV, see Faisal and Suyanto (2019) and Salah and Halim (2020). In this Chapter, a novel machine learning solution for ASV is explored and designed around feature extraction for speech signals; the challenge of biometric cyber-attack mitigation is addressed by seeking to detect when data access is attempted through a deep fake artificial speech generation rather than a human speaker. In the same vein as the biometric verification work for fingerprints of Kachiashvili and Prangishvili (2018), identification and verification speech biometrics will be performed.

The key statistical component of the proposed speech signature representation is based upon the non-stationary functional basis characterisation for speech signal via the Empirical Mode Decomposition. The EMD is used to identify which voice signal components provide discriminatory power in mitigating the risk associated with biometric cyber attacks in Automatic Speaker Verification technology (ASV) frameworks, where the extracted IMF basis functions act as an individual's vocal signature allowing for discrimination of the human voice from synthetic attacks using replicated artificial voice. The EMD has been employed within speech analysis in Sharma et al. (2017*a*); while Wu and Chen (2010) made use of the EMD for the noise-robustness of automatic speech recognition systems. Sethu et al. (2008) focuses on speech-based emotion classification utilising acoustic data and successfully employed the EMD basis functions and the instantaneous frequencies derived through the Hilbert transform. Furthermore, Schlotthauer et al. (2009) used the EMD algorithm to extract the fundamental frequency F_0.

ASV technologies are gaining widespread utilization in contexts of call centers, human-computer interfaces, secure access control for commercial and retail bank-

ing, see Sriskandaraja et al. (2016), and Wu et al. (2017). An ASV system typically extracts speaker characteristics from utterances and compares them to a given speaker synthetic voice model, estimated from its identity. In this context, one may distinguish between text-dependent and text-independent frameworks. The former uses a fixed collection of reference sentences, while the latter exploits purely arbitrarily selected speech utterances. These are usually referred to as Text-Dependent Speaker Verification systems, or TD-SV, versus Text-Independent systems, or TI-SV, see discussions in Sriskandaraja et al. (2016). A further differentiation might be given by speaker-dependent verification systems (SD-SV) or speaker-independent verification systems (SI-SV), where the former are trained by the individual who uses the system, while the latter trained as a system-agnostic to who is then using it. As with any biometric system, ASV is subject to spoofing or presentation attacks, which mimic a target speaker's voice in person or remotely via artificial tools such as voice conversion (VC) or speech synthesis (SS) algorithms. The study of such attacks is of growing significance in areas of the services industry, particularly the financial services sector, where clients' personal data access is increasingly reliant on biometric identification. Spoofing attacks on banking records may be classed as a form of cyber attack. The provided machine learning classifier solution of this Chapter seeks to detect and mitigate losses to data integrity and sensitive information by detecting and preventing such synthetic voice access attacks.

Consequently, a range of approaches is emerging to produce specific countermeasures to mitigate against different types of cyber spoofing attacks (see Wu et al. (2017), Patel and Patil (2017) for ASV and Kabir et al. (2021) for a survey focusing on speaker recognition presenting several countermeasures). The standard approach in many of these countermeasures is to identify speech parametrisations carrying discriminative power to differentiate between spoofed and real voices. The designed techniques make use of a classifier that attempts to distinguish between samples from two distinct populations of utterance, those from authentic voice and those from a synthetic generation of voice, derived from the two classes of speech signals Ramachandran et al. (2002). The raw speech time-domain signals are often transformed into lower-dimensional sets of summary statistics or engineered feature representations for such classifiers, see Wu, Evans, Kinnunen, Yamagishi, Alegre and Li (2015). Furthermore, such countermeasures often rely on standard time-frequency techniques constrained by assumptions such as stationarity or linearity of the underlying speech signal. The speech community has proposed multiple variations of these classical methods to overcome the aforementioned issues (see for example Fan and Hansen (2009), ur Rehman et al. (2017), Jeevan et al. (2017), Chakroborty and Saha (2009), Tapkir et al. (2018), Zouhir and Ouni (2016)) and so dealing with different aspects faced by ASV systems in discriminating spoofed and real voices. The traditional practice foresees the extraction or engineering of the raw speech data features and then conducts the classification task by stacking them within a vector. In this way, the classifier is often polluted by multi-frequency content information all contained in the proposed unique vector. The approach proposed in this Chapter aims to

tackle such a problem by constructing a parsimonious model that separates this frequency information content instead and selects the most discriminant areas of the time-frequency plane regarding the speech scenario analysed.

A recently developed approach dealing with ASV challenges is given by Deep Learning (DL). The reader should refer to Ohi et al. (2021) for an overview of deep learning based speaker recognition approaches that could also be extended to speaker verification tasks. These include multi-stage networks, end-to-end networks, generative networks or meta-learning. As highlighted in Ohi et al. (2021), these techniques are at a stage of minimal investigation with no asserted guidance on how to perform them efficiently and compare them to existing methodologies. Furthermore, DL requires, in general, high computational costs associated with big data training sets, often making them difficult to use in practice, and, therefore, further research is required to establish this direction. In the specific setting of ASV challenges, the idea behind DL methods, particularly the one of Deep Neural Network (DNN) quickly becoming the "new-state-of-the-art" methods, is to identify the formants structure with the complex function using many layers of perceptrons. This procedure is a high-cost learning procedure that will be replaced in this Chapter by the EMD technique, able to capture formant structure with the requirement of much fewer parameters and can be applied to small and large datasets providing a uniform method in this regard. Hence, a sparse architecture in the placement of DNNs is promoted by this work. Afterwards, a much simpler classifier relying on the recent method known as multi-kernel Learning (Gönen and Alpaydın (2011*b*)) combined with the Support Vector Machine is proposed.

Another critical aspect is that not only speech is highly non-stationary and non-linear per se, but when ASV challenges are solved, adverse environments might be the one of interest, making the task even more difficult. For example, the presence of noise affecting human speech during the recording or the need for a very long speech signal to be recorded by the user to train the system or reverberation affecting the system. These challenges are discussed in Jung et al. (2020), where the authors propose a method for short fragments of speech signals tackling the issues above described.

Given the great variety of approaches introduced in the literature and the several databases built and considered by researchers, it is hard to identify a uniform, standard technique unifying the presented framework and tackling the explained issues. The desired and sought technique should carry three main properties: first, non-stationarity and non-linearity should be heavily considered since speech is highly affected by these two characteristics. Often, real-world settings can be further corrupted by adverse environments such as noise, which can cause classical Fourier methods to fail to provide reliable and consistent results across different experiments and noise environments. Secondly, the discrimination power of the classifier should be the centre of attention, and new classification methodologies should be proposed and studied. Third, several benchmark ASV features have been proved to be successful in multiple cases. The focus should be on the

statistical interpretability of the ones able to identify discriminating insights in solving the ASV challenge.

The approach given in this Chapter follows along the familiar line of attack mitigation adopting a classifier framework, and the novelty lies in three components: the ability to treat the feature extraction in a non-stationary formulation; secondly, an ensemble learning multi-kernel classification framework is developed, see Gönen and Alpaydın (2011*b*); thirdly, the interpretability of the given feature extraction framework in terms of formant structures differentiating real and spoofed speech is provided. In this Chapter, it is demonstrated that it can improve the ability to detect attacks when compared to current state-of-the-art methods.

Furthermore, three datasets will be considered for the conducted experiments setting up multiple speech scenarios as text-dependent or text-independent and speaker-dependent and speaker-independent. Among these datasets, two of them are constructed explicitly by the author without a recording laboratory or particular microphones providing the setting encountered in ASV challenges of adverse environments. Therefore, the obtained results will provide robustness in these settings.

The Chapter relies on the framework developed in Chapter 5, making use of the Support Vector Machine (SVM) to solve the problem of biometric speech identification and spoofing detection of synthesised voice more effectively. In a speech environment, the Support Vector Machine has been largely used for speaker recognition and verification, see Campbell et al. (2006), Campbell (1997), Ramachandran et al. (2002), Jaakkola and Haussler (1999*a*). As highlighted in Kinnunen and Li (2010), the SVM is a discriminative classifier, which models the boundary between speakers, in speaker recognition, or speakers and impostors (related to any spoofing attack), in speaker verification. The final object of the learning process is minimising the classifier's probability of error, i.e. the probability of incorrectly labelling a sample point given that sample point and its label. Furthermore, the method known as multiple kernel learning (MKL) replacing single kernels for a combination of them will also be employed, providing outstanding performances in the task of interest. Such a technique has been introduced in Chapter 4, section 4.3. In speaker verification, Longworth and Gales (2008) apply this method through the study of dynamic kernels.

The first section of the Chapter presents the main contribution provided at a statistical level and within a speech framework. Afterwards, a statistical background for the general characterisation of speech signals, focusing on the engineered features and explaining their interpretation. The set of EMD features used to solve the classification task of interest is provided in Chapter 4, subsection 4.1.1. Upon these, and as above explained, a new set of features will be engineered combining the EMD basis and the Mel Frequency Cepstral Coefficients. Hence, the derivation of such a method is introduced in the following sections, and the new features, i.e. the EMD-MFCCs features, are then presented.

The final section of the Chapter presents the case studies conducted to identify the discriminatory power of the proposed features and methodology.

8.0.1 Contributions and Novelty

The contributions of this Chapter involve several core elements: firstly, enhanced non-stationary time-frequency methods applied to perform novel feature extraction techniques for the capture of speech signatures or vocal fingerprints. Secondly, using these new feature extraction methods to formulate a multi-kernel classifier based on Support Vector Machine techniques. This is highly beneficial in the classification tasks depending on the speech system considered: the extracted featured are often combined in a unique vector, and the SVM is then performed. Such a practice should be avoided since it will add noise to the classification problem mixing the formant structure depending on both the individual and the gender. Therefore, if the analysed scenario is text-dependent or text-independent or, for example, speaker-dependent or speaker-independent, the standard operation of considering a unique feature vector characterising the entire time-frequency plane would pollute the classification learning procedure. The third contribution foresees the performance comparisons between benchmark ASV features extracted on the raw data and on the EMD basis functions to highlight that speech is highly non-stationary and that multiple situations generate adverse environments that require the use of an adaptive method relying on the given data system. Afterwards, the proposed methodology is tested through the use of various TTS algorithms within different speech scenarios. To achieve such a goal, the following components has been developed:

1. The existing speech engineering techniques have been extended to non-stationary basis extraction methods and re-express them within a statistical framework. This is achieved via Empirical Mode Decomposition methods, which is used to extract time-domain intrinsic mode basis functions, represented via semi-parametric spline model characterizations.

2. The combination of time-domain non-stationary basis characterisation of the speech signals and the instantaneous frequency characterisations are combined to form a complete time-frequency signature of a person's vocal and speech characteristics. Such basis functions are more amenable to classical speech feature extraction methods in the transformed cepstral domain. This allows for the development of new approaches to EMD-Mel Cepstral speech signature characterisation that is highly effective in capturing individual speakers vocal tract specificities that arise given a speakers glottal airflow shaped by the vocal tract filter as it passes through it to produce speech. These features are then used to distinguish between real speech and artificial computer-generated spoofed synthetic speech by capturing these signature features.

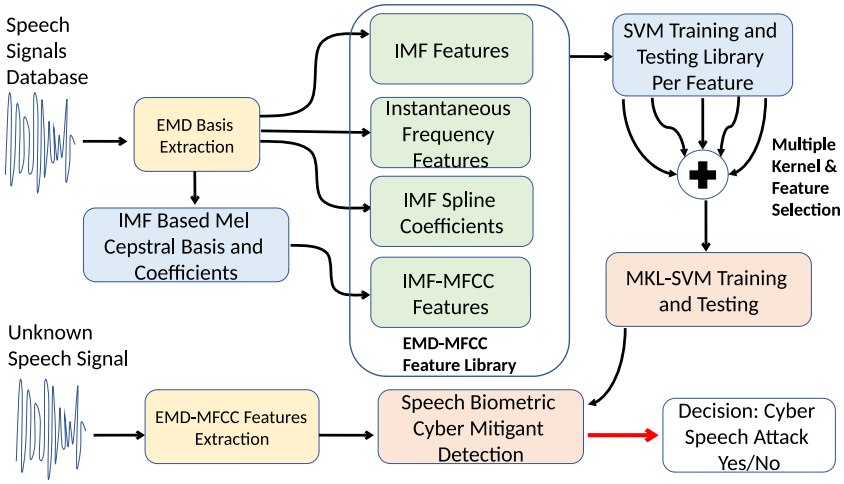


Figure 8.1: Proposed biometric speech cyber risk mitigation system.

3. The resulting speech signature feature characterisations allow to solve important new biometric tasks related to detecting cyber intrusion attempts to access biometrically multi-factor secured data or systems where speech is one of the security factors. A class of multi-kernel support vector machine classifier solutions has been developed to detect such cyber attacks, attempted through synthetically generated speech.

These contributions then form a complete system, summarised in Figure 8.1, for a cyber threat detection framework capable of accurately detecting synthetic spoofed voice attacks on a speech based biometric system secure access.

8.1 Background on Statistical Characterization of Speech Signals

According to the *source-filter model* Huang et al. (2001), a speech signal is the result of the *glottal airflow* shaped by the *vocal tract filter* as it passes through it Kinnunen and Alku (2009). Under such a representation, it is common to consider two main classes of features for an ASV system: voice source features or vocal tract features. The former are indeed related to the source of voiced sounds deriving from the glottal flow; however, numerous studies provide evidence showing that vocal folds features are not as discriminatory as vocal tract features Zheng et al. (2007). In this Chapter, therefore, the focus will be on the vocal tract features and, in particular, on representations that contain information about the resonance properties of the vocal tract, also known as *formants*. An individual's speech formant structures are analogous to that individual's speech fingerprint, thereby characterizing unique traits of the filter model specific to a human. Such features are, therefore, highly discriminatory, as it is challenging for a synthetic voice model to capture these individual-specific characteristics, see Kinnunen and Li (2010). Considering features that can capture information on

formant structures is crucial to mitigate biometric speech attacks on ASV-based security systems successfully.

In this Chapter, the use of non-stationary basis representations for speech enhances the ability to identify formant structures. This will be achieved by employing the EMD basis representations. A novel framework is developed to define adaptive non-stationary features which efficiently detect highly frequent temporal variations characterising the original speech time-series. The EMD features are further combined with MFCCs so that summaries of speech capturing intrinsic non-stationarity and formants structure are contemporaneously detected. Related approaches have been considered mixing these concepts; we cite amongst other Tapkir and Patil (2018) and Hasan and Hansen (2011). In the former, the authors propose the EMD as a dyadic filter in substitution to the mel-filter banks commonly used for the MFCCs. The extracted coefficients were therefore filtered according to the EMD basis. In Hasan and Hansen (2011), authors compute the MFCCs for the speech signal, and, after, the EMD is calculated for each coefficient. The approach of this thesis differs from both since the Mel Frequency Cepstral Coefficients are performed to represent the extracted non-stationary EMD basis themselves. The argument is that this will outperform alternative methods since it removes the requirement of local stationary assumptions that the methods mentioned above required for the first stage of the MFCC transforms. The traditional assumption made in speech is that speech signals should be approximately stationary at 25/30 milliseconds sampling rate under ideal background noise conditions. However, ASV systems would often operate within non-ideal environments affected by background noise or interference, which will be captured along with voices (see Mazaira-Fernandez et al. (2015) and Wu et al. (2008)). Instead, the EMD basis functions will accommodate non-stationarity of any level and so produce more robust features.

8.2 EMD-MFCC Speech Signatures via Pitch and Vocal Resonance

In speech analysis, the formant frequencies act like a characteristic signature of a given speakers vocal tract, like a speech fingerprint that is characteristic of given speakers vocal tract physiology, see Huber et al. (1999), Bashar et al. (2014). Formants are a concentration of speech acoustic energy, usually occurring at approximately each 1,000Hz frequency band, directly related to the oscillatory modes of resonance of an individual vocal tract structure. They are often indexed by F_1, F_2, F_3, etc., where F_0 is termed the *fundamental frequency* and represents the rate at which the vocal folds vibrate. This quantity corresponds to the pitch and coincides with the first harmonic, H_1; *harmonics* are multiple of the fundamental frequency F_0 characterizing the glottal source. Suppose one can extract these features from non-stationary voiced speech created by a human vocal, physical, physiological system. In that case, they may have the potential to be highly discriminatory factors to distinguish a human versus a synthetic

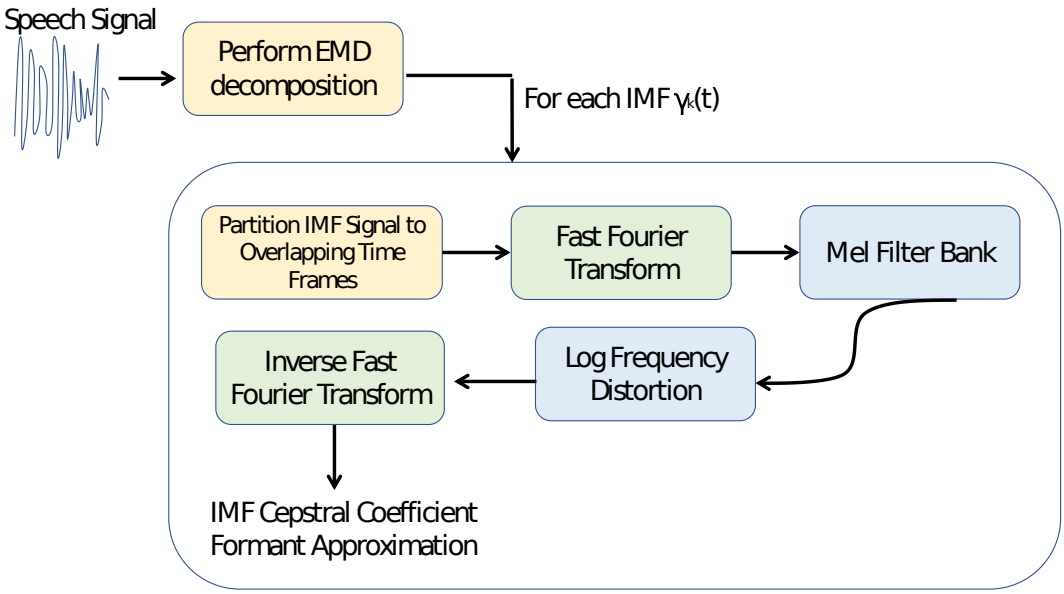


Figure 8.2: Diagram of the proposed methodology characterising EMD-MFCC features for formant detection.

voice as they represent how vocal tracts shape sound sources which therefore have representations unique to an individual.

These formant features are often approximated by a Mel Cepstral basis projection, where the functional coefficients form the MFCC representation of the speech signal that approximate the formants. Such a characterization is working well in capturing formant structure in ideal speech recording environments with sufficient sampling rates to capture local stationary approximations of the non-stationary speech signal. However, in real-world ASV systems that are considered in this thesis, speech is recorded in noisy real-world environments with more compressive sampling rates and background non-stationary noise and distortions. The presence of background noise and distortion have been shown in Mazaira-Fernandez et al. (2015) to render the MFCC estimated coefficients as highly sensitive and not statistically robust to a variety of potential types of background noise and distortions. Furthermore, the compression of the signal prior to transmission to the ASV for comparison in the biometric signal analysis can further create aliasing distortions.
These challenges will be overcome by merging EMD with MFCC, where rather than passing the raw speech signal into the MFCC representation, we will first decompose the speech signal into IMF basis representations, then MFCC representations of each IMF basis will be performed as illustrated in Figure 8.2. This can be shown to robustly estimate the formant structures even in the presence of different speech signal recording distortions and background noise environments.

There are existing works that have explored the development of EMD methods to characterize formant structures, see Bouzid and Ellouze (2004). However, as

explained in Sharma et al. (2017*b*) they suffer from an identification complication known as mode-mixing, which is the inability to align formant structures and IMFs. This occurs since these previous works apply the EMD method to signals already based on stationary Fourier transforms of the non-stationary speech signal. In this thesis, the problem of mode-mixing is avoided by first performing the EMD basis decomposition of the speech signal, then the Mel Frequency Cepstral Coefficients (MFCCs) of each IMF basis is studied. In this way, the formants can be exactly aligned with the ordering of the IMF bases represented through a second stage MFCC family of coefficient functions. The MFCC acts as a warped linear filter for each IMF expressed through a functional coefficient in time and fixed local frequency selective basis. The resulting coefficients of the filter will be non-linearly spaced in their spectral energy so that they can be estimated to align with standing wave patterns of pitch and harmonics of human speech formants.

The MFCC representation is defined as follows, starting from the base Mel-scale:

$$\phi = 2595 \log_{10} \left(1 + \frac{\breve{\omega}}{700} \right), \tag{8.1}$$

where ϕ is the subjective pitch in Mels corresponding to the original frequency $\breve{\omega}$ in Hz Sigurdsson et al. (2006). Consider the l-th IMF $\gamma_l(t)$ extracted from speech signal representation $\widetilde{s}(t)$. Next, a representation of the proposed EMD-MFCC characterisation is provided, followed by a brief numerically stable approximation that also works well in practice. Note that $\gamma_l(t)$ is pre-emphasised and Hamming-windowed, to get $\gamma_l^*(t)$ to guard against issues of aliasing in discrete sample MFCC representations of each IMF basis.

The continuous signal $\gamma_l^*(t)$ is then decimated to a set of T_s evaluated "sample" values in the local window frame. A discrete vector representation is then obtained $\boldsymbol{\gamma}_l^* = \left\{ \gamma_l^* \left(\frac{h\breve{\omega}_s}{T_s} \right) \right\}_{h=1}^{T_s-1}$ for $\breve{\omega}_s$ sampling frequency in Hz. Then perform the spectral transform of the l-th IMF representation $\boldsymbol{\gamma}_l^*$ to obtain local Fourier representation Γ_l^* given by DFT as:

$$\Gamma_l^*(h) = \sum_{n=0}^{T_s-1} \gamma_l^*(t) \exp\left(-\jmath 2\pi h n \breve{\omega}_s / T_s \right) \tag{8.2}$$

The magnitude of spectrum $\left| \Gamma_l^*(h) \right|$ is then scaled in both frequency and magnitude. The frequency is scaled through convolution with a linear Mel filter bank $H(h, m)$, a multiplicative transfer function in the frequency domain, and then the logarithm of the result is taken to stretch or time-dilate the resulting signal. The output of this process is a collection of functional Mel Cepstral Coefficients for the l-th IMF given in the frequency domain by,

$$\begin{aligned} \mathcal{M}_l(m) &= \log_{10} \left(\sum_{h=0}^{T_s-1} P M_l^*(h) \right) \\ &= \log_{10} \left(\sum_{h=0}^{T_s-1} |\Gamma_l^*(h)| \cdot H(h, m) \right) \end{aligned} \tag{8.3}$$

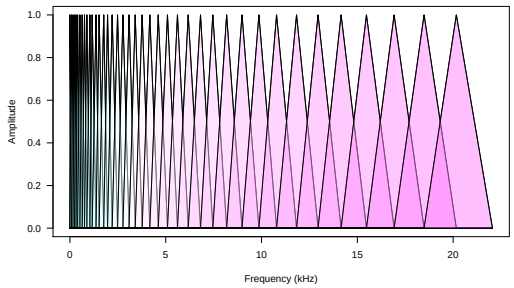


Figure 8.3: The Mel filter bank structure for 40 filters. Each peak represents the center frequency of the filters.

for $m = 1, 2, \ldots, M$, where M is the number of Mel bases used (or order of the filter bank). The Mel filter bank is a sequence of triangular basis defined by the center frequencies $\breve{\omega}_c(m)$ as follows:

$$H(h,m)=\begin{cases} 0 & \breve{\omega}(h) < \breve{\omega}_c(m-1), \\ \frac{\breve{\omega}(h)-\breve{\omega}_c(m-1)}{\breve{\omega}_c(m)-\breve{\omega}_c(m-1)} & \breve{\omega}_c(m-1) \leq \breve{\omega}(h) < \breve{\omega}_c(m), \\ \frac{\breve{\omega}(h)-\breve{\omega}_c(m+1)}{\breve{\omega}_c(m)-\breve{\omega}_c(m+1)} & \breve{\omega}_c(m) \leq \breve{\omega}(h) < \breve{\omega}_c(m+1), \\ 0 & \breve{\omega}(h) \geq \breve{\omega}_c(m+1), \end{cases} \tag{8.4}$$

which satisfies $\sum_{m=1}^{M} H(h,m) = 1$. The center frequencies of the basis are computed through equation (8.1) to approximate the Mel scale. Afterwards, a fixed frequency resolution of the Mel scale is computed, which is a logarithmic scaling of the repetition frequency, obtained by $\Delta\phi = (\phi_{max} - \phi_{min})/(M+1)$, where ϕ_{max} and ϕ_{min} are computed with equation (8.1) by using $\breve{\omega}_{max}$ and $\breve{\omega}_{min}$ respectively and M is the number of basis (filter banks). The center frequencies on the Mel scale are given by $\phi_c(m) = m \cdot \Delta\phi$ for $m = 1, 2, \ldots, M$. In order to obtain such center frequencies, the inverse of equation (8.1) is used and then they are substituted in 8.4 to obtain the Mel filter banks. The Mel Basis is illustrated in Figure 8.3 for 40 filter banks with a sampling frequency of 44.1 kHz giving 1102 samples, which is the one used in our real speech data case study. Note that the higher is the frequency, the wider the filter banks become.
The frequency domain Mel Cepstral functional coefficients are then converted back to the time domain via Discrete Fourier transform which simplifies to a simple Discrete Cosine Transform (DCT) of $\mathcal{M}_k(m)$ to obtain:

$$\mathfrak{m}_l(r) = \sum_{s=1}^{M} \mathcal{M}_l(s) \cos\left[r\frac{\pi}{M}\left(s - \frac{1}{2}\right)\right] \tag{8.5}$$

for $r = 1, 2, \ldots, M$, where $\mathfrak{m}_l(l)$ is the r-th MFCC of the l-th IMF.

Typical values for M in speech applications involve selecting the first 10-30 lowest center frequency cepstral coefficients. In this theses 12 coefficients (the lowest) are therefore retained to model the individual speakers and the synthetic voice.

8.3 Real data study: biometric security for synthetic vs real voice discrimination

In this section, the experiment carried to provide reliable features for ASV system in speaker verification subject to SS spoofing attacks is presented. The final purposes of this analysis: (1) studying the discriminatory power of EMD basis and features directly derived from this decomposition; (2) improving the EMD by combining it with Mel Frequency Cepstral Coefficients. The contribution of this action is to examine how each IMF captures resonance frequencies (formants) of the original speech signal. (3) Studying combinations of these features through a Multi Kernel Learning approach.

The first step in classification tasks using SVM consists of training the hyperparameters of the selected kernels through an in-sample analysis. The second part consists of an out-of-sample analysis to observe the prediction power of the same features. Note that three different datasets are employed in the experiments and will be below described. Appendix F presents some of the in-sample results. Particularly, the individual SVMs of the female and male voices making use of the first dataset of the statistics extracted on the IFs, the Spline Coefficients and the IMFs are given in Tables 1 and 2 of Appendix F, respectively. Furthermore, individual in-sample SVMs results of the IMFs, the Spline Coefficients and the IFs are also given for both voices, using dataset one, in Tables 4 and 5 of this appendix. Last, for dataset one only, the in-sample results of individual SVMs using the EMD-MFCCs features for both sets of voices are provided in Table 3 of Appendix F. Details about the out-of-sample results and Appendix G will be later provided throughout the description of the findings in this section.

The features employed in the experiments are described within the appendix D. Figure 1 of this appendix shows IMFs and IFs for the first dataset of one sentence randomly selected. Furthermore, Figure 2 and Figure 3 of the same appendix presents the results for the t-SNE (see Chapter 5, section 5.2 for details) applied on the statistics extracted on the IMFs and the Spline Coefficients of the IMFs for Speaker 1 and Speaker 2 versus the correspondent synthetic voices (male and female) using dataset one. An equivalent plot for the EMD-MFCCs feature is provided in sections below (see Figure 8.6).

The section is organised as follows: firstly, the experimental set up is described. There are three main experiments conducted to study the discrimination power of the presented methodology making use of three different datasets. Therefore, each experiment is presented. Note that, for the first experiment, a more detailed analysis is provided using wide-band spectrograms, further plots presenting the results and justifying the need for the EMD-MFCCs in this settings.

It is highly relevant at this stage to highlight the following discussion. The primary argument motivating the study of the combinations of EMD and MFCCs is the assumption of stationarity of speech signals. In general, speech is considered stationary at 25 milliseconds sampling rate. If this were the case, the application

of the MFCCs would efficiently capture the harmonics (Fourier basis) of that signal and would have equal discrimination of the cepstral coefficients applied to the IMFs (EMD-MFCCs features). These two methods are indeed able to well-capturing stationarity. However, if the stationarity assumption does not hold anymore, non-stationarity is likely to be non-uniform across all the frequency bands; it may be, for example, increasing, or decreasing, with the frequency range taken into account. Therefore, by merely assuming stationarity at 25 milliseconds rate, equal discrimination of MFCCs and the EMD-MFCCs features should be expected, given that the latter ones consist of a different formulation of a representation accommodating non-stationarity (the IMFs). In the presence of non-stationarity instead, the EMD-MFCCs should better perform. Nevertheless, this is strictly related to the speech signal taken into account. To further explore such a discussion, another experiment is conducted within this Chapter and explained as follows: (1) a bandpass filter is applied to each speech signal at several frequencies. (2) The frequency range taken into account for the bandpass filter corresponds to 1kHz range and is applied from 0 to 5kHz to select different formants. (3) The question at this stage is whether non-stationarity is more prevalent in specific bandwidths than other ones. What is expected from such an experiment is that low-frequency bandwidths would be less problematic in terms of relative performances between MFCCs and EMD-MFCCs features. Reasons behind this would be the presence of the fundamental frequency F_0 which correspond to a stationary component, and, therefore, would carry less discriminatory power. The majority of the difference with respect to performances of MFCCs versus EMD-MFCCs features is therefore expected to be at higher frequencies where is less likely to observe the same stationarity feature

8.3.1 Experimental set up

There are three classes of experiments shown in table 8.2 that foresee the use of three datasets, which are described in table 8.1. The three datasets are firstly described, and then the different types of experiments are presented.

Consider firstly dataset one and dataset two since they rely on two specific classes of sentences, respectively. These two sets are used to test the novel methodology within a text-dependent and a speaker-dependent verification system (TD-SD-SV) relevant to ASV challenges characterised by these conditions. The first dataset involves a set of sentences constructed to be challenging and reflect a real ASV setting in which sentences are not phonetically balanced. These are obtained them from the first text (Inferno) that makes up Dante Alighieri "The Divine Comedy". The second dataset is a reference set based on the IEEE Recommended Practices for Speech Quality Measurements, as described in of Electrical and Engineers (1969), extensively used in speech analysis testing of speaker verification. It sets out seventy-two lists of ten phrases described as the 1965 Revised List of Phonetically Balanced Sentences, otherwise known as the 'Harvard Sentences'. These are widely used in telecommunications, speech, and acoustics research, where standardised and repeatable speech sequences are needed.

Datasets Description

	Dataset	# of Speakers		# of Utterances		System
		Natural	Spoof	Training	Testing	
1	Not Phonetically Balanced	2	6	800	160	TD-SD-SV
2	Phonetically Balanced	2	6	576	576	TD-SD-SV
3	Subset of ASVspoof 2019	28	84	10160	2464	TI-SI-SV

Table 8.1: Description of the datasets employed throughout the various sets of experiments. The number of utterances for each speaker is balanced across each dataset. For example, in dataset one, training set, there are 800 utterances; given that the number of speakers is 8, this means 100 utterances per speaker. This is valid for every other set. For the classification tasks, gender has been taken into account. Hence, the speakers have been divided between male and female voices. The considered methodology aims to detect the energy concentration of the formant structure, which heavily differs amongst these two categories. Each dataset is further described within the text. The procedure applied to extract a subset of the ASVspoof 2019 challenge dataset is presented in 8.3.4.

In both datasets, two real-language sources were used from a female (speaker 1) and a male (speaker 2); for the synthetic speech, five correspondent sources (T1, T2, T3, T4, T5 described in table 8.8) were employed for the female case and one source (T1) for the male one. The synthetic speech voices of all TTS algorithms were selected to have an English accent. The voice recordings were sampled at 44.1kHz without significant channel or background noise to develop a text-dependent scenario relevant for speaker verification tasks Rosenberg (1992). Recording environments of both training and testing voice samples were identical to avoid mismatched conditions (see Ramachandran et al. (2002), and Rosenberg (1992)). Common sentences were used for each speaker and the synthetic voice.

Note that no recording laboratory or specialised microphone was used, and the utterances were recorded in noisy, reverberant environments. This is particularly relevant since it sets up the setting for adverse environments commonly encountered in ASV challenges. Therefore, the obtained results will carry the added feature of robustness to these kinds of speech settings.

The duration of each sentences speech recording was approximately 15sec to 1min maximum producing between 661k and 2,646k samples per spoken sentence. The start and end of each sample were trimmed to remove any non-speech segments and decimated to a set of 60k total samples. Regarding the IMFs extraction procedure, each set of 60k samples for one sentence was then windowed into non-overlapping collections of 5,000 samples and passed to the EMD sifting procedure. Afterwards, the features presented in Table 4.1 were extracted. Note that in some cases, for high-frequency instantaneous frequency features, it is advantageous also to apply a median filter (a window of 2ms was used).

In the first dataset, the total number of recorded sentences was 960, equally proportioned samples of the same sentences across all voice recordings, with 80% randomly selected for training and the rest for testing. In the second dataset, the first sentence from each of the seventy-two lists of the Harvard Sentences was used to construct the training dataset. The testing dataset was given by the second sentence from each of the seventy-two lists of the Harvard Sentences. This led to 1,152 utterances split equally between training and testing sets.

The third dataset corresponds to a subset of the ASVspoof 2019 challenge database described in Todisco et al. (2019). Details on the extracted sets of sentences are given in subsection 8.3.4. The importance of the settings provided by this dataset are remarked: they will tackle text-independent and speaker-independent verification systems (TI-SI-SV) to test the proposed novel methodology in the most general environment encountered in ASV challenges.

Table 8.2 presents the set of experiments considered. With experiment one, it is firstly presented a discussion showing how the EMD-MFCC approach provides more powerful discrimination in detecting individual vocal tracts required in ASV systems compare to other sets of EMD features (IMFs, IFs, Spline Coefficients, etc.). The benchmark model comparison of the traditional MFCC extraction on the raw speech signals is also provided and presented in table 8.4. Furthermore, given the wide variety of features often employed in Speaker Verification or Speaker Recognition tasks (see Sahidullah et al. (2015)), additional benchmark features applied both on the raw data and on the IMFs are performed. Table 8.3 provide a detailed description of such features, with the used configuration and references required for further understanding. Results of these features run on the IMFs are provided in table 8.5.Results for the individual speech features are discussed and then the EMD-MFCC-MKL framework is introduced.

Experiment two focuses on a different aspect often faced by ASV systems: the different TTS algorithms. Several techniques produce a spoofing attack: impersonation, synthetic speech or TTS, voice conversion, and replay. In this Chapter, TTS spoofing attacks are considered only. As highlighted in their work, Kamble et al. (2020) explains how TTS algorithms can nowadays produce high-quality voice through several kinds of methods as concatenative TTS unit selection Hunt and Black (1996), statistical parametric TTS Zen et al. (2009), formant synthesis Tabet and Boughazi (2011) and Deep Learning-based procedures (see Saito et al. (2017), van den Oord et al. (2016), Wang et al. (2017), Partila et al. (2020)). Each of these procedures carries specific pros and cons, highlighted in Figure 8.9. The best performing features for dataset one and dataset two for the female voice only obtained in Experiment one are selected an a similar exercise by considering the different TTS algorithms presented in table 8.8 is repeated. The best performance are provided while the additional results within the Supplement Materials.

Experiment three runs the EMD-MFCC-MKL solution for ASV systems on a selected subset of the ASVspoof 2019 challenge dataset. In this way, a text-

independent and speaker-independent environment is tested. Results will be presented for a range of different TTS algorithms and male and female voices. The focus will be on the best performing cases and the additional results are given in the Supplement Materials.

In each experiment, the focus is on presenting key aspects of the out-of-sample analysis that represent the most challenging cases for assessing the proposed EMD-MFCC methodology. All additional results are provided in the Supplement Materials, all code and data sets, including user guides, are provided at `https://github.com/mcampi111/Speech-Experiment`.

Experiments Description

Experiment	Objective	System	Dataset	Features			Techniques	TTS Algorithm	Gender
				Raw Data	IMFs	Others			
1 Formant Detection	Method Validation	TD-SD-SV	1	Benchmark	Benchmark	IFs, SCs, Cl. Stats.	EMD-SVM, EMD-MFCC-MKL-SVM	T1	M, F
			2	MFCCs	MFCCs	–	EMD-MFCC-MKL-SVM	T1	F
2 TTS Algorithms	Alternative Spoofing Attacks	TD-SD-SV	1,2	MFCCs	MFCCs	–	EMD-MFCC-SVM	T2, T3, T4, T5	F
3 Comparable Dataset	Existing Datasets	TI-SI-SV	3	MFCCs	MFCCs	–	EMD-MFCC-SVM	A01, A02, A04	M, F

Table 8.2: Table describing the three experiments conducted. Note that, in experiment one, both dataset one and dataset two are employed. Note that all the proposed sets of features have been extracted on dataset one and are widely discussed. For the second dataset, the MFCCs on the raw data and the EMD-MFCCs for the female voice only were considered. In experiment two, both datasets are used, and the MFCCs on the raw data and the IMFs basis functions are employed to assess the discrimination power of the EMD-MFCC-MKL-SVM in detecting different types of TTS algorithms. Experiment three provides results for the EMD-MFCC-MKL-SVM applied to a subset of the ASVspoof 2019 challenge dataset considering both the male and the female cases and multiple TTS algorithms. Details are provided within each section related to the different experiments.

Benchmark ASV Features							
Feature	**Acronym**	**Reference**	**Spectrum**	**Filter Type**	**Filterbank Shape**	**Filterbank Dimension**	**Compression/Other**
Mel-Freq. Cep. Coeff.	MFCCs	Thaine and Penn (2019)	Freq. Magn.	Mel-Scale	Triangular	40	Log.
Linear-Freq. Cep. Coeff.	LFCCs	Fan and Hansen (2009)	Freq. Magn.	Linear Freq. Scale	Triangular	40	Log.
Bark-Freq. Cep. Coeff.	BFCCs	ur Rehman et al. (2017)	Freq. Magn.	Bark-Scale	Trapezoidal	40	Log.
Gammatone-Freq. Cep. Coeff.	GFCCs	Jeevan et al. (2017)	Freq. Magn.	ERB Scale (Gammatone)	Approx. Log.	40	Cubic Root
Inverse Mel-Freq. Cep. Coeff.	IMFCCs	Chakroborty and Saha (2009)	Freq. Magn.	Inverted Mel-Scale	Triangular	40	Log.
Linear Predictive Cep. Coeff.	LPCCs	Kumar and Lahudkar (2015)	–	Linear Prediction	Linear	40	LP + Cep. Analysis
Magnitude-based Spectral Root Cep. Coeff.	MSRCCs	Tapkir et al. (2018)	Freq. Magn.	Mel-Scale	Triangular	40	Exponent α
Normalized Gammachirp Cep. Coeff.	NGCCs	Zouhir and Ouni (2016)	Freq. Magn.	Normalized Gammachirp	Triangular	40	Logarithm
Phase-based Spectral Root Cep. Coeff.	PSRCCs	Tapkir et al. (2018)	Freq. Phase	Mel-Scale	Triangular	40	Exponent α
Linear Predictive Coeff.	LPCs	Chougala and Kuntoji (2016)	–	Linear Prediction	Linear	26	LP Analysis
Perceptual Linear Prediction Coeff.	PLPs	Alam et al. (2013)	Freq. Magn.	Bark-Scale	Trapezoidal	26	LP + Cep. Analysis
Rasta Perceptual Linear Prediction Coeff.	RPLPs	Hermansky et al. (1991)	Freq. Magn.	Mel-scale	Triangular	26	LP + Cep. Analysis

Table 8.3: Table describing the selected benchmark ASV features extracted on the raw data and the IMFs for dataset 1. Note that results for the raw data are provided in table 8.4. The results of the IMFs are provided in table 8.5. The number of retained coefficients for every feature is 12. The pre-emphasis used for each feature corresponds to 0.97. When cepstral coefficients are computed, a window of 1024 samples is the length of the FFT, with an overlap of 128 samples, and hamming window is the one applied. Note that all the filters are filterbanks type except for the LPCs and the LPCCS. In these cases, no FFT and, hence, frequency magnitude is passed through the filter. Instead, after the preliminary phase, including pre-emphasis, framing and windowing, a digital all-pole filter is taken into account, and the autocorrelation method is employed to estimate the LPCs. For the LPCCs, a further step is taken to compute the cepstral coefficients directly from the LPCs in a recursive fashion. The reader might refer to Kabir et al. (2021) and Gulzar et al. (2014) for a more detailed description of such a procedure and the presented features. This is the conventional procedure also applied to PLPs and RPLPs; the last column of these two features shows LP + Cep. Analysis indeed, precisely referring to this process.

8.3.2 Experiment One: Biometric Cyber Risk Mitigation via Synthetic vs Real Voice Discrimination

Throughout these sections, the focus will be on the female voice examples to present the results.They generally presented the more challenging task in TD-SD-SV scenarios given the wider variation in spectral energy in the speech signals and higher non-stationary generally present in the formant structures in the 5kHz to 20kHz range. Note that results for the individual SVMs of the male voice are presented in the Supplement Materials.

Firstly, the ability of the benchmark ASV features to classify real and synthetic voices is considered. Such features for the female voice versus the synthetic voice generated with TTS algorithm T1 for dataset one are extracted. This is done firstly on the raw speech data, and then the features are extracted not on the raw speech but rather on the IMF basis function representations of the speech. This combined EMD-feature representation is advocated as a framework able to significantly enhance the discriminatory power of each of the familiar spectral, temporal speech features. Performances are improved universally by adopting this proposed approach of EMD-feature compared to just features on raw speech. Indeed, applying gold standard (Short-Time Scale Discrete Fourier Transform) ST-DFT based feature on the EMD functions produces greater discriminatory power since the EMD non-stationary bases are better adapted to the speech recording environment and the non-stationary nature of the speech signal. Amongst the selected benchmark ASV features, the MFCCs are the best performing and hence selected those to construct the new methodology combining EMD-MFCCs. Further evidence is provided by showing how this method better captures the formant structure of a given speaker through spectrograms, and other plots below presented.

Hence, the baseline reference to the proposed methodology will be the benchmark ASV features and, in particular, the MFCCS constructed from the raw speech data. This contrasts with the proposed methodology of first extracting the IMF bases and applying MFCC to each IMF to produce more significant discrimination. The argument is that the non-stationarity of higher frequency components in speech is more pronounced than lower-frequency components. Consequently, low-frequency bandwidths should be more comparable in terms of relative performances between MFCCs and EMD-MFCCs features. At these frequencies, the fundamental frequency F_0 more closely reflects a stationary component, and, therefore, MFCCs should be equally performing over either method. The majority of the difference is expected at higher frequencies, where it is more likely that non-stationarity will be non-uniformly distributed. It is important to highlight that, in this first experiment, the setting foresees a text-dependent and a speaker-dependent scenario. Hence, only one speaker at a time is considered for the classification task, and speakers use the same sentences. This is highly relevant since when multiple speakers and utterances are considered, the expected results will change. This will be presented in experiment three.

SVM with Speech Benchmark ASV Features

The first results considered are the benchmark comparison, based on applying the benchmark ASV features to the raw speech signal. Table 8.4 presents the results. Note that, for this task, the focus is on the female voice discrimination task with TTS algorithm T1 for dataset one. The configuration applied to obtain such coefficients are presented in table 8.3. One SVM per individual coefficient is performed. $M = 12$ is selected, as is the standard recommendation when utilizing these features in speech analysis. Results for the radial basis function kernel are presented. Other kernels have been employed and produced similar results.

The features are divided into cepstral coefficients or linear prediction coefficients, and they use various filter banks when the former are considered, or different transforms are applied once the linear prediction coefficients are obtained when it comes to the latter ones. These variants find their roots within different purposes. MFCCs, for example, try to capture formants by mimicking the human cochlear auditory capacity; the LFCCs are a similar feature, making use of a linear filter bank rather than the mel-filter bank to obtain a higher frequency resolution at high frequency (Campbell et al. (2018)). BFCCs represents an alternative to MFCCs (Shannon and Paliwal (2003)) whose filter banks should replicate the basilar membrane placed inside the cochlea that contains sensory receptors for hearing and performs spectral analysis for speech intelligibility perception. GFCCs (Liu (2018)) make use of the Gammatone filter bank for their cepstral analysis and model physiological changes in the inner ear and external, middle ear. IMFCCs consider the inverted-mel-filter bank and give a high-frequency resolution to low frequencies rather than high frequencies. We also consider MSRCCs and PSRCCs proposed in Tapkir et al. (2018) whose final goal is to model the human auditory system by the functional relationship between the onset firing rate of auditory neurons and sound pressure level. The former ones capture information about the magnitude spectrum while the latter ones about the phase spectrum. The NGCCs (Zouhir and Ouni (2016)) use a Normalized Gammachirp filter bank and incorporate the properties of the peripheral auditory system aimed to improve robustness in noisy speech settings. Another class of proposed features are the linear prediction coefficients and variations as the LPCCs, the PLPs and the RPLPs. These features rely on the stationarity of the underlying system, and even framing the speech signal into batches at which it turns stationary does not tackle the issue, especially when adverse environments are present. The highest accuracy is achieved by the NGCCs and the PLPs with an accuracy score of 0.850. Next, an equivalent procedure is performed, and these features are extracted on the IMF basis functions of the correspondent dataset. Results are in table 8.5. The proposed methodology relies on this new approach for which, instead of the raw speech, each IMF basis is passed one by one through an individual transformation of the selected features (i.e. MFCCs, LFCCs, BFCCs, etc.) to form adaptive features for the classification of real and synthetic voice. The standard practice of classification problems for this kind

of setting is constructing a vector collecting all the coefficients for the feature of interest or multiple features and then carrying the learning procedure. Since voice is highly biometric, highly non-stationary and adverse environments might arise, such standard procedures tend to create noise in the classification tasks rather than provide discriminant information. Therefore, the idea is to partition the time-frequency plane through a non-stationary and non-linear decomposition method and quantify energy generated by the formant structure. Furthermore, depending on the targeted task, i.e. TD-SD-SV or TI-SI-SV, the discriminant areas might differ given the use of the same utterances or not or the presence of multiple speakers or not or the consideration of gender. All features (except for the LPCs and the PSRCCs) achieve higher accuracy scores on the IMFs, particularly on the highest bases as IMF1 or IMF2, suggesting higher formants of the female speaker voices in a TD-SD-SV environment should provide most of the discriminant power. The MFCCs and the IMFCCs gave the highest accuracy scores. Given these performances, their interpretability and their wide use within SV tasks, we selected the MFCCs to construct new features combined with the EMD. Hence, the focus is on the individual speech features MFCCs extracted on the IMFs and will be further discussed in the following sections. They will be used to construct the EMD-MFCC MKL. Before that, further evidence is provided to show how the EMD-MFCCs better capture formant structures compared to standard MFCCs on the raw speech data.

Formant Detection for Real and Synthetic Voice

In Figure 8.4 the wide-band spectrograms are provided to visualise the formant structure of a given speech signal. The four panels represent the same sentence for Speaker 1, Speaker 2, and the female synthetic and the male synthetic voices. Each spectrogram has been performed on a window of 1024 samples (corresponding approximately to 23 milliseconds), with an overlap of 128 samples, the same pre-emphasis factor and windowing applied for the MFCCs (0.97 and hamming window), a dynamic range of 50dB and frequency range set of 0-10 kHz so that five formants should be visible (one at around each 1 kHz spaced carrier frequency). In Huang et al. (2001) it is noted that the first five formants are the ones necessary for speaker verification. Black lines highlight the five detected formants over time in each sub-figure, which line up with the EMD decomposed IMFs after transformation to IFs.

The top panel corresponds to the female speaker. The first four formants are within 0-5 kHz. This confirms that female speakers tend to have higher formant frequencies due to smaller vocal tracts (see Huber et al. (1999)) and a higher fundamental frequency F_0 compared to males. The second panel shows five formants in the interval 0-5 kHz, typical for a male voice. Furthermore, a lower fundamental frequency generates a smaller interval between voice harmonics resulting in a strengthened formants definition. This shows that for male voice EMD decomposition versus female voice, the IMFs obtained will have energy concentration in different spectrum regions. Consequently, the resulting EMD-

MFCC coefficients, if the Mel Cepstrum bases are kept constant in both cases, will have the coefficients for lower order IMFs being more influential than higher order IMFs. The opposite will be valid for the female voice. Note how the first two spectrograms show how human voices enunciate individual words much more than the last two spectrograms, where the separation between them seems dissipated. Furthermore, the formant structures referring to female voices (the first and the third plots) appear to behave much more alike than those characterising the male voices (the second and the fourth plots). This fact strongly depends on the synthetic voice generation algorithm, which will spread energy across a significant range of frequencies even by synthesising a male voice. Such a fact will result in a less challenging task for detecting synthetic and real male voices than the female case, and it justifies the choice to focus on the female case.

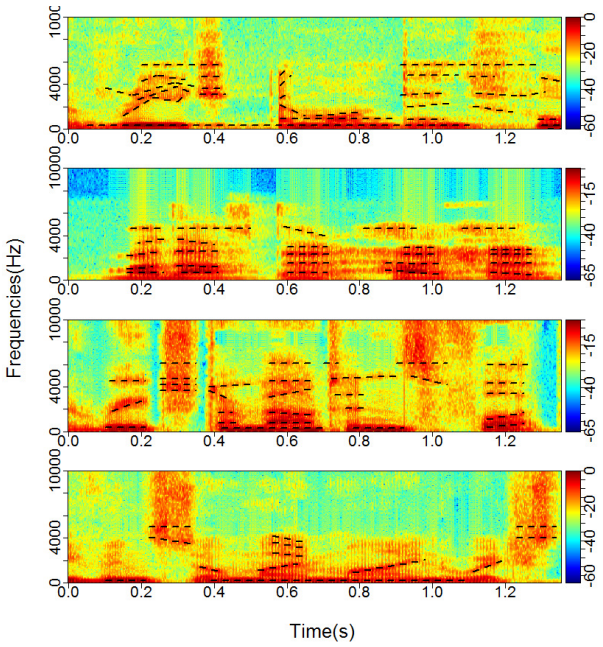


Figure 8.4: Spectrograms of the same sentence for Speaker 1 (top panel), Speaker 2 (second panel), the synthetic female voice (third panel) and the synthetic male voice (bottom panel). Black lines represent formants aimed to be detected by the IMF-Mel Cepstral basis representations. Colour scale in dB.

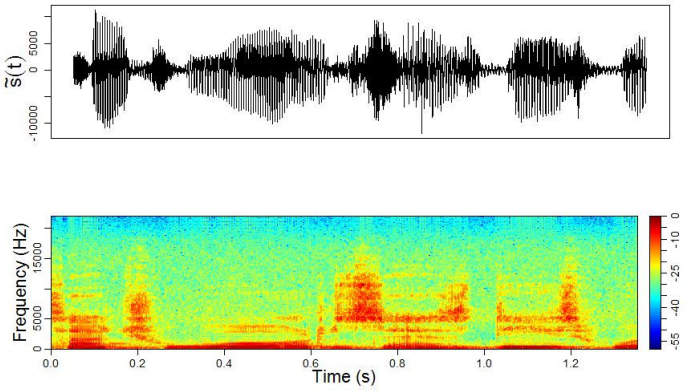
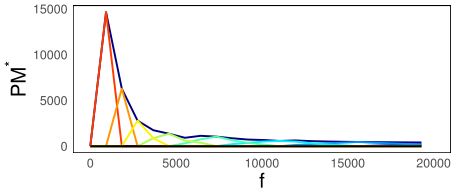
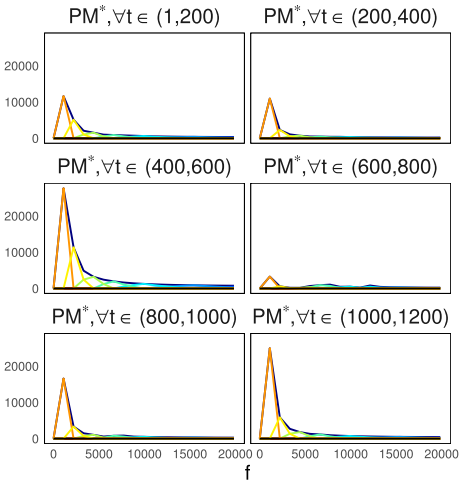


Figure 8.5: The top panel show one of the original sentences considered for Speaker 1; the bottom panel presents its related spectrogram. Colour scale in dB. The sentence corresponds to "When halfway through the journey of our life ".

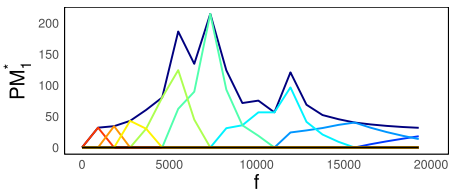
Next, to illustrate the EMD-MFCC method versus the classical MFCC on the raw speech, a sentence randomly from the real female voice recordings is selected, and the speech signal in the time domain and the spectrogram in Figure 8.5 are presented.

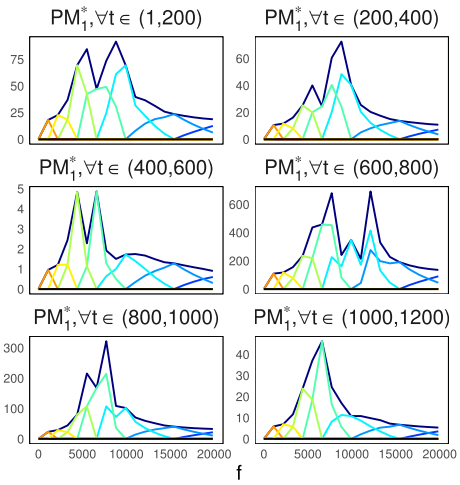
Then the results of the $PM_l^*(h) = |\Gamma_l^*(h)| \cdot H(h,m)$ are plotted and represent the PSD weighted Mel Cepstral bases for indexes $m \in \{1,\ldots,12\}$ in Figure 8.7. The classical situation in which one applies the MFCC directly to the speech signal (Figure 8.7, panel a) is compared to the cases in which the MFCC are instead applied to each IMF individually, precisely the first three IMFs (Figure 8.7, panels b,c,d). In each case, this is done over the entire time interval of the recording, followed by a sequence of local MFCC applications on 200ms windows with no overlap. This demonstrates that the resulting MFCC summary features captured by $PM_l^*(h)$ applied to the raw data signal are not overly responsive, although, in adjacent 200ms windows, there are significant differences in the spectrogram energy signatures, as also demonstrated in Figure 8.5.



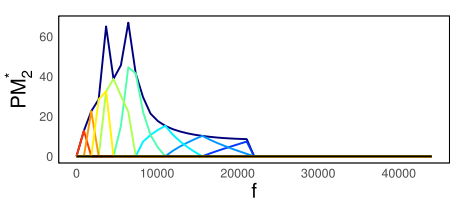


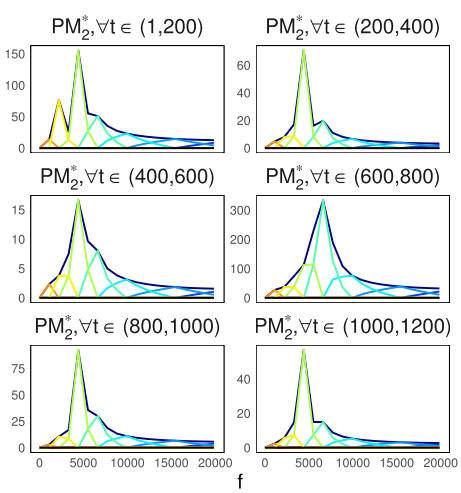
(a) PM^* computed on one of the original sentences $\tilde{s}(t)$ and PM^* computed over batches of $\tilde{s}(t)$.



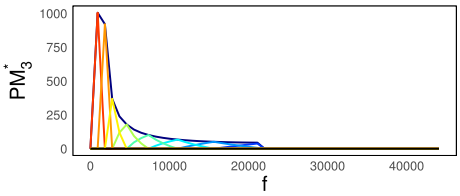


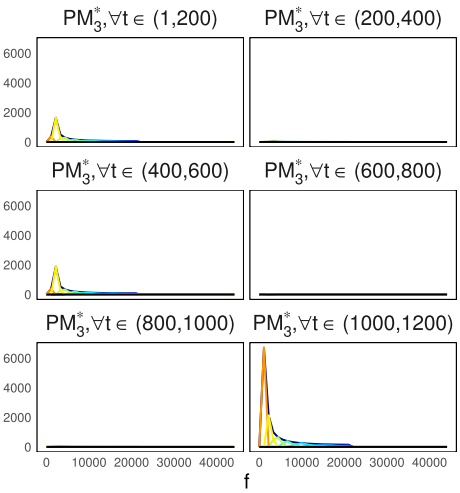
(b) PM_1^* computed on $\gamma_1(t)$ and PM_1^* computed over batches of $\gamma_1(t)$.





(c) PM_2^* computed on $\gamma_2(t)$ and PM_2^* computed over batches of $\gamma_2(t)$.





(d) PM_3^* computed on $\gamma_3(t)$ and PM_3^* computed over batches of $\gamma_3(t)$.

Figure 8.7: This Figure shows four panels. By looking at panel (a), seven subplots can be found. The first and biggest subplot represents the PM^* component of the MFCC decomposition presented in Eqn. 8.3 for one of the original speech signals of Speaker 1 (the female voice). Afterwards, the same quantity is extracted over batches of t as shown in the subfigures below such biggest plot. Panel (b), (c) and (d) take instead into account the correspondent PM_1^*, PM_2^* and PM_3^* components of the MFCC decomposition of $\gamma_1(t)$, $\gamma_2(t)$ and $\gamma_3(t)$, i.e. the first, the second and the third IMFs respectively of the original speech signal considered in panel (a). The time unit of the batches is in ms, and the frequency on the x-axis is in Hz. The y-axes of PM_1^*, PM_2^* and PM_3^* differ from the y-axis of PM^* since the IMFs do not include the residual or tendency.

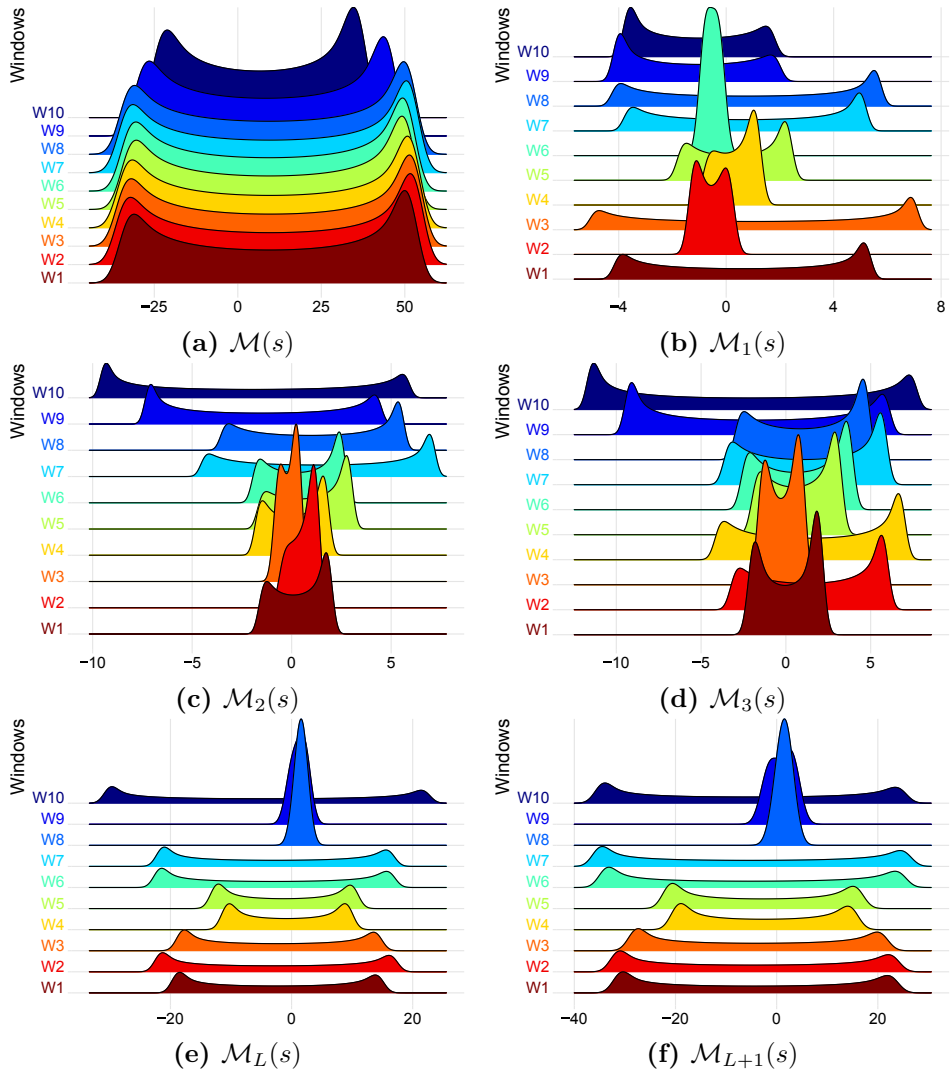


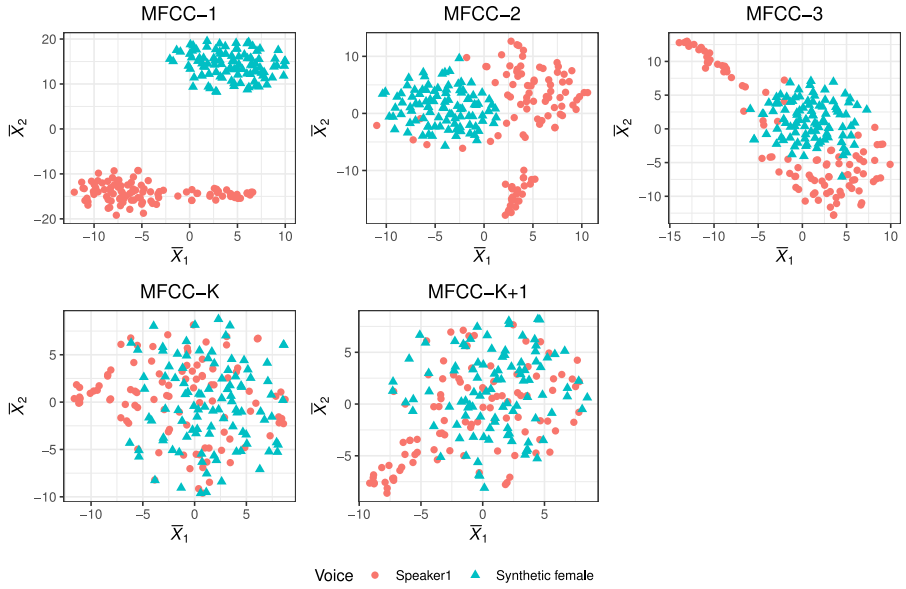
Figure 8.6: The panels represent the coefficient functions $\mathcal{M}_k(s)$ given in Eqn. 8.3 computed on a sliding window for one sentence of Speaker 1. Note that panel (a) refers to the original signal, and the correspondent quantity is denoted as $\mathcal{M}(s)$ with no sub-index. We split the sentence 200ms windows and calculated $\mathcal{M}(s)$ for every window.The procedure is then repeated on the IMFs basis of the same sentence of Speaker 1 and showed the results in the remaining panels obtaining $\mathcal{M}_1(s)$, $\mathcal{M}_2(s)$, $\mathcal{M}_3(s)$, $\mathcal{M}_L(s)$ and $\mathcal{M}_{L+1}(s)$. Remark that K is the last IMF and, in this specific case, equals 14 and $L+1$ corresponds to the residual. The different colours denote the associated window over which the extraction has occurred. Remark that the x-axes differ amongst the panels since the IMFs do not take into account the residual.

Following this discussion, the proposed methodology considers the EMD decomposition followed by the MFCC. The MFCC decomposition is performed under the same set-up for each IMF basis extracted, first for the entire IMF signal, then on successive 200ms windows. This analysis shows that since the IMF-MFCC features adapt to local non-stationarity better than the ST-DFT MFCC analysis, it is possible to capture more responsively the energy variation

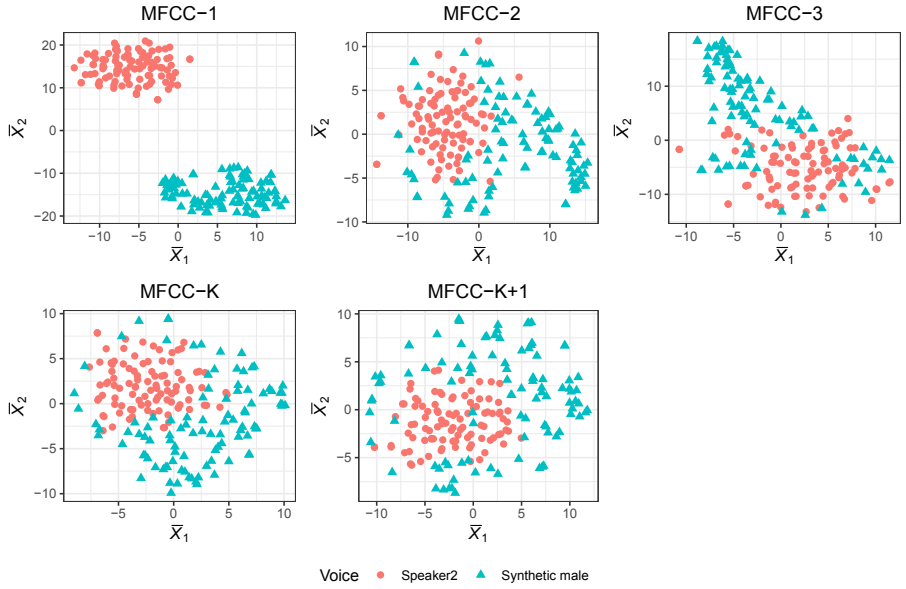
in bands of the formants to a greater degree. The subsequent classification of the biometric speech attack analysis leads to demonstrably better performance in the proposed method over the state of the art methods.

Figure 8.6 further presents the MFCC $\mathcal{M}(s)$ coefficients of the original signal and the EMD-MFCC $\mathcal{M}_l(s)$ coefficients for each IMF (see Eqn. 17). The entire time-domain signal is split into 200ms windows. The coefficient weight function variation of the MFCC is shown, and, importantly, its improved responsivity and selectivity for formants based on the IMF-MFCC framework are also presented.

Figure 8.8 shows the IMF-MFCC discriminatory potential of features between real and synthetic voice via application of the t-SNE projection method, see details in Maaten and Hinton (2008). The details of the t-SNE technique and its utilisation within the case analysis are summarised in the Supplement Materials. The plots demonstrate the discriminatory power of the IMF-MFCC coefficient representations when applied on local windows of length 50ms, producing features vectors in dimension $d = 1068$, after decimation for dimension reduction. One can see that there is evident potential for these IMF-MFCC features to have strong discriminatory power in all IMFs for the male case and hin several IMFs for female ones. As expected, the female lower frequency IMF-MFCC features have less discriminatory power than the higher frequency signatures, and the IMF-MFCC captures this clearly in all sentences as discriminatory between the real and synthetic voice. This indicates that the IMF-MFCC should act very well as spectral signatures to capture an individual's particular vocal tract structure and therefore have a solid performance to mitigate attacks from the synthetic voice.



(a)



(b)

Figure 8.8: Results of t-SNE for the MFCCs of Speaker 1 (top panels) and Speaker 2 (bottom panels). Note that the t-SNE algorithm is presented in the Supplement Materials. For each speaker, five sub-plots are provided related to each IMF taken into account. A PCA step was applied to reduce the initial data dimensionality, 90% of explained variation was retained. The axes represent the two dimensions identified by the t-SNE algorithm denoted as $\bar{X}_1$ and $\bar{X}_2$.

SVM-Speech Feature Library Construction: Classification Performance for Individual Speech Features

Note that the focus is on the female voice examples to present the main findings, whereas similar results for the male voice are provided in the supplementary appendix. In table 8.4, it can be seen that applying MFCC to raw speech produces an accuracy of discrimination between real female voice and synthetic female attack spoofed voices, out of sample for the same sentences, which did not exceed 77.5%. This type of accuracy score is often not acceptable for real-world applications where sensitive private data is seeking to be accessed via voice biometrics.

Experiment 1												
Dataset 1												
OFS Benchmark ASV Features - Raw Data - Dataset 1 - Speaker 1 vs Female Synthetic Voice TTS T1												
Coeff. number	MFCCs	LFCCs	BFCCs	GFCCs	IMFCCs	LPCs	LPCCs	MSRCCs	NGCCs	PLPs	PSRCs	RPLPs
1	0.775	0.500	0.500	0.425	0.500	0.425	0.750	0.500	0.500	0.700	0.525	0.800
2	0.675	0.500	0.575	0.725	0.775	0.750	0.700	0.350	0.700	0.700	0.550	0.750
3	0.550	0.500	0.550	0.775	0.400	0.725	0.725	0.500	0.725	0.725	0.650	0.725
4	0.475	0.500	0.700	0.525	0.075	0.725	0.700	0.475	0.850	0.850	0.375	0.450
5	0.475	0.500	0.650	0.475	0.775	0.425	0.775	0.500	0.775	0.775	0.450	0.450
6	0.650	0.500	0.700	0.775	0.725	0.325	0.700	0.450	0.700	0.450	0.425	0.475
7	0.150	0.475	0.700	0.725	0.725	0.325	0.700	0.500	0.750	0.700	0.525	0.700
8	0.375	0.500	0.700	0.525	0.725	0.700	0.675	0.425	0.800	0.700	0.475	0.675
9	0.375	0.500	0.775	0.725	0.750	0.725	0.225	0.500	0.750	0.400	0.650	0.700
10	0.675	0.550	0.825	0.800	0.700	0.725	0.700	0.500	0.750	0.750	0.450	0.475
11	0.770	0.500	0.700	0.800	0.775	0.725	0.750	0.500	0.775	0.500	0.225	0.500
12	0.775	0.500	0.700	0.725	0.750	0.725	0.725	0.500	0.775	0.500	0.550	0.500

Table 8.4: Out-of-sample results of the SVMs carried with the standard features used in ASV tasks applied to the raw data. The features description is given in table 8.3. Equivalent results for these features applied to the IMFs are provided in table 8.5. Note that each value corresponds to the accuracy achieved by the SVM carried with the coefficient given in the row of the feature given in the column.

The benchmark MFCCs on raw speech is further compared to two sets of features individually trained and tested. The first set corresponds to summary statistics obtained from the EMD applied to the raw speech signal. This produces the summary statistics of the IMF bases, the Instantaneous Frequency signals and the spline coefficients that characterize the IMF bases. These three signals are summarised using the summary statistics described in Table 4.1. These results are in Tables 1 and 2 in the Supplement Materials and demonstrate the out-of-sample classification results for dataset one for Speaker 1 versus synthetic voice attacker and Speaker 2 versus synthetic voice attacker.

One SVM training and then out-of-sample testing per feature component are performed, where, for instance, each sentence was taken, and then each IMF was taken. After, given each IMF, the summary statistics were extracted, and then the SVM training and out-of-sample testing were run for various kernel families. This allows building a library of individual features and their performance in

the real vs synthetic voice discrimination over the voice recordings database. It forms the basis for the multiple kernel learning framework that ultimately creates our proposed EMD-MFCC multi-kernel classification solution. All performances greater than 90% accuracy are bolded when presenting the results, which is a realistic minimum accuracy required for many real-world biometric applications. In general, individual features from the summary statistics of the IMFs and IFs (for a range of kernel choices) outperform the standard comparison of MFCC applied to raw speech in both in-sample out-of-sample analyses. This indicates that the approach employed for constructing IMF-MFCC rather than just MFCC on raw speech signals will outperform the current standard approach in this type of cyber mitigation ASV classification context.

Experiment 1

Dataset 1

OFS Benchmark ASV Features - IMFs - Dataset 1 - Speaker 1 vs Female Synthetic Voice TTS T1

Coeff.	MFCCs					LFCCs					BFCCs					GFCCs				
number	IMF 1	IMF 2	IMF 3	IMF L	IMF L+1	IMF 1	IMF 2	IMF 3	IMF L	IMF L+1	IMF 1	IMF 2	IMF 3	IMF L	IMF L+1	IMF 1	IMF 2	IMF 3	IMF L	IMF L+1
1	0.150	0.425	0.500	0.300	0.750	0.775	0.725	0.500	0.500	0.550	0.825	0.850	0.800	0.425	0.525	0.750	0.825	0.750	0.575	0.500
2	0.325	0.425	0.350	0.575	0.425	0.050	0.350	0.550	0.550	0.375	0.300	0.400	0.725	0.550	0.225	0.750	0.575	0.350	0.500	0.500
3	0.475	0.550	0.350	0.550	0.375	0.650	0.600	0.625	0.625	0.375	0.200	**0.900**	0.250	0.575	0.400	0.225	0.725	0.250	0.475	0.525
4	0.475	0.700	0.350	0.525	0.375	0.550	0.300	0.625	0.625	0.425	0.175	0.200	0.625	0.600	0.350	0.075	0.575	0.750	0.525	0.525
5	0.500	0.700	0.775	0.575	0.325	0.450	0.225	0.600	0.600	0.375	0.125	0.425	0.575	0.525	0.275	0.125	0.200	0.475	0.475	0.525
6	0.500	0.550	0.325	0.650	0.350	0.600	0.200	0.625	0.625	0.375	0.025	0.325	0.400	0.550	0.300	0.450	0.775	0.450	0.525	0.475
7	**0.950**	0.400	0.400	0.575	0.450	0.200	0.325	0.575	0.575	0.375	0.525	0.800	0.250	0.550	0.500	0.375	0.325	0.150	0.475	0.475
8	**1.000**	**1.000**	0.425	0.525	0.450	0.500	0.550	0.525	0.525	0.425	0.800	0.525	0.500	0.500	0.275	0.175	0.350	0.425	0.575	0.500
9	**1.000**	0.850	0.725	0.650	0.725	0.725	0.450	0.600	0.600	0.400	0.500	0.350	0.200	0.575	0.425	0.475	0.875	0.450	0.550	0.450
10	**1.000**	**1.000**	0.825	0.675	0. 450	0.800	0.475	0.550	0.550	0.400	0.500	0.600	0.525	0.600	0.450	0.775	0.550	0.200	0.475	0.450
11	**0.950**	**0.925**	0.625	0.550	0.350	0.850	0.600	0.600	0.600	0.375	0.500	0.800	0.800	0.500	0.325	0.625	0.525	0.375	0.550	0.475
12	0.800	0.650	0.775	0.475	0.350	0.575	0.600	0.575	0.575	0.375	**1.000**	0.750	0.725	0.550	0.550	0.500	0.725	0.875	0.550	0.525

Coeff.	IMFCCs					LPCs					LPCCs					MSRCCs				
number	IMF 1	IMF 2	IMF 3	IMF K	IMF K+1	IMF 1	IMF 2	IMF 3	IMF K	IMF K+1	IMF 1	IMF 2	IMF 3	IMF K	IMF K+1	IMF 1	IMF 2	IMF 3	IMF K	IMF K+1
1	0.675	0.775	0.725	0.425	0.600	0.575	0.625	0.575	0.550	0.550	0.775	0.300	0.800	0.425	0.475	0.625	0.775	0.375	0.600	0.425
2	**1.000**	0.750	0.275	0.550	0.525	0.750	0.475	0.575	0.475	0.575	**0.900**	0.300	0.775	0.450	0.555	0.425	0.600	0.750	0.450	0.375
3	0.450	0.750	0.325	0.625	0.450	0.375	0.425	0.600	0.500	0.475	0.400	0.625	0.800	0.450	0.525	0.550	0.775	0.550	0.450	0.425
4	0.750	0.575	0.550	0.550	0.475	0.450	0.575	0.600	0.550	0.625	0.200	0.525	0.750	0.525	0.430	0.150	0.675	0.475	0.550	0.400
5	0.500	0.050	0.725	0.575	0.475	0.325	0.600	0.650	0.475	0.550	0.725	0.250	0.800	0.500	0.565	0.575	0.350	0.825	0.550	0.400
6	**1.000**	0.375	0.725	0.575	0.400	0.675	0.500	0.650	0.450	0.500	0.750	0.350	0.825	0.475	0.475	**0.925**	0.650	0.325	0.575	0.425
7	0.850	0.475	0.750	0.600	0.450	0.575	0.475	0.650	0.525	0.550	0.650	0.475	0.800	0.475	0.375	0.775	0.525	0.225	0.500	0.425
8	**1.000**	0.475	0.825	0.425	0.350	0.375	0.375	0.725	0.425	0.525	0.725	0.450	0.850	0.475	0.500	0.800	0.500	0.800	0.600	0.425
9	0.500	0.650	0.650	0.625	0.400	0.350	0.425	0.750	0.550	0.550	**0.900**	0.375	0.800	0.425	0.500	0.575	0.425	0.400	0.525	0.400
10	0.750	0.575	0.425	0.500	0.375	0.500	0.475	0.750	0.475	0.550	0.750	0.350	0.750	0.525	0.545	0.600	0.350	0.725	0.575	0.450
11	0.525	0.625	0.200	0.675	0.475	0.600	0.575	0.750	0.525	0.575	0.750	0.350	0.725	0.550	0.454	0.750	0.450	0.800	0.550	0.400
12	0.750	0.625	0.125	0.550	0.500	0.675	0.525	0.750	0.475	0.575	0.700	0.450	0.575	0.475	0.525	0.875	0.500	0.650	0.450	0.425

Coeff.	NGCCs					PLPs					PSRCs					RPLPs				
number	IMF 1	IMF 2	IMF 3	IMF K	IMF K+1	IMF 1	IMF 2	IMF 3	IMF K	IMF K+1	IMF 1	IMF 2	IMF 3	IMF K	IMF K+1	IMF 1	IMF 2	IMF 3	IMF K	IMF K+1
1	0.825	**0.900**	0.825	0.500	0.500	0.505	0.590	0.450	0.300	0.420	0.575	0.350	0.300	0.525	0.450	0.510	0.500	0.600	0.420	0.565
2	0.550	0.475	0.300	0.575	0.600	0.585	0.450	0.455	0.550	0.430	0.500	0.425	0.600	0.525	0.425	0.505	0.505	0.589	0.630	0.400
3	0.175	**1.000**	0.350	0.525	0.525	0.605	0.515	0.650	0.525	0.555	0.450	0.625	0.350	0.400	0.450	0.600	0.432	0.575	0.600	0.475
4	0.125	0.700	0.225	0.525	0.675	0.515	0.575	0.450	0.475	0.565	0.450	0.575	0.450	0.525	0.425	0.420	0.690	0.355	0.515	0.500
5	0.275	0.275	0.450	0.575	0.450	0.775	0.595	0.550	0.555	0.375	0.625	0.525	0.525	0.550	0.450	0.700	0.500	0.450	0.575	0.595
6	0.550	0.800	0.550	0.550	0.525	0.545	0.675	0.655	0.500	0.500	0.625	0.550	0.425	0.500	0.450	0.700	0.685	0.500	0.500	0.450
7	0.475	0.500	0.125	0.600	0.475	0.690	0.500	0.750	0.600	0.454	0.575	0.625	0.550	0.500	0.575	0.625	0.430	0.560	0.525	0.575
8	0.400	0.325	0.425	0.575	0.625	0.850	0.600	0.750	0.600	0.525	0.500	0.450	0.575	0.500	0.425	0.855	0.740	0.675	0.675	0.605
9	0.250	0.875	0.375	0.625	0.475	**1.000**	0.770	0.675	0.675	0.550	0.475	0.375	0.575	0.500	0.600	**0.935**	0.885	0.565	0.600	0.470
10	0.300	0.575	0.300	0.675	0.500	**0.925**	0.650	0.725	0.525	0.500	0.350	0.425	0.475	0.425	0.500	**0.920**	0.780	0.500	0.410	0.525
11	**0.925**	0.500	0.300	0.550	0.500	0.890	0.725	0.725	0.475	0.475	0.400	0.600	0.475	0.550	0.575	0.890	0.700	0.505	0.675	0.550
12	0.625	0.625	0.825	0.600	0.475	0.710	0.685	0.750	0.475	0.425	0.525	0.575	0.400	0.575	0.525	0.785	0.675	0.550	0.525	0.425

Table 8.5: Out-of-sample results of the SVMs carried with the standard features used in ASV tasks applied to the IMFs. The features description is given in table 8.3. Results for these features applied to the raw data are provided in table 8.4. Note thate each value corresponds to the accuracy achieved by the SVM carried with the coefficient given in the row of the IMF basis in the column referring to the feature provided.

The second set of features is obtained from an EMD applied to speech to get IMF bases, then the MFCCs are extracted from each IMF. This is the newly proposed methodology that utilised EMD-MFCC. Table 8.4 shows the out-of-sample results for the EMD-MFCCs of the female voice for dataset one, and table 3 in the Supplement Materials presents the correspondent results for the male voice. The results are presented for the radial basis function kernel choice, and the remaining results for other kernel choices are not presented to reduce space. The way to interpret these results is as part of a stage of constructing a library of individual feature sets to pass to a multiple-kernel learning solution.

EMD-MFCC Multi Kernel Learning SVM Performance

The next step corresponds to presenting the proposed solution by combining the selection of the best-performing features from the EMD along with the EMD-MFCC SVM feature libraries constructed for various kernel choices and individual features above described (note that the results for the individual features are within the Supplement Material). The combination is achieved through the Multi Kernel Learning (MKL) introduced in 4.2. By being the best performing within the individual features studies, the EMD-MFCCs are selected in this task and demonstrated for the female case as the most challenging. Each of these chosen features (individually trained in previous experiments) will be combined according to Eqn. 4.13. The procedure consists of selecting the best kernel amongst the best feature for each feature. As a consequence, the final combined kernel should be more representative of the classification problem. Table 8.6 displays results for Speaker 1 out-of-sample performance for dataset 1. Since the best performing EMD-MFCC features amongst several kernels are selected, the header of table 8.6 is organised as follows: the top row shows which is the basis of interest, i.e. $\gamma_1(t')$, $\gamma_2(t')$, $\gamma_3(t')$, $\gamma_L(t')$ and $\gamma_{L+1}(t')$. The index following MFCC- gives this information. The second row highlights the best individually performing coefficient and, therefore, the one selected for the MKL formulation. The last row shows which kernel offers the best performance for that feature; for example, for column one of the table, for the first IMF $\gamma_1(t')$ (MFCC-1), the best performing coefficient was the 7-th one when a Laplace kernel was used. The rest of the columns can be interpreted equivalently. The header referring to the weights η_m for $m = 1, \ldots, 5$ is then entered. Each row represents a new model and shows the weights η_m defined in Eqn. 26, which are associated with features given at the head of the table. When considered individually, they reflect their out-of-sample performances and, therefore, reflect which feature provides more significant discrimination. Thus, the rows provide the new models' characterisation obtained through the combination rule given in Eqn. 4.12 with related performance in the last column provided by the accuracy score. Note that performances are ordered according to the level of accuracy achieved.

Experiment 1					
Dataset 1					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice TTS T1					
MFCC-1 7^{th} coeff. Laplace	MFCC-2 8^{th} coeff. RBF	MFCC-3 5^{th} coeff. RBF	MFCC-L 10^{th} coeff. RBF	MFCC-L+1 5^{th} coeff. Linear	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.295	0.295	0.222	0.188	–	**1.000**
0.364	0.364	0.272	–	–	**1.000**
0.380	0.380	–	0.240	–	**1.000**
0.422	–	0.311	0.267	–	**1.000**
–	0.416	0.314	0.269	–	**1.000**
0.500	0.500	–	–	–	**1.000**
0.572	–	0.428	–	–	**1.000**
0.610	–	–	0.390	–	**1.000**
–	0.611	–	0.389	–	**0.990**
–	–	0.537	0.463	–	**0.950**
0.246	0.246	0.184	0.157	0.167	**0.900**
0.292	0.292	0.216	–	0.200	**0.900**
–	0.328	0.246	0.208	0.218	**0.900**
0.418	0.048	–	0.262	0.272	0.890
0.325	–	0.242	0.214	0.219	0.890
0.374	0.374	–	–	0.252	0.890
0.415	–	0.309	–	0.276	0.890
–	–	0.362	0.309	0.329	0.890
–	0.436	–	0.277	0.287	0.890
–	0.413	0.310	–	0.277	0.890
–	0.573	0.427	–	–	0.890
–	0.602	–	–	0.398	0.890
–	–	0.529	–	0.471	0.890
0.435	–	–	0.274	0.291	0.880
0.600	–	–	–	0.400	0.880
–	–	–	0.486	0.514	0.870
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice TTS T1					
MFCC 8^{th} coeff. Laplace	MFCC 9^{th} coeff. Laplace	MFCC 11^{th} coeff. RBF	MFCC 12^{th} coeff. RBF	MFCC 8^{th} coeff. Linear	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.195	0.209	0.200	0.197	0.197	0.678

Table 8.6: Multi Kernel Learning SVMs results of the synthetic voice generated with TTS T1 versus Speaker 1 for dataset 1. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy). The first line indicates the considered features, which is always an IMF-MFCC. The IMF indices are given in each MFCC component as -1,-2,-3,-L,-L+1. The second line refers to the coefficient number, and the third line to the selected kernel for that feature. The table represents a model selection comparison in which each row corresponds to a different MKL model combining different sets of features. The numbers in each row refer to the η_m weights as expressed in Eqn. 26. The highlighted accuracy scores correspond to those combinations of features and kernel models greater than 90%. The first portion of the table demonstrates the EMD-MFCC-MKL solutions, while the second portion is the state-of-the-art reference of the classical MFCC-MKL.

Perfect discrimination is achieved when the MFCCs of $\gamma_1(t')$ or $\gamma_2(t')$ are included within the combined kernel. Such findings reinforce the initial analysis of the t-SNE that most of the discrimination between a real female voice and a synthetic female voice lies in the high-frequency MFCC coefficients of the first IMF.

Indeed, the selected coefficients for this case corresponds to the 7-th. Different combinations have been tried, and similar excellent performance was observed. These results are far higher than those of the current state-of-the-art reference of MFCC applied to raw speech when also placed in an MKL-SVM framework. This demonstrates the superior performance of the IMF-MFCC feature class when combined with an MKL-SVM classifier framework. Note that each EMD-MFCC-MKL table will follow the structure described in this section. Note that the individual performances related to the MFCCs on the raw speech signals are not presented for other datasets.

Harvard Phonetically Balanced Sentences Example: Real Speech vs Synthetic Speech Classification

As performed for dataset one, a similar analysis was confirmed by dataset two on the gold standard speech data set given by the Harvard phonetically balanced sentences. Table 8.7 shows results related to Speaker 1, hence the female discrimination case study (as per the above study one). A summary of the EMD-MFCC features is only provided and MKL-SVM classifier compared to the MFCC on raw speech in an MKL-SVM classifier. In this example, a Radial basis function kernel is employed, and the feature set was based upon the EMD-MFCCs that best performed in individual feature classifiers in the out-of-sample analysis.

8.3.3 Experiment Two: Other TTS Algorithms

In this subsection, we replicate the EMD-MFCC-MKL conducted in experiment one by taking into account different Text-To-Speech (TTS) algorithms, presented in Table 8.8. Note that the experiment is replicated for the female voice only, but both dataset one and dataset two are considered. The first TTS algorithm corresponds to the interface of the Google-Text-to-Speech API interface provided by the Python library gTTS. It relies on WaveNet (van den Oord et al. (2016)) and hence uses a Deep Learning procedure. It offers 120 languages and dialects (see `https://cloud.google.com/speech-to-text/docs/languages`). The second TTS algorithm corresponds to Espeak (online at `http://espeak.sourceforge.net/`) that instead employs a formant synthesis procedure. It also provides several languages (the complete list is given online). Afterwards, the Python library Pyttsx is used, a cross-platform text-to-speech wrapper providing access to different TTS tools. Amongst others, the Microsoft Speech Engine SAPI5 is selected (online at `https://docs.microsoft.com/en-us/previous-versions/windows/desktop/ms723627(v=vs.85)`), making use of a concatenative algorithm. The last TTS algorithm is the IBM Watson TTS (whose documentation can be found online at `https://cloud.ibm.com/docs/text-to-speech`), which also provides access to its API through a Python interface and relies on neural voice technologies, hence making use of Deep Neural Network (DNN). The TTS service is in the IBM Watson Cloud and supports a large number of languages, from which we selected the option of UK English.

Experiment 1					
Dataset 2					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice TTS T1					
MFCC-1 8th coeff. RBF	MFCC-2 2nd coeff. RBF	MFCC-3 7th coeff. RBF	MFCC-L 2nd coeff. RBF	MFCC-L+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.170	0.170	0.235	0.268	0.157	**0.944**
0.201	0.201	0.274	0.324	–	**0.944**
0.233	0.241	0.318	–	0.208	**0.923**
–	0.208	0.280	0.328	0.184	**0.923**
0.397	0.409	0.194	–	–	**0.923**
–	0.253	0.341	0.406	–	**0.923**
–	0.308	0.416	–	0.276	**0.909**
0.218	0.226	–	0.352	0.204	**0.902**
0.203	–	0.281	0.328	0.188	0.895
0.275	0.283	–	0.442	–	0.861
–	0.289	–	0.454	0.257	0.861
0.336	0.355	–	–	0.309	0.861
0.491	0.509	–	–	–	0.861
–	0.426	0.574	–	–	0.861
–	0.388	–	0.612	–	0.861
–	0.531	–	–	0.469	0.861
0.250	–	0.348	0.402	–	0.833
0.301	–	0.420	–	0.279	0.833
–	–	0.353	0.413	0.234	0.833
0.285	–	–	0.460	0.255	0.500
0.418	–	0.582	–	–	0.500
0.382	–	–	0.618	–	0.500
0.521	–	–	–	0.479	0.500
–	–	0.463	0.537	–	0.500
–	–	0.603	–	0.397	0.500
–	–	–	0.638	0.362	0.500
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice TTS T1					
MFCC 7h coeff. RBF	MFCC 8th coeff. RBF	MFCC 9th coeff. RBF	MFCC 10th coeff. RBF	MFCC 11th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.202	0.199	0.199	0.199	0.199	0.712

Table 8.7: Multi Kernel Learning SVMs results of the synthetic voice generated with TTS T1 versus Speaker 1 for dataset 2. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy). The first line indicates the considered features, which is always an IMF-MFCC. The IMF indices are given in each MFCC component as -1,-2,-3,-L,-L+1. The second line refers to the coefficient number, and the third line to the selected kernel for that feature. The table represents a model selection comparison in which each row corresponds to a different MKL model combining different sets of features. The numbers in each row refer to the η_m weights as expressed in Eqn. 26. The highlighted accuracy scores correspond to those combinations of features and kernel models greater than 90%. The first portion of the table demonstrates the EMD-MFCC-MKL solutions, while the second portion is the state-of-the-art reference of the classical MFCC-MKL.

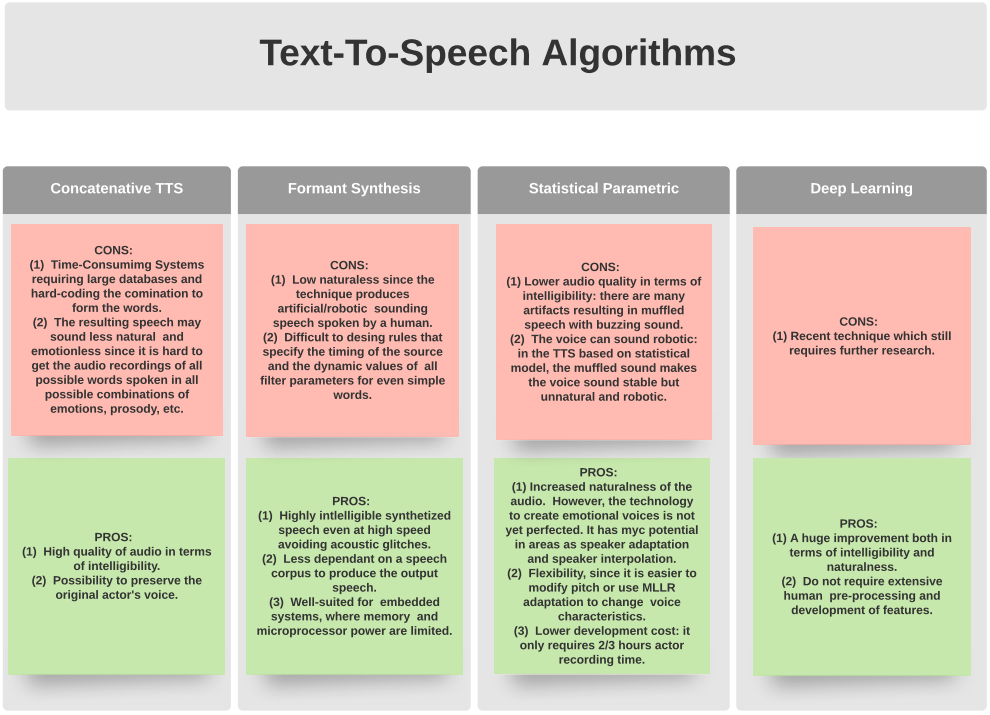


Figure 8.9: Pros and Cons of TTS algorithms.

As in experiment one, individual feature SVMs are firstly carried, hence one for each mel-frequency cepstral coefficient of the obtained IMFs. Results concerning these SVMs are provided in the Supplement Materials in tables 4 and 5 for dataset one and tables 6 and 7 for dataset two. The best performing cepstral coefficients per IMF basis function are selected, and then the EMD-MFCC-MKL procedure as presented in experiment one is carried out. Results for the IBM TTS algorithm are provided in tables 8.10 and 8.11 for dataset one and dataset two, respectively. Results for the remaining algorithms are in Supplement Materials in tables 8, 9, 10 and 11. As in the previous experiments, the best performing MFCCs for the first three IMFs are high-frequency ones confirming our initial claim that most of the discrimination power for female voices should come from these regions of the time-frequency plane. Furthermore, the achieved accuracy levels are consistent with experiment one across both datasets and all the TTS algorithms, with the EMD-MFCCs outperforming the traditional MFCCs on the raw data in each case study. This highlights that the EMD-MFCC-MKL within a TD-SD-SV system is robust to different types of TTS spoofing attacks. Tables 8.10 and 8.11 show that when using the combination of five and four EMD-MFFCs features, a level of accuracy greater than 90% is attained, hence providing the necessary countermeasure for an ASV system.

Experiment 2				
#	TTS Tool	Algorithm	Gender	Accent
T1	Online TTS	Stat. Par.	Male	British En., Harry
T1	Online TTS	Stat. Par.	Female	British En., Emma
T2	Google TTS	Deep Learning	Female	British En., en-GB-Wavenet-A
T3	Espeak	Formant Synthesis	Female	British En., Hazel
T4	IBM Watson TTS	Deep Learning	Female	British En., Charlotte
T5	SAPI5	Concatenative	Female	British En., Mary

Table 8.8: Table describing the Text-To-Speech (TTS) Tools employed in experiment two for comparisons of different Speech Synthesis algorithms producing different synthetic voices and generating different types of attacks. Note that speech generated through TTS T1, corresponding to the online TTS, was obtained from http://www.fromtexttospeech.com/.

8.3.4 Experiment Three: Application on the ASVspoof 2019 challenge Dataset

One of the biggest problems affecting ASV studies is the comparison of various techniques evaluated over different datasets. As a result, since 2015, the research community (Yamagishi et al. (2017), Wu, Kinnunen, Evans, Yamagishi, Hanilçi, Sahidullah and Sizov (2015), Kinnunen et al. (2017), Kinnunen et al. (2018)) has started to release evaluation databases as SAS, ASVspoof 2015, ASVspoof 2017, ASVspoof 2019 challenge, AVspoof, RedDots Replayed databases. Consistently with these purposes, this thesis investigates the new technique combining EMD-MFCCs by employing the ASVspoof 2019 challenge database (Todisco et al. (2019)). Table 8.9 (also at `https://www.asvspoof.org/asvspoof2019/asvspoof2019_evaluation_plan.pdf`) describes the structure of such a dataset. It subdivides into two different scenarios: logical access (LA) and physical access (PA). The former involves spoofing attacks directly injected into the ASV system. Such attacks are generated using text-to-speech synthesis (TTS) and voice conversion (VC) technologies. In the PA scenario, speech is assumed to be captured by a microphone in a physical, reverberant space. Hence, replay spoofing attacks are recordings of bonafide speech assumed to be captured and then represented to the microphone of an ASV system using a replay device. In this Chapter, the Logical Access scenario only is taken into account to target the TTS algorithms used within this database. The LA database contains bonafide speech and spoofed speech data obtained using 17 different TTS and VC systems. Figure 8.10 shows its spoofing attacks structure and the ones extracted for the given experiments. Note that data for the training of TTS and VC systems partly comes from the VCKT database (online at `http://dx.doi.org/10.7488/ds/1994`), but there is no overlap with the data contained in the 2019 database. Among the 17 spoofing voice generation systems, 6 are known attacks, while 11 are unknown. The training and development sets contain known attacks only, while the evaluation set contains 2 known and 11 unknown spoofing attacks. Regarding the 6 known attacks, there are 2 VC

systems and 4 TTS systems. Particularly, VC systems use neural-network-based and spectral-filtering-based approaches (Matrouf et al. (2006)). TTS use either waveform concatenation or neural-network-based speech synthesis using a conventional source-filter vocoder Morise et al. (2016) or a WaveNet-based vocoder van den Oord et al. (2016). We extract three of the TTS spoofing attacks for the training and development sets.

Experiment 3

ASVSpoof 2019 Challenge database

	# of Speakers		# of Utterances			
			Logical Access		Physical Access	
Subset	Male	Female	Natural	Spoof	Natural	Spoof
Training	8	12	2580	22800	5400	48600
Development	8	12	2548	22296	5400	24300
Evaluation	-	-	71747		137457	

Table 8.9: Summary of the ASVspoof 2019 Challenge database as highlighted at https://www.asvspoof.org/index2019.html.

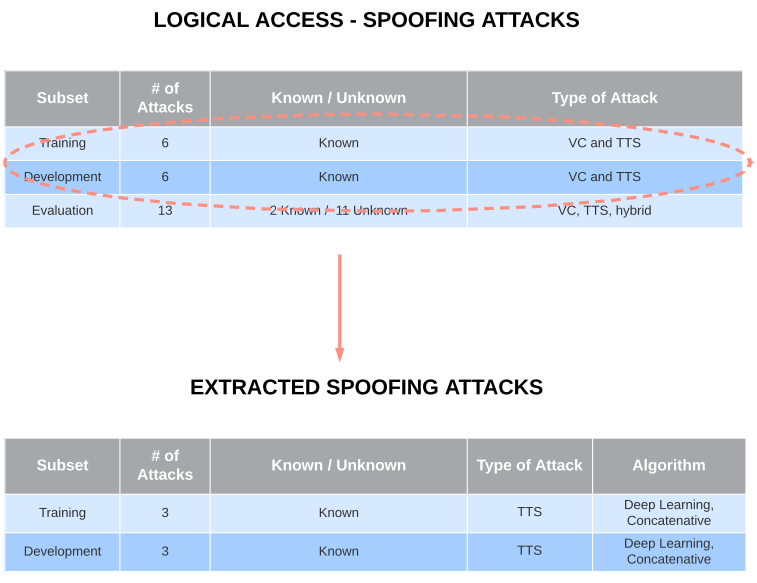


Figure 8.10: Extracted spooging attacks for experiment three from the ASVSpoof 2019 challenge database from the Logical Access set.

Experiment 2					
Dataset 1					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice IBM					
MFCC-1 8th coeff. RBF	MFCC-2 8th coeff. RBF	MFCC-3 9th coeff. RBF	MFCC-L 9th coeff. RBF	MFCC-L+1 11th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.260	0.250	0.220	0.135	0.135	**1.000**
0.305	0.297	0.228	0.170	–	**1.000**
–	0.409	0.397	0.194	–	**1.000**
–	0.331	0.291	0.189	0.189	**1.000**
0.341	0.281	–	0.189	0.189	**0.998**
0.361	0.351	0.288	–	–	**0.998**
0.422	0.410	–	0.168	–	**0.998**
0.412	0.401	–	–	0.187	**0.997**
–	0.400	0.389	–	0.211	**0.997**
0.301	0.292	0.284	–	0.123	**0.996**
0.424	–	0.400	–	0.176	**0.996**
0.433	–	0.409	0.158	–	**0.995**
0.436	–	–	0.282	0.282	**0.993**
0.537	0.463	–	–	–	**0.992**
0.556	–	0.444	–	–	**0.991**
0.349	–	0.329	0.161	0.161	**0.990**
0.605	–	–	0.395	–	**0.990**
–	0.454	–	0.273	0.273	0.899
–	–	0.436	0.282	0.282	0.898
–	–	0.620	–	0.380	0.796
–	0.623	–	0.377	–	0.795
–	0.636	–	–	0.364	0.794
–	–	0.669	0.331	–	0.793
–	0.520	0.480	–	–	0792
–	–	–	0.500	0.500	0.789
0.686	–	–	–	0.314	0.600
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice IBM					
MFCC 8th coeff. RBF	MFCC 9th coeff. RBF	MFCC 9th coeff. RBF	MFCC 6th coeff. RBF	MFCC 7th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.202	0.200	0.200	0.200	0.198	0.675

Table 8.10: Multi Kernel Learning SVMs results of the synthetic voice generated with the IBM TTS algorithm versus Speaker 1 for dataset 1. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy). The first line indicates the considered features, which is always an IMF-MFCC. The IMF indices are given in each MFCC component as -1,-2,-3,-L,-L+1. The second line refers to the coefficient number, and the third line to the selected kernel for that feature. The table represents a model selection comparison in which each row corresponds to a different MKL model combining different sets of features. The numbers in each row refer to the η_m weights as expressed in Eqn. 26. The highlighted accuracy scores correspond to those combinations of features and kernel models greater than 90%. The first portion of the table demonstrates the EMD-MFCC-MKL solutions, while the second portion is the state-of-the-art reference of the classical MFCC-MKL.

Experiment 2					
Dataset 2					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice IBM					
MFCC-1 8th coeff. RBF	MFCC-2 9th coeff. RBF	MFCC-3 9th coeff. RBF	MFCC-L 6h coeff. RBF	MFCC-L+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.239	0.230	0.233	0.101	0.197	**1.000**
0.275	0.265	0.269	0.191	–	**1.000**
0.273	0.262	0.267	–	0.198	**1.000**
0.377	0.309	0.314	–	–	**1.000**
0.380	0.365	–	0.255	–	**1.000**
0.412	0.302	–	0.132	0.154	**0.999**
–	0.397	0.312	0.134	0.157	**0.998**
0.455	0.361	–	–	0.184	**0.998**
0.401	–	0.312	0.135	0.157	**0.986**
0.457	–	0.379	0.164	–	**0.986**
0.437	–	0.375	–	0.188	**0.979**
0.544	0.456	–	–	–	0.899
–	0.458	0.379	0.163	–	0.898
–	0.573	–	0.197	0.230	0.896
–	0.438	0.374	–	0.188	0.889
0.524	–	0.476	–	–	0.879
0.571	–	–	0.198	0.231	0.868
0.733	–	–	0.267	–	0.868
–	0.525	0.475	–	–	0.799
–	0.609	–	0.391	–	0.799
–	0.695	–	–	0.305	0.799
–	–	0.591	0.203	0.236	0.786
–	–	0.642	0.358	–	0.786
–	–	0.629	–	0.371	0.779
–	–	–	0.544	0.456	0.768
0.623	–	–	–	0.377	0.706
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice IBM					
MFCC 8th coeff. RBF	MFCC 8th coeff. RBF	MFCC 9th coeff. RBF	MFCC 7th coeff. RBF	MFCC 8th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.198	0.210	0.197	0.197	0.197	0.715

Table 8.11: Multi Kernel Learning SVMs results of the synthetic voice generated with the IBM TTS algorithm versus Speaker 1 for dataset 2. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy). The first line indicates the considered features, which is always an IMF-MFCC. The IMF indices are given in each MFCC component as -1,-2,-3,-L,-L+1. The second line refers to the coefficient number and the third line to the selected kernel for that feature. The table represents a model selection comparison in which each row corresponds to a different MKL model combining different sets of features. The numbers in each row refer to the η_m weights as expressed in Eqn. 26. The highlighted accuracy scores correspond to those combinations of features and kernel models greater than 90%. The first portion of the table demonstrates the EMD-MFCC-MKL solutions, while the second portion is the state-of-the-art reference of the classical MFCC-MKL.

The generation algorithms for the spoof voices fall into either Deep Learning or Concatenative types and can be compared to the previous results (see table 8.8). In particular, the chosen algorithms are decoded in the LA protocol of the

ASVspoof 2019 challenge as "A01", "A02" and "A04", respectively. The TTS "A01" algorithm is obtained with a neural waveform model, while the TTS "A02" algorithm is generated through a vocoder. Lastly, the TTS "A04" algorithm is a waveform concatenation. There are 8 male and 12 female speakers for bonafide speech utterances and the selected TTS algorithms in the training set. However, there are 12 female and 8 male voices for the bonafide speech in the development set, but 6 female and 4 male voices for the selected synthetic ones. Note that the speakers differ between the training and the development sets, and the utterances differ amongst the speakers. Hence, a text-independent and speaker-independent scenario is the one of interest in this experiment (TI-SI-SV). Furthermore, the number of utterances per speaker and type of speech (i.e. bonafide or synthetic) differ.

Therefore, the selected subset is unbalanced in terms of the number of utterances in the bonafide or natural set versus the spoof groups. This results from a different number of speakers' utterances (this information is not evident in the proposed tables). As a result both the training and development subsets have been balanced. For the training set, the minimum number of utterances available for one speaker is selected and then the same number from each of the other available speakers in every group (bonafide and spoofed) is randomly extracted. This corresponds to 127 utterances for every speaker, 2,540 utterances for every group (i.e. natural, A01, A02, A04), with a total of 10,160 utterances. Regarding the development subset, the number of speakers within each gender is firstly balanced and then the minimum number between the two, giving 4 male and 4 female voices in each group (bonafide, A01, A02 and A04) is randomly selected. Furthermore, the same procedure that has been applied for the training set is also followed and the minimum number of utterances available per speaker within each group has been randomly selected, corresponding to 77. Therefore, each group will have 616 utterances leading to 2464 utterances for the development set. For the experiments, the training set is used to train the individual features required to develop the EMD-MFCC-MKL and the development set for testing such a procedure with the added trait of gender, hence dividing the utterances according to it. Table 8.12 provides a summary of such a dataset. Each utterances speech recording duration was approximately 1sec to 3sec maximum sampled at 16kHz producing between 25k and 150k samples per spoken utterance. The start and end of each sample were trimmed to remove any non-speech segments and decimated to a set of 40k total samples. The procedure concerning the EMD extraction followed the same applied to the other datasets, i.e. each set of 40k samples for one sentence was then windowed into non-overlapping collections of 5,000 samples and passed to the EMD sifting procedure. Then, for each IMFs, $M = 12$ cepstral coefficients were extracted, similarly to experiments one and two. One individual SVM per coefficient per IMF is carried for the female and male cases by considering the three different TTS algorithms. Results of the individual features are provided in the Supplement Materials in tables 14 and 15. For both genders, better performances are achieved by the MFCCs of the second or the third IMF basis function detecting lower speech

Experiment 3						
Dataset 3 - Extracted from LA						
	# of Speakers		**# of Utterances**			
Subset	Male	Female	Natural	A01	A02	A04
Training	8	12	2540	2540	2540	2540
Development	4	4	616	616	616	616

Table 8.12: Summary of the extracted database from the ASVspoof 2019 challenge database to conduct our experiment three. Note that we selected two subsets, i.e. the training and the development. Furthermore, for the spoofed speech, we considered three of the TTS voices only. Note that the datasets is balanced in terms of number of utterances per speaker. We make use of the training set to train our SVMs proposed models and the development set for the testing.

formants and the fundamental frequency. In this context, multiple speakers are trained together through a unique model, and, particularly at high-frequencies of female voices, the non-stationarity of each speaker might be strongly biometric, resulting in out-of-sample accuracy levels of 70% for high cepstral coefficients of IMF1. What is instead detected more efficiently in a TI-SI-SV environment are lower formants and the fundamental frequency depicted by lower cepstral coefficients of IMF3. Therefore, the EMD-MFCCs provide interpretable high-performing features for this kind of speaker verification system. The following step corresponds to the EMD-MFCC-MKL analysis. Results for the female case versus the A01 and A02 TTS algorithms and the male case versus the same TTS algorithms are provided in tables 8.13, 8.14, 8.15 and 8.16. The other results considering the TTS algorithms A04 are in the Supplement Materials in tables 14 and 15. The MKL performances reinforce the findings related to the individual SVMs. In both female and male SVMs, highest accuracy levels (>90%) are shown when the cepstral coefficients of IMF2 and IMF3 have been included in the MKL model. Furthermore, the male EMD-MFCC-MKL performances appear overall higher than the female ones; most of the formants lie, in male voices, at the lower frequency bandwidths and, compared to female formants, present in general lower non-stationarity levels. Hence, better performances are achieved if low cepstral coefficients of IMF2 and IMF3 are considered. Furthermore, the EMD-MFCC-MKL framework provides a higher level of accuracy in every case compared to the individually trained EMD-MFCC. Indeed, in the latter case, no feature achieves an accuracy level greater than 90% (these are in tables 12 and 13 of the Supplement Materials). This strongly supports the proposed methodology. Regarding the TTS algorithms, A04, hence the concatenative approach, represents a more challenging spoofing attack than the A01 and A02.

Experiment 3					
Dataset 3					
OFS EMD-MFCC-MKL - Female Case vs A01 TTS Algorithm					
MFCC-1 3rd coeff. RBF	MFCC-2 2nd coeff. RBF	MFCC-3 3rd coeff. RBF	MFCC-L 1st coeff. RBF	MFCC-L+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
–	0.553	–	0.447	–	**0.982**
–	0.516	0.484	–	–	**0.981**
–	0.557	–	–	0.443	**0.977**
–	0.364	0.342	0.294	–	**0.977**
0.481	0.519	–	–	–	**0.974**
–	0.366	0.343	–	0.291	**0.972**
0.324	0.349	0.327	–	–	**0.971**
–	0.282	0.265	0.228	0.225	**0.969**
0.253	0.272	0.256	0.220	–	**0.966**
–	0.384	–	0.310	0.306	**0.966**
0.339	0.366	–	0.295	–	**0.964**
0.253	0.273	0.256	–	0.217	**0.959**
0.208	0.224	0.210	0.181	0.178	**0.956**
0.263	0.283	–	0.229	0.226	**0.953**
0.341	0.367	–	–	0.292	**0.953**
–	–	0.538	0.462	–	**0.914**
0.267	–	0.270	0.233	0.230	0.877
0.347	–	0.351	0.302	–	0.896
0.348	–	0.352	–	0.299	0.870
0.366	–	–	0.319	0.315	0.731
–	–	0.369	0.318	0.313	0.896
0.497	–	0.503	–	–	0.883
0.535	–	–	0.465	–	0.739
0.538	–	–	–	0.462	0.515
–	–	0.541	–	0.459	0.885
–	–	–	0.503	0.497	0.740
OFS Raw Data MFCC-MKL - Female Case vs A01 TTS Algorithm					
MFCC 3rd coeff. RBF	MFCC 3rd coeff. RBF	MFCC 1st coeff. RBF	MFCC 1st coeff. RBF	MFCC 2nd coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.210	0.210	0.210	0.180	0.180	0.601

Table 8.13: Multi Kernel Learning SVMs results of the female case versus the synthetic voice generated with the A0 TTS algorithm of the ASVspoof challenge dataset. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy).

Experiment 3					
Dataset 3					
OFS EMD-MFCC-MKL - Female Case vs A02 TTS Algorithm					
MFCC-1 1st coeff. RBF	MFCC-2 1st coeff. RBF	MFCC-3 1st coeff. RBF	MFCC-L 1st coeff. RBF	MFCC-L+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
–	0.487	0.513	–	–	**0.976**
–	0.350	0.368	0.283	–	**0.972**
–	0.275	0.289	0.223	0.213	**0.963**
–	0.354	0.372	–	0.274	**0.959**
0.251	0.262	0.275	0.212	–	**0.956**
–	0.553	–	0.447	–	**0.956**
0.319	0.332	0.349	–	–	**0.955**
0.209	0.218	0.229	0.176	0.168	**0.950**
–	–	0.565	0.435	–	**0.950**
0.254	0.264	0.278	–	0.204	**0.938**
–	0.387	–	0.313	0.300	**0.933**
–	0.564	–	–	0.436	**0.932**
0.347	0.361	–	0.292	–	**0.932**
–	–	0.399	0.307	0.294	**0.929**
–	–	0.576	–	0.424	**0.927**
0.341	–	0.373	0.287	–	**0.925**
0.490	0.510	–	–	–	**0.925**
0.477	–	0.523	–	–	**0.922**
0.351	0.366	–	–	0.283	**0.914**
0.271	0.282	–	0.228	0.218	**0.911**
0.345	–	0.377	–	0.278	**0.906**
0.267	–	0.293	0.225	0.215	**0.901**
–	–	–	0.511	0.489	0.724
0.543	–	–	0.457	–	0.724
0.378	–	–	0.318	0.304	0.718
0.554	–	–	–	0.446	0.505
OFS Raw Data MFCC-MKL - Female Case vs A02 TTS Algorithm					
MFCC 1st coeff. RBF	MFCC 3rd coeff. RBF	MFCC 4th coeff. RBF	MFCC 5th coeff. RBF	MFCC 2nd coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.210	0.198	0.247	0.172	0.172	0.585

Table 8.14: Multi Kernel Learning SVMs results of the female case versus the synthetic voice generated with the A02 TTS algorithm of the ASVspoof challenge dataset. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy).

Experiment 3					
Dataset 3					
OFS EMD-MFCC-MKL - Male Case vs A01 TTS Algorithm					
MFCC-1 3rd coeff. RBF	MFCC-2 1st coeff. RBF	MFCC-3 1st coeff. RBF	MFCC-L 1st coeff. RBF	MFCC-L+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
–	0.589	–	0.411	–	**0.972**
–	0.375	0.364	0.261	–	**0.961**
–	0.507	0.493	–	–	**0.953**
–	0.414	–	0.289	0.297	**0.951**
–	0.583	–	–	0.417	**0.950**
0.269	0.274	0.266	0.191	–	**0.948**
–	0.295	0.287	0.206	0.212	**0.948**
0.366	0.373	–	0.260	–	**0.946**
–	0.372	0.361	–	0.267	**0.945**
0.332	0.339	0.329	–	–	**0.943**
0.495	0.505	–	–	–	**0.942**
0.225	0.229	0.222	0.160	0.164	**0.942**
0.289	0.295	–	0.205	0.211	**0.935**
0.267	0.273	0.265	–	0.195	**0.933**
0.364	0.371	–	–	0.266	**0.924**
–	–	0.582	0.418	–	0.883
0.370	–	0.367	0.263	–	0.865
–	–	0.407	0.292	0.300	0.865
0.291	–	0.289	0.207	0.213	0.859
–	–	0.575	–	0.425	0.838
0.502	–	0.498	–	–	0.831
0.368	–	0.364	–	0.269	0.823
–	–	–	0.493	0.507	0.735
0.584	–	–	0.416	–	0.722
0.410	–	–	0.291	0.299	0.708
0.578	–	–	–	0.422	0.500
OFS Raw Data MFCC-MKL - Male Case vs A01 TTS Algorithm					
MFCC 1st coeff. RBF	MFCC 2nd coeff. RBF	MFCC 1st coeff. RBF	MFCC 2nd coeff. RBF	MFCC 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.160	0.160	0.260	0.260	0.160	0.699

Table 8.15: Multi Kernel Learning SVMs results of the male case versus the synthetic voice generated with the A1 TTS algorithm of the ASVspoof challenge dataset.

Experiment 3					
Dataset 3					
OFS EMD-MFCC-MKL - Male Case vs A02 TTS Algorithm					
MFCC-1 3rd coeff. RBF	MFCC-2 1st coeff. RBF	MFCC-3 3rd coeff. RBF	MFCC-L 1st coeff. RBF	MFCC-L+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
–	0.495	0.505	–	–	**0.992**
–	0.348	0.356	–	0.295	**0.981**
0.318	0.337	0.344	–	–	**0.981**
–	0.347	0.354	0.299	–	**0.979**
–	0.268	0.274	0.231	0.227	**0.976**
–	0.537	–	0.463	–	**0.974**
0.247	0.261	0.267	0.225	–	**0.972**
0.248	0.262	0.268	–	0.222	**0.969**
0.202	0.214	0.218	0.185	0.181	**0.964**
–	0.541	–	–	0.459	**0.963**
0.486	0.514	–	–	–	**0.961**
–	0.369	–	0.318	0.313	**0.955**
0.336	0.356	–	0.307	–	**0.953**
0.338	0.358	–	–	0.304	**0.943**
–	–	0.542	0.458	–	**0.942**
0.258	0.274	–	0.236	0.232	**0.942**
0.334	–	0.361	0.305	–	**0.922**
–	–	0.374	0.316	0.310	**0.922**
0.257	–	0.278	0.235	0.231	**0.917**
0.480	–	0.520	–	–	**0.903**
0.336	–	0.363	–	0.301	**0.901**
–	–	0.547	–	0.453	**0.901**
0.356	–	–	0.325	0.319	0.815
0.523	–	–	0.477	–	0.830
0.527	–	–	–	0.473	0.505
–	–	–	0.505	0.495	0.831
OFS Raw Data MFCC-MKL - Male Case vs A02 TTS Algorithm					
MFCC 1st coeff. RBF	MFCC 1st coeff. RBF	MFCC 2nd coeff. RBF	MFCC 1st coeff. RBF	MFCC 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.201	0.201	0.198	0.201	0.198	0.699

Table 8.16: Multi Kernel Learning SVMs results of the male case versus the synthetic voice generated with the A02 TTS algorithm of the ASVspoof challenge dataset.

8.4 Discussion

A new class of speech biometric cyber-attack mitigation framework was developed in the class of ASV systems. This allowed addressing the challenge of the classification of synthetic and real voices. Such a biometric security task needs to account for three main factors: firstly, speech is highly non-stationary and, therefore, methods that can depict such a property are required. Secondly, the fundamental characteristic of a speech signal is its formants structure. Since each individual has distinct vocal tracts, observing formants structure is the

keystone in speech applications. Furthermore, measuring energy concentration around such frequencies should provide the discriminatory power required to differentiate spoofed and bonafide voices. Thirdly, the speech scenario considered provides different settings affecting the interpretation of the identified discrimination power. Hence, the flexibility of the classification technique in this respect is highly required. The method should be adaptive and interpretable, hence dependent on the given speech dataset but relying on a robust technique whose interpretation can be derived according to the scenario of interest (i.e. TD-SD-SV, TD-SI-SV, etc.).

The proposed solution of this Chapter is achieved by building upon existing methodologies and adapting them to work with non-stationary signals more effectively. In this way, more robust features reducing sensitivity and enhancing performance in attack mitigation are achieved. This robust method for speech synthesis spoofing attacks combines EMD and MFCCs with a multi-kernel learning SVM classifier framework. The new formulated feature libraries called EMD-MFCCs are explored and compared in various real data studies of different complexities. Since the IMFs separate frequency bands of the original signals, the employment of the MFCCs relying on the mel-filter allows to observe how frequency formants are concentrated in each IMF. The out-of-sample analysis offers better performances than the current state-of-the-art MFCC based solutions when applied directly to speech signals. Note that the current methodology of MFCC features applied directly to speech and utilised in a multi-kernel learning SVM could not achieve the minimum required standard for classification of 90% typical of biometric security. The new proposed methodology had many instances of out-of-sample performance with accuracies well above this threshold for all experiments taken into account.

The standard practice in these settings is to consider the MFCCs applied to the raw data and then construct a feature vector containing the entire set of coefficients. In this regard, the claim of this work is that the discrimination power identified by the classifier would be reduced and polluted by the different frequency bandwidths, and hence the different formants captured within a unique feature representation. The time-frequency plane must be partitioned with an a posteriori technique since the location of the formants is strictly individual-related and cannot be known a priori. Once this step is achieved, a parsimonious model trained with the computationally efficient classifier SVM-MKL is proposed. At this stage, it is worth noting to highlight that the "new-state-of-art" methods for speech classification tasks highly rely on Deep neural networks (DNNs). This class of methodologies requires a massive amount of data and high computational capacity due to the large volume of training required. The posed objective for a DNN applied in ASV settings, or equivalently in Automatic Speaker Recognition framework, is to learn individual or multiple speakers formants structure (depending on the selected speech scenarios) by training many layers of perceptrons. This procedure is replaced with the proposed methodology through a functional characterisation of the EMD and its basis functions. There-

fore, rather than learning the formants through piece-wise functions using DNN complex layer structures, the procedure it to extract them through the EMD and construct a simpler classifier. This idea proposes a sparse architecture that replaces the DNN learning the formants with an EMD basis representation requiring far fewer parameters and can be applied to small and large datasets. It is computationally very efficient and, through an MKL ensemble method, achieves high accuracy levels in performances similar to the ones often achieved by the DNN when combined.

From a speech scenario perspective, text-dependent, speaker-dependent and text-independent, speaker-independent speaker verification systems have been tested. The proposed EMD-MFCC-MKL performed better than the standard benchmark features applied to the raw speech data in both cases. Furthermore, the created features have proven to produce interpretable machine learning solutions that provide flexibility for the targeted system. Several Text-To-Speech algorithms have been considered for the spoofing attacks in both the proposed scenarios and the studied features capture the synthetic voice better than standard ones. In the case of TI-SI-SV, the concatenative TTS algorithm appears to be the most difficult to capture in both female and male cases.

We showed that the EMD-MFCCs features offer the advantage of more reliable and robust MKL-SVM classifiers. As a result, they can be generalised in different non-stationary and noisy environments. This is particularly important in real-world situations usually associated with speech biometric access ASV technologies where a speaker may be providing a recording of speech through a non-ideal background noise mobile environment. Hence, the signal transmission will not be subject to distortions, and the receiving device would then process reliable speech features to determine if access should be granted to sensitive data.

Chapter 9

Detection of Parkinson's Disease with Speech Signals

This Chapter develops the application framework testing the methodology developed in Chapters 6 and 7. The three system models will be constructed to propose three alternative models explored in the field of health diagnostic, specifically in detecting Parkinson's disease through voice speech signals. This topic has become highly relevant in recent years, given that the standard routine daily life of a patient affected by Parkinson is highly challenged by recurrent visits at the clinic for a periodic assessment to monitor the disease. This is an invasive practice since the patient needs to undertake it quite often to control the progression of Parkinson. Furthermore, the evaluation methods are subjective and based on standard surveys where the patient has to answer a set of pre-established questions. The ideal tool would detect and surveil the disease through telemonitoring solutions. Significant research has been moving towards this direction, and this Chapter intends to take a similar step. The final objective is identifying a model able to detect the presence or absence of the disease by discriminating voice speech samples.

Chapter 6 presented three system models for the stochastic process of the approximated non-stationary signal, denoted as $\tilde{S}(t)$, which is assumed to be distributed according to a Gaussian Process. Every model offers a different solution capturing the non-linearity and non-stationarity of the observed interpolated signal $\tilde{s}(t)$, which, in this case, is represented by a healthy or a sick voice speech sample.

The critical component that needs to be selected now is the kernel function for this classification task. Chapter 4 discussed different classes of kernel method for Gaussian Process. One traditional choice would be to use the class of stationary kernels (see subsection 4.4.1). However, what is needed in this Chapter is to characterise fast pace changes of the speech signal of a sick patient, and these kernels would not perform efficiently. Hence, the requirement for a kernel function that is data-adaptive and data-driven is instead the one of interest so that the strong non-stationary nature of the data generating mechanism that must be described

can be well represented. The chosen kernel function taking care of such a task is the Fisher kernel introduced in Chapter 4, subsection 4.4.4. To construct the settings for such a kernel, the specification of a model describing the underlying signal is firstly required. Indeed, the Fisher kernel relies on the Fisher score defined as the gradient with respect to each parameter of the log-likelihood of the selected model. Then the kernel function is then obtained as the dot product of the Fisher vector. The selected models used to incorporate the different underlying data structures will be the class of ARIMA time-series models. Therefore, the Fisher vector will be computed on the log-likelihood of this model class. The critical point is that the fit of these models will be computed over mini-batches of the original signals. The idea is to characterise local structures of the speech time-series, and therefore, a local fit is required. The innovation will then lie in the formulation of a unique Fisher score vector characterising the given speech population, i.e. healthy or sick. The procedure to achieve such a result will be carefully described in the fitting and testing procedure.

The final goal is to correctly classify a sample speech affected by Parkinson's disease or not. The tool employed to conduct such a task is a statistical Generalised Likelihood Ratio Test as provided in Chapter 6, section 6.39. This test will be further reformulated and defined with the Fisher score vector constructed in through fitting and testing procedures.

The Chapters is organised as follows: firstly, the existing benchmark model for Parkinson Classification are presented. Secondly, the experimental set up is described. This section highlights the settings developed for the experiments and further subsections describing the required evidence to construct the given methodology components. Afterwards, the fitting procedure for the model estimation phase is presented. The following section will instead show the testing procedure for the model validation phase. Results and discussion sections are then provided.

9.1 Novelty and Contribution

This Chapter introduces several contributions. Firstly, it provides a novel framework for the detection of Parkinson's disease testing multiple models. The models take into account a novel methodology promoting a stochastic embedding of the EMD, as presented in Chapter 6. The first system model can be considered the benchmark model of the three and directly embeds the original signal. With speech signals, where non-stationarity, non-linearity and time-varying features are the primary property that one should consider, the standard practice of working with the raw data provide poor performances. Evidence for such a fact is provided within the case study. System model 2 considers the embedding of the original IMFs and, therefore, will provide a tool incorporating non-stationary and non-linear temporal modes of the underlying signal. The case study shows that such information provides good discrimination power. System model 3 instead relies on the background also proposed in 7, which constructs an optimal

partition of the frequency content from which band-limited basis functions are constructed. This time, the idea is to characterise the stochastic processes of the signal's frequency domain by incorporating them within specific bandwidths that should carry different frequency contents over time. The extracted band-limited components of this system model provide high discrimination power in the case study shown below. The introduction of System model 3 to the detection of Parkinson's provide the most significant contribution. The reasons for it is that such a disease could be detected at very early stages through speech. However, the lack of an objective tool rather than a subjective assessment is the biggest issue in this field. The critical problem is that presence of Parkinson's within specific frequency regions cannot be detected at early stages by the judgment of a human being simply hearing it. Therefore, a tool searching these regions and identifying an objective feature able to discriminate it is highly required. The third system model fits such an argument highly, and the author believes that this is only the first step taken in this direction. Isolating distinct frequency bands and searching for discriminating features is the final desired tool that this model should construct. Further research is still required to investigate the best configuration for the models, but the obtained result provides a solid case study.

The second relevant contribution is introducing a methodological voting system to detect Parkinson's disease relying on aggregated signal information computed through the Fisher score vector to form an informative kernel function for Gaussian Processes. This research is ongoing, and further work is still required in this part. The central idea is to provide Gaussian Processes with an adaptive kernel that considers fast changes proper of speech affected by Parkinson's. The selected tool to achieve such a task is the Fisher score vector and then the Fisher kernel. This kernel function is indeed formulated as the gradient of the log-likelihood with respect to its parameter of a given model of the data and, therefore, by definition, efficiently detects structural changes of the underlying data system. The research question at this stage is how to formulate a data-adaptive system characterising a specific population by characterising multiple structural changes as a whole. The fitting and testing procedure developed in this Chapter answer this question and provides a flexible tool for non-stationary and non-linear signal in general. The application of this system to the detection of Parkinson's combined with the EMD stochastic embedding shows promising performances that could be further robustified. The voting system comes into play since a particular data- adaptive decision rule for the classification problem must be defined based on the detected structural changes.

The third relevant component is the definition of a statistical test based on such kernel construction. The GLRT is indeed derived by using the constructed Fisher vectors for the original signal, the IMFs and the IMF-BLs. This produces a novel framework for the detection of Parkinson's disease that relies on a data-adaptive statistical test based on structural changes relying on three different solutions. Further research is required since the GLRT is done on the individual IMFs or band-limited IMFs, but it would be interesting to observe how it behaves when

the IMFs are aggregated back together in a composite weighted fashion, whose weights are assigned according to fitting performances.

These novelties and contribution are now discussed and presented along the Chapter.

9.2 Existing Benchmark Model for Parkinson Classification

The application of this Chapter framework builds upon the background proposed in Kashyap et al. (2020). In this work, the authors introduce an alternative method to detect speech abnormalities caused by Cerebellar Ataxia. This corresponds to impaired coordination due to a dysfunction of the cerebellum, characterised by movements abnormalities as dysmetria, dysdiadochokinesia, dyssynergia and many others. These abnormalities affect all kinds of movements, including speech, and hence defining "ataxic speech". Signs of ataxic speech could be scanning speech ("excess and equal stress"), a reduced speech rate and deviant prosodic (i.e. rhythmical and melodic), modulation of verbal utterances, rhythmical irregularities during (fast) repetitive productions of single or multiple syllables (known as "oral dysdiadochokinesis"), a more significant variation in pitch and loudness and disturbed articulation of both consonants and vowels with reduced intelligibility (see Ackermann and Hertrich (1994), Brendel et al. (2015), Kent et al. (2000)). Several medical conditions could generate ataxic speech; in this Chapter, speech impairment movements caused by Parkinson's disease (PD) (see Ho et al. (1998)) are the ones of interest. PD is a degenerative disorder of the central nervous system resulting from the death of dopamine-containing cells in the substantia nigra, a region of the midbrain. It is the second most common neurodegenerative disorder after Alzheimer's disease Lang and Lozano (1998), Pompili et al. (2020), Bocklet et al. (2013) and includes both motor (tremor, rigidity, bradykinesia, and impairment of postural reflexes) and non-motor signs (cognitive disorders and sleep ande sensory abnormalities). Several studies reported a 70-90% prevalence of speech impairments once the disease makes its appearance (see Ho et al. (1998)). Moreover, it might be one of the earliest PD indicators (see Harel et al. (2004),) with research showing that 29% of patients consider it one of their greatest obstacles (Hartelius and Svensson (1994)). Both motor symptoms and speech movements abnormalities worsen with the progression of the disease in a nonlinear fashion (Harel et al. (2004), Skodda et al. (2009)]. At the final stage of the disease, articulation is frequently the most impaired feature (see Ho et al. (1998), Sapir et al. (1999), Logemann et al. (1978)). Medical treatments or surgical intervention can alleviate the course of the disease; however, there is no definite cure, and, therefore, an early diagnosis is highly critical to lengthen and improve the patient's life (Singh et al. (2007), Tsanas et al. (2009)).

Among the various empirical tests considered for PD dysfunctions evaluation,

there are also speech and voice tests, where an expert is subjectively assessing the patient's ability to perform a range of tasks with a perceptual judgement relying on standardised clinical scales. The standard metric specifically designed to follow PD progression is called the "Unified Parkinson's Disease Rating Scale" (UPDRS) and corresponds to a questionnaire which combines several sections to produce a comprehensive and flexible tool to monitor the course of Parkinson's and the degree of disability. Such a scale was introduced in 1987 and the reader might refer to on Rating Scales for Parkinson's Disease (2003), Martínez-Martín et al. (1994) for further details. The result corresponds to an integer number providing information about the stage of symptoms. Speech has two explicit labels in this questionnaire, namely UPDRS II-5 and UPDRS III-18, ranging between 0-4, with 0 representing the less severe stage given as "Normal speech" and 4 being the most severe stage given as "Unintelligible most of the time". Nevertheless, the requirement for developing an objective uniform tool assessing PD ataxic speech is highly needed; ideally, it would identify acoustic disturbances in displacement, direction and rate (or velocity) (Kashyap et al. (2020)). As highlighted in Bocklet et al. (2013), the final goal of such a tool would be to detect the presence of the disease and, afterwards, to surveil it revealing its advancement through the different stages. In Tsanas et al. (2009), a noninvasive telemonitoring solution is constructed by exploiting linear and nonlinear signal processing algorithms to extract useful clinically features. The authors propose a mapping between dysphonia measures and stages of UPRDS. In these works and, in general, different tasks have been used to evaluate PD speech progressions: voice sustained phonation, rapid syllable repetition, variable reading of short sentence, longer passages and freely spoken spontaneous speech. Moreover, multiple speech features referring to different voice characteristics (as acoustic, prosodic, glottal features) have been considered. The reader might refer to Pompili et al. (2020), Bocklet et al. (2013) and Tsanas et al. (2009) as main references for further description of both tasks and features.

The main contribution of Kashyap et al. (2020) is to consider phase-based cepstral features combined with the magnitude cepstrum as a human signature to detect speech abnormalities of ataxic speech. While the magnitude cepstrum has been widely used in the analysis of ataxic speech (see Jannetts and Lowit (2014), Luna-Webb (2015)), the phase cepstrum has often been discarded for two main reasons: the difficulty in phase wrapping and the conventional view of the human auditory system as "phase deaf". This perspective has recently changed, with several studies testifying that the change of sound phase has an instead significant impact on auditory perception (Laitinen et al. (2013), Paliwal and Alsteris (2003), Schroeder (1959)). Specifically, Kashyap et al. (2020) made use of the modified group delay function (MGD) (Hegde et al. (2007)) to derive phase-based cepstral coefficients (MGDCCs) and combines them with magnitude cepstrum based features known as Mel Frequency Cepstral Coefficients (MFCCs) (Frail et al. (2009), Vikram and Umarani (2013)). A Random Forest and an SVM framework are used to assess the discrimination power of these features in detecting ataxic speech. Furthermore, as a surveillant tool of CA in ataxic speech,

they employed the MGDCCs and grade the severity of ataxia speech, exhibiting a strong correlation with one of the standard clinical rating scale, as the Scale for the Assessment and Rating of Ataxia, SARA (Schmitz-Hübsch et al. (2006)). Another recent approach considering standard voice source information features is developed by Np et al. (2021). Glottal features are estimated by the authors through iterative adaptive inverse filtering and quasi-closed phase glottal inverse filtering methods. This works relies on different classification techniques to solve the problem of Parkinson's detection, namely standard pipeline methods as the Support Vector Machine and and end-to-end approach making use of deep learning architectures instead.

Following the above discussions, the advancements provided in Kashyap et al. (2020), the developments provided in Chapter 8 relying on the MFCCs method and the idea proposed by Np et al. (2021), this Chapter is set as follows: an SVM relying on the gold-standard MFCCs is set as the benchmark classification guideline model and is compared to the three system models through a Likelihood Ratio test to their SVM. The considered dataset, described in subsection 9.3.1, leads to a text-dependent environment where both controls (healthy subjects) and sick patients read a given text. The reasons to employ such a specific set of sentences using the reading text task are clarified below.

One of the features often used in the above works, (see Kashyap et al. (2020) for example) corresponds to the MGDCCs, exploiting the modified group delay function. As studied in Boashash (1992*a*), Boashash and Jones (1992), Boashash (2015), the instantaneous frequency (IF) is a function assigning a frequency to a given time, whereas the group delay (GD) is a function assigning a time to a given frequency and, therefore, the question of interest here is whether the two functions are inverses of each other. In practice, this is not always the case because the IF function may not be invertible. Two conditions need to be verified for the laws of the two functions to be inverse of one another: (1) the variations in time of the IF is monotonic, and (2) the bandwidth-duration (BT) product is sufficiently large. This restricts the signals of interest to be a monocomponent signal whose IF is a monotonic function of time. Furthermore, when this is the case, the laws carry an enclosed physical meaning being the IF describes the frequency modulation of the signal while the GD represents the time delay of the signal. Thus, a monocomponent signal when studying features based on such functions is highly required, or the interpretability of the results might be misleading. Two of the proposed system models introduced in Chapter 6 strongly rely on this discussion and propose stochastic embeddings based on the IMFs, which are, by definition, monocomponent functions. Furthermore, system model 3 is built upon the IFs of the IMFs. The constructed experimental design, presented in subsection 9.3.3, aims to detect the presence or absence of the disease.

9.3 Experimental Set Up

This section presents the case study to illustrate the performances of the three system models introduced in Chapter 6, section 6.3. The proposed application falls into speech analysis with the final goal of discrimination of presence or absence of Parkinson's' disease (PD). The existing benchmark methodologies for classifying PD patients through voice samples have been reviewed. The suggested framework builds upon the work recently introduced byKashyap et al. (2020) and considers the MFCCs as the benchmark feature. These features will be trained and tested within an SVM environment whose performances will be compared to the ones of the three system models proposed in this thesis.

The organization of the section goes as follows: firstly, the selected dataset and its experimental setup are described. Afterwards, a section explaining the required pre-processing and a procedure employed to balance the dataset are presented. The construction of training and testing sets with the experimental design taken into account are then given. The next part compares Gram Matrices of the standard radial basis functions with generated empirical covariance matrices of some real data segments and demonstrates how the standard class of stationary kernel functions cannot detect the complex data structures of the considered problem.

9.3.1 Data Description

The speech dataset employed to develop the introduced methodology is presented in *Mobile Device Voice Recordings at King's College London(MDVR-KCL) from both early and advanced Parkinson's disease patients and healthy controls* (2019). It contains recordings from participants that are healthy or affected by Parkinson's disease. The recording environment use a typical examination room with ten square meters area and a reverberation tome of approximately 500ms to perform the voice recordings. The voice recordings are performed in the realistic situation of doing a phone call and have been performed within the reverberation radius, hence, can be considered as "clean". The reader might refer to Arau-Puchades and Berardi (2013) for further understanding. Such a database was specifically selected given the quality of the recordings but, mostly, for its recording procedure. This could be of high relevance in the development of telemonitoring solutions for PD disease. The authors developed an application making use of the same functionalities as the voice recording module used within the i-PROGNOSIS Smartphone application. The idea behind this is that the voice capturing service runs as a standalone background service on the recording device and triggers voice recordings via on-and-off-hook signals of the Smartphone. The procedure foresees the recording of the microphone signal (instead of the Global System for Mobile Communication or GSM compressed stream) and, therefore, providing a high quality recording with a sample rate of 44.1 kHz and a bit depth of 16 Bit (audio CD quality). The dataset is split between two sets of recordings: in the first one, the selected participants are asked to make

a phone call and then read out two tests: "The North Wind and the Sun" and "Tech. Engin. Computer applications in geography snippet". In the second set of recordings the participants starts a spontaneous dialog with the test executor which starts asking random questions. Details of the recording procedures are further provided within *Mobile Device Voice Recordings at King's College London(MDVR-KCL) from both early and advanced Parkinson's disease patients and healthy controls* (2019).

In the explored case studies the first set of recordings is considered. Hence, the used task to assess ataxic speech in PD disease is reading a given text. The second set of recordings corresponding to spontaneous dialog is considered highly challenging for this assessment. However, it could be employed in further research and used to study surveillance of the disease and its progression. Further details about this are given in subsection 9.3.1.

There are 37 participants in total of which 21 are healthy and 16 are sick, affected by Parkinson's disease at different stage levels. Amongst the 21 healthy participants, 19 are female while 2 are male. For the 16 sick participants, 4 are female and 12 are male. The dataset looks highly unbalanced within both classes, i.e. healthy versus sick and male versus female.

Furthermore, the Parkinson participants are labelled according to the following scores: the HYR score,the UPDRS II-5 score and the UPDRS III-18 score introduced in 9.2. By considering the UPDRS II-5 score, the Parkinson's participants are classified in a range between 0 and 3 at maximum, particularly for the female patients, 2 are at a 0 stage level and 2 are at a 1 stage level. In the case of the sick male patients, 5 male patients are at a 0 stage level, 4 patients at 1 stage level, 2 patients at 2 stage level and 1 patient at a 3 stage level. The HYR score is known as the The Hoehn and Yahr Scale and was firstly published in 1967 (see Hoehn et al. (1998)) and is used to measure how Parkinson's symptoms progress and the level of disability. Stage 0 corresponds to less severe labelled as " No signs of disease", while stage 5 the most severe given as "Needing a wheelchair or bedridden unless assisted". By considering the UPDRS II-5 score, the Parkinson's participants are classified in a range between 0 and 3 at maximum, particularly for the female patients, 2 are at a 0 stage level and 2 are at a 1 stage level. In the case of the sick male patients, 5 male patients are at a 0 stage level, 4 patients at 1 stage level, 2 patients at 2 stage level and 1 patient at a 3 stage level. Figure 9.1 represents a summary of the described database. Two histograms are provided, both separated by gender which is shown on the x-axis. Note that the left one provides information about the healthy patients while the right one about sick patients.

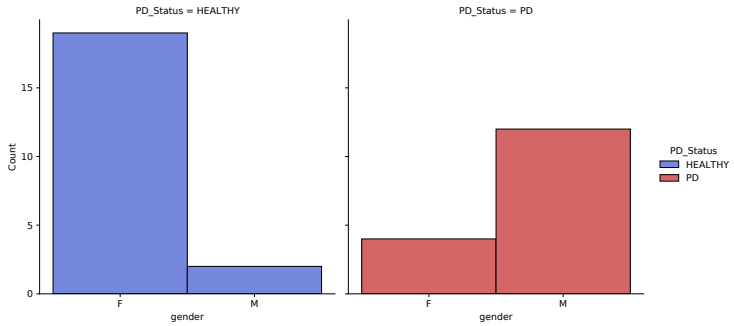


Figure 9.1: Barplots describing the participants of the considered case study. The left show the number of healthy participants of the datataset (controls) and the right one shows the number of sick patients. The x-axis is split within both barplots between gender and the y-axis shows the counts of the patients.

The dataset looks highly unbalanced within both classes, i.e. healthy versus sick and male versus female. By focusing on the PD participants only, Figure 9.2 shows the number of ill patients split according to their UPDRS II-5 score (which goes from 0 to 3 in this dataset). The patients are ordered according to their scores, i.e. the left barplot refers to female and male patients with a UPDRS II-5 score of 0, then the second one show the patients with a UPDRS II-5 score of 1 and so on. Barplots of the other given scores (i.e. UPDRS II-18 and the HYR) have been observed, and no significant differences were found.

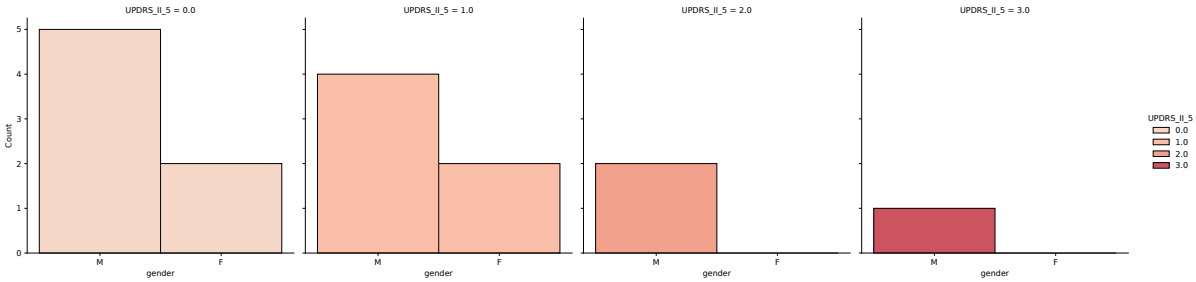


Figure 9.2: Barplots describing the sick patients divided by UPDRS II-5 score. The left barplot shows the sick patients split by gender with UPDRS II-5 score equal to 0. Then, from left to right, equivalent barplots are presented with the UPDRS II-5 score increasing from 0 to 3, which is the maximum assigned score for only one male patient. The x-axis is split between gender and the y-axis shows the count of the patients.

A further level of unbalancedness is therefore introduced. Remark that the ultimate goal is to test the performance of the proposed methodology finalised to the classification of ataxic speech for Parkinson's disease presence detection. Hence, the relevant point is to differentiate between sick patients and controls without considering the sickness stadium of the patients. Therefore, sick patients will be considered as such regardless of their stadiums. However, given the strong unbalancedness identified, the procedures to balance the dataset and pre-process it are presented in the following subsection.

9.3.2 Pre-Processing and Balancing the Dataset

In this subsection, the procedures for pre-processing and balancing the data are presented. The recordings taken into account are the read text only for each subject. Within the recording procedure, each participant was asked to make a phone call and then read two different texts above mentioned. Each audio file corresponds to a continuous, unsegmented recording of the read text at the sampling rate of 44.1kHz. Therefore, there will be one audio file for each patient denoted as $s(t)$. Depending on the patient, the reading order might change, and the recording lengths (due to different reading paces) vary between 73s and 203s. The silence at the beginning and end of the recordings was removed along with the initial participant's dialogue with the interlocutor. For the EMD to be applied, the underlying signal needs to be continuous. Therefore, a cubic spline with knots points placed at the sample points was fit through each of the recordings and denoted as $\tilde{s}(\boldsymbol{t})$. Afterwards, each recording was split into batches of 5000 samples for computational reasons, which approximately corresponds to 0.113 seconds (given a sample rate of 44.1kHz). Given that the audio files have different lengths, then the number of segments for each patient differs. Figure 9.3 shows the number of segments for each patient divided by the scores of the UPDRS II-5 for both female (left panel) and male (right panel) patients. The contained information is highly unbalanced for the number of male and female patients, the different categories of the UPDRS II-5 score and the number of sick and healthy patients. To balance the representation of each patient, the minimum number of segments for each patient by gender was computed, and then, that minimum number of segments was randomly extracted from each other patient. The minima are denoted as N_f and N_m, and, in particular, one has that $N_m = 372$ and $N_f = 442$. Therefore, there will be $N_m \times 14$ segments for the male patients and $N_f \times 23$ segments for the female patients.

9.3.3 Construction of Training and Testing Segments Sets

Once a balanced representation of each patient with respect to the number of segments is obtained, the following step consists of constructing training and testing sets for the classification tasks,i.e. model estimation and model validation. Consider the female case as an example and note that an equivalent procedure is applied to the male case. To construct the training set, one patient is firstly left out for the testing set. Then from the remaining number of patients segments, i.e. $N_f \times 18$ for the healthy case and $N_f \times 3$ for the sick case, 80% of N_f is randomly extracted corresponding to 354 segments. Hence, 354 segments will represent the class of healthy patients, and 354 will represent the class of sick patients, randomly extracted from 18 and 3 equally represented patients. For the testing set instead, 20% of N_f was randomly selected from the two left out patients segments, one for the healthy and one for the sick classes, corresponding to 89 segments. Therefore, there will be 89 segments for the healthy patient left out and 89 segments for the sick patient left out. Then, the left out patients are rotated, and the procedure repeats. Note that, $\tilde{s}(\boldsymbol{t})_0^{tr}$ and

$\tilde{s}(\boldsymbol{t})_1^{tr}$ with $tr = 1, \ldots, N_{tr}$ denote the training set and to $\tilde{s}(\boldsymbol{t})_0^{ts}$ and $\tilde{s}(\boldsymbol{t})_1^{ts}$ with $tr = 1, \ldots, N_{ts}$ denote the testing set. For the male case, there are $N_{tr} = 298$ and $N_{ts} = 75$.

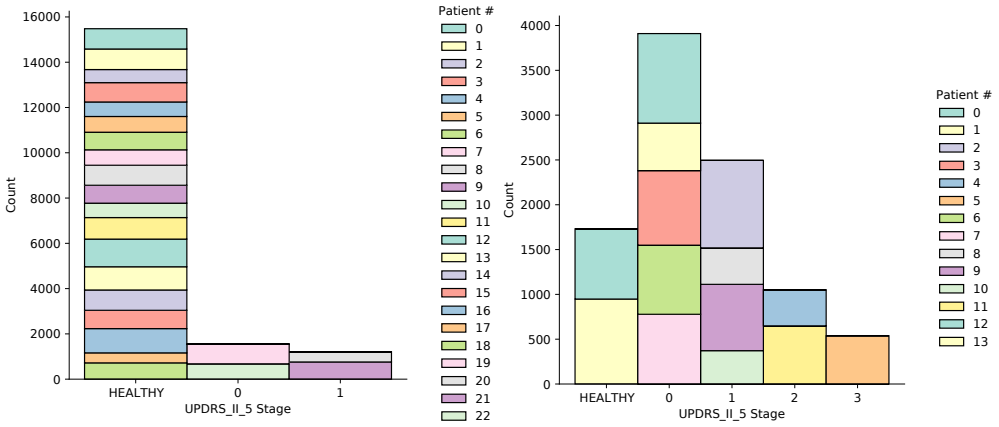


Figure 9.3: Barplots for the number of segments of length 5000 samples (approximately 0.113 seconds) for the female patients (left panels) and the male patients (right panels). The x-axis represents the different stages of the UPDRS II-5 where we also included the healthy patients. The y-axis represents the counts of the segments divided by patient.

9.3.4 The Need for the Fisher Kernel

The goal of this subsection is to provide an example that justifies the use of the Fisher kernel in the developed Gaussian Process framework for Parkinson's detection through speech samples. When Gaussian Processes are the selected tool for classification (or regression), standard kernel functions are usually adopted as the default solution to capture the similarity of the underlying signal. In practice, this is not always an efficient choice, particularly not when the studied phenomenon is affected by non-stationarity and non-linearity. The kernel is a stationary function and, therefore, would not reproduce the sought data structures.

Figure 9.4 shows four different panels presenting two randomly selected segments for the raw data of two male voices (the top ones) with their associated empirical covariance matrices (the bottom ones). Remark that each segment is made of 5000 sample points. The top left panel corresponds to a male, healthy patient segment, while the top right panel represents a segment of a male, sick patient. Hence, the bottom left panel is the covariance matrix for the top left panel, the healthy male voice, and the bottom right panel represents the covariance matrix for the sick, male voice.

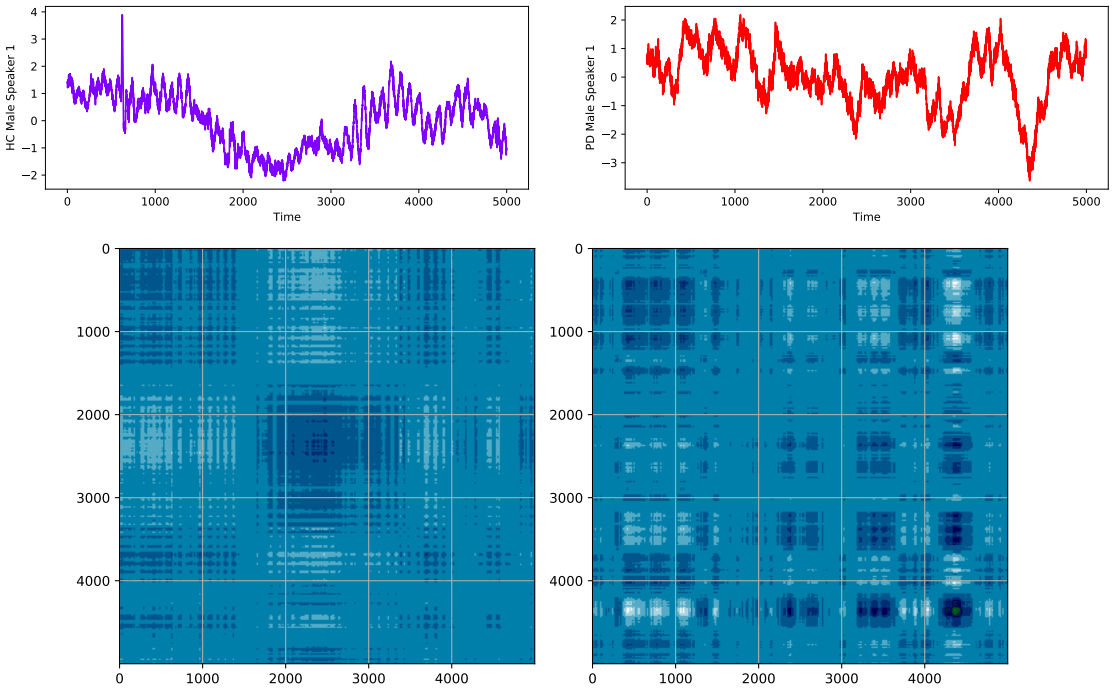


Figure 9.4: Original signals and related empirical covariance matrices of two segments of length 5000 samples of the original speech segments. The left panel (purple) represents the segment of an healthy patient, while, the right panel (red) represents the segment of a sick patient.

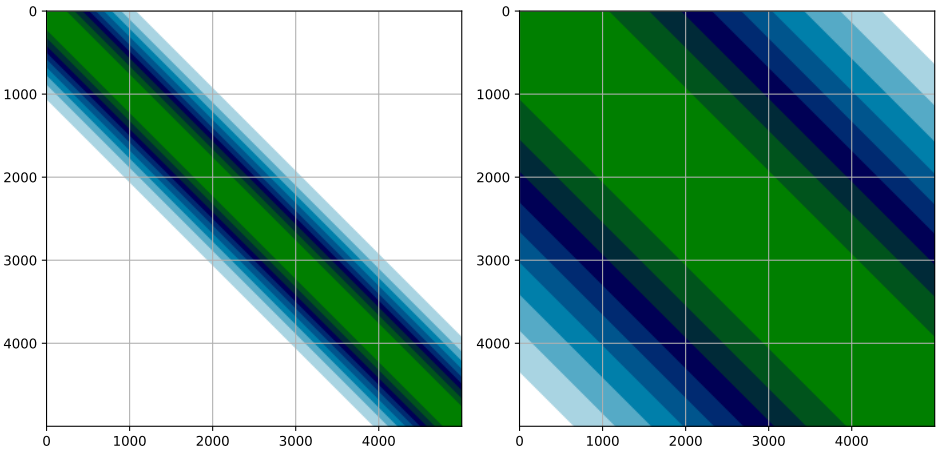


Figure 9.5: Gram Matrices of the radial basis function kernel evaluated on a uniform grid of points of length 5000 with two hyperparameters for the length scale. The left panel represent a Gram Matrix with $l = 0.1$. The right panel represent a Gram Matrix with $l = 2$.

The plots for the covariance matrices show that the underlying structures of the original data are not trivial and that any classical stationary kernel as the radial basis function would fail in detecting it efficiently. As a result, the data-driven kernel, known as the Fisher Kernel (see Chapter 4 for further references,

subsection 4.4.4), was the one employed in this work. As supporting evidence, Figure 9.5 shows two Gram matrices obtained by using a radial basis kernel function. The left Gram matrix has the length hyperparameter $l = 0.01$, while the right one is generated with $l = 2$. By looking at these Gram matrices and the empirical covariance matrices of the randomly selected segments, it is possible to observe how the generated structure of the kernel functions cannot reproduce the non-stationarity carried by the segments. The GPs generated with such Gram matrices would not fit the given signal and, more importantly, would not be able to detect the discrimination factors that provide the classifier with the power to differentiate between sick and healthy patients. The sought discrimination power lies in differences of time-varying velocity, phase and frequency of the speech signals. Therefore, it cannot be captured by these classes of kernel functions.

As a result, an ad hoc procedure characterising local structures of the speech signals is constructed. The idea is to propose a kernel function that detects local changes in the data generating process by being adaptive and data-driven. In this way, refined and fast structural changes characterising a voice affected by Parkinson's disease can be identified more easily. Given these needs, the data-driven Fisher kernel appears to be a reasonable choice. In order to incorporate such a choice in the presented stochastic embedding models, the fitting procedure and testing procedure have explicitly been developed and are described in the following sections.

9.4 The Fitting Procedure for The Estimation Model Phase

In subsections 9.3.2 and 9.3.3, the pre-processing applied to the original signals and the procedures used to extract the training and testing sets were introduced. In this section, the fitting procedure of the time-series models is presented. Consider the female case, for example. Denote the interpolated signals through a cubic spline for a female Parkinsons's voice as $\tilde{s}(t)_1$ and for a healthy female voice as $\tilde{s}(t)_0$, with $t \in [t_0, \dots, t_N]$. Hence, the 0 index refers to a female voice not affected by Parkinson, while the 1 index refers to a female voice affected by it. An equivalent notation can be considered for a male patient. The original voices are firstly split into segments of length 5000. Therefore, the notation for one segment will become $\tilde{s}(\boldsymbol{t}_i)_0$ and $\tilde{s}(\boldsymbol{t}_i)_1$, where $i = 1, \dots, N_f$ are the indices referring to the segment number for one of the two groups, i.e healthy or Parkinson female patients, respectively, and $\boldsymbol{t}_i$ corresponds to an input vector belonging to the following mesh

$$\boldsymbol{T} = \left[\boldsymbol{t}_1, \boldsymbol{t}_2, \dots, \boldsymbol{t}_{N_f}\right] = \left[[t_1, \dots, t_{5000}], [t_{5001}, \dots, t_{10000}], \dots, [t_{N-4999}, \dots, t_N]\right] \tag{9.1}$$

Note that, as described in subsection 9.3.3, the same number of segments were randomly selected for the two classes of healthy and sick patients. Select now the segments for the healthy female voice denoted as $\tilde{s}(\boldsymbol{t}_i)_0$, $i = 1, \dots, N_f$. The goal

is to characterise their local structure through a collection of scorings directly depending on the generative model inducing the data generating process of such a speech type, i.e. healthy and female. To achieve this result, one further splits each segment $\tilde{s}(\boldsymbol{t}_i)_0$ into mini-batches of length 100 sample points (corresponding to 2.2 ms). Therefore, one will have $\tilde{s}(\boldsymbol{t}_i^j)_0$ with $j = 1, \ldots, 50$ and $i = 1, \ldots, N_f$. We further redefine the mesh for the input variable set $\boldsymbol{T}$ referring to a segment $\tilde{s}(\boldsymbol{t}_i)_0$ as

$$\boldsymbol{t}_i = [\boldsymbol{t}_i^1, \ldots, \boldsymbol{t}_i^{50}] = [[t_1, \ldots, t_{100}]^i, [t_{101}, \ldots, t_{200}]^i, \ldots, [t_{4901}, \ldots, t_{5000}]^i] \quad \text{for } i = 1, \ldots, N_f \tag{9.2}$$

Note that, for each mini-batch $\tilde{s}(\boldsymbol{t}_i^j)_0$, a set of ARIMA models given in Table 9.1 will be fit without an intercept. Instead, for each mini-batch $\tilde{s}(\boldsymbol{t}_i^j)_1$, only an ARIMA(3,1,3) with intercept included will be fit. The main reason to do so is that a mini-batch belonging to the sick patients hence $\tilde{s}(\boldsymbol{t}_i^j)_1$ will have a much more complex structure due to faster changes of the speech and, therefore, will require more parameters to be efficiently detected. For a healthy mini-batch instead, all the models given in the table will be fit. Remark that a general ARIMA model with parameters p for the auto-regressive model order, q for the moving-average model order and d representing the number of differencing required to make the time series stationary, is given as follows

$$\alpha(B)\ (1 - B)^d\ \tilde{s}(\boldsymbol{t}_i^j)_0 = \beta(B)\ w(\boldsymbol{t}_i^j)_0 \tag{9.3}$$

where B is a lag operator such that $\alpha(B) = 1 - \alpha_1 B - \cdots - \alpha_p B^p$, $\beta(B) = 1 + \beta_1 B + \cdots + \beta_q B^q$ and $w(\boldsymbol{t}_i^j)_0$ is white noise. The fitting procedure aims to extract the Fisher score vector and hence deriving the Fisher kernel.

ARIMA Model	**p**	**q**
M^1	0	0
M^2	1	0
M^3	0	1
M^4	1	1
M^5	2	0
M^6	2	1
M^7	0	2
M^8	1	2
M^9	2	2
M^{10}	3	0
M^{11}	3	1
M^{11}	3	2
M^{13}	0	3
M^{14}	1	3
M^{15}	2	3

Table 9.1: Fitted ARIMA model for every sub-batch $\tilde{s}(t)_0^{i,b}$ with $b = 1, \ldots, 50$. Note that the sub-indices i and j corresponds to number of segments for the healthy and sick patients, respectively, regardless the gender. Hence, for example, for the female case, $i, j = 1, \ldots, N_f$. The parameter d is omitted since it was set equal to 1 for each of the model.

One has $15 \times 50 \times N_f$ fitted models in total for the healthy mini-batches, and the intent is to identify the one that best describes the considered populations of segments, hence the healthy female one. Note that an equivalent procedure will be carried for the healthy male mini-batches. Instead, for the sick mini-batches, one will have $1 \times 50 \times N_f$ fitted models. The same procedure is applied in the male case.

The fitting procedure for the healthy mini-batches is now introduced. Denote the winning model as $M_0^{h_\star,i,j}$, where $h_\star$ is the h-th winning model across the 15 given in Table 9.1 for each segment $\tilde{s}(\boldsymbol{t}_i^j)_0$. To identify it, consider the Akaike information criterion (AIC). Define AIC for every fitted model on every mini-batch $\tilde{s}(\boldsymbol{t}_i^j)_0$ as follows

$$\mathrm{AIC}_0^{i,j,h} = 2\kappa_0^{i,j,h} - 2\hat{\mathcal{L}}_0^{i,j,h} \ \forall i, \ \forall j \tag{9.4}$$

where $\kappa_0^{i,j,h}$ is the number of estimated parameters in the model and $\hat{\mathcal{L}}_0^{i,j,h}$ represents the log-likelihood for model h computed for the mini-batch $\tilde{s}(\boldsymbol{t}_i^j)_0$ over the input vector $\boldsymbol{t}_i^j$ defined as

$$\hat{\mathcal{L}}_0^{i,j,h} = \mathcal{L}(\tilde{s}(\boldsymbol{t}_i^j)_0, \boldsymbol{t}_i^j; \hat{\theta}_0) = \sum_{j=1}^{100} \log \ell_{\boldsymbol{t}_i^j}(\tilde{s}(\boldsymbol{t}_i^j)_0, \boldsymbol{t}_i^j; \hat{\theta}_0) \tag{9.5}$$

Table 9.2 shows the AICs scores computed from the model fits obtained on all the mini-batches for the healthy female population. The following step is to extract

the best model on every mini-batch amongst the 15 fitted models. By referring to Table 9.2, this means that one model per row will be selected.

Mini-batch	$\mathbf{M}^1$	$\mathbf{M}^2$	....	$\mathbf{M}^{15}$
$\tilde{s}(t)_0^{1,1}$	$\mathrm{AIC}_0^{1,1,1}$	$\mathrm{AIC}_0^{1,1,2}$	...	$\mathrm{AIC}_0^{1,1,15}$
$\tilde{s}(t)_0^{1,2}$	$\mathrm{AIC}_0^{1,2,1}$	$\mathrm{AIC}_0^{1,2,2}$	...	$\mathrm{AIC}_0^{1,2,15}$
...	...	...	...	...
$\tilde{s}(t)_0^{1,50}$	$\mathrm{AIC}_0^{1,50,1}$	$\mathrm{AIC}_0^{1,50,1}$	...	$\mathrm{AIC}_0^{1,50,15}$
$s(t)_0^{2,1}$	$\mathrm{AIC}_0^{2,1,1}$	$\mathrm{AIC}_0^{2,1,2}$	...	$\mathrm{AIC}_0^{2,1,15}$
...	...	...	...	...
$\tilde{s}(t)_0^{2,50}$	$\mathrm{AIC}_0^{2,50,1}$	$\mathrm{AIC}_0^{2,50,2}$	...	$\mathrm{AIC}_0^{2,50,15}$
...	...	...	...	...
$\tilde{s}(t)_0^{N_f,1}$	$\mathrm{AIC}_0^{N_f,1,1}$	$\mathrm{AIC}_0^{N_f,1,1}$	...	$\mathrm{AIC}_0^{N_f,1,15}$
...	...	...		... ...
$\tilde{s}(t)_0^{N_f,50}$	$\mathrm{AIC}_0^{N_f,50,1}$	$\mathrm{AIC}_0^{N_m,50,2}$	...	$\mathrm{AIC}_0^{N_m,50,15}$

Table 9.2: Table summarising all the scorings collected for the mini-batches of the female healthy population of patients, i.e. $\tilde{s}(t)_0$. Note that an equivalent procedure will be applied for the male case.

The best model $M_0^{h_\star}$ will be the one minimising the AIC and hence showing

$$\mathrm{AIC}_0^{h_\star,i,j} = \min_h \mathrm{AIC}_0^{i,j,h} \quad \forall\, i,j \tag{9.6}$$

where $h = 1,\ldots,15$. Afterwards, the set of winners models for each $\tilde{s}(\boldsymbol{t}_i^j)_0$ is identified and given as

$$\left\{M_0^{h_\star,1,1},\ldots,M_0^{h_\star,1,50},M_0^{h_\star,2,1},\ldots,M_0^{h_\star,2,50},\ldots,M_0^{h_\star,N_f,1},\ldots,M_0^{h_\star,N_f,50}\right\} \tag{9.7}$$

The next step consists of selecting N_f winner models, hence one for every segment $\tilde{s}(\boldsymbol{t}_i)_0$ amongst its mini-batches $\tilde{s}(\boldsymbol{t}_i^j)_0$ with $j = 1,\ldots,50$, and, therefore, the ones that provides

$$\mathrm{AIC}_0^{h_\star,i,j} = \min_j \mathrm{AIC}_0^{h_\star,i,j} \quad \forall\, i \tag{9.8}$$

where $i = 1,\ldots,N_f$. Hence, N_f winning models are selected fitted over the mini-batches $\tilde{s}(\boldsymbol{t}_i^j)_0$ as

$$\left\{M_0^{h_\star,1},M_0^{h_\star,2},\ldots,M_0^{h_\star,N_f}\right\} \tag{9.9}$$

Note that, in the above notation, the index of the mini-batches j is dropped since the best model with respect to each segment i is selected. However, the reader should remember that each selected model corresponds to the one fitted over the mini-batches of length 100 samples. Hence, the best model for the segments i was selected amongst the fitted models over the mini-batches $j = 1,\ldots,50$. In order to construct a weighted Fisher score for the population of healthy female patients proposed in the texting procedure, compute the proportion ρ_0^i reflecting

the number of times a model $M_0^{h_\star,i}$ appeared within the set of winning models over the mini-batches as

$$\rho_0^i = \frac{\left|\left\{M_0^{h_\star,1,1}, M_0^{h_\star,1,2}, \ldots, M_0^{h_\star,1,50}, M_0^{h_\star,2,1}, \ldots, M_0^{h_\star,N_f,50}, \right\} = M_0^{h_\star,i}\right|}{N_f} \quad \forall\, i \tag{9.10}$$

Note that $0 \leq \rho_0^i \leq 1$ for $i = 1, \ldots, N_f$ and $\sum_{i=1}^{N_f} \rho_0^i = 1$.

Therefore, from this fitting model procedure, a set of N_f winning models and their associated proportion computed as given above will be computed for the female healthy subjects. Remark that the same practice will be applied for the case of the male healthy participants and a set of N_m winning models will be extracted.

For the case of the sick female patients, the procedure goes exactly as the one presented so far. However, the reader should bear in mind that, given the more complexity of the speech signals associated with the presence of Parkison's disease, then only time-series ARIMA model fitted to the mini-batches given as $\tilde{s}(\boldsymbol{t}_j^i)_1$ for $j = 1, \ldots, 50$ and $i = 1, \ldots, N_f$ is a (3,1,3) ARIMA model with an intercept. Hence the first step of model selection over the mini-batches will not be required. Furthermore, by following such a procedure, the models for sick and healthy population will be nested, and the reference model will be the one of the sick patients indeed. In such a way, the GLRT test will provide reliable results given the requirements of nested models.

Note that, the presented procedures consider the observed approximated original signal, i.e. $\tilde{s}(\boldsymbol{t}_j^i)_0$ and $\tilde{s}(\boldsymbol{t}_j^i)_1$ with varying indices i and j depending on the different families. The same procedures will be repeated on the IMFs, and the band-limited IMFs and Fisher score vectors will be equivalently derived.

Figure 9.6 provides an overview of the fitting procedure proposed for the healthy subjects. It starts with the healthy patient voices on the left, presents the procedure to obtain the segments and then the mini-batches. Afterwards, 15 ARIMA models as given in Table 9.1 are fitted to each mini-batch. The following step selects the winning model over each mini-batch, and then three solutions have been proposed for the model selection stage to then construct the Fisher score vectors used in the testing procedure. Indeed, the take out of the fitting procedure will be the winning models for each population and their associated proportions.

Amongst the three proposed solution, the one proposed in this thesis is solution 1, in which the best model is chosen over the 50 mini-batches of a segment for every segment of the healthy population (i.e. male or female). Then a collection of Fisher scores defined in the following section will be given. The same procedure applies to the case of sick patients; however, at the stage of the fit, there will be only one model considered. The other two solutions propose a more flexible solution in which all the mini-batches are retained and, for solution 2, the best model is selected and then re-fitted across all the mini-batches of that segment and, for solution 3, all the models are retained. The first one resulted

in being optimal and provided more powerful performances. Furthermore, the computational cost of the second and the third solution is a lot higher.

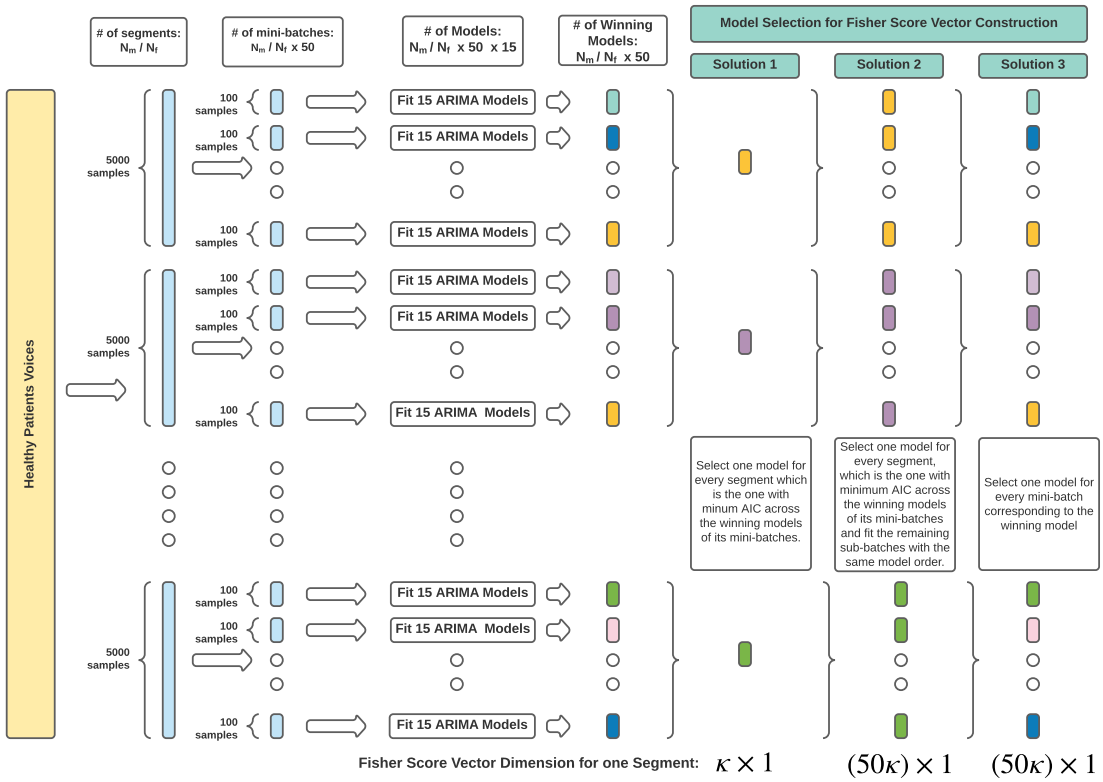


Figure 9.6: Figure showing a diagram for the steps required for the testing procedure of the model validation phase for the healthy subjects (controls).

The next step foresees the description of the testing procedure for the validation model phase which will construct the GLRT test implemented with Fisher vectors for detecting Parkinson's disease. This is presented in the following sections.

9.5 The Testing Procedure for The Validation Model Phase

In this section, the testing procedure employed for the validation model phase is presented. Once the Fisher scores for the population of segments are obtained, a new set of data, the testing set, will be used to validate the discrimination power of the three system models in detecting the presence or absence of Parkinson's disease. In practice, the test will affect the evaluation test statistic given in Eqn. 6.44 since it will evaluate the Fisher scores for the testing data and, then, the test will be directly carried on the obtained scores rather than on the segments. The objective of this section is to present the procedure required to obtain a unique Fisher score computed by aggregating information coming from the set of N_f models for the female case or N_m for the male case and then conduct a GLRT

test with such a derived quantity. This will be done over the test mini-batches for each participant that have been left over. This procedure has been described in the above sections. To present the procedure and for consistency with the fitting procedures, consider the female case as an example.

Consider a test segment denoted as $\tilde{s}(\boldsymbol{t}_i)^{\text{ts}}$, where, as in the fitting procedure, the input variable $\boldsymbol{t}_i$ corresponds to a segment of the interpolated speech of length 5000 samples. As above, each $\tilde{s}(\boldsymbol{t}_i)^{\text{ts}}$ is split into mini-batches of length 100 sample points (corresponding to 2.2ms) and then obtain mini-batches $\tilde{s}(\boldsymbol{t}_i^j)^{\text{ts}}$ with $j = 1, \ldots, 50$ and $i = 1, \ldots, N_{\text{f,test}}$. An equivalent procedure will be carried for the male case This procedures makes use of the set of parameters belonging to each of the identified winning models for each population, hence there will be N_f models for the sick and N_f models for the healthy patients that will be evaluated on the testing data.

The are N_f fitted models obtained from the fitting procedure for both families, i.e. sick and healthy, and each model is evaluated on the test mini-batches. Note that there is no re-fitting at this stage but the evaluation of the testing data. Once that is obtained, then the extraction of the Fisher score vectors is required. The procedure for the computation of the Fisher score vector is given as follows.

Consider a test mini-batch denoted as $\tilde{s}(\boldsymbol{t}_i^j)^{\text{ts}}$. For simplicity of the notation and without loss of generality, the index of the segment is dropped since the testing procedure will be conducted at a mini-batch level. Hence, define the set of testing mini-batches as $\tilde{s}(\boldsymbol{t}^j)^{\text{ts}}$. Note that, there will $N_{f,t} = N_{f,\text{test}} \times 50$ mini-batches per participant in the female case. Hence the index j will vary as $j = 1, \ldots, N_{f,t} = 4450$. An equivalent reasoning apply to the male case where one will have $N_{m,t} = N_{m,\text{test}} \times 50$. The index for the extracted model from the fitting procedure will be denoted as $h_\star^0 = 1, \ldots, N_f$ and $h_\star^1 = 1, \ldots, N_f$, for the healthy and sick families, respectively. Once evaluated the log-likelihood on the set of mini-batches of length 100 samples, then the Fisher scores for each model $h_\star^0$ and $h_\star^1$ will be computed for every mini-batch j and will be given as follows:

$$\begin{aligned} \boldsymbol{U}^j_{\boldsymbol{\theta}_0 \ (100\times\kappa_0^{j,h_\star^0})} &= \nabla\boldsymbol{\theta}_0(\mathcal{L}_0^{j,h_\star^0}) \ \forall \ j, \ \forall h_\star^0 \\ \boldsymbol{U}^j_{\boldsymbol{\theta}_1 \ (100\times\kappa_1^{j,h_\star^1})} &= \nabla\boldsymbol{\theta}_0(\mathcal{L}_1^{j,h_\star^1}) \ \forall \ j, \ \forall h_\star^1 \end{aligned} \tag{9.11}$$

Since the testing procedure will proceed equally on these two introduced Fisher scores, the following notation is introduced

$$\boldsymbol{U}^j_{\boldsymbol{\theta}_v \ (100\times\kappa_v^{j,h_\star^v})} = \nabla\boldsymbol{\theta}_v(\mathcal{L}_v^{j,h_\star^v}) \ \forall \ j, \ \forall h_\star^v \tag{9.12}$$

where the index $v = 0, 1$. Note that the index j is in the right-hand side of the above equation since the log-likelihood considered refers to model $h_\star^v$ evaluated on the mini-batch j. Furthermore, the Fisher score is evaluated at each point of the sample, i.e. the mini-batch j, This score is a matrix, indeed it dimension is $(100 \times \kappa_0^{i,j,h_\star})$, where 100 is the number of samples of the mini-batch and $\kappa_0^{j,h_\star^v}$ is

the number of parameters of the model evaluated on that mini-batch given as

$$\kappa_0^{j,h_\star^v} = p + d + 2 \tag{9.13}$$

To construct the Gram matrices required for the GLRT test, firstly the Fisher score is centred as follows:

$$\boldsymbol{U}^{j\,C}_{\boldsymbol{\theta}_v\ (100\times\kappa_v^{j,h_\star^v})} = \mathbf{V}^\intercal \text{diag}\begin{bmatrix} \hat{\sigma}_{1,1} & .. & .. \\ .. & \hat{\sigma}_{1,1} & .. \\ .. & .. & .. \\ .. & .. & \hat{\sigma}_{\kappa,\kappa} \end{bmatrix}^{-1} \mathbf{V}\ \forall\ j,\ \forall h_\star^v \tag{9.14}$$

where

$$\begin{aligned} \mathbf{V} &= \left[\boldsymbol{U}^{j}_{\boldsymbol{\theta}_v\ (100\times\kappa_v^{j,h_\star^v})}(t) - \hat{\boldsymbol{\mu}}_{\boldsymbol{U}^j_{\boldsymbol{\theta}_v}(t)}\right] \\ \hat{\boldsymbol{\mu}}_{\boldsymbol{U}^j_{\boldsymbol{\theta}_v}} &= \sum_{t=1}^{100} \boldsymbol{U}^{j}_{\boldsymbol{\theta}_v\ (100\times\kappa_0^{j,h_\star^v})}(t,:)\ \ \forall\ j,\ \forall h_\star^v \\ \left[\hat{\boldsymbol{\sigma}}_{\boldsymbol{U}^j_{\boldsymbol{\theta}_v}}\right]_s &= \frac{\sqrt{\left(\boldsymbol{U}^{i}_{\boldsymbol{\theta}_0\ (100\times\kappa_v^{i,h_\star^v})}(t,:)\ -\ \hat{\boldsymbol{\mu}}_{\boldsymbol{U}^j_{\boldsymbol{\theta}_0}}(t,:)\right)^2}}{100}\ \ \forall\ j,\ \forall h_\star^v \end{aligned} \tag{9.15}$$

Note that $\hat{\boldsymbol{\mu}}_{\boldsymbol{U}^j_{\boldsymbol{\theta}_v}}$ represents the sample mean and $\left[\hat{\boldsymbol{\sigma}}_{\boldsymbol{U}^j_{\boldsymbol{\theta}_v}}\right]_s$ represent sample standard deviation estimates of the Fisher score, respectively and are computed over the 100 samples of the mini-batch j linked to its log-likelihood $\mathcal{L}_v^{j,h_\star^v}$, for every MLE estimate. For simplicity, in the notation of the sample mean and sample standard deviation estimates, the dimensionality of the Fisher score is dropped. To avoid ambiguity, within the standard deviation formulation, done over the columns of the Fisher score, i.e. on the 100 samples for each MLE estimate, and highlight that this calculus is done over the column and not over the entire matrix, $t(:)$ has been introduced. The following step consists of summing up the evaluated Fisher score over the 100 samples for each parameter and hence obtaining

$$\boldsymbol{U}^{j,\,C}_{\boldsymbol{\theta}_v\ (1\times\kappa_v^{j,h_\star^v})} = \sum_{t=1}^{100} \boldsymbol{U}^{j\,C}_{\boldsymbol{\theta}_v\ (100\times\kappa_v^{j,h_\star^v})}(t)\ \ \forall\ j,\ \forall h_\star^v \tag{9.16}$$

The left-hand side of the above Fisher score is now of dimension $(1 \times \kappa_b^{j,h_\star^v})$ and does not depend on t anymore. This is because the gradients previously evaluated for each parameter at the values $\boldsymbol{t}$ of the given mini-batch $\tilde{s}(\boldsymbol{t}_j^i)^{\text{test}}$ are now summed up together over the vector $\boldsymbol{t}$ and, therefore, a Fisher score vector evaluated at the MLE estimates is now obtained. However, the important step in this construction is that these Fisher scores are centered across the dimension t. Also, the centring indicator has been dropped on the left-hand side, but the reader should bear in mind that these Fisher vectors have been centered for computational stability reasons. Remark now that each mode $h_\star^v$ corresponds to a winning model extracted from the fitting procedure and that the models

differs amongst them. They carry the same order in the case of sick patients, i.e. always a (3,1,3) ARIMA model but, in the case of the healthy patients these models different between them. Each Fisher score vector has, therefore, a different dimension. To construct a unique Fisher score, the obtained Fisher score vectors are modified by padding zeros within the vector up to the number of maximum possible parameters, being $3+3+2=8$. However, the vector will be ordered in terms of the comprised parameter and formally given as

$$\boldsymbol{T} = [\delta, \alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3] \tag{9.17}$$

where, in order, δ is the intercept of the ARIMA fitted model, $\alpha_1, \alpha_2, \alpha_3$ are the AR parameters and $\beta_1, \beta_2, \beta_3$ the MA paramters. Define now a padding operator given as O given as

$$\boldsymbol{U}^{j,h_\star^v}_{\boldsymbol{\theta}_v\ (1\times\kappa)} = O\left[\boldsymbol{U}^{j,\,C}_{\boldsymbol{\theta}_v\ (1\times\kappa_v^{j,h_\star^v})}\right] \quad \forall\ j,\ \forall h_\star^v \tag{9.18}$$

such that it will return a Fisher vector zero-padded for the elements of $\boldsymbol{T}$ in $\boldsymbol{U}^{i}_{\boldsymbol{\theta}_v\ (1\times\kappa_v^{j,h_\star^v})}$ that are not present. Hence, this new Fisher vector and will always be of dimension $(1\times\kappa)$ with $\kappa = 8$. Note that, for the healthy category, the intercept position will always be zero by construction. Note that the index for the model h^v is now on the left-hand side. Now, at this point, one will have one Fisher score vector of dimension $(1\times\kappa)$ for every population $v=0,1$, every mini-batch $j=1,\ldots,N_{f,t}$, every model $h_\star^v$. To aggregate the information related to every model evaluated on the testing data and hence capturing structural properties provided by the Fisher vector, for every mini-batch, all the Fisher vectors from every model will be summed up together as

$$\tilde{\boldsymbol{U}}^{j}_{\boldsymbol{\theta}_v} = \sum_{h_\star^v=1}^{N_f} \rho_v^{h_\star^v} \boldsymbol{U}^{j,h_\star^v}_{\boldsymbol{\theta}_v\ (1\times\kappa)} \quad \forall j \tag{9.19}$$

where $\rho_v^{h_\star^v}$ is the proportion computed in Equation 9.10 since each Fisher score is weighted according to the proportion of the winning times of that model. Note that in the fitting procedure explanation this was denoted as ρ_v^i and $i=1,\ldots,N_f$ corresponded to the number of models extracted on a mini-batch which provided the best fit and, therefore, i and h indicates the same quantity.

Next, the Gram matrix for the mini-batch $\tilde{s}(\boldsymbol{t}_i^j)_v$ will be defined as

$$\tilde{\mathbf{K}}^{j}_{v\ (\kappa\times\kappa)} = \tilde{\boldsymbol{U}}^{j\ \intercal}_{\boldsymbol{\theta}_v} \tilde{\boldsymbol{U}}^{j}_{\boldsymbol{\theta}_v} \quad \text{for}\ \ j=1,\ldots,N_{f,t} \tag{9.20}$$

To regularise the above matrix due to computational instability that could lead to issues encountered with the inversion of such a matrix or the log-determinant, a covariance shrinkage estimator was considered. The covariance shrinkage estimator of $\tilde{\mathbf{K}}^{j}_{v\ (\kappa\times\kappa)}$ is given by

$$\tilde{\mathbf{K}}^{j\ S}_{v\ (\kappa\times\kappa)} = (1-\gamma)\ \tilde{\mathbf{K}}^{j}_{v\ (\kappa\times\kappa)} + \gamma\ \mathbf{Q}\ \mathbb{I}_\kappa\ \kappa \tag{9.21}$$

where γ is some shrinkage factor, $\mathbb{I}_\kappa$ is the identity matrix of dimension κ and the matrix $\mathbf{Q}$ is given as

$$\mathbf{Q} = \frac{\text{tr}\left[\tilde{\mathbf{K}}^j_{v\ (\kappa\times\kappa)}\right]}{\kappa} \tag{9.22}$$

Once this is derived, then the GLRT test can be computed for every testing mini-batch j for female case (as for the male ones) as follows:

$$\begin{aligned}\hat{L} = &-(\tilde{\boldsymbol{U}}^j_{\boldsymbol{\theta}_0})\left(\tilde{\mathbf{K}}^{j\,S}_0\right)^{-1}(\tilde{\boldsymbol{U}}^j_{\boldsymbol{\theta}_0})^\intercal - \log\left(\det\left[\tilde{\mathbf{K}}^{j\,S}_0\right]\right)\\ &+(\tilde{\boldsymbol{U}}^j_{\boldsymbol{\theta}_1})\left(\tilde{\mathbf{K}}^{j\,S}_1\right)^{-1}(\boldsymbol{U}^j_{\boldsymbol{\theta}_1})^\intercal + \log\left(\det\left[\tilde{\mathbf{K}}^{j\,S}_1\right]\right)\end{aligned} \tag{9.23}$$

In practice, the Generalised Likelihood Ratio Test is evaluated for Fisher score vectors derived from the winning models of the testing set segments with the constructed Gram matrices obtained through the fitting procedure. Figure 9.7 provides a diagram summarising the steps required for the testing procedure. It is applied to one testing mini-batch and then will be repeated on each of the remaining testing mini-batches. As presented, each model for the two category of participants will be evaluated on the given mini-batch. Afterwards, according to the steps introduced above, the two Fisher score vectors will be derive by aggregating the individual Fisher scores evaluated with the different model parameters and will provide $\tilde{\boldsymbol{U}}^j_0$ and $\tilde{\boldsymbol{U}}^j_1$. At that point the Gram Matrices evaluated on that mini-batch can be computed and the GLRT test will be then calculated. This process will be repeated for each mini-batch and every patient. Results will be provided in the following section, where, the proportion of mini-batches failing to reject the null hypothesis, i.e. being sick, will be shown.

As presented in Chapter 6, section 6.4, the GLRT test will be evaluated for system model one on the approximated signal, while, for the other two system models, the same procedure will be conducted on the first three IMFs. Results are provide in the section below.

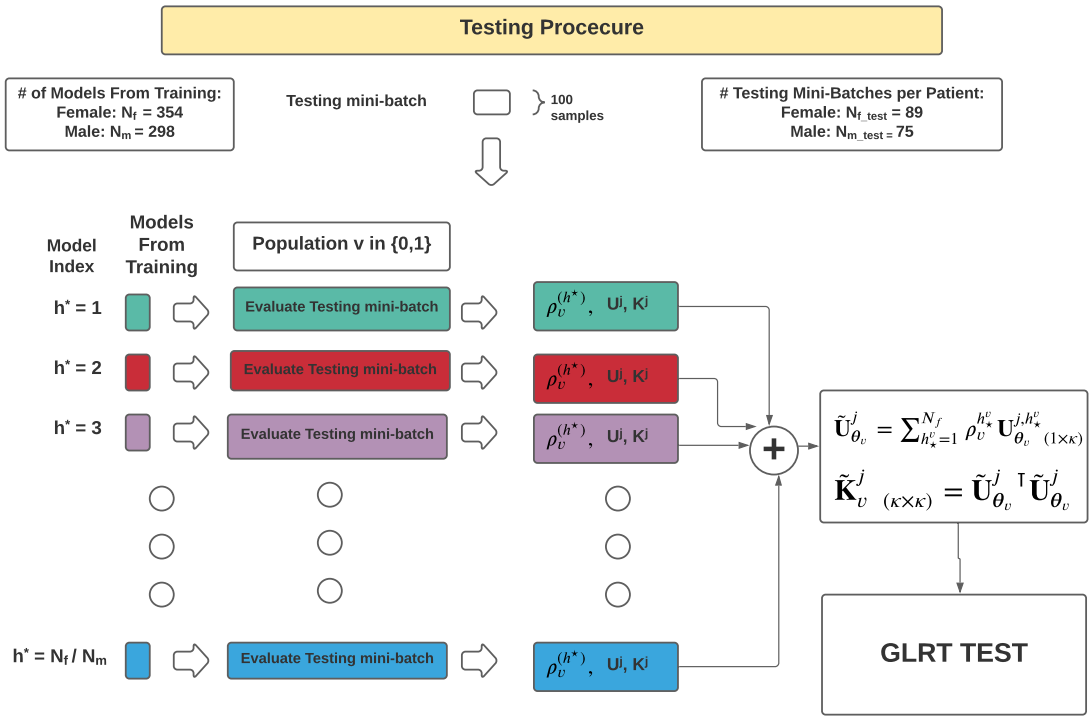


Figure 9.7: Figure showing a diagram for the steps required for the testing procedure of the model estimation phase.

9.6 Results and Discussion

In this section, some results of the presented settings are presented and discussed. The reader should bear in mind that this is ongoing research and, therefore, only partial results are presented.

This set of results shows the performance of 31 randomly selected patients of the 37 of the introduced dataset, of which 19 are healthy patients and 12 are affected by Parkinson's. Hence, the testing set is reduced, and only 31 patients have been rotated when individual performances are provided or employed if more global results testing them all together is provided. Among the 19 healthy subjects, 18 are female, and one is male, while for the 12 affected by Parkinson's disease, 2 of them are female subjects, and 10 are male patients. At this stage, gender is ignored for the classification, but it will be considered for future results since the author believes it will add critical insights useful for the selected application. The pre-processing and balancing of the dataset and the fitting and testing procedures have been applied as presented. Remark that the number of segments which are left out for every patients are $N_{f,\text{test}} = 89$ for the female subjects and $N_{m,\text{test}} = 75$ for the male ones. However, these results are presented at a mini-batch lever. Hence there will be $N_f \times 50 = 4450$ mini-batches for the female patients and $N_m \times 50 = 3750$ mini-batches for the male speech samples.

The first set of results refer to two confusion matrices presented in Figure 9.8. The left confusion matrix refers to the gold standard features selected as benchmark model comparison. These are the MFCCs extracted from the raw data. The adopted procedure for this experiment follows the setting that has been previously introduced in Chapter 5 for the SVM configuration, while the MFCCs have been extracted with the same pre-emphasis and hamming windowed given in Chapter 8. It goes as follows. For the training segments, a set of MFCCs is extracted, and an SVM is trained. Hence, this analysis is conducted at a segment level since the MFCCs are extracted on a segment of length 5000 samples rather than on a mini-batch. The SVM consider the radial basis function only, and cross-validation with 5-k folds has been applied. Once trained, the set of testing segments are passed through the same feature extraction procedure where the set of MFCCs are computed on the testing segments of length 5000 samples. Afterwards, the built SVM has been tested with all the patients together hence with no added difference for female or male patients. Results are provided in the left confusion matrix of Figure 9.8. Remark that 0 corresponds to the class of healthy subjects while 1 to one of the sick patients. The performances for the detection of both classes appear low, particularly for detecting Parkinson's disease.

The right panel of Figure 9.8 represents the performances of system model one presented in Chapter 6. Hence, this aims to test detecting sick and healthy patients by fitting a GP over the original approximated signal. The Fisher scores are computed over the 31 patients considered, and then the GLRT is conducted per mini-batch. Afterwards, the confusion matrix considering all the 31 patients has been constructed, and the results are shown in Figure 9.8. Performances appear to be low again in the detection of both classes.

As expected, the first system model considering a GP on the raw data does not provide good performances for this task. Indeed, the studied signals are highly non-stationary, and the discriminatory power that the classification should provide cannot be depicted by merely considering the raw data. The information that this exercise tries to capture is highly refined, time-varying and strictly relates to the energy change and frequency of the signal. Therefore a method considering the raw data is not powerful enough in this setting. When it comes to the gold standard MFCCs, it is interesting to note that such a feature have been employed as it is common practice in the main literature. Hence the coefficients are extracted and then stacked into a unique vector instead of considering them one by one for the classification task. As suggested in Chapter 8, such a procedure tends to mix the detected energy contents, particularly when affected by non-stationarity, and the classifier yields unreliable results. Further research will be conducted on such models by also improving the performance of the MFCCs with EMD-MFCCs as suggested in Chapter 8. The performance should improve, and it would be interesting to observe which frequency regions detected with the EMD-MFCCs will provide discriminatory power in detecting Parkinson's depending on gender.

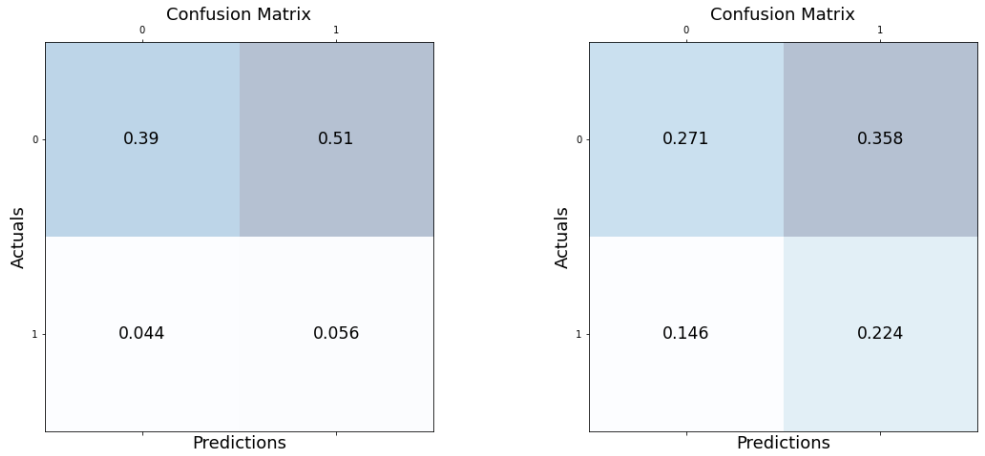


Figure 9.8: Plots representing the confusion matrices for the benchmark model (left panel) using the MFCCs and system model 1 (right panel).

The second part of this initial set of results consider a comparison of the performances of the three system models introduced in Chapter 6. Each model has been trained according to the fitting procedure above described. Note that for system model 2 and system model 3, the procedure has been applied on each of the considered basis functions. Furthermore, it is important to highlight that in this case study, system model 2 considers the first three IMFs only, hence the ones carrying the highest frequency content. As highlighted in Chapter 8, the first three IMFs tend to capture the great majority of formants present in a speech signal and, therefore, these are studied at an initial state. Further research will be conducted to consider all of them. Similar reasoning applies to the IMFs-BL hence the band-limited IMFs. The cross-entropy method has been applied to the first three IFs only. This will be extended to study the behaviour of these basis functions if more IFs are considered for their construction. Note that the cross-entropy method considered is the discrete one using a multinomial importance distribution. Three frequency band-limited bands have been selected as the desired one, but more sophisticated solutions will be later explored.

Results are provided per patient. Hence, for each tested patient, the proportion of the mini-batches that fail to reject H_0 is plotted in Figure 9.9. Remark that H_0 states equality in distribution, and the null hypothesis uses the healthy as the base model. Hence the test will provide equality with respect to healthy subjects. The plot presents three panels. The x-axis shows the patients ordered from the left to the right according to their status, i.e., the 19 healthy subjects and 12 sick ones. The y-axis shows the proportion of mini-batches that fail to reject H_0. The top panel refers to the performances of system model 1, and it is possible to observe that there is not much differentiation across the patients and does not detect any difference in distribution regardless of the status of the given patient. This confirms the finding provided by the confusion matrix presented above, where system model one tested with all the patients at once performs poorly. The second panel presents results for the second system model.

Note that, for every patient, the proportions will be given for the first three IMFs where IMF1 represents the first highest IMF, IMF2 the second-highest and IMF3 the third-highest. This time the test appears to perform more efficiently since for the healthy subjects, a low proportion of mini-batches fails to reject H_0, on average 0.2 across all the healthy patients and with similar performances across the three IMF basis functions. For the sick patients instead, a 50% proportion of the mini-batches for approximately every patient does not appear to fail to reject H_0. Similar behaviours are shown by the three IMFs, consistently as for the healthy patients. These findings are encouraging in the sense that the test appears to be sensible to differentiate between healthy and sick patients when the test is conducted at a patient level. However, a proportion of 50% might not be highly accurate, and more research is required in this direction. The third panel presents the proportion of mini-batches that fails to reject H_0 when system model 3 is the selected model. The test appears to provide more powerful performances than the ones of system model 2 with respect to both classes, i.e. healthy and sick. For the healthy patients, 20% of the mini-batches fail to reject H_0 when the first band-limited IMF is selected. More discriminant performances are achieved by the second and third band-limited IMFs, which appears to fail to reject H_0 at a percentage of 10% or lower. Hence the second and the third band-limited IMFs show more powerful behaviours in detecting healthy subjects. When it comes to sick patients, the test appears to provide more discrimination power compared to the one of system model 2.

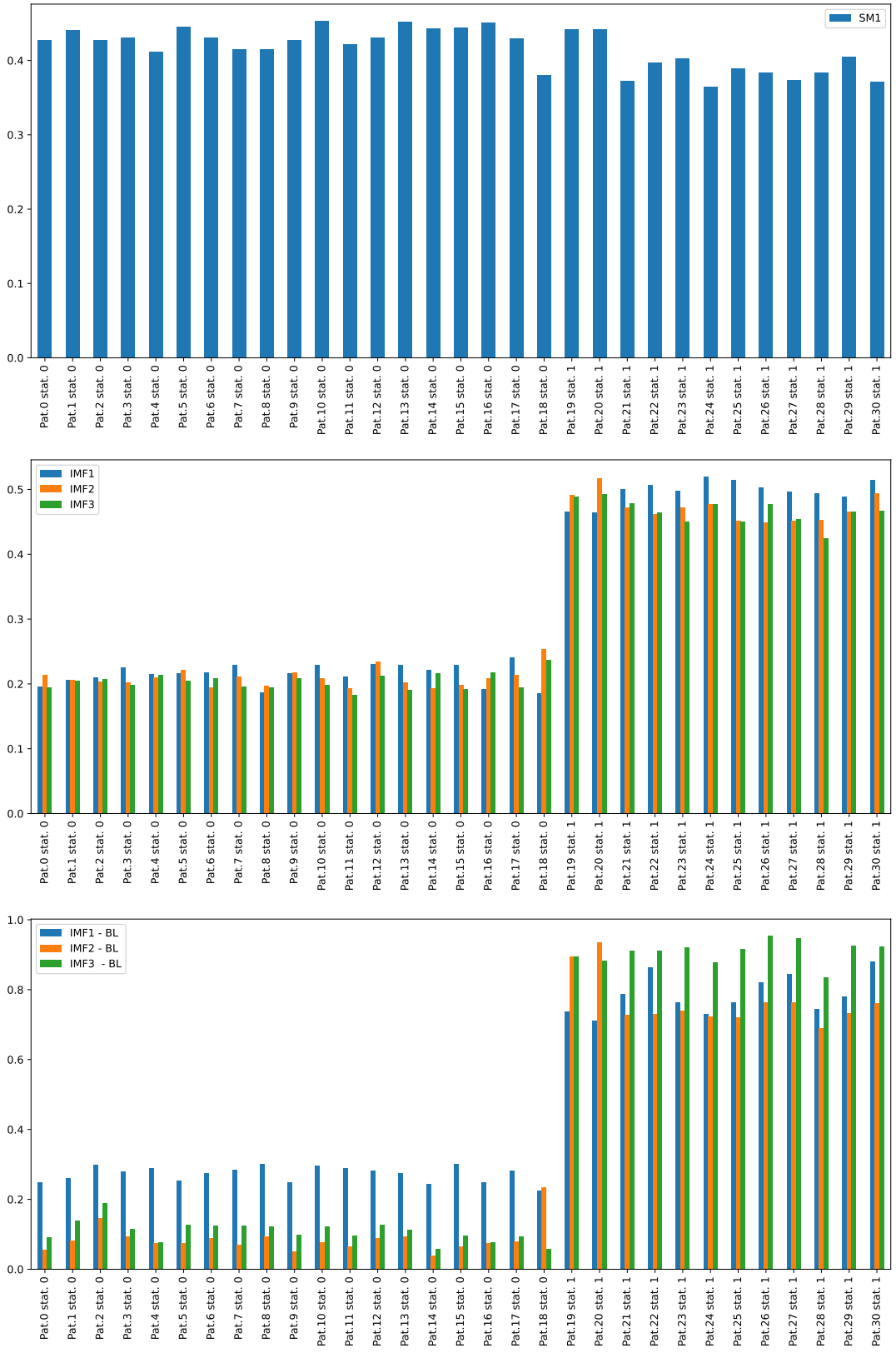


Figure 9.9: Plots representing the proportion of mini-batches that fails to reject H_0 for the three system models introduce in Chapter 6. Note that H_0 tests equality with the healthy population. The x-axis represents the patients ordered according to their status, while the y-axis is the proportion.

Indeed, particularly for IMF3-BL, the proportion of mini-batches that does not fail to reject H_0 is on average 90% for almost every patient. IMF1-BL and IMF2-BL also provide high performances, around 80% on average. While for the healthy subjects, the second and the third band-limited IMFs appear to carry the majority of discrimination power, in the case of detecting a sick patient, IMF1-BL and IMF3-BL are the ones best performing.

Overall, the performance of system model 3 appears to be the most powerful. The detection of both healthy and sick patients conducted per patient and at a level of the mini-batches provides encouraging performances that need further investigation. The same reasoning applies to system model 2, but the performances appear to be lower than system model 3. This study considers 3 IMFs and 3 IMF-BL only. Further research will explore the entire set of IMFs. Furthermore, a more general setting which will be speaker or patient independent providing the most general settings found in practice will be explored. This will also be done by considering the difference in gender given that, as provided in Chapter 8, the location of the formants changes of a great deal between males and females, and therefore, to identify which frequency regions are more affected by the disease and hence will provide strong discrimination power, the classification task must take into account this differentiation.

What has not been explained so far is how, if a speaker-dependent environment is considered, like the one of interest in the second part of this case study, is how to decide if a patient is correctly classified or not. A voting-rule system considering the proportion of mini-batches could be a good solution. Another possibility is going back to the segments from which the mini-batches were extracted, and doing a voting rule on the segments rather than on the mini-batches. More advance solutions could be considered in this will be object of further research.

9.7 Spectrograms of the Segments with GLRT Performances

To further clarify the last comment in the above section presenting the idea of a voting rule for the decision of correctly classifying a patient or not, Figure 9.10 has been provided. This figure presents 6 panels. Particularly, there are three spectrograms, one for every IMF-BL (considered in the above case study) extracted for patient 5, that is an healthy subject, and for a specific test segment (number 25) of length 5000 samples. It is possible to observe how the frequency content is spread across the three IMF-BL. This is an healthy patient and, according to the findings provided in the above section, the second band-limited IMF and the third one should carry the majority of the discrimination power. Each of the spectrograms has a second band plot that is associated with it. This represents a vector of 0 and 1 of length 50 providing the results of the GLRT conducted on the 50 mini-batches of that specific IMF-BL of that particular segment. Note that black represents 0 and white represents 1.

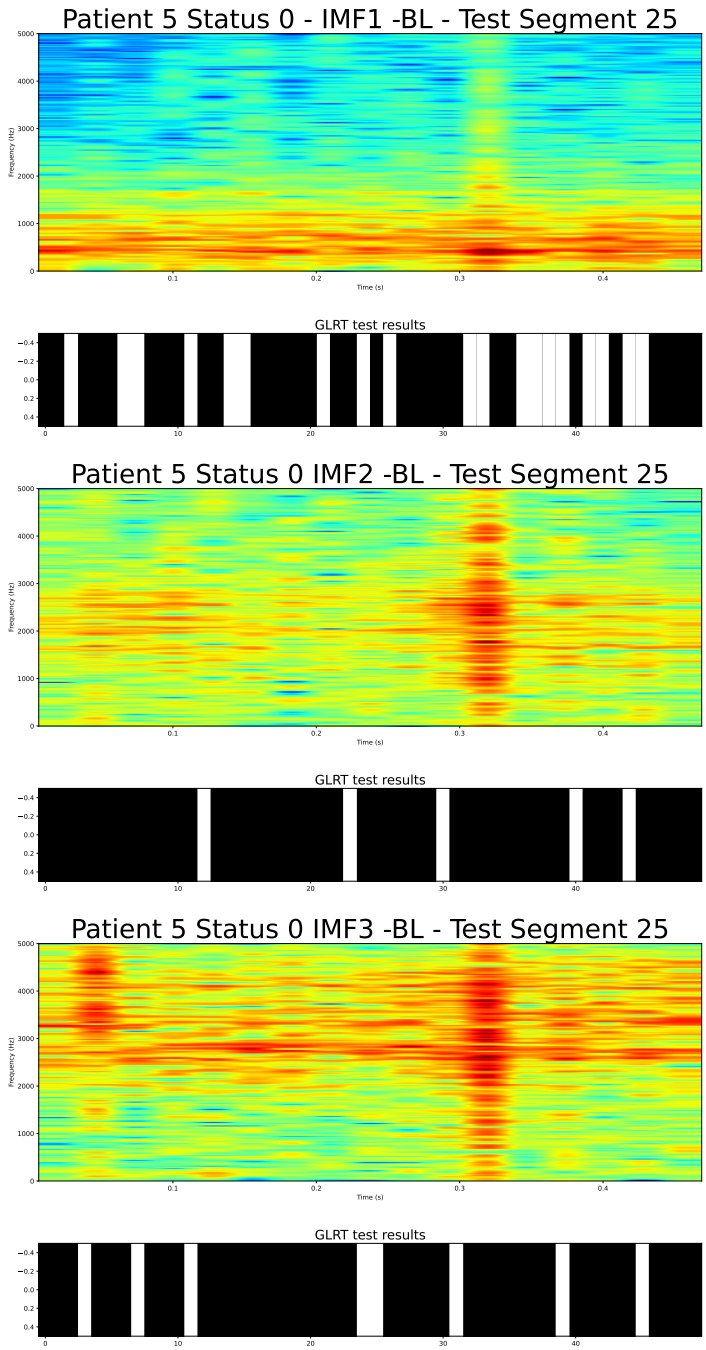


Figure 9.10: Spectrograms of the three band-limited IMFs for segment number 25 for patient 5, whose status is 0 hence is healthy. The top panel represents th spectrogram for IMF1-BL (hence the one carrying the highest frequency content), IMF2-BL is in the middle spectrogram and the last one represents IMF3-BL. Each spectrograms has a further band associated with it, representing the results of the GLRT test carried over the mini-batches of that segment. Note that white corresponds to 1 and black to 0.

It is possible to observe how the band plots of the second and the third spectrograms are indeed providing zeros and hence failure to reject H_0 over the great majority of that IMF-BL suggesting good performances of the GLRT test since detecting the correct class, i.e. healthy.

It might be interesting to study different voting rule, as suggested above, one at a mini-batch level and the other one considering the segments rather than the mini-batch. Further research will be taken in this direction.

Chapter 10

Conclusion and Future Research

This final Chapter provides a review of the main findings of this thesis and highlights some of the open questions with the future research that will be undertaken.

10.1 Summary of the Main Findings

This thesis promotes a statistical background for the non-stationary decomposition method known as the Empirical Mode Decomposition. The first part of the thesis focuses on a review of traditional time-frequency methods and their properties and an explanation of why a non-stationary and non-linear time-frequency method is required. This is highly needed at an applied level within multiple applied areas. Real-world phenomena are strongly affected by time-varying data system generation and a tool able to deal with both domains, the time and the frequency, with an optimised data-adaptive solution is one of the sought objectives of this work.

The EMD is a data-adaptive method whose main criticism is based on the lack of a mathematical basis that allows for a definition in closed form of its basis functions. One of the achievements proposed in this thesis and published in Campi et al. (2021) is a formal definition of the EMD provided in Chapter 3. The definition of its basis functions in closed form can be achieved once a specific representation is selected. The one selected in this work, given its optimality, as presented in Chapter 3, is the cubic spline interpolation. If selected, this representation allows one to derive the coefficients of each IMF basis function recursively as a linear combination of the spline coefficients of the original signal and the coefficients of the previous IMFs. The proof of such a proposition is provided in Appendix A. The recursive property could be advantageous if studying the relationship between the signal and its bases or the contribution of those specific bases to the entire signal. Chapter 3 also deal with the definition in closed form of the IF and its interpretation. This is of particular interest since the IF might provide precious insights into the underlying time-varying signal when used as a statistical feature. Multiple toy examples are provided to explain

different aspects of the sifting procedure used to extract the basis and how the wrong choices at this stage might lead to unreliable results. The author believes that this is highly critical when setting a data analysis exercise relying on this method.

The understanding of the different aspects of the sifting procedure combined with the formal definition of the EMD led to the definition of multiple classes of features defining a time-varying non-stationary library that could then be employed in challenging classification tasks as speech. In Chapter 8, this EMD library has been tested in solving the task of speech verification in various environments and provides robust results in all of them. This is highly encouraging, and further research will be undertaken to define new features based on the EMD to study structural changes of the underlying process to provide a statistical interpretation.

The second part of the thesis introduces three main core components tested in the part III within speech applications.

The first core component is the introduction of different kernel methods and multi-kernel learning procedures, which will be combined with the classification framework of Support Vector Machine to solve challenging speech verification tasks. These tools will allow the development of a statistical framework for non-stationary time-frequency methods to capture speech signatures and vocal fingerprints more efficiently than gold standard exiting speech features. The great advantage provided by multi-kernel methods given in Chapter 4 combined with the SVM of Chapter 5 is to enhance the performance of the classifier that by using different kernels combined and based on multiple time-varying features achieves more accurate discrimination. Such a fact is proved through the Automatic Speaker Verification framework provided in Chapter 8.

The second component represents the most significant contribution of the thesis and defines a stochastic version of the EMD. Such a representation is highly relevant since it allows to formulate proabilistic distributional statements of the IMF basis functions, providing more reliable classification and forecasting solutions. Numerous assumptions are considered, leading to unconditional and conditional distributions of the basis functions developing different sets up. The constructed framework challenges the existing multi-kernel machine learning technique via the proposition of a multi-kernel formulation based on a stochastic process characterised as the convolution of the stochastic processes of the IMFs, which relies on multi-kernel Gaussian Process with an additive structure for the kernel function. This technique is more advanced compared to existing multi-kernel formulation since it allows one to deal with highly non-stationary and non-linear data systems by relying on the IMF basis function and will capture temporal time-varying content of the underlying signal

The third component of part II of this thesis is represented by the definition of an EMD stochastic embedding whose aim is to capture specific bandwidths of the given signal. This is the second most significant contribution of the thesis. It

provides a fully data-adaptive technique dealing with time-varying bases in both time and frequency since it develops an optimal partition of the time-frequency plane based on a discrete quantisation of the instantaneous frequencies. This solves one of the biggest issues affecting time-frequency methods represented by the time-resolution trade-off encountered when non-stationary data comes into play. Indeed, understanding the frequency evolution over time of a signal is one of the biggest challenges for time-frequency methods. The optimisation method proposed in this thesis tackles such an issue.

This thesis has two main applications which are documented in part III. Both of them are in the area of speech analysis.

The first application tackles Automatic Speaker Verification technologies and aims to solve the problem of differentiating an authentic voice from a synthetic one. There are numerous contributions in this part which have been highly reviewed in Chapter 8. The EMD features are tested in this challenging non-stationary setting and are combined with the standard golden method called the Mel-Frequency Cepstral Coefficients. This combination provides the great advantage of shaping a new feature library useful in speech since it allows partitioning the time-frequency plane in an a posteriori fashion and identifying which areas are more discriminant to solve this classification task. Results show that this feature outperforms standard speech methodologies in multiple speech settings, as speaker-dependent or speaker-independent and text-dependent and text-independent.

The second application is relevant for health diagnostics and aims to identify through speech the presence or absence of Parkinson's disease. The stochastic embedding EMD framework is tested in this setting and introduces a novel statistical methodology relying on the use of the Fisher score vector to detect local structural changes of the underlying speech signals. Standard practice in using the Fisher kernel would be using one model over the signal, and the computation of the kernel is carried. The novelty at this stage is that the model of interest will be fitted on mini-batches of the speech signal, with the results that multiple models describe one population of signals (sick or healthy). The Fisher vector used to detect local structure will be an aggregation of multiple Fisher scores that better perform when it comes to detecting fast, local changes proper of patients affected by Parkinson's disease. The results show that adopting such a data-adaptive kernel with the proposed EMD embedding outperforms the traditional methodologies employed.

10.2 Open Questions and Further Research

Multiple research questions could be considered as future research. In this section, some of the most relevant are presented.

The first methodological question that will be covered in the near future concerns the construction of the stochastic embedding. A relevant point to raise is that this

additive multi-kernel stochastic embedding assumes that every input dimension, hence every IMF, is independent of each other. There is no structure modelling the interactions between the IMF basis functions. This is an initial research step taken towards a much more structured model which considers a multi-output Gaussian Process (see Alvarez and Lawrence (2011)) in which the structural dependence present between IMF stochastic processes will also be taken into account. Multiple solutions could be considered, and this is ongoing research.

One of the problems tackled by the third system model given in Chapter 6 is the definition of basis functions which are based on the locations of the instantaneous frequencies. Indirectly, one associated issue with this argument is that the estimation of the instantaneous frequency is a complex problem that has been widely discussed in multiple pieces of literature. In practice, characterising the instantaneous frequency's stochastic process and achieving it in closed form is a non-trivial problem. Further research could also consider this issue.

At an applied level, Chapter 9 aims to detect the presence or absence of Parkinson's disease through speech. Another relevant issue affecting a patient would be monitoring the advancement of the disease. Hence, the objectives of further research will involve the development of a surveillance tool.

Bibliography

Aarts, E. and Korst, J. (1989), *Simulated annealing and Boltzmann machines: a stochastic approach to combinatorial optimization and neural computing*, John Wiley & Sons, Inc.

Abadan, S. S., Shabri, A. and Ismail, S. (2015), Hybrid empirical mode decomposition-arima for forecasting exchange rates, *in* 'AIP Conference Proceedings', Vol. 1643, American Institute of Physics, pp. 256–263.

Abadan, S. and Shabri, A. (2014), 'Hybrid empirical mode decomposition-arima for forecasting price of rice', *Applied Mathematical Sciences* **8**(63), 3133–3143.

Abbasnejad, M. E., Ramachandram, D. and Mandava, R. (2012), 'A survey of the state of the art in learning the kernels', *Knowledge and information systems* **31**(2), 193–221.

Ackermann, H. and Hertrich, I. (1994), 'Speech rate and rhythm in cerebellar dysarthria: An acoustic analysis of syllabic timing', *Folia phoniatrica et logopaedica* **46**(2), 70–78.

Alam, M. J., Kinnunen, T., Kenny, P., Ouellet, P. and O'Shaughnessy, D. (2013), 'Multitaper mfcc and plp features for speaker verification using i-vectors', *Speech communication* **55**(2), 237–251.

Alon, G., Kroese, D. P., Raviv, T. and Rubinstein, R. Y. (2005), 'Application of the cross-entropy method to the buffer allocation problem in a simulation-based environment', *Annals of Operations Research* **134**(1), 137–151.

Alvarez, M. A. and Lawrence, N. D. (2011), 'Computationally efficient convolved multiple output gaussian processes', *The Journal of Machine Learning Research* **12**, 1459–1500.

Ambrogioni, L. and Maris, E. (2019), 'Complex-valued gaussian process regression for time series analysis', *Signal Processing* **160**, 215–228.

An, N., Zhao, W., Wang, J., Shang, D. and Zhao, E. (2013), 'Using multi-output feedforward neural network with empirical mode decomposition based signal filtering for electricity demand forecasting', *Energy* **49**, 279–288.

Arau-Puchades, H. and Berardi, U. (2013), The reverberation radius in an enclosure with asymmetrical absorption distribution, *in* 'Proceedings of Meetings on Acoustics ICA2013', Vol. 19, Acoustical Society of America, p. 015141.

Archambeau, C. and Bach, F. (2011), 'Multiple gaussian process models', *arXiv preprint arXiv:1110.5238* .

Asmussen, S., Kroese, D. P. and Rubinstein, R. Y. (2005), 'Heavy tails, importance sampling and cross–entropy', *Stochastic Models* **21**(1), 57–76.

Awajan, A., Ismail, M. T. and Alwadi, S. (2019), 'A review on empirical mode decomposition in forecasting time series', *Italian Journal of Pure and Applied Mathematics* **43**, 301–323.

Bach, F. (2008), 'Exploring large feature spaces with hierarchical multiple kernel learning', *arXiv preprint arXiv:0809.1493* .

Bach, F. R., Lanckriet, G. R. and Jordan, M. I. (2004), Multiple kernel learning, conic duality, and the smo algorithm, *in* 'Proceedings of the twenty-first international conference on Machine learning', p. 6.

Bashar, M., Ahmed, M. T., Syduzzaman, M., Ray, P. J. and Islam, A. T. (2014), 'Text-independent speaker identification system using average pitch and formant analysis', *International Journal on Information Theory (IJIT)* **3**(3), 23–30.

Bedi, J. and Toshniwal, D. (2018), 'Empirical mode decomposition based deep learning for electricity demand forecasting', *IEEE access* **6**, 49144–49156.

Bedrosian, E. (1963), 'A product theorem for hilbert transforms', *Proceedings of the IEEE* **51**(5), 868–869.

Bishop, C. M. et al. (1995), *Neural networks for pattern recognition*, Oxford university press.

Boashash, B. (1992*a*), 'Estimating and interpreting the instantaneous frequency of a signal. i. fundamentals', *Proceedings of the IEEE* **80**(4), 520–538.

Boashash, B. (1992*b*), 'Estimating and interpreting the instantaneous frequency of a signal. ii. algorithms and applications', *Proceedings of the IEEE* **80**(4), 540–568.

Boashash, B. (2015), *Time-frequency signal analysis and processing: a comprehensive reference*, Academic Press.

Boashash, B. and Jones, G. (1992), Instantaneous frequency and time-frequency distributions, Longman Cheshire.

Bochner, S. (1953), *Lectures on Fourier Integrals...: With an Author's Supplement on Monotonic Functions, Stieltjes Integrals and Harmonic Analysis [Monotone Funktionen, Stieltjessche Integrale and Harmonische Analyse]*, Princeton University Press.

Bochner, S. et al. (1959), *Lectures on Fourier integrals*, Vol. 42, Princeton University Press.

Bocklet, T., Steidl, S., Nöth, E. and Skodda, S. (2013), Automatic evaluation of parkinson's speech-acoustic, prosodic and voice related cues., *in* 'Interspeech', pp. 1149–1153.

Bonizzi, P., Karel, J., De Weerd, P., Lowet, E., Roberts, M., Westra, R., Meste, O. and Peeters, R. (2012), Singular spectrum analysis improves analysis of local field potentials from macaque v1 in active fixation task, *in* '2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society', IEEE, pp. 2945–2948.

Bonizzi, P., Karel, J. M., Meste, O. and Peeters, R. L. (2014), 'Singular spectrum decomposition: A new method for time series decomposition', *Advances in Adaptive Data Analysis* **6**(04), 1450011.

Bouboulis, P. and Theodoridis, S. (2010), 'Extension of wirtinger's calculus to reproducing kernel hilbert spaces and the complex kernel lms', *IEEE Transactions on Signal Processing* **59**(3), 964–978.

Bouzid, A. and Ellouze, N. (2004), Empirical mode decomposition of voiced speech signal, *in* 'First International Symposium on Control, Communications and Signal Processing, 2004.', IEEE, pp. 603–606.

Boyd, S. and Vandenberghe, L. (2004), *Convex optimization*, Cambridge university press.

Brendel, B., Synofzik, M., Ackermann, H., Lindig, T., Schölderle, T., Schöls, L. and Ziegler, W. (2015), 'Comparing speech characteristics in spinocerebellar ataxias type 3 and type 6 with friedreich ataxia', *Journal of neurology* **262**(1), 21–26.

Brigham, E. O. and Morrow, R. (1967), 'The fast fourier transform', *IEEE spectrum* **4**(12), 63–70.

Büyükşahin, Ü. Ç. and Ertekin, Ş. (2019), 'Improving forecasting accuracy of time series data using a new arima-ann hybrid method and empirical mode decomposition', *Neurocomputing* **361**, 151–163.

Campbell, E. L., Hernández, G. and Calvo, J. R. (2018), Feature extraction of automatic speaker recognition, analysis and evaluation in real environment, *in* 'International Workshop on Artificial Intelligence and Pattern Recognition', Springer, pp. 376–383.

Campbell, J. P. (1997), 'Speaker recognition: A tutorial', *Proceedings of the IEEE* **85**(9), 1437–1462.

Campbell, W. M., Sturim, D. E. and Reynolds, D. A. (2006), 'Support vector machines using gmm supervectors for speaker verification', *IEEE signal processing letters* **13**(5), 308–311.

Campi, M., Peters, G. W., Azzaoui, N. and Matsui, T. (2021), 'Machine learning mitigants for speech based cyber risk', *IEEE Access* **9**, 136831–136860.

Chakroborty, S. and Saha, G. (2009), 'Improved text-independent speaker identification using fused mfcc & imfcc feature sets based on gaussian filter', *International Journal of Signal Processing* **5**(1), 11–19.

Chan, Y. (1994), *Wavelet basics*, Springer Science & Business Media.

Chen, Q., Huang, N., Riemenschneider, S. and Xu, Y. (2006), 'A b-spline approach for empirical mode decompositions', *Advances in Computational Mathematics* **24**(1-4), 171–195.

Chepuri, K. and Homem-De-Mello, T. (2005), 'Solving the vehicle routing problem with stochastic demands using the cross-entropy method', *Annals of Operations Research* **134**(1), 153–181.

Chougala, M. and Kuntoji, S. (2016), Novel text independent speaker recognition using lpc based formants, *in* '2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)', IEEE, pp. 510–513.

Cohen, L. (1995), *Time-frequency analysis*, Vol. 778, Prentice hall.

Copson, E. T. and Copson, E. T. (2004), *Asymptotic expansions*, number 55, Cambridge University Press.

Cortes, C. and Vapnik, V. (1995), 'Support-vector networks', *Machine learning* **20**(3), 273–297.

Coughlin, K. and Tung, K.-K. (2004), '11-year solar cycle in the stratosphere extracted by the empirical mode decomposition method', *Advances in space research* **34**(2), 323–329.

Cressie, N. (2015), *Statistics for spatial data*, John Wiley & Sons.

Dai, Z. and Zhu, H. (2020), 'Forecasting stock market returns by combining sum-of-the-parts and ensemble empirical mode decomposition', *Applied Economics* **52**(21), 2309–2323.

Dalpiaz, G., Rubini, R., D'Elia, G., Cocconcelli, M., Chaari, F., Zimroz, R., Bartelmus, W., Haddar, M. et al. (2013), Advances in condition monitoring of machinery in non-stationary operations, *in* 'Proceedings of the third international conference on condition monitoring of machinery in non-stationary operations CMMNO', Springer.

Dalpiaz, G., Rubini, R., D'Elia, G., Cocconcelli, M., Chaari, F., Zimroz, R., Bartelmus, W., Haddar, M. et al. (2013), Advances in condition monitoring of machinery in non-stationary operations, *in* 'Proceedings of the third international conference on condition monitoring of machinery in non-stationary operations cmmno', Springer.

Damianou, A. and Lawrence, N. D. (2013), Deep gaussian processes, *in* 'Artificial intelligence and statistics', PMLR, pp. 207–215.

Dätig, M. and Schlurmann, T. (2004), 'Performance and limitations of the hilbert–huang transformation (hht) with an application to irregular water waves', *Ocean Engineering* **31**(14-15), 1783–1834.

Daubechies, I., Lu, J. and Wu, H.-T. (2011), 'Synchrosqueezed wavelet transforms: An empirical mode decomposition-like tool', *Applied and computational harmonic analysis* **30**(2), 243–261.

De Boer, P.-T., Kroese, D. P., Mannor, S. and Rubinstein, R. Y. (2005), 'A tutorial on the cross-entropy method', *Annals of operations research* **134**(1), 19–67.

de Boor, C. (2001), *A Practical Guide to Splines. Applied Mathematical Sciences*, Springer-Verlag.

Demartines, P. and Hérault, J. (1997), 'Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets', *IEEE Transactions on neural networks* **8**(1), 148–154.

Domínguez-Molina, J. A. and Rocha-Arteaga, A. (2007), 'On the infinite divisibility of some skewed symmetric distributions', *Statistics & probability letters* **77**(6), 644–648.

Duan, W.-y., Huang, L.-m., Han, Y. and Huang, D.-t. (2016), 'A hybrid emd-ar model for nonlinear and non-stationary wave forecasting', *Journal of Zhejiang University-SCIENCE A* **17**(2), 115–129.

Duan, W.-y., Huang, L.-m., Han, Y., Zhang, Y.-h. and Huang, S. (2015), 'A hybrid ar-emd-svr model for the short-term prediction of nonlinear and non-stationary ship motion', *Journal of Zhejiang University-SCIENCE A* **16**(7), 562–576.

Dubin, U. (2002), 'Application of the cross-entropy method to neural computation', *Unpublished master's thesis, Technion* .

Durrande, N., Ginsbourger, D. and Roustant, O. (2012), Additive covariance kernels for high-dimensional gaussian process modeling, *in* 'Annales de la Faculté des sciences de Toulouse: Mathématiques', Vol. 21, pp. 481–499.

Durrande, N., Hensman, J., Rattray, M. and Lawrence, N. D. (2016), 'Detecting periodicities with gaussian processes', *PeerJ Computer Science* **2**, e50.

Duvenaud, D., Nickisch, H. and Rasmussen, C. E. (2011), 'Additive gaussian processes', *arXiv preprint arXiv:1112.4394* .

el Malek, M. B. A. and Hanna, S. S. (2020), 'The hilbert transform of cubic splines', *Communications in Nonlinear Science and Numerical Simulation* **80**, 104983.
URL: *http://www.sciencedirect.com/science/article/pii/S1007570419303028*

Faisal, M. Y. and Suyanto, S. (2019), Specaugment impact on automatic speaker verification system, *in* '2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)', IEEE, pp. 305–308.

Fan, X. and Hansen, J. H. (2009), Speaker identification with whispered speech based on modified lfcc parameters and feature mapping, *in* '2009 IEEE International Conference on Acoustics, Speech and Signal Processing', IEEE, pp. 4553–4556.

Fine, S., Navratil, J. and Gopinath, R. A. (2001), A hybrid gmm/svm approach to speaker identification, *in* '2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 01CH37221)', Vol. 1, IEEE, pp. 417–420.

Frail, R., Godino-Llorente, J., Saenz-Lechon, N., Osma-Ruiz, V. and Fredouille, C. (2009), 'Mfcc-based remote pathology detection on speech transmitted through the telephone channel', *Proc Biosignals* .

Freeman, W. J. (2004), 'Origin, structure, and role of background eeg activity. part 1. analytic amplitude', *Clinical Neurophysiology* **115**(9), 2077–2088.

Gabor, D. (1946), 'Theory of communication. part 1: The analysis of information', *Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering* **93**(26), 429–441.

Garthwaite, P. H., Jolliffe, I. T., Jolliffe, I. and Jones, B. (2002), *Statistical inference*, Oxford University Press on Demand.

Gelfer, M. P. and Young, S. R. (1997), 'Comparisons of intensity measures and their stability in male and female sneakers', *Journal of voice* **11**(2), 178–186.

Ghil, M., Allen, M., Dettinger, M., Ide, K., Kondrashov, D., Mann, M., Robertson, A. W., Saunders, A., Tian, Y., Varadi, F. et al. (2002), 'Advanced spectral methods for climatic time series', *Reviews of geophysics* **40**(1), 3–1.

Gibbs, M. and MacKay, D. J. (1997), 'Efficient implementation of gaussian processes'.

Girolami, G. and Vakman, D. (2002), 'Instantaneous frequency estimation and measurement: a quasi-local method', *Measurement Science and Technology* **13**(6), 909.

Glover, F. and Laguna, M. (1997), 'Modern heuristic techniques for combinatorial optimization'.

Goldberg, D. E. and Holland, J. H. (1988), 'Genetic algorithms and machine learning'.

Gönen, M. and Alpaydın, E. (2011*a*), 'Multiple kernel learning algorithms', *The Journal of Machine Learning Research* **12**, 2211–2268.

Gönen, M. and Alpaydın, E. (2011*b*), 'Multiple kernel learning algorithms', *Journal of machine learning research* **12**(Jul), 2211–2268.

Gulzar, T., Singh, A. and Sharma, S. (2014), 'Comparative analysis of lpcc, mfcc and bfcc for the recognition of hindi words using artificial neural networks', *International Journal of Computer Applications* **101**(12), 22–27.

Guyon, I. and Elisseeff, A. (2006), An introduction to feature extraction, *in* 'Feature extraction', Springer, pp. 1–25.

Hainsworth, S. W. and Macleod, M. D. (2003), 'Time frequency reassignment: A review and analysis'.

Harel, B., Cannizzaro, M. and Snyder, P. J. (2004), 'Variability in fundamental frequency during speech in prodromal and incipient parkinson's disease: A longitudinal case study', *Brain and cognition* **56**(1), 24–29.

Hartelius, L. and Svensson, P. (1994), 'Speech and swallowing symptoms associated with parkinson's disease and multiple sclerosis: a survey', *Folia phoniatrica et logopaedica* **46**(1), 9–17.

Hasan, T. and Hansen, J. H. (2011), Robust speaker recognition in non-stationary room environments based on empirical mode decomposition, *in* 'Twelfth Annual Conference of the International Speech Communication Association'.

Hegde, R. M., Murthy, H. A. and Gadde, V. R. R. (2007), 'Significance of the modified group delay feature in speech recognition', *IEEE Transactions on Audio, Speech, and Language Processing* **15**(1), 190–202.

Herbster, M., Pontil, M. and Wainer, L. (2005), Online learning over graphs, *in* 'Proceedings of the 22nd international conference on Machine learning', pp. 305–312.

Hermansky, H., Morgan, N., Bayya, A. and Kohn, P. (1991), Rasta-plp speech analysis, *in* 'Proc. IEEE Int'l Conf. Acoustics, speech and signal processing', Vol. 1, Citeseer, pp. 121–124.

Hinton, G. E. and Roweis, S. T. (2003), Stochastic neighbor embedding, *in* 'Advances in neural information processing systems', pp. 857–864.

Hinton, G. and Roweis, S. T. (2002), Stochastic neighbor embedding, *in* 'NIPS', Vol. 15, Citeseer, pp. 833–840.

Hlawatsch, F., Boudreaux-Bartels, G. F. et al. (1992), 'Linear and quadratic time-frequency signal representations', *IEEE signal processing magazine* **9**(2), 21–67.

Ho, A. K., Iansek, R., Marigliani, C., Bradshaw, J. L. and Gates, S. (1998), 'Speech impairment in a large sample of patients with parkinson's disease', *Behavioural neurology* **11**(3), 131–137.

Hoehn, M. M., Yahr, M. D. et al. (1998), 'Parkinsonism: onset, progression, and mortality', *Neurology* **50**(2), 318–318.

Hoi, S. C. and Jin, R. (2008), Active kernel learning, *in* 'Proceedings of the 25th international conference on Machine learning', pp. 400–407.

Hoi, S. C., Jin, R. and Lyu, M. R. (2007), Learning nonparametric kernel matrices from pairwise constraints, *in* 'Proceedings of the 24th international conference on Machine learning', pp. 361–368.

Homem-de Mello, T. and Rubinstein, R. Y. (2002), 'Rare event estimation for static models via cross-entropy and importance sampling'.

Hotelling, H. (1933), 'Analysis of a complex of statistical variables into principal components.', *Journal of educational psychology* **24**(6), 417.

Huang, N. E. (2014), *Hilbert-Huang transform and its applications*, Vol. 16, World Scientific.

Huang, N. E., Shen, Z., Long, S. R., Wu, M. C., Shih, H. H., Zheng, Q., Yen, N.-C., Tung, C. C. and Liu, H. H. (1998), 'The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis', *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* **454**(1971), 903–995.

Huang, N., Long, S. and Shen, Z. (1996), 'The mechanism for frequency downshift in nonlinear wave evolution', *Advances in applied mechanics* **32**, 59–117.

Huang, N., Shen, Z. and Long, S. (1999), 'A new view of nonlinear water waves: The Hilbert Spectrum 1', *Annual Reviews in Fluid Mechanics* **31**(1), 417–457.

Huang, T.-l., Ren, W.-x. and Lou, M.-l. (2008), The orthogonal hilbert-huang transform and its application in earthquake motion recordings analysis, *in* '14th World Conference on Earthquake Engineering, Beijing', pp. 12–17.

Huang, X., Acero, A., Hon, H.-W. and Foreword By-Reddy, R. (2001), *Spoken language processing: A guide to theory, algorithm, and system development*, Prentice hall PTR.

Huber, J. E., Stathopoulos, E. T., Curione, G. M., Ash, T. A. and Johnson, K. (1999), 'Formants of children, women, and men: The effects of vocal intensity variation', *The Journal of the Acoustical Society of America* **106**(3), 1532–1542.

Huhle, B., Schairer, T., Schilling, A. and Straßer, W. (2010), Learning to localize with gaussian process regression on omnidirectional image data, *in* '2010 IEEE/RSJ International Conference on Intelligent Robots and Systems', IEEE, pp. 5208–5213.

Hunt, A. J. and Black, A. W. (1996), Unit selection in a concatenative speech synthesis system using a large speech database, *in* '1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings', Vol. 1, IEEE, pp. 373–376.

Ismail, S. and Shabri, A. (2017), Combination model of empirical mode decomposition and svm for river flow forecasting, *in* 'AIP Conference Proceedings', Vol. 1830, AIP Publishing LLC, p. 080005.

Ismail, S., Shabri, A. and Abadan, S. S. (2015), Empirical mode decomposition coupled with least square support vector machine for river flow forecasting, *in* 'AIP Conference Proceedings', Vol. 1643, American Institute of Physics, pp. 232–241.

Jaakkola, T., Diekhans, M. and Haussler, D. (2000), 'A discriminative framework for detecting remote protein homologies', *Journal of computational biology* **7**(1-2), 95–114.

Jaakkola, T. and Haussler, D. (1999*a*), Exploiting generative models in discriminative classifiers, *in* 'Advances in neural information processing systems', pp. 487–493.

Jaakkola, T. S., Diekhans, M. and Haussler, D. (1999), Using the fisher kernel method to detect remote protein homologies., *in* 'ISMB', Vol. 99, pp. 149–158.

Jaakkola, T. S. and Haussler, D. (1999*b*), Probabilistic kernel regression models, *in* 'Seventh International Workshop on Artificial Intelligence and Statistics', PMLR.

Jaakkola, T. S., Haussler, D. et al. (1999), 'Exploiting generative models in discriminative classifiers', *Advances in neural information processing systems* pp. 487–493.

Jacobs, R. A. (1988), 'Increased rates of convergence through learning rate adaptation', *Neural networks* **1**(4), 295–307.

Jannetts, S. and Lowit, A. (2014), 'Cepstral analysis of hypokinetic and ataxic voices: correlations with perceptual and other acoustic measures', *Journal of Voice* **28**(6), 673–680.

Jawanpuria, P., Nath, J. S. and Ramakrishnan, G. (2015), 'Generalized hierarchical kernel learning', *Journal of Machine Learning Research* **16**(20), 617–652.

Jeevan, M., Dhingra, A., Hanmandlu, M. and Panigrahi, B. (2017), Robust speaker verification using gfcc based i-vectors, *in* 'Proceedings of the International Conference on Signal, Networks, Computing, and Systems', Springer, pp. 85–91.

Jung, Y., Choi, Y., Lim, H. and Kim, H. (2020), 'A unified deep learning framework for short-duration speaker verification in adverse environments', *IEEE Access* **8**, 175448–175466.

Junsheng, C., Dejie, Y. and Yu, Y. (2006), 'Research on the intrinsic mode function (imf) criterion in emd method', *Mechanical systems and signal processing* **20**(4), 817–824.

Kabir, M. M., Mridha, M., Shin, J., Jahan, I. and Ohi, A. Q. (2021), 'A survey of speaker recognition: Fundamental theories, recognition methods and opportunities', *IEEE Access* .

Kachiashvili, K. J. and Prangishvili, A. I. (2018), 'Verification in biometric systems: problems and modern methods of their solution', *Journal of Applied Statistics* **45**(1), 43–62.

Kamble, M. R., Sailor, H. B., Patil, H. A. and Li, H. (2020), 'Advances in anti-spoofing: from the perspective of asvspoof challenges', *APSIPA Transactions on Signal and Information Processing* **9**.

Kashyap, B., Pathirana, P. N., Horne, M., Power, L. and Szmulewicz, D. (2020), 'Quantitative assessment of speech in cerebellar ataxia using magnitude and phase based cepstrum', *Annals of biomedical engineering* **48**(4), 1322–1336.

Keith, J. and Kroese, D. P. (2002), Sequence alignment by rare event simulation, *in* 'Proceedings of the Winter Simulation Conference', Vol. 1, IEEE, pp. 320–327.

Kent, R. D., Kent, J. F., Duffy, J. R., Thomas, J. E., Weismer, G. and Stuntebeck, S. (2000), 'Ataxic dysarthria', *Journal of Speech, Language, and Hearing Research* **43**(5), 1275–1289.

Kim, D. and Oh, H.-S. (2009), 'Emd: A package for empirical mode decomposition and hilbert spectrum.', *R J.* **1**(1), 40.

Kinnunen, T. and Alku, P. (2009), On separating glottal source and vocal tract information in telephony speaker verification, *in* '2009 IEEE International Conference on Acoustics, Speech and Signal Processing', IEEE, pp. 4545–4548.

Kinnunen, T., Lee, K. A., Delgado, H., Evans, N., Todisco, M., Sahidullah, M., Yamagishi, J. and Reynolds, D. A. (2018), 't-dcf: a detection cost function for the tandem assessment of spoofing countermeasures and automatic speaker verification', *arXiv preprint arXiv:1804.09618* .

Kinnunen, T. and Li, H. (2010), 'An overview of text-independent speaker recognition: From features to supervectors', *Speech communication* **52**(1), 12–40.

Kinnunen, T., Sahidullah, M., Delgado, H., Todisco, M., Evans, N., Yamagishi, J. and Lee, K. A. (2017), 'The asvspoof 2017 challenge: Assessing the limits of replay spoofing attack detection'.

Kisi, O., Latifoğlu, L. and Latifoğlu, F. (2014), 'Investigation of empirical mode decomposition in forecasting of hydrological time series', *Water resources management* **28**(12), 4045–4057.

Kodera, K., De Villedary, C. and Gendrin, R. (1976), 'A new method for the numerical analysis of non-stationary signals', *Physics of the Earth and Planetary Interiors* **12**(2-3), 142–150.

Kroese, D. P., Rubinstein, R. Y., Cohen, I., Porotsky, S. and Taimre, T. (2011), 'Cross-entropy method", *European Journal of Operational Research* **31**, 276–283.

Kumar, P. and Lahudkar, S. (2015), 'Automatic speaker recognition using lpcc and mfcc', *International Journal on Recent and Innovation Trends in Computing and Communication* **3**(4), 2106–2109.

Laitinen, M.-V., Disch, S. and Pulkki, V. (2013), 'Sensitivity of human hearing to changes in phase spectrum', *Journal of the Audio Engineering Society* **61**(11), 860–877.

Lanckriet, G. R., Cristianini, N., Bartlett, P., Ghaoui, L. E. and Jordan, M. I. (2004), 'Learning the kernel matrix with semidefinite programming', *Journal of Machine learning research* **5**(Jan), 27–72.

Lang, A. E. and Lozano, A. M. (1998), 'Parkinson's disease', *New England Journal of Medicine* **339**(16), 1130–1143.

Lázaro-Gredilla, M., Quiñonero-Candela, J., Rasmussen, C. E. and Figueiras-Vidal, A. R. (2010), 'Sparse spectrum gaussian process regression', *The Journal of Machine Learning Research* **11**, 1865–1881.

Le Van Quyen, M., Foucher, J., Lachaux, J.-P., Rodriguez, E., Lutz, A., Martinerie, J. and Varela, F. J. (2001), 'Comparison of hilbert transform and wavelet methods for the analysis of neuronal synchrony', *Journal of neuroscience methods* **111**(2), 83–98.

Li, R. and Wang, Y. (2008), Short-term wind speed forecasting for wind farm based on empirical mode decomposition, *in* '2008 International Conference on Electrical Machines and Systems', IEEE, pp. 2521–2525.

Liang, H., Bressler, S. L., Buffalo, E. A., Desimone, R. and Fries, P. (2005), 'Empirical mode decomposition of field potentials from macaque v4 in visual spatial attention', *Biological cybernetics* **92**(6), 380–392.

Lin, C.-S., Chiu, S.-H. and Lin, T.-Y. (2012), 'Empirical mode decomposition–based least squares support vector regression for foreign exchange rate forecasting', *Economic Modelling* **29**(6), 2583–2590.

Lin, L. and Dunson, D. B. (2014), 'Bayesian monotone regression using gaussian process projection', *Biometrika* **101**(2), 303–317.

Lin, L. and Hongbing, J. (2009), 'Signal feature extraction based on an improved emd method', *Measurement* **42**(5), 796–803.

Liu, G. K. (2018), 'Evaluating gammatone frequency cepstral coefficients with neural networks for emotion recognition from speech', *arXiv preprint arXiv:1806.09010* .

Logemann, J. A., Fisher, H. B., Boshes, B. and Blonsky, E. R. (1978), 'Frequency and cooccurrence of vocal tract dysfunctions in the speech of a large sample of parkinson patients', *Journal of Speech and hearing Disorders* **43**(1), 47–57.

Longworth, C. and Gales, M. J. (2008), Multiple kernel learning for speaker verification, *in* '2008 IEEE International Conference on Acoustics, Speech and Signal Processing', IEEE, pp. 1581–1584.

Luna-Webb, S. (2015), 'Comparison of acoustic measures in discriminating between those with friedreich's ataxia and neurologically normal peers'.

Maaten, L. v. d. and Hinton, G. (2008), 'Visualizing data using t-sne', *Journal of machine learning research* **9**(Nov), 2579–2605.

MacKay, D. J. (1992), 'A practical bayesian framework for backpropagation networks', *Neural computation* **4**(3), 448–472.

MacKay, D. J. (1997), 'Gaussian processes-a replacement for supervised neural networks?'.

Mainardi, F. and Rogosin, S. (2008), 'The origin of infinitely divisible distributions: from de finetti's problem to levy-khintchine formula', *arXiv preprint arXiv:0801.1910* .

Manjula, M., Sarma, A. and Mishra, S. (2011), Detection and classification of voltage sag causes based on empirical mode decomposition, *in* '2011 Annual IEEE India Conference', IEEE, pp. 1–5.

Martínez-Martín, P., Gil-Nagel, A., Gracia, L. M., Gómez, J. B., Martinez-Sarries, J., Bermejo, F. and Group, C. M. (1994), 'Unified parkinson's disease rating scale characteristics and structure', *Movement disorders* **9**(1), 76–83.

Massouleh, S. M. and Kordkheili, S. H. (2019), 'Experimental investigation of empirical mode decomposition by reduction of end effect error', *Physica A: Statistical Mechanics and its Applications* **534**, 122171.

Matrouf, D., Bonastre, J.-F. and Fredouille, C. (2006), Effect of speech transformation on impostor acceptance, *in* '2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings', Vol. 1, IEEE, pp. I–I.

Mazaira-Fernandez, L. M., Álvarez-Marquina, A. and Gómez-Vilda, P. (2015), 'Improving speaker recognition by biometric voice deconstruction', *Frontiers in Bioengineering and Biotechnology* **3**, 126.
URL: *https://www.frontiersin.org/article/10.3389/fbioe.2015.00126*

Melo, J. (2012), Gaussian processes for regression: a tutorial, *in* 'Technical Report', University of Porto.

Melville, W. (1983), 'Wave modulation and breakdown', *Journal of Fluid Mechanics* **128**, 489–506.

Mendoza, E., Valencia, N., Muñoz, J. and Trujillo, H. (1996), 'Differences in voice quality between men and women: Use of the long-term average spectrum (ltas)', *Journal of Voice* **10**(1), 59–66.

Micchelli, C. A., Xu, Y. and Zhang, H. (2006), 'Universal kernels.', *Journal of Machine Learning Research* **7**(12).

Ming, D., Williamson, D. and Guillas, S. (2021), 'Deep gaussian process emulation using stochastic imputation', *arXiv preprint arXiv:2107.01590* .

Mobile Device Voice Recordings at King's College London(MDVR-KCL) from both early and advanced Parkinson's disease patients and healthy controls (2019).
URL: *https://zenodo.org/record/2867216.YG7HhuhKjD4*

Moreno, P. J. and Rifkin, R. (2000), Using the fisher kernel method for web audio classification, *in* '2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 00CH37100)', Vol. 4, IEEE, pp. 2417–2420.

Morise, M., Yokomori, F. and Ozawa, K. (2016), 'World: a vocoder-based high-quality speech synthesis system for real-time applications', *IEICE TRANSACTIONS on Information and Systems* **99**(7), 1877–1884.

Neal, R. M. (1993), Bayesian learning via stochastic dynamics, *in* 'Advances in neural information processing systems', pp. 475–482.

Neal, R. M. (1996), Priors for infinite networks, *in* 'Bayesian Learning for Neural Networks', Springer, pp. 29–53.

Neal, R. M. (2012), *Bayesian learning for neural networks*, Vol. 118, Springer Science & Business Media.

Nguyen, C. H. and Ho, T. B. (2007), Kernel matrix evaluation., *in* 'IJCAI', pp. 987–992.

Np, N., Schuller, B. and Alku, P. (2021), 'The detection of parkinsons disease from speech using voice source information', *IEEE/ACM Transactions on Audio, Speech, and Language Processing* .

of Electrical, I. and Engineers, E. (1969), 'Ieee recommended practice for speech quality measurements', *IEEE transactions on audio and electroacoustics* **17**(3), 225–246.

Ohi, A. Q., Mridha, M., Hamid, M. A. and Monowar, M. M. (2021), 'Deep speaker recognition: Process, progress, and challenges', *IEEE Access* **9**, 89619–89643.

Okolobah, V. A. and Ismail, Z. (2013), 'New approach to peak load forecasting based on emd and anfis', *Indian Journal of Science and Technology* **6**(12), 5600–6.

on Rating Scales for Parkinson's Disease, M. D. S. T. F. (2003), 'The unified parkinson's disease rating scale (updrs): status and recommendations', *Movement Disorders* **18**(7), 738–750.

O'Sullivan, F. (1986), 'A statistical perspective on ill-posed inverse problems (with discussion)', *Stat sci* **1**, 505–527.

Packard, N. H., Crutchfield, J. P., Farmer, J. D. and Shaw, R. S. (1980), 'Geometry from a time series', *Physical review letters* **45**(9), 712.

Paliwal, K. K. and Alsteris, L. (2003), Usefulness of phase spectrum in human speech perception, *in* 'Eighth European Conference on Speech Communication and Technology'.

Papoulis, A. (1977), *Signal analysis*, McGraw-Hill.

Parra, G. and Tobar, F. (2017), 'Spectral mixture kernels for multi-output gaussian processes', *arXiv preprint arXiv:1709.01298* .

Partila, P., Tovarek, J., Ilk, G. H., Rozhon, J. and Voznak, M. (2020), 'Deep learning serves voice cloning: how vulnerable are automatic speaker verification systems to spoofing trials?', *IEEE Communications Magazine* **58**(2), 100–105.

Patel, T. B. and Patil, H. A. (2017), 'Significance of source–filter interaction for classification of natural vs. spoofed speech', *IEEE Journal of Selected Topics in Signal Processing* **11**(4), 644–659.

Peters, G. (2017), 'Statistical machine learning and data analytic methods for risk and insurance', *Available at SSRN 3050592* .

Pompili, A., Solera-Urena, R., Abad, A., Cardoso, R., Guimaraes, I., Fabbri, M., Martins, I. P. and Ferreira, J. (2020), 'Assessment of parkinson's disease medication state through automatic speech analysis', *arXiv preprint arXiv:2005.14647* .

Potter, S. (1999), 'Nonlinear time series modelling: An introduction', *Journal of Economic Surveys* **13**(5), 505–528.

Qian, S. and Chen, D. (1996), *Joint time-frequency analysis: methods and applications*, Prentice-Hall, Inc.

Qin, S. and Zhong, Y. M. (2006), 'A new envelope algorithm of hilbert–huang transform', *Mechanical Systems and Signal Processing* **20**(8), 1941–1952.

Rahimi, A., Recht, B. et al. (2007), Random features for large-scale kernel machines., *in* 'NIPS', Vol. 3, Citeseer, p. 5.

Raina, R., Battle, A., Lee, H., Packer, B. and Ng, A. Y. (2007), Self-taught learning: transfer learning from unlabeled data, *in* 'Proceedings of the 24th international conference on Machine learning', pp. 759–766.

Ramachandran, R. P., Farrell, K. R., Ramachandran, R. and Mammone, R. J. (2002), 'Speaker recognition general classifier approaches and data fusion methods', *Pattern Recognition* **35**(12), 2801–2821.

Rasmussen, C. E. and Williams, C. K. I. (2005), *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*, The MIT Press.

Remes, S., Heinonen, M. and Kaski, S. (2017), 'Non-stationary spectral kernels', *arXiv preprint arXiv:1705.08736* .

Rice, S. O. (1944), 'Mathematical analysis of random noise', *The Bell System Technical Journal* **23**(3), 282–332.

Rice, S. O. (1945), 'Mathematical analysis of random noise', *Bell system technical journal* **24**(1), 46–156.

Rilling, G., Flandrin, P., Goncalves, P. et al. (2003), On empirical mode decomposition and its algorithms, *in* 'IEEE-EURASIP workshop on nonlinear signal and image processing', Vol. 3, Citeseer, pp. 8–11.

Ripley, B. D. (2007), *Pattern recognition and neural networks*, Cambridge university press.

Rosenberg, A. E. (1992), 'Recent research in automatic speaker recognition', *Advances in speech signal processing* .

Rubinstein, R. (1999), 'The cross-entropy method for combinatorial and continuous optimization', *Methodology and computing in applied probability* **1**(2), 127–190.

Rubinstein, R. Y. (1997), 'Optimization of computer simulation models with rare events', *European Journal of Operational Research* **99**(1), 89–112.

Rubinstein, R. Y. (2001), Combinatorial optimization, cross-entropy, ants and rare events, *in* 'Stochastic optimization: algorithms and applications', Springer, pp. 303–363.

Rubinstein, R. Y. and Kroese, D. P. (2004), *The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation, and machine learning*, Vol. 133, Springer.

Ruelle, D. (1980), 'Strange attractors', *Math. Intelligencer* **2**, 37–48.

Sahidullah, M., Kinnunen, T. and Hanilçi, C. (2015), 'A comparison of features for synthetic speech detection'.

Saito, Y., Takamichi, S. and Saruwatari, H. (2017), 'Statistical parametric speech synthesis incorporating generative adversarial networks', *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **26**(1), 84–96.

Salah, K. and Halim, S. (2020), Kernel function and dimensionality reduction effects on speaker verification system, *in* '2020 International Conference on Electrical Engineering (ICEE)', IEEE, pp. 1–4.

Salakhutdinov, R. and Hinton, G. E. (2007), Using deep belief nets to learn covariance kernels for gaussian processes., *in* 'NIPS', Vol. 7, Citeseer, pp. 1249–1256.

Salomon, D. (2011), *The Computer Graphics Manual*, Springer.

Sammon, J. W. (1969), 'A nonlinear mapping for data structure analysis', *IEEE Transactions on computers* **100**(5), 401–409.

Samo, Y.-L. K. and Roberts, S. (2015), 'Generalized spectral kernels', *arXiv preprint arXiv:1506.02236* .

Sandsten, M. (2016), 'Time-frequency analysis of time-varying signals and non-stationary processes', *Lund University* .

Sapir, S., Pawlas, A., Ramig, L., Countryman, S., O'BRIEN, C., Hoehn, M. and Thompson, L. (1999), 'Speech and voice abnormalities in parkinson disease: relation to severity of motor impairment, duration of disease, medication, depression, gender and age', *NCVS Status and Progress Report* **14**, 149–161.

Sauer, A., Gramacy, R. B. and Higdon, D. (2021), 'Active learning for deep gaussian process surrogates', *Technometrics* (just-accepted), 1–39.

Schlotthauer, G., Torres, M. E. and Rufiner, H. L. (2009), A new algorithm for instantaneous f 0 speech extraction based on ensemble empirical mode decomposition, *in* '2009 17th European Signal Processing Conference', IEEE, pp. 2347–2351.

Schmitz-Hübsch, T., Du Montcel, S. T., Baliko, L., Berciano, J., Boesch, S., Depondt, C., Giunti, P., Globas, C., Infante, J., Kang, J.-S. et al. (2006), 'Scale for the assessment and rating of ataxia: development of a new clinical scale', *Neurology* **66**(11), 1717–1720.

Schölkopf, B., Smola, A. J., Bach, F. et al. (2002), *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT press.

Scholl, S. (2021), 'Fourier, gabor, morlet or wigner: Comparison of time-frequency transforms', *arXiv preprint arXiv:2101.06707* .

Schroeder, M. R. (1959), 'New results concerning monaural phase sensitivity', *The Journal of the Acoustical Society of America* **31**(11), 1579–1579.

Schwartz, M., Bennett, W. R. and Stein, S. (1996), *Communication systems and techniques*, John Wiley & Sons.

Sethu, V., Ambikairajah, E. and Epps, J. (2008), Empirical mode decomposition based weighted frequency feature for speech-based emotion classification, *in* '2008 IEEE International Conference on Acoustics, Speech and Signal Processing', IEEE, pp. 5017–5020.

Sewell, M. (2011), 'The fisher kernel: a brief review', *RN* **11**(06), 06.

Shannon, B. J. and Paliwal, K. K. (2003), A comparative study of filter bank spacing for speech recognition, *in* 'Microelectronic engineering research conference', Vol. 41, Citeseer, pp. 310–12.

Sharma, R., Vignolo, L., Schlotthauer, G., Colominas, M. A., Rufiner, H. L. and Prasanna, S. (2017*a*), 'Empirical mode decomposition for adaptive am-fm analysis of speech: A review', *Speech Communication* **88**, 39–64.

Sharma, R., Vignolo, L., Schlotthauer, G., Colominas, M. A., Rufiner, H. L. and Prasanna, S. (2017*b*), 'Empirical mode decomposition for adaptive am-fm analysis of speech: A review', *Speech Communication* **88**, 39–64.

Shawe-Taylor, J., Cristianini, N. et al. (2004), *Kernel methods for pattern analysis*, Cambridge university press.

Sigurdsson, S., Petersen, K. B. and Lehn-Schiøler, T. (2006), Mel frequency cepstral coefficients: An evaluation of robustness of mp3 encoded music., *in* 'ISMIR', pp. 286–289.

Singh, N., Pillay, V. and Choonara, Y. E. (2007), 'Advances in the treatment of parkinson's disease', *Progress in neurobiology* **81**(1), 29–44.

Sinha, A. and Duchi, J. C. (2016), Learning kernels with random features., *in* 'NIPS', pp. 1298–1306.

Skodda, S., Rinsche, H. and Schlegel, U. (2009), 'Progression of dysprosody in parkinson's disease over time—a longitudinal study', *Movement disorders: official journal of the Movement Disorder Society* **24**(5), 716–722.

Smith, N. and Gales, M. (2001), Speech recognition using svms, *in* 'NIPS'.

Smith, N. and Niranjan, M. (2000), Data-dependent kernels in svm classification of speech patterns, *in* 'Sixth International Conference on Spoken Language Processing'.

Sriskandaraja, K., Sethu, V., Ambikairajah, E. and Li, H. (2016), 'Front-end for antispoofing countermeasures in speaker verification: Scattering spectral decomposition', *IEEE Journal of Selected Topics in Signal Processing* **11**(4), 632–643.

Steutel, F. W., Kent, J. T., Bondesson, L. and Barndorff-Nielsen, O. (1979), 'Infinite divisibility in theory and practice [with discussion and reply]', *Scandinavian Journal of Statistics* pp. 57–64.

Steutel, F. W. and Van Harn, K. (2003), *Infinite divisibility of probability distributions on the real line*, CRC Press.

Stolarski, Ł. (2017), 'Intensity of the reader's voice in the reading aloud of fiction: Effects of the character's gender', *Studia Anglica Posnaniensia* **52**(3), 285–323.

Sun, W. and Wang, Y. (2018), 'Short-term wind speed forecasting based on fast ensemble empirical mode decomposition, phase space reconstruction, sample entropy and improved back-propagation neural network', *Energy Conversion and Management* **157**, 1–12.

Tabet, Y. and Boughazi, M. (2011), Speech synthesis techniques. a survey, *in* 'International Workshop on Systems, Signal Processing and their Applications, WOSSPA', IEEE, pp. 67–70.

Tanabe, H., Ho, T. B., Nguyen, C. H. and Kawasaki, S. (2008), Simple but effective methods for combining kernels in computational biology, *in* '2008 IEEE International Conference on Research, Innovation and Vision for the Future in Computing and Communication Technologies', IEEE, pp. 71–78.

Tao, X., Chongguang, L. and Yukun, B. (2017), 'An improved eemd-based hybrid approach for the short-term forecasting of hog price in china', *Agricultural Economics* **63**(3), 136–148.

Tapkir, P. A., Patil, A. T., Shah, N. and Patil, H. A. (2018), Novel spectral root cepstral features for replay spoof detection, *in* '2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)', IEEE, pp. 1945–1950.

Tapkir, P. and Patil, H. A. (2018), Novel empirical mode decomposition cepstral features for replay spoof detection., *in* 'Interspeech', pp. 721–725.

Thaine, P. and Penn, G. (2019), Extracting mel-frequency and bark-frequency cepstral coefficients from encrypted signals., *in* 'INTERSPEECH', pp. 3715–3719.

Thayaparan, T. (2000), Linear and quadratic time-frequency representations, Technical report, DEFENCE RESEARCH ESTABLISHMENT OTTAWA (ONTARIO).

Titchmarsh, E. C. (1948), 'Introduction to the theory of fourier integrals'.

Tobar, F., Bui, T. D. and Turner, R. E. (2015), 'Learning stationary time series using gaussian processes with nonparametric kernels', *Advances in Neural Information Processing Systems* **28**, 3501–3509.

Todisco, M., Wang, X., Vestman, V., Sahidullah, M., Delgado, H., Nautsch, A., Yamagishi, J., Evans, N., Kinnunen, T. and Lee, K. A. (2019), 'Asvspoof 2019: Future horizons in spoofed and fake audio detection', *arXiv preprint arXiv:1904.05441* .

Tompkins, A. and Ramos, F. (2018), Fourier feature approximations for periodic kernels in time-series modelling, *in* 'Proceedings of the AAAI Conference on Artificial Intelligence', Vol. 32.

Tong, Y. L. (2012), *The multivariate normal distribution*, Springer Science & Business Media.

Torgerson, W. S. (1952), 'Multidimensional scaling: I. theory and method', *Psychometrika* **17**(4), 401–419.

Tsanas, A., Little, M., McSharry, P. and Ramig, L. (2009), 'Accurate telemonitoring of parkinson's disease progression by non-invasive speech tests', *Nature Precedings* pp. 1–1.

Turner, R. and Sahani, M. (2011), Probabilistic amplitude and frequency demodulation, *in* 'Advances in Neural Information Processing Systems', pp. 981–989.

ur Rehman, F., Kumar, C., Kumar, S., Mehmood, A. and Zafar, U. (2017), Vq based comparative analysis of mfcc and bfcc speaker recognition system, *in* '2017 International Conference on Information and Communication Technologies (ICICT)', IEEE, pp. 28–32.

van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A. and Kavukcuoglu, K. (2016), Wavenet: A generative model for raw audio, *in* '9th ISCA Speech Synthesis Workshop', pp. 125–125.

Van der Wilk, M., Rasmussen, C. E. and Hensman, J. (2017), 'Convolutional gaussian processes', *arXiv preprint arXiv:1709.01894* .

Vautard, R. and Ghil, M. (1989), 'Singular spectrum analysis in nonlinear dynamics, with applications to paleoclimatic time series', *Physica D: Nonlinear Phenomena* **35**(3), 395–424.

Vautard, R., Yiou, P. and Ghil, M. (1992), 'Singular-spectrum analysis: A toolkit for short, noisy chaotic signals', *Physica D: Nonlinear Phenomena* **58**(1-4), 95–126.

Vikram, C. and Umarani, K. (2013), 'Pathological voice analysis to detect neurological disorders using mfcc and svm', *Int. J. Adv. Electr. Electron. Eng* **2**(4), 87–91.

Ville, J. (1948), 'Theorie et application dela notion de signal analytique', *Câbles et transmissions* **2**(1), 61–74.

Ville, J. (1958), Theory and application of the notion of complex signal, Technical report, RAND CORP SANTA MONICA CA.

Vinokourov, A. and Girolami, M. (2001), 'Document classification employing the fisher kernel derived from probabilistic hierarchic corpus representations'.

Volodina, V. and Williamson, D. (2020), 'Diagnostics-driven nonstationary emulators using kernel mixtures', *SIAM/ASA Journal on Uncertainty Quantification* **8**(1), 1–26.

Wahlberg, P. and Schreier, P. J. (2010), 'On the instantaneous frequency of gaussian stochastic processes', *arXiv preprint arXiv:1007.1069* .

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S. et al. (2017), 'Tacotron: Towards end-to-end speech synthesis', *arXiv preprint arXiv:1703.10135* .

Whitham, G. B. (2011), *Linear and nonlinear waves*, Vol. 42, John Wiley & Sons.

Wigner, E. P. (1997), On the quantum correction for thermodynamic equilibrium, *in* 'Part I: Physical Chemistry. Part II: Solid State Physics', Springer, pp. 110–120.

Williams, C. K. (1998), Prediction with gaussian processes: From linear regression to linear prediction and beyond, *in* 'Learning in graphical models', Springer, pp. 599–621.

Wilson, A. and Adams, R. (2013), Gaussian process kernels for pattern discovery and extrapolation, *in* 'International conference on machine learning', pp. 1067–1075.

Wilson, A. G. (2014), Covariance kernels for fast automatic pattern discovery and extrapolation with Gaussian processes, PhD thesis, Citeseer.

Wilson, A. G., Gilboa, E., Cunningham, J. P. and Nehorai, A. (2014), Fast kernel learning for multidimensional pattern extrapolation., *in* 'NIPS', pp. 3626–3634.

Wilson, A. G., Knowles, D. A. and Ghahramani, Z. (2011), 'Gaussian process regression networks', *arXiv preprint arXiv:1110.4411* .

Wu, K.-H. and Chen, C.-P. (2010), Empirical mode decomposition for noise-robust automatic speech recognition, *in* 'Eleventh Annual Conference of the International Speech Communication Association'.

Wu, Q., Zhang, L. and Xia, B. (2008), Robust auditory-based speech feature extraction using independent subspace method, *in* 'Advances in Cognitive Neurodynamics ICCN 2007', Springer, pp. 405–409.

Wu, Z., Evans, N., Kinnunen, T., Yamagishi, J., Alegre, F. and Li, H. (2015), 'Spoofing and countermeasures for speaker verification: A survey', *speech communication* **66**, 130–153.

Wu, Z., Kinnunen, T., Evans, N., Yamagishi, J., Hanilçi, C., Sahidullah, M. and Sizov, A. (2015), Asvspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge, *in* 'Sixteenth Annual Conference of the International Speech Communication Association'.

Wu, Z., Yamagishi, J., Kinnunen, T., Hanilçi, C., Sahidullah, M., Sizov, A., Evans, N., Todisco, M. and Delgado, H. (2017), 'Asvspoof: The automatic speaker verification spoofing and countermeasures challenge', *IEEE Journal of Selected Topics in Signal Processing* **11**(4), 588–604.

Xia, C. and Wang, Z. (2020), 'Drivers analysis and empirical mode decomposition based forecasting of energy consumption structure', *Journal of Cleaner Production* **254**, 120107.

Xiong, T., Bao, Y. and Hu, Z. (2014), 'Does restraining end effect matter in emd-based modeling framework for time series prediction? some experimental evidences', *Neurocomputing* **123**, 174–184.

Yaglom, A. (2012), *Correlation theory of stationary and related random functions: Supplementary notes and references*, Springer Science & Business Media.

Yahya, N. A., Samsudin, R. and Shabri, A. (2017), 'Tourism forecasting using hybrid modified empirical mode decomposition and neural network', *Int. J. Advance Soft Compu. Appl* **9**(1), 14–31.

Yamagishi, J., Kinnunen, T. H., Evans, N., De Leon, P. and Trancoso, I. (2017), 'Introduction to the issue on spoofing and countermeasures for automatic speaker verification', *IEEE Journal of Selected Topics in Signal Processing* **11**(4), 585–587.

Yang, H.-L. and Lin, H.-C. (2016), 'An integrated model combined arima, emd with svr for stock indices forecasting', *International Journal on Artificial Intelligence Tools* **25**(02), 1650005.

Yang, Z., Wilson, A., Smola, A. and Song, L. (2015), A la carte–learning fast kernels, *in* 'Artificial Intelligence and Statistics', pp. 1098–1106.

Zaremba, A. and Peters, G. (2020), 'Statistical causality for multivariate nonlinear time series via gaussian processes', *Available at SSRN 3609497* .

Zen, H., Tokuda, K. and Black, A. W. (2009), 'Statistical parametric speech synthesis', *speech communication* **51**(11), 1039–1064.

Zhang, J., Marszalek, M., Lazebnik, S. and Schmid, C. (2007), 'Local features and kernels for classification of texture and object categories: A comprehensive study', *International journal of computer vision* **73**(2), 213–238.

Zhang, Z. and Hong, W.-C. (2019), 'Electric load forecasting by complete ensemble empirical mode decomposition adaptive noise and support vector regression with quantum-based dragonfly algorithm', *Nonlinear Dynamics* **98**(2), 1107–1136.

Zhao, X., Chen, X., Xu, Y., Xi, D., Zhang, Y. and Zheng, X. (2017), 'An emd-based chaotic least squares support vector machine hybrid model for annual runoff forecasting', *Water* **9**(3), 153.

Zhao, Y., Atlas, L. E. and Marks, R. J. (1990), 'The use of cone-shaped kernels for generalized time-frequency representations of nonstationary signals', *IEEE Transactions on Acoustics, Speech, and Signal Processing* **38**(7), 1084–1091.

Zheng, N., Lee, T. and Ching, P.-C. (2007), 'Integration of complementary acoustic features for speaker recognition', *IEEE Signal Processing Letters* **14**(3), 181–184.

Zheng, Z.-W., Chen, Y.-Y., Zhou, X.-W., Huo, M.-M., Zhao, B. and Guo, M. (2013), 'Short-term wind power forecasting using empirical mode decomposition and rbfnn', *International Journal of Smart Grid and Clean Energy* **2**(2), 192–99.

Zhu, B., Han, D., Wang, P., Wu, Z., Zhang, T. and Wei, Y.-M. (2017), 'Forecasting carbon price using empirical mode decomposition and evolutionary least squares support vector regression', *Applied energy* **191**, 521–530.

Zouhir, Y. and Ouni, K. (2016), 'Feature extraction method for improving speech recognition in noisy environments.', *J. Comput. Sci.* **12**(2), 56–61.

Appendices

Appendix A

Consider D iteration of the sifting process (i.e. producing D IMFs). The $D-th$ IMF can be represented by a spline whose coefficients are a linear combination of the coefficients of $S(t)$ and the coefficients of the mean envelopes of the previous extracted IMFs. By taking into account the first sifting process, we show the above statement and then extend it to the $D-th$ case.

Consider the spline interpolating the original set of data $x(t)$, i.e. $S(t)$. By being at the initial step of the sifting procedure, i.e. step 0, we modify such equation:

$$S(t) = \sum_{i=1}^{n-1} \left(a_i^0 t^3 + b_i^0 t^2 + c_i^0 t + d_i^0\right) \mathbb{1}\left(t \in [t_{i-1}, t_i]\right) \tag{1}$$

where the upper indices of the coefficients state the step at which the sifting procedure is. Consider the upper and lower envelope of $S(t)$. Such splines are evaluated at times of the original spline $S(t)$, i.e. at t, as:

$$S\left(e_1^S\right) = \sum_{j=1}^{N-1} \left(a_{j,0}^{e_1} t^3 + b_{j,0}^{e_1} t^2 + c_{j,0}^{e_1} t + d_{j,0}^{e_1}\right) \mathbb{1}\left(t \in [t_{j-1}, t_j]\right) \tag{2}$$

$$S\left(e_2^S\right) = \sum_{j=0}^{n-1} \left(a_{j,0}^{e_2} t^3 + b_{j,0}^{e_2} t^2 + c_{j,0}^{e_2} t + d_{j,0}^{e_2}\right) \mathbb{1}\left(t \in [t_{j-1}, t_j]\right) \tag{3}$$

Let us put $S(t) = c_0$ and $m_{1,0} = \frac{S(e_1^S)+S(e_2^S)}{2}$ its mean envelope, where the first index corresponds to the number of the IMF that the sifting procedure is extracting (the first one here) and the second one the step of this sifting. Consider a simple example where $n = 3$ and the number of steps of the sifting equal to 2. The above definitions can be represented by:

$$\begin{aligned} c_0 = S(t) = \sum_{i=1}^{N-1} \left(a_i^0 t^3 + b_i^0 t^2 + c_i^0 t + d_i^0\right) \mathbb{1}\left(t \in [t_{i-1}, t_i]\right) = \\ a_1^0 t_1^3 + b_1^0 t_1^2 + c_1^0 t_1 + d_1^0 + a_2^0 t_2^3 + b_2^0 t_2^2 + c_2^0 t_2 + d_2^0 \end{aligned} \tag{4}$$

$$\begin{aligned} m_{1,0} = \frac{S(e_1^S) + S(e_2^S)}{2} = \frac{1}{2}\left[a_{1,0}^{e_1} t_1^3 + b_{1,0}^{e_1} t_1^2 + c_{1,0}^{e_1} t_1 + d_{1,0}^{e_1} + a_{2,0}^{e_1} t_2^3 + b_{2,0}^{e_1} t_2^2 + c_{2,0}^{e_1} t_2 + d_{2,0}^{e_1}\right] + \\ \left[a_{1,0}^{e_2} t_1^3 + b_{1,0}^{e_2} t_1^2 + c_{1,0}^{e_2} t_1 + d_{1,0}^{e_2} + a_{2,0}^{e_2} t_2^3 + b_{2,0}^{e_2} t_2^2 + c_{2,0}^{e_2} t_2 + d_{2,0}^{e_2}\right] \end{aligned} \tag{5}$$

We look at the first step of the first sifting procedure computed by $c_0 - m_{1,0}$. By

rearranging and grouping we obtain:

$$\begin{aligned} c_0 - m_{1,0} = &\left[a_{1,0} - \frac{1}{2}\left(a_{1,0}^{e_1} + a_{1,0}^{e_2}\right)\right] t_1^3 + \left[a_{2,0} - \frac{1}{2}\left(a_{2,0}^{e_1} + a_{2,0}^{e_2}\right)\right] t_2^3 + \\ &\left[b_{1,0} - \frac{1}{2}\left(b_{1,0}^{e_1} + b_{1,0}^{e_2}\right)\right] t_1^2 + \left[b_{2,0} - \frac{1}{2}\left(b_{2,0}^{e_1} + b_{2,0}^{e_2}\right)\right] t_2^2 + \\ &\left[c_{1,0} - \frac{1}{2}\left(c_{1,0}^{e_1} + c_{1,0}^{e_2}\right)\right] t_1 + \left[c_{2,0} - \frac{1}{2}\left(c_{2,0}^{e_1} + c_{2,0}^{e_2}\right)\right] t_2 + \\ &\left[d_{1,0} - \frac{1}{2}\left(d_{1,0}^{e_1} + d_{1,0}^{e_2}\right)\right] + \left[d_{2,0} - \frac{1}{2}\left(d_{2,0}^{e_1} + d_{2,0}^{e_2}\right)\right] \end{aligned} \tag{6}$$

To then consider the second (and last in this simple case) step of the sifting extracting the first IMF, consider the following equalities:

- $a_1^1 = a_{1,0} - \frac{1}{2}\left(a_{1,0}^{e_1} + a_{1,0}^{e_2}\right)$
- $a_2^1 = a_{2,0} - \frac{1}{2}\left(a_{2,0}^{e_1} + a_{2,0}^{e_2}\right)$
- $b_1^1 = b_{1,0} - \frac{1}{2}\left(b_{1,0}^{e_1} + b_{1,0}^{e_2}\right)$
- $b_2^1 = b_{2,0} - \frac{1}{2}\left(b_{2,0}^{e_1} + b_{2,0}^{e_2}\right)$
- $c_1^1 = c_{1,0} - \frac{1}{2}\left(c_{1,0}^{e_1} + c_{1,0}^{e_2}\right)$
- $c_2^1 = c_{2,0} - \frac{1}{2}\left(c_{2,0}^{e_1} + c_{2,0}^{e_2}\right)$
- $d_1^1 = d_{1,0} - \frac{1}{2}\left(d_{1,0}^{e_1} + d_{1,0}^{e_2}\right)$
- $d_2^1 = d_{2,0} - \frac{1}{2}\left(d_{2,0}^{e_1} + d_{2,0}^{e_2}\right)$

By substituting the above equalities in 6, the next result is obtained:

$$c_0 - m_{0,0} = a_1^1 t_1^3 + a_2^1 t_2^3 + b_1^1 t_1^2 + b_2^1 t_2^2 + c_1^1 t_1 + c_2^1 t_2 + d_1^1 + d_2^1 \tag{7}$$

Define this intermediate step as $h_{1,1} = c_0 - m_{1,0}$ where the extracted component $h_{1,1}$ is not a final IMF yet. Therefore, the algorithm keeps running. At this stage, the mean envelope is given by $m_{1,1} = \frac{S(e_1^{h_{1,1}}) + S(e_2^{h_{1,1}})}{2}$. The second step of the sifting is:

$$\begin{aligned} h_{1,2} = h_{1,1} - m_{1,1} = &\left[a_{1,1} - \frac{1}{2}\left(a_{1,1}^{e_1} + a_{1,1}^{e_2}\right)\right] t_1^3 + \left[a_{2,1} - \frac{1}{2}\left(a_{2,1}^{e_1} + a_{2,1}^{e_2}\right)\right] t_2^3 + \\ &\left[b_{1,1} - \frac{1}{2}\left(b_{1,1}^{e_1} + b_{1,1}^{e_2}\right)\right] t_1^2 + \left[b_{2,1} - \frac{1}{2}\left(b_{2,1}^{e_1} + b_{2,1}^{e_2}\right)\right] t_2^2 + \\ &\left[c_{1,1} - \frac{1}{2}\left(c_{1,1}^{e_1} + c_{1,1}^{e_2}\right)\right] t_1 + \left[c_{2,1} - \frac{1}{2}\left(c_{2,1}^{e_1} + c_{2,1}^{e_2}\right)\right] t_2 + \\ &\left[d_{1,1} - \frac{1}{2}\left(d_{1,1}^{e_1} + d_{1,1}^{e_2}\right)\right] + \left[d_{2,1} - \frac{1}{2}\left(d_{2,1}^{e_1} + d_{2,1}^{e_2}\right)\right] \end{aligned} \tag{8}$$

Define the following equalities:

- $a_1^2 = a_{1,1} - \frac{1}{2}\left(a_{1,1}^{e_1} + a_{1,1}^{e_2}\right)$
- $a_2^2 = a_{2,1} - \frac{1}{2}\left(a_{2,1}^{e_1} + a_{2,1}^{e_2}\right)$
- $b_1^2 = b_{1,1} - \frac{1}{2}\left(b_{1,1}^{e_1} + b_{1,1}^{e_2}\right)$
- $b_2^2 = b_{2,1} - \frac{1}{2}\left(b_{2,1}^{e_1} + b_{2,1}^{e_2}\right)$
- $c_1^2 = c_{1,1} - \frac{1}{2}\left(c_{1,1}^{e_1} + c_{1,1}^{e_2}\right)$
- $c_2^2 = c_{2,1} - \frac{1}{2}\left(c_{2,1}^{e_1} + c_{2,1}^{e_2}\right)$

- $d_1^2 = d_{1,1} - \frac{1}{2}\left(d_{1,1}^{e_1} + d_{1,1}^{e_2}\right)$
- $d_2^2 = d_{2,1} - \frac{1}{2}\left(d_{2,1}^{e_1} + d_{2,1}^{e_2}\right)$

Substituting them within 8 provides the next equation:

$$h_{1,2} = h_{1,1} - m_{1,1} = a_1^2 t_1^3 + a_2^2 t_2^3 + b_1^2 t_1^2 + b_2^2 t_2^2 + c_1^2 t_1 + c_2^2 t_2 + d_1^2 + d_2^2 \quad (9)$$

This corresponds to the first extracted IMF, i.e. $c_1 = h_{1,2}$. By considering the above equation and substituting all the equalities that we have taken into account, we get:

$$\begin{aligned} c_1 = &\left[a_{1,0} - \frac{1}{2}\left(a_{1,0}^{e_1} + a_{1,0}^{e_2}\right) - \frac{1}{2}\left(a_{1,1}^{e_1} + a_{1,1}^{e_2}\right)\right] t_1^3 + \left[a_{2,0} - \frac{1}{2}\left(a_{2,0}^{e_1} + a_{2,0}^{e_2}\right) - \frac{1}{2}\left(a_{2,1}^{e_1} + a_{2,1}^{e_2}\right)\right] t_2^3 \\ &+ \left[b_{1,0} - \frac{1}{2}\left(b_{1,0}^{e_1} + b_{1,0}^{e_2}\right) - \frac{1}{2}\left(b_{1,1}^{e_1} + b_{1,1}^{e_2}\right)\right] t_1^2 + \left[b_{2,0} - \frac{1}{2}\left(b_{2,0}^{e_1} + b_{2,0}^{e_2}\right) - \frac{1}{2}\left(b_{2,1}^{e_1} + b_{2,1}^{e_2}\right)\right] t_2^2 \\ &+ \left[c_{1,0} - \frac{1}{2}\left(c_{1,0}^{e_1} + c_{1,0}^{e_2}\right) - \frac{1}{2}\left(c_{1,1}^{e_1} + c_{1,1}^{e_2}\right)\right] t_1 + \left[c_{2,0} - \frac{1}{2}\left(c_{2,0}^{e_1} + c_{2,0}^{e_2}\right) - \frac{1}{2}\left(c_{2,1}^{e_1} + c_{2,1}^{e_2}\right)\right] t_2 \\ &+ \left[d_{1,0} - \frac{1}{2}\left(d_{1,0}^{e_1} + d_{1,0}^{e_2}\right) - \frac{1}{2}\left(d_{1,1}^{e_1} + d_{1,1}^{e_2}\right)\right] + \left[d_{2,0} - \frac{1}{2}\left(d_{2,0}^{e_1} + d_{2,0}^{e_2}\right) - \frac{1}{2}\left(d_{2,1}^{e_1} + d_{2,1}^{e_2}\right)\right] \end{aligned} \quad (10)$$

A more general way to express 10 is given by:

$$\begin{aligned} c_1 = &\left\{ \sum_{n=1}^{N-1} \left[a_{i,0} - \sum_{j=0}^{G-1} \frac{1}{2}\left(a_{i,j}^{e_1} + a_{i,j}^{e_2}\right)\right] t^3 + \sum_{n=1}^{N-1} \left[b_{i,0} - \sum_{j=0}^{G-1} \frac{1}{2}\left(b_{i,j}^{e_1} + b_{i,j}^{e_2}\right)\right] t^2 \right. \\ &\left. + \sum_{n=1}^{N-1} \left[c_{i,0} - \sum_{j=0}^{G-1} \frac{1}{2}\left(c_{i,j}^{e_1} + c_{i,j}^{e_2}\right)\right] t + \sum_{n=1}^{N-1} \left[d_{i,0} - \sum_{j=0}^{G-1} \frac{1}{2}\left(d_{i,j}^{e_1} + d_{i,j}^{e_2}\right)\right] \right\} \mathbb{1}\left(t \in [t_{i-1}, t_i]\right) \end{aligned} \quad (11)$$

where G are the number of sifting steps to stop the sifting procedure and hence, identify the first IMF. Here $G = 2$.
Define now c_D as the $D-th$ extracted IMF. The number of sifting steps required to extract this IMF is the sum of all the previous ones and the ones necessary to extract it. We define it as Q. Therefore, the $D - th$ IMF can be expressed as:

$$\begin{aligned} c_D = c_0 - \sum_{i=0}^{Q-1} m_{D,i} = &\left\{ \sum_{i=1}^{N-1} \left[a_{i,0} - \sum_{j=0}^{Q-1} \frac{1}{2}\left(a_{i,j}^{e_1} + a_{i,j}^{e_2}\right)\right] t^3 + \sum_{i=1}^{N-1} \left[b_{i,0} - \sum_{j=0}^{Q-1} \frac{1}{2}\left(b_{i,j}^{e_1} + b_{i,j}^{e_2}\right)\right] t^2 \right. \\ &\left. \sum_{i=1}^{N-1} \left[c_{i,0} - \sum_{j=0}^{Q-1} \frac{1}{2}\left(c_{i,j}^{e_1} + c_{i,j}^{e_2}\right)\right] t + \sum_{i=1}^{N-1} \left[d_{i,0} - \sum_{j=0}^{Q-1} \frac{1}{2}\left(d_{i,j}^{e_1} + d_{i,j}^{e_2}\right)\right] \right\} \mathbb{1}\left(t \in [t_{i-1}, t_i]\right) \end{aligned} \quad (12)$$

By considering the following equalities:

- $a_i^Q = a_{i,0} - \sum_{j=0}^{Q-1} \frac{1}{2}\left(a_{i,j}^{e_1} + a_{i,j}^{e_2}\right)$
- $c_i^Q = c_{i,0} - \sum_{j=0}^{Q-1} \frac{1}{2}\left(c_{i,j}^{e_1} + c_{i,j}^{e_2}\right)$
- $b_i^Q = b_{i,0} - \sum_{j=0}^{Q-1} \frac{1}{2}\left(b_{i,j}^{e_1} + b_{i,j}^{e_2}\right)$
- $d_i^Q = d_{i,0} - \sum_{j=0}^{Q-1} \frac{1}{2}\left(d_{i,j}^{e_1} + d_{i,j}^{e_2}\right)$

The equation 12 can be written as:

$$c_D = c_0 - \sum_{i=0}^{Q-1} m_{D,i} = \sum_{i=1}^{n-1} \left(a_i^Q t^3 + b_i^Q t^2 + c_i^Q t + d_i^Q \right) \mathbb{1} \left(t \in [t_{i-1}, t_i] \right) \tag{13}$$

Appendix B

The algorithm for each spline is provided in this appendix.

- **B-Spline**

  The notation slightly changes within due to consistency with respect to the set of knots defined as $t := (t_i)$.

Algorithm 3: B-spline

Input: Discrete points $(t_i, x_i)_{i=1:n}$ on the interval $[t_0, t_n]$
Output: Cubic B-Spline

Define $k = 3$ to obtain a cubic B-spline.
Set $B_{i,1,t} = \begin{cases} 1 & \text{if} \quad t_i \leq t \leq t_{i+1} \\ 0 & \text{otherwise} \end{cases}$

foreach $i = 0, \dots n-k-1$ **do**
  foreach $j = i, \dots, i+k-1$ **do**
    Compute $\alpha_{t,k} B_{t,1}$

Compute

$$S(t) = \sum_{i=0}^{(n-k-1)} B_{i,k,t} = \sum_{i=0}^{(n-k-1)} \sum_{j=i}^{(i+k-1)} \alpha_{t,k} B_{t,1}$$

- **Natural and clamped cubic spline**

  The algorithms for a natural cubic spline and a clamped cubic spline are given by:

Algorithm 4: Natural cubic spline

Input: Discrete points $(t_i, x_i)_{i=1:n}$ on the interval $[t_0, t_n]$
Output: $(a_i, b_i, c_i, d_i)_{i=0:n-1}$ spline coefficients
foreach $i = 0 \dots n-1$ **do**
  $h_i = t_{i+1} - t_i$; $l_i = \frac{1}{h_i}(x_{i+1} - x_i)$
  $v_i = 2(h_{i-1} + h_i)$; $u_i = 6(l_i - l_{i-1})$; $z_i = S''(t_i)$
  to get: $u_i = h_{i-1} z_{-1} + v_i z_i + h_i z_{i+1}$
Set $z_0 = z_n = 0$ - natural cubic spline boundary condition.
Compute the z_i coefficients given by the system above described.
foreach $i = n-1, n-2, \dots 0$ **do**
  $a_i = \frac{z_{i+1} - z_i}{6h_i}$;
  $b_i = \frac{z_i}{2}$;
  $c_i = \frac{x_{i+1} - x_i}{h_i} - h_i \frac{2z - i + z_{i+1}}{6}$;
  $d_i = x_i$

Algorithm 5: Clamped cubic spline

Input: Discrete points $(t_i, x_i)_{i=1:n}$ on the interval $[t_0, t_n]$
Output: $(a_i, b_i, c_i, d_i)_{i=0:n-1}$ spline coefficients
foreach $i = 0 \dots n-1$ **do**
  $h_i = t_{i+1} - t_i$; $l_i = \frac{1}{h_i}(x_{i+1} - x_i)$
  $v_i = 2(h_{i-1} + h_i)$; $u_i = 6(l_i - l_{i-1})$; $z_i = S''(x_i)$
  to get: $u_i = h_{i-1}z_{-1} + v_i z_i + h_i z_{i+1}$

Set $z_0 = \frac{3l_0}{h_0} - \frac{3x_0^{'}}{h_0} - \frac{z_1}{2}$ and $z_n = \frac{3x_n^{'}}{h_{n-1}} - \frac{3b_{n-1}}{h_{n-1}} - \frac{z_{n-1}}{2}$ - clamped cubic spline boundary condition.
Compute the z_i coefficients through a tridiagonal system.
foreach $i = n-1, n-2, \dots 0$ **do**
  $a_i = \frac{z_{i+1} - z_i}{6h_i}$;
  $b_i = \frac{z_i}{2}$;
  $c_i = \frac{x_{i+1} - x_i}{h_i} - h_i \frac{2z - i + z_{i+1}}{6}$;
  $d_i = x_i$

- **Akima Splines**

Algorithm 6: Akima spline

Input: Discrete points $(t_i, x_i)_{i=1:n}$ on the interval $[t_0, t_n]$
Output: $(a_i, b_i, c_i, d_i)_{i=0:n-1}$ spline coefficients
Set $m_0 = 2m_1 - m_2$, $m_1 = 2m_2 - m_3$, $m_{N-1} = 2m_{N-2} - m_{N-2}$ and $m_N = 2m_{N-1} - m_{N-2}$.
foreach $i = 2 \dots n-2$ **do**
  Compute the quantities: $m_{i-2} = \frac{x_{i-1} - x_{i-2}}{t_{i-1} - t_{i-2}}$
  $S'(t_i) = \frac{|m_{i+2} - m_{i+1}|(m_{i-1}) + |m_{i-1} - m_{i-2}|(m_{i+1})}{|m_{i+2} - m_{i+1}| + |m_{i-1} - m_{i-2}|}$
foreach $i = 0 \dots n-1$ **do**
  $d_i = x_i$
  $c_i = S'(t_i)$
  $b_i = \frac{[3(x_{i+1} - x_i)/(t_{i+1-t_i}) - 2S'(t_i) - S'(t_{i+1})]}{t_{i+1} - t_i}$
  $a_i = \frac{[S'(t_i) + S'(t_{i+1}) - 2(x_{i+1} - x_i)/(t_{i+1} - t_i)]}{(t_{i+1} - t_i)^2}$

- **The segment power function**

The algorithm used to implement this spline is the following:

Algorithm 7: Segment power function algorithm

Input: Discrete points $(t_i, x_i)_{i=1:n}$ on the interval $[t_0, t_n]$
Output: $S(t)$ spline
foreach $i = 0 \dots n-1$ **do**

- interpolate P_{i-1}, P_i, P_{i+1} according to 3.44 as follows:

$$S_i(t) = \begin{cases} \left(\frac{t-t_i}{t_{i-1}-t_i}\right)^{\beta} \left[\frac{(t_{i+1}-t_i)x_{i-1}-(t_i-t_{i-1})x_{i+1}}{t_{i+1}-t_{i-1}}\right] + \frac{x_{i+1}-x_{i-1}}{t_{i+1}-t_{i-1}}(t-t_i) + x_i, & t \le t_i, \\ \left(\frac{t-t_i}{t_{i+1}-t_i}\right)^{\beta} \left[\frac{(t_{i+1}-t_i)x_{i-1}-(t_i-t_{i-1})x_{i+1}}{t_{i+1}-t_{i-1}}\right] + \frac{x_{i+1}-x_{i-1}}{t_{i+1}-t_{i-1}}(t-t_i) + x_i, & t \ge t_i, \end{cases}$$

  with $\beta = 2.5$ and $S_i(t)$ single valued smooth curve.

- interpolate P_i, P_{i+1}, P_{i+2} according to the above representation to obtain $S_{i+1}(t)$

- Splice the two curves $S_i(t)$ and $S_{i+1}(t)$ as:

$$S(t) = \frac{t_{i+1}-t}{t_{i+1}-t_i} S_i(t) + \frac{t-t_i}{t_{i+1}-t_i} S_{i+1}(t)$$

After having spliced every $S_i(t)$, the final result is $S(t)$ over $[t_0, t_n]$

- **Binomial Operator**

The algorithm of this alternative procedure to decompose a signal is presented below:

Algorithm 8: EMD sifting procedure through B-spline

Input: Discrete points $(t_i, x(t_i))_{i=1:n}$ on the interval $[0, T]$
Output: IMF's basis functions
repeat
  repeat
    - Identify all the local extrema of $x(t)$
    - Apply the operator $V_{t^h,k}$ to the signal x
    - Compute $h = s - V_{t^h,k}$

  until *an IMF $c(t)$ is obtained*;
  Update the initial data by subtracting the obtained IMF,
  $x(t) \leftarrow x(t) - c(t)$.
until *Having obtained a tendency $r(t)$ (a curve with at most one extremum) from the updated data*;

Appendix C

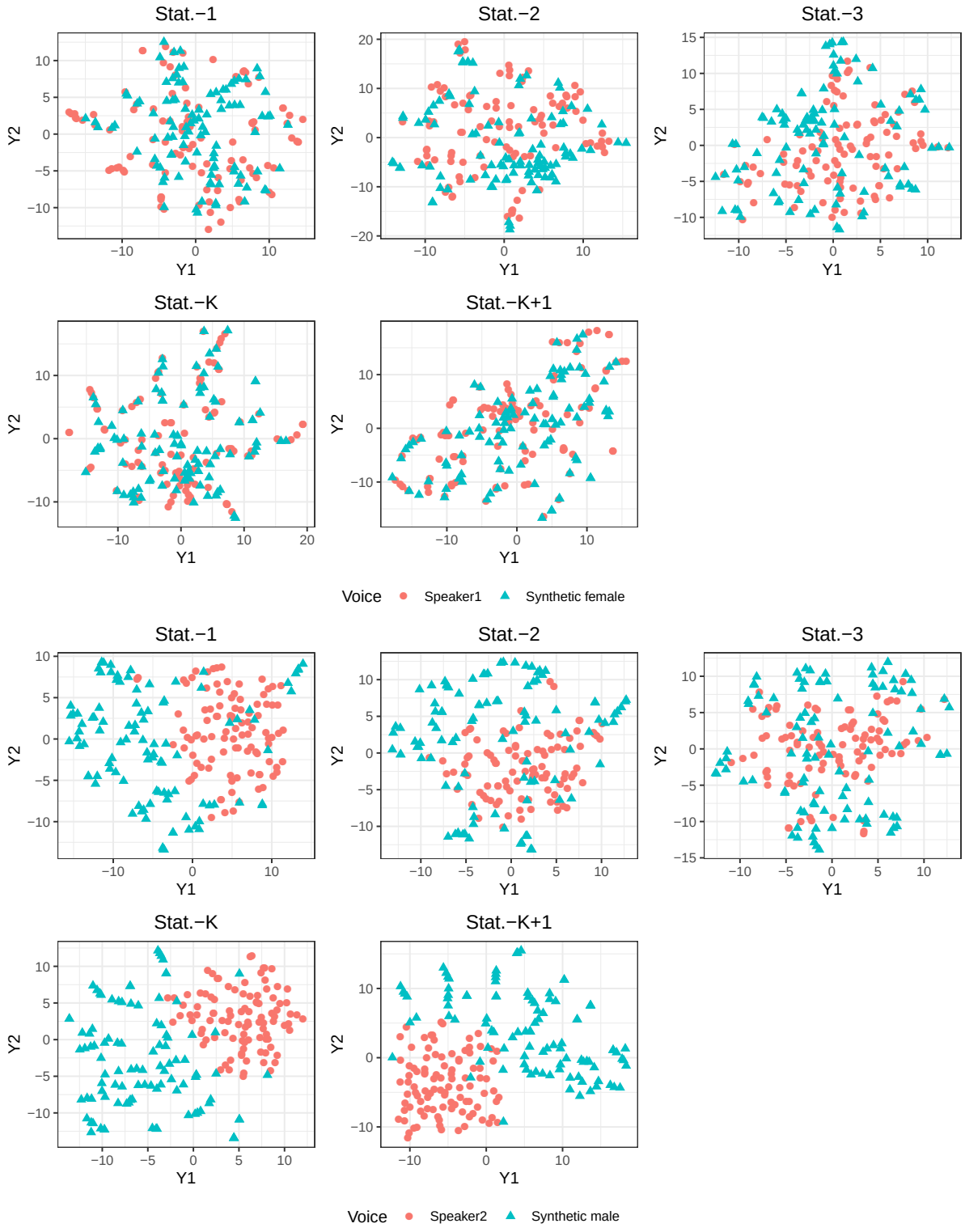


Figure 1: Results of t-SNE for the statistics of Speaker 1 (top panels) and Speaker 2 (bottom panels) versus the two different synthetic voices respectively. In each case, a PCA step was applied in each case to reduce the initial data dimensionality (from 70 to 50). The axes represent the two dimensions identified by the t-SNE algorithm denoted as $Y1$ and $Y2$.

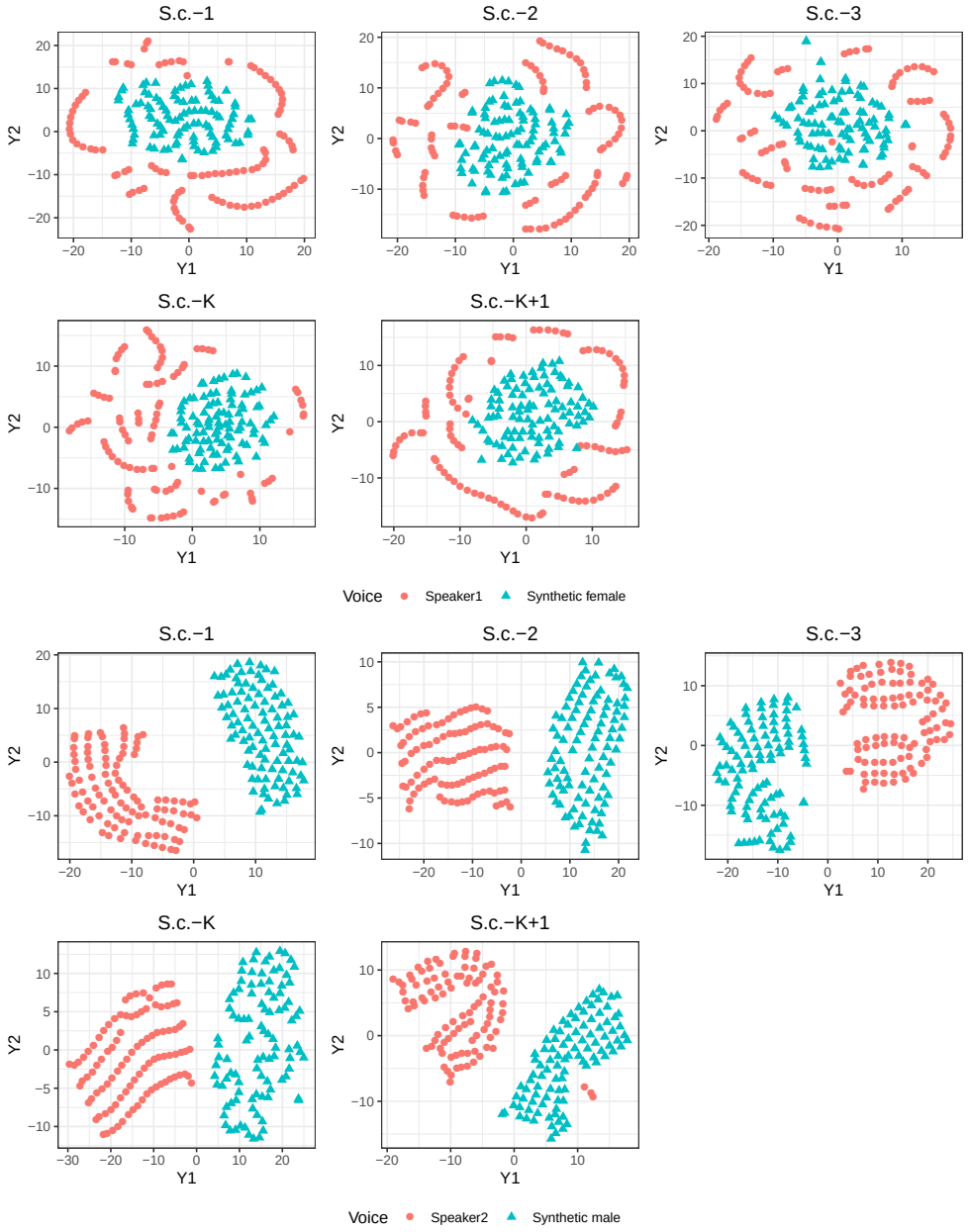


Figure 2: Results of t-SNE for the spline coefficients of Speaker 1 (top panels) and Speaker 2 (bottom panels) versus the two different synthetic voices respectively. For each speakers, 5 sub-plots are given related to each IMF taken into account. A PCA step was applied to reduce the initial data dimensionality (from 180000 to 200). The axes represent the two dimensions identified by the t-SNE algorithm denoted as $Y1$ and $Y2$.

Appendix D

In this document, we present further spectrograms for the same voice samples of Figure 5 of the main paper with a wider frequency interval and the spectrograms of the corresponding IMFs in Figure 4. Since we recorded each signal at a

sampling frequency of 44.1 kHz, we check a higher range of frequencies. What it is possible to observe is that for a female voice, formants seem to be spread out between 0 to 20 kHz; for a male voice instead, the majority of the formants are all concentrated at the very low frequencies. In the third panel, related to the Synthetic Voice, energy seems to be cut at a certain threshold; by being generated as the sum of past coefficients, a natural threshold is always given for the synthetic voice. To provide further evidence of our aim in taking MFCCs of the IMFs, we also compute spectrograms of the extracted IMFs of the signals shown in Figure 5. Figure 4 presents the results in three panels related to Speaker 1, Speaker 2 and the Synthetic voice. Same frequency range and measure of time (milliseconds) are considered. In each panel, there are five sub-figures showing spectrograms of the IMFs. It can be observed that no formants are detected by the last last IMF $\gamma_{k-1}(t')$ and the residual $\gamma_k(t')$. By looking at Speaker 1, most of the formant frequencies are identified by the first three IMFs. Higher formants seem to be recognised by both $\gamma_1(t')$ and $\gamma_2(t')$, as they contain the highest frequency content. While $\gamma_3(t')$ seems to detect the first formant along with the glottal source or fundamental frequency F_0. Such findings can be better visible in the spectrograms for Speaker 2; here, most of the formants are in $\gamma_1(t')$. Differently, $\gamma_2(t')$ seems to detect only the first formant, while $\gamma_3(t')$ seems better to catch the fundamental frequency. For the spectrograms of the Synthetic voice, $\gamma_1(t')$ appears to detect much less frequency formants information; indeed, most of it seems to appear later in $\gamma_2(t')$ and $\gamma_3(t')$. This differences could be of particular relevance when it comes to the computation of the MFCCs of the IMFs and their discriminatory power.

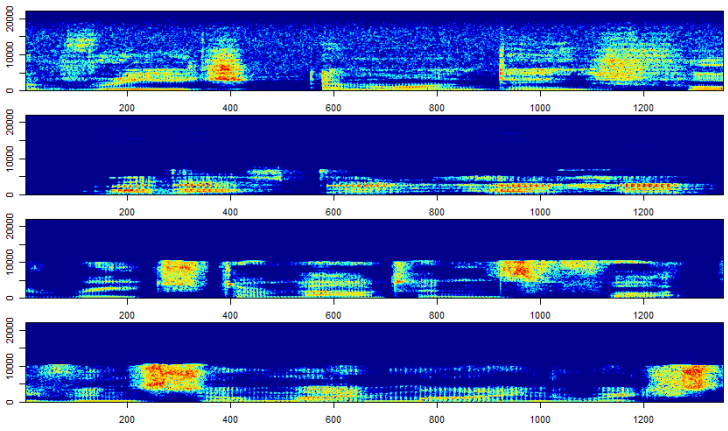


Figure 3: From the top to the bottom: spectrograms of one the sentences for Speaker 1, Speaker 2 and Synthetic voice respectively . The x-axis represents the time in milliseconds while the y-axis is the frequency in Hz (range from 0 to 25000Hz).

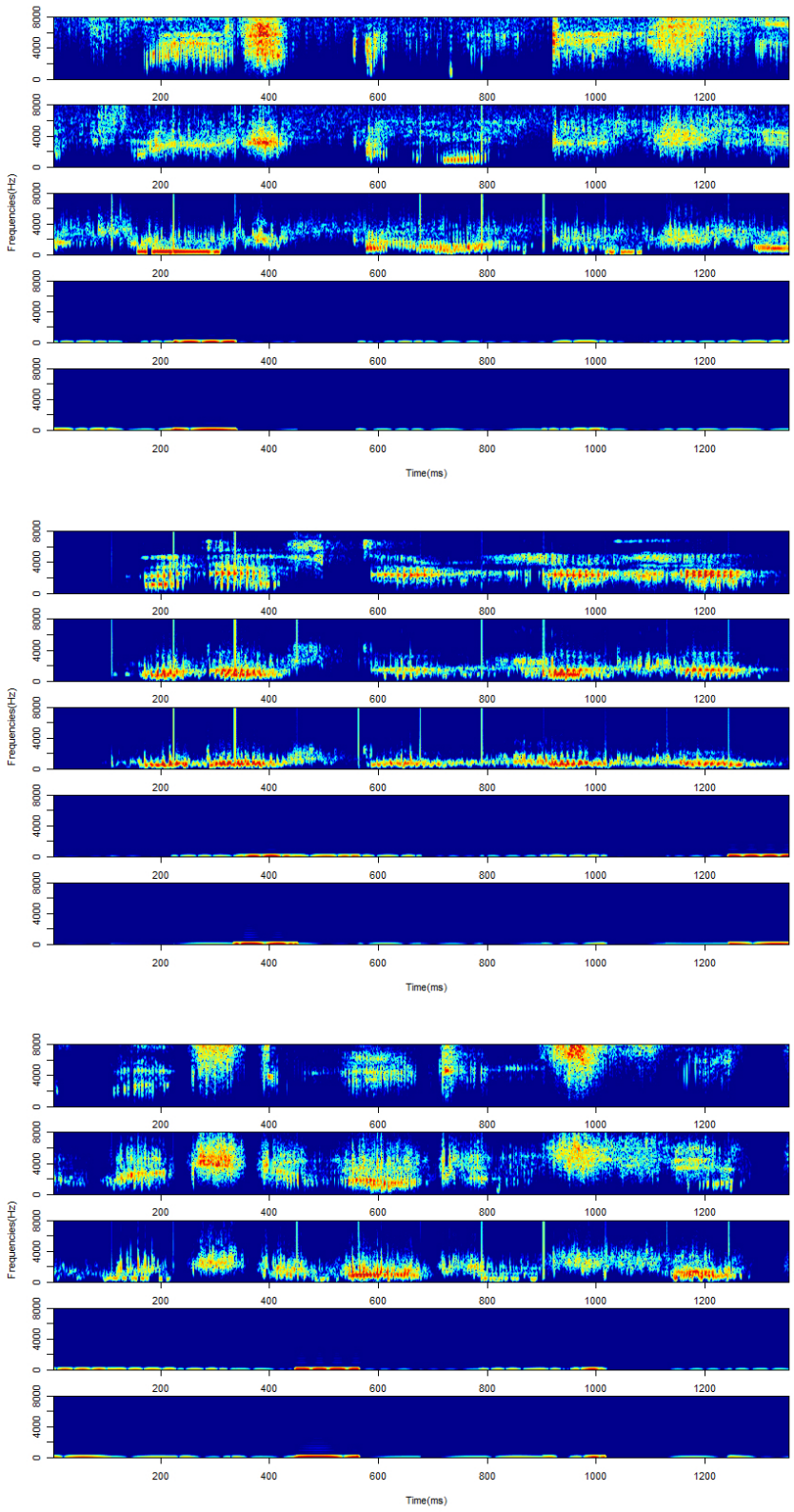


Figure 4: Spectrograms of the IMFs extracted by signals represented in 8.4. They refer to Speaker 1, Speaker 2 and the synthetic voice. There are five sub-figures for each panel showing in order $\gamma_1(t')$, $\gamma_2(t')$, $\gamma_3(t')$, $\gamma_k(t')$ and $\gamma_{k+1}(t')$.

Appendix E

In this section, we describe the results obtained in the in-sample analysis given in appendix 10.2. Given the low performances of IMFs and instantaneous frequencies for both speakers across all the other kernels, their excellent results for the Laplace and the Bessel functions are ignored. This is likely an overfit due to different training of the hyperparameters above remarked. By focusing on the statistics, high results of the ones extracted on instantaneous frequency and spline coefficients are provided within both Speakers. Performances seem to decrease with IMF index, which supports evidence that most of the discriminatory power, even for this specific feature should lie in higher frequency components. By focusing on the statistics of the IMFs instead, they provide good separation for Speaker 1 and Speaker 2. While for the former only the statistics of $\gamma_3(t')$ are well-performing, when it comes to the male voice instead, this feature gives good result for the first three IMFs in the majority of the kernels. Regarding spline coefficients, they appear to overfit in most of the investigated cases. In general, the scores of Speaker 2 appear to be overall higher than the ones of Speaker 1. The next step is the study of the EMD-MFCC features. Within 10.2, table 3 shows results related to the SVMs of both speakers. By looking at Speaker 1, the MFCCs of the first IMF provide, overall, excellent separation and, in some cases, overfit. By moving across the IMFs, performances decrease with the IMF index, as observed in figure 8.8. Focusing on the index of the coefficients within the first two IMFs, all of them provide high accuracies. MFCCs of $\gamma_3(t')$ are high in general with only a few exceptions. Results concerning $\gamma_k(t')$ and $\gamma_{k+1}(t')$ are instead constant around a score of 0.500. SVMs for Speaker 2 against the synthetic voice show similar performances, with $\gamma_1(t')$ giving fewer coefficients with perfect discrimination. In general, all the coefficients of the first three IMFs perform well, whereas, the last IMF and the residual perform poorly for the majority of the coefficients.

Kernel	IMF Index	Statistics IF					Statistics Spline Coefficients					Statistics IMFs				
		Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
Radial Basis	1	0.990	0.990	0.980	1.000	0.980	0.980	0.980	0.961	1.000	0.960	0.670	0.673	0.666	0.680	0.660
	2	0.970	0.970	0.943	1.000	0.940	0.990	0.990	0.980	1.000	0.980	0.670	0.685	0.654	0.720	0.620
	3	0.970	0.969	0.979	0.960	0.980	0.970	0.970	0.943	1.000	0.940	0.750	0.736	0.777	0.700	0.800
	K-1	0.610	0.561	0.641	0.500	0.720	0.990	0.990	0.980	1.000	0.980	0.540	0.616	0.528	0.740	0.340
	K	0.580	0.588	0.576	0.600	0.560	0.970	0.970	0.943	1.000	0.940	0.570	0.605	0.559	0.660	0.480
Laplace	1	1.000	1.000	1.000	1.000	1.000	0.990	0.990	0.980	1.000	0.980	0.650	0.623	0.674	0.580	0.720
	2	0.970	0.970	0.960	0.980	0.960	0.990	0.990	0.980	1.000	0.98	0.570	0.565	0.571	0.560	0.580
	3	0.960	0.959	0.979	0.940	0.980	0.980	0.980	0.961	1.000	0.960	0.710	0.723	0.690	0.760	0.660
	K-1	0.610	0.589	0.622	0.560	0.660	0.980	0.979	1.000	0.960	1.000	0.500	0.218	0.500	0.140	0.860
	K	0.550	0.526	0.555	0.500	0.600	0.970	0.970	0.943	1.000	0.940	0.500	0.264	0.500	0.180	0.820
Polynomial	1	0.990	0.990	0.980	1.000	0.980	1.000	1.000	1.000	1.000	1.000	0.620	0.406	0.928	0.260	0.980
	2	0.970	0.970	0.943	1.000	0.940	0.980	0.980	0.9800	0.980	0.980	0.670	0.702	0.639	0.780	0.560
	3	0.970	0.969	0.979	0.960	0.980	1.000	1.000	1.000	1.000	1.000	0.530	0.675	0.515	0.980	0.080
	K-1	0.610	0.561	0.641	0.500	0.720	0.990	0.990	0.980	1.000	0.980	0.480	0.490	0.480	0.500	0.460
	K	0.580	0.588	0.576	0.600	0.560	0.980	0.980	0.961	1.000	0.960	0.440	0.416	0.434	0.400	0.480
Sigmoid	1	0.990	0.989	1.000	0.980	1.000	0.980	0.980	0.961	1.000	0.960	0.610	0.530	0.666	0.440	0.780
	2	0.980	0.980	0.980	0.980	0.980	0.990	0.990	0.980	1.000	0.980	0.595	0.735	0.500	0.820	
	3	0.940	0.938	0.958	0.920	0.960	0.990	0.990	0.980	1.000	0.980	0.757	0.735	0.780	0.720	1.000
	K-1	0.610	0.613	0.607	0.620	0.600	0.980	0.980	0.961	1.000	0.960	0.530	0.525	0.530	0.520	0.540
	K	0.530	0.543	0.528	0.560	0.500	0.980	0.980	0.961	1.000	0.960	0.440	0.440	0.440	0.440	0.440
Bessel	1	0.990	0.989	1.000	0.980	1.000	0.990	0.990	0.980	1.000	0.980	0.511	0.611	0.440	0.720	0.975
	2	0.980	0.980	0.961	1.000	0.960	1.000	1.000	1.000	1.000	1.000	0.560	0.333	0.687	0.220	0.900
	3	0.950	0.950	0.941	0.960	0.940	0.980	0.979	1.000	0.960	1.000	0.810	0.800	0.844	0.760	0.860
	K-1	0.550	0.457	0.575	0.380	0.720	0.940	0.937	0.978	0.900	0.980	0.500	0.479	0.500	0.460	0.540
	K	0.520	0.529	0.519	0.540	0.500	0.990	0.990	0.980	1.000	0.980	0.630	0.626	0.632	0.620	0.640
Vanilla	1	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.630	0.633	0.627	0.640	0.620
	2	0.980	0.980	0.961	1.000	0.960	0.970	0.970	0.943	1.000	0.940	0.710	0.701	0.723	0.68	0.740
	3	0.940	0.938	0.958	0.920	0.960	1.000	1.000	1.000	1.000	1.000	0.660	0.711	0.617	0.840	0.480
	K-1	0.550	0.536	0.553	0.520	0.580	0.970	0.969	1.000	0.940	1.000	0.460	0.460	0.460	0.460	0.460
	K	0.610	0.597	0.617	0.580	0.640	0.980	0.980	0.961	1.000	0.960	0.560	0.551	0.562	0.540	0.580

Table 1: In-sample results of SVMs of Synthetic female voice versus Speaker 1

Kernel	IMF Index	Statistics IF					Statistics Spline Coefficients					Statistics IMFs				
		Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
Radial Basis	1	0.980	0.980	0.961	1.000	0.960	0.980	0.980	0.961	1.000	0.960	0.900	0.900	0.900	0.900	0.900
	2	0.840	0.829	0.886	0.780	0.900	0.990	0.989	1.000	0.980	1.000	0.800	0.814	0.758	0.880	0.720
	3	0.800	0.800	0.800	0.800	0.800	0.980	0.979	1.000	0.960	1.000	0.840	0.840	0.840	0.840	0.840
	K-1	0.850	0.854	0.830	0.880	0.820	0.980	0.979	1.000	0.960	1.000	0.860	0.875	0.790	0.980	0.740
	K	0.710	0.681	0.756	0.620	0.800	0.990	0.989	1.000	0.980	1.000	0.880	0.886	0.839	0.940	0.820
Laplace	1	0.990	0.990	0.980	1.000	0.980	0.990	0.989	1.000	0.980	1.00	0.920	0.921	0.903	0.940	0.900
	2	0.870	0.863	0.911	0.820	0.920	0.970	0.970	0.943	1.000	0.940	0.810	0.825	0.762	0.900	0.720
	3	0.770	0.792	0.721	0.880	0.660	0.980	0.979	1.000	0.960	1.000	0.920	0.923	0.888	0.960	0.880
	K-1	0.860	0.872	0.800	0.960	0.760	0.990	0.990	0.980	1.000	0.980	0.830	0.841	0.789	0.900	0.760
	K	0.660	0.738	0.600	0.960	0.360	0.980	0.980	0.961	1.000	0.960	0.910	0.914	0.872	0.960	0.860
Polynomial	1	0.960	0.961	0.925	1.000	0.920	1.000	1.000	1.000	1.000	1.000	0.750	0.796	0.671	0.980	0.520
	2	0.690	0.755	0.623	0.960	0.420	1.000	1.000	1.000	1.000	1.000	0.670	0.751	0.602	1.000	0.340
	3	0.59	0.709	0.549	1.000	0.180	1.000	1.000	1.000	1.000	1.000	0.610	0.719	0.561	1.000	0.220
	K-1	0.900	0.909	0.833	1.000	0.800	0.990	0.990	0.980	1.000	0.980	0.840	0.862	0.757	1.000	0.680
	K	0.600	0.714	0.555	1.000	0.200	1.000	1.000	1.000	1.000	1.000	0.720	0.781	0.641	1.000	0.440
Sigmoid	1	0.990	0.990	0.980	1.000	0.980	1.000	1.000	1.000	1.000	1.000	0.840	0.862	0.757	1.000	0.680
	2	0.830	0.841	0.789	0.900	0.760	0.990	0.989	1.000	0.980	1.000	0.740	0.786	0.666	0.960	0.520
	3	0.760	0.785	0.709	0.880	0.640	1.000	1.000	1.000	1.000	1.000	0.800	0.814	0.758	0.880	0.720
	K-1	0.850	0.859	0.807	0.920	0.780	1.000	1.000	1.000	1.000	1.000	0.830	0.824	0.851	0.800	0.860
	K	0.760	0.769	0.740	0.800	0.720	1.000	1.000	1.000	1.000	1.000	0.780	0.819	0.694	1.000	0.560
Bessel	1	1.000	1.000	1.000	1.000	1.000	0.510	0.671	0.505	1.000	0.020	0.760	0.806	0.675	1.000	0.520
	2	0.790	0.774	0.837	0.720	0.860	0.980	0.979	1.000	0.960	1.000	0.640	0.735	0.581	1.000	0.280
	3	0.710	0.728	0.684	0.780	0.640	0.500	0.666	0.500	1.000	0.000	0.550	0.181	1.000	0.1000	1.000
	K-1	0.860	0.865	0.833	0.900	0.820	1.000	1.000	1.000	1.000	1.000	0.720	0.781	0.641	1.000	0.440
	K	0.620	0.720	0.569	0.980	0.260	0.520	0.675	0.510	1.000	0.040	0.730	0.787	0.649	1.000	0.460
Vanilla	1	0.980	0.980	0.961	1.000	0.960	0.990	0.990	0.980	1.000	0.980	0.850	0.869	0.769	1.000	0.700
	2	0.780	0.780	0.780	0.780	0.780	1.000	1.000	1.000	1.000	1.000	0.750	0.770	0.711	0.840	0.660
	3	0.700	0.693	0.708	0.680	0.720	0.980	0.979	1.000	0.960	1.000	0.800	0.803	0.788	0.820	0.780
	K-1	0.810	0.825	0.762	0.900	0.720	0.990	0.990	0.980	1.000	0.980	0.860	0.872	0.800	0.960	0.760
	K	0.730	0.732	0.725	0.740	0.720	1.000	1.000	1.000	1.000	1.000	0.910	0.909	0.918	0.900	0.920

Table 2: In-sample results of SVMs of Speaker 2 vs synthetic male voice

Speaker1 - In sample

MELFCC	IMF 1					IMF 2					IMF 3					IMF K-1					IMF K				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.890	0.886	0.914	0.860	0.920	0.830	0.821	0.866	0.780	0.880	0.810	0.800	0.844	0.760	0.860	0.510	0.363	0.518	0.280	0.740	0.560	0.576	0.555	0.600	0.520
2	0.960	0.958	1.000	0.920	1.000	0.900	0.895	0.934	0.860	0.940	0.950	0.949	0.959	0.940	0.960	0.580	0.588	0.576	0.600	0.560	0.460	0.470	0.461	0.480	0.440
3	0.980	0.979	1.000	0.960	1.000	0.880	0.875	0.913	0.840	0.920	0.900	0.901	0.884	0.920	0.880	0.420	0.408	0.416	0.400	0.440	0.470	0.522	0.475	0.580	0.360
4	1.000	1.000	1.000	1.000	1.000	0.920	0.923	0.888	0.960	0.880	0.820	0.826	0.796	0.860	0.780	0.560	0.541	0.565	0.520	0.600	0.560	0.614	0.546	0.700	0.420
5	1.000	1.000	1.000	1.000	1.000	0.660	0.666	0.653	0.680	0.640	0.630	0.584	0.666	0.520	0.740	0.550	0.563	0.547	0.580	0.520	0.470	0.430	0.465	0.400	0.540
6	1.000	1.000	1.000	1.000	1.000	0.850	0.851	0.843	0.860	0.840	0.700	0.716	0.678	0.760	0.640	0.520	0.586	0.515	0.680	0.360	0.500	0.489	0.500	0.480	0.520
7	0.900	0.897	0.916	0.880	0.920	0.980	0.980	0.980	0.980	0.980	0.490	0.474	0.489	0.460	0.520	0.610	0.635	0.596	0.680	0.540	0.510	0.524	0.509	0.540	0.480
8	1.000	1.000	1.000	1.000	1.000	0.880	0.882	0.865	0.900	0.860	1.000	1.000	1.000	1.000	1.000	0.610	0.628	0.600	0.660	0.560	0.590	0.601	0.584	0.620	0.560
9	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.81	0.800	0.844	0.760	0.860	0.530	0.515	0.531	0.500	0.560	0.530	0.543	0.528	0.560	0.500
10	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.810	0.795	0.860	0.740	0.880	0.560	0.551	0.562	0.540	0.580	0.540	0.566	0.535	0.600	0.480
11	1.000	1.000	1.000	1.000	1.000	0.990	0.989	1.000	0.980	1.000	0.600	0.565	0.619	0.520	0.680	0.560	0.500	0.578	0.440	0.680	0.560	0.560	0.560	0.560	0.560
12	1.000	1.000	1.000	1.000	1.000	0.950	0.951	0.924	0.980	0.920	0.555	0.545	0.551	0.540	0.560	0.600	0.600	0.600	0.600	0.600	0.510	0.423	0.514	0.360	0.660

Speaker2 - In sample

MELFCC	IMF 1					IMF 2					IMF 3					IMF K-1					IMF K				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.840	0.843	0.826	0.860	0.820	0.890	0.888	0.897	0.880	0.900	0.870	0.873	0.849	0.900	0.840	0.940	0.940	0.923	0.960	0.920	0.940	0.940	0.940	0.940	0.940
2	0.850	0.854	0.830	0.880	0.820	0.740	0.734	0.750	0.720	0.760	0.750	0.774	0.704	0.860	0.640	0.780	0.800	0.733	0.880	0.680	0.750	0.747	0.755	0.740	0.760
3	0.900	0.905	0.857	0.960	0.840	0.720	0.740	0.689	0.800	0.640	0.700	0.736	0.656	0.840	0.560	0.810	0.815	0.792	0.840	0.780	0.660	0.673	0.648	0.700	0.620
4	0.680	0.659	0.704	0.620	0.740	0.780	0.796	0.741	0.860	0.700	0.740	0.745	0.730	0.760	0.720	0.780	0.800	0.733	0.880	0.680	0.680	0.673	0.687	0.660	0.700
5	0.920	0.924	0.875	0.980	0.860	0.620	0.641	0.607	0.680	0.560	0.770	0.772	0.764	0.780	0.760	0.810	0.811	0.803	0.820	0.800	0.600	0.642	0.580	0.720	0.480
6	0.980	0.980	0.980	0.980	0.980	0.730	0.737	0.716	0.760	0.700	0.650	0.653	0.647	0.660	0.640	0.780	0.796	0.741	0.860	0.700	0.610	0.666	0.582	0.780	0.440
7	0.970	0.970	0.943	1.000	0.940	0.830	0.821	0.866	0.78	0.88	0.640	0.653	0.629	0.680	0.600	0.800	0.795	0.812	0.780	0.820	0.640	0.700	0.600	0.840	0.440
8	0.870	0.865	0.893	0.840	0.900	0.880	0.882	0.865	0.900	0.860	0.80	0.811	0.767	0.860	0.740	0.800	0.800	0.800	0.800	0.800	0.630	0.666	0.606	0.740	0.520
9	1.000	1.000	1.000	1.000	1.000	0.920	0.924	0.875	0.98	0.860	0.730	0.737	0.716	0.760	0.700	0.810	0.800	0.844	0.760	0.860	0.630	0.593	0.658	0.540	0.720
10	1.000	1.000	1.000	1.000	1.000	0.850	0.854	0.830	0.880	0.820	0.780	0.800	0.733	0.880	0.680	0.780	0.796	0.741	0.860	0.700	0.660	0.679	0.642	0.720	0.600
11	1.00	1.000	1.000	1.000	1.000	0.850	0.854	0.830	0.880	0.820	0.770	0.762	0.787	0.740	0.800	0.840	0.836	0.854	0.820	0.860	0.680	0.666	0.695	0.640	0.720
12	0.990	0.990	0.980	1.000	0.980	0.910	0.912	0.886	0.940	0.880	0.580	0.655	0.555	0.800	0.360	0.800	0.814	0.758	0.880	0.720	0.680	0.709	0.650	0.780	0.580

Table 3: In-sample results of SVMs for both Speakers versus same gender synthetic voices.

Kernel	IMF Index	Instantaneous Frequency					IMFs					Spline Coefficients				
		Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
Radial Basis	1	0.620	0.568	0.657	0.500	0.740	0.520	0.538	0.518	0.560	0.480	1.000	1.000	1.000	1.000	1.000
	2	0.670	0.645	0.697	0.600	0.740	0.470	0.158	0.384	0.100	0.840	0.960	0.958	1.000	0.920	1.000
	3	0.580	0.511	0.611	0.440	0.720	0.430	0.359	0.410	0.320	0.540	0.980	0.980	0.980	0.980	0.980
	K-1	0.520	0.612	0.513	0.760	0.280	0.560	0.476	0.588	0.400	0.720	0.980	0.979	1.000	0.960	1.000
	K	0.520	0.314	0.550	0.220	0.820	0.520	0.489	0.522	0.460	0.580	1.000	1.000	1.000	1.000	1.000
Laplace	1	0.935	0.930	1.000	0.870	1.000	1.000	1.000	1.000	1.000	1.000	0.995	0.995	0.990	1.000	0.990
	2	0.875	0.861	0.962	0.780	0.970	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	3	0.940	0.936	0.988	0.890	0.990	1.000	1.000	1.000	1.000	1.000	0.990	0.989	1.000	0.980	1.000
	K-1	0.965	0.963	1.000	0.930	1.000	1.000	1.000	1.000	1.000	1.000	0.995	0.995	0.990	1.000	0.990
	K	0.930	0.928	0.947	0.910	0.950	0.880	0.892	0.806	1.000	0.760	0.995	0.995	0.990	1.000	0.990
Polynomial	1	0.630	0.654	0.614	0.700	0.560	0.450	0.466	0.452	0.480	0.420	1.000	1.000	1.000	1.000	1.000
	2	0.640	0.689	0.606	0.800	0.480	0.500	NA	NA	0.000	1.000	1.000	1.000	1.000	1.000	1.000
	3	0.680	0.692	0.666	0.720	0.640	0.500	NA	NA	0.000	1.000	1.000	1.000	1.000	1.000	1.000
	K-1	0.570	0.547	0.577	0.520	0.620	0.490	0.504	0.490	0.520	0.460	1.000	1.000	1.000	1.000	1.000
	K	0.460	0.602	0.476	0.820	0.100	0.580	0.618	0.566	0.680	0.480	1.000	1.000	1.000	1.000	1.000
Sigmoid	1	0.610	0.589	0.622	0.560	0.66	0.450	0.537	0.463	0.640	0.260	1.000	1.000	1.000	1.000	1.000
	2	0.620	0.558	0.666	0.480	0.760	0.530	0.459	0.540	0.400	0.660	1.000	1.000	1.000	1.000	1.000
	3	0.710	0.632	0.862	0.500	0.920	0.480	0.518	0.482	0.560	0.400	1.000	1.000	1.000	1.000	1.000
	K-1	0.530	0.338	0.571	0.240	0.820	0.480	0.458	0.478	0.440	0.520	1.000	1.000	1.000	1.000	1.000
	K	0.450	0.552	0.465	0.680	0.220	0.570	0.612	0.557	0.680	0.460	1.000	1.000	1.000	1.000	1.000
Bessel	1	0.805	0.784	0.876	0.710	0.900	1.000	1.000	1.000	1.000	1.000	0.975	0.974	0.989	0.960	0.990
	2	0.755	0.723	0.831	0.640	0.870	0.870	0.884	0.793	1.000	0.740	0.960	0.958	1.000	0.920	1.000
	3	0.785	0.777	0.806	0.750	0.820	0.975	0.975	0.961	0.990	0.960	0.965	0.965	0.951	0.980	0.950
	K-1	0.535	0.130	1.000	0.070	1.000	0.985	0.984	1.000	0.970	1.000	0.970	0.969	0.989	0.950	0.990
	K	0.585	0.314	0.904	0.190	0.980	0.820	0.833	0.775	0.900	0.740	0.990	0.990	0.990	0.990	0.990
Vanilla	1	0.620	0.641	0.607	0.680	0.560	0.380	0.392	0.384	0.400	0.360	0.990	0.990	0.980	1.000	0.980
	2	0.590	0.568	0.600	0.540	0.640	0.520	0.428	0.529	0.360	0.680	1.000	1.000	1.000	1.000	1.000
	3	0.610	0.606	0.612	0.600	0.620	0.440	0.404	0.431	0.380	0.500	0.980	0.980	0.980	0.980	0.980
	K-1	0.53	0.543	0.528	0.560	0.500	0.530	0.505	0.533	0.480	0.580	1.000	1.000	1.000	1.000	1.000
	K	0.390	0.429	0.403	0.460	0.320	0.530	0.560	0.526	0.600	0.460	1.000	1.000	1.000	1.000	1.000

Table 4: In-sample results of SVMs of Synthetic voice vs Speaker 1.

Kernel	IMF Index	Instantaneous Frequency					IMFs					Spline Coefficients				
		Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
Radial Basis	1	0.600	0.583	0.608	0.560	0.640	0.500	0.603	0.500	0.760	0.240	0.990	0.990	0.980	1.000	0.980
	2	0.550	0.545	0.551	0.540	0.560	0.520	0.500	0.521	0.480	0.560	1.000	1.000	1.000	1.000	1.000
	3	0.560	0.592	0.551	0.640	0.480	0.470	0.629	0.483	0.900	0.040	1.000	1.000	1.000	1.000	1.000
	K-1	0.630	0.713	0.582	0.920	0.340	0.690	0.710	0.666	0.760	0.620	0.980	0.980	0.961	1.000	0.960
	K	0.540	0.656	0.523	0.880	0.200	0.730	0.703	0.780	0.640	0.820	0.980	0.980	0.961	1.000	0.960
Laplace	1	0.920	0.923	0.881	0.970	0.870	0.565	0.230	1.000	0.130	1.000	0.955	0.953	0.978	0.930	0.980
	2	0.960	0.960	0.942	0.980	0.940	0.985	0.984	1.000	0.970	1.000	0.845	0.824	0.948	0.730	0.960
	3	0.955	0.955	0.950	0.960	0.950	0.995	0.994	1.000	0.990	1.000	0.930	0.931	0.913	0.950	0.910
	K-1	0.980	0.980	0.970	0.990	0.970	1.000	1.000	1.000	1.000	1.000	0.955	0.955	0.941	0.970	0.940
	K	0.980	0.980	0.970	0.990	0.970	0.910	0.901	1.000	0.820	1.000	0.975	0.975	0.970	0.980	0.970
Polynomial	1	0.580	0.543	0.595	0.500	0.660	0.570	0.690	0.539	0.960	0.180	1.000	1.0000	1.000	1.000	1.000
	2	0.560	0.450	0.600	0.360	0.760	0.530	0.113	1.000	0.06	1.000	0.990	0.990	0.980	1.000	0.980
	3	0.620	0.577	0.650	0.520	0.720	0.510	0.039	1.000	0.020	1.000	1.000	1.000	1.000	1.000	1.000
	K-1	0.740	0.697	0.833	0.600	0.880	0.660	0.645	0.673	0.620	0.700	1.000	1.000	1.000	1.000	1.000
	K	0.670	0.611	0.742	0.520	0.820	0.760	0.777	0.724	0.840	0.680	1.000	1.000	1.000	1.000	1.000
Sigmoid	1	0.620	0.641	0.607	0.680	0.560	0.530	0.447	0.542	0.38	0.68	0.980	0.979	1.000	0.960	1.000
	2	0.530	0.515	0.531	0.500	0.560	0.530	0.534	0.529	0.540	0.520	0.99	0.989	1.000	0.980	1.000
	3	0.520	0.520	0.520	0.520	0.520	0.420	0.452	0.428	0.480	0.360	0.980	0.979	1.000	0.960	1.000
	K-1	0.760	0.773	0.732	0.820	0.700	0.600	0.545	0.631	0.480	0.720	1.000	1.000	1.000	1.000	1.000
	K	0.590	0.577	0.595	0.560	0.620	0.550	0.545	0.551	0.540	0.560	1.000	1.000	1.000	1.000	1.000
Bessel	1	0.765	0.772	0.747	0.800	0.730	0.745	0.657	1.000	0.490	1.000	0.655	0.729	0.600	0.930	0.380
	2	0.795	0.748	0.968	0.610	0.980	0.915	0.907	1.000	0.830	1.000	0.505	0.595	0.503	0.730	0.280
	3	0.795	0.748	0.968	0.610	0.980	0.580	0.391	0.710	0.270	0.890	0.765	0.801	0.693	0.950	0.580
	K-1	0.795	0.822	0.725	0.950	0.640	0.980	0.979	1.000	0.960	1.000	0.605	0.710	0.560	0.970	0.240
	K	0.795	0.822	0.725	0.950	0.640	0.530	0.113	1.000	0.060	1.000	0.745	0.791	0.668	0.970	0.520
Vanilla	1	0.590	0.577	0.595	0.560	0.620	0.540	0.465	0.555	0.400	0.680	1.000	1.000	1.000	1.000	1.000
	2	0.530	0.459	0.540	0.400	0.660	0.500	0.537	0.500	0.580	0.420	1.000	1.000	1.000	1.000	1.000
	3	0.620	0.641	0.607	0.680	0.560	0.510	0.449	0.512	0.400	0.620	1.000	1.000	1.000	1.000	1.000
	K-1	0.770	0.776	0.754	0.800	0.740	0.560	0.521	0.571	0.480	0.640	1.000	1.000	1.000	1.000	1.000
	K	0.620	0.568	0.657	0.500	0.740	0.530	0.543	0.528	0.560	0.500	1.000	1.000	1.000	1.000	1.000

Table 5: In-sample results of SVMs of Speaker 2 vs synthetic male voice.

Appendix F

Results of the out-of-sample analysis are provided in tables 15 for Speaker 1 and 2 and 5 for Speaker 2. The statistics of the IMFs confirm similar behaviour as in the in-sample analysis. Such feature performs highly well in the first three IMFs across all the kernels for the female voice. For the male ones instead, within the radial basis function and the Laplace kernel, the statistics of the first three IMFs present high scores; while, within the sigmoid kernel, the ones related to $\gamma_k(t')$ provides better performances. In the case of Bessel function only $\gamma_3(t')$ gives good results and for the linear kernels, statistics of $\gamma_1(t')$ and $\gamma_2(t')$ are the best performing. When it comes to the statistic of the spline coefficients, several aspects can be found. In the case of Speaker 1 versus female synthetic voice, low accuracies can be seen in most kernels apart from the Laplace and the Sigmoid ones. In these two cases, all the statistics present good performances. For Speaker 2 versus male synthetic voice, only $\gamma_3(t')$ shows good performances of these two kernels. Contrasting behaviours of the statistics of the I.F. are also found: while in the case of Speaker 1 all the cases show low scores of the performances, for Speaker 2 instead, this feature seem to carry a good discriminatory power with respect to $\gamma_1(t')$, $\gamma_2(t')$ and $\gamma_3(t')$ in the case of Radial Basis, Laplace and Bessel kernels.
The spline coefficients, the IMFs and the instantaneous frequencies do no provide good performances (some exceptions for Speaker 1 within the spline coefficients). Table 17 presents the out-of-sample analysis for the EMD-MFCC features. Performances of Speaker 1 versus the synthetic voice related to $\gamma_1(t')$ are low for the first half of the coefficients, while, MFCCs 7, 8, 9 and 10 provide high scores. For $\gamma_2(t')$, only two coefficients perform well: the fourth one and the eighth one. Regarding the coefficients of $\gamma_3(t')$, high performances are provided by coefficients 5, 9 and 12; whereas $\gamma_k(t')$ does not provide any coefficient with performances scores higher than 0.700. Only two coefficients of $\gamma_{k+1}(t')$ show a prediction power: the first and eighth ones. Compared to the in-sample analysis, $\gamma_1(t')$ presents most of its prediction power in coefficients of higher frequencies; moving across the IMFs reduces the performances scores. SVMs for Speaker 2 versus the male synthetic voice look different; for $\gamma_1(t')$, higher MFCCs coefficients carry a stronger prediction power. Same for $\gamma_2(t')$, even if coefficients 8 and 12 also perform well. For $\gamma_3(t')$ instead, both highest and lowest coefficients provide high accuracies scores. Regarding $\gamma_k(t')$ and $\gamma_{k+1}(t')$, while for the in-sample analysis no coefficients seem to carry a discriminatory power, table 17 shows that several coefficients in both IMFs provide high performances.

Discrepancies of accuracy found in favour of Speaker 2 across both sets of features may be addressed by the spreading of the energy of the synthetic male voice across a higher range of frequencies compared to classical male voices. Such a fact may be strictly related to the synthetic voice generator taken into account. By focusing on the statistics first, the one related to the instantaneous frequencies may further justified the above fact: what is indeed found is that statistics of I.F. for the male case, especially the one related to higher frequencies, carry

discriminatory power for some of the kernels. While in the case of a female voice, this result is not found. Therefore, the frequency domain may be a more powerful tool for male voices. Regarding the statistics of the spline coefficients, these show low performances in both speakers apart for some kernels within the female case. Further investigation with this respect is required. The IMFs statistics of $\gamma_1(t')$, $\gamma_2(t')$ and $\gamma_3(t')$ better classify different signal sources in the female case; while, for Speaker 2 versus the male voice the statistics of $\gamma_1(t')$ are providing better performances.
Failures or poor achievements of the IMFs and instantaneous frequencies are possibly due to major computational issues; whereas, the spline coefficients overfit in almost all the explored cases. One objective of future research consists in selecting the optimal number of points so that coefficients which are more efficient in this classification task can be generated.

Overall, the EMD-MFCC features show the best results. Table 17 provides that MFCCs of Speaker 1 versus the synthetic voice concerning $\gamma_1(t')$ provide better performances. In particular, MFCCs of high frequencies carry the majority of the prediction power. This may be due to more intense formants of a real voice compared to a spoofed one, resulting in less energy concentration at higher MFCCs; another reason may be a different location of the formants of an authentic female voice and a synthetic female one. Therefore, if we have a female voice and a female attack, the highest MFCCs of the first IMF represent the discriminative feature. Spectrograms in 10.2 further highlight such a fact: $\gamma_1(t')$ captures most of the formants of Speaker 1, which, compared to the synthetic voice ones, seem highly stronger. Moreover, spectrograms with higher frequency ranges show that the artificial voice is cut at a certain threshold, and the MFCCs may detect such a fact. According to Mendoza et al. (1996), higher frequencies in female voices at the level of the third formant are affected by aspiration noise, producing "breathiness"; our finding are in line to such work in the sense that this aspiration noise is probably not present within a synthesised voice and MFCCs of $\gamma_1(t')$ captures this trait.

In the case of a male voice and a male attack instead, a slightly different analysis is required. Most of the prediction power seems to be found within higher coefficients of both $\gamma_1(t')$ and $\gamma_2(t')$, which further justifies the finding of the statistics extracted on the instantaneous frequencies. Such a fact provide additional evidence that a synthetic male voice may, in practice, show similar behaviour to a synthetic female one and, therefore, coefficients of IMF2 catching other formants which are not aligned as in the female case provide further power to differentiate bonafide and spoofed male voices.

To further motivate the use of the EMD basis functions against the use of the same engineered features applied to the original signal, a baseline reference study is provided in 10.2. Statistics of the IMFs extracted on the raw data provide perfect discrimination in the case of Speaker 1 versus the synthetic female voice for all kernels; while, in the male case, low performances are achieved in almost all kernels. To further investigate the female case, a frequency alignment changing

the tone or pitch of Speaker is carried simulating the use of a specific machine in Speech Spoofing attacks which equalise voices to perform a more sophisticated attack. Table 9.4 shows the results. From perfect discrimination, some of the results dropped down. However, high scores are still obtained by only employing statistics of the raw data. Therefore, another interesting finding is provided: by removing the skewness, which results to be positive in a real female voice and negative for a synthetic female one, then the results decreased steeply to 0.5 scores of accuracy. Therefore, the skewness may be a useful feature in terms of discriminatory power between female synthetic and real voices. Further evidence of such a fact would be explored.

When it comes to the same analysis of the raw data on the MFCCs, perfect discrimination is achieved within the male case, while in the female one only the first and the last coefficient seems to perform well. By applying the same reasoning and conducting an alignment in the pitch or tone for the male voice, the performances drop-down, as shown in table 9.8. To further justifies such results, which seem to be non-consistent with the IMFs, the study of the bandpass filter is carried. Results are printed in 10.2 only for the case affecting the frequency range of 4kHz to 5kHz, hence higher formants, and only for the first three IMFs. Further investigation is required since other formants, i.e. frequency ranges have to be taken into account.

By bearing in mind the discussion provided at the beginning of this section which regards the assumption of stationarity of speech, the bandpass filter study is provided to offer a clearer insight of the speaker verification problem in the two settings of male and female attacks. By firstly focusing on the performance of the MFCCs on the original signal with a bandpass filter affecting 4kHz to 5kHz in table 9.14, the following considerations need to be made. In comparison to the ones given in table 9.6, performances increase overall. Reasons behind this are enclosed by the use of a bandpass filter; it highlights formants at those frequencies and, by doing so, the resulting new formant will be strongly more stationary. As a result, the MFCCs, which exploit the use of a stationary transform as the Discrete Fourier Transform, better detect the underlying structure of the harmonics of the speech. Therefore, higher performances can be obtained. This is also reflected within the IMFs, which in general, show better performances (in the case of the bandpass filter, results of only the first three IMFs are considered).

In the case of the male voice, by comparing results of table 9.7 and table 9.18, adding a bandpass filter at high frequencies strongly affects the performances causing them to decrease much lower. Motivations behind this that has to be further explored are the followings: (1) as previously observed through the statistics on the IF or through the spectrograms, formants of the synthetic male voice seem to lie at a much higher frequencies than real male voice Speaker 2. (2) Compared to female formants, male formants are much more stationary. Therefore the MFCCs of the original signal performs perfectly. By adding additional stationarity at a specific frequency makes them much less difficult to distinguish, and the MFCCs applied on the bandpass filtered raw data poorly fail. However,

the MFCCs of the second and the third IMFs are still able to capture differences of the signals. Further investigations are required to motivate such a finding. A male voice is more stationary compared to female ones, and therefore different strategies have to be considered.

In the case of cross-cases, identifying guidelines concerning the intensity or strength of the voices with respect genders seem to be an unresolved issue (Stolarski, 2017, Gelfer and Young, 1997). In general, however, no significant difference is revealed between men and women in voice loudness (resulting in different intensity of the formants) and, therefore, providing recommendations in our context to distinguish real and synthetic signals with the added contrast of genders remains a challenging task requiring more investigations.

Experiment 1																
Dataset 1																
Kernel	IMF Index	Statistics IF (median filtered)					Statistics Spline Coefficients					Statistics IMFs				
		Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
Radial Basis	1	0.420	0.440	0.430	0.450	0.400	0.500	0.666	0.500	1.000	0.000	**0.925**	0.918	1.000	0.850	1.000
	2	0.820	0.840	0.760	0.950	0.650	0.500	0.666	0.500	1.000	0.000	**0.975**	0.974	1.000	0.950	1.000
	3	0.650	0.630	0.670	0.600	0.850	0.500	0.666	0.500	1.000	0.000	**0.975**	0.974	1.000	0.950	1.000
	K-1	0.520	0.420	0.540	0.350	0.750	0.525	0.677	0.512	1.000	0.050	0.550	0.470	0.571	0.400	0.700
	K	0.320	0.340	0.330	0.350	0.300	0.500	0.666	0.500	1.000	0.000	0.425	0.439	0.428	0.450	0.400
Laplace	1	0.170	NA	0.000	0.000	0.400	0.850	0.869	0.769	1.000	0.700	0.850	0.823	1.000	0.700	1.000
	2	0.350	0.070	0.120	0.050	0.650	0.850	0.869	0.769	1.000	0.700	**0.975**	0.974	1.000	0.950	1.000
	3	0.320	NA	0.00	0.00	0.85	0.800	0.833	0.714	1.000	0.600	**0.975**	0.974	1.000	0.950	1.000
	K-1	0.500	0.330	0.500	0.250	0.750	0.500	NA	NA	0.000	1.000	0.500	NA	NA	0.000	1.000
	K	0.350	0.350	0.350	0.350	0.300	**1.000**	1.000	1.000	1.000	1.000	0.600	0.529	0.642	0.450	0.750
Polynomial	1	0.500	NA	NA	0.000	0.400	0.450	NA	0.000	0.000	0.900	0.600	0.333	1.000	0.200	1.000
	2	0.380	NA	0.000	0.000	0.650	0.500	NA	NA	0.000	1.000	**0.975**	0.974	1.000	0.950	1.000
	3	0.420	NA	0.000	0.000	0.850	**0.950**	0.952	0.909	**1.000**	0.900	**1.000**	1.000	1.000	1.000	1.000
	K-1	0.600	0.530	0.640	0.450	0.750	0.500	NA	NA	0.000	1.000	0.425	0.488	0.440	0.550	0.300
	K	0.420	NA	0.000	0.000	0.300	0.500	NA	NA	0.000	1.000	0.550	0.470	0.571	0.4000	0.700
Sigmoid	1	0.170	NA	0.000	0.000	0.400	**0.925**	0.926	0.904	0.950	0.900	0.875	0.857	1.000	0.750	1.000
	2	0.350	NA	0.000	0.000	0.650	**1.000**	1.000	1.000	1.000	1.000	0.850	0.823	1.000	0.700	1.000
	3	0.350	NA	0.000	0.000	0.850	**1.000**	1.000	1.000	1.000	1.000	**0.950**	0.947	1.000	0.900	1.000
	K-1	0.500	0.330	0.500	0.250	0.750	**1.000**	1.000	1.000	1.000	1.000	0.525	0.486	0.529	0.450	0.600
	K	0.470	0.460	0.470	0.450	0.300	**1.000**	1.000	1.000	1.000	1.000	0.375	0.358	0.368	0.350	0.400
Bessel	1	0.100	NA	0.000	0.000	0.400	0.500	NA	NA	0.000	1.000	0.875	0.857	1.000	0.750	1.000
	2	0.320	NA	0.000	0.000	0.650	0.500	NA	NA	0.000	1.000	0.650	0.461	1.000	0.300	1.000
	3	0.320	NA	0.000	0.000	0.850	0.500	NA	NA	0.000	1.000	0.875	0.857	1.000	0.750	1.000
	K-1	0.600	0.600	0.600	0.600	0.750	0.500	NA	NA	0.000	1.000	0.375	0.324	0.352	0.300	0.450
	K	0.450	0.520	0.460	0.600	0.300	0.500	NA	NA	0.000	1.000	0.450	0.560	0.466	0.700	0.200
Linear	1	0.170	NA	0.000	0.000	0.400	0.500	NA	NA	0.000	1.000	**0.950**	0.947	1.000	0.900	1.000
	2	0.400	0.080	0.170	0.050	0.650	**1.000**	1.000	1.000	1.000	1.000	**0.950**	0.947	1.000	0.900	1.000
	3	0.380	NA	0.000	0.000	0.850	**0.950**	0.952	0.909	1.000	0.900	**0.975**	0.974	1.000	0.950	1.000
	K-1	0.500	0.440	0.500	0.400	0.750	**1.000**	1.000	1.000	1.000	1.000	0.450	0.312	0.416	0.250	0.650
	K	0.520	0.560	0.520	0.600	0.300	**1.000**	0.947	1.000	0.900	1.000	0.475	0.432	0.470	0.400	0.550

Table 6: Out-of-sample SVMs results of statistics of EMD features of Speaker 1 versus the synthetic female voice with dataset 1.

Experiment 1																
Dataset 1																
Kernel	IMF Index	Statistics IF					Statistics Spline Coefficients					Statistics IMFs				
		Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
Radial Basis	1	**1.000**	1.000	1.000	1.000	1.000	0.500	NA	NA	0.000	1.000	0.825	0.800	0.933	0.700	0.950
	2	0.650	0.562	0.750	0.450	0.850	0.500	NA	NA	0.000	1.000	0.550	0.571	0.545	0.600	0.500
	3	0.650	0.562	0.750	0.450	0.850	0.650	0.461	1.000	0.300	1.000	0.400	0.250	0.333	0.200	0.600
	K-1	0.625	0.482	0.777	0.350	0.900	0.525	0.095	1.000	0.050	1.000	0.425	0.080	0.200	0.050	0.800
	K	0.550	0.500	0.562	0.450	0.650	0.425	0.080	0.200	0.050	0.800	0.450	0.214	0.375	0.150	0.750
Laplace	1	**1.000**	1.000	1.000	1.000	1.000	0.500	NA	NA	0.000	1.000	0.750	0.722	0.812	0.650	0.850
	2	0.750	0.705	0.857	0.600	0.900	0.825	0.787	1.000	0.650	1.000	0.550	0.590	0.541	0.650	0.450
	3	0.750	0.722	0.812	0.650	0.850	0.625	0.705	0.580	0.900	0.350	0.375	0.285	0.333	0.250	0.500
	K-1	0.600	0.500	0.666	0.400	0.800	0.725	0.784	0.645	1.000	0.450	0.400	0.076	0.166	0.050	0.750
	K	0.575	0.653	0.551	0.800	0.350	0.450	0.560	0.466	0.700	0.200	0.475	0.275	0.444	0.200	0.750
Polynomial	1	0.625	0.716	0.575	0.950	0.300	0.550	0.400	0.600	0.300	0.800	0.625	0.716	0.575	0.950	0.300
	2	0.575	0.701	0.540	1.000	0.150	0.850	0.823	1.000	0.700	1.000	0.575	0.701	0.540	1.000	0.150
	3	0.525	0.641	0.515	0.850	0.200	0.500	0.666	0.500	1.000	0.000	0.525	0.641	0.515	0.850	0.200
	K-1	0.375	0.137	0.222	0.100	0.650	0.500	0.666	0.500	1.000	0.000	0.375	0.137	0.222	0.100	0.650
	K	0.425	0.378	0.411	0.350	0.500	0.425	0.596	0.459	0.850	0.000	0.425	0.378	0.411	0.350	0.500
Sigmoid	1	0.600	0.652	0.576	0.750	0.450	0.525	0.296	0.571	0.200	0.850	0.600	0.652	0.576	0.750	0.450
	2	0.375	0.489	0.413	0.600	0.150	0.675	0.518	1.000	0.350	1.000	0.375	0.489	0.413	0.600	0.150
	3	0.400	0.368	0.388	0.350	0.450	0.825	0.787	1.000	0.650	1.000	0.400	0.368	0.388	0.350	0.450
	K-1	0.450	0.153	0.333	0.100	0.800	0.500	0.666	0.500	1.000	0.000	0.450	0.153	0.333	0.100	0.800
	K	0.475	0.400	0.466	0.350	0.600	0.450	0.620	0.473	0.900	0.000	0.475	0.400	0.466	0.350	0.600
Bessel	1	**1.000**	1.000	1.000	1.000	1.000	0.500	0.666	0.500	1.000	0.000	0.525	0.666	0.513	0.950	0.100
	2	0.725	0.685	0.800	0.60	0.85	0.650	0.461	1.000	0.300	1.000	0.475	0.631	0.486	0.900	0.050
	3	0.850	0.842	0.888	0.800	0.900	0.500	0.666	0.500	1.000	0.000	0.475	NaN	0.000	0.000	0.950
	K-1	0.600	0.466	0.700	0.350	0.850	0.500	0.666	0.500	1.000	0.000	0.500	0.615	0.500	0.800	0.200
	K	0.600	0.714	0.555	1.000	0.200	0.450	0.620	0.473	0.900	0.000	0.450	0.476	0.454	0.500	0.400
Linear	1	0.575	0.604	0.565	0.650	0.500	0.500	0.285	0.500	0.200	0.800	0.575	0.604	0.565	0.650	0.500
	2	0.625	0.693	0.586	0.850	0.400	0.850	0.823	1.000	0.700	1.000	0.625	0.693	0.586	0.850	0.400
	3	0.500	0.523	0.500	0.550	0.450	0.500	0.666	0.500	1.000	0.000	0.500	0.523	0.500	0.550	0.450
	K-1	0.400	0.076	0.166	0.050	0.750	0.500	0.666	0.500	1.000	0.000	0.400	0.076	0.166	0.050	0.750
	K	0.475	0.222	0.428	0.150	0.800	0.425	0.596	0.459	0.850	0.000	0.475	0.222	0.428	0.150	0.800

Table 7: Out-of-sample SVMs results of median filtered statistics of EMD features of Speaker 2 versus the synthetic male voice with dataset 1.

Baseline References: Statistics and MFCCs of the Original Voice Recordings

Speaker1 vs synthetic female voice					
Kernel Family	Accuracy	F1-score	Precision	Sens.	Spec.
RBF	1.000	1.000	1.000	1.000	1.000
Laplace	1.000	1.000	1.000	1.000	1.000
Polynomial	1.000	1.000	1.000	1.000	1.000
Sigmoid	1.000	1.000	1.000	1.000	1.000
Bessel	1.000	1.000	1.000	1.000	1.000
Vanilla	1.000	1.000	1.000	1.000	1.000

Table 8: Out-of-sample SVMs with statistics extracted on the original voice recordings.

Speaker2 vs synthetic male voice					
Kernel Family	Accuracy	F1-score	Precision	Sens.	Spec.
RBF	0.675	0.628	0.733	0.550	0.800
Laplace	0.625	0.594	0.647	0.550	0.700
Polynomial	0.500	0.666	0.500	1.000	0.000
Sigmoid	0.475	0.511	0.478	0.550	0.400
Bessel	0.525	0.536	0.523	0.550	0.500
Vanilla	0.800	0.809	0.772	0.850	0.750

Table 9: Out-of-sample SVMs with statistics extracted on the original voice recordings.

Speaker1 with frequency alignment vs synthetic female voice					
Kernel Family	Accuracy	F1-score	Precision	Sens.	Spec.
RBF	0.525	0.677	0.512	1	0.050
Laplace	0.900	0.909	0.833	1	0.800
Polynomial	1.000	1.000	1.000	1	1.000
Sigmoid	0.725	0.784	0.645	1	0.450
Bessel	0.975	0.975	0.952	1	0.950
Vanilla	1.000	1.000	1.000	1	1.000

Table 10: Out-of-sample SVMs with statistics extracted on the original voice recordings. For this experiment, a frequency alignment have been carried before applying the SVM.

Speaker1 with frequency alignment vs synthetic female voice					
Kernel Family	Accuracy	F1-score	Precision	Sens.	Spec.
RBF	0.500	0.666	0.500	1.000	0.000
Laplace	0.500	0.666	0.500	1.000	0.000
Polynomial	0.400	0.454	0.416	0.500	0.300
Sigmoid	0.450	0.541	0.464	0.650	0.250
Bessel	0.500	0.666	0.500	1.000	0.000
Vanilla	0.625	0.693	0.586	0.850	0.400

Table 11: Out-of-sample SVMs with statistics extracted on the original voice recordings. For this experiment, a frequency alignment have been carried before applying the SVM.

Speaker1 vs synthetic female voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	0.775	0.709	1.000	0.550	1.000
2	0.675	0.628	0.733	0.550	0.800
3	0.550	0.470	0.571	0.400	0.700
4	0.475	0.553	0.481	0.650	0.300
5	0.475	0.553	0.481	0.650	0.300
6	0.650	0.461	1.000	0.300	1.000
7	0.150	0.227	0.208	0.250	0.050
8	0.075	0.051	0.052	0.050	0.100
9	0.075	NaN	0.000	0.000	0.150
10	0.350	0.187	0.250	0.150	0.550
11	0.575	0.370	0.714	0.250	0.900
12	0.750	0.666	1.000	0.500	1.000

Table 12: Out-of-sample SVMs with MFCCs extracted on the original voice recordings.

Speaker2 vs synthetic male voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	1.000	1.000	1.000	1.000	1.000
2	1.000	1.000	1.000	1.000	1.000
3	1.000	1.000	1.000	1.000	1.000
4	1.000	1.000	1.000	1.000	1.000
5	1.000	1.000	1.000	1.000	1.000
6	1.000	1.000	1.000	1.000	1.000
7	1.000	1.000	1.000	1.000	1.000
8	1.000	1.000	1.000	1.000	1.000
9	1.000	1.000	1.000	1.000	1.000
10	1.000	1.000	1.000	1.000	1.000
11	1.000	1.000	1.000	1.000	1.000
12	1.000	1.000	1.000	1.000	1.000

Table 13: Out-of-sample SVMs with MFCCs extracted on the original voice recordings.

Speaker 2 with frequency alignment vs synthetic male voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	0.600	0.466	0.700	0.350	0.850
2	0.775	0.808	0.703	0.950	0.600
3	0.750	0.687	0.916	0.550	0.950
4	0.525	0.095	1.000	0.050	1.000
5	0.650	0.562	0.750	0.450	0.850
6	0.525	0.344	0.555	0.250	0.800
7	0.675	0.648	0.705	0.600	0.750
8	0.675	0.754	0.606	1.000	0.350
9	0.550	0.250	0.750	0.150	0.950
10	0.550	0.181	1.000	0.100	1.000
11	0.750	0.722	0.812	0.650	0.850
12	0.800	0.777	0.875	0.700	0.900

Table 14: Out-of-sample SVMs with MFCCs extracted on the original voice recordings. For this experiment, a frequency alignment have been carried before applying the SVM.

IMFs and IFs and Spline Coefficients of IMFs

Kernel	IMF Index	Instantaneous Frequency					IMFs					Spline Coefficients				
		Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
Radial Basis	1	0.600	0.578	0.611	0.550	0.650	0.375	0.242	0.307	0.200	0.550	0.50	NA	NA	0.000	1.000
	2	0.500	0.500	0.500	0.500	0.500	0.400	0.200	0.300	0.150	0.650	0.500	NA	NA	0.000	1.000
	3	0.625	0.634	0.619	0.650	0.600	0.425	0.080	0.200	0.050	0.800	0.500	NA	NA	0.000	1.000
	K-1	0.550	0.357	0.625	0.250	0.850	0.475	0.400	0.466	0.350	0.600	0.500	NA	NA	0.000	1.000
	K	0.400	0.333	0.375	0.300	0.500	0.550	0.590	0.541	0.650	0.450	0.500	NA	NA	0.000	1.000
Laplace	1	0.550	0.500	0.562	0.450	0.650	0.400	0.200	0.300	0.150	0.650	0.800	0.833	0.714	1.000	0.600
	2	0.550	0.590	0.541	0.650	0.450	0.425	0.549	0.451	0.700	0.150	0.550	0.689	0.526	1.000	0.1000
	3	0.650	0.708	0.607	0.850	0.450	0.425	0.510	0.444	0.600	0.250	0.475	NaN	0.000	0.000	0.950
	K-1	0.525	0.641	0.515	0.850	0.200	0.375	0.137	0.222	0.100	0.650	0.500	0.666	0.500	1.000	0.000
	K	0.475	0.631	0.486	0.900	0.050	0.400	NaN	0.000	0.000	0.800	0.375	0.390	0.380	0.400	0.350
Polynomial	1	0.575	0.585	0.571	0.600	0.550	0.500	0.500	0.500	0.500	0.500	0.600	0.619	0.590	0.650	0.550
	2	0.600	0.636	0.583	0.700	0.500	0.500	NA	NA	0.000	1.000	0.700	0.750	0.642	0.900	0.500
	3	0.550	0.550	0.550	0.550	0.550	0.500	NA	NA	0.000	1.000	0.650	0.720	0.600	0.900	0.400
	K-1	0.625	0.634	0.619	0.650	0.600	0.450	0.476	0.454	0.500	0.400	0.500	0.666	0.500	1.000	0.000
	K	0.275	0.256	0.263	0.250	0.300	0.475	0.222	0.428	0.150	0.800	0.500	0.655	0.500	0.950	0.050
Sigmoid	1	0.600	0.578	0.611	0.550	0.650	0.600	0.652	0.576	0.750	0.450	0.575	0.585	0.571	0.600	0.550
	2	0.600	0.555	0.625	0.500	0.700	0.625	0.651	0.608	0.700	0.550	0.750	0.782	0.692	0.900	0.600
	3	0.575	0.540	0.588	0.500	0.650	0.475	0.571	0.482	0.700	0.250	0.650	0.720	0.600	0.900	0.400
	K-1	0.450	0.352	0.428	0.300	0.600	0.450	0.450	0.450	0.450	0.450	0.500	0.666	0.500	1.000	0.000
	K	0.325	0.425	0.370	0.500	0.150	0.450	0.521	0.461	0.600	0.300	0.500	0.655	0.500	0.950	0.050
Bessel	1	0.550	0.500	0.562	0.450	0.650	0.475	0.487	0.476	0.500	0.450	0.350	0.071	0.125	0.050	0.650
	2	0.575	0.604	0.565	0.650	0.500	0.400	0.368	0.388	0.350	0.450	0.375	0.074	0.142	0.050	0.700
	3	0.625	0.693	0.586	0.850	0.400	0.400	0.428	0.409	0.450	0.350	0.550	0.250	0.750	0.150	0.950
	K-1	0.525	0.641	0.515	0.850	0.200	0.500	0.545	0.500	0.600	0.400	0.500	NA	NA	0.000	1.000
	K	0.500	0.666	0.500	1.000	0.000	0.525	0.612	0.517	0.750	0.300	0.500	NA	NA	0.000	1.000
Vanilla	1	0.550	0.550	0.550	0.550	0.550	0.500	0.500	0.500	0.500	0.500	0.625	0.680	0.592	0.800	0.450
	2	0.500	0.523	0.500	0.550	0.450	0.650	0.666	0.636	0.700	0.600	0.650	0.708	0.607	0.850	0.450
	3	0.500	0.500	0.500	0.500	0.500	0.475	0.571	0.482	0.700	0.250	0.600	0.384	0.833	0.250	0.950
	K-1	0.475	0.432	0.470	0.400	0.550	0.475	0.511	0.478	0.550	0.400	0.500	0.666	0.500	1.000	0.000
	K	0.450	0.388	0.437	0.350	0.550	0.475	0.461	0.473	0.450	0.500	0.500	0.655	0.500	0.950	0.050

Table 15: Out-of-sample SVMs of Speaker 1 versus the synthetic female voice.

Kernel	IMF Index	Instantaneous Frequency					IMFs					Spline Coefficients				
		Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
Radial Basis	1	0.300	0.222	0.250	0.200	0.400	0.675	0.711	0.640	0.800	0.550	0.500	0.666	0.500	1.000	0.000
	2	0.425	0.410	0.421	0.400	0.450	0.625	0.444	0.857	0.300	0.950	0.500	0.666	0.500	1.000	0.000
	3	0.625	0.615	0.631	0.600	0.650	0.475	0.631	0.486	0.900	0.050	0.500	0.666	0.500	1.000	0.000
	K-1	0.475	0.461	0.473	0.450	0.500	0.575	0.484	0.615	0.400	0.750	0.500	0.666	0.500	1.000	0.000
	K	0.575	0.653	0.551	0.800	0.350	0.500	0.166	0.500	0.100	0.900	0.500	0.666	0.500	1.000	0.000
Laplace	1	0.575	0.540	0.588	0.500	0.650	0.625	0.705	0.580	0.900	0.350	0.575	0.690	0.542	0.950	0.200
	2	0.625	0.651	0.608	0.700	0.550	0.575	0.679	0.545	0.900	0.250	0.475	0.461	0.473	0.450	0.500
	3	0.650	0.740	0.588	1.000	0.300	0.500	0.629	0.500	0.850	0.150	0.600	0.703	0.558	0.950	0.250
	K-1	0.525	0.627	0.516	0.800	0.250	0.500	0.642	0.500	0.900	0.100	0.525	0.677	0.512	1.000	0.050
	K	0.525	0.595	0.518	0.700	0.350	0.500	0.666	0.500	1.000	0.000	0.500	0.666	0.500	1.000	0.000
Polynomial	1	0.300	0.176	0.214	0.150	0.450	0.500	NA	NA	0.000	1.000	0.575	0.585	0.571	0.600	0.550
	2	0.450	0.352	0.428	0.300	0.600	0.500	NA	NA	0.000	1.000	0.500	0.583	0.500	0.700	0.300
	3	0.575	0.451	0.636	0.350	0.800	0.500	NA	NA	0.000	1.000	0.625	0.651	0.608	0.700	0.550
	K-1	0.525	0.344	0.555	0.250	0.800	0.475	0.222	0.428	0.150	0.800	0.450	0.607	0.472	0.850	0.050
	K	0.550	0.400	0.600	0.300	0.800	0.550	0.400	0.600	0.300	0.800	0.500	0.642	0.500	0.900	0.100
Sigmoid	1	0.300	0.125	0.166	0.100	0.500	0.700	0.700	0.700	0.700	0.700	0.550	0.571	0.545	0.600	0.500
	2	0.475	0.432	0.470	0.400	0.550	0.725	0.645	0.909	0.500	0.950	0.475	0.571	0.482	0.700	0.250
	3	0.500	0.500	0.500	0.500	0.500	0.500	0.523	0.500	0.550	0.450	0.600	0.636	0.583	0.700	0.500
	K-1	0.625	0.516	0.727	0.400	0.850	0.550	0.437	0.583	0.350	0.750	0.475	0.631	0.486	0.900	0.050
	K	0.675	0.648	0.705	0.600	0.750	0.575	0.540	0.588	0.500	0.650	0.475	0.631	0.486	0.900	0.050
Bessel	1	0.575	0.540	0.588	0.500	0.650	0.525	0.641	0.515	0.850	0.20	0.525	0.536	0.523	0.550	0.500
	2	0.625	0.651	0.608	0.700	0.550	0.525	0.654	0.514	0.900	0.150	0.400	0.454	0.416	0.500	0.300
	3	0.650	0.740	0.588	1.000	0.300	0.550	0.678	0.527	0.950	0.150	0.525	0.558	0.521	0.600	0.450
	K-1	0.500	0.600	0.500	0.750	0.250	0.500	0.642	0.500	0.900	0.100	0.550	0.625	0.535	0.750	0.350
	K	0.525	0.595	0.518	0.700	0.350	0.500	0.666	0.500	1.000	0.000	0.575	0.679	0.545	0.900	0.250
Vanilla	1	0.300	0.125	0.166	0.100	0.500	0.750	0.736	0.777	0.700	0.800	0.575	0.585	0.571	0.600	0.550
	2	0.425	0.410	0.421	0.400	0.450	0.725	0.645	0.909	0.500	0.950	0.500	0.583	0.500	0.700	0.300
	3	0.475	0.461	0.473	0.450	0.500	0.525	0.536	0.523	0.550	0.500	0.625	0.651	0.608	0.70	0.55
	K-1	0.500	0.285	0.500	0.200	0.800	0.550	0.470	0.571	0.400	0.700	0.425	0.581	0.457	0.800	0.050
	K	0.525	0.512	0.526	0.500	0.550	0.500	0.500	0.500	0.500	0.50 0	0.500	0.655	0.500	0.950	0.050

Table 16: Out-of-sample SVMs of Speaker 2 versus the synthetic male voice.

EMD-MFCCs

Speaker1 - out of sample

MFCC	IMF 1					IMF 2					IMF 3					IMF K-1					IMF K				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.150	0.260	0.230	0.300	0.000	0.425	0.596	0.459	0.850	0.000	0.500	0.666	0.500	1.000	0.000	0.300	0.461	0.375	0.600	0.000	0.750	0.666	1.000	0.500	1.000
2	0.325	NaN	0.000	0.000	0.650	0.425	NaN	0.000	0.000	0.850	0.350	0.071	0.125	0.050	0.650	0.575	0.638	0.555	0.750	0.400	0.425	0.258	0.363	0.200	0.650
3	0.225	NaN	0.000	0.000	0.450	0.550	0.653	0.531	0.850	0.250	0.350	0.071	0.125	0.050	0.650	0.550	0.590	0.541	0.650	0.450	0.375	0.468	0.407	0.550	0.200
4	0.075	NaN	0.000	0.000	0.150	0.700	0.647	0.785	0.550	0.850	0.350	0.071	0.125	0.050	0.650	0.525	0.558	0.521	0.600	0.450	0.375	0.242	0.307	0.200	0.550
5	0.050	NaN	0.000	0.000	0.100	0.150	NaN	0.000	0.000	0.300	0.775	0.808	0.703	0.950	0.600	0.575	0.540	0.588	0.500	0.650	0.325	0.228	0.266	0.200	0.450
6	0.050	0.050	0.050	0.050	0.050	0.375	0.285	0.333	0.250	0.500	0.125	NaN	0.000	0.000	0.250	0.650	0.681	0.625	0.750	0.550	0.350	0.277	0.312	0.250	0.450
7	0.775	0.816	0.689	1.000	0.550	0.200	0.058	0.071	0.050	0.350	0.400	0.400	0.400	0.400	0.400	0.575	0.622	0.560	0.700	0.450	0.350	0.277	0.312	0.250	0.450
8	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.425	0.148	0.285	0.100	0.750	0.525	0.424	0.538	0.350	0.700	0.725	0.731	0.714	0.750	0.700
9	1.000	1.000	1.000	1.000	1.000	0.550	0.181	1.000	0.100	1.000	0.725	0.775	0.655	0.950	0.500	0.650	0.666	0.636	0.700	0.600	0.300	0.263	0.277	0.250	0.350
10	0.750	0.666	1.000	0.500	1.000	0.400	NaN	0.000	0.000	0.800	0.175	NaN	0.000	0.000	0.350	0.675	0.697	0.652	0.750	0.600	0.250	0.318	0.291	0.350	0.150
11	0.600	0.333	1.000	0.200	1.000	0.350	NaN	0.000	0.000	0.700	0.325	0.181	0.230	0.150	0.500	0.550	0.571	0.545	0.600	0.500	0.325	0.307	0.315	0.300	0.350
12	0.625	0.400	1.000	0.250	1.000	0.325	0.372	0.347	0.400	0.250	0.700	0.769	0.625	1.000	0.400	0.475	0.487	0.476	0.500	0.450	0.325	0.341	0.333	0.350	0.300

Speaker2 - out of sample

MFCC	IMF 1					IMF 2					IMF 3					IMF K-1					IMF K				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.450	0.153	0.333	0.100	0.800	0.375	NaN	0.000	0.000	0.750	0.500	0.333	0.500	0.250	0.750	0.425	NaN	0.000	0.000	0.850	0.500	0.166	0.500	0.100	0.900
2	0.725	0.685	0.800	0.600	0.850	0.700	0.625	0.833	0.500	0.900	0.925	0.923	0.947	0.900	0.950	0.550	0.400	0.600	0.300	0.800	0.500	0.523	0.500	0.550	0.450
3	0.675	0.580	0.818	0.450	0.900	0.625	0.571	0.666	0.500	0.750	0.575	0.484	0.615	0.400	0.750	0.525	0.344	0.555	0.250	0.800	0.500	0.523	0.500	0.550	0.450
4	0.825	0.810	0.882	0.750	0.900	0.650	0.681	0.625	0.750	0.550	0.750	0.791	0.678	0.950	0.550	0.600	0.500	0.666	0.400	0.800	0.475	0.487	0.476	0.500	0.450
5	1.000	1.000	1.000	1.000	1.000	0.775	0.756	0.823	0.700	0.850	0.625	0.634	0.619	0.650	0.600	0.575	0.451	0.636	0.350	0.800	0.475	0.511	0.478	0.550	0.400
6	1.000	1.000	1.000	1.000	1.000	0.850	0.850	0.850	0.850	0.850	0.500	0.473	0.500	0.450	0.550	0.450	0.153	0.333	0.100	0.800	0.500	0.523	0.500	0.550	0.450
7	0.850	0.863	0.791	0.950	0.750	0.700	0.727	0.666	0.800	0.600	0.500	0.473	0.500	0.450	0.550	0.450	0.266	0.400	0.200	0.700	0.475	0.487	0.476	0.500	0.450
8	0.775	0.742	0.866	0.650	0.900	0.900	0.904	0.863	0.950	0.850	0.475	0.511	0.478	0.550	0.400	0.525	0.387	0.545	0.300	0.750	0.500	0.523	0.500	0.550	0.450
9	1.000	1.000	1.000	1.000	1.000	0.975	0.974	1.000	0.950	1.000	0.750	0.772	0.708	0.850	0.650	0.425	0.258	0.363	0.200	0.650	0.450	0.476	0.454	0.500	0.400
10	1.000	1.000	1.000	1.000	1.000	0.975	0.974	1.000	0.950	1.000	0.850	0.842	0.888	0.800	0.900	0.425	0.148	0.285	0.100	0.750	0.525	0.558	0.521	0.600	0.450
11	1.000	1.000	1.000	1.000	1.000	0.900	0.904	0.863	0.950	0.850	0.675	0.580	0.818	0.450	0.900	0.400	0.142	0.250	0.100	0.700	0.475	0.461	0.473	0.450	0.500
12	1.000	1.000	1.000	1.000	1.000	0.900	0.909	0.833	1.000	0.800	0.525	0.387	0.545	0.300	0.750	0.400	0.142	0.250	0.100	0.700	0.525	0.558	0.521	0.600	0.450

Table 17: Out-of-sample SVMs of Speaker 1 versus the synthetic female voice (top table) and Speaker 2 versus the male synthetic (bottom table) with kernel corresponding to the Radial Basis Function.

Bandpass filter on formant between 4000Hz and 5000Hz

Speaker1 vs synthetic female voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	0.850	0.823	1.000	0.700	1.000
2	0.575	0.622	0.560	0.700	0.450
3	0.400	0.076	0.166	0.050	0.750
4	0.750	0.736	0.777	0.700	0.800
5	0.650	0.611	0.687	0.550	0.750
6	0.650	0.681	0.625	0.750	0.550
7	0.425	0.148	0.285	0.100	0.750
8	0.675	0.628	0.733	0.550	0.800
9	0.675	0.697	0.652	0.750	0.600
10	0.425	0.303	0.384	0.250	0.600
11	0.650	0.611	0.687	0.550	0.750
12	0.450	0.266	0.400	0.200	0.700

Table 18: Out-of-sample results of SVMs of the MFCCs of the raw data.

Speaker1 vs synthetic female voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	0.900	0.888	1.000	0.800	1.000
2	0.375	0.358	0.368	0.350	0.400
3	0.600	0.466	0.700	0.350	0.850
4	0.550	0.500	0.562	0.450	0.650
5	0.700	0.714	0.681	0.750	0.650
6	0.700	0.700	0.700	0.700	0.700
7	0.750	0.772	0.708	0.850	0.650
8	0.625	0.594	0.647	0.550	0.700
9	0.500	0.444	0.500	0.400	0.600
10	0.775	0.790	0.739	0.850	0.700
11	0.725	0.717	0.736	0.700	0.750
12	0.675	0.682	0.666	0.700	0.650

Table 19: Out-of-sample results of SVMs of the MFCCs of IMF1

Speaker1 vs synthetic female voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	0.825	0.787	1.000	0.650	1.000
2	0.325	0.490	0.393	0.650	0.000
3	0.500	0.285	0.500	0.200	0.800
4	0.750	0.791	0.678	0.950	0.550
5	0.375	0.358	0.368	0.350	0.400
6	0.650	0.611	0.687	0.550	0.750
7	0.825	0.787	1.000	0.650	1.000
8	0.350	0.277	0.312	0.250	0.450
9	0.800	0.750	1.000	0.600	1.000
10	0.550	0.500	0.562	0.450	0.650
11	0.550	0.470	0.571	0.400	0.700
12	0.775	0.709	1.000	0.550	1.000

Table 20: Out-of-sample results of SVMs of the MFCCs of IMF2

Speaker1 vs synthetic female voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	1.000	1.000	1.000	1.000	1.000
2	0.700	0.571	1.000	0.400	1.000
3	0.475	0.086	0.333	0.050	0.900
4	0.575	0.260	1.000	0.150	1.000
5	0.600	0.384	0.833	0.250	0.950
6	0.600	0.333	1.000	0.200	1.000
7	0.500	0.090	0.500	0.050	0.950
8	0.475	0.086	0.333	0.050	0.900
9	0.525	0.095	1.000	0.050	1.000
10	0.650	0.500	0.875	0.350	0.950
11	0.475	0.086	0.333	0.050	0.900
12	0.500	0.090	0.500	0.050	0.950

Table 21: Out-of-sample results of SVMs of the MFCCs of IMF3

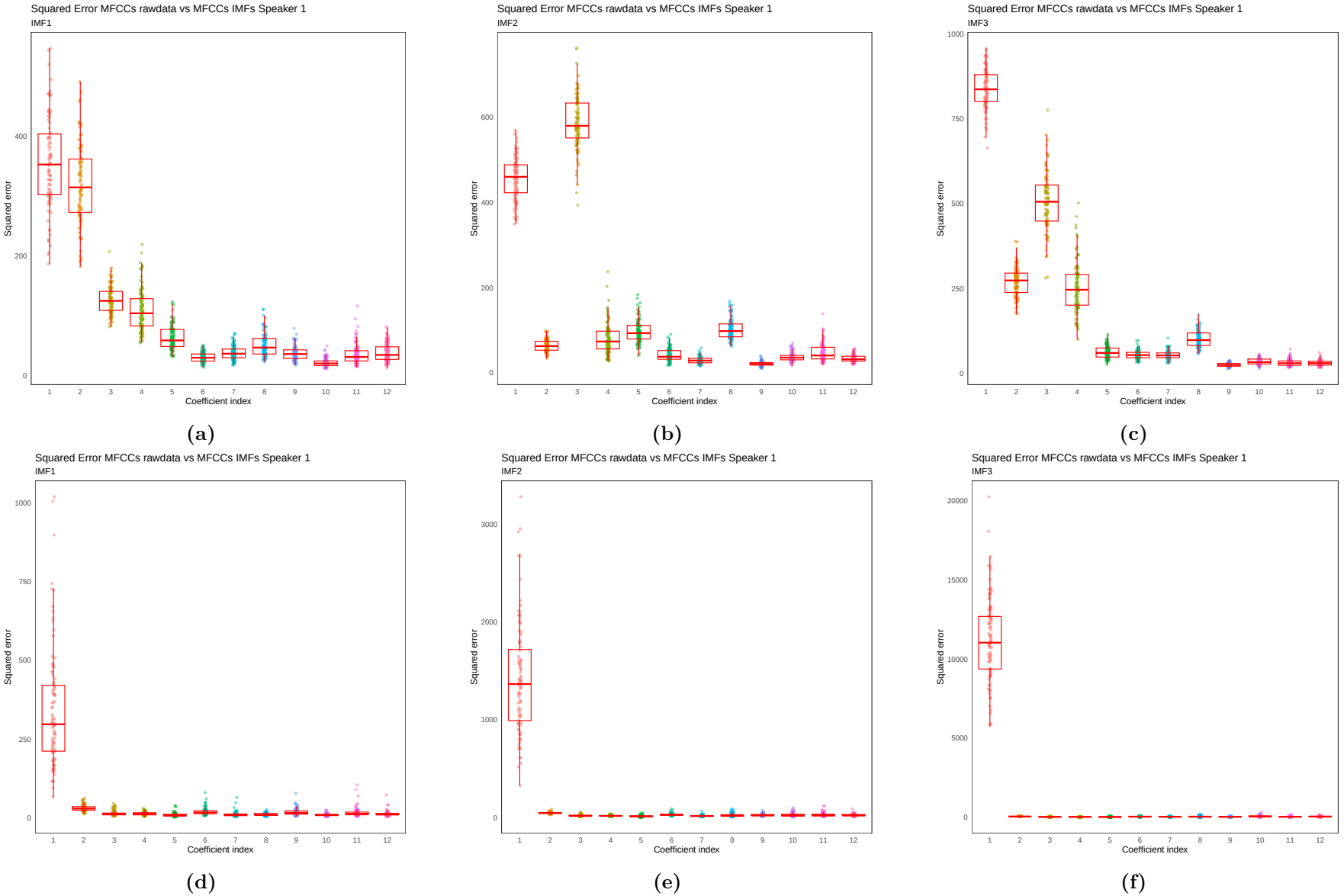


Figure 5: Speaker1 vs female voice - ideal case a), b), c). The other are the one convoluted signals with bandpass filter affecting 4,000Hz to 5,000 Hz.

Speaker2 vs synthetic male voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	0.975	0.975	0.952	1.000	0.950
2	0.525	0.536	0.523	0.550	0.500
3	0.525	0.240	0.600	0.150	0.900
4	0.575	0.540	0.588	0.500	0.650
5	0.550	0.500	0.562	0.450	0.650
6	0.550	0.437	0.583	0.350	0.750
7	0.600	0.578	0.611	0.550	0.650
8	0.625	0.545	0.692	0.450	0.800
9	0.650	0.650	0.650	0.650	0.650
10	0.700	0.625	0.833	0.500	0.900
11	0.575	0.370	0.714	0.250	0.900
12	0.650	0.588	0.714	0.500	0.800

Table 22: Out-of-sample results of SVMs of the MFCCs of Raw data

Speaker2 vs synthetic male voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	0.500	NA	NA	0.000	1.00
2	0.500	NA	NA	0.000	1.00
3	0.500	NA	NA	0.000	1.00
4	0.500	0.666	0.500	1.000	0.000
5	0.500	0.666	0.500	1.000	0.000
6	0.500	0.666	0.500	1.000	0.000
7	0.500	0.666	0.500	1.000	0.000
8	0.500	0.666	0.500	1.000	0.000
9	0.500	0.666	0.500	1.000	0.000
10	0.625	0.716	0.575	0.950	0.300
11	0.500	0.666	0.500	1.000	0.000
12	0.500	0.655	0.500	0.950	0.050

Table 23: Out-of-sample results of SVMs of the MFCCs of IMF1

Speaker2 vs synthetic male voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	0.450	NaN	0.000	0.000	0.900
2	0.400	NaN	0.000	0.000	0.800
3	0.425	NaN	0.000	0.000	0.850
4	0.500	NA	NA	0.00	1.000
5	0.600	0.333	1.000	0.20	1.000
6	1.000	1.000	1.000	1.00	1.000
7	1.000	1.000	1.000	1.000	1.000
8	0.000	NaN	0.000	0.000	0.000
9	1.000	1.000	1.000	1.00	1.000
10	0.925	0.918	1.000	0.85	1.000
11	0.900	0.909	0.833	1.00	0.800
12	1.000	1.000	1.000	1.00	1.000

Table 24: Out-of-sample results of SVMs of the MFCCs of IMF2

Speaker2 vs synthetic male voice					
Coeff. number	Accuracy	F1-score	Precision	Sens.	Spec.
1	0.500	NA	NA	0.000	1.000
2	1.000	1.000	1.000	1.000	1.000
3	0.900	0.909	0.833	1.000	0.800
4	0.950	0.952	0.909	1.000	0.900
5	1.000	1.000	1.000	1.000	1.000
6	0.950	0.952	0.909	1.000	0.900
7	0.000	NaN	0.000	0.000	0.000
8	0.900	0.909	0.833	1.000	0.800
9	0.000	NaN	0.000	0.000	0.000
10	0.825	0.787	1.000	0.650	1.000
11	0.600	0.714	0.555	1.000	0.200
12	0.500	NA	NA	0.000	1.000

Table 25: Out-of-sample results of SVMs of the MFCCs of IMF3

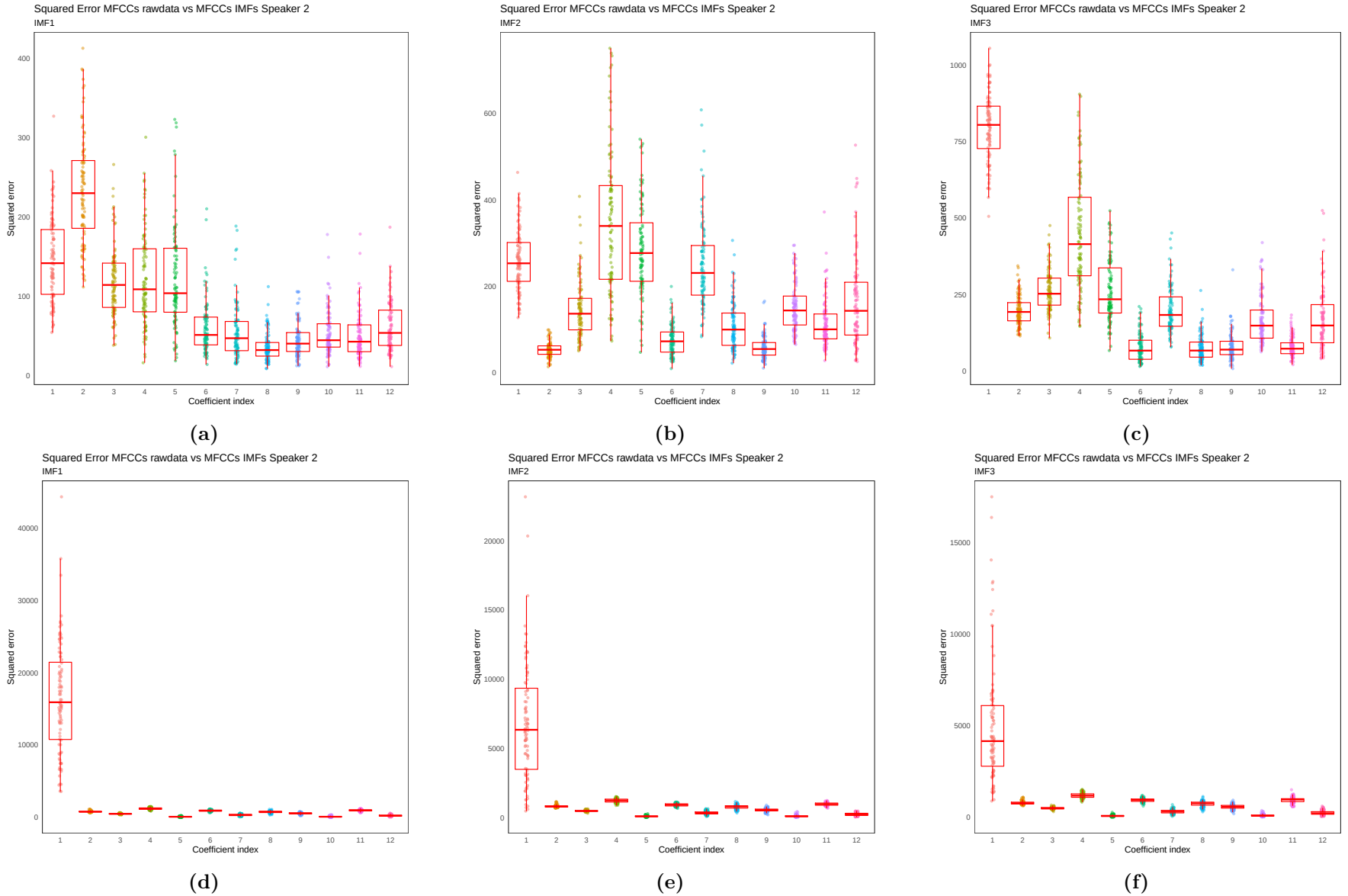


Figure 6: Speaker2 vs male voice - ideal case a), b), c). The other are the one convoluted signals with bandpass filter affecting 4,000Hz to 5,000 Hz.

Experiment 2

Dataset 1

Speaker1 vs Female synthetic voice generated with Espeak TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.450	0.620	0.473	0.900	0.000	0.500	0.666	0.5000	1.000	0.000	0.500	0.666	0.500	1.000	0.000	0.825	0.787	1.000	0.650	1.000	**0.950**	0.947	1.000	0.900	1.000
2	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.0000	1.000	1.000	**0.925**	0.930	0.869	1.000	0.850	0.550	0.571	0.545	0.600	0.500	0.750	0.750	0.750	0.750	0.750
3	**0.975**	0.975	0.952	1.000	0.950	**1.000**	1.000	1.000	1.000	1.000	**0.975**	0.975	0.952	1.000	0.950	0.450	0.421	0.444	0.400	0.500	0.800	0.800	0.800	0.800	0.800
4	0.600	0.714	0.555	1.000	0.200	**0.950**	0.950	0.950	0.950	0.950	**1.000**	1.000	1.000	1.000	1.000	0.450	0.476	0.454	0.500	0.400	0.825	0.820	0.842	0.800	0.850
5	0.525	0.677	0.512	1.000	0.050	**0.900**	0.904	0.863	0.950	0.850	0.675	0.697	0.652	0.750	0.600	0.475	0.511	0.478	0.550	0.400	0.825	0.820	0.842	0.800	0.850
6	0.500	0.642	0.500	0.900	0.100	0.850	0.857	0.818	0.900	0.800	0.775	0.808	0.703	0.950	0.600	0.375	0.358	0.368	0.350	0.400	0.800	0.800	0.800	0.800	0.800
7	**1.000**	1.000	1.000	1.000	1.000	**0.925**	0.918	1.000	0.850	1.000	**1.000**	1.000	1.000	1.000	1.000	0.425	0.439	0.428	0.450	0.400	0.825	0.820	0.8421	0.800	0.850
8	**1.000**	1.000	1.000	1.000	1.000	**0.975**	0.975	0.952	1.000	0.950	0.675	0.745	0.612	0.950	0.400	0.375	0.358	0.368	0.350	0.400	0.800	0.789	0.833	0.750	0.850
9	**1.000**	1.000	1.000	1.000	1.000	**0.925**	0.926	0.904	0.950	0.900	**0.925**	0.923	0.947	0.900	0.950	0.475	0.400	0.466	0.350	0.600	0.850	0.842	0.888	0.800	0.900
10	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	0.825	0.837	0.782	0.900	0.750	0.575	0.564	0.578	0.550	0.600	0.850	0.842	0.888	0.800	0.900
11	0.675	0.734	0.620	0.900	0.450	**0.975**	0.975	0.952	1.000	0.950	0.850	0.842	0.888	0.800	0.900	0.325	0.4000	0.360	0.450	0.200	0.850	0.842	0.888	0.800	0.900
12	0.775	0.808	0.703	0.950	0.600	**0.975**	0.975	0.952	1.000	0.950	0.825	0.820	0.842	0.800	0.850	0.475	0.461	0.473	0.450	0.500	0.825	0.820	0.842	0.800	0.850

Speaker1 vs Female synthetic voice with Google TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.400	0.571	0.444	0.800	0.000	0.450	0.621	0.474	0.900	0.000	0.500	0.667	0.500	1.000	0.000	0.275	0.431	0.355	0.550	0.000	0.175	0.298	0.259	0.350	0.00
2	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	0.825	0.851	0.741	1.000	0.650	0.375	0.324	0.353	0.300	0.450	0.500	0.545	0.500	0.600	0.400
3	0.775	0.816	0.690	1.000	0.550	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	0.350	0.133	0.200	0.100	0.600	0.575	0.564	0.579	0.550	0.600
4	0.825	0.851	0.741	1.000	0.650	**0.925**	0.927	0.905	0.950	0.900	**0.975**	0.974	1.000	0.950	1.000	0.350	0.187	0.250	0.150	0.550	0.550	0.438	0.583	0.350	0.750
5	0.725	0.776	0.655	0.950	0.500	0.775	0.816	0.690	1.000	0.550	0.725	0.766	0.667	0.900	0.550	0.375	0.359	0.368	0.350	0.400	0.550	0.500	0.562	0.450	0.650
6	0.600	0.714	0.556	1.000	0.200	0.550	0.640	0.533	0.800	0.300	0.800	0.833	0.714	1.000	0.600	0.350	0.316	0.333	0.300	0.400	0.550	0.500	0.562	0.450	0.650
7	**0.975**	0.976	0.952	1.000	0.950	**0.975**	0.974	1.000	0.950	1.000	0.850	0.842	0.889	0.800	0.900	0.350	0.316	0.333	0.300	0.400	0.525	0.457	0.533	0.400	0.650
8	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.00	0.775	0.816	0.690	1.000	0.550	0.425	0.439	0.429	0.450	0.400	0.550	0.471	0.571	0.400	0.700
9	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	**0.950**	0.950	0.950	0.950	0.950	0.650	0.650	0.650	0.650	0.650	0.550	0.500	0.562	0.450	0.650
10	**1.000**	1.000	1.000	1.000	1.000	**0.975**	0.976	0.952	1.000	0.950	**0.900**	0.909	0.833	1.000	0.800	0.450	0.500	0.458	0.550	0.350	0.550	0.500	0.562	0.450	0.650
11	**0.950**	0.952	0.909	1.000	0.900	**0.900**	0.909	0.833	1.000	0.800	**0.925**	0.923	0.947	0.900	0.950	0.550	0.526	0.556	0.500	0.600	0.575	0.514	0.600	0.450	0.700
12	0.700	0.769	0.625	1.000	0.400	**1.000**	1.000	1.000	1.000	1.000	0.800	0.789	0.833	0.750	0.850	0.5000	0.524	0.5000	0.550	0.450	0.625	0.571	0.667	0.500	0.750

Table 26: Out-of-sample SVMs results of EMD-MFCCs features conducted with Radial Basis function as kernel with dataset 1.

Experiment 2

Dataset 1

Speaker1 vs Female synthetic voice generated with IBM TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.475	0.644	0.487	0.950	0.000	0.475	0.644	0.487	0.950	0.000	0.500	0.667	0.500	1.000	0.000	0.100	0.182	0.167	0.200	0.000	0.175	0.298	0.259	0.350	0.000
2	0.875	0.889	0.800	1.000	0.750	**0.975**	0.974	1.000	0.950	1.000	**0.900**	0.905	0.864	0.950	0.850	0.425	0.378	0.412	0.350	0.500	0.600	0.600	0.600	0.600	0.600
3	0.625	0.727	0.571	1.000	0.250	**1.000**	1.000	1.000	1.000	1.000	**0.950**	0.952	0.909	1.000	0.900	0.450	0.312	0.417	0.250	0.650	0.400	0.429	0.409	0.450	0.350
4	0.575	0.691	0.543	0.950	0.200	0.875	0.878	0.857	0.900	0.850	**0.925**	0.927	0.905	0.950	0.900	0.450	0.312	0.417	0.250	0.650	0.650	0.632	0.667	0.600	0.700
5	0.500	0.667	0.500	1.000	0.000	0.625	0.667	0.600	0.750	0.500	0.775	0.800	0.720	0.900	0.650	0.550	0.500	0.562	0.450	0.650	0.600	0.579	0.611	0.550	0.650
6	0.500	0.667	0.500	1.000	0.000	0.825	0.837	0.783	0.900	0.750	0.550	0.654	0.531	0.850	0.250	0.475	0.364	0.462	0.300	0.650	0.600	0.600	0.600	0.600	0.600
7	0.750	0.800	0.667	1.000	0.500	**0.975**	0.976	0.952	1.000	0.950	**0.900**	0.900	0.900	0.900	0.900	0.500	0.412	0.500	0.350	0.650	0.525	0.558	0.522	0.600	0.45
8	**1.000**	1.000	1.000	1.000	1.000	**0.975**	0.976	0.952	1.000	0.950	0.725	0.776	0.655	0.950	0.500	0.500	0.375	0.500	0.300	0.700	0.575	0.585	0.571	0.600	0.550
9	**1.000**	1.000	1.000	1.000	1.000	**0.950**	0.952	0.909	1.000	0.900	**0.950**	0.950	0.950	0.950	0.950	0.600	0.556	0.625	0.500	0.70	0.375	0.390	0.381	0.400	0.350
10	**1.000**	1.000	1.000	1.000	1.000	0.825	0.844	0.760	0.950	0.700	0.625	0.694	0.586	0.850	0.400	0.575	0.541	0.588	0.500	0.650	0.575	0.564	0.579	0.550	0.600
11	**1.000**	1.000	1.000	1.000	1.000	0.775	0.816	0.690	1.000	0.550	0.675	0.667	0.684	0.650	0.700	0.550	0.471	0.571	0.400	0.700	0.600	0.579	0.611	0.550	0.650
12	**0.975**	0.976	0.952	1.000	0.950	0.625	0.727	0.571	1.00	0.25	0.850	0.864	0.792	0.950	0.750	0.600	0.556	0.625	0.500	0.700	0.575	0.585	0.571	0.600	0.550

Speaker1 vs Female synthetic voice with Sapi5 TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.150	0.261	0.231	0.30	0.000	0.250	0.400	0.333	0.500	0.000	0.475	0.644	0.487	0.950	0.000	0.550	0.182	1.000	0.100	1.000	0.825	0.788	1.000	0.650	1.000
2	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	0.875	0.872	0.895	0.850	0.900	0.425	0.343	0.400	0.300	0.550	0.725	0.703	0.765	0.650	0.800
3	**0.950**	0.952	0.909	1.000	0.900	**1.000**	1.000	1.000	1.000	1.000	**0.950**	0.952	0.909	1.000	0.900	0.425	0.303	0.385	0.250	0.600	0.725	0.686	0.800	0.600	0.850
4	0.825	0.851	0.741	1.000	0.650	**0.925**	0.927	0.905	0.950	0.900	**0.975**	0.974	1.000	0.950	1.000	0.450	0.312	0.417	0.250	0.650	0.825	0.811	0.882	0.750	0.900
5	0.700	0.760	0.633	0.950	0.450	0.775	0.791	0.739	0.850	0.700	0.475	0.222	0.429	0.150	0.800	0.500	0.333	0.500	0.250	0.750	0.825	0.811	0.882	0.750	0.900
6	**0.975**	0.974	1.000	0.950	1.000	0.725	0.686	0.800	0.600	0.850	0.775	0.809	0.704	0.950	0.600	0.500	0.333	0.500	0.250	0.750	0.750	0.722	0.812	0.650	0.850
7	**1.000**	1.000	1.000	1.000	1.000	**0.900**	0.895	0.944	0.850	0.950	**0.975**	0.974	1.000	0.950	1.000	0.525	0.424	0.538	0.350	0.700	0.750	0.706	0.857	0.600	0.900
8	**1.000**	1.000	1.000	1.000	1.000	0.825	0.788	1.000	0.650	1.000	0.700	0.769	0.625	1.000	0.400	0.500	0.333	0.500	0.250	0.750	0.775	0.743	0.867	0.650	0.900
9	**1.000**	1.000	1.000	1.000	1.000	**0.975**	0.976	0.952	1.000	0.950	**0.925**	0.923	0.947	0.900	0.950	0.500	0.231	0.500	0.150	0.850	0.725	0.703	0.765	0.650	0.800
10	**1.000**	1.000	1.000	1.000	1.000	**0.925**	0.930	0.870	1.000	0.850	0.725	0.784	0.645	1.000	0.450	0.525	0.424	0.538	0.350	0.700	0.800	0.778	0.875	0.700	0.900
11	**0.950**	0.947	1.000	0.900	1.000	0.850	0.870	0.769	1.000	0.700	0.425	0.148	0.286	0.100	0.750	0.500	0.412	0.500	0.350	0.650	0.750	0.722	0.812	0.650	0.850
12	0.700	0.727	0.667	0.800	0.600	0.875	0.889	0.800	1.000	0.750	0.750	0.706	0.857	0.60	0.90	0.500	0.412	0.500	0.35	0.65	0.800	0.778	0.875	0.700	0.900

Table 27: Out-of-sample SVMs results of EMD-MFCCs features conducted with Radial Basis function as kernel with dataset 1.

Experiment 2

Dataset 2

Speaker1 vs Female synthetic voice generated with Espeak TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	**0.986**	0.986	1.000	0.972	1.000
2	0.694	0.676	0.719	0.639	0.750	0.833	0.836	0.824	0.847	0.819	0.882	0.881	0.887	0.875	0.889	0.479	0.444	0.476	0.417	0.542	0.986	0.986	1.000	0.972	1.000
3	**1.000**	1.000	1.000	1.000	1.000	**0.993**	0.993	1.000	0.986	1.000	0.896	0.892	0.925	0.861	0.931	0.458	0.451	0.457	0.444	0.472	0.986	0.986	1.000	0.972	1.000
4	**1.000**	1.000	1.000	1.000	1.000	0.833	0.833	0.833	0.833	0.833	0.743	0.726	0.778	0.681	0.806	0.514	0.407	0.522	0.333	0.694	0.979	0.979	1.000	0.958	1.000
5	**0.993**	0.993	1.000	0.986	1.000	0.826	0.806	0.912	0.722	0.931	0.694	0.681	0.712	0.653	0.736	0.486	0.448	0.484	0.417	0.556	0.986	0.986	1.000	0.972	1.000
6	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	0.639	0.623	0.652	0.597	0.681	0.500	0.455	0.500	0.417	0.583	0.986	0.986	1.000	0.972	1.000
7	**1.000**	1.000	1.000	1.000	1.000	**0.938**	0.936	0.957	0.917	0.958	0.875	0.870	0.909	0.833	0.917	0.500	0.438	0.500	0.389	0.611	0.958	0.957	1.000	0.917	1.000
8	**1.000**	1.000	1.000	1.000	1.000	0.875	0.862	0.966	0.778	0.972	0.785	0.786	0.781	0.792	0.778	0.472	0.513	0.476	0.556	0.389	**0.979**	0.979	1.000	0.958	1.000
9	**1.000**	1.000	1.000	1.000	1.000	0.889	0.889	0.889	0.889	0.889	**0.938**	0.935	0.970	0.903	0.972	0.493	0.523	0.494	0.556	0.431	**0.958**	0.957	1.000	0.917	1.000
10	0.646	0.724	0.593	0.931	0.361	0.792	0.803	0.762	0.847	0.736	0.882	0.878	0.910	0.847	0.917	0.514	0.493	0.515	0.472	0.556	**0.972**	0.971	1.000	0.944	1.000
11	0.806	0.791	0.855	0.736	0.875	0.618	0.636	0.608	0.667	0.569	0.764	0.730	0.852	0.639	0.889	0.472	0.519	0.477	0.569	0.375	**0.951**	0.949	1.000	0.903	1.000
12	0.847	0.820	1.000	0.694	**1.000**	0.708	0.720	0.692	0.750	0.667	0.875	0.864	0.950	0.792	0.958	0.472	0.513	0.476	0.556	0.389	**0.965**	0.964	1.000	0.931	1.000

Speaker1 vs Female synthetic voice with Google TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	**0.917**	0.915	0.929	0.903	0.931	**0.944**	0.946	0.921	0.972	0.917	**0.979**	0.979	1.000	0.958	1.000	0.465	0.483	0.468	0.500	0.431	0.493	0.529	0.494	0.569	0.417
2	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	**0.938**	0.939	0.920	0.958	0.917	0.549	0.539	0.551	0.528	0.569	0.549	0.545	0.549	0.542	0.556
3	0.896	0.904	0.835	0.986	0.806	**1.000**	1.000	1.000	1.000	1.000	**0.993**	0.993	1.000	0.986	1.000	0.542	0.500	0.550	0.458	0.625	0.604	0.623	0.595	0.653	0.556
4	**1.000**	1.000	1.000	1.000	1.000	**0.979**	0.979	0.986	0.972	0.986	0.861	0.851	0.919	0.792	0.931	0.562	0.519	0.576	0.472	0.653	0.375	0.348	0.364	0.333	0.417
5	**0.979**	0.979	1.000	0.958	1.000	0.847	0.857	0.805	0.917	0.778	0.764	0.754	0.788	0.722	0.806	0.562	0.496	0.585	0.431	0.694	0.604	0.601	0.606	0.597	0.611
6	**0.986**	0.986	1.000	0.972	1.000	0.889	0.882	0.938	0.833	0.944	0.708	0.708	0.708	0.708	0.708	0.424	0.450	0.430	0.472	0.375	0.611	0.646	0.593	0.708	0.514
7	**1.000**	1.000	1.000	1.000	1.000	**0.979**	0.979	0.986	0.972	0.986	0.819	0.803	0.883	0.736	0.903	0.458	0.473	0.461	0.486	0.431	0.618	0.641	0.605	0.681	0.556
8	**1.000**	1.000	1.000	1.000	1.000	**0.979**	0.980	0.960	1.000	0.958	**0.924**	0.921	0.955	0.889	0.958	0.542	0.515	0.547	0.486	0.597	0.604	0.612	0.600	0.625	0.583
9	**1.000**	1.000	1.000	1.000	1.000	**0.993**	0.993	1.000	0.986	1.000	**0.944**	0.942	0.985	0.903	0.986	0.542	0.476	0.556	0.417	0.667	0.618	0.641	0.605	0.681	0.556
10	**1.000**	1.000	1.000	1.000	1.000	**0.972**	0.971	1.000	0.944	1.000	0.868	0.872	0.844	0.903	0.833	0.521	0.489	0.524	0.458	0.583	0.625	0.635	0.618	0.653	0.597
11	0.542	0.653	0.525	0.861	0.222	**0.958**	0.957	1.000	0.917	1.000	**0.812**	0.780	0.941	0.667	0.958	0.542	0.468	0.558	0.403	0.681	0.604	0.632	0.590	0.681	0.528
12	0.542	0.641	0.527	0.819	0.264	**0.986**	0.986	1.000	0.972	1.000	0.861	0.846	0.948	0.764	0.958	0.528	0.534	0.527	0.542	0.514	0.611	0.632	0.600	0.667	0.556

Table 28: Out-of-sample SVMs results of EMD-MFCCs features conducted with Radial Basis function as kernel with dataset 2.

Experiment 2

Dataset 2

Speaker1 vs Female synthetic voice generated with IBM TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	**0.986**	0.986	1.000	0.972	1.000	**0.958**	0.959	0.946	0.972	0.944	**0.931**	0.933	0.897	0.972	0.889	0.521	0.481	0.525	0.444	0.597	0.542	0.577	0.536	0.625	0.458
2	**0.951**	0.953	0.922	0.986	0.917	**0.972**	0.972	0.972	0.972	0.972	0.875	0.871	0.897	0.847	0.903	0.521	0.543	0.519	0.569	0.472	0.493	0.510	0.494	0.528	0.458
3	0.694	0.732	0.652	0.833	0.556	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	0.507	0.536	0.506	0.569	0.444	0.521	0.517	0.521	0.514	0.528
4	**0.903**	0.897	0.953	0.847	0.958	0.833	0.829	0.853	0.806	0.861	**0.924**	0.923	0.930	0.917	0.931	0.500	0.471	0.500	0.444	0.556	0.521	0.524	0.521	0.528	0.514
5	0.778	0.789	0.750	0.833	0.722	0.736	0.732	0.743	0.722	0.750	0.799	0.794	0.812	0.778	0.819	0.479	0.497	0.481	0.514	0.444	0.514	0.533	0.513	0.556	0.472
6	0.806	0.831	0.734	0.958	0.653	**0.951**	0.951	0.958	0.944	0.958	0.604	0.623	0.595	0.653	0.556	0.479	0.503	0.481	0.528	0.431	0.521	0.549	0.519	0.583	0.458
7	**0.986**	0.986	0.986	0.986	0.986	**0.944**	0.944	0.957	0.931	0.958	0.861	0.846	0.948	0.764	0.958	0.479	0.483	0.479	0.486	0.472	0.451	0.423	0.446	0.403	0.500
8	**1.000**	1.000	1.000	1.000	1.000	0.861	0.863	0.851	0.875	0.847	**0.944**	0.944	0.944	0.944	0.944	0.458	0.466	0.459	0.472	0.444	0.535	0.568	0.530	0.611	0.458
9	**1.000**	1.000	1.000	1.000	1.000	**0.965**	0.965	0.972	0.958	0.972	**0.979**	0.979	1.000	0.958	1.000	0.486	0.532	0.488	0.583	0.389	0.472	0.433	0.468	0.403	0.542
10	**1.000**	1.000	1.000	1.000	1.000	**0.938**	0.936	0.957	0.917	0.958	0.729	0.702	0.780	0.639	0.819	0.521	0.504	0.522	0.486	0.556	0.549	0.575	0.543	0.611	0.486
11	**1.000**	1.000	1.000	1.000	1.000	0.792	0.795	0.784	0.806	0.778	0.674	0.689	0.658	0.722	0.625	0.472	0.479	0.473	0.486	0.458	0.535	0.579	0.529	0.639	0.431
12	0.569	0.699	0.537	1.000	0.139	0.764	0.795	0.702	0.917	0.611	0.882	0.874	0.937	0.819	0.944	0.479	0.476	0.479	0.472	0.486	0.528	0.564	0.524	0.611	0.444

Speaker1 vs Female synthetic voice with Sapi5 TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	**0.924**	0.924	0.918	0.931	0.917	**0.931**	0.934	0.887	0.986	0.875	**0.972**	0.973	0.959	0.986	0.958	0.618	0.621	0.616	0.625	0.611	0.910	0.910	0.904	0.917	0.903
2	**0.993**	0.993	1.000	0.986	1.000	**0.993**	0.993	0.986	1.000	0.986	**0.965**	0.965	0.972	0.958	0.972	0.521	0.561	0.518	0.611	0.431	0.701	0.703	0.699	0.708	0.694
3	**0.875**	0.885	0.821	0.958	0.792	**0.993**	0.993	1.000	0.986	1.000	**1.000**	1.000	1.000	1.000	1.000	0.465	0.476	0.467	0.486	0.444	0.743	0.745	0.740	0.750	0.736
4	**0.993**	0.993	1.000	0.986	1.000	**0.958**	0.958	0.958	0.958	0.958	**0.903**	0.901	0.914	0.889	0.917	0.528	0.534	0.527	0.542	0.514	0.778	0.784	0.763	0.806	0.750
5	**0.944**	0.942	0.985	0.903	0.986	0.854	0.844	0.905	0.792	0.917	0.778	0.775	0.786	0.764	0.792	0.514	0.485	0.516	0.458	0.569	0.792	0.792	0.792	0.792	0.792
6	**1.000**	1.000	1.000	1.000	1.000	**0.944**	0.941	1.000	0.889	1.000	0.736	0.721	0.766	0.681	0.792	0.514	0.507	0.514	0.500	0.528	0.812	0.797	0.869	0.736	0.889
7	**1.000**	1.000	1.000	1.000	1.000	**0.979**	0.979	0.986	0.972	0.986	**0.944**	0.942	0.985	0.903	0.986	0.556	0.429	0.600	0.333	0.778	0.806	0.800	0.824	0.778	0.833
8	**1.000**	1.000	1.000	1.000	1.000	**0.903**	0.903	0.903	0.903	0.903	**0.910**	0.909	0.915	0.903	0.917	0.479	0.503	0.481	0.528	0.431	0.799	0.803	0.787	0.819	0.778
9	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	**0.965**	0.964	1.000	0.931	1.000	0.486	0.519	0.488	0.556	0.417	0.826	0.825	0.831	0.819	0.833
10	**1.000**	1.000	1.000	1.000	1.000	**1.000**	1.000	1.000	1.000	1.000	0.847	0.841	0.879	0.806	0.889	0.500	0.493	0.500	0.486	0.514	0.806	0.806	0.806	0.806	0.806
11	0.618	0.721	0.568	0.986	0.250	**0.986**	0.986	1.000	0.972	1.000	0.757	0.729	0.825	0.653	0.861	0.500	0.507	0.500	0.514	0.486	0.812	0.800	0.857	0.750	0.875
12	0.542	0.670	0.523	0.931	0.153	**0.972**	0.971	1.000	0.944	1.000	**0.924**	0.920	0.969	0.875	0.972	0.472	0.486	0.474	0.500	0.444	0.792	0.795	0.784	0.806	0.778

Table 29: Out-of-sample SVMs results of EMD-MFCCs features conducted with Radial Basis function as kernel with dataset 2.

Experiment 2					
Dataset 1					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice Espeak					
MFCC-1 8th coeff. RBF	MFCC-2 10th coeff. RBF	MFCC-3 7th coeff. RBF	MFCC-K 1st coeff. RBF	MFCC-K+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.211	0.211	0.211	0.170	0.199	**1.000**
0.263	0.263	0.263	0.212	–	**1.000**
–	0.267	0.267	0.215	0.252	**1.000**
0.267	–	0.267	0.215	0.252	**0.999**
0.267	0.267	–	0.215	0.252	**0.998**
0.254	0.254	0.254	–	0.239	**0.997**
0.333	0.333	0.333	–	–	**0.997**
0.356	0.356	–	0.287	–	**0.995**
0.340	0.340	–	–	0.321	**0.995**
–	–	0.364	0.293	0.343	0.899
0.500	0.500	–	–	–	0.899
0.356	–	0.356	0.287	–	0.899
0.340	–	0.340	–	0.321	0.899
–	0.356	0.356	0.287	–	0.898
–	0.340	0.340	–	0.321	0.898
–	0.500	0.500	–	–	0.897
0.364	–	–	0.293	0.343	0.895
–	0.554	–	0.446	–	0.895
0.500	–	0.500	–	–	0.893
–	0.364	–	0.293	0.343	0.892
–	0.514	–	–	0.486	0.890
0.554	–	–	0.446	–	0.891
–	–	–	0.460	0.540	0.888
–	–	0.554	0.446	–	0.888
–	–	0.514	–	0.486	0.887
0.514	–	–	–	0.486	0.700
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice Espeak					
MFCC 8th coeff. RBF	MFCC 8th coeff. RBF	MFCC 9th coeff. RBF	MFCC 10th coeff. RBF	MFCC 10th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.197	0.210	0. 199	0.197	0.197	0.685

Table 30: MKL-SVMs results of the synthetic voice generated with the Espeak TTS algorithm versus Speaker 1 for dataset 1. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy).

Experiment 2					
Dataset 1					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice GTTs					
MFCC-1 8th coeff. RBF	MFCC-2 8th coeff. RBF	MFCC-3 3rd coeff. RBF	MFCC-K 9th coeff. RBF	MFCC-K+1 2nd coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.247	0.247	0.247	0.151	0.110	**1.000**
0.277	0.277	0.277	0.169	–	**1.000**
0.290	0.290	0.290	–	0.129	**1.000**
0.327	0.327	–	0.200	0.145	**1.000**
–	0.327	0.327	0.200	0.145	**1.000**
0.327	–	0.327	0.200	0.145	**0.999**
0.333	0.333	0.333	–	–	**0.998**
0.383	0.383	–	0.234	–	**0.997**
0.409	0.409	–	–	0.182	**0.996**
0.383	–	0.383	0.234	–	**0.995**
0.500	0.500	–	–	–	0.899
–	0.500	0.500	–	–	0.899
0.500	–	0.500	–	–	0.899
0.409	–	0.409	–	0.182	0.898
–	0.486	–	0.297	0.216	0.898
–	0.383	0.383	0.234	–	0.898
–	0.692	–	–	0.308	0.898
–	–	0.486	0.297	0.216	0.898
–	0.621	–	0.379	–	0.897
0.621	–	–	0.379	–	0.896
0.486	–	–	0.297	0.216	0.893
–	0.409	0.409	–	0.182	0.890
0.692	–	–	–	0.308	0.803
–	–	0.692	–	0.308	0.797
–	–	–	0.579	0.421	0.795
–	–	0.621	0.379	–	0.795
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice GTTs					
MFCC 9th coeff. RBF	MFCC 8th coeff. RBF	MFCC 8th coeff. RBF	MFCC 7th coeff. RBF	MFCC 6th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.197	0.207	0.201	0.197	0.197	0.705

Table 31: MKL-SVMs results of the synthetic voice generated with the GTTs TTS algorithm versus Speaker 1 for dataset 1. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy).

Experiment 2					
Dataset 1					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice SAPI5					
MFCC-1 8th coeff. RBF	MFCC-2 9th coeff. RBF	MFCC-3 4th coeff. RBF	MFCC-K 10th coeff. RBF	MFCC-K+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.237	0.230	0.230	0.112	0.191	**1.000**
0.293	0.285	0.285	0.138	–	**1.000**
0.267	0.259	0.259	–	0.215	**1.000**
0.308	0.299	–	0.145	0.248	**1.000**
–	0.302	0.302	0.147	0.250	**0.998**
0.340	0.330	0.330	–	–	**0.998**
0.308	–	0.299	0.145	0.248	**0.975**
0.409	0.398	–	0.193	–	**0.995**
0.439	–	–	0.207	0.354	0.899
0.409	–	0.398	0.193	–	0.899
0.360	0.350	–	–	0.290	0.889
–	–	0.432	0.210	0.358	0.899
0.507	0.493	–	–	–	0.899
0.360	–	0.350	–	0.290	0.898
–	0.500	0.500	–	–	0.888
–	0.402	0.402	0.195	–	0.897
–	0.354	0.354	–	0.293	0.897
–	0.432	–	0.210	0.358	0.885
0.507	–	0.493	–	–	0.875
0.679	–	–	0.321	–	0.865
0.554	–	–	–	0.446	0.845
–	0.673	–	0.327	–	0.807
–	0.547	–	–	0.453	0.800
–	–	0.673	0.327	–	0.799
–	–	0.547	–	0.453	0.799
–	–	–	0.370	0.630	0.787
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice SAPI5					
MFCC 9th coeff. RBF	MFCC 9th coeff. RBF	MFCC 10th coeff. RBF	MFCC 10th coeff. RBF	MFCC 5th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.198	0.208	0.209	0.192	0.192	0.745

Table 32: MKL-SVMs results of the synthetic voice generated with the SAPI5 TTS algorithm versus Speaker 1 for dataset 1. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy).

Experiment 2					
Dataset 2					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice Espeak					
MFCC-1 8th coeff. RBF	MFCC-2 7th coeff. RBF	MFCC-3 9th coeff. RBF	MFCC-K 1st coeff. RBF	MFCC-K+1 3rd coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.206	0.192	0.192	0.206	0.203	**1.000**
0.259	0.241	0.241	0.259	–	**1.000**
0.260	0.242	0.242	–	0.256	**1.000**
0.255	0.238	–	0.255	0.251	**0.999**
0.255	–	0.238	0.255	0.251	**0.999**
–	0.242	0.242	0.260	0.256	**0.997**
0.350	0.325	0.325	–	–	**0.999**
0.341	0.318	–	0.341	–	**0.997**
0.343	0.319	–	–	0.338	**0.994**
0.341	–	0.318	0.341	–	**0.994**
0.343	–	0.319	–	0.338	**0.993**
0.518	0.482	–	–	–	**0.990**
–	0.327	0.327	–	0.346	**0.990**
0.335	–	–	0.335	0.330	**0.908**
–	0.325	0.325	0.350	–	**0.907**
–	0.319	–	0.343	0.338	**0.900**
–	0.500	0.500	–	–	0.888
0.518	–	0.482	–	–	0.886
0.500	–	–	0.500	–	0.885
–	0.486	–	–	0.514	0.866
–	0.482	–	0.518	–	0.850
–	–	0.319	0.343	0.338	0.798
–	–	0.486	–	0.514	0.788
–	–	0.482	0.518	–	0.789
0.504	–	–	–	0.496	0.700
–	–	–	0.504	0.496	0.708
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice Espeak					
MFCC 8th coeff. RBF	MFCC 9th coeff. RBF	MFCC 7th coeff. RBF	MFCC 8th coeff. RBF	MFCC 11th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.210	0.197	0.197	0.197	0.197	0.710

Table 33: MKL-SVMs results of the synthetic voice generated with the Espeak TTS algorithm versus Speaker 1 for dataset 2. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy).

Experiment 2					
Dataset 2					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice GTTs					
MFCC-1 7th coeff. RBF	MFCC-2 8th coeff. RBF	MFCC-3 9th coeff. RBF	MFCC-K 2nd coeff. RBF	MFCC-K+1 10th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.250	0.244	0.235	0.125	0.146	**1.000**
0.293	0.286	0.275	0.146	–	**1.000**
0.286	0.279	0.268	–	0.167	**1.000**
0.327	0.319	–	0.163	0.191	**1.000**
–	0.326	0.313	0.166	0.195	**1.000**
0.343	0.335	0.322	–	–	**1.000**
0.404	0.395	–	0.201	–	**1.000**
0.391	0.382	–	–	0.228	**1.000**
–	0.405	0.389	0.207	–	**1.000**
–	0.391	0.376	–	0.233	**1.000**
–	0.475	–	0.242	0.283	**1.000**
0.506	0.494	–	–	–	**1.000**
–	0.662	–	0.338	–	**1.000**
–	0.626	–	–	0.374	**1.000**
–	–	0.653	0.347	–	**0.993**
–	–	0.617	–	0.383	**0.993**
–	0.510	0.490	–	–	**0.993**
0.331	–	0.311	0.165	0.193	**0.993**
0.410	–	0.385	0.205	–	**0.993**
–	–	0.464	0.247	0.289	**0.993**
0.397	–	0.372	–	0.231	**0.979**
0.516	–	0.484	–	–	**0.986**
0.480	–	–	0.239	0.280	0.889
0.667	–	–	0.333	–	0.889
–	–	–	0.461	0.539	0.889
0.632	–	–	–	0.368	0.600
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice GTTs					
MFCC 8th coeff. RBF	MFCC 8th coeff. RBF	MFCC 9th coeff. RBF	MFCC 11th coeff. RBF	MFCC 10th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.208	0.198	0.198	0.198	0.198	0.778

Table 34: MKL-SVMs results of the synthetic voice generated with the GTTs TTS algorithm versus Speaker 1 for dataset 2. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy).

Experiment 2					
Dataset 2					
OFS EMD-MFCC-MKL - Speaker 1 vs Female Synthetic Voice SAPI5					
MFCC-1 8th coeff. RBF	MFCC-2 9th coeff. RBF	MFCC-3 4th coeff. RBF	MFCC-K 10th coeff. RBF	MFCC-K+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.234	0.229	0.225	0.101	0.211	**1.000**
0.297	0.290	0.286	0.127	–	**1.000**
0.261	0.255	0.251	–	0.234	**1.000**
0.303	0.296	–	0.130	0.272	**0.999**
0.304	–	0.292	0.130	0.273	**0.986**
–	0.299	0.294	0.131	0.275	**0.972**
0.340	0.332	0.327	–	–	**0.952**
0.416	0.406	–	0.178	–	**0.952**
0.348	0.340	–	–	0.313	**0.949**
0.418	–	0.402	0.179	–	**0.902**
0.350	–	0.336	–	0.314	**0.902**
0.506	0.494	–	–	–	**0.900**
–	0.413	0.406	0.181	–	0.899
–	0.344	0.339	–	0.317	0.899
–	0.424	–	0.186	0.390	.890
–	–	0.420	0.187	0.393	0.890
0.510	–	0.490	–	–	0.886
–	0.504	0.496	–	–	0.886
–	0.695	–	0.305	–	0.883
–	0.521	–	–	0.479	0.883
–	–	0.691	0.309	–	0.883
–	–	0.517	–	0.483	0.880
–	–	–	0.323	0.677	0.743
0.700	–	–	0.300	–	0.743
0.429	–	–	0.184	0.386	0.729
0.526	–	–	–	0.474	0.700
OFS Raw Data MFCC-MKL - Speaker 1 vs Female Synthetic Voice SAPI5					
MFCC 10th coeff. RBF	MFCC 10th coeff. RBF	MFCC 8th coeff. RBF	MFCC 9th coeff. RBF	MFCC 8th coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.210	0.210	0.198	0.191	0.191	0.785

Table 35: MKL-SVMs results of the synthetic voice generated with the SAPI5 TTS algorithm versus Speaker 1 for dataset 2. We select the best features according to their performances when individually tested (i.e. through the out-of-sample accuracy).

Experiment 3

Dataset 3

Female Case versus A01 TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.789	0.810	0.735	0.903	0.675	0.843	0.855	0.791	0.932	0.753	0.797	0.814	0.752	0.886	0.708	0.700	0.746	0.647	0.880	0.519	0.692	0.734	0.645	0.851	0.532
2	0.893	0.897	0.864	0.932	0.854	0.847	0.858	0.802	0.922	0.773	0.740	0.754	0.716	0.795	0.685	0.532	0.541	0.531	0.552	0.513	0.536	0.552	0.533	0.571	0.500
3	0.847	0.851	0.830	0.873	0.821	0.864	0.878	0.795	0.981	0.747	0.852	0.866	0.791	0.958	0.747	0.545	0.571	0.541	0.604	0.487	0.575	0.585	0.571	0.601	0.549
4	0.823	0.818	0.842	0.795	0.851	0.756	0.771	0.727	0.821	0.692	0.831	0.849	0.767	0.951	0.711	0.558	0.582	0.553	0.614	0.503	0.549	0.563	0.546	0.581	0.516
5	0.706	0.680	0.747	0.623	0.789	0.763	0.772	0.744	0.802	0.724	0.830	0.845	0.774	0.932	0.727	0.562	0.587	0.555	0.623	0.500	0.547	0.562	0.544	0.581	0.513
6	0.651	0.662	0.641	0.685	0.617	0.766	0.788	0.722	0.867	0.666	0.792	0.802	0.765	0.844	0.740	0.563	0.587	0.557	0.620	0.506	0.541	0.554	0.538	0.571	0.510
7	0.703	0.708	0.696	0.721	0.685	0.760	0.784	0.712	0.873	0.646	0.794	0.803	0.769	0.841	0.747	0.565	0.580	0.561	0.601	0.529	0.545	0.558	0.543	0.575	0.516
8	0.795	0.822	0.729	0.942	0.649	0.664	0.698	0.634	0.776	0.552	0.753	0.748	0.765	0.731	0.776	0.555	0.569	0.552	0.588	0.523	0.550	0.567	0.547	0.588	0.513
9	0.685	0.680	0.691	0.669	0.701	0.571	0.593	0.565	0.623	0.519	0.726	0.740	0.703	0.782	0.669	0.570	0.589	0.564	0.617	0.523	0.555	0.581	0.549	0.617	0.494
10	0.677	0.690	0.664	0.718	0.636	0.682	0.666	0.701	0.633	0.731	0.726	0.738	0.706	0.773	0.679	0.565	0.579	0.561	0.597	0.532	0.544	0.556	0.542	0.571	0.516
11	0.708	0.720	0.692	0.750	0.666	0.571	0.567	0.573	0.562	0.581	0.661	0.693	0.633	0.766	0.555	0.562	0.571	0.559	0.584	0.539	0.552	0.571	0.548	0.597	0.506
12	0.755	0.748	0.770	0.727	0.782	0.560	0.573	0.557	0.591	0.529	0.698	0.721	0.670	0.779	0.617	0.565	0.573	0.562	0.584	0.545	0.550	0.581	0.544	0.623	0.477

Female Case versus A02 TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.818	0.830	0.780	0.886	0.750	0.847	0.861	0.792	0.942	0.753	0.886	0.891	0.856	0.929	0.844	0.705	0.751	0.649	0.893	0.516	0.679	0.723	0.635	0.841	0.516
2	0.756	0.752	0.765	0.740	0.773	0.745	0.767	0.707	0.838	0.653	0.800	0.813	0.765	0.867	0.734	0.508	0.523	0.508	0.539	0.477	0.562	0.554	0.564	0.545	0.578
3	0.727	0.711	0.755	0.672	0.782	0.722	0.744	0.691	0.805	0.640	0.852	0.854	0.842	0.867	0.838	0.528	0.531	0.527	0.536	0.519	0.589	0.600	0.585	0.617	0.562
4	0.703	0.688	0.724	0.656	0.750	0.672	0.681	0.663	0.701	0.643	0.813	0.832	0.756	0.925	0.701	0.539	0.546	0.538	0.555	0.523	0.570	0.581	0.566	0.597	0.542
5	0.677	0.612	0.766	0.510	0.844	0.675	0.688	0.662	0.718	0.633	0.792	0.810	0.747	0.883	0.701	0.531	0.546	0.529	0.565	0.497	0.575	0.581	0.572	0.591	0.558
6	0.628	0.615	0.638	0.594	0.662	0.690	0.720	0.656	0.799	0.581	0.654	0.657	0.652	0.662	0.646	0.542	0.552	0.540	0.565	0.519	0.583	0.589	0.580	0.597	0.568
7	0.711	0.708	0.715	0.701	0.721	0.709	0.732	0.680	0.792	0.627	0.662	0.697	0.632	0.776	0.549	0.529	0.540	0.528	0.552	0.506	0.576	0.588	0.572	0.604	0.549
8	0.737	0.755	0.706	0.812	0.662	0.718	0.708	0.733	0.685	0.750	0.727	0.744	0.701	0.792	0.662	0.541	0.557	0.538	0.578	0.503	0.584	0.586	0.584	0.588	0.581
9	0.593	0.485	0.659	0.383	0.802	0.662	0.692	0.636	0.760	0.565	0.729	0.722	0.741	0.705	0.753	0.544	0.557	0.541	0.575	0.513	0.576	0.585	0.573	0.597	0.555
10	0.685	0.650	0.732	0.584	0.786	0.610	0.576	0.632	0.529	0.692	0.774	0.767	0.792	0.744	0.805	0.547	0.559	0.545	0.575	0.519	0.575	0.583	0.572	0.594	0.555
11	0.812	0.814	0.806	0.821	0.802	0.661	0.663	0.658	0.669	0.653	0.739	0.750	0.718	0.786	0.692	0.544	0.557	0.541	0.575	0.513	0.576	0.580	0.575	0.584	0.568
12	0.732	0.724	0.747	0.701	0.763	0.674	0.678	0.669	0.688	0.659	0.706	0.711	0.699	0.724	0.688	0.545	0.562	0.542	0.584	0.506	0.575	0.576	0.574	0.578	0.571

Female Case versus A04 TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.701	0.727	0.669	0.795	0.607	0.713	0.740	0.676	0.818	0.607	0.726	0.755	0.681	0.847	0.604	0.683	0.739	0.629	0.896	0.471	0.690	0.743	0.634	0.896	0.484
2	0.664	0.670	0.658	0.682	0.646	0.677	0.709	0.645	0.786	0.568	0.649	0.683	0.623	0.756	0.542	0.516	0.515	0.516	0.513	0.519	0.542	0.548	0.541	0.555	0.529
3	0.589	0.599	0.585	0.614	0.565	0.688	0.731	0.643	0.847	0.529	0.698	0.733	0.657	0.828	0.568	0.550	0.564	0.547	0.581	0.519	0.565	0.571	0.563	0.578	0.552
4	0.615	0.639	0.602	0.682	0.549	0.643	0.664	0.627	0.705	0.581	0.669	0.717	0.626	0.841	0.497	0.544	0.559	0.541	0.578	0.510	0.563	0.564	0.563	0.565	0.562
5	0.541	0.543	0.540	0.545	0.536	0.578	0.577	0.578	0.575	0.581	0.679	0.718	0.640	0.818	0.539	0.539	0.548	0.537	0.558	0.519	0.573	0.581	0.571	0.591	0.555
6	0.573	0.592	0.567	0.620	0.526	0.575	0.624	0.559	0.705	0.445	0.651	0.669	0.636	0.705	0.597	0.532	0.541	0.531	0.552	0.513	0.571	0.577	0.570	0.584	0.558
7	0.614	0.614	0.614	0.614	0.614	0.664	0.698	0.634	0.776	0.552	0.542	0.567	0.538	0.601	0.484	0.545	0.561	0.542	0.581	0.510	0.578	0.583	0.576	0.591	0.565
8	0.620	0.634	0.611	0.659	0.581	0.664	0.692	0.638	0.756	0.571	0.576	0.590	0.571	0.610	0.542	0.547	0.556	0.545	0.568	0.526	0.573	0.581	0.571	0.591	0.555
9	0.662	0.667	0.658	0.675	0.649	0.623	0.645	0.610	0.685	0.562	0.544	0.537	0.545	0.529	0.558	0.537	0.540	0.537	0.542	0.532	0.571	0.578	0.569	0.588	0.555
10	0.578	0.574	0.579	0.568	0.588	0.562	0.533	0.570	0.500	0.623	0.589	0.580	0.593	0.568	0.610	0.541	0.540	0.541	0.539	0.542	0.570	0.575	0.568	0.581	0.558
11	0.581	0.596	0.576	0.617	0.545	0.581	0.584	0.580	0.588	0.575	0.547	0.569	0.543	0.597	0.497	0.552	0.563	0.549	0.578	0.526	0.571	0.578	0.569	0.588	0.555
12	0.604	0.631	0.590	0.679	0.529	0.523	0.539	0.521	0.558	0.487	0.597	0.611	0.591	0.633	0.562	0.541	0.547	0.539	0.555	0.526	0.571	0.581	0.568	0.594	0.549

Table 36: Out-of-sample SVMs results of EMD-MFCCs features for the female case conducted with Radial Basis function as kernel with Dataset 3

Experiment 3

Dataset 3

Male Case versus A01 TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.729	0.752	0.693	0.821	0.636	0.776	0.792	0.740	0.851	0.701	0.756	0.782	0.708	0.873	0.640	0.571	0.676	0.543	0.896	0.247	0.584	0.691	0.550	0.929	0.240
2	0.729	0.766	0.673	0.890	0.568	0.631	0.667	0.609	0.737	0.526	0.679	0.693	0.664	0.724	0.633	0.510	0.481	0.511	0.455	0.565	0.490	0.502	0.491	0.513	0.468
3	0.763	0.776	0.735	0.821	0.705	0.588	0.657	0.562	0.789	0.386	0.727	0.765	0.672	0.886	0.568	0.518	0.496	0.520	0.474	0.562	0.503	0.522	0.503	0.542	0.464
4	0.735	0.731	0.744	0.718	0.753	0.532	0.573	0.527	0.627	0.438	0.599	0.680	0.566	0.854	0.344	0.521	0.514	0.522	0.506	0.536	0.518	0.547	0.516	0.581	0.455
5	0.666	0.683	0.649	0.721	0.610	0.675	0.708	0.643	0.789	0.562	0.581	0.670	0.553	0.851	0.312	0.541	0.549	0.539	0.558	0.523	0.506	0.521	0.506	0.536	0.477
6	0.597	0.565	0.615	0.523	0.672	0.576	0.584	0.574	0.594	0.558	0.549	0.635	0.533	0.786	0.312	0.513	0.505	0.513	0.497	0.529	0.505	0.512	0.505	0.519	0.490
7	0.701	0.698	0.705	0.692	0.711	0.633	0.623	0.640	0.607	0.659	0.609	0.655	0.586	0.744	0.474	0.519	0.518	0.520	0.516	0.523	0.505	0.504	0.505	0.503	0.506
8	0.761	0.763	0.759	0.766	0.756	0.472	0.510	0.476	0.549	0.396	0.659	0.679	0.642	0.721	0.597	0.539	0.555	0.536	0.575	0.503	0.539	0.549	0.537	0.562	0.516
9	0.539	0.570	0.534	0.610	0.468	0.537	0.544	0.536	0.552	0.523	0.537	0.574	0.532	0.623	0.451	0.534	0.533	0.534	0.532	0.536	0.503	0.513	0.503	0.523	0.484
10	0.649	0.672	0.631	0.718	0.581	0.597	0.624	0.585	0.669	0.526	0.619	0.637	0.608	0.669	0.568	0.541	0.554	0.538	0.571	0.510	0.524	0.536	0.523	0.549	0.500
11	0.672	0.664	0.680	0.649	0.695	0.555	0.597	0.546	0.659	0.451	0.524	0.567	0.520	0.623	0.425	0.537	0.540	0.537	0.542	0.532	0.502	0.510	0.502	0.519	0.484
12	0.677	0.660	0.697	0.627	0.727	0.586	0.649	0.563	0.766	0.406	0.519	0.591	0.514	0.695	0.344	0.532	0.543	0.531	0.555	0.510	0.500	0.511	0.500	0.523	0.477

Male Case versus A02 TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.792	0.805	0.757	0.860	0.724	0.836	0.845	0.802	0.893	0.779	0.771	0.789	0.733	0.854	0.688	0.735	0.763	0.691	0.851	0.620	0.724	0.763	0.669	0.886	0.562
2	0.651	0.678	0.630	0.734	0.568	0.661	0.682	0.642	0.727	0.594	0.851	0.850	0.855	0.844	0.857	0.531	0.552	0.528	0.578	0.484	0.542	0.548	0.541	0.555	0.529
3	0.795	0.792	0.805	0.779	0.812	0.615	0.650	0.596	0.714	0.516	0.852	0.854	0.842	0.867	0.838	0.531	0.533	0.531	0.536	0.526	0.526	0.534	0.525	0.542	0.510
4	0.735	0.736	0.735	0.737	0.734	0.523	0.576	0.518	0.649	0.396	0.744	0.766	0.705	0.838	0.649	0.529	0.520	0.530	0.510	0.549	0.536	0.557	0.533	0.584	0.487
5	0.656	0.661	0.651	0.672	0.640	0.596	0.599	0.594	0.604	0.588	0.666	0.715	0.623	0.838	0.494	0.539	0.530	0.541	0.519	0.558	0.544	0.557	0.541	0.575	0.513
6	0.599	0.592	0.603	0.581	0.617	0.545	0.532	0.548	0.516	0.575	0.610	0.634	0.598	0.675	0.545	0.536	0.545	0.534	0.555	0.516	0.542	0.559	0.539	0.581	0.503
7	0.685	0.683	0.688	0.679	0.692	0.625	0.632	0.621	0.643	0.607	0.752	0.745	0.766	0.724	0.779	0.541	0.534	0.542	0.526	0.555	0.547	0.561	0.544	0.578	0.516
8	0.688	0.687	0.690	0.685	0.692	0.567	0.595	0.558	0.636	0.497	0.838	0.825	0.894	0.766	0.909	0.526	0.531	0.525	0.536	0.516	0.541	0.556	0.538	0.575	0.506
9	0.563	0.603	0.553	0.662	0.464	0.610	0.632	0.599	0.669	0.552	0.753	0.723	0.825	0.643	0.864	0.528	0.537	0.526	0.549	0.506	0.532	0.541	0.531	0.552	0.513
10	0.679	0.689	0.668	0.711	0.646	0.606	0.623	0.596	0.653	0.558	0.737	0.733	0.745	0.721	0.753	0.529	0.540	0.528	0.552	0.506	0.532	0.547	0.530	0.565	0.500
11	0.700	0.706	0.692	0.721	0.679	0.557	0.546	0.560	0.532	0.581	0.659	0.671	0.648	0.695	0.623	0.541	0.543	0.540	0.545	0.536	0.528	0.540	0.526	0.555	0.500
12	0.719	0.717	0.723	0.711	0.727	0.638	0.658	0.623	0.698	0.578	0.617	0.650	0.598	0.711	0.523	0.536	0.553	0.533	0.575	0.497	0.531	0.541	0.530	0.552	0.510

Male Case versus A04 TTS Algorithm

MELFCC	IMF 1					IMF 2					IMF 3					IMF K					IMF K+1				
	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.	Accuracy	F1-score	Prec.	Sens.	Spec.
1	0.627	0.677	0.597	0.782	0.471	0.638	0.698	0.599	0.838	0.438	0.614	0.678	0.581	0.812	0.416	0.555	0.662	0.534	0.870	0.240	0.558	0.669	0.535	0.893	0.224
2	0.547	0.613	0.535	0.718	0.377	0.552	0.630	0.537	0.763	0.341	0.567	0.617	0.553	0.698	0.435	0.529	0.526	0.530	0.523	0.536	0.485	0.478	0.485	0.471	0.500
3	0.568	0.591	0.561	0.623	0.513	0.521	0.616	0.514	0.769	0.273	0.591	0.679	0.559	0.864	0.318	0.519	0.523	0.519	0.526	0.513	0.518	0.506	0.519	0.494	0.542
4	0.581	0.594	0.576	0.614	0.549	0.511	0.590	0.508	0.705	0.318	0.558	0.646	0.539	0.805	0.312	0.528	0.524	0.528	0.519	0.536	0.513	0.510	0.513	0.506	0.519
5	0.545	0.581	0.539	0.630	0.461	0.539	0.585	0.532	0.649	0.429	0.541	0.662	0.524	0.899	0.182	0.526	0.518	0.527	0.510	0.542	0.487	0.485	0.487	0.484	0.490
6	0.463	0.504	0.468	0.545	0.380	0.528	0.554	0.525	0.588	0.468	0.524	0.636	0.515	0.831	0.218	0.536	0.536	0.536	0.536	0.536	0.489	0.489	0.489	0.490	0.487
7	0.539	0.540	0.539	0.542	0.536	0.567	0.594	0.559	0.633	0.500	0.489	0.588	0.492	0.731	0.247	0.524	0.511	0.526	0.497	0.552	0.495	0.488	0.495	0.481	0.510
8	0.505	0.506	0.505	0.506	0.503	0.536	0.604	0.527	0.708	0.364	0.567	0.606	0.556	0.666	0.468	0.532	0.534	0.532	0.536	0.529	0.487	0.485	0.487	0.484	0.490
9	0.479	0.498	0.480	0.516	0.442	0.549	0.590	0.541	0.649	0.448	0.547	0.595	0.538	0.666	0.429	0.531	0.533	0.531	0.536	0.526	0.484	0.489	0.484	0.494	0.474
10	0.549	0.574	0.544	0.607	0.490	0.576	0.621	0.562	0.695	0.458	0.610	0.636	0.597	0.682	0.539	0.532	0.540	0.531	0.549	0.516	0.482	0.488	0.483	0.494	0.471
11	0.567	0.582	0.562	0.604	0.529	0.547	0.605	0.536	0.695	0.399	0.599	0.649	0.577	0.740	0.458	0.531	0.533	0.531	0.536	0.526	0.487	0.494	0.487	0.500	0.474
12	0.482	0.474	0.482	0.468	0.497	0.562	0.626	0.546	0.734	0.390	0.552	0.642	0.535	0.802	0.302	0.532	0.549	0.530	0.568	0.497	0.482	0.483	0.482	0.484	0.481

Table 37: Out-of-sample SVMs results of EMD-MFCCs features for the male case conducted with Radial Basis function as kernel with Dataset 3

Appendix I

The third algorithm shows the procedure adopted in the Multi Kernel Learning section to compute MKL-SVM out-of-sample results. Remark that π_m is the accuracy of each SVM and the index m refers to the feature taken into account. The process can be synthesized as follows: (1) by only considering the out of sample performance, compute for each kernel in each feature $\pi_m - \delta$; (2) take the average over all the IMFs index within each kernel corresponding to $s_m = \sum_{i=1}^{5} (\pi_{i,m} - \delta)/5$, where the index i relates to the IMF index. (3) Choose the kernel with the higher score computed in (2) for each feature. (4) Select the features that have to be considered in the multi-kernel learning process and then standardise according to 4.13, so that each η_m can be obtained. (5) Run a final SVM with the new computed kernel on the testing set.

Algorithm 9: MKL algorithm

Input: Out of sample accuracies π_m for each selected feature and kernel
Test training (20 sentences)
Output: MKL-SVM out-of-sample
begin
  Set $\delta = 0.1$
  For each kernel and for each feature
  for $i = 1$ **to** 5 **do**
    compute $s_m = \sum_{i=1}^{5} (\pi_{i,m} - \delta)/5$
  Select the best kernel for each feature according to the scores s_m
  Compute the final weights as $\eta_m = s_m / \sum_{m=1}^{M} s_m$ where M can vary depending on the SVM
  Compute a final out of sample SVM by using the new computed kernel according to 4.12.

Experiment 3					
Dataset 3					
OFS EMD-MFCC-MKL - Female Case vs A04 TTS Algorithm					
MFCC-1 1st coeff. RBF	MFCC-2 1st coeff. RBF	MFCC-3 1st coeff. RBF	MFCC-K 1st coeff. RBF	MFCC-K+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
–	0.495	0.505	–	–	**0.925**
–	0.512	–	0.488	–	**0.917**
–	0.336	0.343	0.320	–	**0.911**
–	0.335	0.342	–	0.323	0.891
0.327	0.333	0.340	–	–	0.890
–	0.509	–	–	0.491	0.886
0.495	0.505	–	–	–	0.883
0.248	0.253	0.258	0.241	–	0.881
–	0.254	0.259	0.242	0.245	0.881
0.335	0.341	–	0.325	–	0.860
–	0.343	–	0.327	0.330	0.860
0.247	0.252	0.258	–	0.243	0.847
0.200	0.203	0.208	0.194	0.196	0.846
–	–	0.517	0.483	–	0.839
0.252	0.257	–	0.244	0.247	0.823
0.333	0.340	–	–	0.327	0.813
–	–	0.348	0.324	0.328	0.773
0.332	–	0.346	0.322	–	0.771
0.251	–	0.261	0.243	0.246	0.750
–	–	0.515	–	0.485	0.744
0.490	–	0.510	–	–	0.742
0.508	–	–	0.492	–	0.727
–	–	–	0.497	0.503	0.726
0.331	–	0.344	–	0.325	0.721
0.339	–	–	0.329	0.332	0.716
0.505	–	–	–	0.495	0.500
OFS Raw Data MFCC-MKL - Female Case vs A04 TTS Algorithm					
MFCC 1st coeff. RBF	MFCC 1st coeff. RBF	MFCC 2nd coeff. RBF	MFCC 4th coeff. RBF	MFCC 2nd coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.240	0.238	0.166	0.189	0.166	0.680

Table 38: Multi Kernel Learning SVMs results of the female case versus the synthetic voice generated with the A04 TTS algorithm of the ASVspoof challenge dataset.

Experiment 3					
Dataset 3					
OFS EMD-MFCC-MKL - Male Case vs A04 TTS Algorithm					
MFCC-1 1st coeff. RBF	MFCC-2 1st coeff. RBF	MFCC-3 1st coeff. RBF	MFCC-K 1st coeff. RBF	MFCC-K+1 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
–	0.542	–	0.458	–	**0.912**
–	0.512	0.488	–	–	**0.904**
–	0.357	0.341	0.302	–	0.896
–	0.540	–	–	0.460	0.880
0.495	0.505	–	–	–	0.878
–	0.371	–	0.314	0.316	0.878
0.347	0.354	–	0.300	–	0.877
–	0.356	0.340	–	0.304	0.872
0.334	0.341	0.325	–	–	0.862
–	0.274	0.261	0.232	0.233	0.862
0.259	0.265	0.253	0.224	–	0.856
0.346	0.353	–	–	0.301	0.852
0.266	0.272	–	0.230	0.232	0.851
0.259	0.264	0.252	–	0.225	0.846
0.211	0.216	0.206	0.183	0.184	0.844
–	–	0.530	0.470	–	0.825
–	–	0.360	0.319	0.321	0.781
0.352	–	0.343	0.304	–	0.779
0.270	–	0.263	0.233	0.235	0.766
–	–	0.528	–	0.472	0.763
0.506	–	0.494	–	–	0.758
0.351	–	0.343	–	0.306	0.739
–	–	–	0.498	0.502	0.718
0.536	–	–	0.464	–	0.705
0.366	–	–	0.316	0.318	0.693
0.535	–	–	–	0.465	0.500
OFS Raw Data MFCC-MKL - Male Case vs A04 TTS Algorithm					
MFCC 1st coeff. RBF	MFCC 2nd coeff. RBF	MFCC 2nd coeff. RBF	MFCC 5th coeff. RBF	MFCC 1st coeff. RBF	Accuracy
η_1	η_2	η_3	η_4	η_5	
0.189	0.198	0.212	0.212	0.189	0.662

Table 39: Multi Kernel Learning SVMs results of the male case versus the synthetic voice generated with the A04 TTS algorithm of the ASVspoof challenge dataset.

Appendix H

We assume that the vectors ψ^i for $i = 1, \ldots, S$ are iid realisations from $g(\psi; \varphi)$. We assume that the elements of the random vector ψ are independent random variables such that their joint distribution can be factorised as

$$g(\psi; \varphi) = \prod_{m=1} g_{\omega_m}(\omega_m; \varphi_m) \prod_{d=1}^{D} g_{s_{m,d}}(s_{m,d}; \varphi_{m,d}) \tag{14}$$

We assume that each probability distribution $g_x(x; \varphi_x)$ corresponds a univariate normal distribution parametrized by mean μ_x and a standard divination σ_x^2. Hence, $\varphi_x = [\mu_x, \sigma_x^2]$. We expect that the iterative updates of the importance sampling distribution shrink σ_x^2 towards zero and consequently, the sampling distribution become a degenerate one and the μ_x would correspond to optimal partition of $\mathcal{I}$ and $\mathcal{T}$. Therefore, we propose

$$W_m \sim \mathcal{N}\Big(\mu_m, \sigma_m^2\Big) \text{ and } S_{m,d} \sim \mathcal{N}\Big(\mu_{m,d}, \sigma_{m,d}^2\Big) \tag{15}$$

where for the probability density function of $X \sim \mathcal{N}\Big(\mu, \sigma^2\Big)$ such that $a \leq X \leq b$ is the following

$$g_x(x, \mu, \sigma^2) = \frac{1}{\sigma}\phi\left(\frac{x-\mu}{\sigma}\right),$$

where $\phi(x)$ is the probability density function of a standard normal random variable

$$\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-0.5x^2}. \tag{16}$$

In order for the sampled ψ to belong to the feasible set Ψ, we need $\omega_0 \leq W_m \leq \omega_M$ and $t_0 \leq S_{m,d} \leq t_N$. The objective function of the estimation problem is then formulated as

$$\Lambda(\phi) = \sum_{i=1}^{S} \left\{ \mathbf{1}_{\left\{KL(\hat{\pi}, \pi; \psi^{(s)}) \leq \gamma\right\}} \sum_{m=1}^{M} \sum_{d=1}^{D} \left\{ \log g_{\omega_m}(\omega_m^{(s)}; \varphi_m) + \log g_{s_{m,d}}(s_{m,d}^{(s)}; \varphi_{m,d}) \right\} \right\} \tag{17}$$

with maximizers with respect to ϕ given by

$$\begin{cases} \frac{\partial \Lambda(\phi)}{\partial \mu_m} = D \sum_{i=1}^{S} \left\{ \mathbf{1}_{\left\{KL(\hat{\pi}, \pi; \psi^{(s)}) \leq \gamma\right\}} \frac{\omega_m^{(s)} - \mu_m}{\sigma_m^2} \right\} = 0 \\ \frac{\partial \Lambda(\phi)}{\partial \sigma_m^2} = \frac{D}{2} \sum_{i=1}^{S} \left\{ \mathbf{1}_{\left\{KL(\hat{\pi}, \pi; \psi^{(s)}) \leq \gamma\right\}} \left(-\frac{1}{\sigma_m^2} + \frac{(\omega_m^{(s)} - \mu_m)^2}{(\sigma_m^2)^2} \right) \right\} = 0 \\ \frac{\partial \Lambda(\phi)}{\partial \mu_{m,d}} = \sum_{i=1}^{S} \left\{ \mathbf{1}_{\left\{KL(\hat{\pi}, \pi; \psi^{(s)}) \leq \gamma\right\}} \frac{s_{m,d}^{(s)} - \mu_{m,d}}{\sigma_{m,d}^2} \right\} = 0 \\ \frac{\partial \Lambda(\phi)}{\partial \sigma_{m,d}^2} = \frac{D}{2} \sum_{i=1}^{S} \left\{ \mathbf{1}_{\left\{KL(\hat{\pi}, \pi; \psi^{(s)}) \leq \gamma\right\}} \left(-\frac{1}{\sigma_{m,d}^2} + \frac{(s_{m,d}^{(s)} - \mu_{m,d})^2}{(\sigma_{m,d}^2)^2} \right) \right\} = 0 \end{cases}$$

and consequently

$$\hat{\mu}_m = \frac{\sum_{i=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\leq\gamma\right\}}\omega_m^{(s)}}{\sum_{i=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\leq\gamma\right\}}},\ \hat{\sigma}_m^2 = \frac{\sum_{i=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\leq\gamma\right\}}(\omega_m^{(s)} - \mu_m)^2}{\sum_{i=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\leq\gamma\right\}}},$$

$$\hat{\mu}_{m,d} = \frac{\sum_{i=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\leq\gamma\right\}}s_{m,d}^{(s)}}{\sum_{i=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\leq\gamma\right\}}},\ \hat{\sigma}_{m,d}^2 = \frac{\sum_{i=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\leq\gamma\right\}}(s_{m,d}^{(s)} - \mu_{m,d})^2}{\sum_{i=1}^{S} \mathbf{1}_{\left\{KL(\hat{\pi},\pi;\psi^{(s)})\leq\gamma\right\}}}.$$