Parkinson's Disease Prediction using Logistic Regression

Mini Project Report

Submitted for the Partial Fulfillment of the Requirements

for the Award of the Degree of

Master of Computer Applications

By

MEENAKSHI SAJEEV

MUT23MCA-2044

Under the guidance of

Ms. JISS KURUVILLA

ASSISTANT PROFESSOR

Department of Computer Applications

MUTHOOT INSTITUTE OF TECHNOLOGY & SCIENCE

VARIKOLI P O, PUTHENCRUZ, ERNAKULAM DISTRICT, KERALA

(Affiliated to A P J Abdul Kalam Technological University, Thiruvananthapuram, Kerala)

November – 2024

Muthoot
Institute of Technology & Science

Department of Computer Applications

BONAFIDE CERTIFICATE

This is to certify that the Mini Project Report entitled ***"Parkinson's Disease Prediction using Logistic Regression"*** *has been submitted by* ***Ms Meenakshi Sajeev*** *Reg. No.* ***MUT23MCA-2044*** *for the partial fulfillment of the requirements for the award of the degree of Master of Computer Applications (MCA) of A P J Abdul Kalam Technological University, Kerala during the year 2024.*

Place: Varikoli

Date:

Project Guide	**Project Coordinators**	**HoD**
Ms Jiss Kuruvilla	Dr Smitha Anu Thomas Dr Sujithra Sankar	Dr Saritha K

Submitted for the Final Evaluation held on ……

Name and Signature of the Examiner

DECLARATION

I, undersigned hereby declare that the Mini Project Report ***"Parkinson's Disease Prediction using Logistic Regression"****, submitted for the partial fulfilment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under the supervision of* ***Ms Jiss Kuruvilla*** *. This submission represents my ideas in my own words and where ideas or words of others also have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.*

Meenakshi Sajeev

MUT23MCA-2044

Place:

Date:

ACKNOWLEDGEMENT

I express my heartfelt gratitude to God for granting me the strength and wisdom o complete this project successfully.

I extend my sincere thanks to **Dr Neelakantan P C, Principal, Dr Chikku Abraham, Vice Principal and Dr Shajimon K John, Dean of Academics**, for providing the necessary facilities to carry out this project.

I would like to thank **Dr Saritha K** , Head of the Department of Computer Applications, for her guidance and support throughout this endeavor.

I extend my appreciation to **Dr Smitha Anu Thomas and Dr Sujithra Sankar** , my project coordinators, for their valuable insights and guidance.

Special thanks to **Ms Jiss Kuruvilla** ,my Project Guide, for her invaluable mentorship, encouragement and support at every stage of this project.

I am grateful to all the teaching and non-teaching staff of Department of Computer Applications for their assistance and cooperation during the course of this project.

I also wish to acknowledge the support and understanding of my friends and family, whose encouragement kept me motivated throughout this journey.

Meenakshi Sajeev
MUT23MCA-2044

ABSTRACT

Parkinson's disease (PD) is the second most common age-related neurological disorder, affecting millions of people worldwide in whom approximately 1% of the population aged over 60 are affected. The progression of PD makes its diagnosis often difficult because various motor and cognitive symptoms come into play that overlaps with a host of other conditions, like normal ageing and essential tremor. Visible symptoms often occur at around age 50, including problems with walking or communication. Early detection is a doorway for intervention; in fact, early symptoms especially slight abnormalities of speech are hard to be identified by clinicians. This project takes up the Logistic Regression algorithm technique applied to the characteristics analysis of voice as factors predictive of PD. For this, the Logistic Regression model was trained to classify healthy people and PD people using a dataset of 342 samples with 24 features sampled from voice recordings. Such notable accuracy achieved by the Logistic Regression classifier underscores its potential as an effective tool for early detection. To date, no cure is available for PD; however, some medications are capable of helping relief symptoms such that patients may continue life styles by controlling complications. This research concentrates on the application of machine learning in the enhancement of capabilities for diagnosing patients afflicted with Parkinson's disease by voice analysis and it establishes early interventions to prevent complications

TABLE OF CONTENTS

LIST OF SYMBOLS

Symbols	Definitions
δ	Delta
β	Beta
σ	Sigma

LIST OF ABBREVIATIONS

Abbreviations	Definitions
PD	Parkinson's Disease
LR	Logistic Regression
HC	Healthy Control
CSV	Comma Separated Values
SMOTE	Synthetic Minority Over-sampling Technique
MDVP	Multidimensional Voice Program
IQR	Interquartile Range
ML	Machine Learning

LIST OF FIGURES

LIST OF TABLES

CHAPTER – 01

INTRODUCTION

1.1 Definition

Parkinson's disease (PD) is a neurodegenerative disease of mainly the central nervous system that affects both the motor and non-motor systems of the body. The symptoms usually emerge slowly, and, as the disease progresses, non-motor symptoms become more common. Usual symptoms include tremors, slowness of movement, rigidity, and difficulty with balance, collectively known as parkinsonism. Parkinson's disease dementia, falls and neuropsychiatric problems such as sleep abnormalities, psychosis, mood swings, or behavioral changes may also arise in advanced stages.[1]

1.2 Significance

Applying machine learning to predict Parkinson's Disease (PD) through voice analysis is an innovative approach that has revolutionized the support for patients in managing the early stages of the disease diagnosis. Utilizing voice data offers a non-invasive, convenient, and cost-effective method to detect early signs of PD, even before its motor symptoms become apparent. In this, Logistic Regression (LR) is used to address the challenges in managing PD by categorizing basic attributes such as tremor along with pitch and jitter features found in voice recordings, indicating the risk of the disease.The Logistic Regression algorithm is trained to distinguish between normal and Parkinsonian speech patterns by using a dataset that includes speech recordings from both healthy individuals and those with PD. This allows for precise prediction of PD, making it valuable for telemedicine and remote patient care. Machine learning assists in treating patients in the early stages of the disease, ultimately enhancing treatment outcomes and decreasing the requirement for clinical assessment, which in turn offers potential financial gains for healthcare systems and patients.

1.3 Uses

Machine learning for Parkinson's Disease prediction has valuable applications in healthcare, as it can detect subtle signs of the disease like changes in voice or motor functions before noticeable symptoms appear, enabling early diagnosis.The use of early detection may prompt actions that

would help to arrest the worsening of the disease and the resulting effects on the patient. It provides an easy and cheap means of assessing the patients hence less complex or cheap medical procedures are required. Furthermore, it promotes the provision of healthcare through telemedicine hence making diagnosis and follow-up easy among the people living in the rural or other regions that cannot access healthcare services easily. Instead, machine learning models can provide constant monitoring of the progression of the ailment, permit the doctors to design treatment regimens that are based on the specific patient's parameters. In general, it improves the clinical approach as well as the development of the drug by providing faster and more effective understanding of the disease.

1.4 Contribution

In this project, the focus was on utilizing Logistic Regression (LR) for the detection of PD. This dataset is composed of a range of biomedical voice measurements from 31 people, 23 with Parkinson's disease (PD) and 8 without PD. Each column is a particular voice measure, and each row corresponds to one of 195 voice recordings from these individuals ("name" column).Each class was meticulously utilized for training, testing, and validation purposes. Through the application of Logistic Regression, the model was trained to accurately differentiate between the parkinson's disease, enabling efficient disease identification and prediction. The project's importance lies in its potential to transform healthcare by offering a dependable tool for early Parkinson's Disease detection through voice analysis. Through the application of machine learning methods, Logistic Regression (LR), this project enables prompt diagnosis, leading to early intervention and ultimately better patient outcomes.

1.5 Objective

The main focus of this project is to build an Logistic Regression -based model that can predict the probability of Parkinson's disease (PD) by examining vocal recordings. The goals of the project include designing a dependable system that requires voice analysis to help clinicians in the early detection of PD using feature selection, data preprocessing, and model validation. In specific, The critical voice features that are related to Parkinson's condition are the focus of this project to improve the early diagnosis of the disease so that treatments can be laparized at the earliest and better outcome is achieved for the patients. In the long run, it is envisioned that an affordable and user-

friendly diagnostic test will be developed to help health care providers in the prevention and treatment of Parkinson's Disease.

In the introductory chapter, the focus is on the critical importance of early detection of Parkinson's Disease in healthcare, particularly through the analysis of voice patterns and their correlation with the disease.. It avers that there exist advanced technologies such as Logistic Regression (LR) models in the modern age, that can be employed in the prediction of Parkinson's Disease and needless to mention, the early management and treatment of patients affected by such diseases has become the cornerstone of all medical practice. In addition, it presents a foundation for the details to be discussed later in the text with regard to the methodology and results and the relevance of the research in the context of healthcare advancement and early diagnosis.

1.6 Agile Methodology

Agile methodology is a project management framework that breaks projects down into several dynamic phases, commonly known as sprints. The Agile framework is an iterative methodology. After every sprint, teams reflect and look back to see if there was anything that could be improved so they can adjust their strategy for the next sprint.

Scrum is a common Agile methodology for small teams and also involves sprints. The team is led by a Scrum master whose main job is to clear all obstacles for others executing the day-to-day work. Scrum teams meet daily to discuss active tasks, roadblocks, and anything else that may affect the development team.

Sprint planning: This event kicks off the sprint. Sprint planning outlines what can be delivered in a sprint (and how).

Sprint retrospective: This recurring meeting acts as a sprint review—to iterate on learnings from a previous sprint that will improve and streamline the next one. [14]

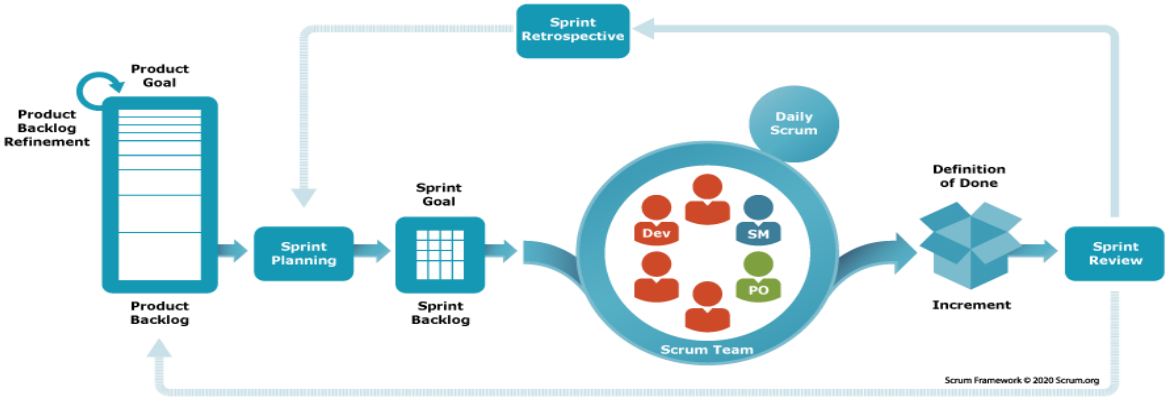


Fig 1.1 Scrum Framework

1.7 Organisation of Project

Here by I'm organising my project work into 8 chapters.

Chapter 2: Literature Review
Reviews existing research on Logistic Regression -based PD prediction.

Chapter 3: System Design

Outlines the architecture of the Logistic Regression model,including preprocessing steps and model selection

criteria

Chapter 4:Methodology

Describes the dataset,preprocessing of the data and model training

Chapter 5: Results and Discussions

Presents and analyzes experimental results,comparing them with existing literature.

Chapter 6: Conclusion
Summarizes findings,discusses implications,and suggests future research directions.

Chapter 7: Future Work
Explores potential enhancements to the model and future research avenues.

Chapter 8: References
List all cited references to ensure academic integrity.

CHAPTER – 02

LITERATURE SURVEY

In this chapter, we explore the domain of early detection of Parkinson's disease (PD) through voice analysis, emphasizing the considerable progress made possible by the incorporation of machine learning algorithms. Our exploration examines various studies that have introduced innovative methods for accurately identifying and predicting Parkinson's disease through vocal patterns. By examining a diverse range of datasets and methodologies employed in these studies, we aim to underscore the critical role of technology in addressing healthcare challenges and improving patient outcomes. This introductory overview lays the groundwork for a comprehensive analysis of recent research efforts focused on enhancing early diagnosis and management strategies for Parkinson's disease.

Several research studies have investigated the application of machine learning algorithms for predicting Parkinson's disease via voice analysis. A significant study by Mandal and Sairam (2020) presented a robust machine learning framework for Parkinson's disease prediction using a dataset from the UCI Machine Learning Repository, which contains 195 instances with 23 attributes related to voice measurements. Their study employed advanced methods such as sparse multinomial Logistic Regression, rotation forest ensemble with support vector machines, artificial neural networks, and boosting techniques. To enhance the feature selection process, they introduced a Bayesian network optimized with a Tabu search algorithm and Haar wavelets as a projection filter. The highest reported accuracy for the Logistic Regression models was 100%, with sensitivity and specificity values of 0.983 and 0.996, respectively. These findings emphasize the importance of feature selection and the combination of multiple models to achieve high accuracy, thereby reducing the chances of misdiagnosis. The study conducted experiments at a 95% and 99% confidence level, showing strong statistical reliability for the results.

Another notable contribution to this field is the research conducted by Ahmed et al. (2022), which focused on using voice signals to classify Parkinson's disease. This study used the UCI Parkinson's Disease Dataset, consisting of 195 instances and 22 attributes, primarily representing voice signal

features like fundamental frequency, jitter, and shimmer. Six algorithms were tested: Stochastic Gradient Descent (SGD) Classifier, Extreme Gradient Boosting (XGB) Classifier, Logistic Regression, Random Forest, K-Nearest Neighbor (KNN), and Decision Tree. The voice signal features, including intensity and spectrum, were extracted and analysed, leading to accurate classification results. Among the models, Random Forest achieved the highest accuracy of 97%, followed by XGB and KNN with 95% accuracy, and Logistic Regression with 91%. This study highlights the importance of voice data as a diagnostic tool, demonstrating how machine learning can enhance the accuracy of non-invasive PD diagnosis.

CHAPTER – 03

SYSTEM DESIGN

3.1 Existing System

This chapter brings forth a system design motivated towards the early detection of Parkinson's disease through voice analysis relying on support vector machine-based algorithms. Here, focus would be on developing an efficient system that can identify and classify with high accuracy people prone to Parkinson's disease based on vocal characteristics. To achieve this, the system exploits a database containing voice recordings from patients diagnosed with Parkinson's and age- and gender-matched healthy controls, with an emphasis on important acoustic features for the disorder. Fundamental constituents of system design include preprocessing strategies related to audio signal enhancement and feature extraction, and strategic architecture choice for the Logistic Regression model type, and highly training and validation procedure with the purpose of implementing them to improve the models' efficiency. It shall comprise stages like data gathering, preprocessing, feature extraction, model training, validation, and inference among its design. Through these stages, the system works towards providing actionable insights for the healthcare professional so that timely intervention and better management of the disease- Parkinson's can be ensured.

The proposed Logistic Regression -based approach will improve early detection and diagnosis because the system ensures reliable and scalable solutions for the evaluation of voice characteristics in individuals.This system design holds promise in improved patient outcomes and addresses significant problems in neurodegenerative disease management, advancing the diagnostic accuracy while allowing early intervention. It is therefore crucial that there be early identification of the disease to improve treatment and management of it, and thus underlines the requirement for developing a robust system which applies machine learning techniques towards analyzing vocal impairments associated with the condition.

3.2 Proposed System

This proposed system utilizes Logistic Regression machine learning algorithms for the detection and classification of Parkinson's Disease based on voice analysis. Based on a data set comprising voice recordings from a set of patients with PD and a set of healthy controls, the model will be designed to robustly predict the risk of PD based on the characteristics of the voice. Thus, the system uses powerful Logistic Regression in dealing with high-dimensional data to identify subtle voice changes that can describe early stages of the disease with precision. The system works through careful procedures in feature extraction, model training, and validation for giving actionable insight to health care professionals. Finally, through early diagnosis and intervention, this machine learning system would work towards better patient outcomes and proper management of the disease, leading towards bettering healthcare practices and quality of life amongst those affected.

3.2.1 Technologies Used

Python

Python is a high-level, easy-to-learn, very versatile, and widely-used interpreted programming language. Guido van Rossum began developing Python in the late 1980s and released it in 1991. Since then, it has expanded to rank among the most broadly used programming languages in the world. It is possible to support many different disciplines including but not limited to web development, data analysis, machine learning, artificial intelligence, scientific computing, and more with its huge ecosystems of libraries and frameworks.

Python is one of those languages that best exhibit simplicity in syntax, focusing more on simplicity and readability than almost anything else, and so allows programmers to write readable, concise code. High readability makes Python a great choice for both beginners and professional developers alike. Flexibility is the other thing offered by Python because it supports numerous programming paradigms-procedural, object-oriented, and functional programming-and lets developers opt for the strategy best suited for them.

Python has an extremely thorough set of libraries and toolkits, with key ones including NumPy and pandas for complex data analysis, and then the machine learning possibilities with TemorFlow and even scikit-learn. The dynamic typing and garbage collection also make the whole aspect easier and

accelerate development. Moreover, a high level of community support from developers, educators, and enthusiasts also contributes in supporting and promoting this language, providing support to new learners, and promoting innovative ideas within numerous fields.

Python is a widely popular language that is easy to learn and is used by both an enormous number of users as well as a large ecosystem, which makes it powerful and versatile. Other benefits include web development, data analysis, machine learning, and scientific computing. These are a few of the many applications that favor it because of its robust community, a large library, and its ease of use.

Google Colab

Google Colab short for Google Colaboratory, is an entirely cloud-based environment that is very much designed to make collaboration-friendly coding and experimentation more comfortable, particularly in Python. The user can write and execute Python code directly from a web browser, and all this without having to install anything on his local machine. Launched last 2017, Google Colab quickly gained popularity among data scientists, researchers, and educators: accessible and easy to use. Among other features, the provision of free access to very powerful hardware with GPUs and TPUs provides a huge acceleration for training machine learning models. This is useful for operations involving vast datasets and complex calculations.

The Google Colab service offers Jupyter notebook-like features, hence making it easier for users to draft documents that should contain code but also rich text content, such as images or links. This makes it ideal for documentation, sharing, and collaborating on data science projects. Users can easily share their notebooks with other users with real-time collaboration and feedback. In addition, it works well with Google Drive so that users can store and access the notebooks and datasets easily. It supports many libraries and frameworks such as TensorFlow and PyTorch from the deep learning realm, NumPy, and pandas for data manipulation, making it very versatile in machine learning, data analysis, and scientific computing.

3.2.2 Architectural Design

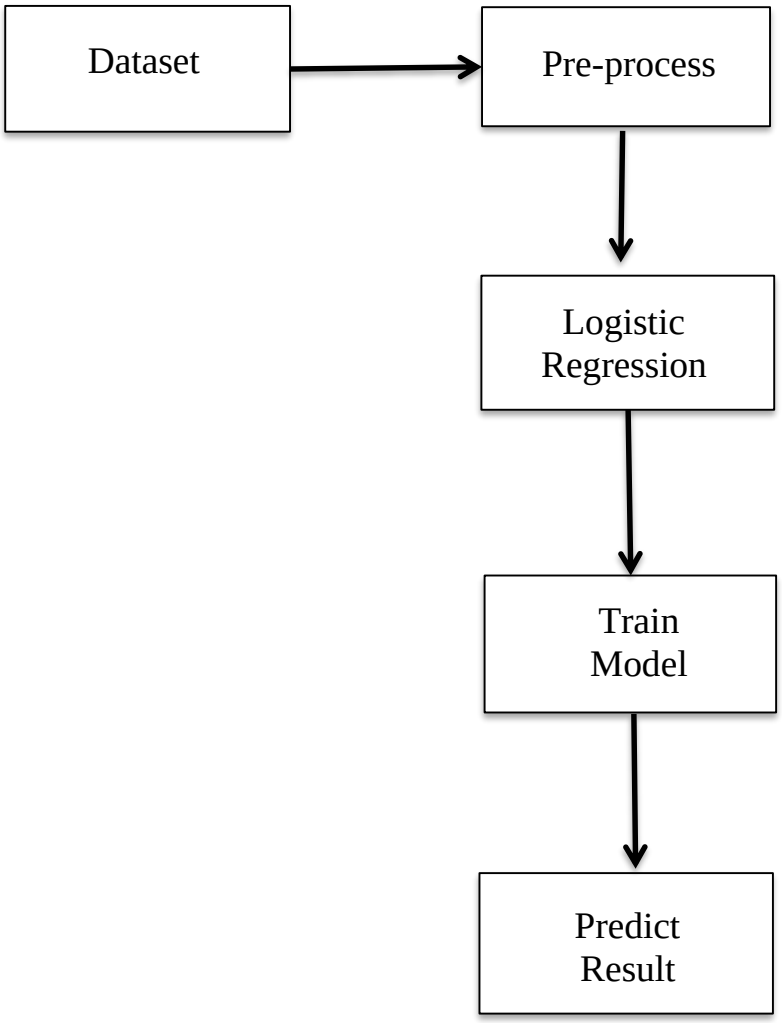


Fig 3.1 Architectural Design

CHAPTER – 04

METHODOLOGY

The development of a structured methodology-based Parkinson's disease predictive model will primarily rely on machine learning techniques. The first step will involve an exhaustive literature review, the rationale behind which is to synthesize the findings from previously conducted studies and help identify critical gaps in the current body of knowledge. This review is essential for selecting the appropriate datasets and identifying the best-suited machine learning algorithms for this particular problem. The methodology subsequent to the literature review for data gathering, preprocessing, feature extraction, and model development would consist of a proper systematic approach to each step involved in the process. The aim of the project would be to enhance the early detection accuracy of methods in the case of Parkinson's disease and contribute value to research within healthcare fields.

4.1 Dataset

Parkinson's Data Set Description The Parkinson's Data Set consists of measurements of biomedical voice that arise from a broad range of measurement types taken on 31 patients, of which 23 were diagnosed to have Parkinson's disease (PD). Comments on the Dataset Each column of the dataset is a specific measure of voice, while each row corresponds to one of 195 voice recordings obtained from the subjects listed in column "name". This dataset aims to distinguish between healthy individuals and those who have the disease PD, according to the "status" column-a value of 0 to represent a healthy subject, while a value of 1 indicates a Parkinson's patient. The dataset is ASCII CSV, with an average of six recordings per patient, and each recording is shown in a row.[3].
For more details, you can explore the dataset on
https://www.kaggle.com/datasets/vikasukani/parkinsons-disease-data-set

Voice Measure	Meaning
Name	ASCII name of subject and recording number(categorical variables)
MDVP:FO(HZ)	Average vocal fundamental frequency(Numerical variables).
MDVP:Fhi(HZ)	Maximum vocal fundamental frequency(Numerical variables).
MDVP:Flo(HZ)	Minimum vocal fundamental frequency(Numerical variables).
MDVP:Jitter(%)	Several measures of variation in fundamental frequency (Numerical variables).
MDVP:Jitter(Abs)	
MDVP:RAP	
MDVP:PPQ	
MDVP:DDP	
MDVP:Shimmer	Several measures of variation in amplitude (Numerical variables).
MDVP:Shimmer(dB)	
Shimmer:APQ3	
Shimmer:APQ5	
MDVP:APQ	
Shimmer:DDA	
NHR	Measures of the ratio of noise to tonal components in the voice(Numrical variables)
HNR	
status	0 for HC and 1 for PD(Numerical variables)
RPDE	Nonlinear dynamical complexity measures(Numerical variables)
D2	Measures of the ratio of noise to tonal components in the voice(Numrical variables)
DFA	
spread1	Nonlinear measures of fundamental frequency variation(Numerical variables).
spread2	
PPE	

Table 4.1 Dataset Attributes

4.2 Product Backlog

Backlog Id	User Stories	Description
PD1	As a user ,I want to do literature review and understand data	• Perform a literature review on existing parkinson's disease detection methods. • Explore the dataset and its features • Clean the dataset • Discussing about the preprocessing of data
PD2	As a user, I want to do data preprocessing & Feature engineering	• Scale and normalize numerical features. • Perform feature selection. • Split data into training, validation, and test sets. • Handle Outliers
PD3	As a user ,I want to do Model selection and train the model	• Choose algorithms based on literature review findings. • Implement initial models • Train the models • Evaluate model accuracy.
PD4	As a user, I want to analyse and optimise the model.	• Perform Hyperparameter tuning. • Evaluate model accuracy,Precision,recall,F1-score • Generate confusion matrix and classification report. • Evaluate sensitivity and specificity
PD5	As a user, I want to improve the model & test it	• Evaluate the model on test data • Visualize confusion matrix • Compare different models
PD6	As a user, I want to do Final checks and Make a report	• Double-check everything • Discuss the Project summary and findings • Write the conclusion • Prepare the project report

Table 4.2 Product backlog

4.3 Sprint and Burndown Charts

Backlog Id	User Stories	Description
PD1	As a user ,I want to do literature review and understand data	• Perform a literature review on existing parkinson's disease detection methods. • Explore the dataset and its features • Clean the dataset • Discussing about the preprocessing of data

Backlog ID	User Stories	Initial Estimate	Jul-22	Jul-23	Jul-26	Jul-30	Jul-31	Aug-02	Aug-05	Aug-06
		Day 0	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8
PD1	Literature review	3	1	1	1					
PD1	Gathering Dataset	1					1			
PD1	Clean dataset	1						1		
PD1	Preprocessing	3							1	2
Remaining Effort		8	7	6	5	5	4	3	2	0
Ideal Trend		8	7	6	5	4	3	2	1	0

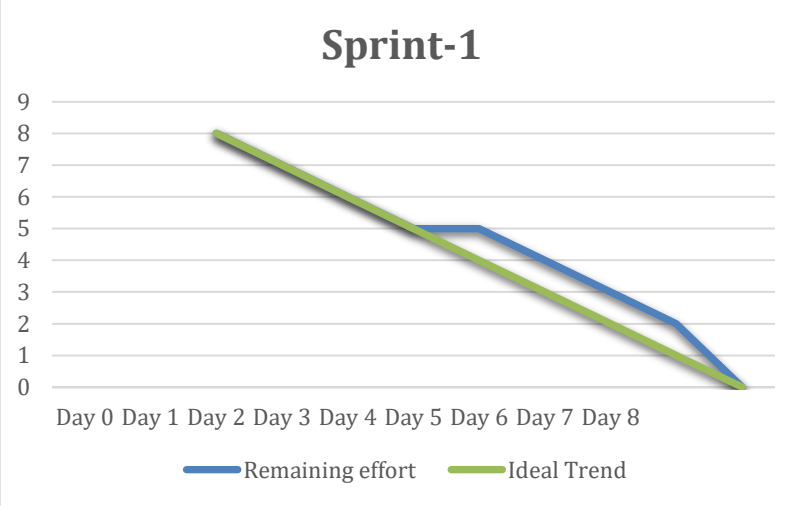


Fig 4.1 Sprint 1

Backlog Id	User Stories	Description
PD2	As a user, I want to do data preprocessing & Feature engineering	• Scale and normalize numerical features. • Perform feature selection. • Split data into training, validation, and test sets. • Handle Outliers

Backlog ID	User Stories	Initial Estimate	Aug-09	Aug-10	Aug-13	Aug-15	Aug-16	Aug-20	Aug-21	Aug-23
		Day 0	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8
PD2	Feature Scaling	1	1							
PD2	Feature Selection	3			1	1	1			
PD2	Handle outliers	3					1	2		
PD2	Dataset Splitting	1								1
Remaining Effort		8	7	7	6	5	3	1	1	0
Ideal Trend		8	7	6	5	4	3	2	1	0

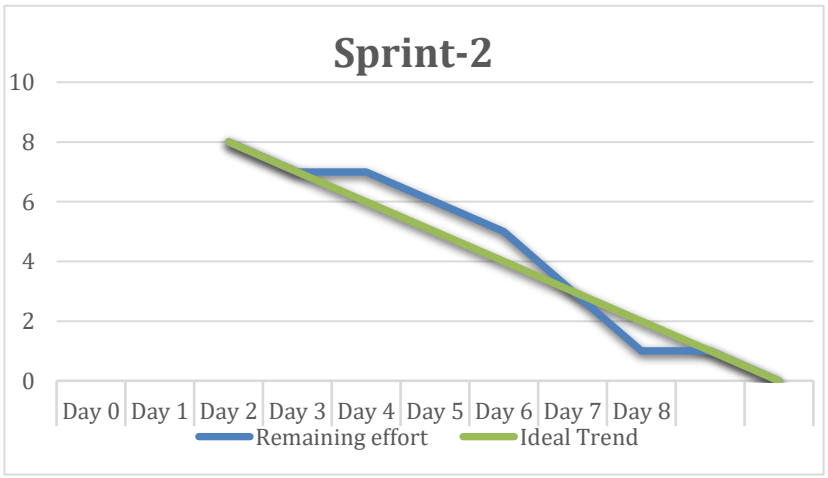


Fig 4.2 Sprint 2

Backlog Id	User Stories	Description
PD3	As a user ,I want to do Model selection and train the model	• Choose algorithms based on literature review findings. • Implement initial models • Train the models • Evaluate model accuracy.

Backlog ID	User Stories	Initial Estimate	Aug-26	Aug-27	Aug-30	Aug-31	Sep-02	Sep-03	Sep-06	Sep-08
		Day 0	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8
PD3	Model Selection	3	1	2						
PD3	Implementation of model	2				1		1		
PD3	Train the model	1						1		
PD3	Performance Evaluation	2							2	
Remaining Effort		8	7	5	5	4	4	2	0	0
Ideal Trend		8	7	6	5	4	3	2	1	0

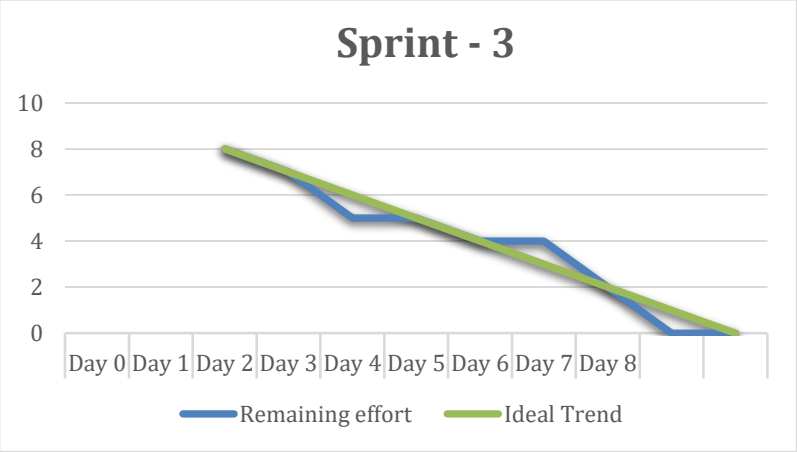


Fig 4.3 Sprint 3

Backlog Id	User Stories	Description
PD4	As a user, I want to analyse and optimise the model.	• Perform Hyperparameter tuning. • Evaluate model accuracy,Precision,recall,F1-score • Generate confusion matrix and classification report. • Evaluate sensitivity and specificity

Backlog ID	User Stories	Initial Estimate	Sep-10	Sep-11	Sep-13	Sep-23	Sep-25	Sep-28	Sep-30	Oct-01
		Day 0	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8
PD4	Perform Hyperparameter tuning.	4		1	1		2			
PD4	Evaluation Metrics	2					1	1		
PD4	Confusion matrix and classification report	1						1		
PD4	sensitivity and specificity	1								1
Remaining Effort		8	7	6	4	3	2	1	0	0
Ideal Trend		8	7	6	5	4	3	2	1	0

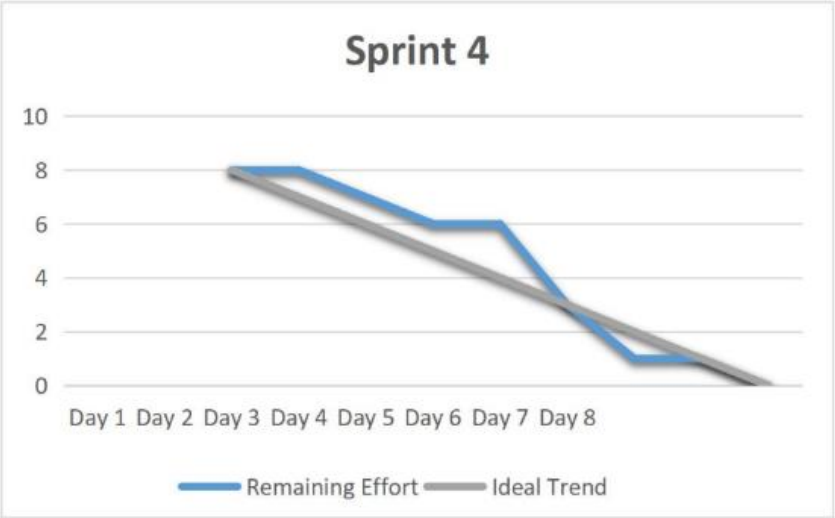


Fig 4.4 Sprint 4

Backlog Id	User Stories	Description
PD5	As a user, I want to improve the model & test it	• Evaluate the model on test data • Visualize confusion matrix • Compare different models

Backlog ID	User Stories	Initial Estimate	Oct-04	Oct-08	Oct-09	Oct-11	Oct-13	Oct-16	Oct-18	Oct-19
		Day 0	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8
PD5	Evaluate the model on test data	4	1	1			1		1	
PD5	Visualize confusion matrix	2			1			1		
PD5	Compare different models	2								2
Remaining Effort		8	7	6	5	5	4	3	2	0
Ideal Trend		8	7	6	5	4	3	2	1	0

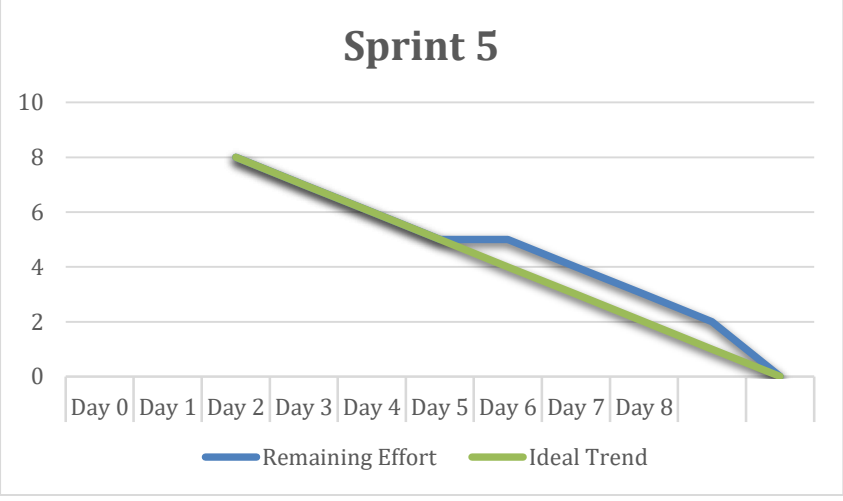


Fig 4.5 Sprint 5

Backlog Id	User Stories	Description
PD6	As a user, I want to improve the model & test it	• Double-check everything • Discuss the Project summary and findings • Write the conclusion • Prepare the project report

Backlog ID	User Stories	Initial Estimate	Oct-21	Oct-22	Oct-24	Oct-26	Oct-28	Oct-30	Nov-01	Nov-04
		Day 0	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8
PD6	Double-check everything	2		1			1			
PD6	Project summary and findings	2			1	1				
PD6	Conclusion	1						1		
PD6	outline of project report	3							1	2
Remaining Effort		8	8	7	6	5	4	3	2	0
Ideal Trend		8	7	6	5	4	3	2	1	0

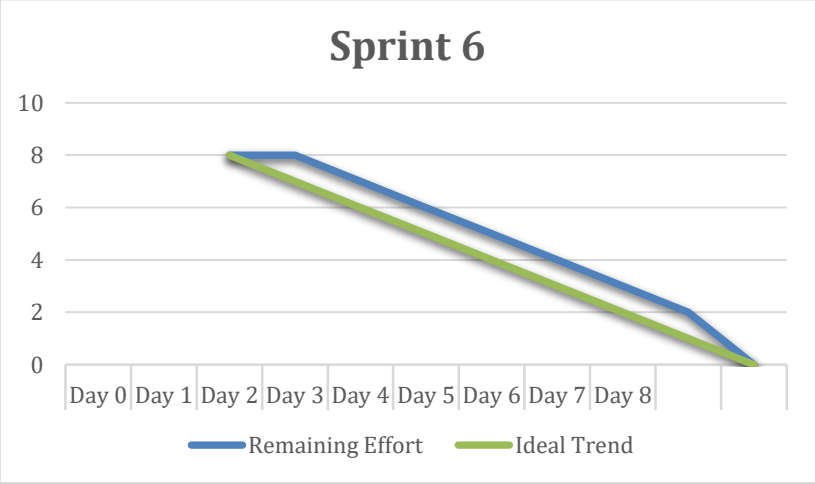


Fig 4.6 Sprint 6

4.4 Methods

The suggested method, which is used to classify whether the patient has PD or not, is developed based on the usage of Google Colab environment and Python language. The structural methodology of the proposed model is divided into the following six steps: data preprocessing, features selection, Synthetic Minority Over-sampling Technique (SMOTE), hyperparameter tuning (GridSearchCV), machine and deep learning classification models, and performance evaluation.

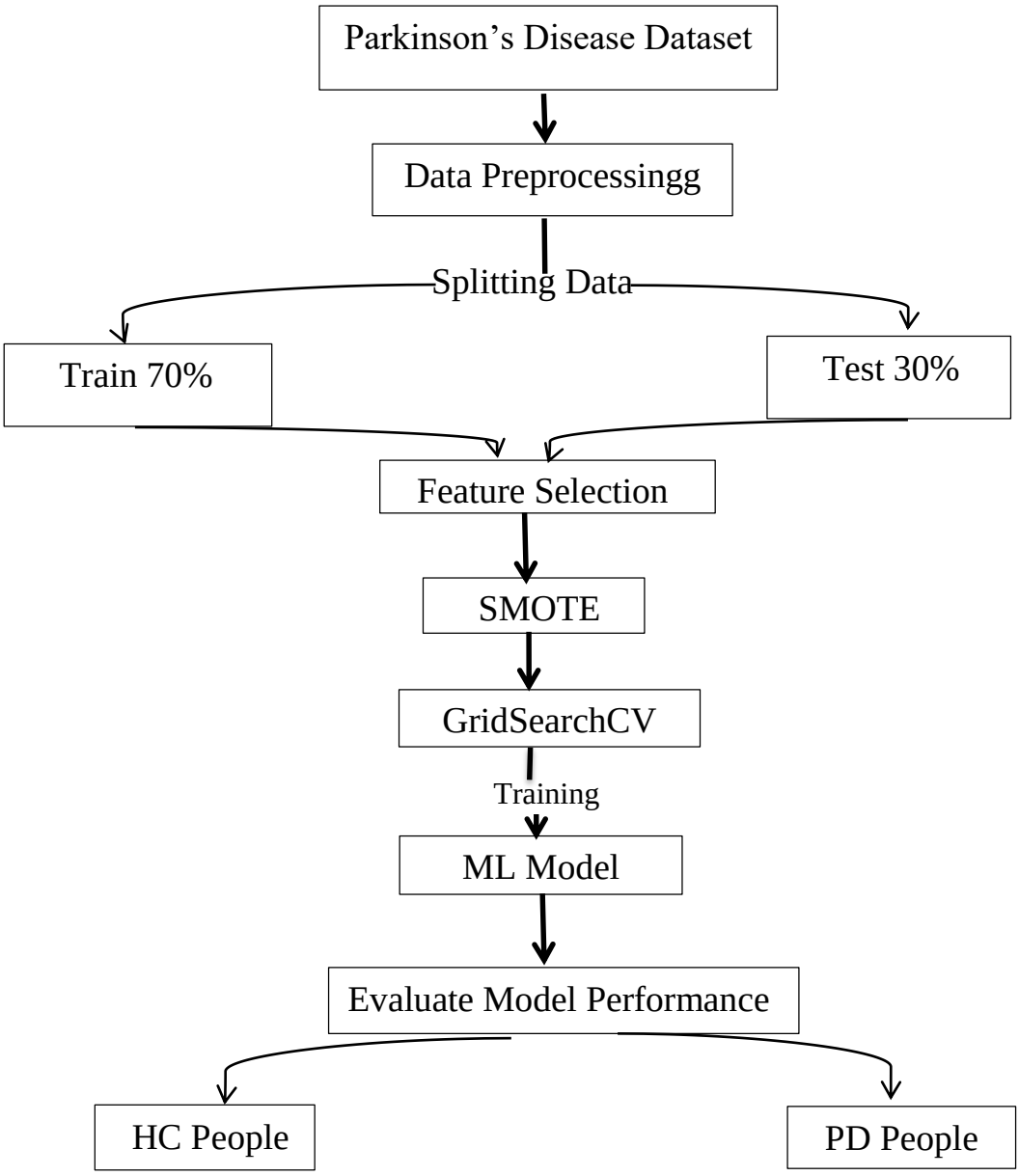


Fig 4.7 Steps of Proposed Model

4.4.1 Data Preprocessing

Majorly among data preprocessing, the most significant fact is that it allows a model to appropriately understand the features that it learns about and de-noise information [5]. The dataset had been imported into Google Colab as a CSV file using the Pandas package. After removing any duplicate entries or null values, utilized the "status" column to see that the data set was skewed at 147 for PD and 48 for HC, which is equivalent to 24.62% for HC and 75.38% for PD. Since the dataset is in an imbalanced state, it prevents chances of under-fitting and over-fitting of the model. Divided the data set into 80:20 ratio for train/test split. The training set contains known outputs, and what it learns from this output may be extrapolated to other data sets. Each feature is scaled individually by computing the relevant statistics on the samples in the training set. Then, the mean and standard deviation are saved and used on later data with the help of the transform in StandardScaler[6]. Equation 1 expresses the mathematical form of StandardScaler normalization.
For this paper, I used several libraries, among which are; NumPy, Pandas, Matplotlib, Seaborn, and Scikit-learn abbreviated as Sklearn.

NumPy is Python's basic package for scientific computation. It allows you to add any kind of mathematical operation within the code. On top of that, it lets you have vast multidimensional arrays and matrices within your code. The Pandas library is good for manipulation and data analysis, used while importing datasets into an organized dataset. Matplotlib and Seaborn are the back bones of Python data visualization. Matplotlib refers to a Python library that can be used in plotting 2D graphs with the help of other libraries, such as Numpy and Pandas; it uses Seaborn in the plotting of graphs using Matplotlib, Pandas, and Numpy. The last one is Sklearn, the most usable and robust package in Python for machine learning purposes. It gives a consistency interface based on Python as well as tools for classification, regression, clustering, and dimensionality reduction.

$$\text{Standard Scaler} = \frac{X_i - mean(X)}{stdev(X)} \qquad (1)$$

The index i varies from 1 to n,representing each individual data points in the dataset,where n is the total number of observations.

4.4.2 Feature Selection

Feature selection[6] is an important preprocessing step for machine learning, which includes the process of identifying and selecting a subset of relevant features from the original dataset. This has as its main purpose the improvement of model performance, reduction of overfitting, and increasing the interpretability by removing redundant or irrelevant data. It was then during this phase that SelectKBest with Fisher's score (chi-squared)[7] had identified the most important features associated with the dataset. The statistical measure uses the ratio of variance between classes and variance within classes to measure the relevance of features. Fisher's score is calculated by the following formula:

$$F = \frac{Var(C_1) + Var(C_2)}{Var(c)}$$

Where C_1 and C_2 represent the different classes and C represents the overall dataset. The more significant the value of Fisher's the better is the feature distinguishing between the classes. The latest research has been able to show that to actually improve the performance and explainability of the model, suitable feature selection methods, including SelectKBest, must be applied to it. It would have selected the features depending on the score and efficiently removed data of lower relevance from the input to bring about the training process to be more efficient. In our analysis, we considered all the features that had received top scores:

MDVP:Fo(Hz), spread1, MDVP:Fhi(Hz), MDVP:Flo(Hz), MDVP:Jitter(Abs), and MDVP:Shimmer(dB).

Figure 4.2 illustrates the importance of such features, at the y-axis we list their respective Fisher's scores, which usually highlight what each feature contributes to distinguishing between healthy individuals and patients with Parkinson's disease. Feature selection is critical to optimize the predictiveness of our classification models.

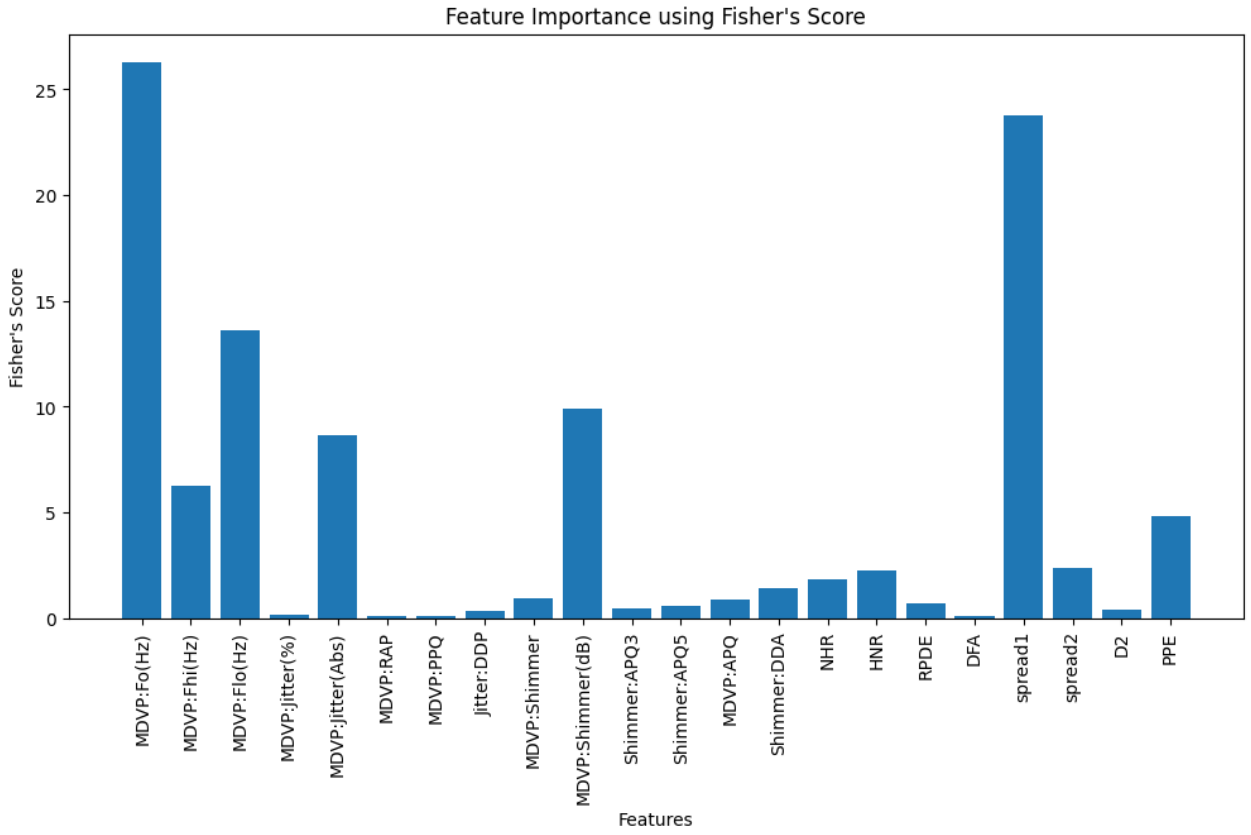
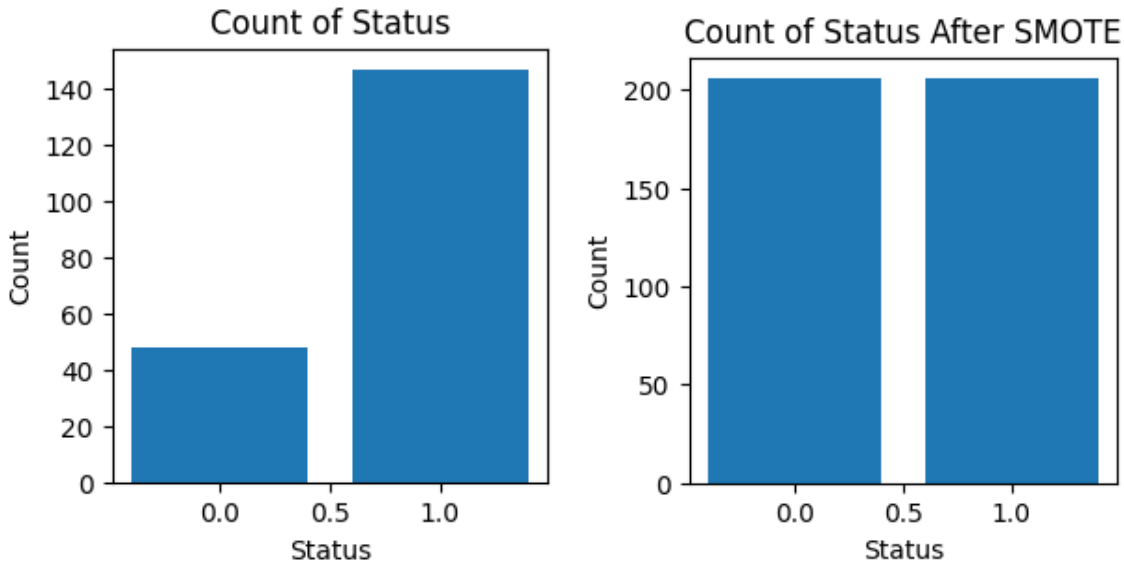


Fig 4.8 feature selection using Fisher's Score

4.4.3 Synthetic Minority Over-sampling Technique (SMOTE)

In addressing the imbalance problem in dataset, we implemented a multidimensional approach making our machine learning model robust and reliable. Figure 4.4(a) gives an idea of the dataset. It comprised 147 Parkinson's patients and 48 healthy controls (HC). To balance it, we used the Synthetic Minority Over-sampling Technique (SMOTE)[8], which allowed us artificially increase the samples in the under represented class. This therefore balanced PWP and HC after employing SMOTE. The point of record equality is depicted in Fig. 4.4(b) to be in an equally balanced level standing at 206 records for each class. This approach balances the dataset effectively by generating synthetic samples based on existing instances in the minority class, thereby making the dataset used even more representative for training purposes. Though oversampling the minority class may lead to some duplicate instances sooner or later, SMOTE enhances the diversity of the constructed

dataset because it generates the new samples through a process of linear interpolation among the existing minority samples. This technique not only corrects the imbalance issue but also improves the generalization capability of the model, thereby positively adding up to the overall diagnostic accuracy and performance of our models.



(a) (b)

Fig 4.9 Label Imbalance

4.4.4 Distribution Analysis of Individual Attribute

The Python libraries used are Matplotlib and Seaborn to go through distribution attributes in the dataset. It illustrates distribution plots using the function from Seaborn, showing them for individual columns and then introducing a function to do this in turn for all numerical columns. This clearly indicates why normal distribution is important to machine learning. It mentions Central Limit Theorem, which states the dataset samples must approximately follow the normal distribution for good generalization by algorithms. However, it's accepted that not all columns will follow Gaussian curves exactly. It is observed that most columns are nearly normally distributed with minor skewness. Therefore, it may be applicable to fit a broader population representation. Some of the individual distribution is demonstrated in Figure 4.4. This depicts distribution analysis of various

attributes, with a prime focus on MDVP (Multidimensional Voice Program) parameters, among others. This figure is likely to provide a view about how these attributes are distributed in the dataset, thereby pointing out the patterns of normality or skewness to apply machine learning algorithms effectively.

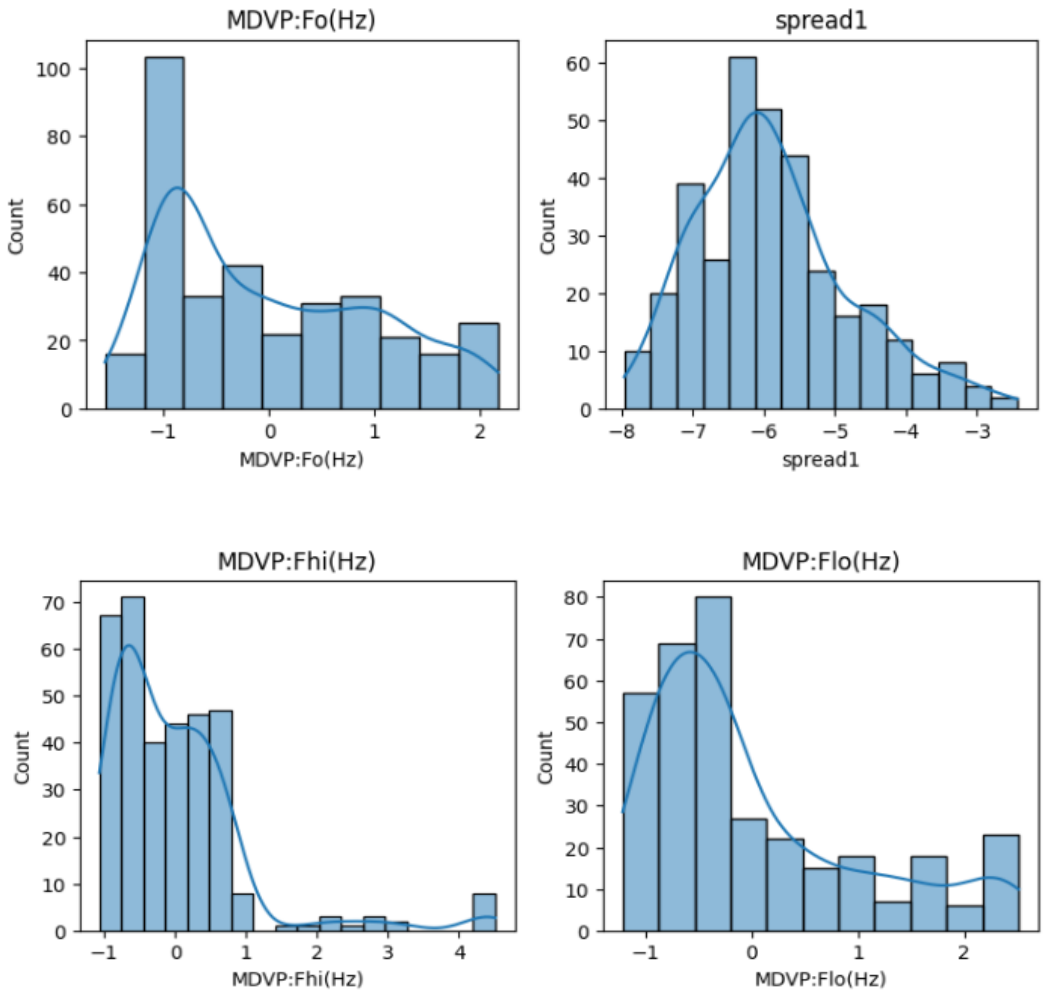


Fig 4.10 Distribution analysis of attributes

4.4.5 Handling Outliers

In this stage, box plots have been applied in order to graphically identify potential outliers[10] from features considered in the dataset. The box plot is a standard data display according to a five-number summary: minimum, first quartile, median, third quartile, and maximum. Box plots are also very effective in highlighting outliers —data points that are markedly different from the rest of the data.

Using the Python library Seaborn, we produced box plots of the above feature selection: See Fig. 4.5. Box plots enable us to visualize the distribution of every single feature clearly. They help identify outliers whose values are improbable compared to the others in the data set. Outliers are the points outside the whisker of a box plot.

Having found the outliers, I applied the IQR method to filter out these anomalies. IQR is defined as the subtraction of the third quartile, Q3, from the first quartile, Q1, thus yielding a measure of the spread of the dataset. To define the limits for outlier detection, I therefore computed:

$$\text{Lower Bound} = Q1 - 1.5 \times IQR$$

$$\text{Upper Bound} = Q3 - 1.5 \times IQR$$

Data Points which were outside these limits were identified as outliers and were thus removed from the dataset in the end. After removing the outliers, the total dataset was re-tested, and the box plots were plotted excluding the outliers, as shown in Fig. 4.6 in the subsequent steps. In this way, only valid data points were present. Valid data points can prevent biased outcomes for the trained or evaluated models.

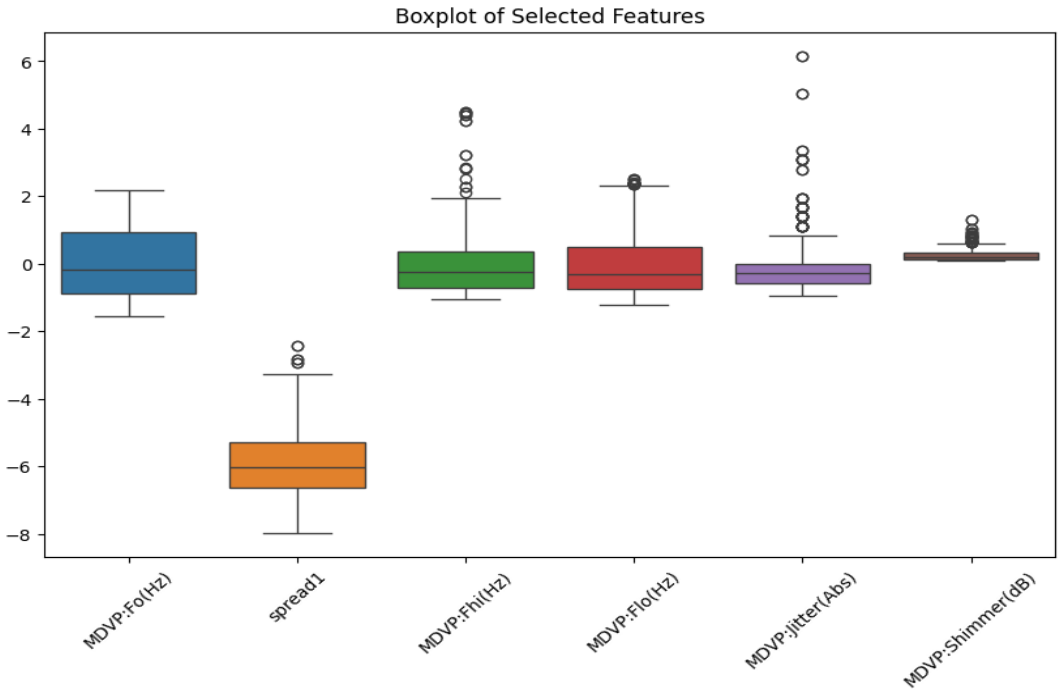


Fig 4.11 Box plot before removing outliers

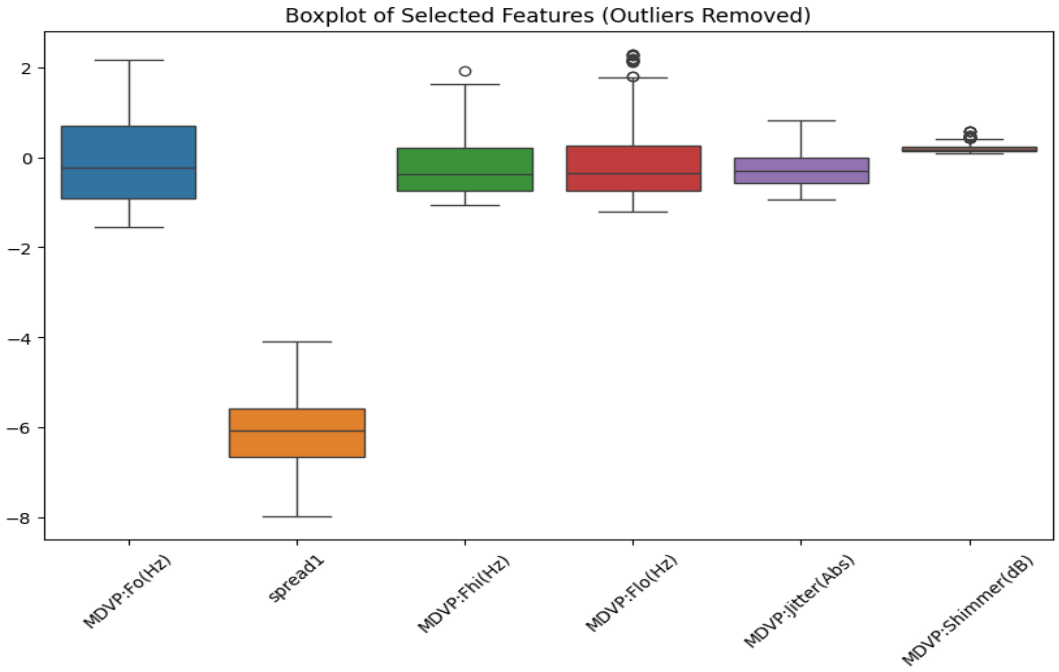


Fig 4.12 Box plot after removing outliers

4.4.6 Train Test Split

Train-test splitting is the fundamental approach in machine learning used for cross-validation of a model. The data set divides into two sections. Normally, data is divided 80% into training data set and 20% into testing data set. However, these percentages may vary based on the amount of complexity of data that might be involved. The training data set is then used to train the model such that it learns its underlying pattern, correlation, or trend. By exposing the model to a significant part of the data, it develops an appreciation of the relationships existing between input features and the target variable.

Once the model is trained, test set comes into picture. This part of data is isolated only during training so that there is a fair assessment of the performance of the model in terms of predictive accuracy. The proxy for new, unseen data, test set thus represents actual performance on the model. Through performance measurement on the test set, a measure of accuracy, precision, recall, and F1 score can be calculated to clearly give a picture of how effective the model is.

This will ensure that the model does not memorize data (overfitting) but instead learns generalizable patterns applicable to unseen data. In this project, the train-test split has played an important role in offering an objective assessment towards the model's ability to correctly classify patients with Parkinson's and healthy individuals, therefore offering a robust and reliable model at the end.

4.4.7 Model Training

The training of models in machine language feeds an ML algorithm with data to help identify and learn good values for all the attributes involved. Some of the types of machine learning models include: supervised and unsupervised learning.

Supervised learning is possible only if the training data contains both the input and output values. This dataset in which both the inputs as well as the output which are supposed to occur are available is called the supervisory signal. The training is based on how much the processed result differs from the documented one when the inputs are fed into the model.

Unsupervised learning determines patterns in the data. Then, it uses extra data in fitting the patterns or clusters. Also, in this process, it is an iterative approach which improves the accuracy

based on correlation to the expected patterns or clusters. In this method, there is no reference output dataset.

A training model is a dataset used to train an ML algorithm. It consists of the sample output data and the corresponding sets of input data, which may have a specific influence on the output. The training model is used in running the input data through the algorithm such that it correlates the processed output against the sample output. Such a correlation generates the result, and the model is modified in accordance with the result obtained. Such an iterative process is called "model fitting". The accuracy of the training dataset or the validation dataset proves to be very critical for the precision of the model. [11]

Logistic Regression(LR)

Logistic regression is a supervised machine learning algorithm that accomplishes binary classification tasks by predicting the probability of an outcome, event, or observation. The model delivers a binary or dichotomous outcome limited to two possible outcomes: yes/no,

0/1, or true/false.For example, it could be used to predict whether a customer will churn or not, whether a patient has a disease or not, or whether a loan will be repaid or not.

It involves the following steps:

1. Linear Combination of Features: Combine the input features using learned weights to compute a linear equation.

$$z=\beta\ 0 +\beta\ 1\ x\ 1 +\beta\ 2\ x\ 2 +\cdots+\beta\ n\ x\ n$$

2. Applying the Sigmoid Function: Use the sigmoid function to transform the linear combination into a probability between 0 and 1.

$$\sigma(z)= 1/(1+e^{\wedge} -z)$$

3. Decision Boundary: Classify the instance as class 1 if the predicted probability is 0.5 or greater; otherwise, classify it as class 0.

If $P(y=1|\ X)\geq 0.5$, the model predicts class 1 (positive class).

If $P(y=1|\ X)<0.5$, the model predicts class 0 (negative class).

4.Cost Function (Log-Loss):

$$\text{Log-Loss} = \frac{-1}{m}\sum_{i=1}^{m}[y_i log(\widehat{y_i}) + (1 - y_i)log(1 - \widehat{y_i})]$$

5.Training the Model: Optimize the weights by minimizing the log-loss through an iterative process called gradient descent.

$\beta j=\beta j-\alpha\partial\beta j\partial J(\beta)$

6. Making Predictions: After training, use the learned weights to predict the probability of new instances belonging to class 1.

Types :

1. Binary Logistic Regression

This is the simplest form of logistic regression, where the outcome variable has two possible outcomes (0 or 1). The model predicts the probability that the dependent variable belongs to a particular class.

2. Multinomial Logistic Regression

Used when the outcome variable has more than two categories (more than two classes). It generalizes binary logistic regression by allowing for multiple classes and uses a softmax function to handle the probabilities of each class.

3. Ordinal Logistic Regression

This type is applied when the dependent variable is ordinal, meaning that the categories have a specific order (e.g., ratings from 1 to 5). It accounts for the order in its predictions.

4.4.8 Hyperparameter Tuning(GridSearchCV)

In the logistic regression, hyperparameter tuning is crucial for optimizing the model's performance. For this project, I utilized GridSearchCV to identify the best hyperparameter values for my logistic regression model. GridSearchCV systematically evaluates various combinations of hyperparameters to determine the optimal settings that enhance predictive performance. It can run tests in parallel, allowing each configuration to be evaluated independently without reliance on prior results.

During this project, I configured GridSearchCV with the following parameters:

Estimator: This is the logistic regression model being utilized.

Param Grid: This is a dictionary containing the hyperparameter names and the corresponding values to explore during the tuning process. The following hyperparameters were considered:

C: Inverse of regularization strength; smaller values specify stronger regularization.

solver: Algorithm to use for optimization (e.g., 'liblinear', 'lbfgs', 'newton-cg', 'sag', 'saga').

penalty: The type of regularization to apply ('l1', 'l2', 'elasticnet', or 'none').

cv: This is an integer indicating the number of folds for K-fold cross-validation. It ensures that the model is validated using different subsets of the data, providing a robust assessment of its performance.

Hyperparameters resulting after carrying out the grid search:

C: 10

solver: 'liblinear'

penalty: 'l1'

These values affirm the general working of the logistic regression model with regard to the proper classification of voice recordings-that is to say discrimination between the Parkinson's patients and the normal controls.

4.4.9 Performance Evaluation

Performance evaluation is essential for assessing the effectiveness of the trained machine learning model in this project, where the Logistic Regression (LR) algorithm was used to predict Parkinson's disease based on voice data. A confusion matrix has been used in order to better visualize the number of true positives, true negatives, false positives, and false negatives, which provides an excellent view of how well the model works. In addition, a classification report was generated to outline the most important metrics such as accuracy, precision, recall, and F1-score. Indeed, these metrics provided insight into whether or not the model could correspondingly predict well on samples, thereby showing its suitability for large-scale applications in performance and ensuring

it attains high accuracy with class balance, which verifies its reliability for Parkinson's disease detection.

4.4.9.1 Accuracy

Accuracy represents the percentage of correctly classified events out of all instances tested. It is the simple statistic calculated as the number of correct predictions divided by the total number of predictions made. This depicts the overall accuracy of a model at doing the right predictions. Being inaccurate, even though is not a fantastic measure for class-imbalanced datasets since it will indicate that one class greatly outweighs the others.

$$\text{Accuracy} = \frac{\text{True Positive(TP)} + \text{True Negative(TN)}}{\text{True Positive(TP)+False Positive(FP)+True Negative(TN)+False Negative(FN)}}$$

4.4.9.2 Confusion Matrix

A confusion matrix is defined as a table representing a classification task that assesses the performance of a learning model on a set of test data with known true values. The table goes deep into how well those predictions made by your classification model compare to the actual class labels in the test data set. There are rows and columns. Each row indicates the true class labels, while each column indicates the class labels predicted. The diagonal of this matrix implies that every correct number of predictions for every class points to the number of instances by which the model has correctly predicted the class. In contrast, off-diagonal elements imply misclassifications; they tell us how many instances of something have been wrongly predicted under some other class by the model.

While for the detection of Parkinson's disease, confusion matrices are important in visualizing and further understanding the performance of these classification algorithms: it enables correctly to depict quality of predictions for real positive, false negatives, false positives, and true negatives. Actually, confusion matrices play a central role in evaluating the correctness of the classification model by calculating such metrics as accuracy, precision, recall, and F1-score. Accuracy essentially calculates the absolute correctness of the total prediction, whereas precision calculates the percentage of truly positive cases occurring among all instances predicted to be positive. The F1-

score is a balanced metric that considers false positives and false negatives-it is the harmonic mean of precision and recall. The following was used in generating the confusion matrix in detecting Parkinson's disease:

1. Prepare Test Data: A separate test data set that contained known true class labels-that is, an independent set not involving the elements applied to training or validation stages-included healthy controls and patients with Parkinson's.
2. Model Prediction: Using the trained model, classified class labels of the samples were predicted in the test dataset.
3. Comparison of Predictions: Class labels assigned by the model was compared to actual true labels for the samples
4. Categories of Confusion Matrix:
a)TP(True Positive): The model has correctly identified a case of Parkinson's disease.
b)TN(True Negative): The model has correctly identified a healthy control
c)FP(False Positive): The model has misclassified a healthy control as suffering from Parkinson's disease.
d) False Negative (FN): The model predicts healthy controls when the individual actually has Parkinson's disease.
5. Count Occurrences: The counts for every category (TP, TN, FP, FN) were tallied according to the comparison between predicted and true labels.
6. Outcome Matrix Organisation: A matrix is created where the rows are given by the actual class label-healthy controls and Parkinson's disease patients-and columns by the predicted class labels.
The cells in the resulting matrix were filled with the respective category counts, which gives an overall view of how well the model is classifying the samples.

4.4.9.3 F1- Score

The F1 score is a crucial metric for evaluating the performance of classification models, significantly for cases where class distribution is skewed . This includes their utility in scenarios like the detection of Parkinson's disease. It is a form of the harmonic mean of precision and recall, which can quite effectively balance the two conflicting costs against each other. Precision refers

to the positive predictions made by the model, which illustrates how many of these positive cases have actually been diagnosed as such. Recall, in contrast, measures the extent to which the model has correctly identified all relevant instances-a measure of how many actual cases were positive cases that the model actually correctly predicted.

The F1 score is calculated using the formula:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

This score lies between 0 and 1. The closer to 1 the score is, the better the model is. A score of 1 represents the ideal value of precision and recall. On the other end, a score of 0 is terrible. The F1 score can now be leveraged in the detection of Parkinson's disease to evaluate the might of the model to suitably classify the patients with minimal false positives, implying that the model does not only detect the disease well but also has a reduced possibility of misclassifying healthy people. This metric is more important for healthcare applications because both precision and recall are crucial determinants of diagnostic and therapeutic effectiveness.

4.4.9.4 Precision

Precision is a vital metric used to evaluate the performance of classification models, particularly in contexts where the cost of false positives is high.It measures how accurate a model's positive predictions are-that is, how many of the instances that the model predicted were positive are truly positive. The formula for precision is as follows:

$$\text{Precision} = \frac{\text{True Positive(TP)}}{\text{True Positive(TP)} + \text{False Positive(FP)}}$$

In this formula, the True Positives (TP) is defined to be the number of cases that were in reality positive and rightly classified, whereas False Positives are those instances wrongly classified as being positive. High precision score tells that the model has a low quantity of false positives which means that when it makes a prediction of positive class there is probably a high chance that that's really what it is. It is what is urgently needed in medical diagnostic tests, more so for something as

debilitating as Parkinson's disease - false positives because they cause distress to patients who then undergo unnecessary testing. Precision is thus essential for stakeholders to clearly understand the reliability of positive predictions from the model and base their decisions on outputs produced.

4.4.9.5 Sensitivity/Recall

Sensitivity, also known as recall or true positive rate, is an important measure in a classification model, especially when every occurrence in the set needs to be classified correctly. This measures the percentage of correct identifying models of all instances within a set that are actually positive. Sensitivity can be calculated using the following equation:

$$\text{Sensitivity} = \text{Recall} = \frac{\text{True Positive(TP)}}{\text{True Positive(TP)} + \text{False Negative(FN)}}$$

In the above equation, True Positives (TP) represents the correct instances that the model correctly predicts as positive, and False Negatives (FN) is defined as actual positive instances that have been wrongly classified by the model as negative. A high value sensitivity score reflects that the model is sensitive to positive cases. This is especially important in applications such as disease diagnosis, where failing to detect a condition has adverse results. For example, in the patient diagnosis of Parkinson's disease, the model has to be sensitive while indicating the presence of the disease in most patients before appropriate intervention for treatment. The sensitivity analysis will make the practitioners check the existence of all instances of positives in the model, hence leading to effective decision-making and outcome treatments in such patients.

4.4.9.6 Classification Report

A classification report is a summary that lists out the performance of a classification model, giving key metrics of how well the model has been done on a certain dataset. Generally, it covers several key metrics for each class, precision, recall, F1-score, and support. Precision refers to the accuracy that the positive predictions are made by a model; which goes in terms of relating the positive predictions that have appeared among all those predicted positively. Recall, which is sensitivity, goes on to relate with the identification of all relevant positive instances produced by a model; wherein true positive predictions appear among all actual positives. A balanced F1-score is the

harmonic mean of precision and recall where a single metric balances both precision and recall. In cases where classes are imbalanced, it becomes more meaningful. Support indicates the actual occurrences of each class within a dataset in context, so it can help explain the other metrics.

CHAPTER – 05

RESULTS AND DISCUSSION

Results refer to the findings or results achieved from a particular research or experiment put across in an organized manner like on tables, graphs, or descriptive narritives. In essence, these are the raw data collected and analyzed in order to answer the questions of the investigation or hypothesis.
Discussion actually interprets and explains the meaning of the result in the light of the research question or problems. It includes a comparison of findings with related literature, an explanation of any unexpected results, discussion of implications, and recommendations or directions for further research. The purpose of the discussion section is to enlighten understanding of results and their significance.

Accuracy Measure

The percentage of correctly categorized occurrences among all evaluated instances is known as accuracy.
Mathematically accuracy is calculated using the following formula:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

The accuracy obtained for the Logistic regression model is given below.

Accuracy = 76.81%

Confusion Matrix

A confusion matrix is a table used in classification as an aid in evaluating how well a machine learning model performs on a set of known test data. This matrix shows off-diagonal instances that illustrate misclassification where the class picked by the model was wrong. The number of correct predictions for every class is shown by the main diagonal of the matrix.
The figure below is the confusion matrix of this Logistic regression model:

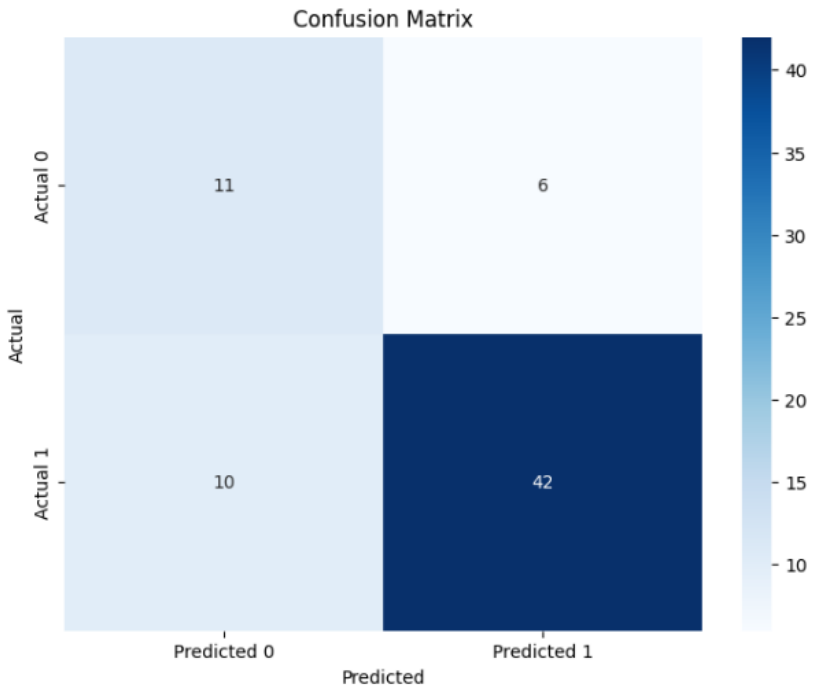


Fig 5.1 Confusion Matrix

F1-Score

The F1-Score provides a single score that balances false positives and false negatives by taking the harmonic mean of presicion and recall.

The following formula is used to determine the F1 score mathematically:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Recall

Recall is calculated as the ratio of true positives(TP) to the sum of true positives and false negatives(FN), expressed as below:

$$\text{Recall} = \frac{\text{True Positive(TP)}}{\text{True Positive(TP)} + \text{False Negative(FN)}}$$

Precision

Precision is a metric that counts the percentage of true positives ,or accurately predicted positive cases,across all instances that are projected to be positive(false positive plus true positives).
The formula for precision is :

$$\text{Precision} = \frac{\text{True Positive(TP)}}{\text{True Positive(TP)+False Positive(FP)}}$$

Classification Report

Metrics like precision,recall,F1-Score,and support for every class in the dataset are provided in the classification report,which is an overview of the performance of a classification model. These metrics aid in assessing how well the model can categorize.

	Precision	recall	f1-score	support
0	0.52	0.65	0.58	17
1	0.88	0.81	0.84	52
accuracy			0.77	69
macro avg	0.70	0.73	0.71	69
weighted avg	0.79	0.77	0.78	69

Table 5.1 Classification Report of Model

CHAPTER – 06

CONCLUSION

In conclusion, this project was centered on early Parkinson's disease diagnostics through the deployment of machine learning algorithms. The framework of this dataset was jointly developed by Oxford University and the National Centre for Voice and Speech, Denver. The initial set consisted of 147 records of patients diagnosed with Parkinson's and 48 healthy controls, resulting in a class imbalance. To address this, we applied the SMOTE oversampling and undersampling techniques. In the balanced dataset, there were 206 records for each class. The preprocessing was significant, as it allowed the model not to be biased by the class that was in the majority, ensuring better generalization across the other classes.

Further preprocessing of the data included removing unnecessary features and improving data quality through the use of Fisher's score and the IQR method. Fisher's score allowed us to order the most relevant features that distinguished between Parkinson's patients and healthy controls, while the IQR method helped identify and deal with outliers. After preprocessing, the data was divided into a training set and a testing set, allowing the model to train on one portion of the data while its performance was validated on another portion that was not visible to the model.

For model development, we focused solely on the logistic regression algorithm due to its effectiveness in handling high-dimensional data and binary classification problems like Parkinson's detection. After training the logistic regression model, it achieved an accuracy of 76.81% on the test data. The performance was evaluated using a confusion matrix and classification report, which provided insights into precision, recall, and F1-score. Overall, the model performed satisfactorily, and there remains ample room for improvement, such as optimal hyperparameter tuning or utilizing more sophisticated feature selection methods that might further enhance the model's accuracy.

Furthermore, logistic regression has shown promise in effectively diagnosing Parkinson's disease with a satisfactory degree of accuracy. The model's performance indicates that non-invasive early diagnosis tools might be developed using such machine learning techniques. Fine-tuning the model

with more data sources may help in future work to improve precision further, and this work can contribute to the development of telemedicine and early disease detection for patients, leading to better health outcomes.

The project also highlights the critical aspect of data handling and preprocessing in machine learning applications, especially in healthcare. By addressing some of these critical issues—class imbalance and feature selection—we were able to enhance the differentiation between healthy controls and Parkinson's patients. Key to this process was the use of SMOTE to balance the dataset and the careful selection of relevant features utilizing Fisher's score and IQR filtering to achieve an accuracy of 76.81%. The efficiency of the model itself is therefore of less concern in this regard; arguably, it is the quality and structure of the input data that proved key. Further techniques such as cross-validation and hyperparameter tuning could be employed for further improvement of the model, which will hopefully enhance the precision of its prediction performance and make it even more suited for real-world applications in medicine.

CHAPTER – 07

FUTURE WORK

This project has produced considerable results on the detection of Parkinson's disease based on machine learning; however, there are a variety of opportunities for further work and improvements aimed at enhancing the quality of the model, as well as generalizing and applying it in practice.

More diverse and larger data sources need to be fed into the dataset. The current dataset, as informative as it is, requires additional patient records and more diverse demographic data. Data involving the voices of individuals from different regions, languages, and age groups would make the model more robust and generalizable to a larger population. Collections of longitudinal data, where patients' voice recordings are kept over time, may allow the model to capture the progression of Parkinson's disease more accurately, potentially enabling the prediction of disease stages.

Another promising direction is experimenting with more advanced machine learning algorithms and deep learning models. This project focused on logistic regression; future work could explore modeling other variants, such as Gradient Boosting Machines, XGBoost, or even deeper models like Convolutional Neural Networks for audio data. The capability of handling sequential voice data fits well within neural networks, particularly RNNs or LSTM networks, which can identify patterns over time.

Integration of feature selection and optimization techniques is another approach to improving the predictive power of the model. In future work, feature selection may be automated using methods like recursive feature elimination (RFE) or genetic algorithms. Hyperparameter tuning can be performed using grid or random search techniques to optimize the performance of the logistic regression model, potentially achieving better accuracy than the current 76.81%.

Another very important direction for future work is to integrate interpretability and explainability into the model. By using SHAP (Shapley Additive Explanations) values or LIME (Local Interpretable Model-agnostic Explanations), it will be possible to understand the features that most

impact the detection of Parkinson's disease. This would not only increase trust in the model but also provide clinicians with actionable insights that bridge the gap between machine learning predictions and real-world clinical decisions.

Future work will focus on further enhancing the accuracy and reliability of this Parkinson's disease detection model by increasing the size of the dataset, exploring advanced models, optimizing feature selection, and incorporating explainability. All these advancements bring us closer to creating handy and non-invasive tools to assist in the early detection and better management of Parkinson's disease.

CHAPTER – 07

REFERENCES

[1] https://en.wikipedia.org/wiki/Parkinson%27s_disease

[2] Mandal, I., & Sairam, N. (2020). New machine-learning algorithms for prediction of Parkinson's disease. *International Journal of Systems Science*, *45*(3), 647-666.

[3] Ahmed, I., Aljahdali, S., Khan, M. S., & Kaddoura, S. (2022). Classification of Parkinson disease based on patient's voice signal using machine learning. *Intelligent Automation and Soft Computing*, *32*(2), 705

[4] Alshammri, R., Alharbi, G., Alharbi, E., & Almubark, I. (2023). Machine learning approaches to identify Parkinson's disease using voice signal features. *Frontiers in artificial intelligence*, *6*, 1084001.

[5] Rahman, S., Hasan, M., Sarkar, A. K., & Khan, F. (2023). Classification of Parkinson's disease using speech signal with machine learning and deep learning approaches.*European Journal of Electrical Engineering and Computer Science*,*7*(2), 2027.

[6] https://www.geeksforgeeks.org/understanding-logistic-regression/

[7] https://machinelearningmastery.com/logistic-regression-for-machine-learning/

[8] https://statquest.org/statquest-logistic-regression/

[9] https://towardsdatascience.com/logistic-regression-model-tuning-with-scikit-learn-part-1-425142e01af5

[10] https://scikit-learn.org/stable/auto_examples/model_selection/plot_confusion_matrix.html