Dissertation on

Early detection of Parkinson's disease using machine learning

Thesis submitted towards partial fulfilment
of the requirements for the degree of

Master of Technology in IT (Courseware Engineering)

Submitted by
Biki Samanta

EXAMINATION ROLL NO.: M4CWE24015
UNIVERSITY REGISTRATION NO.: 160372 of 2021-22

Under the guidance of
Mr. Joydeep Mukherjee

School of Education Technology
Jadavpur University

Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata-700032
India
2024

M.Tech. IT (Courseware Engineering)
Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata, India

CERTIFICATE OF RECOMMENDATION

This is to certify that the thesis entitled **"Early detection of Parkinson's disease using machine learning"** is a bonafide work carried out by **Biki Samanta** under our supervision and guidance for partial fulfillment of the requirements for the degree of Master in Multimedia Development in School of Education Technology, during the academic session 2023-2024.

SUPERVISOR
School of Education Technology
Jadavpur University,
Kolkata-700 032

DIRECTOR
School of Education Technology
Jadavpur University,
Kolkata-700 032

DEAN - FISLM
Jadavpur University,
Kolkata-700 032

M.Tech. IT (Courseware Engineering)
Course affiliated to
Faculty of Engineering and Technology
Jadavpur University
Kolkata, India

CERTIFICATE OF APPROVAL **

This foregoing thesis is hereby approved as a credible study of an engineering subject carried out and presented in a manner satisfactory to warranty its acceptance as a prerequisite to the degree for which it has been submitted. It is understood that by this approval the undersigned does not endorse or approve any statement made or opinion expressed or conclusion drawn therein but approves the thesis only for the purpose for which it has been submitted.

Committee of final examination for evaluation of the Thesis ----------

---

** Only in case the thesis is approved**.**

DECLARATION OF ORIGINALITY AND COMPLIANCE OF ACADEMIC ETHICS

I hereby declare that this thesis contains literature survey and original research work by the undersigned candidate, as part of his **Master of Technology in IT (Courseware Engineering)** studies.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by this rule and conduct, I have fully cited and referenced all materials and results that are not original to this work.

NAME	:	BIKI SAMANTA
EXAMINATION ROLL NUMBER	:	M4CWE24015
REGISTRATION NUMBER	:	160372 of 2021-2022
THESIS TITLE	:	EARLY DETECTION OF PARKINSON'S DISEASE USING MACHINE LAERNING.

SIGNATURE: DATE:

Acknowledgement

I feel fortunate while presenting this dissertation at **the School of Education Technology, Jadavpur University, Kolkata,** in the partial fulfilment of the requirement for the degree of **M.Tech in Information Technology(Courseware Engineering).**

I hereby take this opportunity to show my gratitude towards my mentor **Mr. Joydeep Mukherjee,** who has guided and helped me with all possible suggestions, support, aspiring advice, and constructive criticism along with illuminating views on different issues of this dissertation which helped me throughout my work.

Besides my guide, I would like to thank **Prof.(Dr.)Matangini Chattopadhyay,** Director of School of Education Technology and **Dr. Saswati Mukherjee,** for their support, encouragement and timely advice I do wish to thank all the departme-ntal support staff and everyone else who has different contributions to this dissertation.

Finally, my special gratitude to my parents who have invariably sacrificed and supported me and made me achieve this height.

Date:

Place: Kolkata

Biki Samanta

Examination Roll Number: M4CWE24015

University Registration Number: 160372 of 2021-2022

M.Tech in IT (Courseware Engineering)

School of Education Technology

Jadavpur University

Kolkata:700032

Table of Content

LIST OF ABBREVIATIONS

PD : Parkinson's Disease.

EDA : Exploratory Data Analysis.

SVM: Support Vector Machine.

CH : Correlation Heatmap.

PCA : Principal Component Analysis.

CART : Classification and Regression Tree.

NHR : Noise to Harmonics Ratio.

DFA : Detrended Fluctuation Analysis.

RFE : Recursive Feature Elimination.

AI : Artificial intelligence.

RF : Random Forest.

RDPE : Recurrence Period Density Entropy.

PPE : Pitch Period Entropy

LIST OF FIGURES

LIST OF TABLES

Executive Summary

Parkinson's disease is a neurological condition that progresses over time and mostly affects movement. Parkinson's disease is named for the British physician James Parkinson, who first identified the disorder in 1817. The disease is represented by a range of motor symptoms, include rigidity, tremors, muscle movement (slowness of movement), and postural instability. Non-motor symptoms can also appear as depression, dysfunction of the autonomic nervous system, trouble sleeping, and cognitive impairment.
The primary method of diagnosing Parkinson's disease is based on clinical symptoms because there are yet not conclusive laboratory tests or imaging studies that confirm the diagnosis. Neuroimaging methods like MRI and Data scan, however, can assist in ruling out other illnesses that present with comparable symptoms.

The goals of Parkinson's disease treatment are to lessen symptoms and enhance the lives. The two most often given drugs are levodopa, a precursor to dopamine that can pass the blood-brain barrier and be turned into dopamine, and dopamine agonists, which replicate the actions of dopamine in the brain. The propose automate classification process helps to take prompt therapeutic strategy and subsequent clinical treatment.

The methodology utilizes a range of machine learning classifiers, such as SVM, Random Forest, XGBoost to perform an extensive analysis aimed at early Parkinson's disease identification. Features extraction methodology is used. Finding an appropriate classifier with the best accuracy and fastest execution time is the goal. The dataset is obtained through Kaggle, and a thorough analysis and feature extraction process is being performed.
The approach produces a significant accuracy rate, suggesting potential for predictive power. The method's efficiency in execution time is especially noticeable, which increases its applicability in real-world scenarios. Following research need to concentrate on improving techniques to decrease execution time and enhance accuracy while preserving a higher level of precision.

CHAPTER 1

1. Introduction

1.1 Overview

Parkinson's disease (PD) is a neurological condition that affects millions of older people worldwide and is a major factor in the global rates of disability. In addition to significantly patients and families, the rising incidence of Parkinson's disease also puts a demand on public resources. In order to address this urgent problem, early detection techniques are required in order to enable prompt interventions, which will lessen the worldwide burden of disability. Furthermore, by preserving functional independence, individuals with Parkinson's disease may live longer due to early detection of the disease.

Finding economical and effective ways to forecast Parkinson's disease (PD) with high accuracy becomes essential to achieving early detection. The likelihood of an early diagnosis and intervention can be increased by putting into practice accessible screening techniques.

Parkinson Disease is among the most prevalent neurological conditions. According to Parkinson's disease. There are about 23 instances of Parkinson's disease (PD) for every 120,000 individuals, or 62,000 cases annually, and the average age of start is over 60 years old. According to reports, the prevalence of Parkinson's disease (PD) rises to 1.5% to 3% in 70 years of age and older, from 1.5% in 50 years of age and older. But it's crucial to remember that these figures don't include cases that are still undiagnosed. In the field of Parkinson disease prediction, a lot of research efforts produce encouraging results, but different approaches on use different amounts of data, making it difficult to determine which is the best. Researchers using the same datasets for testing and training may increase performance results, which is a common problem.

1.2 Problem Statement

- Investigate alternate diagnostic modalities that provide effectiveness and affordability.
- Detection of Parkinson Disease at early stage.

1.3 Objectives

- To develop a different diagnostic strategy that reduces the cost burden on people and healthcare systems, making Parkinson's disease diagnosis more accessible and reasonably priced.
- To create algorithms in machine learning to spot Parkinson's disease (PD) in its early stages so that treatment can begin on time.
- To create a diagnostic system that can diagnose patients and start therapy more quickly by reducing down on examination hours. The goal of this objective is to improve patients after treatment outcome result and minimize delays in the process of diagnosis.

1.4 Assumptions and Scopes

1.4.1 Assumptions

- User must have to download the Parkinson patient's dataset for train and testing in Machine learning.
- Developer must have a proper system setup of Python 3.7 or later version, create virtual environment for machine learning and install machine learning Libraries in PC.

1.4.2 Scopes

- To learn about Python, Data Analysis, Machine learning algorithms, Features selection and extraction algorithm.
- To learn about connection of dataset with the environment and how virtual environment works.
- Acquire in-depth-knowledge on utilizing libraries like numpy libraries, pandas' libraries, seaborn libraries and how it works.

1.5 Concepts and Problem Analysis

Artificial intelligence (AI) in the form of machine learning enables software programs to improve prediction accuracy. Machine learning algorithms forecast new values by using past data as input. A neurodegenerative condition affecting the central nervous system is Parkinson's disease. Tremors, stiffness, bradykinesia (slow movement), and postural instability are the hallmarks of the illness. The diagnosis of Parkinson's disease is usually made after a patient's physical examination and medical history are review. However, because the symptoms of Parkinson's disease are frequently mild and might pass for other illnesses, getting a diagnosis early on can be difficult. Machine Learning techniques can help physicians diagnose Parkinson's disease more accurately by giving them quantitative, objective information regarding a patient's symptoms.

A number of issues must be resolved before Machine Learning techniques are extensively applied to Parkinson's disease diagnosis. The requirement for Machine Learning models to be trained on sizable, excellent datasets is one difficulty. Obtaining this might be challenging because Parkinson's disease is a very uncommon condition. This implies that in individuals who have not been included in the training dataset, the Model must be able to correctly diagnose Parkinson's disease. Since Parkinson's disease is a complicated condition with a wide range of manifestations, achieving this can be challenging. In other words, the models must be able to give information.

1.5.1 Exploratory Data Analysis in Python

Exploratory Data Analysis (EDA) is an essential phase in the data analysis process that entails examining, analyzing, and displaying data in order to produce significant insights. It uses statistical techniques and visualizations to identify patterns, trends, and relationships in the data. This aids in the formulation of theories and the direction of further research.

Exploratory Data Analysis (EDA) is the main step in the process of various data analysis. It helps data to visualize the patterns, characteristics, and relationships between variables.

Various exploratory data analysis methods like:

- Reading dataset
- Analyzing the data
- Checking for the duplicates
- Missing Values Calculation
- Exploratory Data Analysis
    - Univariate Analysis
    - Bivariate Analysis
    - Multivariate Analysis

1.5.2 Support vector machine

A supervised machine learning technique that is well known for its effectiveness in classification problems is the Support Vector Machine (SVM). Fundamentally, SVM finds the best hyperplane to maximize the margin between data points from distinct classes and divides them. Even in high-dimensional domains, SVM can reliably identify data points due to this margin optimization technique.

When data cannot be separated linearly, SVM uses a method known as the kernel trick to translate the data into a higher-dimensional space where linear separation is possible. This enables SVM to efficiently classify nonlinear data and manage complex decision boundaries.

1.5.3 Random Forest Algorithm

In the field of machine learning, the Random Forest algorithm is a strong tool known for its predictability and resilience.
With a random subset of the dataset and a random subset of features at each partitioning step, Random Forest constructs a large number of Decision Trees during the training phase. Because of its innate unpredictability, each individual tree is more diverse, which encourages generalization and lowers the possibility of overfitting.

1.5.4 Adaboost

Using a boosting strategy, AdaBoost trains each weak learner one after the other on a reworked dataset. AdaBoost focuses the weak learners on the more difficult data points in each iteration by giving the incorrectly categorized examples from the previous iteration higher weights. By iteratively improving its predictions, AdaBoost's performance can be gradually enhanced through the process of adaptive learning.
AdaBoost uses a weighted voting approach to aggregate all weak learners' predictions during the prediction phase. The final prediction of each weak learner is weighted according to its accuracy; more weights are assigned to models with higher accuracy. AdaBoost creates a strong learner that performs better than any single model by combining the predictions of several weak learners.

1.5.6 Xgboost

The cutting-edge machine learning technique known as XGBoost, or eXtreme Gradient Boosting, is well-known for its quickness, user-friendliness, and remarkable results when applied to big datasets. XGBoost is created by Tianqi Chen and gained popularity due to its performance in several data science contests. It is now a recommended tool for machine learning.

XGBoost is a standout feature due to its exceptional effectiveness and efficiency. With the use of methods like gradient boosting and parallel processing, XGBoost can effectively handle big datasets with millions of samples and features. It is appropriate for real-time applications and situations with constrained computational resources due to its streamlined implementation, which guarantees quick training and prediction times.

A priority in the design of XGBoost is user convenience. XGBoost has reasonable default parameters that perform well on a variety of datasets, in contrast to many machine learning algorithms that necessitate substantial parameter adjustment and optimization to obtain optimal performance. This implies that users do not require additional settings to begin using XGBoost immediately upon installation, making it suitable for both inexperienced and seasoned practitioners.

1.5.7 Feature Extraction Technique

In machine learning and data analysis, feature extraction is essential because it converts unprocessed data into a more interpretable and manageable representation. Two different approaches are used to extract features: features selected by Principal Component Analysis (PCA) are expanded using the Correlation Heatmap (CH) method.

Expanding Features using Correlation Heatmap (CH) Selected by PCA

The most useful characteristics can be found when lowering the dimensionality of the dataset using PCA, a dimensionality reduction technique. Nevertheless, the underlying relationships between variables might not be adequately captured by the features that PCA kept.

The chosen features are expanded using the Correlation Heatmap (CH) technique in order to get over this restriction. Using a heatmap, the pairwise correlations between the retained PCA features are displayed in the Correlation Heatmap approach. Further links and patterns among features can be found by examining the correlation matrix. These correlations are then used to produce new features, expanding the feature space and improving the model's prediction ability.

Significance of Extraction of Features

A crucial preprocessing stage in machine learning, feature extraction affects the effectiveness and interpretability of predictive models. With the help of cutting-edge methods like correlation heatmap analysis and information gain assessment, that will be able to add more meaningful variables to the dataset and improve the model's predictive power.

Correlation heatmap analysis and feature expansion based on information gain are integrated with feature extraction techniques like PCA to emphasize the significance of extracting pertinent information from the dataset while reducing the effects of dimensionality. These methods help to assisting in the creation of reliable and accurate forecasting models.

1.6 Organization of Thesis

(i). Chapter 1 – This Chapter contains the introduction of thesis which includes overview, problem statement, Objectives, assumptions and scope, background concept.

(ii). Chapter 2 – This chapter contains the literature review to carry out the research work.

(iii). Chapter 3 –This chapter contains proposed approach, dataset description, data analysis concept that used to detect Parkinson disease.

(iv). Chapter 4 - This chapter includes Results and Analysis which shows results comes out from different machine learning algorithm.

(v). Chapter 5 – This chapter contains Comparative Analysis which includes Result after PCA algorithm implementation and Result after existing model implementation.

(vi). Chapter 6 – This chapter describe about the Conclusion of the work.

(vii). References – All the references are given here.

(viii). Appendix – All the code snippets are provided in this section.

CHAPTER 2

2. Literature Review

Karapinar, Senturk [1], presented an innovative approach to using Recursive Feature Elimination(RFE)and Feature Importance (FI) features selection algorithm for the determination of the most relevant features to be used in the classification task and gave special attention to elements that are taken out of the midst of vowels. According to this method, specific sounds captured during vowel pronunciation may offer crucial information for differentiating between different of types. Support vector machine(SVM) models with Recursive features Elimination, for classification and Regression Tree with Features Importance feature selection method Used to determine Parkinson patient or not. This work shows how important it is to use phonetic elements when categorizing speech and how useful acoustic analysis may be for tasks involving natural language processing.

Benba et al[2], suggested a method for identifying Parkinson's disease patients by analyzing speech abnormalities. By figuring out the mean squared values of sound, the audio data and successfully extract audio features from each sound sample able to compress. Several vector types in conjunction with Support Vector Machines (SVM) used to distinguish between patients with Parkinson's disease and healthy persons. The SVM with linear kernels, notably, had the best classification accuracy, reaching 91%. This work demonstrates the effectiveness of using SVM in conjunction with feature extraction techniques to accurately diagnose Parkinson's disease based on speech issues, highlighting the potential benefits of this approach for enhanced medical screening and evaluation procedures.

Li et al [3], proposed a novel strategy that blends the Classification and Regression Tree (CART) algorithm with a community learning algorithm, presenting a classification algorithm based on Parkinson's illness. First, speech samples with strong discriminability choose iteratively using the CART method. After that, a community learning algorithm is created by combining Extreme Learning Machine (ELM), Support Vector Machines (SVM), and Random Forest (RF), all of which trained using optimal training instances. Through the use of this combination strategy, which incorporates community learning, RF, and CART algorithms, suggested algorithm is able to attain an 86% average classification accuracy. This illustrates how combining of various machine learning algorithms might improve the classification accuracy of Parkinson's illness. This method advances medical screening and diagnosis procedures by

enhancing diagnostic accuracy and utilizing the advantages of various algorithms in addition to optimizing feature selection.

Vadovsky et al[4], created decision tree models within the R Studio interface, a variety of decision tree algorithms used, such as C4.5, C5.0, Random Forest, and CART. To identify the best accurate classification strategy, these decision tree models used to categorize individual voice signals. According to the findings, the decision tree model performed best on the dataset comprising data from several people, with a mean accuracy of 66%. This result implies that decision tree algorithms can successfully identify speech signals for Parkinson's disease diagnosis, especially when applied in combination. To investigate optimization strategies or different algorithms that might be able to further increase classification accuracy, more investigation may be necessary.

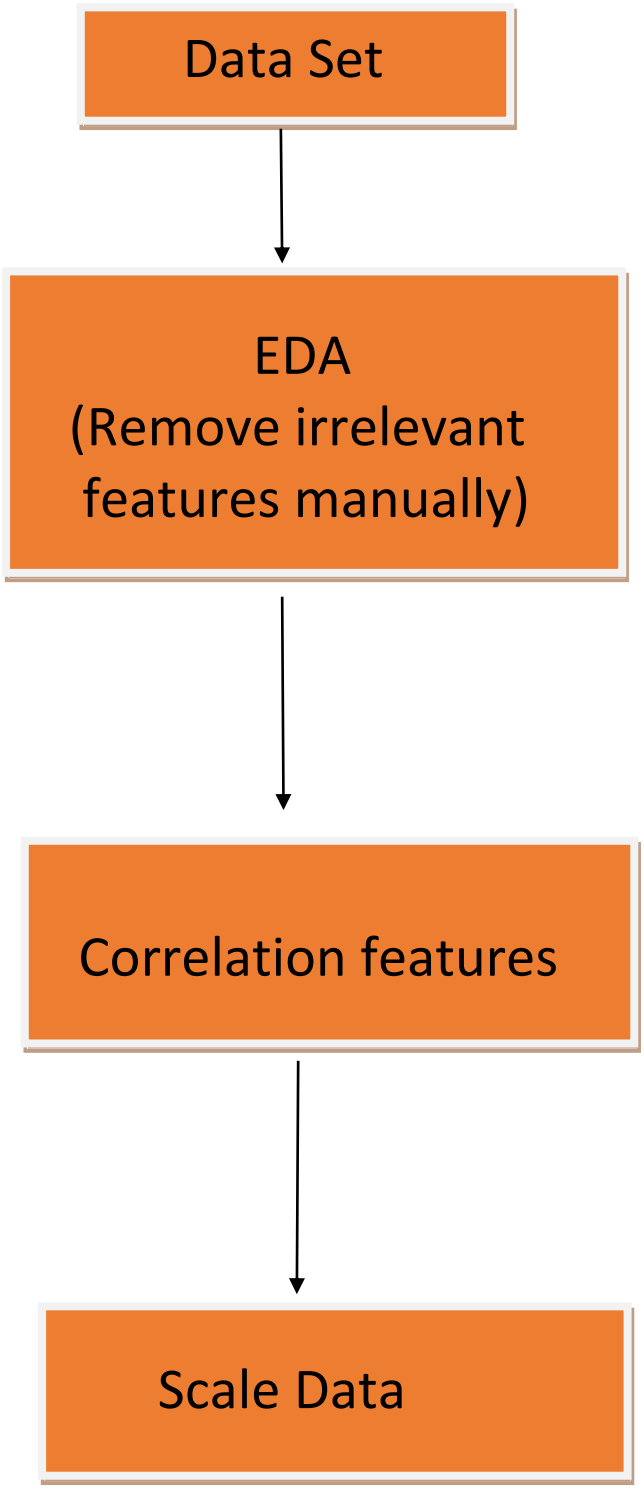
Linear Discriminant Analysis (LDA) is used in a study by Su et al[5], to distinguish sound samples from Parkinson's sufferers and healthy persons. Improving feature selection's ability to discriminate between the two groups of disease and health person is the main goal. The study found that using dynamic feature selection improved classification accuracy over using all of the specified features. This result emphasizes how crucial it is to use discriminant analysis methods, such as LDA, throughout the feature selection process in order to find the most relevant characteristics for class distinction. LDA improves classification accuracy and expands the potential of sound-based analysis for Parkinson's disease diagnosis by concentrating on the most important aspects.

A new method called the Deep Multi-Layer Sensor (DMLP) classifier is presented in a study by Wan et al. [6] for behaviour analysis intended to forecast the course of Parkinson's disease using data from smartphones. This study examined speech and movement patterns recorded by smartphone accelerometers at different times during the day in order to assess the intensity of actions performed by people with Parkinson’s disease.

The suggested approach combined the use of linear regression with well-known machine learning techniques, including M5P, DMLP, Random Forests, and k-Nearest Neighbors (k-NN). The work aims to utilize the complementary strengths of various machine learning techniques, such as DMLP and classical regression, to achieve more accurate prediction of Parkinson's disease development by integrating them.

CHAPTER 3

3. Proposed Approach



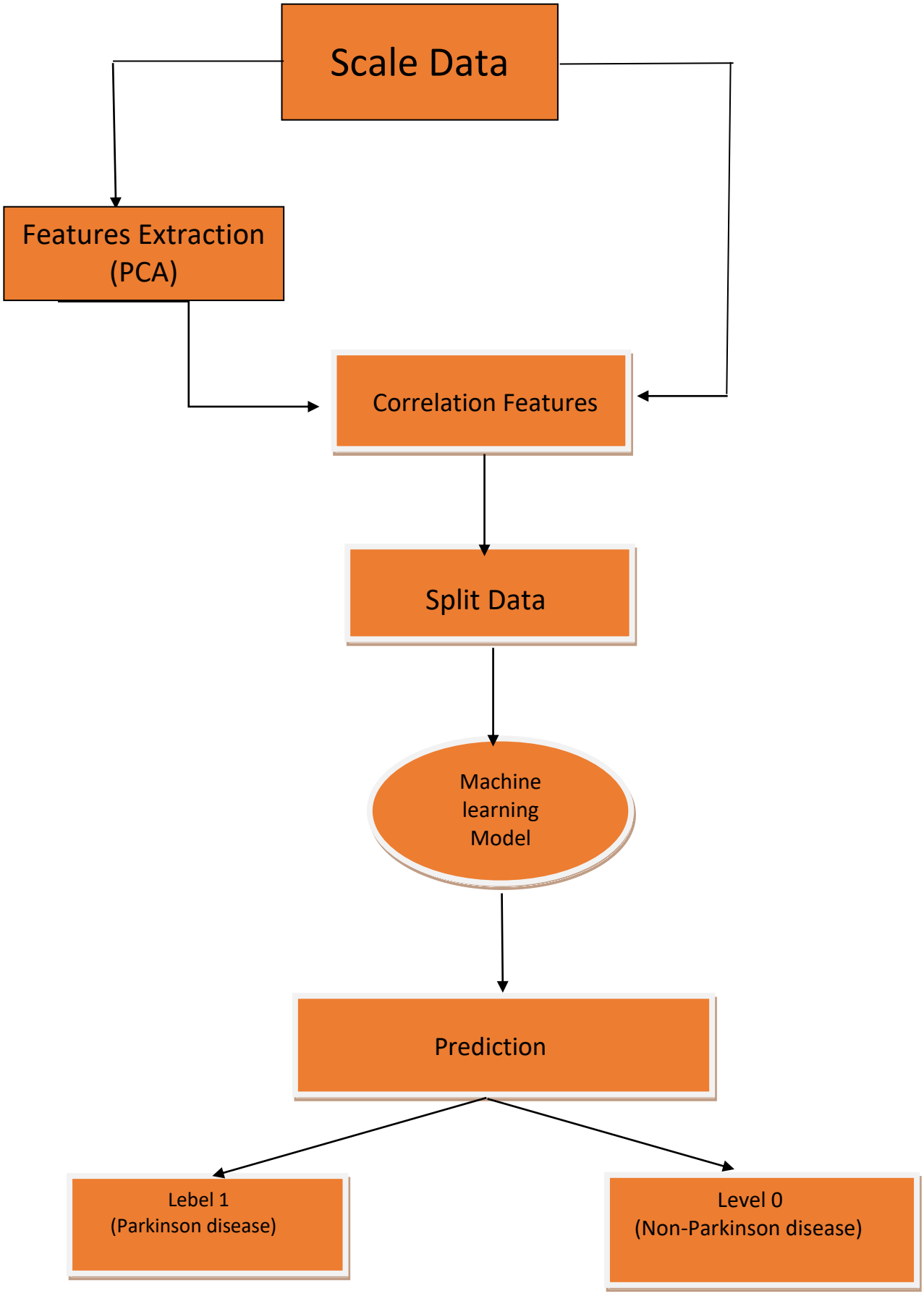


Fig.1 : Diagram of proposed Approach

3.1 Dataset Description

Traditional diagnosis of Parkinson's Disease involves a patient taking a neurological history of the patient and observing motor skills in various situations. Since there is no definitive laboratory test to diagnose PD, diagnosis is often difficult, particularly in the early stages when motor effects are not yet severe. Monitoring progression of the disease over time requires repeated clinic visits by the patient. An effective screening process, particularly one that doesn't require a clinic visit, would be beneficial. Since PD patients exhibit characteristic vocal features, voice recordings are a useful and non-invasive tool for diagnosis. If machine learning algorithms could be applied to a voice recording dataset to accurately diagnosis PD, this would be an effective screening step prior to an appointment with a patient.

Table 1 : features name and definition

Features Name	Features Description
name	ASCII subject name and recording number
MDVP:Fo(Hz)	Average vocal fundamental frequency
MDVP:Fhi(Hz)	Maximum vocal fundamental frequency
MDVP:Flo(Hz)	Minimum vocal fundamental frequency
MDVP:Jitter(%),MDVP:Jitter(Abs),MDVP:RAP,MDVP:PPQ,Jitter:DDP	Several measures of variation in fundamental frequency
MDVP:Shimmer,MDVP:Shimmer(dB),Shimmer:APQ3,Shimmer:APQ5,MDVP:APQ,Shimmer:DDA	Several measures of variation in amplitude
NHR,HNR	Two measures of ratio of noise to tonal components in the voice
status	Health status of the subject (one) - Parkinson's, (zero) – healthy
RPDE,D2	Two nonlinear dynamical complexity measures

DFA	Signal fractal scaling exponent
spread1,spread2,PPE	Three nonlinear measures of fundamental frequency variation

3.1.1 MDVP:Fo(Hz)

It denotes a specific acoustic characteristic taken from the voice signal. It specifically indicates the average voice fundamental frequency (F0), which is measured in Hertz. The fundamental frequency is the lowest frequency component of a periodic waveform, corresponding to the perceived pitch of a voice. Individuals with Parkinson's disease may have changes in vocal features, including variations in fundamental frequency, as a result of motor deficits affecting the laryngeal muscles and vocal folds.

"MDVP:Fo(Hz)" feature into analysis to improve the accuracy of Parkinson's disease (PD) detection. If the MDVP:Fo(Hz) score is less than 145 corresponding to other parameter that shows in the lower section, this suggests a trait linked with Parkinson's disease patients. Below image in X-axis represent ‘status’ and in Y-axis represent ‘MDVP:Fho(Hz)’.This extra criterion serves as a threshold for identifying individuals with a decreased average vocal fundamental frequency, which is a common finding in Parkinson's disease patients. By incorporating this criterion to improve the precision and reliability of PD classification model based on speech data.

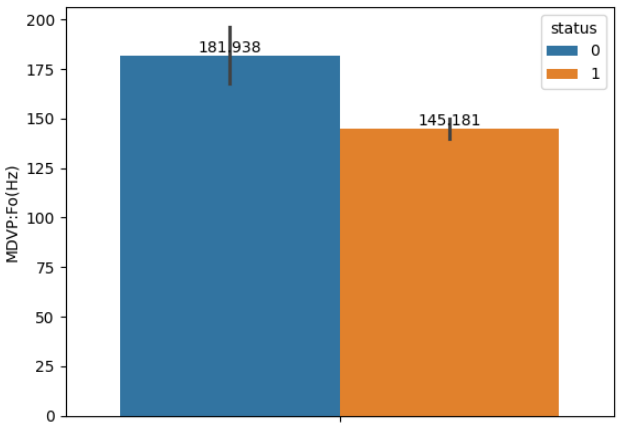


Fig. 2 : Bar chart of MDVP:Fo(HZ)
SOURCE: Self Recorded

3.2.2 MDVP:Fhi(Hz)

It denotes a specific acoustic characteristic taken from the voice signal. This attribute denotes the highest voice fundamental frequency (F0) measured in Hertz.

The fundamental frequency is the lowest frequency component of a periodic waveform, corresponding to the perceived pitch of a voice. Individuals with Parkinson's disease may have changes in vocal features, including variations in fundamental frequency, as a result of motor deficits affecting the laryngeal muscles and vocal folds.

This extra criterion establishes a threshold for identifying persons having a reduced maximum voice fundamental frequency, as is frequent in Parkinson's disease patients. Below image in X-axis represent status and in Y-axis represent MDVP:Fhi(Hz).

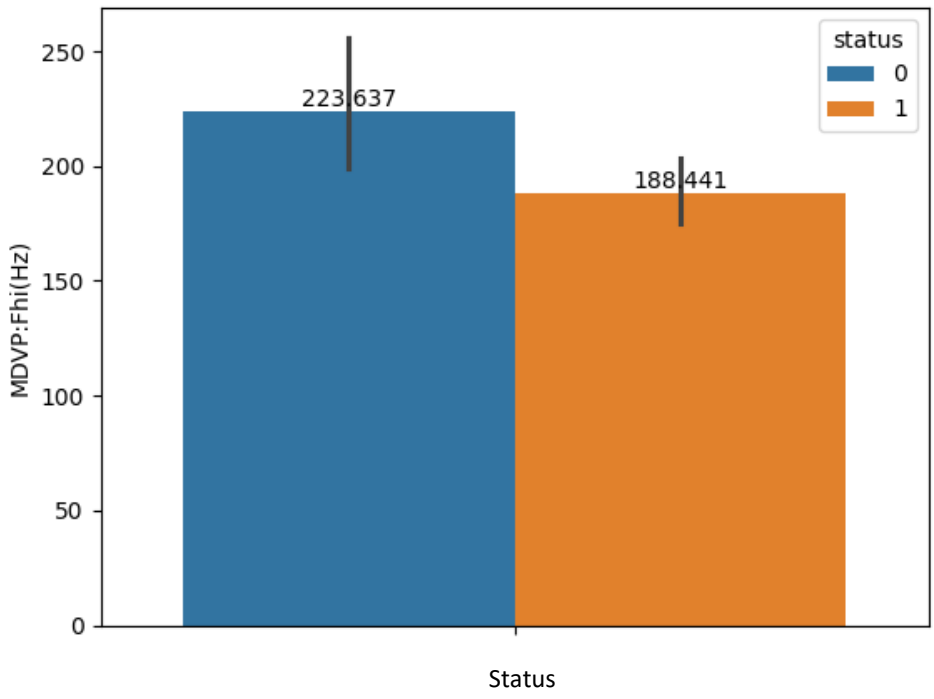


Fig. 3 : Bar chart of MDVP:Fhi(HZ)
SOURCE: Self Recorded

3.1.3 MDVP:Flo(Hz)

The precision with which Parkinson's disease (PD) is detected is dependent on the value of MDVP:Flo(Hz). If it is less than 110, it suggests a trait link with Parkinson's disease.

This additional criterion creates a threshold for identifying persons with a lower minimum voice fundamental frequency, which is common in people with Parkinson's disease. By include this criterion in research to improve the precision and reliability of voice-based PD classification model. Following thorough, preprocessing and feature selection, a number of machine learning methods for categorization, including Support Vector Machines (SVM), Random Forest, and Neural Networks. The addition of the MDVP:Flo(Hz) threshold broadens feature collection, increasing the discriminative capability of model Using this criterion, as well as other pertinent features and advanced machine learning algorithms, methodology aims to achieve accurate and reliable Parkinson's disease identification with voice data. This technique has the potential to enable early diagnosis and intervention, ultimately leading to better patient care and results.

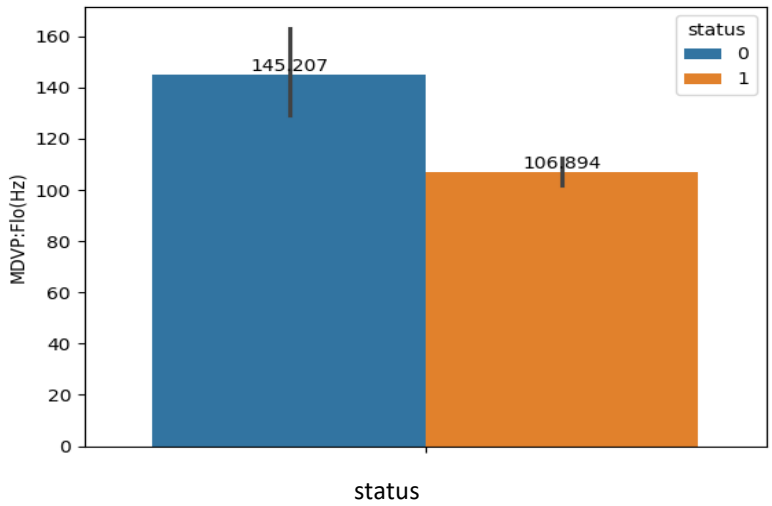


Fig. 4 : Bar chart of MDVP:Flo(HZ)

SOURCE: Self Recorded

3.1.4 MDVP:Shimmer,MDVP:Shimmer(dB),Shimmer:APQ3,Shimmer:APQ5,MDVP:APQ,Shimmer:DDA :

Incorporating criteria for several measures of amplitude variation, such as MDVP:Shimmer, MDVP:Shimmer(dB), Shimmer:APQ3, Shimmer:APQ5, MDVP:APQ, and Shimmer:DDA, into suggested technique improves the accuracy of Parkinson's disease (PD) identification using voice data.
Specifically, if the following criteria are satisfied:
MDVP-Shimmer has a value of 0.017.
The MDVP:Shimmer(dB) value is less than 0.162.
Shimmer:APQ3 is higher than 0.0176.
Shimmer:APQ5 is higher than 0.02.
MDVP:APQ > 0.062, whereas Shimmer:DDA = 0.05.

Then the patient is diagnosed with Parkinson's disease. These criteria establish thresholds for identifying people's vocal characteristics show certain patterns associated with Parkinson's disease. By incorporating these parameters to improve the precision and reliability PD classification model based on voice data.
Following painstaking preprocessing and feature selection stages, use a variety of machine learning methods for categorization, including Support Vector Machines (SVM), Random Forest, and Neural Networks.

3.1.5 NHR

In the context of voice analysis for Parkinson's disease (PD) identification, NHR stands for "Noise to Harmonics Ratio." It measures the ratio of noise to tone components in the voice signal. NHR estimates the proportion of non-harmonic components, which are considered noise, vs harmonic components, which constitute the tonal characteristics of the voice.
When the NHR score is larger than 0.029, it shows that there is a higher proportion of noise in the voice signal compared to tone components. This can indicate voice abnormalities or disruptions, which are common in people with Parkinson's disease. Individuals with NHR readings above this threshold may be classed as PD patients.
When NHR is fewer than 21, it indicates a decreased noise-to-tone ratio in the voice transmission. This demonstrates a more balanced distribution of harmonic and non-harmonic components, which is typical of healthy vocal patterns.
In summary, NHR is a quantitative assessment of the balance of noise and tone components in the voice stream. Setting thresholds based on NHR values allows us to identify people's speech features show specific patterns associated with Parkinson's

illness. Integrating NHR criteria into improves the accuracy and reliability of PD classification models based on voice data, resulting in better diagnostic outcomes and patient treatment.

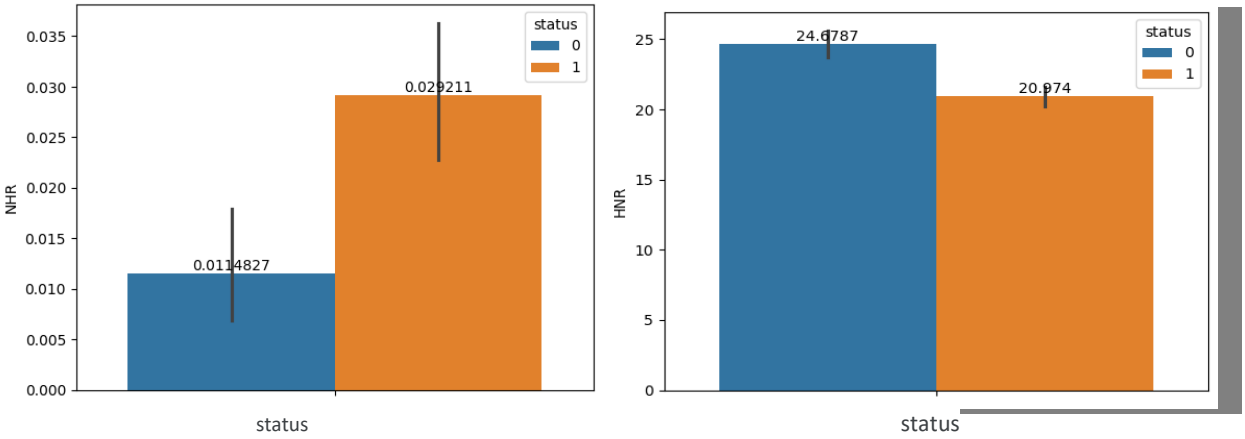


Fig. 5 : Bar chart of NHR AND HNR
SOURCE: Self Recorded

3.1.6 RPDE, D2

RPDE quantifies the density of recurrence points in a dynamical system's reconstruct phase space. It represents the consistency or predictability of voice signal dynamics. D2 is a measure of the fractal dimension or complexity of the reconstructed phase space trajectory. It describes the level of irregularity or complexity in voice signal dynamics.

When assessing these measures for Parkinson's Disease detection: If the RPDE number is larger than 0.516, it suggests a higher density of recurrence points, which may result in more irregular or unpredictable voice dynamics. A D2 value greater than 2.5 indicates a higher fractal dimensionality or complexity in voice signal dynamics.

Individuals with RPDE scores is0.516 and D2 values is 2.5 may have Parkinson's disease-related vocal features. These people may be diagnosed with Parkinson's disease because vocal dynamics are more irregular, unpredictable, or complex than healthy people.

Including RPDE and D2 measurements to improve the accuracy and reliability of PD classification models based on speech data. By establishing thresholds based on these metrics to identify people's speech features show certain patterns linked with Parkinson's disease, resulting in better diagnostic outcomes and patient management.

3.1.7 DFA

In the context of Parkinson's disease (PD) identification, DFA (Detrended Fluctuation Analysis) evaluates the voice signal's fractal scaling exponent. Specifically, DFA measures the signal's long-range correlation qualities, indicating the underlying self-similarity or fractal structure in the data. The signal fractal scaling exponent obtained using DFA represents the degree of correlation between different segments of the speech signal over different time scales. A larger fractal scaling exponent indicates more long-range correlations or signal persistence, whereas a lower exponent implies weaker correlations or more random fluctuations.

The DFA value offers information about the complexity and regularity of the voice signal dynamics. Changes in DFA values may indicate changes in the underlying physiological processes linked with Parkinson's disease. Analyzing DFA of voice signals enables researchers to identify small changes in vocal qualities that may be early markers of Parkinson's disease or beneficial for disease monitoring.

3.1.8 Spread1, Spread2, PPE

Three nonlinear metrics used to analyse fundamental frequency variation in the context of speech analysis for Parkinson's disease (PD) identification.

Spread1 : Spread1 is a nonlinear measure that quantifies the distribution of fundamental frequency values in a voice stream. It measures the variation in pitch or vocal frequency across distinct portions of the signal.

Spread2 : Spread2, like spread1, analyses the spread or dispersion of fundamental frequency values in a voice transmission. Spread2 may use a different mathematical formulation or approach to measure this variance than spread1.

PPE : PPE is a nonlinear measure that describes the irregularity or unpredictability of the fundamental frequency pattern in a voice stream. It evaluates the entropy or disorderliness of pitch periods, which reflects the level of variety or complexity in vocal frequency modulation.

3.2 Data-Analysis

3.2.1 Data check

The data collect online is very clean, which means there is no missing numbers or problematic bits that need to be fix. Because of this, do not have to spend time cleaning up the data before beginning the analysis.

3.2.2 Redundant Data

Recognizing that redundant features might offer little benefit to the prediction model and may generate unneeded noise, take proactive steps to address this issue. So, in data name column has no relevance, so delete this column and, investigate data distribution across features using visualization approaches such as bar charts, count plots, and histograms.

This method helps to detect any patterns or abnormalities and make informed decisions about which features to keep or discard in the future modelling process. By deleting superfluous columns that ensures that only the most important and informative variables are included in the study, which improve the model's predictive power.

3.2.3 Plotting Histogram

Histograms proved very effective for displaying the distribution of specific attributes. A histogram is a graphical depiction of the frequency distribution of data values for a particular variable. By evaluating the shape, centre and spread of each histogram, learn about the underlying distribution of feature values and probable outliers.

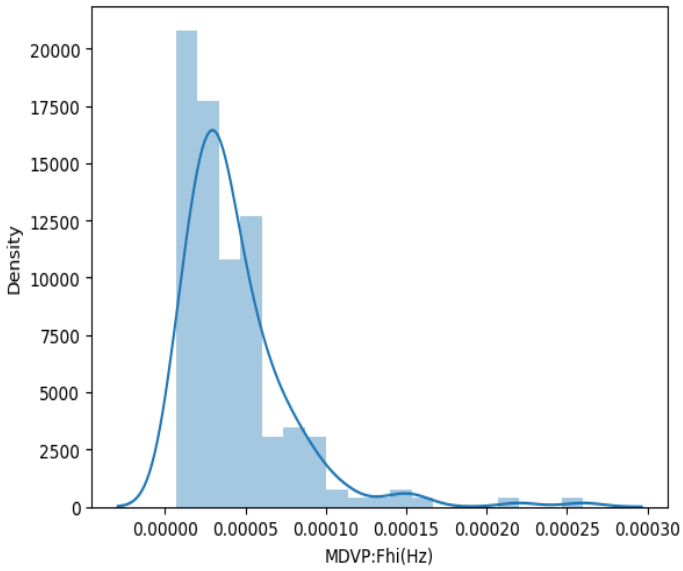


Fig. 6 : HISTOGRAM of MDVP:Fhi(HZ)

SOURCE: Self Recorded

The histogram consists of vertical bars. On the x-axis, each bar represents the number of data points that fall within a certain interval (bin). The height of each bar represents the frequency or density of data points inside that interval. The smooth curve placed over the histogram denotes a probability density function (PDF) or a kernel density estimate. It estimates the underlying continuous distribution of the data. The data appears to be skewed to the right (positively skewed), as the histogram bars are larger on the left side. The line graph indicates a peak at a given value, followed by a slow drop.

Overall, this image helps us understand how the variable "MDVP:Fhi(Hz)" is distributed in the dataset.

3.2.4 Correlation Analysis

It created a correlation matrix to measure the associations between attributes and the target variable. Several features, including Jitter:DDP, MDVP:APQ, MDVP:Jitter(Abs), MDVP:PPQ, MDVP:RAP, MDVP:Shimmer, MDVP:Shimmer(dB), NHR, PPE, Shimmer:APQ3, Shimmer:APQ5, Shimmer:DDA, and spread1, exhibited significant correlations with the target variable.

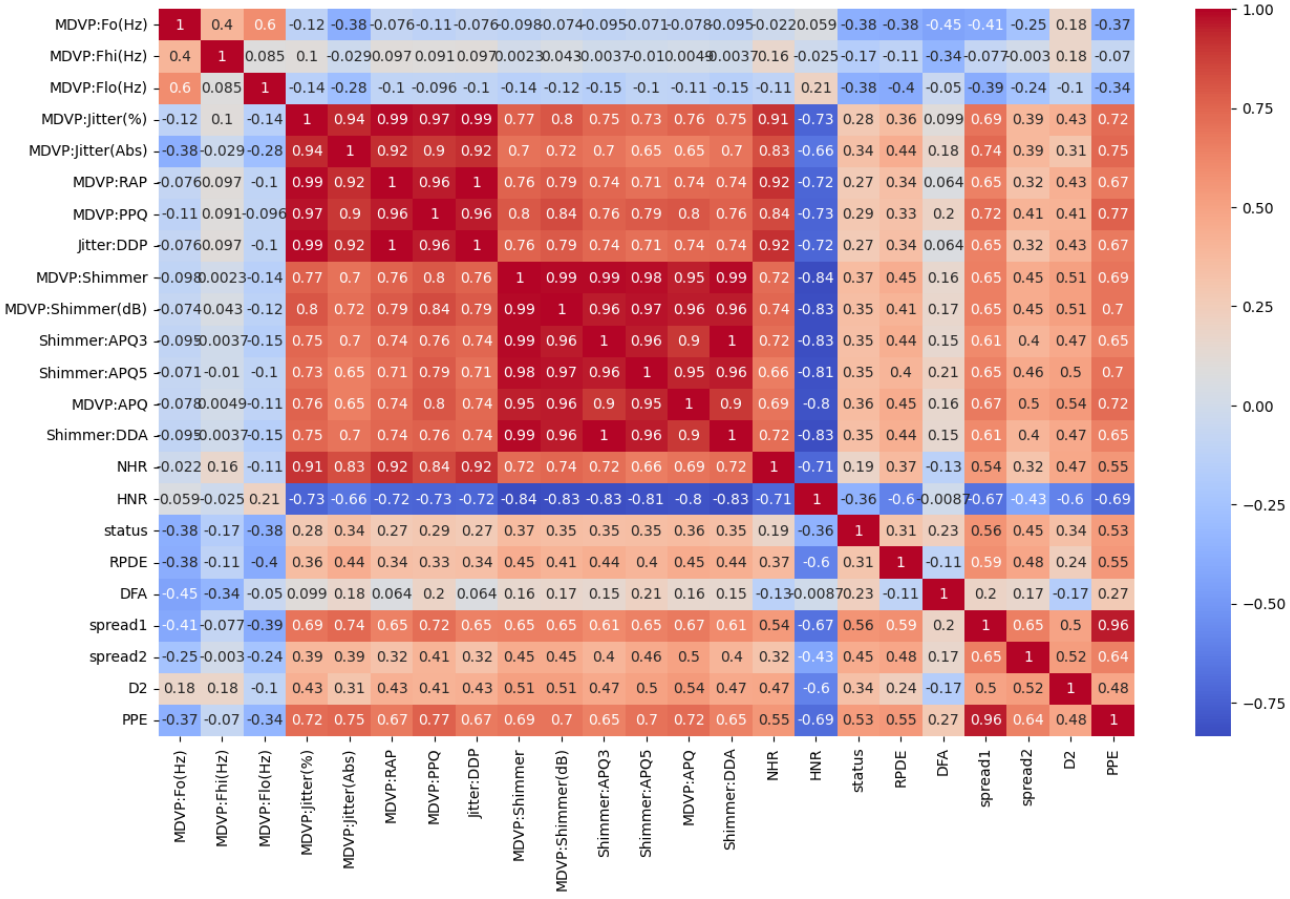


Fig. 7 : CORRELATION MATRIX OF ALL FEATURES
SOURCE: Self Recorded

3.3 Feature Selection Process

To improve modelling performance, the dataset's dimensionality is reduced using Principal Component Analysis (PCA). Initially, a collection of five characteristics is chosen for dimensionality reduction. Following that, a PCA Data Frame is created and use to apply PCA for the feature matrix (X data). After that, the dataset is divided into training and testing sets for model evaluation. To obtain insight into the correlations between variables, a correlation matrix is created. Following that, a range of machine learning algorithms, including Support Vector Machine (SVM), Random Forest, and XGBoost, is used to train and test prediction models on the dataset. This comprehensive approach enables the investigation of several modelling strategies and performance in predicting the target variable. A correlation threshold of 0.8 is established to identify the most highly correlated features. Features exhibiting both positive and negative correlations surpassing this threshold are utilized to expand the original feature space. This expansion involved creating new features by summing the values of correlated features.A correlation heatmap depicting the relationships between features in the Parkinson's dataset is presented in.

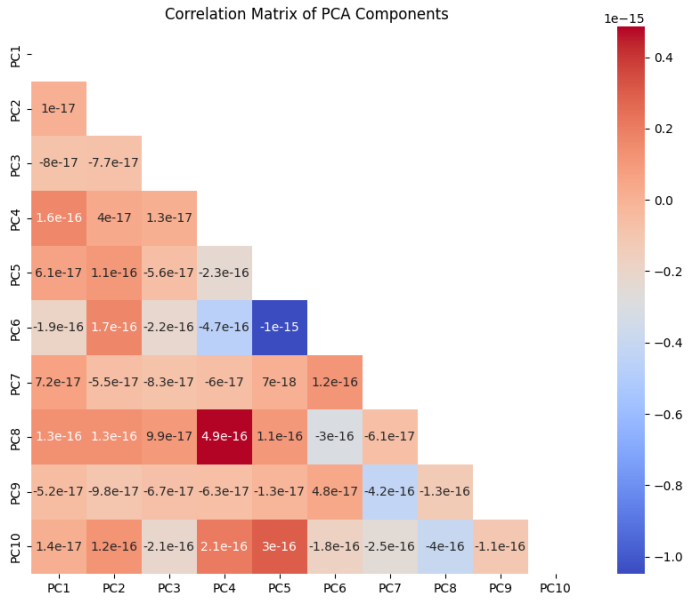


Fig. 8: CORRELATION MATRIX OF PCA FEATURES

SOURCE: Self Recorded

The correlation matrix of the PCA components sheds light on the interactions between distinct main components. During examine, it is clear that the bulk of principal component pairs have low correlation values, nearing zero. This discovery is consistent with the core goal of PCA, which is to turn correlated variables into uncorrelated ones.

The occurrence of low correlation values across principal components indicates that PCA successfully transform the original dataset into a series of orthogonal components, each capturing a unique feature of the data's variability. This result highlights the usefulness of PCA in reducing the data structure and aiding further analysis and interpretation. The diagonal elements of the correlation matrix indicate the self-correlations of each principal component, from PC1 to PC10. Notably, all diagonal elements have values that are very near to zero. This discovery implies that each primary component is not associated with itself. In other words, there is no direct relationship or association between the values of each primary component. This feature emphasizes the orthogonality of the principal components, which means capture separate and independent dimensions of variability within the dataset. The presence of near-zero self-correlations demonstrates PCA's efficiency in minimizing multicollinearity and simplifying the data structure for later analysis.

The off-diagonal elements of the correlation matrix represent the pairwise correlations between various principal components. Off-diagonal correlations are close to zero, with many falling within the order of 10^-16. The near-zero correlation between primary components implies that are effectively uncorrelated with one another. In other words, the values of one major component have a minimal linear connection with another. This absence of correlation emphasizes the orthogonality of the main components, which means that each component captures distinct and independent dimensions of variation within the dataset. The absence of substantial off-diagonal correlations underlines the success of PCA in translating the original variables into a set of orthogonal components, allowing for a better understanding of the fundamental structure of the data.

CHAPTER 4

4. Results and Analysis

By using Recursive Feature Elimination (RFE), the model's accuracy determines to be 93.84%. However, using Principal Component Analysis (PCA) for feature extraction the accuracy is 95% while also significantly reducing computation time.

PCA is a dimensionality reduction approach that converts the original features to a lower-dimensional space while retaining the maximum variance in the data. By preserving only the most informative components, PCA lowers dataset redundancy and noise, resulting in better model performance.

In this scenario, PCA most likely detect the most relevant features while excluding the less important ones, resulting in a more efficient and effective data representation. The higher accuracy indicates that the decreased feature set caught the underlying structure of the data more effectively, resulting in superior prediction performance.

Furthermore, the lower dimensionality allowed for faster calculation because fewer characteristics are used in the modelling process. This savings in calculation time is especially useful for huge datasets or real-time applications where speed is critical.

Overall, the use of PCA for feature extraction results in improved accuracy and shorter computation time, making it a promising strategy for improving model performance.

TABLE 2：classification accuracies of different features set

Technique	precision	F1-score	Accuracy(%)	Recall	Speed (millisec)
SVM	91	96	92	100	1.34
XGBoost	94	97	95	100	1.77

CHAPTER 5

5. Comparative Analysis

5.1 Result after PCA Algorithm Implementation

Table 3: Result after PCA Algorithm Implementation

Technique	precision	F1-score	Accuracy(%)	Recall	Speed (millisec)
SVM	91	96	92	97	1.34
XGBoost	94	97	95	100	1.77

5.2 Result after Existing model Implementation

Table 4 : Result after Existing model Implementation

Technique	precision	F1-score	Accuracy(%)	Recall	Speed (millisec)
SVM	90	94	90	98	1.73
XGBoost	92	97	93.84	98	1.91

Table 3 shows XGBoost attains an accuracy of 95%, while SVM's accuracy is just 92%. This suggests that a larger percentage of the test dataset's instances are correctly predicted by XGBoost overall. The precision of SVM is 91%, whereas that of XGBoost is 94%. The precision of the model is determined by dividing all of its positive predictions by the percentage of true positive forecasts. As a result, XGBoost outperforms SVM in preventing false positives by a little margin.

SVM achieves a recall of 97%, whereas XGBoost achieves a 100% recall. The percentage of accurate positive predictions among all real positive cases in the test dataset is measured by recall. While SVM misses 3% of positive occurrences, XGBoost has a 100% recall, meaning it never misses a single positive instance.

Existing Model :

The existing model's RFE method clearly chooses features according to significance, which may provide light on which features are most pertinent to the classification objective.

XGBoost is renowned for its capacity to manage intricate relationships in data and frequently exhibits strong performance in the absence of rigorous feature selection or preprocessing. However, because of its explicit feature selection method and linear decision bounds, Support Vector Machines with RFE may be easier to understand.

From Table 4 shows the SVM classifier gets an accuracy of 90%, XGBoost obtains 94% accuracy. This suggests that a larger percentage of the test dataset's instances are correctly predicted overall by the XGBoost model.

The SVM classifier has a precision of 91%, but XGBoost has a precision of 92%. The precision of the model is determined by dividing all of its positive predictions by the percentage of true positive forecasts. As a result, XGBoost outperforms the SVM classifier in preventing false positives by a little margin.

98% recall is achieved by both models. The percentage of accurate positive predictions among all real positive cases in the test dataset is measured by recall. With very little difference of XGBoost, both models capture the majority of positive events with comparable effectiveness.

In terms of accuracy and precision, the current model with XGBoost performs better than the suggested model with the SVM classifier. On the other hand, recall performance is similar for both models. As a result, for this specific assignment, the current model with XGBoost may perform better overall based on the metrics given than the suggested model with the SVM classifier.

Chapter 6

6. Conclusion and Futures Scopes

6.1 Conclusion

This research work provides a thorough examination of machine learning methods for Parkinson's disease categorization, resulting in significant insights and potential areas. The landscape of diagnostic accuracy and computational efficiency in Parkinson's disease classification by employing thorough feature selection strategies and rigorously testing several algorithms to findings show that XGBoost is the pinnacle of performance, with an astonishing 95% accuracy while preserving ideal computational efficiency. This highlights the relevance of using advanced optimization techniques and ensemble learning methodologies in medical diagnosis. The success of XGBoost paves the way for the creation of powerful diagnostic tools that can help clinicians reliably diagnose Parkinson's disease at an early stage, allowing for timely intervention and individualized treatment regimens. Furthermore . The study emphasizes the importance of testing a wide range of machine learning algorithms and feature extraction methodologies to find the most successful approaches for disease categorization. Each algorithm has distinct strengths and capabilities.

6.2 Future Scopes

In future research, explore potential for improving diagnostic accuracy and efficiency. To improve classification accuracy, different feature selection or reduction techniques can be investigated in further research. In essence of study not only adds to the growing body of knowledge in Parkinson's disease diagnosis, but it also highlights the revolutionary power of machine learning in healthcare.

References

[1] **Zehra Karapinar Senturk**" **Early diagnosis of Parkinson's disease using** machine learning algorithms" Medical Hypotheses ELSEVIRE,2020.109603

[2] Benba, A., Jilbab, A., Hammouch, A., & Sandabad, S., "Voiceprints analysis using MFCC and SVM for detecting patients with Parkinson's disease", In Electrical and Information Technologies (ICEIT), pp. 300- 304, IEEE, 2015

[3] Li, Yang, L., Wang, P., Zhang, C., Xiao, J., Zhang, Y., & Qiu, M., "Classification of Parkinson's Disease by Decision Tree Based Instance Selection and Ensemble Learning Algorithms", Journal of Medical Imaging and Health Informatics, 7(2), 444-452, 2017

[4] Vadovský, M., & Paralič, J., "Parkinson's disease patients classification based on the speech signals", In Applied Machine Intelligence and Informatics (SAMI), 2017 IEEE 15th International Symposium on pp. 321-326, IEEE, 2017

[5] Su, M., & Chuang, K. S., "Dynamic feature selection for detecting Parkinson's disease through voice signal", In RF Wireless Technologies for Biomedical and Healthcare Applications (IMWS-BIO), 2015 IEEE MTT-S 2015 International Microwave Workshop Series on pp. 148-149, IEEE, 2015

[6] Wan, S., Liang, Y., Zhang, Y., & Guizani, M., "Deep multi-layer perceptron classifier for behavior analysis to estimate Parkinson's disease severity using smartphones", IEEE Access, 6, pp. 36825-36833, 2018

[7] Pavallo F, Moschetti A, Esposito D, Maremmani C, Rovini E. Upper limb motor pre-clinical assessment in Parkinson's disease using machine learning. Park Relat Disord 2019

[8] Almeida JS, et al. Detecting Parkinson's disease with sustained phonation and speech signals using machine learning techniques. Pattern Recogn Lett 2019

[9] Wang Y, Wang AN, Ai Q, Sun HJ. An adaptive kernel-based weighted extreme learning machine approach for effective detection of Parkinson's disease. Biomed Signal Process Control 2017

[10] Yaman O, Ertam F, Tuncer T. Automated Parkinson's disease recognition based on statistical pooling method using acoustic features. Med Hypotheses 2019

[11] Salmanpour MR, et al. Optimized machine learning methods for prediction off cognitive outcome in Parkinson's disease. Comput Biol Med 2019

[12] Sakar, B. E., Isenkul, M. E., Sakar, C. O., Sertbas, A., Gurgen, F., Delil, S., ... & Kursun, O., "Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings", IEEE Journal of Biomedical and Health Informatics, 17(4), 828-834, 2013

[13] Ekinci, E. Omurca, S.I. and Acun, N., "A Comparative Study on Machine Learning Techniques using Titanic Dataset", 7th International Conference on Advanced Technologies, pp. 411-416, 2018.

[14] Meghraoui, D., Boudraa, B., Merazi-Meksen, T., & Boudraa, M., "Parkinson's Disease Recognition by Speech Acoustic Parameters Classification", In Modelling and Implementation of Complex Systems, pp. 165-173, Springer, 2016.

[15] Cavallo F, Moschetti A, Esposito D, Maremmani C, Rovini E. Upper limb motor pre-clinical assessment 2019

[16] Prashanth R, Dutta Roy S. Novel and improved stage estimation in Parkinson's disease using clinical scales and machine learning. Neurocomputing 2018

[17] Wan KR, Maszczyk T, See AAQ, Dauwels J, King NKK. A review on microelectrode recording selection of features for machine learning in deep brain stimulation surgery for Parkinson's disease. Clin Neurophysiol 2019

[18] Benba, A., Jilbab, A., Hammouch, A., & Sandabad, S., "Voiceprints analysis using MFCC and SVM for detecting patients with Parkinson's disease", In Electrical and Information Technologies (ICEIT), pp. 300- 304, IEEE, 2015.

[19] Sakar, B. E., Isenkul, M. E., Sakar, C. O., Sertbas, A., Gurgen, F., Delil, S., ... & Kursun, O., "Collection and analysis of a Parkinson speech dataset with multiple types of sound recordings", IEEE Journal of Biomedical and Health Informatics, 17(4), 828-834, 2013.

[20] Zhu W, Zeng N, Wang N. Sensitivity, specificity, accuracy, associated confidence interval and ROC analysis with practical SAS ® implementations. 2010.

Appendix

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
import time
from xgboost import XGBClassifier
from sklearn.metrics import classification_report
from scipy.stats import pearsonr, spearmanr
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import classification_report
from sklearn.feature_selection import RFE
from sklearn.ensemble import RandomForestClassifier
from sklearn.decomposition import PCA
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from xgboost import XGBClassifier
```

```
#Read the data
df=pd.read_csv('/content/parkinsons.csv')
df.head()# To see the first 5 rows of our dataset we use head()

df.drop(['name'],axis=1,inplace=True)

sns.countplot(x="status",data=df)

ax=sns.countplot(data=df,x="status",y="MDVP:Fo(Hz)",hue="MDVP:Fhi(Hz)")
for bars in ax.containers:
    ax.bar_label(bars)


mdvp_fo = df['MDVP:Fo(Hz)']
mdvp_fhi = df['MDVP:Fhi(Hz)']

pearson_corr, pearson_p_value = pearsonr(mdvp_fo, mdvp_fhi)
spearman_corr, spearman_p_value = spearmanr(mdvp_fo, mdvp_fhi)
```

```
print("Pearson's correlation coefficient:", pearson_corr)
print("Spearman's correlation coefficient:", spearman_corr)

sns.distplot(df["MDVP:Fhi(Hz)"])
df['MDVP:Fo(Hz)']=np.log(df['MDVP:Fo(Hz)']+1)

ax=sns.barplot(y="D2",data=df,hue="status")
for bars in ax.containers:
    height = bars.get_height()
    if height >= threshold:
        ax.bar_label(bars, label='PD', color='white', fontsize=10,
label_type='edge')
    else:
        ax.bar_label(bars, label='Not PD', color='black', fontsize=10,
label_type='edge')

plt.show()
corr = df.corr()
plt.figure(figsize=(15, 9))
sns.heatmap(corr, annot=True, cmap='coolwarm')


def correlation(dataset, threshold):
    col_corr = set()
    corr_matrix = dataset.corr()
    for i in range(len(corr_matrix.columns)):
        for j in range(i):
            if (corr_matrix.iloc[i, j]) > threshold:
                colname = corr_matrix.columns[i]
                col_corr.add(colname)
    return col_corr

corr_features = correlation(df, 0.85)
len(set(corr_features))


X=df.drop(columns="status",axis=1)
Y=df['status']


scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)


X_train, X_test, y_train, y_test = train_test_split(X_scaled, Y,
test_size=0.2, random_state=7)


svm_estimator = SVC(kernel='linear')
selector = RFE(svm_estimator, n_features_to_select=10)
```

```
selector = selector.fit(X_train, y_train)


x_train_selected = selector.transform(X_train)


svm_model = SVC(kernel='linear')
svm_model.fit(x_train_selected, y_train)
x_val_selected = selector.transform(X_test)
y_pred = svm_model.predict(x_val_selected)
print(classification_report(y_test, y_pred))

rfe_estimator = RandomForestClassifier()
selector = RFE(rfe_estimator, n_features_to_select=None)
selector.fit(X_train, y_train)

x_train_transformed = selector.transform(X_train)
x_test_transformed = selector.transform(X_test)


start_train = time.time()

model = XGBClassifier()
model.fit(x_train_transformed, y_train)

end_train = time.time()
training_time = (end_train - start_train) * 1000
print("Training Time:", round(training_time, 2), "milliseconds")
start_pred = time.time()

preds = model.predict(x_test_transformed)
end_pred = time.time()
prediction_time = (end_pred - start_pred) * 1000
print("Prediction Time:", round(prediction_time, 2), "milliseconds")

print(classification_report(y_test, preds))



n_components = 10

# Initialize PCA
pca = PCA(n_components=n_components)
X_pca = pca.fit_transform(X_scaled)

df_pca = pd.DataFrame(X_pca, columns=[f'PC{i+1}' for i in
range(n_components)])

# Calculate the correlation matrix
correlation_matrix = df_pca.corr()
```

```
mask = np.triu(np.ones_like(correlation_matrix, dtype=bool))

# Plot the heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', mask=mask)
plt.show()

X_pca_df = pd.DataFrame(X_pca)
correlation_matrix = X_pca_df.corrwith(df, axis=0).abs()
threshold = 0.7
selected_features = []
for col in correlation_matrix.index:
    highly_correlated_feature_indices = np.where(correlation_matrix[col] >
threshold)[0]
    highly_correlated_features =
df.columns[highly_correlated_feature_indices].tolist()
    selected_features.extend(highly_correlated_features)
extended_features_pca_ch = pd.concat([X_pca_df, df[selected_features]],
axis=1)

extended_features_pca_ch.describe()


# Split the extended features
X_train, X_test, y_train, y_test = train_test_split(extended_features_pca_ch,
Y, test_size=0.2, random_state=7)

xgb_model = XGBClassifier()
xgb_model.fit(X_train, y_train)

y_pred = xgb_model.predict(X_test)

# Evaluate the accuracy of the model
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)


start_train = time.time()

model = XGBClassifier()
model.fit(X_train_pca, y_train_pca)

end_train = time.time()
training_time = (end_train - start_train) * 1000
print("Training Time:", round(training_time, 2), "milliseconds")


start_pred = time.time()

preds = model.predict(X_test_pca)
```

```
# Calculate the time taken for prediction
end_pred = time.time()
prediction_time = (end_pred - start_pred) * 1000
print("Prediction Time:", round(prediction_time, 2), "milliseconds")

# Print classification report
print("XGB")
print(classification_report(y_test, preds))
```

```
Training Time: 33.77 milliseconds
Prediction Time: 1.77 milliseconds
XGB
              precision    recall  f1-score   support

           0       1.00      0.71      0.83         7
           1       0.94      1.00      0.97        32

    accuracy                           0.95        39
   macro avg       0.97      0.86      0.90        39
weighted avg       0.95      0.95      0.95        39
```