FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Predictive models in the diagnosis of Parkinson's disease through voice analysis

Tomás Freitas Gonçalves

Master in Informatics and Computing Engineering

Supervisor: João Correia dos Reis

February 23, 2024

Predictive models in the diagnosis of Parkinson's disease through voice analysis

Tomás Freitas Gonçalves

Master in Informatics and Computing Engineering

February 23, 2024

Abstract

Parkinson's disease (PD) is a chronic and progressive long-term degenerative disorder of the central nervous system. First described in 1817 by Doctor James Parkinson, it is known to mainly affect the motor system of its patients. The symptoms usually develop slowly, starting with symptoms in the motor system and potentially evolving into non-motor symptoms as the disease worsens. It is estimated that 90% of patients with Parkinson's end up having speech disorders.

This dissertation delves into the domain of predictive models for Parkinson's disease analysis through speech recordings, exploring the application of Machine Learning (ML) approaches. Commencing with an in-depth examination of the disease's background, encompassing causes, symptoms, and diagnostic methodologies, it is followed by a comprehensive literature review investigating ML techniques for Parkinson's disease detection, emphasizing data collection, dysphonia measures, data wrangling, and classification methodologies.

The implementation phase unfolds with an exploration of diverse datasets, including the *Italian Parkinson's Voice and Speech* dataset*, Mobile Device Voice Recordings at King's College London, Synthetic Vowels of Speakers with Parkinson's Disease and Parkinsonism* dataset and *Voice Samples for Patients with Parkinson's Disease and Healthy Controls* dataset. The approach encompasses feature extraction, data exploration, modeling, and evaluation. Two types of feature extraction are used, incorporating traditional acoustic features and embeddings extracted using pre-trained Deep Neural Networks (DNN).

The two types of features adopted displayed robustness, as the use of interpretable features reached up to 95% accuracy, while non-interpretable embeddings had a top-performing value of 99%. Although non-interpretable embeddings revealed higher accuracies in top-performing algorithms, it came at the cost of efficiency, taking up to 10 times more during the training phase. The discussion section dissects the efficiency of the models, presenting a nuanced analysis of their predictive performance and computational efficiency, setting up a transition into evaluating the viability of a future clinical setting implementation.

Resumo

A doença de Parkinson (PD) é uma doença degenerativa crónica e progressiva do sistema nervoso central. Descrita pela primeira vez em 1817 por James Parkinson, é conhecida por afetar principalmente o sistema motor do paciente. Os sintomas desenvolvem-se lentamente, começando por afetar o sistema motor e evoluindo potencialmente para sintomas não motores ao longo da progressão da doença. Estima-se que 90% dos pacientes com doença de Parkinson desenvolvam distúrbios na fala eventualmente.

Esta dissertação investiga o domínio dos modelos preditivos para análise da doença de Parkinson por meio de gravações de voz, explorando abordagens de Machine Learning (ML). Começando por apresentar informações relevantes sobre a doença, abrangendo causas, sintomas e metodologias de diagnóstico, segue-se uma revisão da literatura, investigando técnicas de ML para deteção da doença de Parkinson, enfatizando a colheita de dados, medidas de disfonia, exploração e transformação de dados e métodos de classificação.

A fase de implementação desenrola-se com a exploração de diversos conjuntos de dados, incluindo o *Italian Parkinson's Voice and Speech* dataset, *Mobile Device Voice Recordings at King's College London* dataset, *Synthetic Vowels of Speakers with Parkinson's Disease and Parkinsonism* dataset, e *Voice Samples for Patients with Parkinson's Disease and Healthy Controls* dataset. A abordagem abrange extração de características, exploração de dados, criação de modelos e avaliação. Dois tipos de extração de características são utilizadas, incorporando características acústicas tradicionais em primeiro lugar, e representações (embeddings) extraídas usando Redes Neuronais Profundas (DNN) pré-treinadas.

Os dois métodos de extração de características mostraram robustez. A utilização de características interpretáveis tradicionais atingiu um máximo de 95% de precisão, enquanto as representações não interpretáveis apresentaram um valor de desempenho máximo de 99%. Embora estas representações exibam valores de precisão mais altos nos seus algoritmos de melhor desempenho, existem compromissos ao nível da eficiência, onde a fase de treino pode demorar até 10 vezes mais. A secção de discussão disseca a eficiência dos modelos, apresentando uma análise diferenciada de sua capacidade preditiva e eficiência computacional, estabelecendo uma transição para avaliar a viabilidade duma implementação futura em ambientes clínicos.

Acknowledgements

I must express my sincere appreciation to my advisor, Prof. João Correia dos Reis, whose continuous support and expertise significantly contributed to the development and refinement of this dissertation. Special thanks to Dra. Margarida Calejo and Mariana Seco for their invaluable assistance in shaping the medical foundation of this research.

The deepest gratitude to my parents for their encouragement (and temporary — I hope — financial support) throughout my academic journey. Their endless incentive has been the cornerstone of my success, giving me the strength to overcome challenges. Their belief in my abilities has been a constant source of motivation, and I am truly fortunate to have such role models in my life.

To my friends, who have brought me motivation to study, laughter, and often generous judgments regarding the stereotypes associated with my academic degree. The shared stories throughout the years have made this journey not just bearable but truly enjoyable.

Tomás Freitas Gonçalves

"In theory, there is no difference between theory and practice,
In practice, there is."

Benjamin Brewster

Contents

List of Figures

List of Tables

Abbreviations and Symbols

DNN	Deep Neural Network
HC	Healthy Control
IQR	Interquartile Range
KDE	Kernel Density Estimate
KNN	K-Nearest Neighbor
LR	Logistic Regression
MDVR-KCL	Mobile Device Voice Recordings at King's College London dataset
ML	Machine Learning
NB	Naive Bayes
PwP	Patient(s) with Parkinson
PD	Parkinson's Disease
PCA	Principal Component Analysis
RF	Random Forest
SVM	Support Vector Machines
TRAP	Tremors, Rigidity, Akinesia, and Postural instability
UPDRS	Unified Parkinson's Disease Rating Scale
VIF	Variance Inflation Factor
VSP-PDHC	Voice Samples for Patients with Parkinson's Disease and Healthy Controls dataset

Chapter 1

Introduction

1.1 Context

Parkinson's disease (PD) is a chronic and progressive long-term degenerative disorder of the central nervous system. First described in 1817 by Doctor James Parkinson, it is known to mainly affect the motor system of its patients. Although the exact causes of the disease are unknown, genetic and environmental factors are believed to cause it. In Portugal, around 20 thousand people are estimated to suffer from PD. The symptoms usually develop slowly, starting with signs in the motor system and potentially evolving into non-motor symptoms, such as dementia, as the disease worsens. The most obvious ones are tremors, rigidity, stiffness, and slowness of movements. Due to the lack of control of the larynx and facial muscles at a later stage, patients usually display an absence of expression and a monotonic voice. It is estimated that 90% of patients with Parkinson's end up having speech disorders, a consequence of the decrease of dopamine production in the midbrain, which causes a chemical imbalance and hinders the precision of movement.

The following work proposes to identify PD based on recordings of human voice signals. Different data collection methods will be studied in order to maximize the performance of the proposed classification system. Essential features should be extracted, and Machine Learning (ML) classification should be applied to classify the presence of Parkinson's disease based on the features extracted.

Building on the knowledge gained from exploring the disease, the work will study the ability to differentiate Parkinson's patients from non-Parkinson ones using a classification system. Particular emphasis will be placed on assessing the significance of various factors, including feature extraction methods, data-wrangling techniques, and classification.

1.2 Motivation

Within the discipline of artificial intelligence, ML has become a revolutionary field that enables computers to learn from data and make accurate predictions or judgments without requiring explicit programming. With its remarkable capacity to extract patterns combined with the vast amounts of data available today, ML has changed a wide range of sectors and areas, and healthcare is no exception. With the rapid evolution of technology, its role in our society's well-being is more relevant than ever.

It is estimated that treatment and care for a patient with PD in Europe costs between 2620€ and 9820€ [4]. It is known that patients' voices become abnormal from the early stages of PD, and analyzing it can be essential to help professionals identify early signs of PD. In the future, it could allow patients access to technology that can be used by themselves, requiring only a microphone, such as those in smartphones.

Data is more accessible to collect and store than ever before, making it highly available for personal and professional use. Considering that modern hospitals should be able to collect information from many patients, the data can be used along with ML technologies to help doctors diagnose patients, facilitating the process.

To grasp how such a technology can be possible, we must first learn more about the disease and how doctors can recognize it.

1.3 Objectives and Research Questions

The main objective of the following work is to evaluate the possibility of developing a tool to classify PD cases through voice recordings. Other objectives would be analyzing the state of the art, comparing and understanding the effects of different recording methods, comparing different machine learning algorithms, and validating and analyzing results.

It is crucial to keep in mind the ethical implications of this work, as data collection should be consented to, and recordings should not be able to be traced back to patients.

Considering this, it is possible to formulate the following research questions:

- Can embeddings generated by Deep Neural Networks (DNN) provide better results than acoustic features extracted from the audio?
- Since DNN-based embeddings require more computational power, how viable are they for clinical settings?

1.4 Dissertation structure

This document is structured as follows:

Introduction: The Introduction presents the dissertation's context, motivation, main objectives, and research questions.

Disease Background: Provides a comprehensive overview of the subject matter, offering valuable insights into the disease's causes, symptoms, and diagnosis.

Literature review: Synthesizes the existing related work relevant to the dissertation's topic. It contains details on data collection, measures of dysphonia in patients, treatment of data, and machine learning classification and results.

Implementation: This chapter is dedicated to detailing the practical aspects of the research. It begins by presenting the datasets employed in the study, offering an overview. Subsequently, the chapter outlines the approach followed in the implementation, focusing on critical steps such as feature extraction, data exploration, wrangling, modeling, and evaluation.

Tests and Results: The Tests and Results chapter details the testing strategy and presents the outcomes of each iteration of the implemented machine learning models. The section on Tests divides the chapter into multiple phases, describing the conditions of each distinct iteration of our process. The Results section presents the findings from each iteration of the testing phases.

Discussion: Critical examination of the predictive metrics and observations outlined from the Results.

Conclusion: Culmination of the entire dissertation and provides its final reflection and summary.

Chapter 2

Disease Background

2.1 Causes

The Substantia Nigra is a section of the midbrain that has a crucial role in controlling the motor movement of the body. It releases dopamine, allowing communication between different parts of the nervous system. These messages between neurotransmitters are essential for good motor coordination in the human body [1].

Following the death of the nerve cells in the Substantia Nigra, dopamine production between these receptors is vastly reduced. This chemical imbalance causes a handicap in the motor skills of the patient's body [5], resulting in rigidity, tremors, instability, or bradykinesia (slowness of movement). As such, it is considered to be directly responsible for PD.

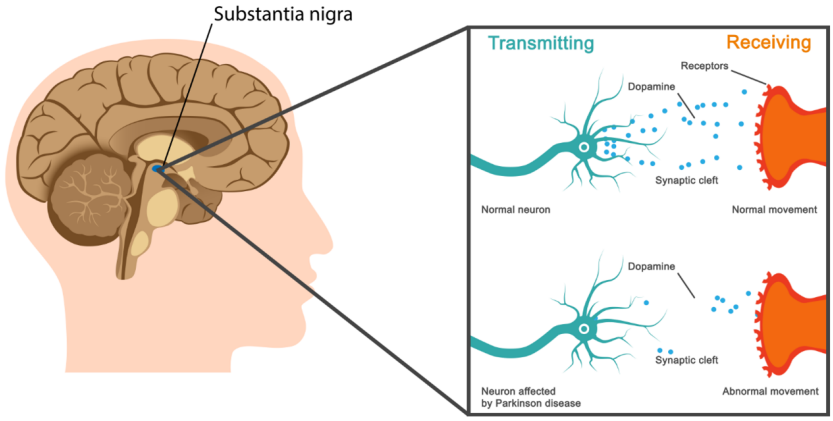

Figure 2.1: Dopamine production in a normal neuron compared to a neuron affected by PD.

The death of the nerve cells in the Substantia Nigra happens progressively. Symptoms start showing when the number of healthy nerve cells reaches around 20% [6].

The exact causes of this deterioration are still mostly unknown, and some theories point to a mutation in the neurons that synthesize dopamine (Dopaminergic cell group) [6]. It seems that some patients, especially elders, have also experienced PD symptoms after taking certain prescription drugs [7]. According to epidemiology studies, certain risk factors, like exposure to toxins and some chemicals, such as pesticides, are believed to have a catalytic effect on PD.

2.2 Symptoms

Although symptoms vary from patient to patient, the disease is known to affect the motor system of its patients mainly, and symptoms appear slowly. The first signs become visible in the motor system and eventually evolve into other non-motor symptoms, such as dementia, as the disease progresses. The most visible signs are tremors, rigidity, stiffness, and slowness of movements. Due to the lack of control of the larynx and facial muscles at a later stage, the voice starts being affected, along with the patient's face commonly displaying an absence of expression.

The acronym TRAP is a common mnemonic representing the acute symptoms of PD: Tremors, Rigidity, Akinesia, and Postural instability [8].

- **Tremors**

  Tremors correspond to the shakiness of body parts when at rest, supported by gravity, and not performing any specific activity. Present in around 70% of patients; this corresponds to the most common initial presentation of symptoms.

- **Rigidity**

  Rigidity refers to the stiffness and resistance present in the muscles. It occurs in between 70% and 90% of PD patients, and it can hinder the free movement of the body. However, in the initial phases of the disease, it can be mistaken for other orthopedic problems, such as joint pain or arthritis.

- **Akinesia**

  Akinesia is a decrease in the ability to initiate and perform smooth movements. There is also an increase in reaction times. Patients often describe it as a form of weakness or clumsiness.

- **Postural instability**

  Postural instability consists of a tendency to fall, the inability to keep a firm and steady posture, and a constant feeling of imbalance.

In addition to the TRAP symptoms, PD can directly or indirectly cause several other non-motor symptoms, such as depression, anxiety, sleep deprivation, increased blood pressure, heart rate, and digestion difficulties.

2.2.1 Voice

Apart from the five main symptoms described previously, Patients with Parkinson's Disease (PwP) demonstrate disturbances in their voices. This can be considered the primary non-motor signal displayed, and it is estimated that 90% of PwP end up having speech disorders [1, 9]. Since the muscles in the mouth become impaired, primarily due to akinesia, there is a decrease in amplitude and speed in movements of the jaw, tongue, and lips [10]. The voice can become monotone and hoarse, and patients can have difficulty pronouncing certain words or sounds and experience reduced loudness [1, 11]. In phonation, affected patients display difficulty closing vocal folds, which causes instability in the vibration patterns. There is an observable impact in the mean fundamental frequency, jitter, and shimmer values [12].

The following work will mainly focus on this aspect of the disease. It will study in detail the causes and consequences of PD on speech and the ability to recognize cases of PD from voice analysis.

2.3 Diagnosis

PD is considered a clinical diagnosis. This means it is subjective and depends on the doctor's judgment according to the evaluation of the signs observed in the patient [13].

2.3.1 Unified Parkinson's disease rating scale

Doctors use the unified Parkinson's disease rating scale (UPDRS) to evaluate the severity of the symptoms of PD. This scale was updated in 2007 by the Movement Disorder Society, and the most common terminology used today is MDS-UPDRS [14, 15, 1].

During the examination, the doctor will ask the patient to perform specific tasks while he evaluates them. Each part of UPDRS is analyzed separately, and the final result is the sum of all the scores. The higher the score obtained, the more critical the symptoms. This score can be used to evaluate the progression of PD and its response to medication.

The examination consists of four parts:

- **Part I**: consists of the non-motor symptoms associated with the disease.
- **Part II**: evaluates the motor symptoms related to the day-to-day life of the patient in question, such as rigidity, shakiness, and akinesia.
- **Part III**: assesses the motor examinations like balance, posture, and walking. This part has the most considerable influence on the final score.

- **Part IV**: relates to motor complications in response to medication. As such, it is not evaluated in non-medicated patients.

2.3.2 Sustained phonetic exercises

One of the main ways of recognizing the effects of PD on a patient's voice is to ask him to perform sustained phonetic exercises. This type of exercise seems to be the most common speech-recording method for work related to the detection of PD. It is one of many components (along with clinical evaluation, medical history, and neurological examination) of a comprehensive assessment for PD. [16, 3, 17, 18, 19]

These consist of sustaining a particular vowel (usually */a/* or */e/* sounds) for several seconds. Sustained phonetic exercises allow the evaluation of certain aspects of the voice, like its pitch range, stability, and breath control, and allow for the extraction of essential characteristics like mean fundamental frequency, measures of jitter and shimmer, harmonics-to-noise, etc... The exercises can reveal abnormalities to the evaluation professional, and deviation from expected results can indicate the existence of the disease.

Chapter 3

Literature Review: Machine Learning approaches to PD detection

This chapter will provide an overview of the research regarding ML techniques for detecting PD through voice recordings. As such, it will enable us to find valuable knowledge and methods to apply previous advancements in the domain to our problem.

For each of the studies reviewed, advantageous methods can be extracted to combine the most practical techniques in each of the studies, allowing our final approach to be differentiated and tailored to our goal.

The share of the extracted features, pre-processing details, and code for each of the studied approaches is vastly helpful in comparing methodologies and the robustness of results. However, few studies display their speech recordings, specifically used features, algorithms, and code. Similar sets of features and speech feature extraction libraries are used in some cases where this information is available.

3.1 Data collection

The study in [2] aims to evaluate the potential of voice recordings as a screening tool for PD. It is mentioned that most other studies use acoustically controlled conditions and high-quality, specialized voice recording equipment for voice recordings. In this case, the differentiation point is that the data is collected in resource-constrained settings. The authors used standard telephone networks. This allowed for a much more considerable amount of data. The authors gathered data from a population of 6531 people (1078 PD and 5453 control).

The advantages of this approach are that recordings are much easier to gather, and extensive data can be extracted quickly and efficiently. This study can be instrumental in understanding the prediction capabilities of lower-resolution recordings.

The study in [20] used three speech databases. Two of them were from Portuguese speakers and had no ambient noise control measures. They were collected from employees of the *Portuguese Association of Parents and Friends of the Mentally Deficient Citizen* [21]. The authors of another study personally provided the first Portuguese database, collecting data from 22 PwPs between the ages of 44 and 79, containing 1002 speech lines recorded using a portable voice recorder connected to a microphone. Each patient has attributed a level between 1 and 4, depending on the severity of the disease. The authors then collected the second database, which comprises 30 healthy speakers between 20 and 71, containing 785 speech lines closely following the script of the first database. Similarly to the first study, the collection was performed using the microphone from a headset device. The third database, sourced from the *UCI Machine Learning Repository* [22], includes sustained vowel recordings from 28 PD patients, with 18 recordings used in this study.

The UCI repository used in [20] and the study in [2] asked their participants to perform sustained vowel sounds (*/a/* or */o/*). This exercise seems to be the most common and is also used in the studies [16, 3, 17, 18, 19]. Sustained vowels are often preferred over free speech since the method is generic and language-independent. It helps overcome problems from different accents, intonation, and linguistic backgrounds. It is simple for all participants to perform without requiring many instructions and helps prevent fatigue in patients with advanced stages of PD.

Some studies propose the evaluation of free speech. Despite being affected by the mentioned factors, it often requires patients to read while speaking. The multi-tasking and the effects of PD on eye movement can create a bias in the collected data [23].

A sampling speed of 44.1kHz is generally used, and audio samples are often trimmed to clips of one and two seconds of unaffected pitch and intonation to prevent issues with high signal fluctuation. It is common to record multiple utterances per patient to obtain an average feature vector for each person [20, 19, 24, 3, 25, 11].

The study in [19] focuses on the recording of sound over a smartphone. It compares the performance of professional microphones and smartphones. The authors state that using professional microphones in previous works achieved accuracy rates between 0.85 and 1, while a similar methodology applied to smartphone recordings resulted in an accuracy rate of 0.9294. However, it can be challenging to compare results from different studies since there are differences in the databases, such as the age of participants, stage of PD, and gender balance. The study mentions that there is a considerable degradation in performance when changing from controlled to uncontrolled conditions.

From the mentioned studies, it can assumed that recording on a smartphone will be sufficient to allow decently accurate predictions as long as it is done in controlled conditions (voices recorded at home by the patients themselves or at the hospital following a recording protocol and control of the information about patients by the medical doctor at a charge). Sustained phonation exercises seem appropriate.

Since one of the primary focuses is to develop a ML pipeline that classifies audio, it should be able to receive voice recordings and use this input as a starting point. As such, starting the pipeline from raw voice recording files is imperative. In the quest to gather the necessary data for training and testing our algorithm, the objective is to find databases that provide raw voice recordings in their original audio file formats. There appeared to be a limited number of databases that met this criteria, emphasizing the scarcity of such resources.

Four audio file databases were found:

- Italian Parkinson's Voice and Speech Database. [26].
- Mobile Device Voice Recordings at King's College London (MDVR-KCL) [27]
- The Synthetic Vowels dataset from the Czech Republic [28].
- The Voice Samples for Patients with Parkinson's Disease and Healthy Controls (VSP-PDHC) dataset [29].

3.2 Dysphonia Measures

3.2.1 Acoustic features

The fundamental feature (*F0*) is recurrent among studies and considered crucial for speech signal analysis [2, 30, 14, 1]. It refers to the lowest frequency component of a sound wave. For human speech, it is the rate at which the muscles in the larynx vibrate during the performed phonation exercises. As such, *F0* determines the pitch of the voice—meaning that on a person with a higher-pitched voice, the mean *F0* frequency will be higher. Considering this, the mean *F0* value for a healthy male is around 120Hz and 220Hz for a healthy female [3].

Jitter and Shimmer are characterized by the deviation in *F0*'s frequency and amplitude, respectively [30, 1]. As mentioned, these characteristics seem to be highly affected in PwP's speech.

The Praat software [31] is a free computer software package for speech analysis in phonetics. It was designed and developed by Professors of the University of Amsterdam. Using Praat, it is possible to perform voice signal analysis, measure acoustic properties, and study phonetics. It is mentioned in a vast number of other studies and appears to be the most commonly used tool for audio feature extraction. [17, 20, 18, 3, 32, 33].

Parselmouth is the Python library that provides a Python interface for Praat. It allows researchers and developers to apply the functionalities of Praat to their Python applications. With Parselmouth, it is possible to automate phonetic analyses, extract acoustic features from audio recordings, and perform various tasks related to speech signal processing. It seems to be the most common way to extract the traditional measures for fundamental frequency, jitter, shimmer, and harmonics-to-noise ratios used in the analysis of PD.

The code made available by the author of [34] [1] uses Parselmouth to extract a set of acoustic features. The other studies identified in the literature [34, 1, 35, 36, 37, 38, 30, 11] consistently demonstrate a shared practice of extracting a highly similar set of features to the one mentioned before. All the evidence and uniformity across studies suggest that Praat and Parselmouth are established standards for extracting acoustic features of speech. The commonality in feature sets indicates a consensus within the research community regarding the relevance and effectiveness of these features.

3.2.2 Non-interpretable features

The study in [38] addresses a significant gap in the existing literature by comparing interpretable and non-interpretable speech-based representations for the automatic detection of PD. The study employs six datasets comprising speech recordings from diverse languages, enabling evaluation in mono-lingual, multi-lingual, and cross-lingual scenarios. In the study, the authors want to challenge the assumption that complex models consistently outperform simpler ones.

The interpretable features encompass prosodic, linguistic, and cognitive descriptors observable in the voice, while non-interpretable features include embedding features extracted from pre-trained Deep Neural Networks (x-vectors, TRILLsson, Wav2Vec 2.0, and HuBERT) [38]. The authors describe the extracted embeddings as a condensed and semantically rich representation of the input audio signals.

3.3 Pre-Processing

Data normalization is often a common concept in pre-processing and is found across multiple studies. To avoid problems with scaling and make data compatible with most ML techniques, it usually needs to be centered and scaled [3, 25, 20]. Possibilities for scaling include *StanderScaler* [20] and *MinMaxScaler* [39].

Since large numbers of noisy and redundant features negatively affect the performance of the learning algorithms, dimensionality reduction, feature transformation, and feature selection can also be considered.

Feature reduction is another essential concept in ML. It allows us to reduce the complexity of our models by giving a subset of essential features. The Variable Inflation Factor (VIF) is a metric that calculates the amount of multicollinearity in a set of multiple variables. By calculating the VIF for all the features in the dataset, we can distinguish which variables have high levels of multicollinearity. Removing variables whose VIF values are over a certain threshold can be helpful for ML algorithms that are not robust to dependent predictors [40].

The studies in [20, 15, 2] use Principal Component Analysis (PCA) to transform a significant number of correlated features into a smaller number of uncorrelated orthogonal features called

[1] The code was reviewed in conjunction with the official documentation of the Parselmouth library

principal components. However, in some cases, PCA may lead to worse results. PCA assumes a linear relationship between variables. If the relationship is non-linear, it may not capture the relevant information effectively [41]. In this case, non-linear dimensionality reduction techniques can be more appropriate. To support the outcomes of PCA, the authors in [20] also use a linear regression technique. If both obtain the same results, the risk of ill effects is reduced due to the techniques' limitations.

In scenarios where the dataset is limited, or there is a considerable class imbalance (meaning that single classes are over-represented), there is a possibility that the model is unable to learn to differentiate between classes, opting to decide for the most represented class. One possible approach is to assign weights to each class based on the occurrences during training [12]. Data augmentation can also be used to artificially increase the size of the dataset by applying transformations to the existing data. This approach helps prevent overfitting caused by the model memorizing specific patterns in the training data instead of learning generalizable features [15].

3.4 Classification and Results

For the classification step, several ML algorithms can be used. The best algorithm can differ depending on the dataset's characteristics and features. After reviewing different studies, a list of the algorithms used and the studies in which they are used were extracted.

- Support vector machines [39, 14, 19, 3, 36, 11, 15, 1, 35, 42]
- Random Forest [1, 43, 14, 19]
- Gradient Boosting [1, 25, 19]
- Logistic Regression [1, 43, 19, 34, 20, 24, 15, 44]
- k-Nearest Neighbour [1, 14, 3]
- Naive-Bayes [3, 14, 34, 11, 24, 1, 20, 42]
- Adaboost [42, 12, 24]

Since the performance of algorithms depends heavily on the dataset and features, it is only fair to compare the algorithms when the other variables are identical. The accuracies for each classifier can be compared inside the scope of each study for studies that used more than one classifier.

The authors in [1] found that, although results were good across all the algorithms used, Random Forest outperformed the other classifiers with 97%. For the classification, the authors of this study use a dataset comprising a range of biomedical tone of voice estimations. The results obtained in the study are represented by the table extracted from the study, in 3.1.

Table 3.1: Comparison of results of all algorithms used in [1].

Classifier	Test data score	Train data score	Total accuracy score
SGD	0.20	0.80	91.666667%
XGB	0.8461538461538461	1.0	95.081967213114
Logistic regression	0.87179487179487	0.82608695	0.91666666666
Random forest	0.9487179487179487	1.0	0.9710144927536232
KNN	0.9230769230769231	1.0	0.9545454545454546
Decision tree	0.7948717948717948	0.8173913043478261	0.9583333333333334

The study in [2] obtained a top-balanced accuracy of 67.34% using 27 acoustic features and a 10-fold cross-validation approach. The accuracy obtained for each algorithm is given in the table 3.2. The best accuracy corresponds to the SVM algorithm. However, the other algorithms seem to get similar results.

Although the performance of this model can seem somewhat modest compared to other studies, this can be attributed to the difference in recording conditions. Participants used their telephones over phone calls with variable quality, background noise, and signal strength, as opposed to the acoustically controlled conditions of other studies. The study works with data collected in significantly simplified recording conditions.

Table 3.2: Out-of-sample performance measures for the three classifiers using 10-fold CV with 100 iterations on the balanced dataset in [2].

Classifier	**Number of Optimal Features**	**Sensitivity**	**Specificity**	**Balanced Accuracy**
SVM	27	**67.43%**	**67.25%**	67.34%
Random Forests	27	66.38%	66.20%	66.29%
Adaboost	27	63.11%	63.60%	63.36%

Note: highest scores are highlighted in bold.

For the study in [3], most algorithms presented high performances, with the best being SVM and KNN, with an accuracy of 96%. This study used 126 features extracted from a multi-level analysis of isolated word recordings. The results can be observed in the table 3.3 extracted from the study.

Table 3.3: Performance comparison among the 6 classifiers used in [3].

Classifier	**Male**		**Female**	
	Validation set	**Test set**	**Validation set**	**Test set**
SVM	96% ± 3.22	74% ± 18.95	98% ± 2.46	90% ± 7.12
DT	95% ± 4.46	64% ± 17.34	100% ± 0	65% ± 19.56
NB	73% ± 41.10	50% ± 28.36	92% ± 5.65	77% ± 22.36
kNN	96% ± 2.46	74% ± 15.56	99% ± 1.61	97% ± 3.42
Ensemble bagged trees	92% ± 5.05	60% ± 19.56	96% ± 1.31	56% ± 0
Ensemble subspace discriminant	94% ± 5.26	71% ± 16.29	99% ± 1.31	96% ± 3.42

The study in [35] extracts 17 features (containing traditional acoustic and non-traditional measures). From these features, a set of 10 highly uncorrelated measures is selected. The authors provide the best results from the exhaustive search of all possible combinations of these measures in order to obtain the best accuracy value. Four different combinations were able to attain the highest accuracy of 91.4%.

The study in [38] presents results for three types of experiments:

- **Mono-lingual:** models trained and tested on six different data sets separately.
- **Multi-lingual:** models trained using training data from all the languages and tested on each separately.
- **Cross-lingual:** models trained with data from all but one language used for testing.

The three setups allow us to compare the performance of interpretable and non-interpretable features. The multi-lingual and cross-lingual also let us explore the language robustness of the features used and investigate whether using more data from different languages benefits the classification results.

For interpretable feature-based models, the mean of the best F1-scores obtained from each language was 0.81 in mono-lingual, 0.81 in multi-lingual, and 0.71 in cross-lingual experiments. For non-interpretable feature-based models, instead, they were 0.85 in mono-lingual, 0.88 in multi-lingual, and 0.79 in cross-lingual experiments. Specifically, using TRILLsson for feature extraction provided the most stable and accurate results across tasks and data sets.

The results obtained in [38] show that in experiments using recordings from one unique language as well as those that are multi-language, non-interpretable features outperform interpretable ones, mainly using the heavier models HuBERT, Wav2Vec 2.0, and TRILLsson. However, the study mentions that the performance discrepancy between using interpretable and non-interpretable features wasn't always remarkable. The interpretable features do not significantly narrow the performance gap with the non-interpretable models. Nonetheless, they facilitate interpretability and error analysis.

The study concludes that both interpretable and non-interpretable models exhibit satisfactory generalization capabilities in multi-lingual and cross-lingual scenarios. It suggests potential clinical applications, such as using interpretable features for simpler models to predict PD presence and progression, providing insights into the relationship between speech patterns and the disease.

From the literature review, no single algorithm or method is associated with the highest results. As mentioned, the model's accuracy depends on many other factors. Consequently, different combinations of techniques can be established to find the pipeline that gives the best classification results for the datasets.

Chapter 4

Implementation

4.1 Datasets

The effectiveness of predictive models in analyzing Parkinson's disease through speech recordings depends on the quality and diversity of the data used for training and evaluation. This section presents an overview of the datasets assembled to develop the models.

This research methodology prioritized using datasets that supply raw audio files rather than relying on extracted features. This preference derives from the ambition to construct a robust pipeline that initiates directly from the voice recordings. While datasets with raw audio files are instrumental in achieving this goal, it is worth noting that such comprehensive datasets are inherently scarce and challenging to find online, primarily due to ethical and privacy reasons, accentuating the significance of the data used for this study.

The following sections describe the particularities of each dataset, diving deeper into their distinctive characteristics.

4.1.1 Italian Parkinson's Voice and Speech

The Italian Parkinson's Voice and Speech dataset [26] serves as a foundational element for these predictive models. This dataset comprises recordings from individuals across various age groups and health conditions, providing diverse samples. The participants involved in this dataset were categorized as follows:

- **Healthy Group 1:** 15 individuals aged between 19 and 29.
- **Healthy Group 2:** 22 individuals aged between 60 and 77.
- **PD Patients:** 28 individuals aged between 40 and 80 years.

The study was conducted in 2017, and all participants were recruited from Italy. The severity of PD in patients was assessed using the Hoehn and Yahr scale, with twenty-six patients below stage

4, one at stage 4, and one at stage 5. The participants read exercises and pronounced vowels (/a/, /e/, /i/, /o/, /u/ sounds) within an echo-free room, maintaining a distance of 15cm to 25cm from the microphone. Multiple voice samples were collected from each participant, resulting in a total of 495 recordings of vowel pronunciations. The dataset was often referred to as the "Italian dataset" during the following chapters of the dissertation.

4.1.2 Mobile Device Voice Recordings at King's College London (MDVR-KCL)

The MDVR-KCL dataset [27] captures the voice recordings of 37 participants collected at King's College London in September 2017. The participants are categorized as follows:

- **Healthy group:** 21 individuals
- **PwP:** 16 individuals

Situated within a typical examination room with approximately ten square meters and a reverberation time of about 500ms, the dataset leverages a realistic scenario resembling a phone call setup. The participants held the phone to their preferred ear, ensuring that the microphone was in direct proximity to the mouth. The recording procedure utilized a Motorola Moto G4 Smartphone.

Participants read the same paragraph of text during the recording sessions. Each participant's data was labeled with the Hoehn & Yahr (H&Y) score, UPDRS II part 5, and UPDRS III part 18 scale ratings, indicating the stage of the disease. These ratings were determined through expert assessments. Each recording lasts approximately 150 seconds and has been segmented into smaller parts, generating around 22 segments per recording by splitting at locations where the speaker was silent for half a second or more (the data processing method was collected from the study in [34]). The final dataset contains 816 PwP.

4.1.3 Synthetic Vowels of Speakers with Parkinson's Disease and Parkinsonism

The Synthetic Vowels dataset [28] from the Czech Republic includes recordings from 83 participants with the following characteristics:

- **PD Patients:** 22 individuals
- **Multiple System Atrophy Patients:** 21 individuals
- **Progressive supranuclear palsy Patients:** 18 individuals
- **Healthy patients:** 22 individuals

The recordings involved 337 sustained vowels in a low ambient noise environment, recorded using a Beyerdynamic Opus 55 headset condenser microphone. Trained specialists instructed participants to perform prolonged vowels (*/a/* and */i/*) in modal voice, with at least two repetitions for each vowel. In the case of this dataset, only the PD and healthy samples were used, resulting in a

total of 92 recordings. Similar to the first dataset, for simplification purposes, it will be referred to as the "Czech Dataset."

4.1.4 Voice Samples for Patients with Parkinson's Disedataset'sealthy Controls (VSP-PDHC)

The dataset is from 2023 and was extracted from the study in [29]. The audio was recorded using participants' telephones and focused on the prolonged pronunciation of the vowel */a/*. The final dataset contains a total of 81 recordings, and the population of the studying is the following:

- **Healthy control patients:** 41 individuals
- **PD Patients:** 40 individuals

These diverse datasets constitute a rich source of information for training and evaluation. The variations in age, health conditions, and recording conditions contribute to the robustness and generalizability of the models.

The data presented in the table 4.1 encapsulates the components extracted from the datasets and utilized in the study. It is important to note that the information in this table represents exclusively the parts of the datasets pertinent to this study's research objectives. Nonessential data within the datasets, which do not contribute directly to the scope of the study, have been intentionally excluded from the information in the table.

Table 4.1: Main characteristics of the used datasets

Datasets	Total # of audio files	Minutes of audio	Patients	PD	HC	Vocal exercises	Recording device
Italian	495	44:19	65	43%	57%	Sustained vowels	Microphone
MDVR-KCL	816	17:08	37	58%	42%	Reading text	Smartphone
Czech	92	24:00	44	52%	48%	Sustained vowels	Microphone
VSP-PDHC	81	02:42	81	49%	51%	Sustained vowels	Smartphone

4.2 Approach

The research focuses on a structured approach, encompassing five key phases: Feature Extraction, Data exploration and wrangling, Modeling, and Evaluation. Embracing an iterative methodology, each phase is continuously refined and adjusted to optimize the performance of previous iterations. This iterative nature of the process reflects a commitment to continuous improvement, aiming to fine-tune the model and enhance its diagnostic accuracy.

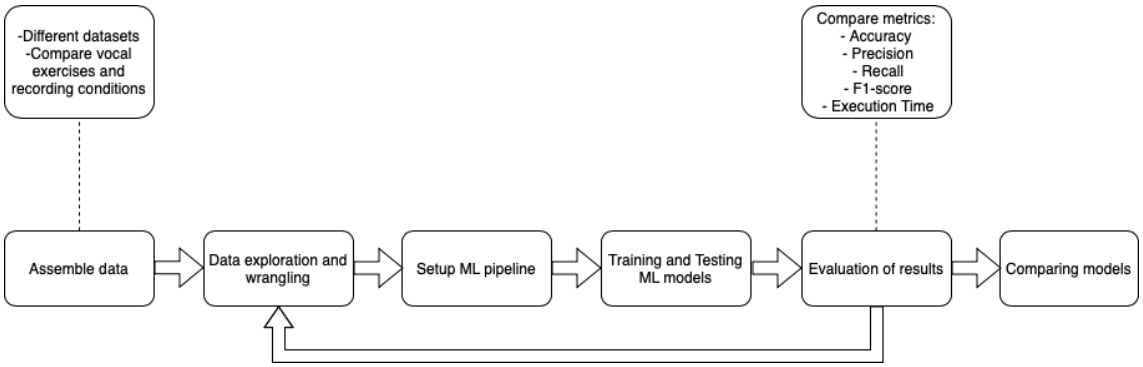

Figure 4.1: Work Pipeline

4.2.1 Feature Extraction

Two distinct characterizations were adopted to represent speech traits connected to the beginning and progression of PD. The first involves interpretable features, encompassing pitch, fundamental frequency, shimmer values, and other relevant acoustic features. The second relies on non-interpretable features derived from embeddings extracted using pre-trained DNNs.

The datasets were organized into distinct directories based on the two categories of interest: Healthy Control (HC) and Parkinson's Disease (PD). An iterative process was employed, where each audio file was assigned a binary label. A new feature column, PD, acted as the classification model's target variable. Audio files located within folders corresponding to PwP were assigned a value of 1, indicating the presence of PD. In contrast, those from the Control folders received a value of 0, signifying the absence of the condition.

Regarding interpretable features, Parselmouth was used to extract the traditionally studied features [29, 38, 33, 17]. The table 4.2 displays the 15 extracted features, providing a general description of each. [35, 36, 33, 29, 45]

Table 4.2: Interpretable features extracted using Parselmouth

Feature	Description
duration	Duration of the audio clip
meanF0	Mean fundamental frequency
stdevF0	Standard deviation of fundamental frequency
hnr	Harmonics to noise ratio
localabsoluteJitter	Average absolute difference between consecutive periods of fundamental frequency (μs)
rapJitter	Relative Average Perturbation
ppq5Jitter	Average absolute difference between a period and the average of its four closest neighbors (percentage)
ddpJitter	Average absolute difference between consecutive differences between consecutive periods
localShimmer	Average absolute difference between the amplitudes of consecutive periods
localdbShimmer	Average absolute base-10 logarithm of the difference between the amplitudes of consecutive periods (dB)
apq3Shimmer	Three-point Amplitude Perturbation Quotient
apq5Shimmer	Five-point Amplitude Perturbation Quotient
apq11Shimmer	Eleven-point Amplitude Perturbation Quotient
ddaShimmer	Average absolute difference between consecutive differences between the amplitudes of consecutive periods

The feature extraction methods for non-interpretable embeddings followed the work by [38]. Before feature extraction, all recordings were iteratively resized to 16 kHz, and the models used were the following.

TRILLsson

TRILLsson is a model designed for paralinguistic feature extraction, focusing on non-verbal elements of speech, like tone, pitch, and rhythm. It is tailored for speech emotion recognition, synthetic speech detection, and dysarthria classification. The model undergoes fully self-supervised training on Audioset [46] and Libri-light datasets [47]. The teacher model (CAP12) is trained on YT-U. Built upon the CAP12 model, TRILLsson utilizes a Conformer-based architecture with over 600 million parameters. TRILLsson models are created through knowledge distillation from the original CAP12. This distillation process transfers knowledge from the larger CAP12 to a smaller, faster, and more mobile-friendly TRILLsson model while maintaining performance. In terms of usage, TRILLsson is compatible only with 16 kHz audio files. Due to the complexity of feature extraction from long recordings, speech recordings are divided into 10-second segments.

For each 10-second segment, TRILLsson extracts fixed-sized embeddings of 1024 features. The final embedding for a complete recording is obtained by averaging the embeddings extracted from its 10-second segments.

Wav2Vec 2.0

Wav2Vec 2.0 is a self-supervised architecture designed for learning speech representations. Featuring 95 million parameters, it is pre-trained on LibriSpeech (LS-960) [48] and has demonstrated effectiveness in capturing robust features for various speech-related tasks, including speaker identification and verification, dementia, dysfluency, detection of vocal fatigue, and PD. A ported version of S3PRL's Wav2Vec 2.0 base model is used for the SUPERB benchmark. The model is adapted for PD detection by employing a single-layer representation, with 768-dimensional intermediate representations from the 4th layer identified as optimal. In terms of usage, long recordings are separated into 10-second segments. The model extracts embeddings for every 20ms within each 10s audio segment. A 1-D feature vector of size 768 representing each audio segment is obtained by computing the mean of the embeddings along the time axis.

HuBERT

HuBERT (Hidden-unit BERT model) is a prominent self-supervised learning model featuring 90 million parameters for speech recognition. It is also pre-trained on the LibriSpeech dataset [48]. A ported version of S3PRL's HuBERT base model is utilized in this study. A single-layer representation is employed for PD detection, with the 7th layer identified as optimal. In terms of usage, it is similar to Wav2Vec, as long speech recordings are segmented into 10-second intervals, zero-padded if shorter. The model extracts embeddings at each layer, generating an embedding for every 20ms within each 10s audio segment. A 1-D feature vector of size 768 is derived by computing the mean of the embeddings along the time axis. The final embedding for each recording is determined as the average of feature vectors from its 10s segments.

4.2.2 Data exploration and wrangling

After performing the extraction of features, a comprehensive data exploration and wrangling process was executed over the set of interpretable features to ensure the dataset's quality and relevance for subsequent modeling. The following key actions were performed:

- **Handling Missing Values and Type Consistency** Rows containing undefined values were dropped from the dataset. Additionally, a thorough verification was undertaken to confirm that all dataset values were the same type.

- **Data Exploration:** Boxplots and Kernel Density Estimate (KDE) plots of all variables were generated to visualize data distribution, identify potential outliers, and gain insights into the data distribution.

Furthermore, pair plots for each pair of variables were generated, and the correlation between variables was calculated to understand the relationships and dependencies within the dataset.

The percentage of cases in each class was compared for every dataset to assess the balance of the target variable.

- **Feature Engineering:** The duration column, deemed irrelevant for the study, was removed from the datasets to enhance its relevance for subsequent modeling.
- **Outlier Treatment:** The Inter-quartile Range Method (IQR) [49] was employed for outlier reduction. It identifies outliers in numerical columns (except for the target column) and adjusts them to be within a reasonable range defined by the upper whisker *Q3* and lower bound *Q1*. This helps handle extreme values that might impact the analysis or modeling process.
- **Data Normalization:** The dataset underwent *MinMax* normalization to standardize the scale of features, facilitating more effective modeling [34, 39].
- **Class Imbalance Check:** An assessment was made to determine the percentage of representation of each class in the dataset and whether or not there was an imbalance in its distribution.
- **Multicollinearity Analysis:** VIF were computed for each independent variable to address potential multicollinearity issues. Variables with high VIF scores, indicating a strong correlation with others, were considered for removal [40].
- **Principal Component Analysis:** PCA was employed to explore dimensionality reduction and capture essential information in the data. It allows dimensionality reduction by using an orthogonal transformation to transform a set of correlated variables into a shorter set of values of linearly uncorrelated variables [32, 36, 15]. Two sets of variables were combined through PCA and are demonstrated in the following Table 4.3:

Table 4.3: Generated features through PCA

Original features	Generated features
localJitter localabsoluteJitter rapJitter ppq5Jitter ddpJitter	JitterPCA
localShimmer localdbShimmer apq3Shimmer aqpq5Shimmer apq11Shimmer ddaShimmer	ShimmerPCA

For the non-interpretable features, the generated features were scaled using a *StandardScaler*, following the approach by the authors of [38].

4.2.2.1 Oservations

After conducting the exploration and wrangling of the interpretable features, several key observations emerged.

The examination of *null* values and data consistency across the different datasets revealed varying degrees of completeness.

- The Italian dataset displayed 988 non-null entries.
- The MDVR-KCL dataset initially comprised 1630 rows, but after the wrangling process, it was reduced to 1584 rows.
- The Czech dataset had 182 non-null rows.
- VSP-PDHC dataset had 160 non-null rows.

All the variables consisted uniquely of *float64* values.

From the table 4.1, a noteworthy finding was that classes were well-distributed across all datasets, indicating a balanced representation of the two categories.

In order to visualize the effects of the outlier reduction in the data, KDE plots and boxplots were generated for all the variables before and after the process. The following section will use plots from the Italian dataset as an example.

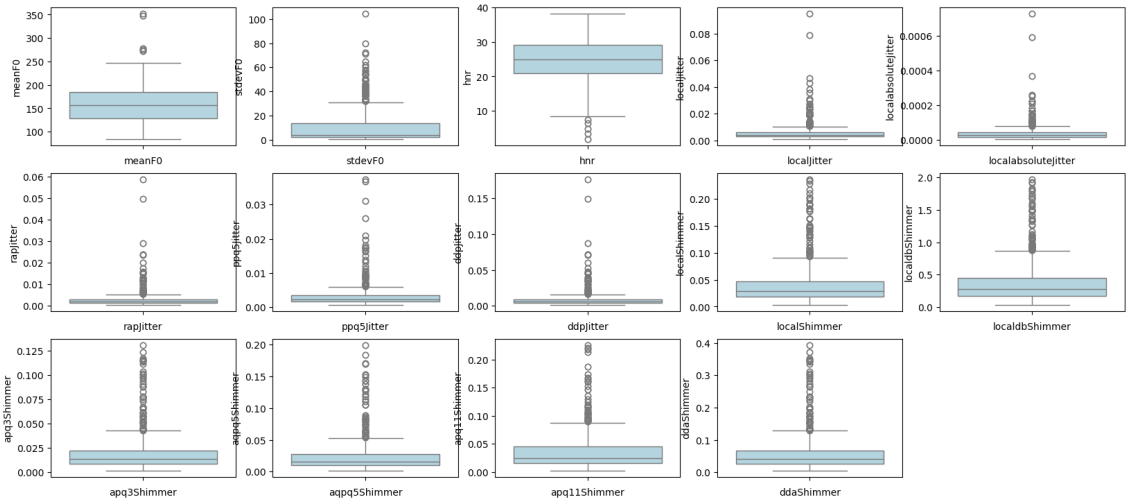

Figure 4.2: Boxplots for the features extracted from Italian Parkinson's Voice and Speech dataset before outlier reduction

From the figure 4.2, outliers appear to be present from the above box plot in most of the columns.

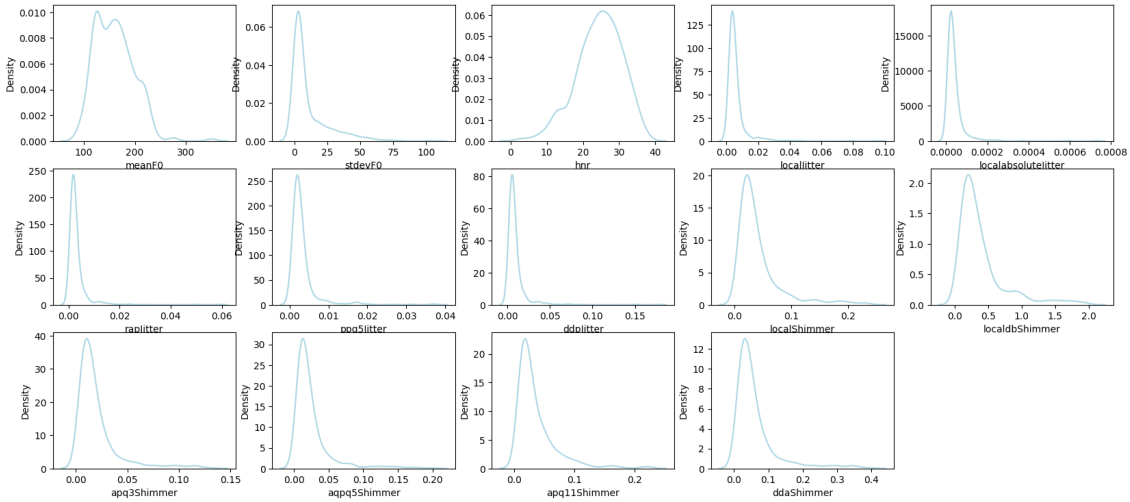

Figure 4.3: KDE plots for the features extracted from Italian Parkinson's Voice and Speech dataset before outlier reduction

Analyzing the KDE plots before outlier reduction in 4.3, there seems to be asymmetry in the distribution of the variables: the data appears to be skewed.

After IQR reduction, the following box and KDE plots in figures 4.4 and 4.5 were obtained.

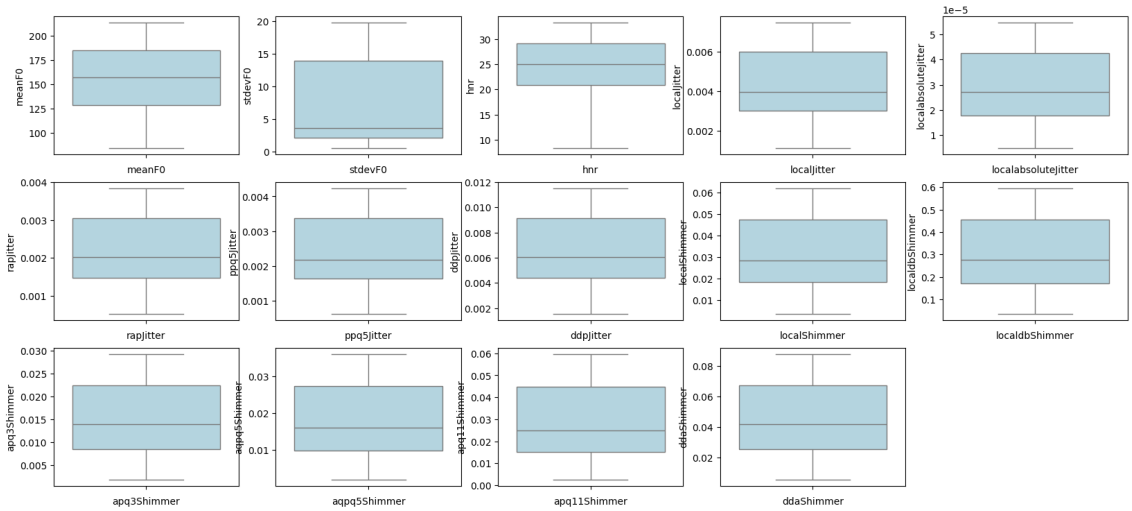

Figure 4.4: Boxplots for the features extracted from Italian Parkinson's Voice and Speech dataset after outlier reduction

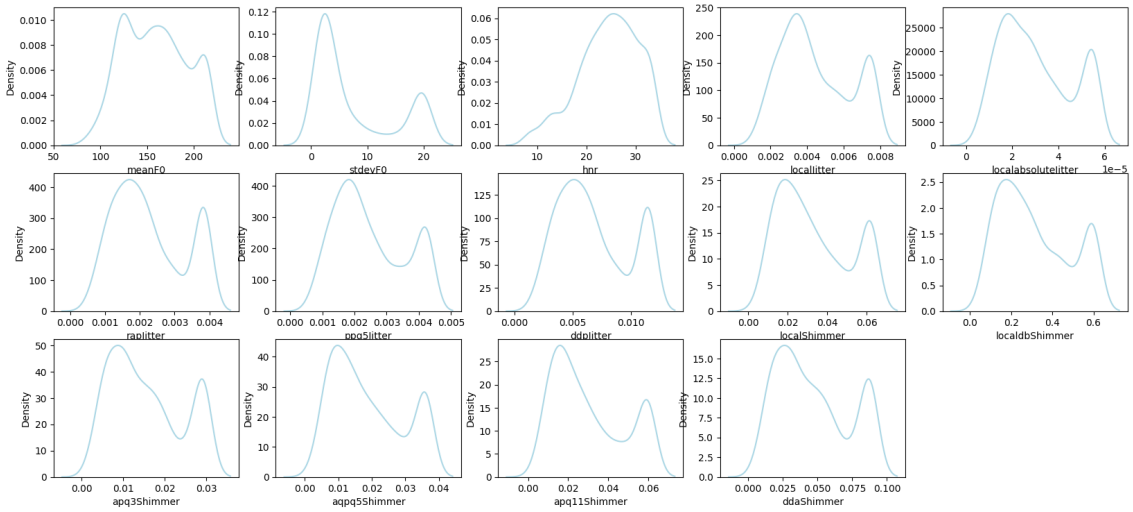

Figure 4.5: KDE plots for the features extracted from Italian Parkinson's Voice and Speech dataset after outlier reduction

The outliers are no longer present in the box plots 4.4, and there is a significant decrease in the skewness of most features 4.5. Skewness was calculated before and after the outlier reduction step to confirm the observation further.

The study in [50] suggests a reference of skewness normality as an absolute value < 2.

Table 4.4: Skewness Before and After Outlier Reduction

Feature	**Skewness Before**	**Skewness After**
meanF0	0.770975	0.100716
stdevF0	2.312534	0.888818
hnr	-0.509380	-0.537554
localJitter	6.625774	0.372810
localabsoluteJitter	7.584736	0.388884
rapJitter	7.374782	0.404535
ppq5Jitter	5.031927	0.425352
ddpJitter	7.374782	0.404535
localShimmer	2.498569	0.455654
localdbShimmer	2.207649	0.454174
apq3Shimmer	2.745459	0.408995
aqpq5Shimmer	3.195733	0.469003
apq11Shimmer	2.570736	0.549992
ddaShimmer	2.745459	0.408995

The features *stdevF0, localJitter, localabsoluteJitter, rapJitter, ppq5Jitter, ddpJitter, localShimmer, localdbShimmer, apq3Shimmer, aqpq5Shimmer, apq11Shimmer and ddaShimmer* appear to have high skewness values in the beginning, which is no longer the case after outlier reduction. Understandably, outliers were responsible for the skewness of the data. As such, it is possible to avoid further skewness reduction of the data.

In order to explore the relationships between different attributes, a pair plot for all the combinations of features was constructed.

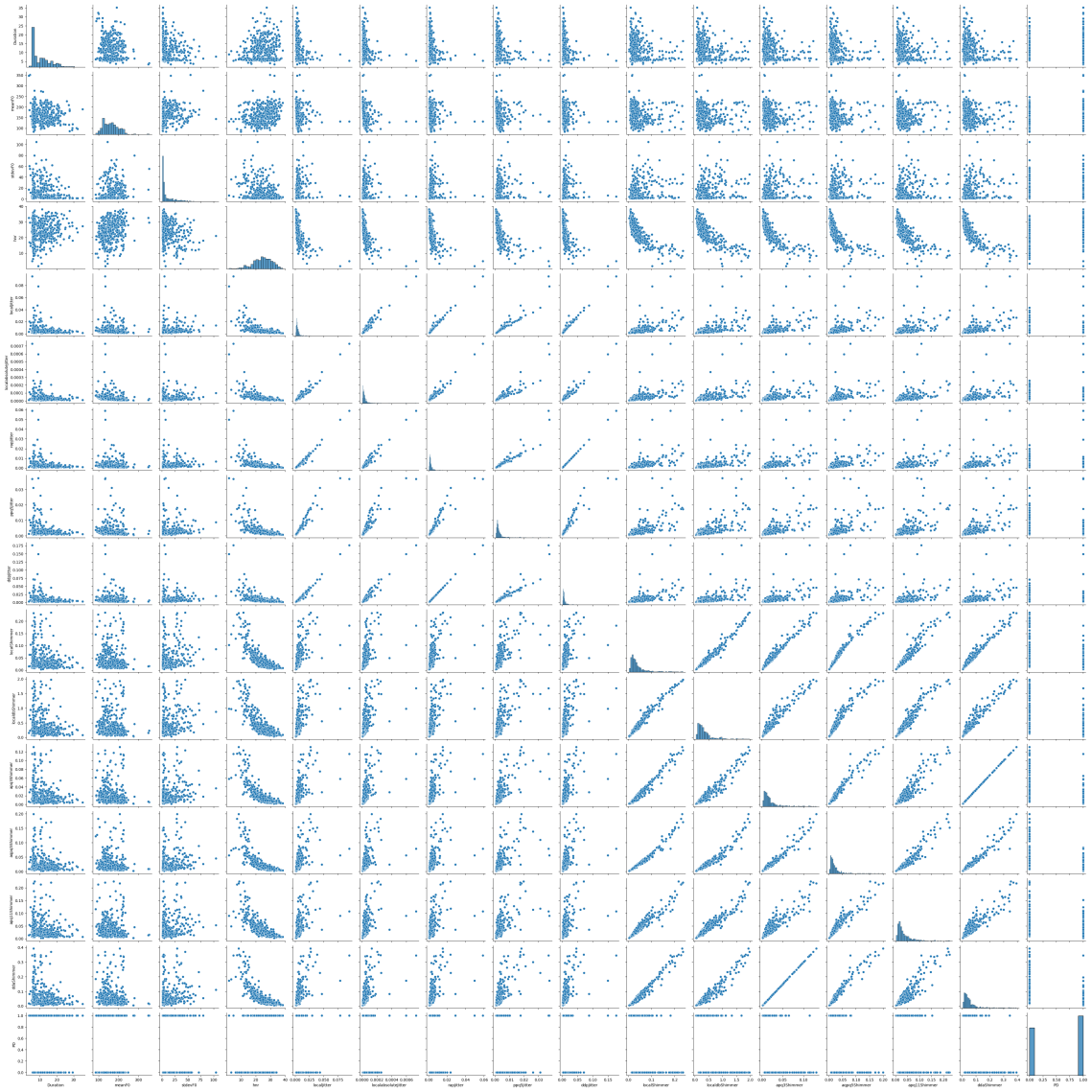

Figure 4.6: Pair plots for the features extracted from Italian Parkinson's Voice and Speech dataset

Due to the high count of attributes, the visibility in the pair plot is limited, making it challenging to identify correlations visually. Consequently, a decision was made to perform correlation analysis to assess the relationships between variables quantitatively. Using the *corr()* function from the *Pandas* library, a correlation matrix for every pair of features was plotted. Figure 4.7 shows the pairs of features with high correlations (absolute value over 0.80, excluding the diagonal values corresponding to 1).

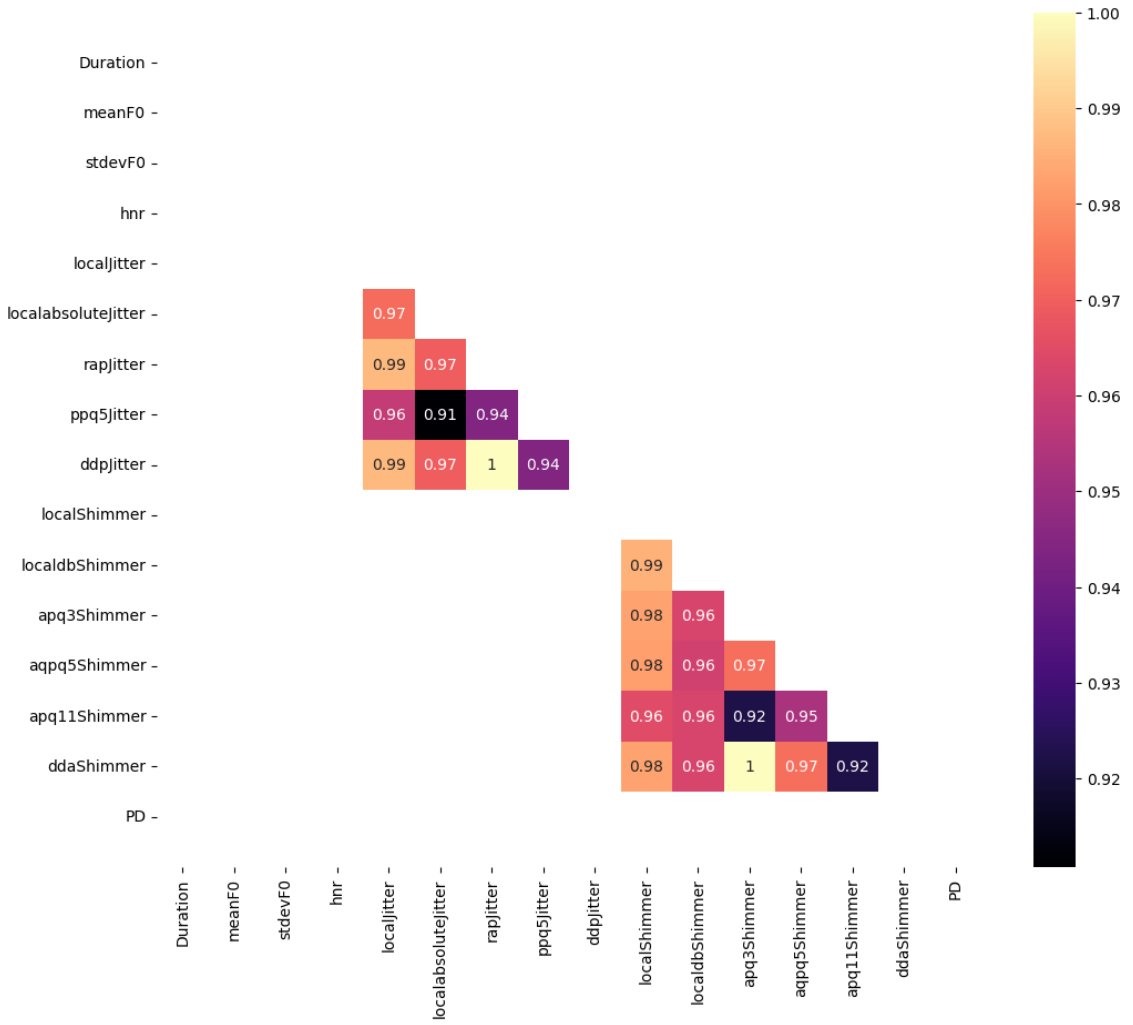

Figure 4.7: Correlation of features extracted from Italian Parkinson's Voice and Speech dataset

As expected, the features related to Shimmer and Jitter have high correlations between themselves.

4.2.3 Modeling

After the treatment explained previously, the data was used to train the following set of common machine-learning algorithms:

- **Logistic Regression:** One of the most frequently used model-based methods. It uses a logistic function and describes possible outcomes of single trials by measuring the relationship between the independent and outcome values [51, 52].

- **K-Nearest Neighbor:** The most well-known supervised learning algorithm in pattern classification. For a defined k value, it classifies a data point based on the majority class among its k closest neighbors in a feature space. The choice of the k value is crucial as it is highly

data-dependent. A more considerable value means fewer distinct classification boundaries but can suppress noise [51, 53].

- **Naïve Bayes:** Probabilistic classification algorithm based on Bayes' theorem [51, 53, 45]
- **Support Vector Machines:** Binary classification algorithm that separates classes by defining a maximum margin and finding a hyperplane in the high-dimensional space [51, 53, 45].
- **Random Forest:** Ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes for classification tasks. [51, 54, 45]
- **Bagging:** Ensemble learning method that aims to improve the stability and accuracy of ML algorithms, especially for high-variance models. In classification, a Bagging Classifier trains multiple instances of a base classifier on different subsets of the training data [51, 40, 14].
- **AdaBoost:** Ensemble technique that combines weak learners sequentially, giving more weight to misclassified instances in each iteration to improve overall performance [55].
- **XGBoost:** Implementation of gradient-boosted decision trees designed for speed and performance [1].
- **Neural Network:** Computational model inspired by the structure and function of the human brain, comprising interconnected nodes (neurons) organized in layers [51, 45, 32].

To combine the multiple predictions that resulted from the different algorithms, a Soft Voting Classifier was implemented. The probabilities generated by each model are averaged, and the class with the highest probability is given as the final result.

Based on the study in [29], another method that generates a spectrogram image for every audio file and utilizes a pre-trained convolutional neural network (Inception V3) with transfer learning to make predictions based on the pictures was tested. However, the method was deemed too computationally costly and inviable for further development.

In the pursuit of enhancing the predictive performance of these classification models, a critical aspect of the methodology involved the exploration of hyperparameter tuning across all selected algorithms. Recognizing hyperparameters' pivotal role in fine-tuning and optimizing machine learning models, the code systematically adjusted these parameters for each algorithm employed in the classification framework. A 5-fold cross-validation grid search enhanced this iterative tuning process, allowing the identification of the most effective configurations and models to capture intricate patterns within the data better and improve overall predictive accuracy. The following table details the specific hyperparameters and values tested for each algorithm.

Table 4.5: Hyperparameter tuning

Algorithm	Values	Description
Logistic Regression	• solvers = ['newton-cg', 'lbfgs', 'liblinear'] • penalty = ['l2'] • c_values = [100, 10, 1.0, 0.1, 0.01]	• Solver optimization algorithms • Penalty • Regularization parameter
KNN	• n_neighbors = [1...19] • weights = ['uniform', 'distance'] • p = [1, 2]	• Number of neighbors • Weight assigned • Distance metric
Naive Bayes	No hyperparameters	
SVM	• C = [0.1, 1, 10, 100, 1000] • gamma = [1, 0.1, 0.01, 0.001, 0.0001] • kernel = ['rbf']	• Regularization parameter • Kernel coefficient • Type of kernel
Random Forest	• n_estimators = [50, 100, 150] • max_features = [3, 5, 'sqrt', 'log2'] • max_depth = [None, 10, 20, 30]	• Number of trees, • Number of features, • Max depth of trees
Bagging	• n_estimators = [10, 20, 30, 40, 50]	• Number of estimators
Adaboost	• n_estimators = [10, 20, 30, 40, 50]	• Number of estimators
Xgboost	• learning_rate = [0.01, 0.1, 0.2] • n_estimators = [50, 100, 200] • max_depth = [3, 4, 5]	• Learning rate • Number of estimators • Max depth of trees
Neural Network	• hidden_layer_sizes = [(50,), (100,), (50, 50)] • activation = ['relu', 'tanh'] • solver = ['adam', 'sgd'] • learning_rate_init = [0.001, 0.01, 0.1]	• Size of hidden layers • Activation function • Solver algorithm • Initial learning rate

The code automates searching for the best hyperparameters using a grid search. The *GridSearchCV* method from *Scikit-learn* library was utilized.

4.2.4 Evaluation

In the evaluation phase, several steps were undertaken to guarantee the model's performance and generalization capabilities.

First, the dataset was partitioned into training and testing sets using an 80-20 percentage split ratio. This allowed the model to be trained on a substantial portion of the data while the remaining portion was reserved for evaluating the model's performance on unseen data. Additionally, to enhance the robustness of the evaluation process, *k-fold* cross-validation was employed with five splits. Here, the dataset is divided into five subsets, and the model is trained and tested five times, each using a different subset for testing and the remaining subsets for training. This process helps mitigate the impact of data variability and ensures a complete estimation of the model's performance across different subsets of the data [30].

To gain a deeper insight into the model's sanity, tables comparing each predicted probability and actual value were generated for each algorithm. The data in the tables was plotted to visually represent how well the model's probability predictions align with the actual values. Potential errors were identified in the prediction process by examining the tables and plots.

Figure 4.8 displays the generated tables for the final iteration of the prediction model using interpretable features generated from the Italian dataset.

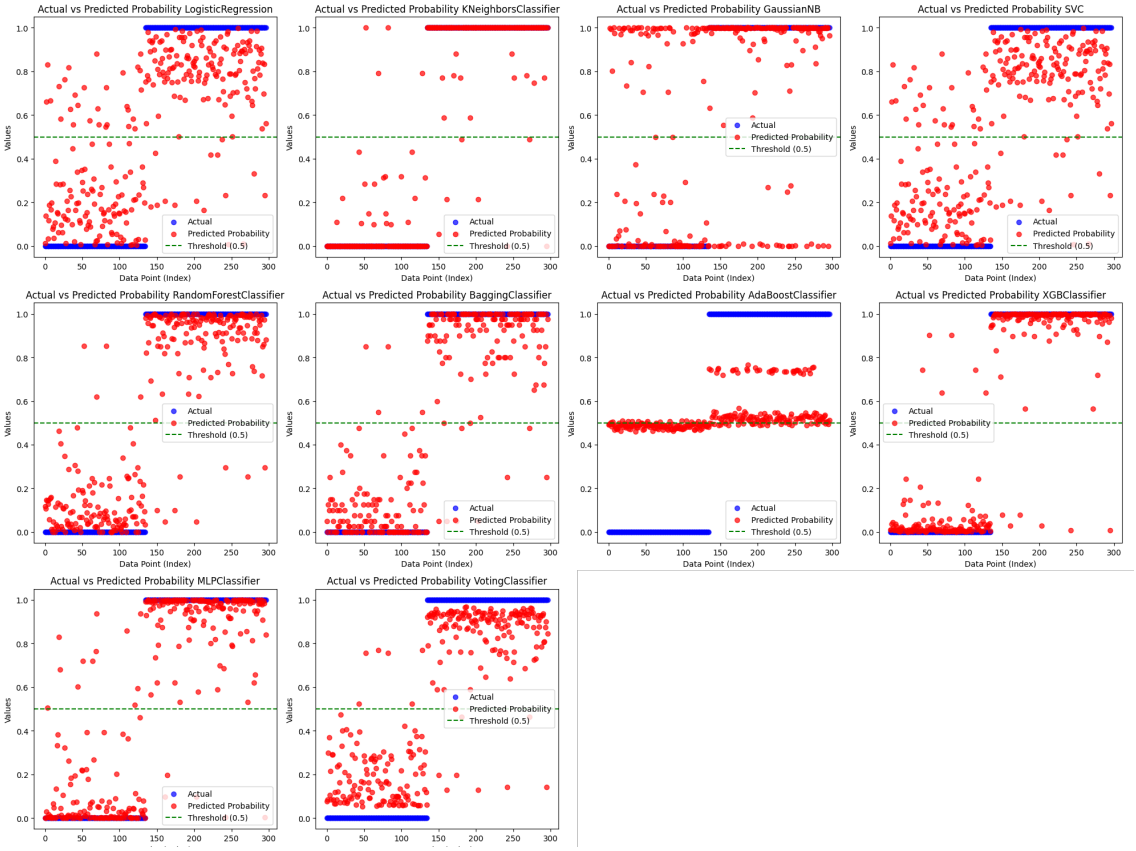

Figure 4.8: Comparison of predictions and actual values using traditional features from the Italian dataset

For the evaluation of the study's models, various metrics were employed. The confusion matrix was the first metric used since it is designed to evaluate binary classification. It allows the comparison of four essential components [32, 56]:

- **True Positive (TP):** Instances where the model correctly predicts the positive class.
- **True Negative (TN)**: Instances where the model correctly predicts the negative class.
- **False Positive (FP):** Instances where the model incorrectly predicts the positive class (Type I error).
- **False Negative (FN):** Instances where the model incorrectly predicts the negative class (Type II error).

For a better visualization of the table, consider the example in 4.6:

Table 4.6: Confusion Matrix example

Predicted	**Actual**	
	Positive	**Negative**
Positive	True Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	True Negative (TN)

Accompanying the confusion matrix, a table with calculations for the following metrics was calculated after each algorithm. [40, 12, 9]

- **Accuracy:** Accuracy represents the ratio of correctly predicted instances to the total cases in the dataset. It provides a general measure of the model's correctness.
- **Precision:** Precision measures the proportion of actual positive results out of all positive instances.
- **Recall:** Recall measures the proportion of true positives from all the positive existing results (true positives + false negatives).
- **F1-Score:** The F1-Score combines precision and recall into a single metric (using the harmonic mean between the two values). It represents the model's accuracy.

  Collectively, these metrics provide valuable insights into the model's performance, aiding its refinement and optimization.

Considering the previously defined metrics, the table 4.7 presents the formula for each one.

Table 4.7: Classification metrics and formulas

Metric	**Formula**
Accuracy	$\frac{TP+TN}{TP+FP+TN+FN}$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
F1-score	$2 * \frac{recall * precision}{recall + precision}$

In the scope of this problem, achieving a balance between precision and recall is crucial. While high precision ensures accurate identification of positive cases, high recall prevents missing individuals with the disease. The F1-Score combines precision and recall and provides a solid evaluation metric.

The balanced class distribution in all the used datasets contributes to a more stable evaluation environment. Since the datasets maintain a close-to-50% distribution between positive and negative classes, it can be expected that the values of the metrics do not have significant discrepancies.

In addition to traditional performance metrics, an essential aspect of the evaluation encompasses the computational efficiency of the models. Recognizing that algorithms of varying complexities were employed, it becomes pertinent to consider the time each model takes to train and run.

Chapter 5

Tests and Results

5.1 Tests

This segment marks a pivotal phase in the development of the classification model. After setting up a collection of valuable experiments for data wrangling, modeling, and evaluation, an initial approach was defined, serving as a robust foundation for the classification problem.

Each iteration within this testing phase was driven by a systematic trial and error process, where the results of each test shaped the trajectory of the following experiments. The methodology adopted was dynamic, with each new test building upon the outcomes and lessons learned from its predecessor. Through this iterative process, the classification model was refined, and it was possible to define a pipeline that could be applicable to different datasets.

5.1.1 Initial approach

In the preliminary phase of the experimentation, the base approach was defined. For this step, the data used came from the *Italian Parkinson's Voice and Speech* dataset and the *MDVR-KCL* dataset, which were the most complete datasets and provided two types of vocal exercises (sustained vowel and text reading).

For the data treatment, *null* and duplicate values were removed, type consistency was verified, and normalization was applied. The implemented machine learning algorithms were Logistic Regression, K-Nearest Neighbor, Naïve Bayes, Support Vector Machines, Random Forest, Bagging, AdaBoost, and XGBoost. All the algorithms were implemented with hyperparameter tuning and five *k-fold* cross-validation. In regards to evaluation, all of the techniques described were used.

5.1.2 Phase 2

From the correlation analysis, it became noticeable that some independent attributes had high correlations. Even though some ML algorithms are robust against this, it was worthwhile to try to

reduce it as much as possible and compare the results.

During this second phase, Multicollinearity was tackled using VIF. The VIF value was computed for each feature, and the one displaying the highest VIF value was iteratively dropped from the table. This was repeated until all variables showed a low enough VIF score. The absolute VIF <10 was considered as the threshold value in this case.

After this process, the set of features obtained differed for the two datasets

- **Italian dataset:** *stdevF0, hnr, localJitter, ddpJitter, apq11Shimmer, PD*.
- **MDVR-KCL dataset:** *stdevF0, hnr, rapJitter, apq3Shimmer, PD*

5.1.3 Phase 3

In Phase 3 of the iterative model development, based on the results, it was decided to abandon the multicollinearity reduction step introduced in the previous phase, opting to explore alternatives. A key adjustment in this iteration involved implementing *MinMax* scaling to the features. By incorporating scaling, the aim was to standardize the range of values across all features, ensuring a uniform impact during the model training process.

5.1.4 Phase 4

For the fourth iteration, the two datasets were combined to analyze the potential advantages derived from their mixture. This iteration evaluates the potential advantages of leveraging the joint information in both datasets. By integrating the datasets, the objective was to assess the model's performance when trained on the combined dataset and compare it to the performance achieved using each dataset individually. This iteration provides insights into how model performance may vary when combining data from two types of vocal exercises: sustained vowel phonation and text reading.

5.1.5 Phase 5

PCA was incorporated as a technique to consolidate the *Shimmer* and *Jitter* variables. Recognizing the interdependencies between these variables, PCA combined them into more concise and informative representations. The Jitter variables, including *localJitter*, *localabsoluteJitter*, *rapJitter*, *ppq5Jitter*, and *ddpJitter*, were integrated into a singular *JitterPCA* component. Simultaneously, the Shimmer variables, *localdbShimmer*, *apq3Shimmer*, *aqpq5Shimmer*, *apq11Shimmer*, and *ddaShimmer*, were unified under the composite label *ShimmerPCA*. This process allowed to condense the original features into meaningful but shorter components, simplifying the complexity of the used features while preserving essential information.

5.1.6 Phase 6

During this step, the PCA used in the previous phase was removed. The IQR outlier reduction method was incorporated to address potential outliers. This refinement aimed to guarantee a more robust and reliable model by reducing the impact of extreme values within the dataset.

Furthermore, the soft voting algorithm was implemented. Averaging all the probability predictions made by the base classifiers allowed for a more direct decision-making process and united all the predictions into one.

Additionally, two other datasets were introduced to evaluate further the model's performance: *Synthetic Vowels of Speakers with Parkinson's Disease and Parkinsonism* [28] dataset and *Voice Samples for Patients with Parkinson's Disease and Healthy Controls* [29] dataset.

5.1.7 Phase 7

A notable shift in the feature extraction and data preprocessing approach was introduced in Phase 7, marking a departure from the previous iterations. The conventional feature extraction and data wrangling steps were replaced with a more advanced approach involving the pre-trained DNNs for feature extraction, specifically leveraging three distinct representations: TRILLsson, Wav2Vec 2.0, and HUBERT.

Unlike the previous feature extraction method that produced interpretable acoustic features, the new embeddings encapsulate complex patterns and relationships within the audio data. The transition from interpretable features to embeddings reflects a strategic choice to enable the model to capture intricate nuances and variations in the audio files. By doing so, the model gains the potential to uncover hidden patterns that contribute to a more refined and accurate classification of PD.

The application of *StandardScale* after feature extraction standardizes the feature values, ensuring that they are on a comparable scale before being fed into the machine learning model. This standardization prevents features with different magnitudes from dominating the training process.

5.2 Results

5.2.1 Predictive metrics and observations

- **Initial phase:**

Table 5.1: Initial phase performance metrics for the Italian Parkinson's voice and speech dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.81	0.82	0.81	0.81
KNN	0.72	0.72	0.72	0.72
Naive Bayes	0.78	0.78	0.78	0.78
SVM	0.76	0.76	0.76	0.76
Random Forest	0.86	0.86	0.86	0.86
Bagging	0.85	0.85	0.85	0.85
AdaBoost	0.86	0.86	0.86	0.86
XGBoost	**0.87**	**0.87**	**0.87**	**0.87**

Table 5.2: Initial phase performance metrics for the MDVR-KCL dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.76	0.76	0.76	0.76
KNN	0.75	0.75	0.75	0.75
Naive Bayes	0.70	0.71	0.70	0.70
SVM	0.80	0.80	0.80	0.80
Random Forest	**0.81**	**0.81**	**0.81**	**0.81**
Bagging	0.76	0.76	0.76	0.76
AdaBoost	0.78	0.78	0.78	0.78
XGBoost	0.80	0.80	0.80	0.80

In the initial phase of this experimentation, the focus was on establishing a baseline for the approach. Several implemented machine learning algorithms demonstrated notable performance, especially considering it was an initial iteration.

When applied to the Italian dataset, the algorithms' accuracy had a mean value of 0.83, and the top-performing algorithms were XGBoost, AdaBoost, and Random Forest, with accuracies of 0.87, 0.86, and 0.86.

The MDVR-KCL dataset obtained a mean accuracy value of 0.77; the best algorithms were Random Forest, XGBoost, and SVM with 0.81, 0.80, and 0.80 accuracies.

- **Phase 2:**

Table 5.3: Phase 2 performance metrics for the Italian Parkinson's voice and speech dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	**0.86**	**0.87**	**0.86**	**0.86**
KNN	0.82	0.83	0.82	0.82
Naive Bayes	0.81	0.82	0.81	0.81
SVM	**0.86**	**0.87**	**0.86**	**0.86**
Random Forest	0.80	0.81	0.80	0.80
Bagging	0.81	0.81	0.81	0.81
AdaBoost	0.79	0.79	0.79	0.79
XGBoost	0.78	0.78	0.78	0.78

Table 5.4: Phase 2 performance metrics for the MDVR-KCL dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.65	0.64	0.65	0.64
KNN	0.63	0.62	0.63	0.62
Naive Bayes	0.66	0.66	0.66	0.65
SVM	**0.67**	**0.66**	**0.67**	**0.66**
Random Forest	0.64	0.64	0.64	0.64
Bagging	0.63	0.63	0.63	0.63
AdaBoost	0.63	0.62	0.63	0.62
XGBoost	0.65	0.64	0.65	0.64

When considering the changes made in the second iteration, the mean accuracies obtained for the two datasets were recalculated. There was a marginal decrease from 0.83 to 0.81 for the Italian dataset. For the MDVR-KCL dataset, there is a more substantial decline from 0.77 to 0.65.

Notably, with 0.86 accuracies, Logistic Regression and SVM were the top-performing algorithms for the Italian dataset, and SVM was the best for the MDVR-KCL dataset with 0.67 accuracy. On the other hand, ensemble methods, including Random Forest, Bagging, AdaBoost, and XGBoost, experienced a discernible decline in performance, particularly in terms of accuracy, compared to the initial phase.

After phase 2, the results showcase resistance to feature variations when applied to the Italian dataset, maintaining a relatively stable accuracy. However, a contrasting pattern emerges when examining the results from the MDVR-KCL dataset, where all algorithms display a significant decrease in accuracy.

From the results obtained in this iteration, it is possible to extract intriguing observations regarding the resilience of algorithms to variations in the feature set, with notable distinctions between the Italian and MDVR-KCL datasets.

- **Phase 3:**

Table 5.5: Phase 3 performance metrics for the Italian Parkinson's voice and speech dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.86	0.87	0.86	0.86
KNN	**0.88**	**0.88**	**0.88**	**0.88**
Naive Bayes	0.78	0.78	0.78	0.78
SVM	0.84	0.85	0.84	0.84
Random Forest	0.86	0.86	0.86	0.86
Bagging	0.86	0.86	0.86	0.86
AdaBoost	0.86	0.86	0.86	0.86
XGBoost	0.87	0.87	0.87	0.87

Table 5.6: Phase 3 performance metrics for the MDVR-KCL dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.76	0.76	0.76	0.76
KNN	0.75	0.75	0.75	0.75
Naive Bayes	0.70	0.70	0.70	0.70
SVM	0.76	0.76	0.76	0.76
Random Forest	**0.80**	**0.80**	**0.80**	**0.80**
Bagging	0.76	0.76	0.76	0.76
AdaBoost	0.79	0.78	0.79	0.78
XGBoost	**0.80**	**0.80**	**0.80**	**0.80**

For this iteration, improved scores across various metrics were observed in many cases. Results showed mean accuracies of 0.86 and 0.76 for Italian and MDVR-KCL datasets. Specifically, algorithms that exhibited strong performance in the Initial Phase, such as XGBoost, Random Forest, Bagging, and AdaBoost, maintained or slightly improved their accuracy, precision, recall, and F1-Score metrics by incorporating *MinMax* scaling. For the Italian dataset, KNN achieved the highest accuracy of 0.88, closely followed by XGBoost and several other algorithms with values of 0.87, including AdaBoost, Bagging, Random Forest, and Logistic Regression. For the MDVR-KCL dataset, XGBoost and Random Forest emerged with the highest accuracy of 0.80.

- **Phase 4:**

Table 5.7: Phase 4 performance metrics for the Italian Parkinson's voice and speech and MDVR-KCL datasets combined

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.70	0.70	0.70	0.70
KNN	0.73	0.73	0.73	0.72
Naive Bayes	0.63	0.63	0.63	0.62
SVM	0.78	0.78	0.78	0.78
Random Forest	**0.80**	**0.80**	**0.80**	**0.79**
Bagging	0.78	0.78	0.78	0.78
AdaBoost	0.76	0.76	0.76	0.76
XGBoost	0.79	0.79	0.79	0.79

The combination of data from the Italian Parkinson's Voice and Speech dataset and the MDVR-KCL dataset resulted in a model with performance falling between that of the individual datasets. With a mean accuracy of 0.77, the combined results were inferior to those obtained from the Italian dataset, but they surpassed the performance achieved with the MDVR-KCL dataset. The top-performing algorithm was Random Forest, which obtained an accuracy of 0.80.

- **Phase 5:**

Table 5.8: Phase 5 performance metrics for the Italian Parkinson's voice and Speech dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.86	0.86	0.86	0.86
KNN	**0.88**	**0.88**	**0.88**	**0.88**
Naive Bayes	0.83	0.84	0.83	0.83
SVM	0.85	0.85	0.85	0.85
Random Forest	0.84	0.85	0.84	0.84
Bagging	0.82	0.83	0.82	0.82
AdaBoost	0.80	0.80	0.80	0.80
XGBoost	0.86	0.86	0.86	0.86

Table 5.9: Phase 5 performance metrics for the MDVR-KCL dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.75	0.75	0.75	0.75
KNN	0.79	0.78	0.79	0.78
Naive Bayes	0.79	0.79	0.79	0.78
SVM	0.78	0.78	0.78	0.78
Random Forest	0.78	0.78	0.78	0.78
Bagging	0.76	0.76	0.76	0.76
AdaBoost	0.78	0.78	0.78	0.78
XGBoost	**0.79**	**0.79**	**0.79**	**0.79**

In the current phase, there was a return to observing both Italian and MDVR-KCL datasets separately. Comparing the results obtained to Phase 3 makes sense since the only variable between the two iterations is using PCA in the current iteration.

In both datasets, some algorithms demonstrated consistent or slight improvements compared to Phase 3. The mean accuracies obtained were 0.85 and 0.78 for Italian and MDVR-KCL datasets, which are on par with the previous results.

Upon closer examination of the metrics from the best-performing algorithms in the current iteration and Phase 3, it is observable that PCA introduced a slight trade-off in the performance of certain algorithms. The top-performing algorithms in the current iteration displayed similar results compared to the top-performing algorithms of Phase 3.

- **Phase 6:**

Table 5.10: Phase 6 performance metrics for the Italian Parkinson's Voice and Speech dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.87	0.87	0.87	0.87
KNN	0.91	0.91	0.91	0.91
Naive Bayes	0.77	0.77	0.77	0.77
SVM	**0.95**	**0.95**	**0.95**	**0.95**
Random Forest	0.89	0.89	0.89	0.89
Bagging	0.88	0.88	0.88	0.88
AdaBoost	0.85	0.85	0.85	0.85
XGBoost	0.92	0.92	0.92	0.92
Neural Network	0.80	0.84	0.80	0.79
Voting Classifier	0.90	0.90	0.90	0.90

Table 5.11: Phase 6 performance metrics for the MDVR-KCL dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.75	0.75	0.75	0.75
KNN	0.77	0.77	0.77	0.77
Naive Bayes	0.65	0.65	0.65	0.65
SVM	0.81	0.81	0.81	0.81
Random Forest	0.83	0.83	0.83	0.83
Bagging	**0.84**	**0.85**	**0.84**	**0.84**
AdaBoost	0.80	0.80	0.80	0.80
XGBoost	0.83	0.83	0.83	0.83
Neural Network	0.78	0.79	0.78	0.78
Voting Classifier - soft	0.82	0.82	0.82	0.81

Table 5.12: Phase 6 performance metrics for the Synthetic vowels of speakers with Parkinson's disease and Parkinsonism dataset

Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-score**
Logistic Regression	0.67	0.68	0.67	0.66
KNN	0.72	0.73	0.72	0.72
NB	0.61	0.61	0.61	0.61
SVM	0.61	0.61	0.61	0.61
Random Forest	0.72	0.73	0.72	0.72
Bagging	0.72	0.73	0.72	0.72
AdaBoost	0.61	0.61	0.61	0.61
XGBoost	**0.78**	**0.78**	**0.78**	**0.78**
Neural Network	0.67	0.68	0.67	0.66
Voting Classifier - soft	0.72	0.73	0.72	0.72

Table 5.13: Phase 6 performance metrics for the Voice Samples for Patients with Parkinson's Disease and Healthy Controls dataset

Algorithm	**Accuracy**	**Precision**	**Recall**	**F1-score**
Logistic Regression	0.50	0.55	0.50	0.52
KNN	0.69	0.71	0.69	0.69
NB	0.25	0.30	0.25	0.20
SVM	0.62	0.67	0.62	0.64
Random Forest	0.56	0.54	0.56	0.55
Bagging	0.69	0.71	0.69	0.69
AdaBoost	0.56	0.59	0.56	0.57
XGBoost	**0.69**	**0.76**	**0.69**	**0.70**
Neural Network	0.62	0.62	0.62	0.62
Voting Classifier - soft	0.62	0.62	0.62	0.62

Phase 6 discards the changes made in Phases 4 and 5. Once again, it makes sense to compare the results of Phase 6 with those from Phase 3, which served as the foundational iteration for the current phase.

The accuracies of several algorithms, such as KNN, SVM, Random Forest, Bagging, AdaBoost, and XGBoost, show notable improvements in Phase 6. Particularly noteworthy is the remarkable accuracy achieved by SVM, reaching 0.95 accuracy in the Italian dataset, and Bagging, reaching 0.84 in the MDVR-KCL datasets.

The added Neural Network algorithm reached values of 0.80 accuracy for the Italian dataset and 0.78 for the MDVR-KCL dataset. The soft voting algorithm reaches slightly lower results, although not far from the top-performing algorithms: 0.80 and 0.78 for Italian and MDVR-KCL datasets.

Regarding the other datasets, the results fell somewhat short compared to the previous two. Notably, XGBoost achieved the highest score for both datasets, with an accuracy of 0.78 for the Czech dataset and 0.69 for the VSP-PDHC dataset.

One noticeable difference is the relatively low performance of the Naive Bayes algorithm in all datasets compared to previous iterations and other algorithms in the same iteration. There is a drop in accuracy for the Italian dataset from 0.83 to 0.73 and for the MDVR-KCL dataset from 0.79 to 0.66.

- **Phase 7:**

Wav2Vec

Table 5.14: Phase 7 performance metrics using Wav2Vec for the Italian dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.85	0.85	0.85	0.85
KNN	0.82	0.84	0.82	0.82
NB	0.76	0.78	0.76	0.76
SVM	0.90	0.90	0.90	0.90
Random Forest	0.84	0.84	0.84	0.84
Bagging	0.81	0.81	0.81	0.81
AdaBoost	0.87	0.87	0.87	0.87
XGBoost	0.81	0.81	0.81	0.81
Neural Network	**0.91**	**0.91**	**0.91**	**0.91**
Voting Classifier (Soft)	0.90	0.90	0.90	0.90

Table 5.15: Phase 7 performance metrics using Wav2Vec for the MDVR-KCL dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.80	0.81	0.80	0.80
KNN	0.72	0.72	0.72	0.72
NB	0.63	0.64	0.63	0.64
SVM	0.81	0.81	0.81	0.81
Random Forest	0.74	0.75	0.74	0.73
Bagging	0.73	0.73	0.73	0.72
AdaBoost	0.71	0.71	0.71	0.71
XGBoost	0.73	0.73	0.73	0.73
Neural Network	**0.82**	**0.82**	**0.82**	**0.82**
Voting Classifier (Soft)	0.82	0.82	0.82	0.81

Table 5.16: Phase 7 performance metrics using Wav2Vec for the Czech dataset

Algorithms	Acc	Precision	Recall	F1-Score
Logistic Regression	0.42	0.41	0.42	0.41
KNN	0.63	0.64	0.63	0.58
NB	**0.68**	**0.68**	**0.68**	**0.67**
SVM	0.53	0.50	0.53	0.50
Random Forest	0.63	0.64	0.63	0.63
Bagging	0.58	0.56	0.58	0.54
AdaBoost	0.58	0.60	0.58	0.58
XGBoost	0.58	0.58	0.58	0.58
Neural Network	0.53	0.50	0.53	0.50
Voting Classifier (Soft)	0.53	0.50	0.53	0.50

Table 5.17: Phase 7 performance metrics using Wav2Vec for the VSP-PDHC dataset

Algorithms	Acc	Precision	Recall	F1-Score
Logistic Regression	0.53	0.56	0.53	0.54
KNN	0.59	0.57	0.59	0.58
NB	**0.65**	**0.65**	**0.65**	**0.65**
SVM	0.35	0.12	0.35	0.18
Random Forest	0.59	0.7	0.59	0.59
Bagging	0.59	0.6	0.59	0.59
AdaBoost	0.59	0.64	0.59	0.6
XGBoost	0.41	0.51	0.41	0.4
Neural Network	**0.65**	**0.65**	**0.65**	**0.65**
Voting Classifier (Soft)	**0.65**	**0.65**	**0.65**	**0.65**

Trillsson

Table 5.18: Phase 7 performance metrics using Trillsson for the Italian dataset

Algorithms	Acc	Precision	Recall	F1-Score
Logistic Regression	0.98	0.98	0.98	0.98
KNN	0.80	0.80	0.80	0.80
NB	0.68	0.69	0.68	0.68
SVM	**0.99**	**0.99**	**0.99**	**0.99**
Random Forest	0.88	0.88	0.88	0.88
Bagging	0.90	0.90	0.90	0.90
AdaBoost	0.87	0.88	0.87	0.87
XGBoost	0.91	0.91	0.91	0.91
Neural Network	0.96	0.96	0.96	0.96
Voting Classifier (Soft)	0.96	0.96	0.96	0.96

Table 5.19: Phase 7 performance metrics using Trillsson for the MDVR-KCL dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.90	0.90	0.90	0.90
KNN	0.83	0.84	0.83	0.83
NB	0.67	0.68	0.67	0.65
SVM	0.87	0.88	0.87	0.87
Random Forest	0.85	0.87	0.85	0.85
Bagging	0.84	0.85	0.84	0.83
AdaBoost	0.85	0.86	0.85	0.85
XGBoost	**0.90**	**0.91**	**0.90**	**0.90**
Neural Network	0.88	0.88	0.88	0.88
Voting Classifier (Soft)	0.90	0.91	0.90	0.89

Table 5.20: Phase 7 performance metrics using Trillsson for the Czech dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.79	0.81	0.79	0.79
KNN	**0.79**	**0.86**	**0.79**	**0.79**
NB	0.58	0.58	0.58	0.58
SVM	0.68	0.70	0.68	0.69
Random Forest	0.58	0.62	0.58	0.57
Bagging	0.68	0.70	0.68	0.69
AdaBoost	0.63	0.64	0.63	0.63
XGBoost	0.53	0.55	0.53	0.53
Neural Network	0.63	0.66	0.63	0.63
Voting Classifier (Soft)	0.68	0.74	0.68	0.68

Table 5.21: Phase 7 performance metrics using Trillsson for the VSP-PDHC dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.65	0.73	0.65	0.65
KNN	0.47	0.56	0.47	0.47
NB	0.65	0.65	0.65	0.65
SVM	0.59	0.64	0.59	0.60
Random Forest	0.65	0.73	0.65	0.65
Bagging	**0.76**	**0.86**	**0.76**	**0.77**
AdaBoost	**0.76**	**0.86**	**0.76**	**0.77**
XGBoost	0.59	0.81	0.59	0.57
Neural Network	0.65	0.73	0.65	0.65
Voting Classifier (Soft)	0.65	0.68	0.65	0.65

Hubert

Table 5.22: Phase 7 performance metrics using Hubert for the Italian dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.94	0.94	0.94	0.94
KNN	0.90	0.90	0.90	0.90
NB	0.77	0.77	0.77	0.77
SVM	0.96	0.96	0.96	0.96
Random Forest	0.89	0.90	0.89	0.89
Bagging	0.86	0.86	0.86	0.86
AdaBoost	0.90	0.92	0.90	0.90
XGBoost	0.91	0.92	0.91	0.91
Neural Network	**0.98**	**0.98**	**0.98**	**0.98**
Voting Classifier (Soft)	0.95	0.95	0.95	0.95

Table 5.23: Phase 7 performance metrics using Hubert for the MDVR-KCL dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.88	0.88	0.88	0.88
KNN	0.77	0.78	0.77	0.77
NB	0.63	0.64	0.63	0.63
SVM	**0.93**	**0.93**	**0.93**	**0.93**
Random Forest	0.83	0.84	0.83	0.82
Bagging	0.84	0.85	0.84	0.83
AdaBoost	0.82	0.82	0.82	0.82
XGBoost	0.88	0.88	0.88	0.88
Neural Network	0.92	0.92	0.92	0.92
Voting Classifier (Soft)	0.91	0.91	0.91	0.91

Table 5.24: Phase 7 performance metrics using Hubert for the Czech dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.63	0.64	0.63	0.63
KNN	0.63	0.64	0.63	0.63
NB	0.79	0.79	0.79	0.79
SVM	0.79	0.81	0.79	0.79
Random Forest	0.74	0.77	0.74	0.74
Bagging	0.74	0.74	0.74	0.74
AdaBoost	**0.84**	**0.89**	**0.84**	**0.84**
XGBoost	0.74	0.77	0.74	0.74
Neural Network	0.68	0.82	0.68	0.67
Voting Classifier (Soft)	0.79	0.81	0.79	0.79

Table 5.25: Phase 7 performance metrics using Hubert for the VSP-PDHC dataset

Algorithms	**Acc**	**Precision**	**Recall**	**F1-Score**
Logistic Regression	0.71	0.72	0.71	0.71
KNN	0.76	0.80	0.76	0.77
NB	0.65	0.68	0.65	0.65
SVM	0.71	0.72	0.71	0.71
Random Forest	0.65	0.73	0.65	0.65
Bagging	0.76	0.80	0.76	0.77
AdaBoost	0.76	0.80	0.76	0.77
XGBoost	**0.76**	**0.86**	**0.76**	**0.77**
Neural Network	0.76	0.80	0.76	0.77
Voting Classifier (Soft)	0.76	0.80	0.76	0.77

TRILLsson and Hubert demonstrate exceptional performance on the Italian dataset, with high accuracy, precision, recall, and F1-scores across various algorithms. The voting classifier reaches an accuracy of 0.96 for Trillsson and 0.95 for Hubert, and top-performing algorithms for both (SVM and NN) get 0.99 and 0.98, respectively. Wav2vec also performs well, albeit slightly trailing behind TRILLsson and Hubert, with its voting classifier displaying an accuracy of 0.90.

For the MDVR-KCL dataset, Hubert and TRILLsson gave higher results from all three feature extraction methods than Wav2vec. Its voting algorithms showed an accuracy of 0.91 and 0.90.

For the Czech dataset, once again, Hubert exhibits competitive performance. TRILLsson especially shows bad performance when compared to Hubert and Wav2vec. Comparing the voting algorithm for each one, there is a significant gap between accuracies. The highest value was 0.79, obtained by Hubert. Then, Trillsson had 0.68 accuracy, and Wav2Vec had 0.53.

In the case of the VSP-PDHC dataset, the soft voting classifier displayed the same accuracy for Wav2Vec and Trillsson, being 0.65. The best results were obtained by Hubert, where the soft voting classifier got 0.76 accuracy.

5.2.2 Execution times

Even though dealing with a long-term degenerative disease like PD may not necessitate real-time predictions, it can be interesting to consider the extracting and training times to understand further the practical viability of deploying these models in clinical settings. The evaluation not only focuses on predictive accuracy but extends to the efficiency of model execution as well. It ensures that there is an analysis of the balance between computational demands and predictive performance. This approach enables the informed decisions regarding the selection and deployment of models.

The computational efficiency of each algorithm across the different datasets was carefully monitored to obtain a glimpse into the computational demands of the predictive models. The training time was measured for all the algorithms using the set of traditional acoustic features and non-traditional embeddings (corresponding to Phases 6 and 7) for every dataset. Recognizing the variability in dataset sizes, the training times were normalized, presenting the results as time per 100 audio files, rounded to the second. Since the extractions consider the features of the audio, and every sample corresponds precisely to one set of features, the differences in the length of the audio samples were neglected.

The following values were measured on the final models for interpretable and non-interpretable feature extractions. They were performed on a base MacBook Air from 2019, featuring a 1.6GHz dual-core Intel Core i5 processor and 8GB of 2133MHz LPDDR3 onboard memory.

5.2.2.1 Traditional Acoustic Features

Table 5.26: Algorithm training times per 100 files using traditional acoustic features

Dataset	**LR**	**KNN**	**NB**	**SVM**	**RF**	**Bagging**	**AdaBoost**	**XGBoost**	**NN**	**Voting**
Italian	00:02	00:02	00:01	00:23	00:16	00:02	00:01	00:08	00:42	00:20
MDVR-KCL	00:02	00:01	00:01	00:44	00:07	00:01	00:01	00:07	00:44	00:37
Czech	00:07	00:09	00:01	00:11	01:12	00:03	00:08	00:20	00:52	00:15
VSP-PDHC	00:10	00:02	00:01	00:10	00:57	00:05	00:05	00:17	00:49	00:19
Mean	00:04	00:02	<00:01	00:17	00:36	00:03	00:03	00:13	00:47	00:19

Among the evaluated machine learning algorithms applied to interpretable features, NB, KNN, Bagging, Adaboost, and LR stand out for their notably low training times, with averages ranging between 0 and 4 seconds per 100 files.

On the other end of the spectrum, XGBoost and SVM exhibit comparatively higher training times, averaging around 13 and 17 seconds per 100 files, respectively.

Notably, RF and NN display the highest training times among the evaluated algorithms, with averages of 36 and 47 seconds per 100 files.

5.2.2.2 DNN Embeddings

Trillsson

Table 5.27: Algorithm training times per 100 files featuring embeddings extracted using Trillsson

Algorithms	**LR**	**KNN**	**NB**	**SVM**	**RF**	**Bagging**	**Adaboost**	**XGBoost**	**NN**	**Voting**
Italian	00:09	00:02	00:01	01:13	00:29	00:33	00:15	02:43	00:52	01:03
MDVR-KCL	00:23	00:04	00:01	01:47	00:21	00:59	00:14	01:50	00:30	02:12
Czech	00:18	00:01	00:01	00:10	01:20	00:23	00:16	01:26	01:05	00:22
VSP-PDHC	00:20	00:02	00:01	00:11	01:19	00:26	00:16	01:55	01:47	00:42
Mean	00:19	00:02	00:01	00:42	00:54	00:30	00:15	01:53	00:59	00:52

Wav2vec

Table 5.28: Algorithm training times per 100 files featuring embeddings extracted using Wav2vec

Algorithms	**LR**	**KNN**	**NB**	**SVM**	**RF**	**Bagging**	**Adaboost**	**XGBoost**	**NN**	**Voting**
Italian	00:07	00:01	00:01	00:48	00:21	00:24	00:09	01:45	00:26	00:33
MDVR-KCL	00:23	00:04	00:01	01:14	00:22	01:01	00:16	01:46	00:33	00:59
Czech	00:11	00:02	00:01	00:11	00:57	00:15	00:13	01:05	00:48	00:22
VSP-PDHC	00:22	00:04	00:01	00:09	01:02	00:19	01:05	00:10	00:53	00:22
Mean	00:17	00:03	00:01	00:30	00:39	00:21	00:14	01:25	00:40	00:28

Hubert

Table 5.29: Algorithm training times per 100 files featuring embeddings extracted using Hubert

Algorithms	**LR**	**KNN**	**NB**	**SVM**	**RF**	**Bagging**	**Adaboost**	**XGBoost**	**NN**	**Voting**
Italian	00:07	00:02	00:01	00:36	00:20	00:25	00:09	01:28	00:19	00:30
MDVR-KCL	00:19	00:04	00:01	01:18	00:20	00:48	00:13	01:36	00:32	01:11
Czech	00:11	00:01	00:01	00:10	00:57	00:15	00:11	01:02	00:46	00:22
VSP-PDHC	00:22	00:02	00:01	00:12	01:18	00:23	00:12	01:36	01:29	00:27
Mean	00:15	00:02	00:01	00:24	00:38	00:24	00:12	01:32	00:39	00:29

Training the NB and KNN algorithms using the specified embedding methods delivers short training times comparable to those achieved with traditional features, clocking in at under 3 seconds per 100 files.

However, the performance increases when considering the other algorithms. The third fastest is Adaboost, with training times per 100 files between 12 and 15 seconds. XGBoost, being the

slowest, exhibits relatively long execution times. The training duration ranges between 01:25 and 01:53 per 100 files.

By calculating the ratio of the average execution times using embeddings over traditional features, the values in table 5.30 were obtained.

Table 5.30: Ratio of running time using embeddings over traditional features

	LR	**KNN**	**NB**	**SVM**	**RF**	**Bagging**	**Adaboost**	**XGBoost**	**NN**	**Voting**
Ratio	3.87	1.00	1.07	1.77	1.08	9.47	4.54	7.28	0.86	1.47

When comparing the two approaches, NN, KNN, NB, and RF display similar training times. The remaining algorithms exhibit between 1.47 and 9.47 times more to train 100 samples using embeddings.

Chapter 6

Discussion

6.1 Iterative process

From the first iteration of the classification approach, the model gave us quite promising results, showcasing the effectiveness of different machine learning algorithms. The algorithms consistently demonstrated high accuracy and well-balanced precision, recall, and F1-score.

Observing the accuracies in the first phase, Random Forest, Bagging, AdaBoost, and XGBoost demonstrated high results for both datasets (Italian and MDVR-KCL). These ensemble methods appeared to correctly capture complex patterns in the voice data, emphasizing their potential as robust classifiers. The performance of these algorithms aligns with the ensemble nature of their approach, which uses multiple weaker models to create a powerful predictive system. The remaining algorithms, LR, KNN, NB, and SVM, also exhibited respectable results, scoring between 0.72 and 0.81 accuracy for the Italian dataset and between 0.70 and 0.80 for the MDVR-KCL dataset. These observations highlight the competence of both linear and non-linear approaches in this task. The models' competitive performance in this context supports their suitability for classification of Parkinson's disease.

The more modest performance of KNN and Naive Bayes in the first iteration initially suggested that the simplicity of these models could bring limitations in capturing the data patterns. However, the later metrics showed they could provide decent results and not fall behind other, more complex algorithms.

The performance metrics obtained during the initial phase served as a benchmark for evaluating the effectiveness of later refinements and enhancements in the classification model.

In the second phase of the experimentation, the study delved deeper into refining the model, mainly focusing on feature reduction techniques and their impact on algorithmic performance. While these algorithms exhibited competitive results in the initial phase, their effectiveness diminishes when the set of features is reduced. Introducing multicollinearity mitigation through

the VIF appears to hurt the algorithms' performances. This effect was particularly observable in the MDVR-KCL dataset. The responsibility can be attributed to the feature reduction, resulting in only four non-target features.

While LR and SVM maintained relative robustness in Phase 2, ensemble methods show a more mixed response, suggesting their diminished effectiveness was also caused by the reduction of features. Considering that most datasets can be robust against multicollinearity, it was decided that keeping the original set of features was more beneficial than using the VIF reduction technique.

The decision to omit the multicollinearity reduction step and the shift of focus toward scaling the features using *MinMax* provided better results than the previous two iterations. This decision emphasized the importance of scaling techniques.

The results of Phase 4 gave a view of the model's performance when trained and tested on joint information from two distinct datasets. This phase introduced a new layer of complexity by combining data from different vocal exercises. The decline in performance implies that the training data from each dataset may be effective in classifying data from the same dataset but is probably incapable of classifying data from the other. This sensitivity can be attributed to variations in speech tasks and potential differences in recording conditions.

Phase 5 presented consistent or slight improvements when comparing the values to Phase 3. It suggests that the incorporation of PCA successfully condenses the Shimmer and Jitter variables into informative components without sacrificing much predictive power. The application of PCA maintained the model's ability to capture essential patterns in the data. While PCA successfully simplifies the feature space, it seems to have introduced little benefit in the discriminative power of the model when considering the top-performing algorithms from both iterations. Considering that the PCA-based approach generally produced comparable results, the decision was to discontinue PCA for further iterations. Instead, utilizing the complete set of features was favored. This shift was encouraged by the belief that, in this context, maintaining the richness of the original feature space proved to be more beneficial for achieving optimal accuracy.

The noticeable improvements in various algorithms observed during Phase 6 suggest that the refinements made in the iteration positively influenced the overall classification performance. Besides this, the Neural Network algorithm added a layer of complexity, and the soft voting classifier guaranteed balanced accuracy for further iterations.

The introduction of two additional datasets, the *Synthetic Vowels of Speakers with Parkinson's Disease and Parkinsonism* (Czech) dataset, and the *Voice Samples for Patients with Parkinson's Disease and Healthy Controls* (VSP-PDHC) dataset, represented an opportunity to evaluate the model further. This inclusion aimed to estimate the generalization capabilities of the predictive model across a larger range of datasets, providing a better understanding of its usability across varied scenarios.

The accuracy achieved on the Czech and VSP-PDHC datasets during Phase 6 did not surpass that of the Italian and MDVR-KCL datasets. This result was expected, considering that the Czech and VSP-PDHC datasets have fewer samples than the others. The longer recordings in the Czech dataset give it an advantage over the VSP-PDHC dataset, as the latter has less than 3 minutes of total audio. This emphasized the importance of having an adequate dataset to guarantee a reliable model.

Regarding the Naive Bayes algorithm and its significant drop in performance observed after Phase 6, it can be attributed to the algorithm's assumption that the features are conditionally independent given the class label. Consequently, the performance of a Naive Bayes classifier can decrease due to the implementation of the IQR outlier reduction method, considering the method alters the distribution of the features and possibly eliminates instances that the algorithm could have previously deemed relevant for classification. As a result, it can negatively impact the model's ability to make accurate predictions. Besides, the Naive Bayes algorithm is relatively robust to noisy data and outliers.

When analyzing the non-traditional embeddings used during Phase 7, the feature extraction methods demonstrate effectiveness in capturing relevant information. Their performance varies across datasets, with Hubert consistently standing out as the strongest overall performer.

Comparing the results obtained from traditional feature extraction methods from Phase 6 and the feature extraction of embeddings from DNNs, the sets of non-interpretable features from the three embedding extraction methods seem to match, with at least one of them consistently outperforming the results obtained from the traditional features. This can be expected since embeddings capture more nuanced and intricate patterns within the audio. Hubert, in particular, exhibits robust performance across datasets, showcasing its versatility in handling diverse speech recordings. The difference, however, is not abysmal, as both interpretable and non-interpretable features have similar ranges of precision metrics.

Examining the tables comparing the predictions with the actual values (in A), the impact of the types of feature representation was observed. Figure A.1 represents the model trained on traditional features from the Italian dataset. The probability predictions of certain algorithms, like LR, SVM, and XGBoost, appear highly dispersed across the entire domain. This suggests that the features may lack the specificity to confidently distinguish between positive and negative cases.

A significant transformation occurs after the introduction of embeddings into the models. Comparing the previous plot with Figures A.5, A.9, and A.13, which contain the predictions for the same dataset using embeddings, it is clear that they become significantly more polarized for the same algorithms. This indicates that embeddings create features that enable a more distinct separation between PD and control cases. The predictions are confident, even when using simpler algorithms such as LR or KNN. This highlights the importance of the extraction and treatment of data, emphasizing that the effectiveness of the models depends not only on sophisticated algorithms but also on the richness of the underlying data.

This shift from interpretable features to embeddings emphasizes the trade-off between interpretability and predictive power. While traditional features provide meaningful insights into the dataset and showcase features that medical professionals can effortlessly understand, the DNN embeddings in the subsequent phase demonstrate a capacity to uncover hidden patterns and nuances, contributing to better classification accuracy. Adopting feature extraction methods from DNNs, including Trillson, Wav2Vec, and Hubert, marks a reliable alternative to traditional features, resulting in either matching or superior performance in the classification.

It is possible to compare the results with those from other studies that adopted similar approaches. Upon reviewing the study in [35], which utilized a set comprising both interpretable and non-interpretable features, they achieved a maximum accuracy of 91.4%. In the study by [38], the mono-lingual implementation using traditional features resulted in an accuracy of 81%, while non-interpretable features with embeddings yielded an accuracy of 85%. Examining the work of [34], who also employed the same Italian and MDVR-KCL datasets, an accuracy of 87% was attained using only acoustic interpretable features, and 98% was achieved by combining interpretable features with mel-frequency cepstrum coefficient features.

In comparison to the results reported in the studies, the current models exhibit noteworthy performance metrics, providing overall comparable and superior results in certain cases. Using interpretable features alone, they reached up to 95% accuracy (using SVM with the data from the Italian Dataset) and up to 99% using embeddings (SVM with TRILLsson embeddings extracted from the Italian dataset).

6.2 Efficiency

The analysis of training times for the varied ML algorithms using both types of features provided us with insights into the computational efficiency of the models. A crucial observation arises when comparing the training times across traditional features and embeddings. While NB and KNN maintain short training times consistent with traditional features, the dynamics shift for other algorithms. The slower execution times on algorithms like XGBoost and Bagging indicate a trade-off between predictive performance and computational efficiency. The observed range, between 1.47 and 10 times higher for the other algorithms using embeddings, underscores the importance of careful consideration in selecting algorithms and feature types to achieve an optimal balance between computational efficiency and predictive performance.

In the context of the datasets used in the current experimentation, the analysis unveils a notable difference in the time required for feature extraction, training all the algorithms, and testing. The process of dealing with interpretable features takes up to 20 minutes. However, transitioning to non-traditional embeddings can extend it to almost two hours. This difference is particularly accentuated in the Italian and MDVR-KCL datasets containing the most extensive audio samples. This temporal escalation poses a potential challenge, mainly when working with larger datasets, necessitating a thoughtful approach to balance the richness of feature representation and the

associated computational costs. Therefore, the decision of the best approach heavily depends on the dataset used, computational power, and available time. As the study expands to include more extensive datasets, the implications of increased computational demands become critical for the practical deployment of predictive models in clinical applications.

6.3 Challenges

During the development, some main challenges were encountered, mainly related to data acquisition and processing.

Initially, this dissertation aimed to leverage speech recordings acquired in collaboration with partners at Hospital Pedro Hispano, intending to conduct a comprehensive analysis that compared results across multiple recording conditions and methodologies. However, unforeseen delays in obtaining the required recordings produced a shift in the approach. Publicly available datasets were used to maintain the momentum of this research and adapt to the unexpected circumstances. This adjustment redirected the focal point of the study from the original intention of comparing various recording conditions and methods to a more concentrated data exploration and treatment, feature extraction, and modeling using the available resources.

One of the challenges resulting from the shift in the study was the scarcity of public datasets that shared audio recordings related explicitly to Parkinson's disease. Despite extensive efforts to identify suitable datasets, a recurring issue was encountered—most studies did not share their audio data, often due to the privacy protection of the participants. While some studies provided the tables with extracted features, the absence of raw audio recordings posed a significant disadvantage, as feature extraction played a pivotal role in the research. Since the models' performance strongly depends on the data, more diverse datasets would allow us to further evaluate the models' viability.

Feature extraction presented its own set of challenges. Despite Parselmouth emerging as a standard tool for interpretable feature extraction in the observed studies, some investigations demonstrated additional features without providing sufficient details about their extraction methodology. This lack of transparency raised concerns about the reproducibility of these studies and introduced ambiguity regarding the exact nature of the features utilized. The challenges in extracting these traditional features prompted the exploration of less conventional libraries and tools for feature extraction, with the attempts often being unsuccessful.

The data wrangling phase introduced another layer of complexity. Every new transformation applied to the data had to be controlled to prevent unintended alterations. Ensuring that data transformations occurred as expected and did not compromise the integrity of the dataset was imperative for the subsequent modeling stages. This verification proved challenging as the tables featured large amounts of abstract data. Any unintentional changes to the data could potentially introduce biases that would sabotage the validity of the predictive models.

To address this challenge, manual checks were performed repetitively to validate the consistency of the transformed data in each step, minimizing the risk of introducing practices that could compromise the interpretability of the obtained results.

Chapter 7

Conclusion

The study embarked on an exploration of predictive models for detecting Parkinson's disease by analyzing speech recordings. The journey started with investigating the disease's background, shedding light on its causes, symptoms, and diagnosis. The literature review explored the applicability of machine learning approaches to the problem, emphasizing the importance of different data collection, data wrangling, and classification strategies in PD detection.

Following an iterative process for feature extraction, data exploration, and modeling, its approach evolved through seven main phases. The results incited a discussion of performance, efficiency, and the challenges faced throughout development.

The iterative methodology illustrated the effectiveness of ensemble methods like Random Forest, Bagging, AdaBoost, and XGBoost in capturing intricate patterns in voice data, but also that less complex algorithms like Logistic Regression and K-Nearest Neighbor are similarly capable of providing accurate predictions, revealing unexpected resilience and challenging the assumption that more complex models inherently yield superior results.

Introducing additional datasets highlighted the model's potential for generalization and the crucial role of dataset characteristics in achieving accurate predictions.

The efficiency analysis revealed the trade-off between computational speed and predictive performance, enhancing the importance of a thoughtful selection of algorithms and features tailored to the dataset's scale and complexity.

The utilization of both interpretable and non-interpretable features has proven to be a successful strategy, surpassing or aligning with the benchmarks set by previous studies and highlighting the effectiveness of this approach.

In the domain of the research, the expression "there is no such thing as a free lunch" resonates with particular significance. This encapsulates the fundamental concept that no single model or algorithm universally excels across all scenarios. As the iterative process unfolded, the nuanced

interplay between algorithmic choices, feature engineering, and dataset characteristics was witnessed. The concept stresses the importance of tailored, context-specific decisions in predictive modeling, where the ideal strategy depends on considering trade-offs, dataset characteristics, and computational efficiency.

Considering future work, including self-recorded data emerges as a valuable possibility. Training and testing models on a dataset recorded explicitly for the study can provide insights into the models' performance in controlled environments, where recording conditions and participant characteristics are well-documented. This approach could facilitate the exploration of the model's effectiveness and be an opportunity to transition from controlled experimental settings to a real-world application.

In essence, this dissertation not only contributes to the research in Parkinson's disease analysis but also provides practical insights for developing robust predictive models, mainly from speech recordings. The findings highlight the interdisciplinary nature of this research, combining medical knowledge with machine learning methodologies. As it navigates the diagnosis of Parkinson's disease, this study lays a foundation for future advancements, encouraging continued exploration and refinement in predictive models.

References

[1] Imran Ahmed, Sultan Aljahdali, Muhammad Shakeel Khan, and Sanaa Kaddoura. Classification of parkinson disease based on patient's voice signal using machine learning. *Intelligent Automation and Soft Computing*, 32:705–722, 2022.

[2] Siddharth Arora and Athanasios Tsanas. Assessing parkinson's disease at scale using telephone-recorded speech: Insights from the parkinson's voice initiative. *Diagnostics*, 11, 10 2021.

[3] Federica Amato, Luigi Borzì, Gabriella Olmo, and Juan Rafael Orozco-Arroyave. An algorithm for parkinson's disease speech classification based on isolated words analysis. *Health Information Science and Systems*, 9:32, 12 2021.

[4] Sonja von Campenhausen, Yaroslav Winter, Antonio Rodrigues e Silva, Christina Sampaio, Evzen Ruzicka, Paolo Barone, Werner Poewe, Alla Guekht, Céu Mateus, Karl P. Pfeiffer, Karin Berger, Jana Skoupa, Kai Bötzel, Sabine Geiger-Gritsch, Uwe Siebert, Monika Balzer-Geldsetzer, Wolfgang H. Oertel, Richard Dodel, and Jens P. Reese. Costs of illness and care in parkinson's disease: An evaluation in six countries. *European Neuropsychopharmacology*, 21:180–191, 2 2011.

[5] J. Massano and K. P. Bhatia. Clinical approach to parkinson's disease: Features, diagnosis, and principles of management. *Cold Spring Harbor Perspectives in Medicine*, 2:a008870–a008870, 6 2012.

[6] Juliana Rajão Guerra and Doutor R João Manuel S Tavares. Parkinson's disease diagnosis: A machine learning and data mining based approach, 2019.

[7] B Thanvi and S Treadwell. Drug induced parkinsonism: a common cause of parkinsonism in older people. *Postgraduate Medical Journal*, 85:322–326, 6 2009.

[8] Christopher Frank, Giovanna Pari, John P Rossiter, and Mb Bch. Cme approach to diagnosis of parkinson disease, 2006.

[9] Jan Hlavnika, Roman Cmejla, Tereza Tykalová, Karel Šonka, Evzen Ruzicka, and Jan Rusz. Automated analysis of connected speech reveals early biomarkers of parkinson's disease in patients with rapid eye movement sleep behaviour disorder. *Scientific Reports*, 7, 2 2017.

[10] H Ackermann and W Ziegler. Articulatory deficits in parkinsonian dysarthria: an acoustic analysis. *Journal of Neurology, Neurosurgery Psychiatry*, 54:1093–1098, 12 1991.

[11] Md Sakibur Rahman Sajal, Md Tanvir Ehsan, Ravi Vaidyanathan, Shouyan Wang, Tipu Aziz, and Khondaker Abdullah Al Mamun. Telemonitoring parkinson's disease using machine learning by combining tremor and voice analysis. *Brain Informatics*, 7, 12 2020.

[12] Martin Korbinian and Heinrich Strauß. Modelling of speech aspects in parkinson's disease by multitask deep learning modellieren von sprachaspekten bei parkinson mittels multitask deep learning, 2019.

[13] Ranadeep Deb, Sizhe An, Ganapati Bhat, Holly Shill, and Umit Y. Ogras. A systematic survey of research trends in technology usage for parkinson's disease, 8 2022.

[14] Anil Kumar, Shubham Bind, Arvind Kumar Tiwari, and Anil Kumar Sahani. A survey of machine learning based approaches for parkinson disease prediction, 2022.

[15] Mohamed Shaban. Deep learning for parkinson's disease diagnosis: A short survey. *Computers*, 12:58, 3 2023.

[16] Mohammad Shahbakhi, Danial Taheri Far, and Ehsan Tahami. Speech analysis for diagnosis of parkinson's disease using genetic algorithm and support vector machine. *Journal of Biomedical Science and Engineering*, 07:147–156, 2014.

[17] Laureano Moro-Velazquez, Jorge A. Gomez-Garcia, Julian D. Arias-Londoño, Najim Dehak, and Juan I. Godino-Llorente. Advances in parkinson's disease detection and assessment using voice and speech: A review of the articulatory and phonatory aspects. *Biomedical Signal Processing and Control*, 66, 4 2021.

[18] Evaldas Vaiciukynas, Antanas Verikas, Adas Gelzinis, and Marija Bacauskiene. Detecting parkinson's disease from sustained phonation and speech signals. *PLOS ONE*, 12:e0185613, 10 2017.

[19] Javier Carrón, Yolanda Campos-Roca, Mario Madruga, and Carlos J. Pérez. A mobile-assisted voice condition analysis system for parkinson's disease: assessment of usability conditions. *BioMedical Engineering Online*, 20, 12 2021.

[20] Diogo Braga, Ana M. Madureira, Luis Coelho, and Reuel Ajith. Automatic detection of parkinson's disease based on acoustic analysis of speech. *Engineering Applications of Artificial Intelligence*, 77:148–158, 1 2019.

[21] Betul Erdogdu Sakar, M. Erdem Isenkul, C. Okan Sakar, Ahmet Sertbas, Fikret Gurgen, Sakir Delil, Hulya Apaydin, and Olcay Kursun. Collection and analysis of a parkinson speech dataset with multiple types of sound recordings. *IEEE Journal of Biomedical and Health Informatics*, 17:828–834, 7 2013.

[22] Betul-Isenkul M. Sakar C. Sertbas Ahmet Olcay, Sakar and Fikret Kursun Gurgen. Uci machine learning repository., 2014.

[23] R. A. Armstrong. Visual symptoms in parkinson's disease. *Parkinson's Disease*, 2011:1–9, 2011.

[24] Wee Shin Lim, Shu I. Chiu, Meng Ciao Wu, Shu Fen Tsai, Pu He Wang, Kun Pei Lin, Yung Ming Chen, Pei Ling Peng, Yung Yaw Chen, Jyh Shing Roger Jang, and Chin Hsien Lin. An integrated biometric voice and facial features for early detection of parkinson's disease. *npj Parkinson's Disease*, 8, 12 2022.

[25] Ilias Tougui, Abdelilah Jilbab, and Jamal El Mhamdi. Machine learning smart system for parkinson disease classification using the voice as a biomarker. *Healthcare Informatics Research*, 28:210–221, 7 2022.

[26] Giovanni Dimauro, Vincenzo Di Nicola, Vitoantonio Bevilacqua, Danilo Caivano, and Francesco Girardi. Assessment of speech intelligibility in parkinson's disease using a speech-to-text system. *IEEE Access*, 5:22199–22208, 2017.

[27] Hagen Jaeger, Dhaval Trivedi, and Michael Stadtschnitzer. Mo- bile device voice recordings at king's college london (mdvr-kcl) from both early and advanced parkin- son's disease patients and healthy controls. 2020.

[28] Jan Hlavnicka, Roman Cmejla, Jiri Klempir, Evzen Ruzicka, and Jan Rusz. Acoustic tracking of pitch, modal, and subharmonic vibrations of vocal folds in parkinson's disease and parkinsonism. *IEEE Access*, 7:150339–150354, 2019.

[29] Anu Iyer, Aaron Kemp, Yasir Rahmatallah, Lakshmi Pillai, Aliyah Glover, Fred Prior, Linda Larson-Prior, and Tuhin Virmani. A machine learning method to process voice samples for identification of parkinson's disease. *Scientific Reports*, 13, 12 2023.

[30] Athanasios Tsanas, Max A. Little, and Lorraine O. Ramig. Remote assessment of parkinson's disease symptom severity using the simulated cellular mobile telephone network. *IEEE Access*, 9:11024–11036, 2021.

[31] P. Boersma. Praat: Doing phonetics by computer, 2006.

[32] Lucijano Berus, Simon Klancnik, Miran Brezocnik, and Mirko Ficko. Classifying parkinson's disease based on acoustic measures using artificial neural networks. *Sensors (Switzerland)*, 19, 1 2019.

[33] Chethan C R. Diagnosis of parkinson disorder through speech data. *International Journal for Research in Applied Science and Engineering Technology*, 8:337–341, 9 2020.

[34] Adedolapo Aishat Toye and Suryaprakash Kompalli. Comparative study of speech analysis methods to predict parkinson's disease. 11 2021.

[35] Max A. Little, Patrick E. McSharry, Eric J. Hunter, Jennifer Spielman, and Lorraine O. Ramig. Suitability of dysphonia measurements for telemonitoring of parkinson's disease. *IEEE Transactions on Biomedical Engineering*, 56:1015–1022, 4 2009.

[36] Betul Erdogdu Sakar, Gorkem Serbes, and C. Okan Sakar. Analyzing the effectiveness of vocal features in early telediagnosis of parkinson's disease. *PLoS ONE*, 12, 8 2017.

[37] Athanasios Tsanas, Max A. Little, Patrick E. McSharry, and Lorraine O. Ramig. Accurate telemonitoring of parkinsons disease progression by noninvasive speech tests. *IEEE Transactions on Biomedical Engineering*, 57:884–893, 4 2010.

[38] Anna Favaro, Yi Ting Tsai, Ankur Butala, Thomas Thebaud, Jesús Villalba, Najim Dehak, and Laureano Moro-Velázquez. Interpretable speech features vs. dnn embeddings: What to use in the automatic assessment of parkinson's disease in multi-lingual scenarios. *Computers in Biology and Medicine*, 166, 11 2023.

[39] Sanjana Singh and Wenyao Xu. Robust detection of parkinson's disease using harvested smartphone voice data: A telemedicine approach. *Telemedicine and e-Health*, 26:327–334, 3 2020.

[40] Preeti Khera and Neelesh Kumar. Novel machine learning-based hybrid strategy for severity assessment of parkinson's disorders. *Medical and Biological Engineering and Computing*, 60:811–828, 3 2022.

[41] Saurabh Prasad and Lori Mann Bruce. Limitations of principal components analysis for hyperspectral target recognition. *IEEE Geoscience and Remote Sensing Letters*, 5:625–629, 10 2008.

[42] Farhad Abedinzadeh Torghabeh, Seyyed Abed Hosseini, and Elham Ahmadi Moghadam. Enhancing parkinson's disease severity assessment through voice-based wavelet scattering, optimized model selection, and weighted majority voting. *Medicine in Novel Technology and Devices*, 20, 12 2023.

[43] John Prince, Fernando Andreotti, and Maarten De Vos. Multi-source ensemble learning for the remote prediction of parkinson's disease in the presence of source-wise missing data. *IEEE Transactions on Biomedical Engineering*, 66:1402–1411, 5 2019.

[44] Tomáš Kouba, Vojtech Illner, and Jan Rusz. Study protocol for using a smartphone application to investigate speech biomarkers of parkinson's disease and other synucleinopathies: Smartspeech. *BMJ Open*, 12, 6 2022.

[45] Salama A. Mostafa, Aida Mustapha, Mazin Abed Mohammed, Raed Ibraheem Hamed, N. Arunkumar, Mohd Khanapi Abd Ghani, Mustafa Musa Jaber, and Shihab Hamad Khaleefah. Examining multiple feature evaluation and classification methods for improving the diagnosis of parkinson's disease. *Cognitive Systems Research*, 54:90–99, 5 2019.

[46] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. pages 776–780. IEEE, 3 2017.

[47] J. Kahn, M. Riviere, W. Zheng, E. Kharitonov, Q. Xu, P.E. Mazare, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux. Libri-light: A benchmark for asr with limited or no supervision. pages 7669–7673. IEEE, 5 2020.

[48] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. pages 5206–5210. IEEE, 4 2015.

[49] Ch. Sanjeev Kumar Dash, Ajit Kumar Behera, Satchidananda Dehuri, and Ashish Ghosh. An outliers detection and elimination framework in classification task of data mining. *Decision Analytics Journal*, 6:100164, 3 2023.

[50] Rick H. Hoyle. *Structural Equation Modeling - Concepts, Issues and Applications*. Sage Publications, 1995.

[51] James Gareth, Daniela Witten, Trevor Hastie, Robert Tibshirani, and Jonathan Taylor. *An Introduction to Statistical Learning*. 2023.

[52] Begüm Çığşar and Deniz Ünal. Comparison of data mining classification algorithms determining the default risk. *Scientific Programming*, 2019:1–8, 2 2019.

[53] Suganya A., Mohanapriya N., and B. Kalaavathi. Lung nodule classification techniques for low dose computed tomography (ldct) scan images as survey. *International Journal of Computer Applications*, 131:12–15, 12 2015.

[54] Jacob T. Vanderplas. *Python Data Science Handbook*. 11 2016.

[55] Robert E Schapire. A brief introduction to boosting, 1999.

[56] Damir Krstinić, Maja Braović, Ljiljana Šerić, and Dunja Božić-Štulić. Multi-label classifier performance evaluation with confusion matrix. pages 01–14. Academy and Industry Research Collaboration Center (AIRCC), 6 2020.

Appendix A

Comparison of predicted probabilities and actual values

In Appendix A, we present the collection of plots that depict the comparison between the actual values and the values predicted by our models. For the predictions based on interpretable features, we use the implementation from Phase 6, and for non-interpretable features, we use the model from Phase 7.

These plots serve as a visual representation of the model's performance. By including these plots, we aim to provide a comprehensive and transparent overview of the predictive accuracy achieved by our machine learning model across different refinement phases.

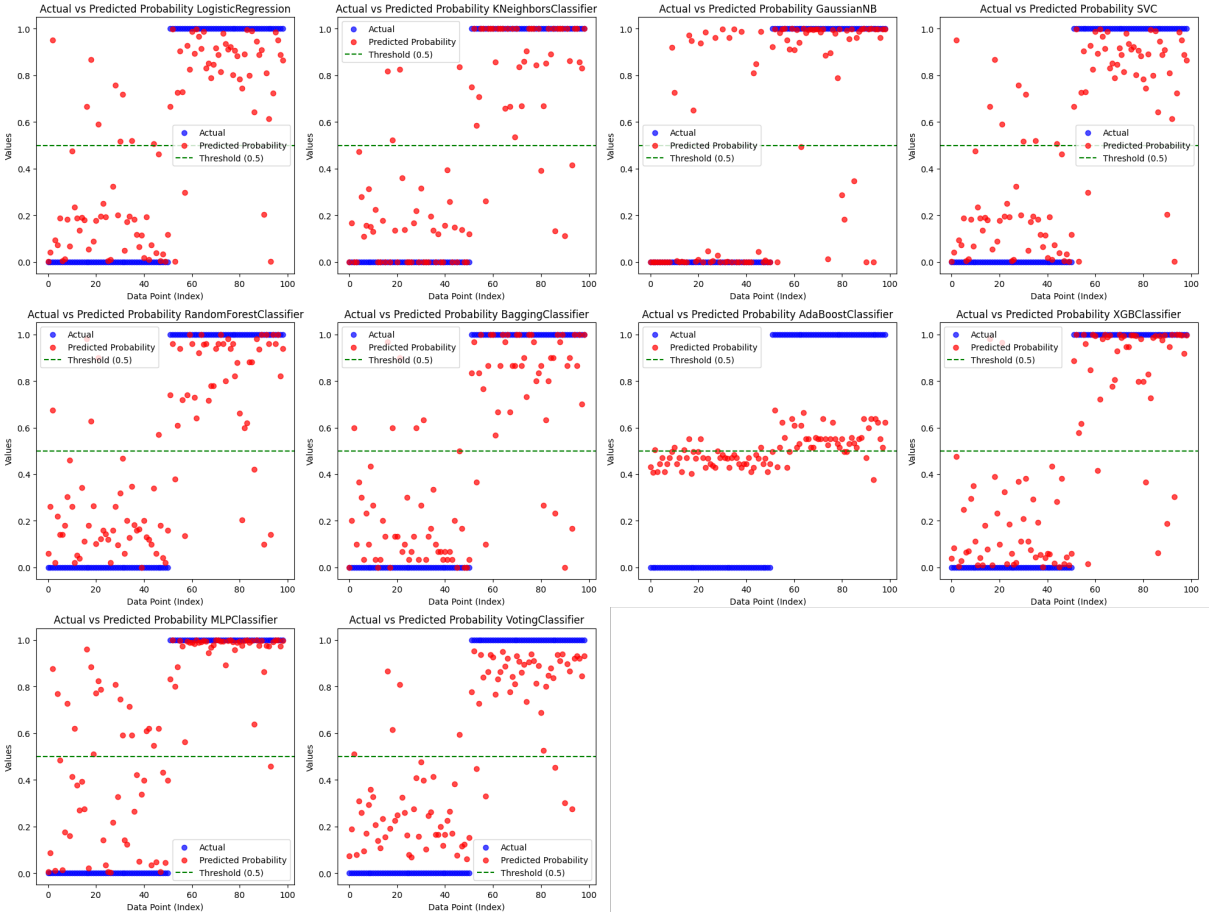

Figure A.1: Comparison of predictions and actual values using traditional features from the Italian dataset

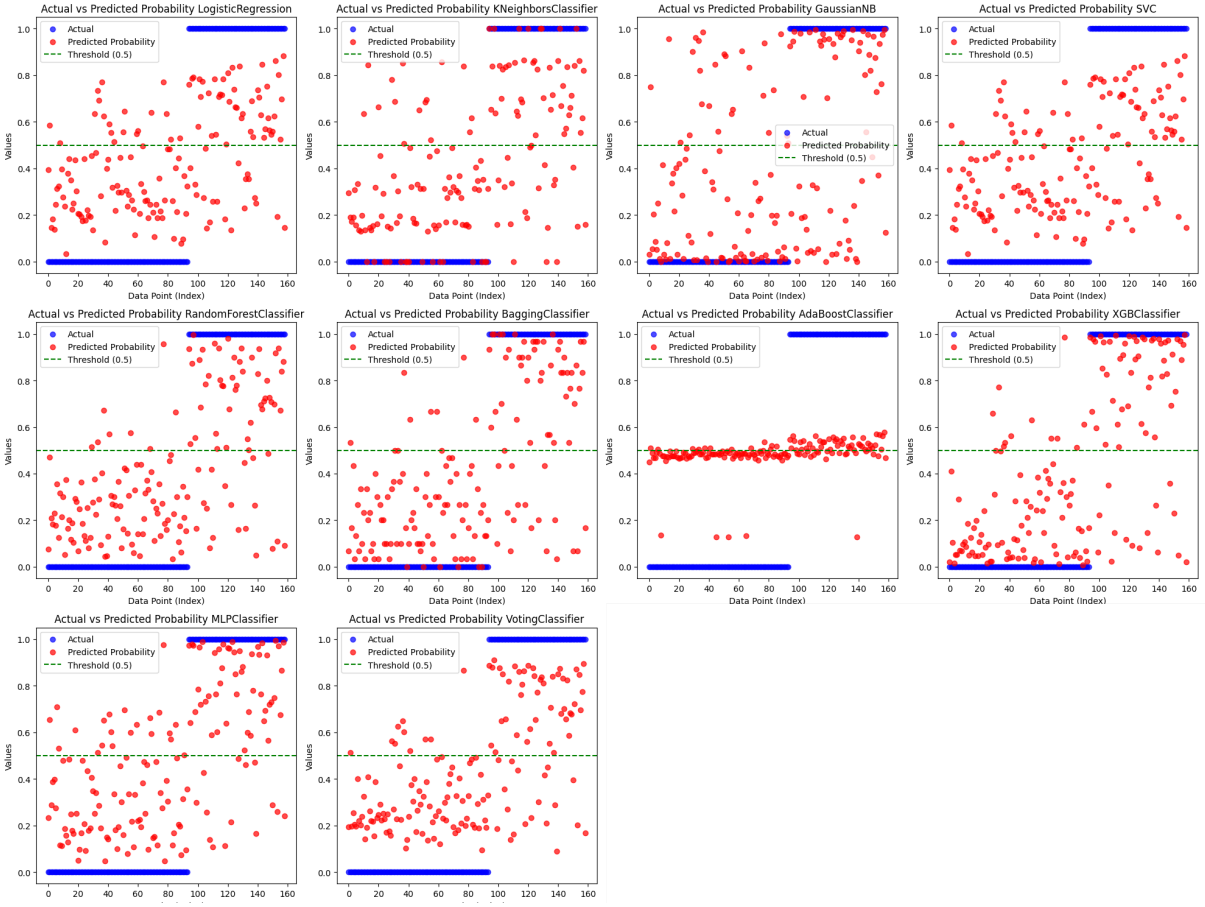

Figure A.2: Comparison of predictions and actual values using traditional features from the MDVR-KCL dataset

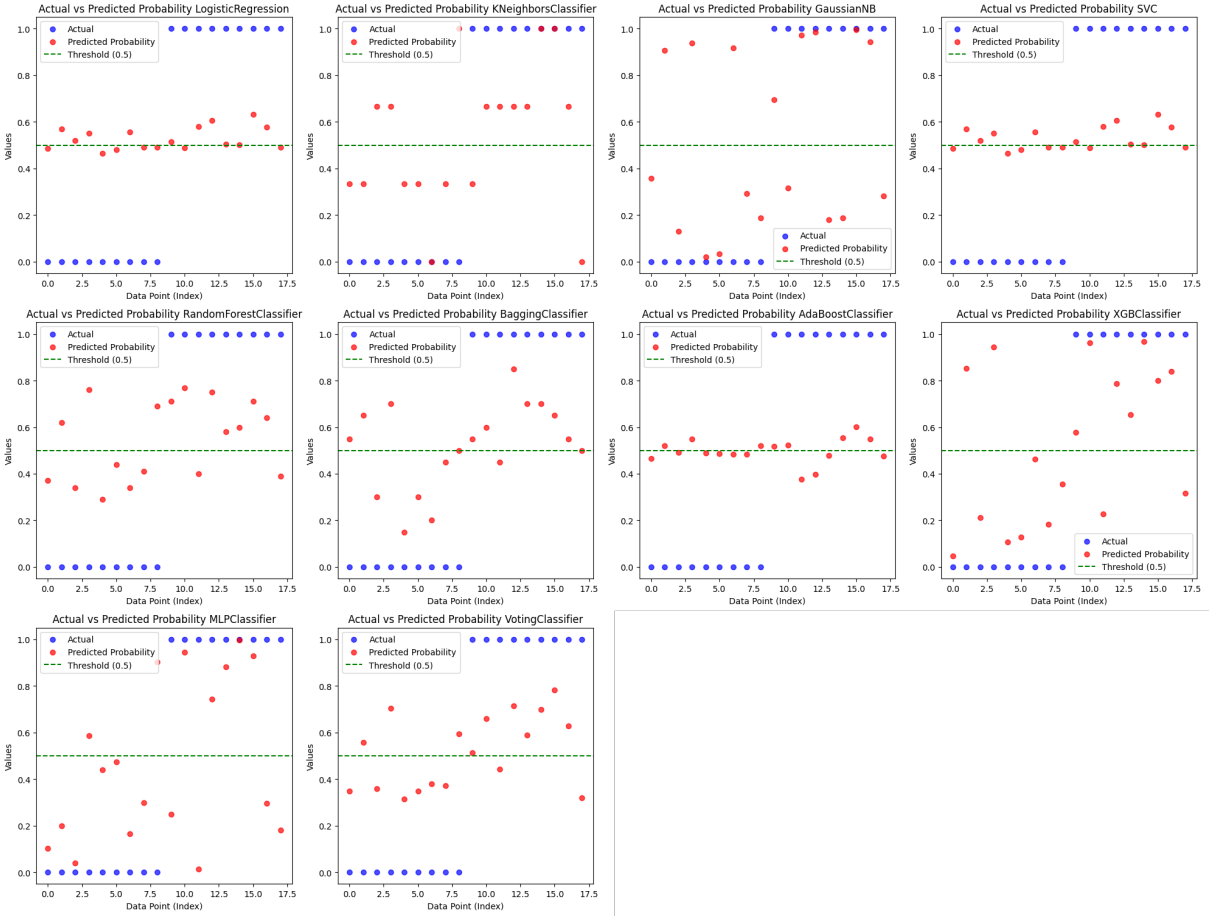

Figure A.3: Comparison of predictions and actual values using traditional features from the Czech dataset

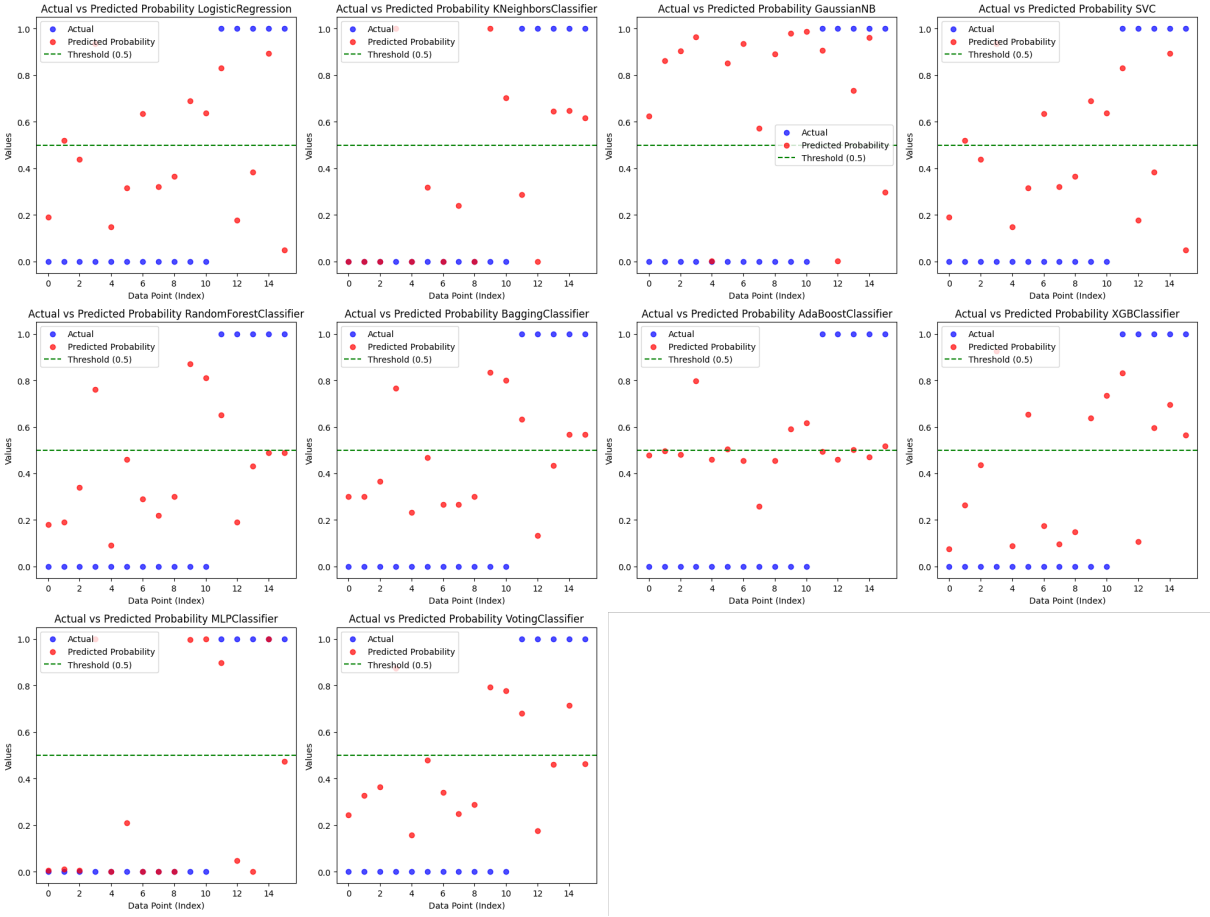

Figure A.4: Comparison of predictions and actual values using traditional features from the VSP-PDHC dataset

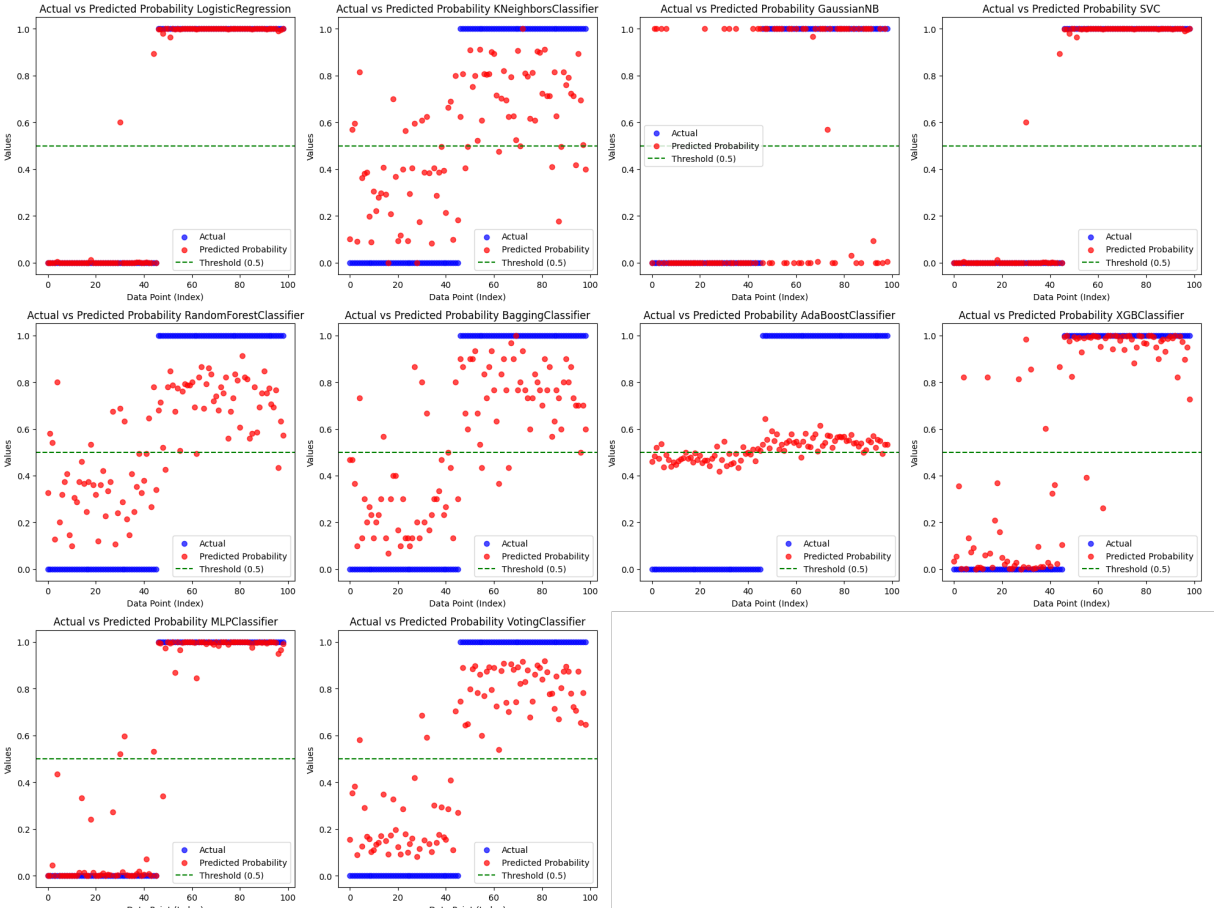

Figure A.5: Comparison of predictions and actual values using Trillsson features from the Italian dataset

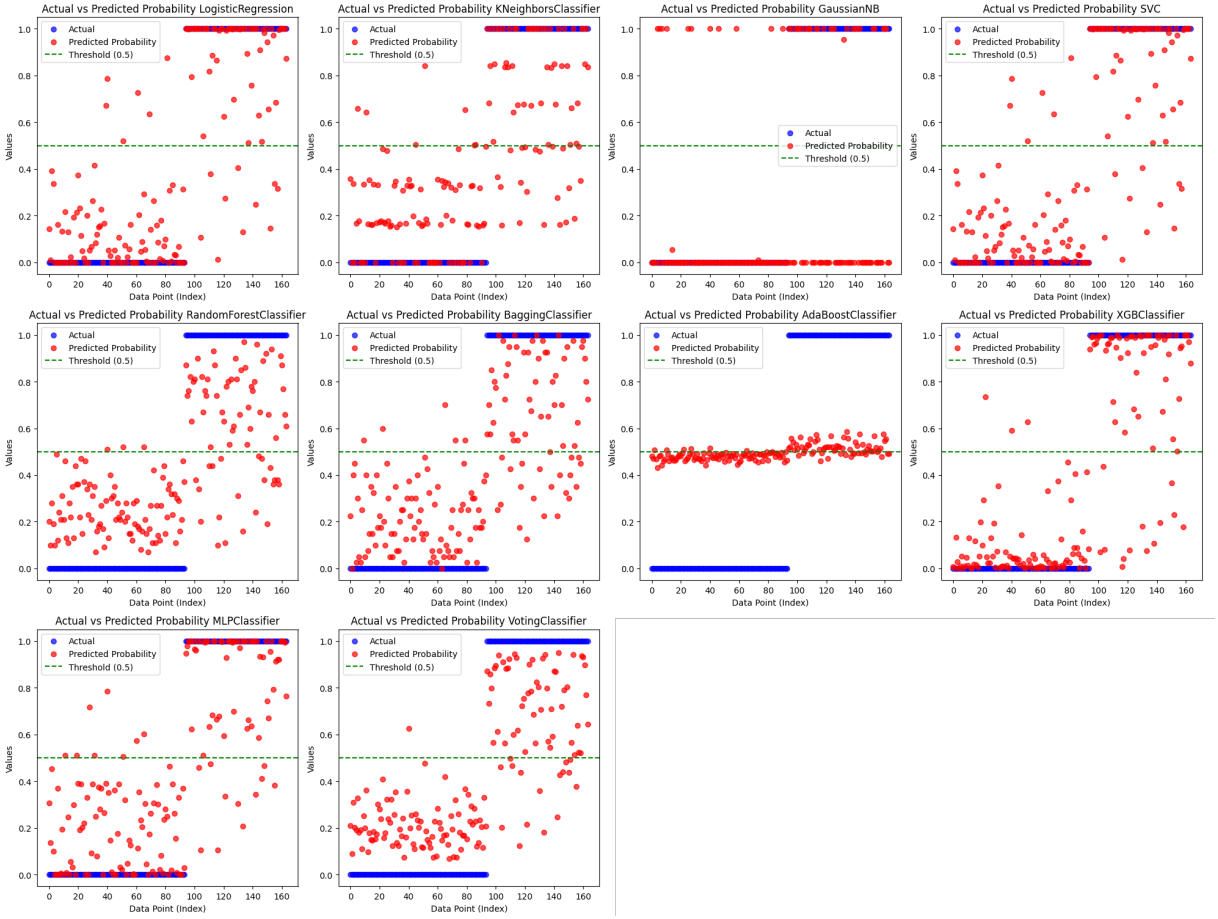

Figure A.6: Comparison of predictions and actual values using Trillsson features from the MDVR-KCL dataset

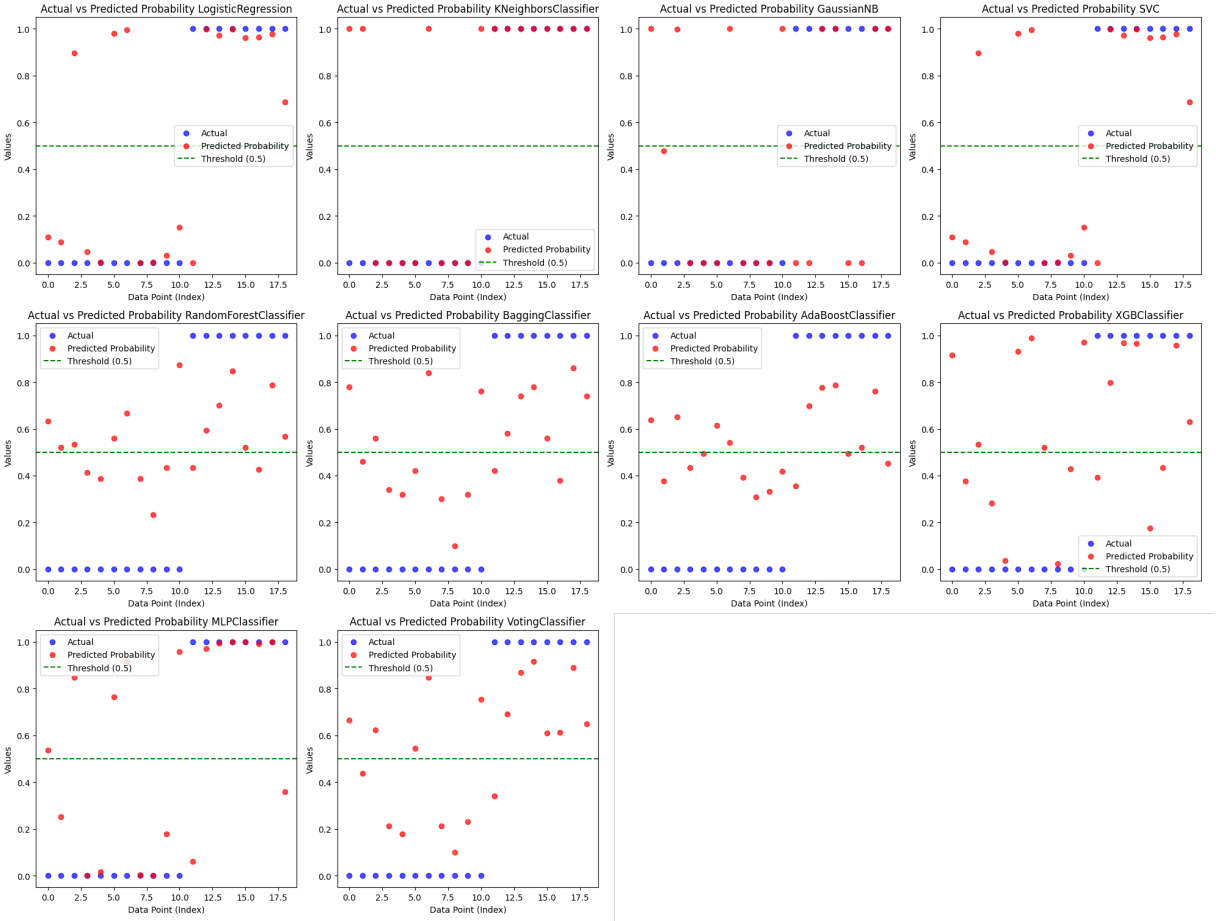

Figure A.7: Comparison of predictions and actual values using Trillsson features from the Czech dataset

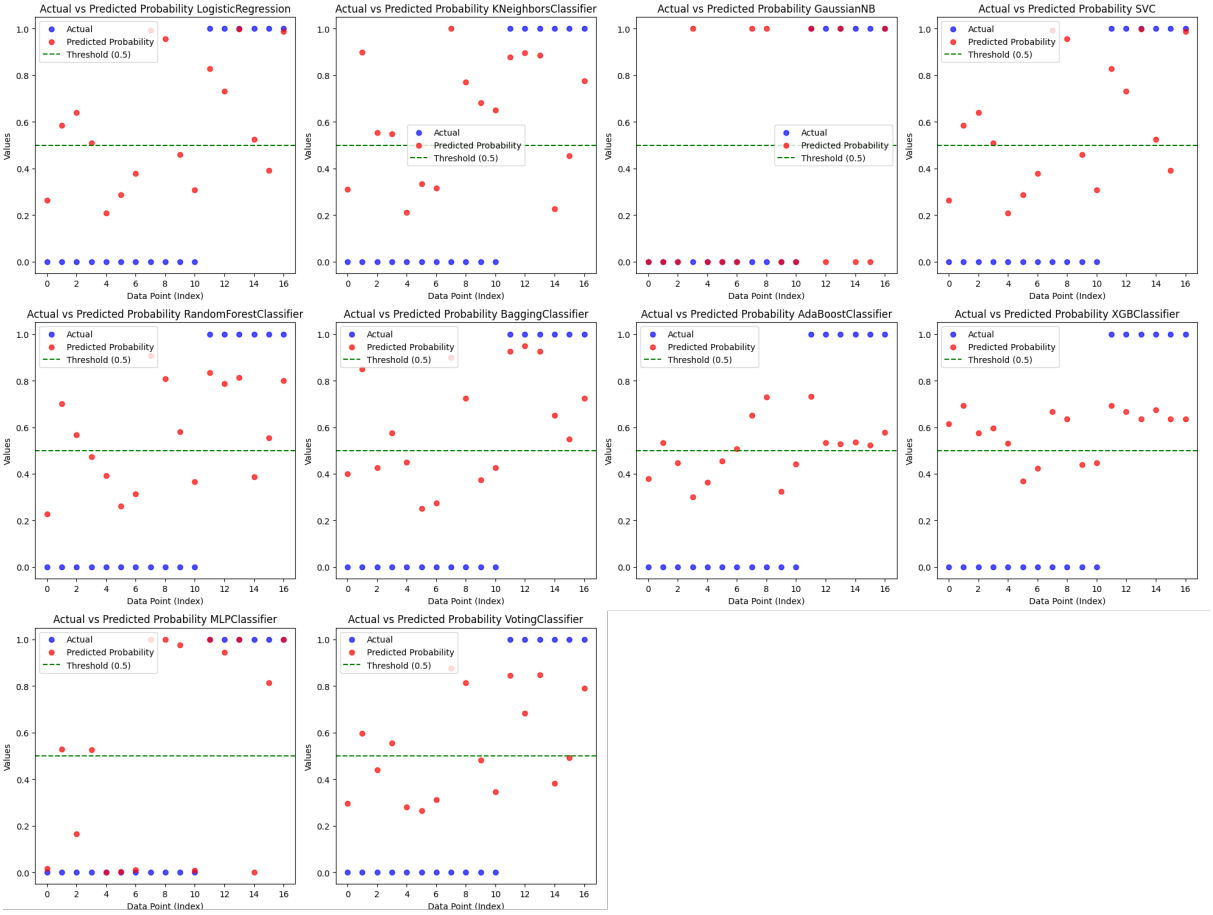

Figure A.8: Comparison of predictions and actual values using Trillsson features from the VSP-PDHC dataset

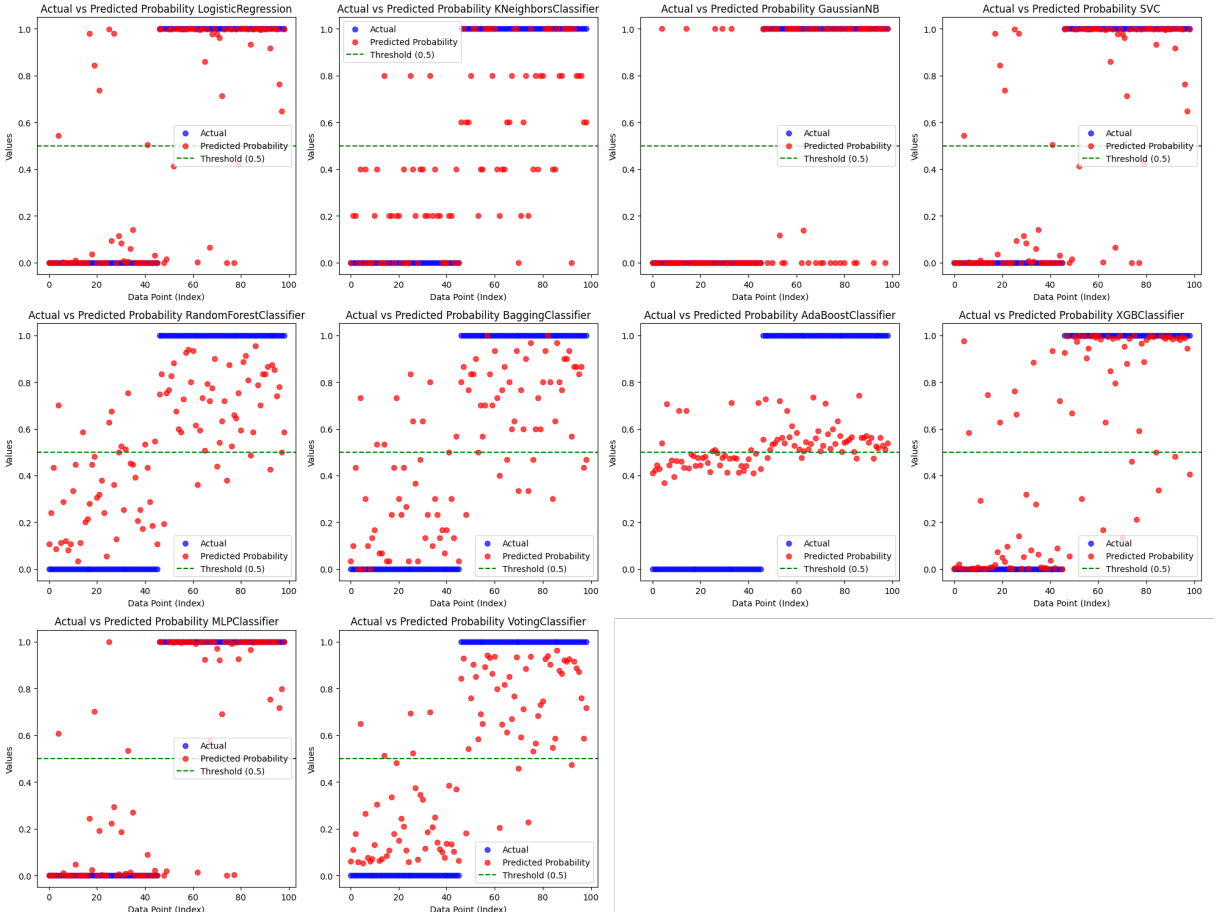

Figure A.9: Comparison of predictions and actual values using WAV2VEC features from the Italian dataset

Figure A.10: Comparison of predictions and actual values using WAV2VEC features from the MDVR-KCL dataset

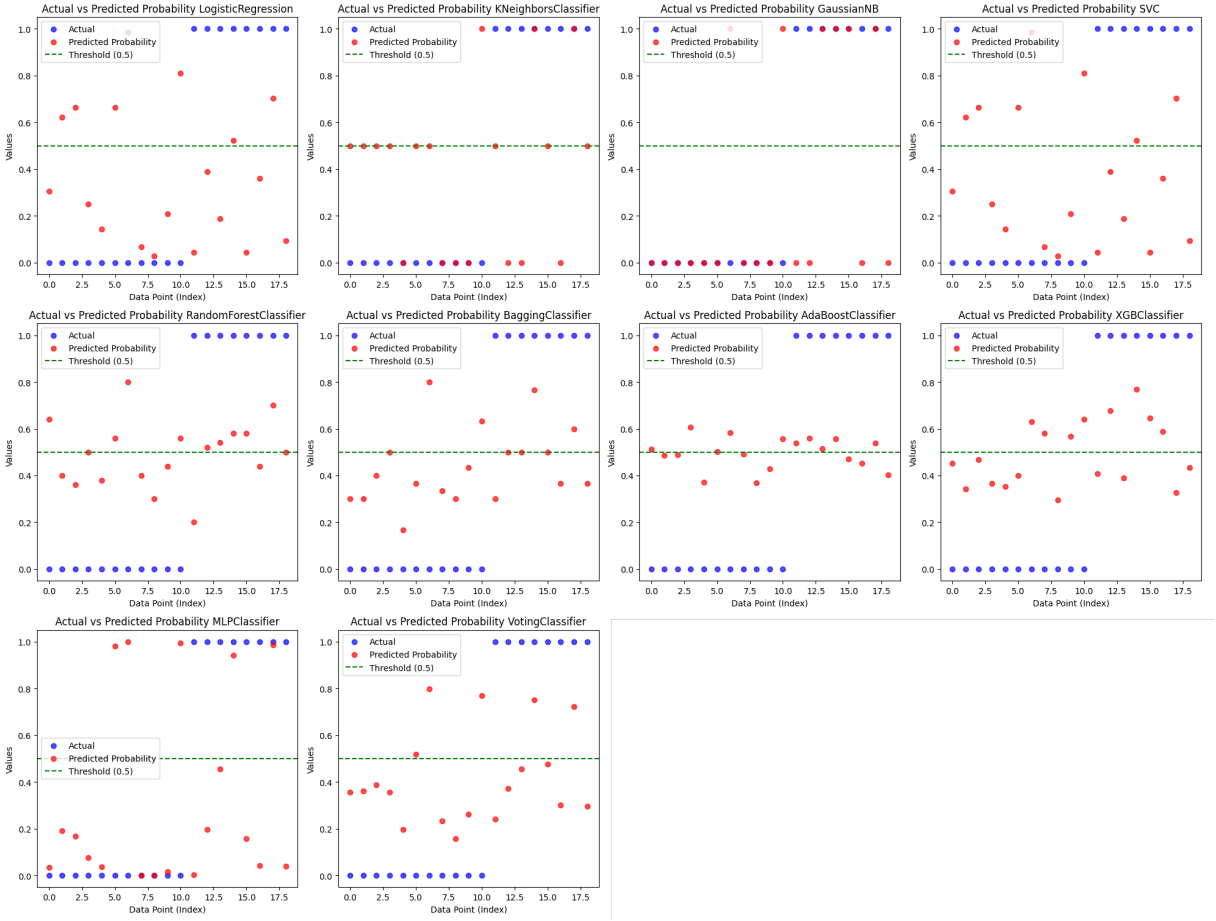

Figure A.11: Comparison of predictions and actual values using WAV2VEC features from the Czech dataset

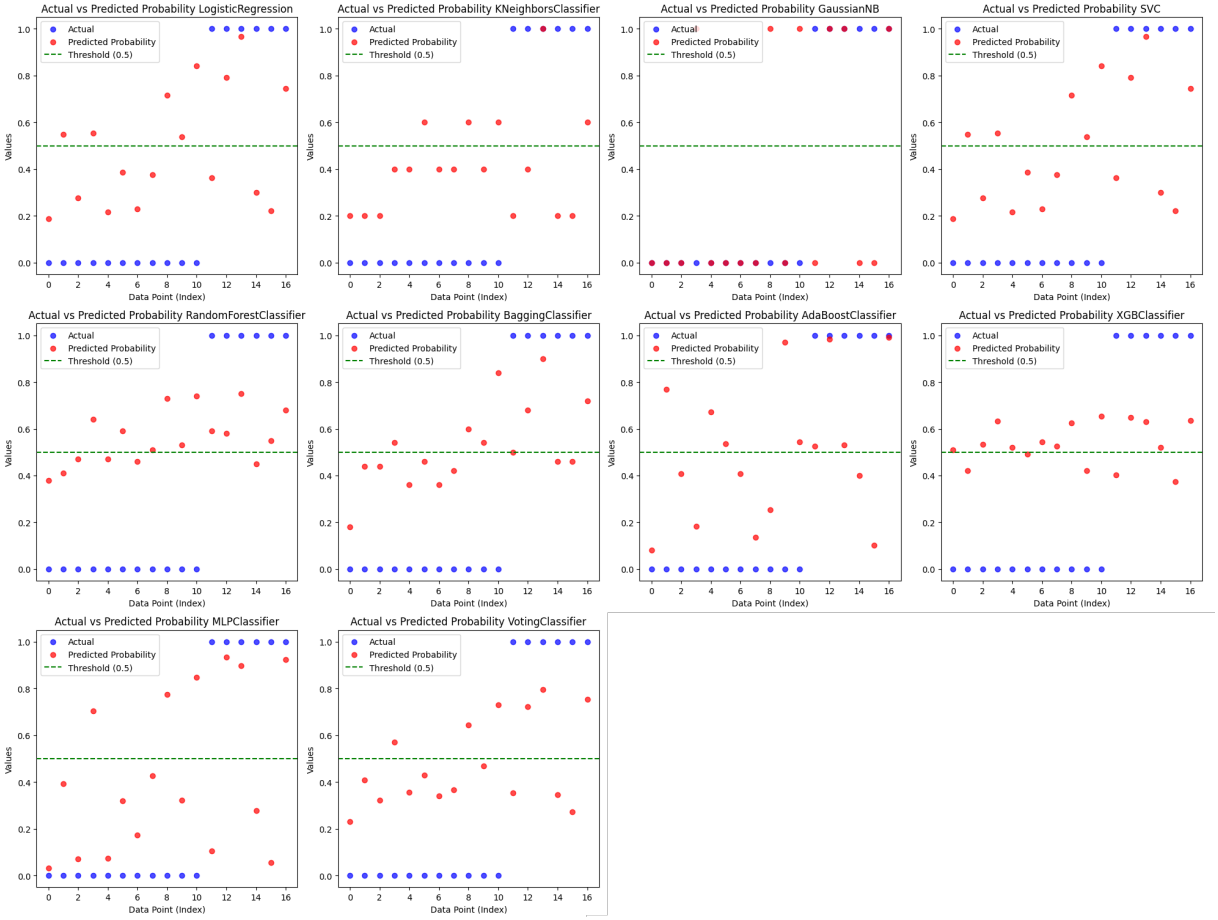

Figure A.12: Comparison of predictions and actual values using WAV2VEC features from the VSP-PDHC dataset

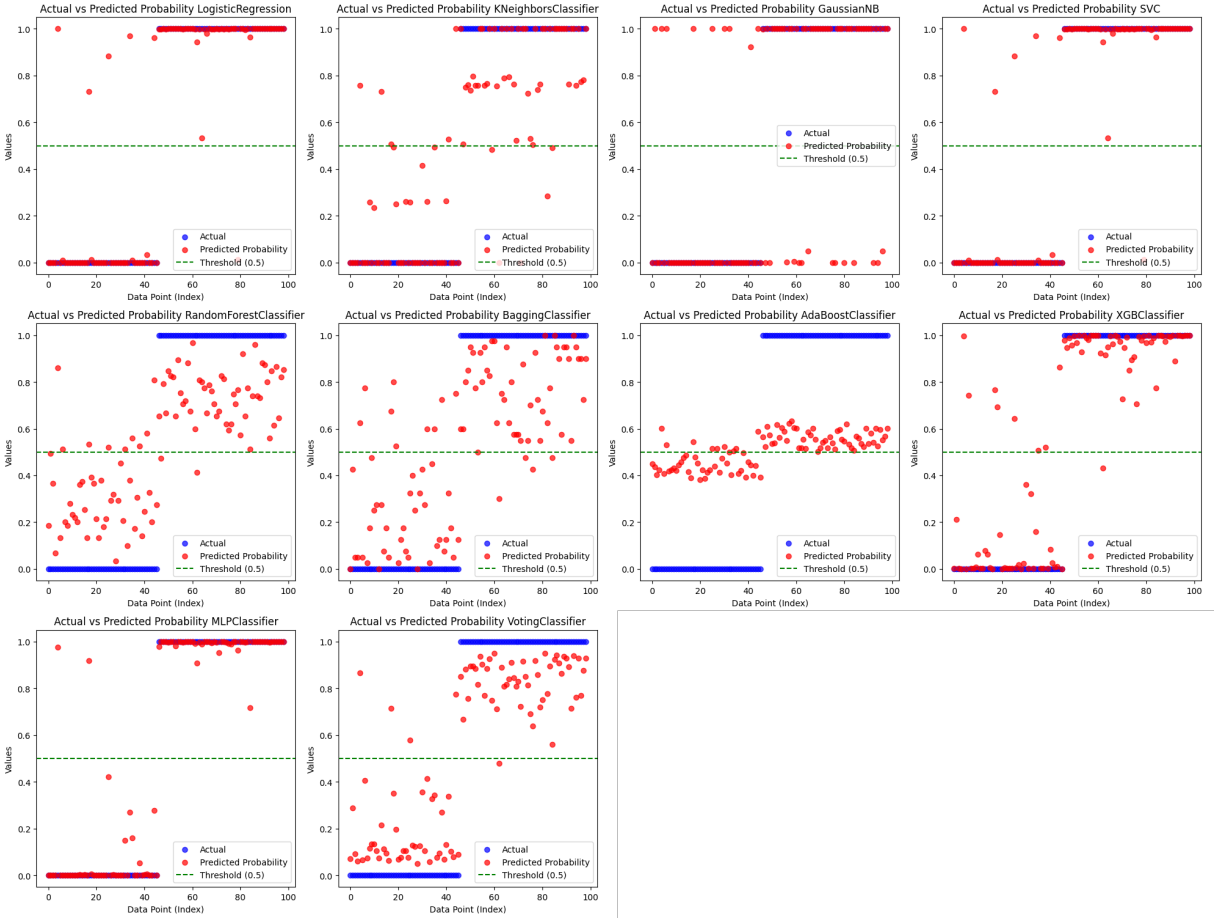

Figure A.13: Comparison of predictions and actual values using Hubert features from the Italian dataset

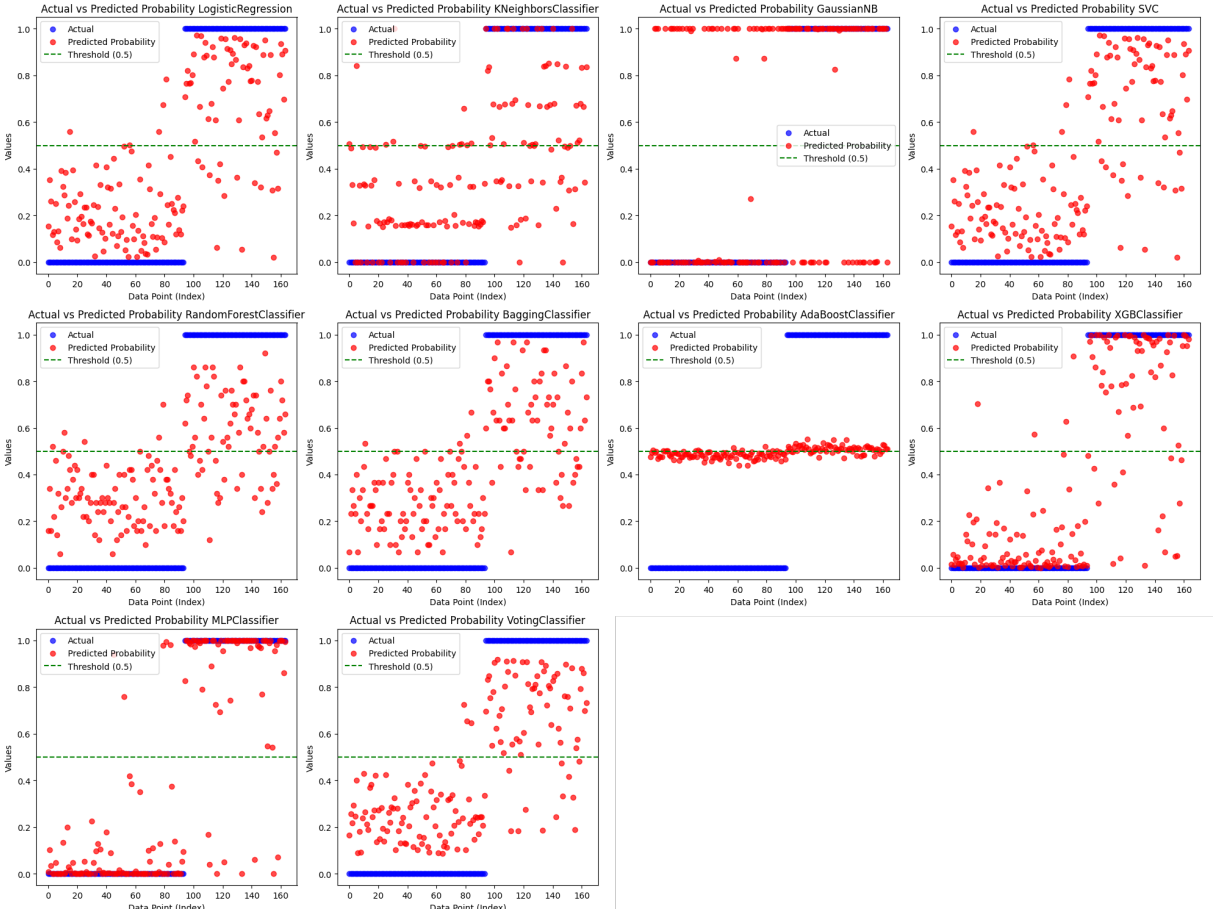

Figure A.14: Comparison of predictions and actual values using Hubert features from the MDVR-KCL dataset

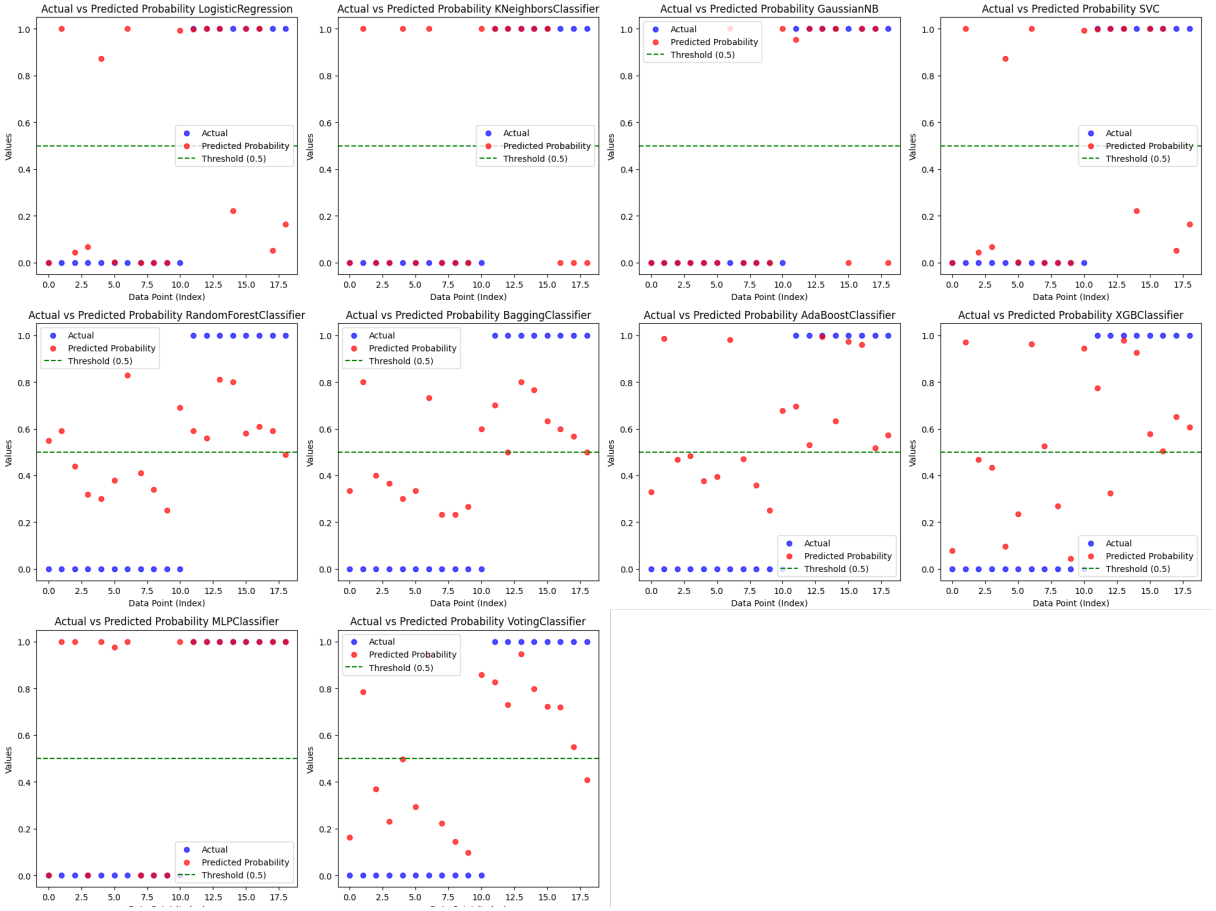

Figure A.15: Comparison of predictions and actual values using Hubert features from the Czech dataset

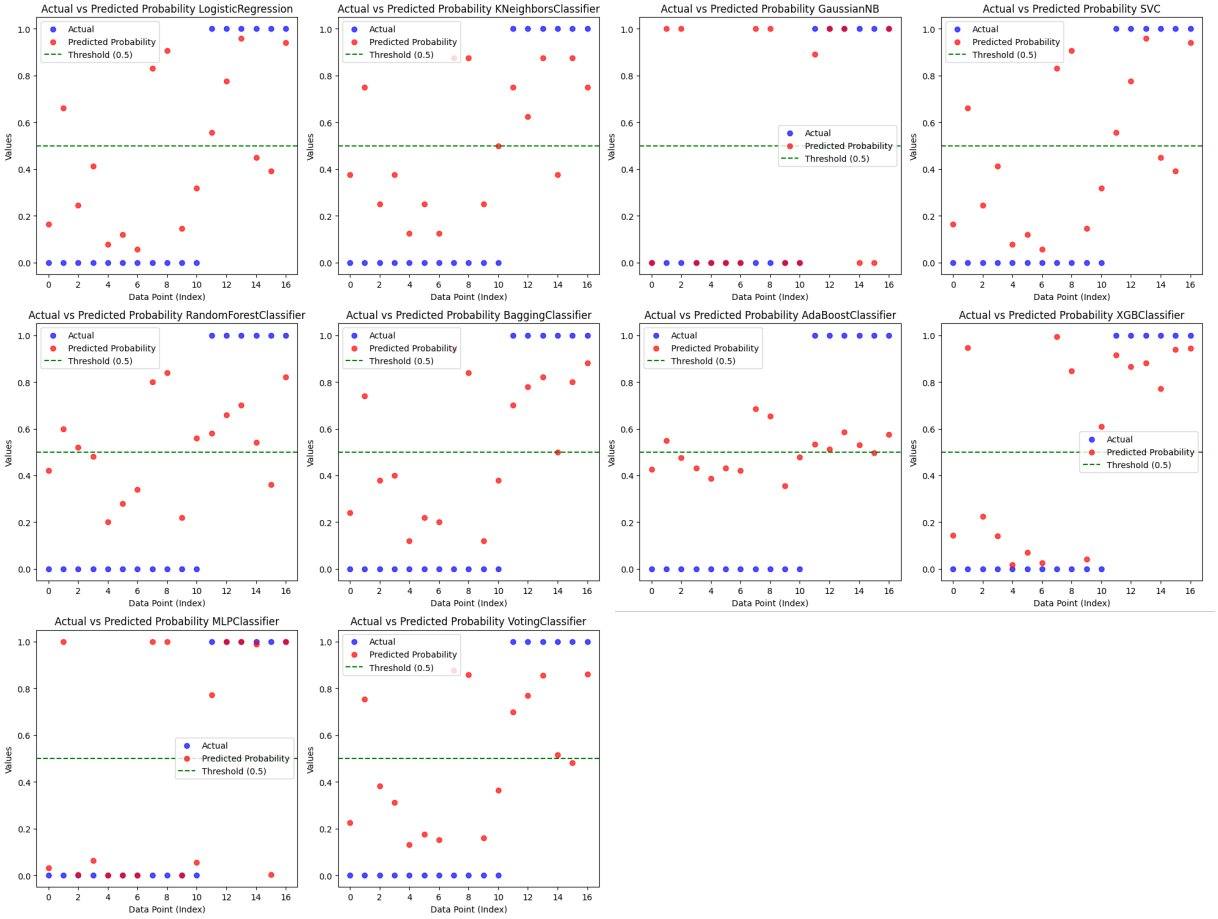

Figure A.16: Comparison of predictions and actual values using Hubert features from the VSP-PDHC dataset