Student Declaration of Authorship

Course code and name:	F21MP Masters Project and Dissertation
Type of assessment:	**Individual**
Coursework Title:	August 2022-23 Dissertation Submission
Student Name:	Abhishek Suresh
Student ID Number:	H00391957

Declaration of authorship. By signing this form:

- **I declare** that the work I have submitted for individual assessment OR the work I have contributed to a group assessment, is entirely my own. I have NOT taken the ideas, writings or inventions of another person and used these as if they were my own. My submission or my contribution to a group submission is expressed in my own words. Any uses made within this work of the ideas, writings or inventions of others, or of any existing sources of information (books, journals, websites, etc.) are properly acknowledged and listed in the references and/or acknowledgements section.

- I confirm that I have read, understood and followed the University's Regulations on plagiarism as published on the University's website, and that I am aware of the penalties that I will face should I not adhere to the University Regulations.

- I confirm that I have read, understood and avoided the different types of plagiarism explained in the University guidance on Academic Integrity and Plagiarism

Student Signature *(type your name):* *Abhishek Suresh*

Date: *20/08/2023*

Copy this page and insert it into your coursework file in front of your title page.
For group assessment each group member must sign a separate form and all forms must be included with the group submission.

Your work will not be marked if a signed copy of this form is not included with your submission.

Exploring Parkinson's Diagnosis Using Feature Shuffling And Logistic Regression

by

Abhishek Suresh

H00391957

Submitted for the degree of

MSc in Data Science

SCHOOL OF MATHEMATICS AND COMPUTER SCIENCE

HERIOT WATT UNIVERSITY

August 2023

Acknowledgements

I would like to thank my supervisor Dr. Alistair, McConnell for all his support since I joined Heriot Watt University, and especially during his guidance on this research plan. I also thank Marta, Vallejo for her valuable help in acquiring and understanding the data used in this project as well as for her valuable insights on different algorithms.

I also thank my Friends and Family for their support and guidance through out.

Figure table

List of Tables

Table of Contents

Glossary

AUC Area Under Curve.

MCC Matthews Correlation Coefficient.

ML Machine Learning.

PCA Principal Component Analysis.

PD Parkinson's Disease.

SVM Support Vector Machine.

Abstract

Parkinson's disease (PD) is a neurological ailment affecting millions of individuals throughout the world. Early identification and proper diagnosis of Parkinson's disease (PD) are critical for optimal therapy and disease management. Machine learning approaches have showed promise in the early identification and diagnosis of Parkinson's disease in recent years.

The goal of this study plan is to look at how feature shuffling and logistic regression might be used to uncover useful characteristics for PD classification.

To do this, I will be utilising a dataset collected from Leeds General Infirmary that includes 26 control persons and 56 patients.

The most essential characteristics found by feature shuffling will be used to develop a prediction model for PD classification using logistic regression. The logistic regression model's accuracy is then accessed using a range of performance indicators, including sensitivity, specificity, and area under the receiver operating characteristic (ROC) curve.

To validate our results, we will compare our model's performance to that of other machine learning approaches typically used in PD classification, such as support vector machines (SVMs).

Ultimately, the goal of this study plan is to contribute to the development of useful diagnostic tools for Parkinson's disease by utilising machine learning approaches. We anticipate that by finding the most significant traits for Parkinson's disease categorization, will increase the accuracy and reliability of PD diagnosis, resulting in improved therapy and disease management.

Chapter 1

Introduction and Objectives

1.1 Introduction

Machine learning uses a variety of algorithms to learn from the data. Every field of research, including robotics, education, and travel as well as health care, has effectively put machine learning into practice. The major use of machine learning methods in healthcare is illness diagnosis. By using machine learning to analyse clinical data, it is possible to diagnose a patient's illness quickly, which will assist to begin treatment for them right away. ML methods are equally capable of diagnosing unusual diseases as they are of diagnosing prevalent diseases.

One such ailment that is exceedingly challenging to identify is Parkinson's disease. Parkinson's disease is a neurological condition that progresses and has both motor and non motor symptoms (DeMaagd Philip, 2015).According to the Parkinson's disease foundation, there are presently 1 million Americans living with the condition (DeMaagd Philip, 2015). As there are no conclusive tests to confirm the diagnosis of PD, a clinical diagnosis necessitates an examination of the patient's medical history.

For this project, I am using a dataset collected from Leeds teaching hospitals from 87 participants, including 29 healthy patients and 58 sick patients. The participants were asked to design an Archimedean spiral pentagon using an inking stylus on a pressure-sensitive tablet (Vallejo et al. 2017). Different data like the hand used (dominant or non-dominant), number of attempts, area of error between template pentagon and pentagon drawn by the subject, etc. were collected as features being processed with the regression model.

In this study, I propose a regression model using feature shuffling for the selection of requisite features, which is then used in a logistic regression model to study the accuracy and effectiveness of predicting Parkinson's disease from the dataset.

Feature shuffling is a type of feature selection method where the values of a specific variable are shuffled and how the permutation affects the performance of the algorithm is observed.

Logistic regression is a better strategy for cases requiring binary and linear classification as it provides discrete outputs which helps in classification. This classification approach works brilliantly with linearly separable classes and is exceedingly easy to use. It is a system of classification that businesses frequently utilize.

Different regression models and feature selection methods are discussed in the Literature Review chapter.

1.2 Aims and Objectives

This project aims to develop a regression model using feature shuffling and logistic regression and compare its predicting accuracy and effects with other developed models in predicting Parkinson's disease. The comparison study mainly considers the predicting accuracy and other evaluation metrics. More specifically, the objectives of this study are:

- Development of a feature shuffling method and logistic regression machine model.
- Filtering and extracting features from the dataset that optimize the accuracy and performance of predicting Parkinson's disease.
- Investigating the effects of feature shuffling on the accuracy of prediction.
- Investigating the effects of different parameter values of logistic regression model on the accuracy of the prediction and run time of the model.
- Comparing the performance of the new model against other logistic regression models and SVM models developed for this same cause.

The rest of the document is organized as follows:
In the literature review section,

1. A summary of Parkinson's disease along with its symptoms and diagnosis is discussed.
2. Existing Diagnosing methods and new technologies used in diagnosing is discussed.
3. Machine learning models and evaluation methods are discussed.

In the methodology section, the methods proposed by defining the requirements and data used for this project are explained. Later the implementation of the project and its evaluations are explained and finally ended with a conclusion along with some insights to the future works.

Chapter 2

Literature Review

This chapter will discuss PD, its treatments and symptoms, and the current medical practices for its diagnosis. Some of the core techniques and ideas in machine learning is then discussed to understand the objectives of this study in detail. Then, the Evaluation methods of different machine learning models are explained with more focus on logistic regression and SVM models. Finally, at the end of this chapter, a critical review of some related works in this area of study is done.

2.1 Parkinson's Disease

PD is a severe degenerative neurological condition for which there is no recognized treatment. About 1-2 percent of those "over 50" are affected by PD, and there are 1.5 million instances in the United States of America alone. Motor deficits caused by resting tremors, bradykinesia, postural instability, and stiffness are among the clinical characteristics of PD. Non-motor symptoms of PD include autonomic, cognitive, and mental issues. (2007; Thomas and Beal, p. 183). One of the most crucial truths is that males are more likely than women to get PD (Korell and Tanner, 2004).

The study from the Office of health economics suggested that around 60,000 to 80,000 people suffer from PD in the United Kingdom. The study also suggests that there were 15,000 patients with Parkinson's in the hospital or residential care, 22,000 handicapped in the community, and more than 30,000 in the community but not handicapped. (Marsden, 1994).

The Study performed by the Association of British Neurologists estimated that within a population of 250,000, at least 400 would be suffering from PD, of whom 342 would have a significant disability (Marsden, 1994). According to the study mentioned in (Marsden, 1994),5 percent of PD is caused due to monogenic causes. That means 5 percent of overall PD is caused due to changes or mutations in a single gene. But casual mutation is not found in the majority of the cases, but the disease is caused due to variety of genetic as well as lifestyle factors in an aging brain.

2.2 Symptom's of Parkinson's Disease

PD causes different nonmotor and motor symptoms. The main classical symptoms of Parkinson's disease include tremors, rigidity, and akinesia (difficulty in starting movement) or bradykinesia (slowness in movement). The tremor is usually regular, rapid, and rhythmic in the case of PD. It occurs at rest and tends to disappear temporarily when tried to move voluntarily (Abudi et al. 1997).

Rigidity is one of the most common symptoms. According to the discussion by (Abudi et al. 1997), two types of muscle tone changes are observed: the clasp knife and cogwheel, or lead-pipe rigidity. Cogwheel rigidity is a type of rigidity where resistance is a rhythmic jerking sensation and lead-pipe rigidity has a uniform resistance. The patient usually uses a masked face because of facial muscle rigidity. The blinking sensation gets reduced and rigidity of the mouth muscles results in dysarthria and causes the patient's voice to get muffled (Abudi et al. 1997).

All these symptoms are mentally distressing and socially embarrassing to both patients and their families. Because of the patient's speech and communication, the patient often feels lonely. Depression is common during such stages, and the patient's appearance may also affected because of the inability to perform basic activities (Abudi et al. 1997).

Not all individuals experience symptoms, and when they do, they can change over time in terms of severity or even go away and come back. Drugs may result in additional symptoms after a protracted course of treatment for the illness. The most common side effect, called dyskinesia, causes some body parts to move unintentionally.

Because of all these factors, each patient's clinical history is distinct (Ahlrichs and Lawo, 2013). PD is also known first to affect one side of the body (the left or right side) more than the other. This asymmetry also has an impact on symptoms (Verreyt et al., 2011). (NationalCollaboratingCentreFor, 2006).

2.3 Current Methods of Diagnosis of Parkinson's Disease

Parkinson's disease is diagnosed mostly on clinical grounds because there is no reliable test. The combination of motor traits and exclusionary symptoms is widely used to build a diagnosis (Rao et al 2003). When patients have a typical presentation, the diagnosis is simple; Nevertheless, distinguishing PD from other forms of syndromes is difficult early in the disease when symptoms overlap with other syndromes (Tolosa et al., 2006).

A number of rating scales are used to assess motor impairment and disability in PD patients; However, the majority of the validity and reliability of these measures has yet to be thoroughly investigated.

- The Hoehn and Yahr scale, which spans from 0 (no signs of disease) to 5 (wheelchair-bound or bedridden unless supported), is widely used to compare groups of patients and provide a general evaluation of disease progression. Nevertheless, the scale is not linear and may not even be ordered, which means that some persons will have more problems with stage 2 than those with stage 3 (Perlmutter, 2009). The rating was later adjusted to add half-scores after stages 1 and 2, although no testing was conducted on this version (Goetz et al. 2003). The Hoehn and Yahir staging is used for subject description, and ratings are based on subject examination (Perlmutter, 2009).

- The most popular is the Unified Parkinson's Disease Rating Scale (UPDRS) (Jankovic, 2008). (NationalCollaboratingCentreFor, 2006). This scale is a subjective method of assessing the severity of the sickness, thus its validity depends on the examiner. It is reliable when utilized by professionals with training, but not when those professionals lack sufficient expertise. Instruments are necessary to make the evaluation objective or to aid professionals in assessing the severity and course of the ailment and better understanding of it (Saunders-Pullman, et al., 2008). The UPDRS has four subscales. Each includes, in that sequence, behaviour and mood, everyday living, clinician evaluation of PD motor demonstration, and therapeutic problems. Data for subscales 1, 2, and 4 are obtained from patients and carers, and the scale value for subscale 3 is determined by examination (Goetz et al. 2003).

- The Schwab and England ADL Scale is a "activities of daily living" assessment that is commonly used to estimate a patient's capacity to function. The rating is completed through interviews with patients and a secondary source. The rating ranges from 0 percent to 100 percent, with 5 percent increments in between (Perlmutter, 2009).

The UPDRS has been used for mass tests for Parkinson's disease (Racette et al. 2009). Parkinsonism relates to clinical assumptions and makes no inferences about the condition's causation. As a result, these screens try to uncover more broad disorders such as drug-induced parkinsonism and PD.

2.4 New Methods and technologies of Diagnosis of Parkinson's Disease

In medical diagnosis, machine learning algorithms are effectively utilised to diagnose illnesses based on a data set collected with the patient's symptoms. By investigating hidden traits or data that are not addressed in the clinical diagnosis of PD, ML approaches can increase diagnostic accuracy (Kononenko, 2001).

Figure 1 depicts the whole flow of the machine learning method for the diagnosis of Parkinson's disease.

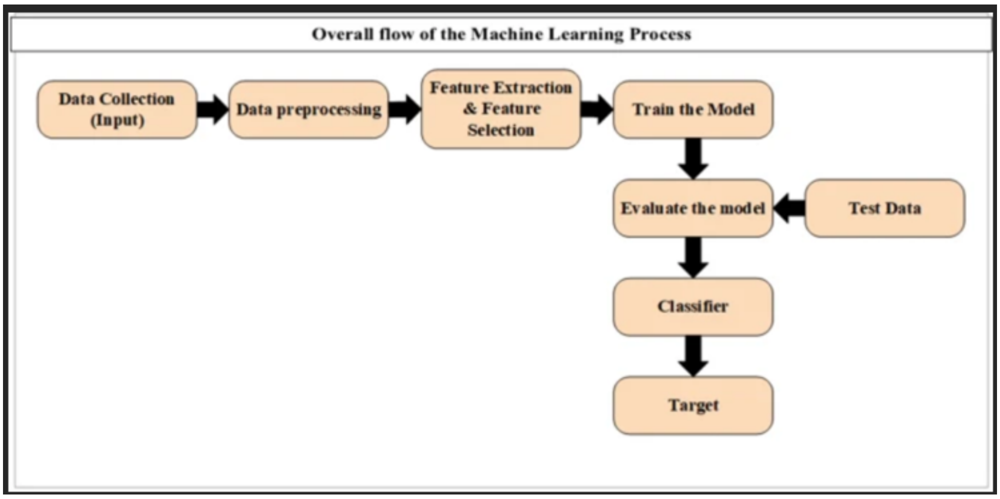


Figure 2.1: Overall Flow of Machine Learning Process(Saravanan et al. 2022)

In the last few years, Machine learning algorithms had a significant role in solving complex problems and have acted as a valuable tool for physicians (Gagliano et al. 2017).

The recent study conducted and explained by (Ali et al. 2019) focused on exploring features and samples from the voice dataset of the patients. Multiple types of samples per subject were used in this study and have obtained a decent accuracy in diagnosing PD.

Modern mobile computing technologies make it easier to measure symptoms objectively over time. As a result, modern systems based on mobile technology can support a variety of monitoring, diagnosis, and rehabilitation applications (Lukowicz 2004). In fact, between 2008 and 2019, the number of papers reporting the use of wearable and mobile technology for PD research multiplied by 100 (Deb et al. [no date]).

Inertial Measurement Units (IMUs), force and pressure plates, biopotential sensors, and optical motion-capturing systems are examples of common gadgets. Accelerometer and gyroscope sensors, which are typically found in IMUs, can capture crucial information for assessing the symptoms. Similarly to this, the force sensors in a force plate give details about the patient's balance and posture. Electroencephalogram (EEG) and electromyogram (EMG) sensors monitor brain activity and muscle reaction, respectively, as a complementing modalities. In contrast, optical motion-capturing devices like Microsoft Kinect and VICON track patient movement while they are walking around. As communication technologies like Zigbee and Bluetooth are increasingly used, connecting these sensors has also become simple (Deb et al. 2021).

Godinho et al. (2016) reviewed 168 articles after searching the PubMed database and grouped the studies based on the type of device used. They hypothesized that the evaluated devices may be used to objectively measure PD symptoms such as postural control, bradykinesia, tremor, freezing, dyskinesia, gait, and daily activity/physical activity. Wearable technology assessments of PD symptoms have become more common in recent years (Rovini et al. 2017). The diagnosis and treatment of PD, as well as other pathologies, have shown promise for wearable sensors.

2.5 Machine Learning Algorithms

The emergence and subsequent growth of technologies like big data, business intelligence, and other applications that call for automation can be linked to the demand for sophisticated machine learning and developing technologies (Alloghani et al., 1970).

Machine learning is a subset of artificial intelligence that employs automated ways to address challenges based on prior facts and data without unduly altering the fundamental mechanism (Sandhu and Itkikar, 1970). Artificial intelligence is the process of making machines intelligent by developing algorithms and employing other computer techniques. It contains algorithms that think, act, and perform jobs in ways that are often beyond the capabilities of humans.

In contrast to AI applications, machine learning finds solutions to issues based on historical or prior examples. Machine learning is the process of discovering hidden patterns in data and applying those patterns to categorize or forecast a problem-related occurrence. For AI devices to work effectively, information is required, and machine learning provides that knowledge. In a word, machine learning algorithms are built into machines so that information may be collected from them and used to speed up and improve processes.

Machine learning techniques can either be supervised or unsupervised, even though some authors also refer to other algorithms as reinforcement learning as they learn data and uncover patterns to an environment. Nonetheless, both supervised and unsupervised machine learning techniques are acknowledged in the majority of research.

2.5.1 Supervised Learning

Machine learning algorithms that need supervision fall under this category. Training and test data sets are created from the input dataset. The output variable that has to be predicted or categorized using a machine learning algorithm will be in the training data set. The training data set in supervisee learning is used to discover data patterns, which are then applied to the test dataset for prediction or classification (Dey 2016).

The workflow of supervised learning and the three most famous supervised learning have been discussed further.

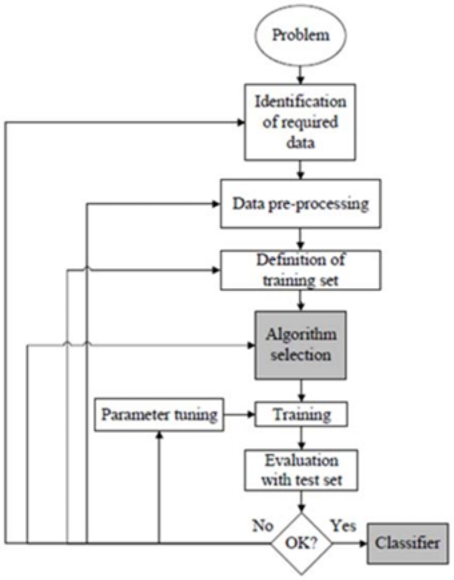


Figure 2.2: Workflow of Supervised Machine learning algorithm (Dey 2016)

The figure shows the typical workflow of a supervised machine-learning algorithm. The data set is split into training and test set and checked whether the knowledge gained from studying the training set can classify the test set correctly. If the classification is gone wrong, then parameter tuning and algorithm change are considered.

The four types supervised machine learning algorithms are mainly:

- Decision Trees
- Naïve Bayes
- Support Vector Machine
- Logistic Regression

Decision Trees

A supervised learning system known as a decision tree organises characteristics by ranking them according to values. It is mostly employed for categorization needs. There are nodes and branches in every tree. The branches indicate the values that the nodes can take, and the nodes represent attributes in a dataset that has to be categorised(Dey 2016).

An illustration of a decision tree algorithm may be seen in the picture below.

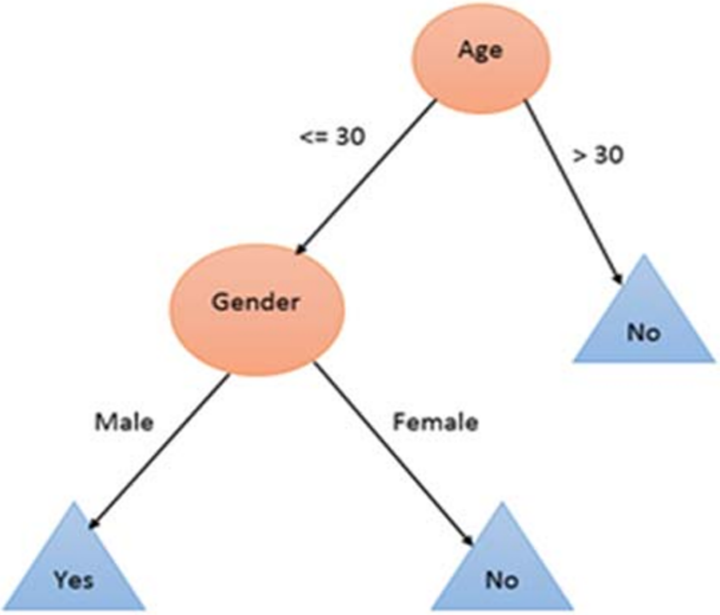


Figure 2.3: Example of Decision Tree Algorithm (Dey 2016)

Decision Tree is used in diagnosing Parkinson's disease in a study conducted by (Aich et al. 2018). The study was done on voice data set and impressive results were also attained by the researchers. An accuracy of 96.83 percent was obtained using random forest model in the study.

Naïve Bayes

The supervised learning technique known as Naive Bayes focuses mostly on text categorization. Its primary usage is for classification and clustering. The conditional probability attribute affects how the algorithm is built. It is comparable to decision trees, however, in this instance, trees are created based on their likelihood of occurring. Bayesian networks are another name for these generated trees (Lowd and Domingos 2005).

Naïve Bayes was used in the study conducted in (Ghazali et al. 2018) for the diagnosis of Parkinson's disease using the voice dataset of 31 subjects. The study obtained a decent results with an accuracy of 89.46 percent.

Support Vector Machine

One of the most modern machine learning methods is the Support Vector Machine (SVM). Once more, categorization is its primary function. SVM operates on the margin calculation principle (Dey 2016). Data are analysed using SVM, which finds patterns in the dataset. In a multidimensional space, SVM creates hyper-planes to demarcate various class boundaries. The margins are drawn to reduce the classification error by increasing the distance between the margin and the separated classes(Dey 2016).

The Support Vector Machine's operation is depicted in the picture below, where two classes are divided between the hyper planes.

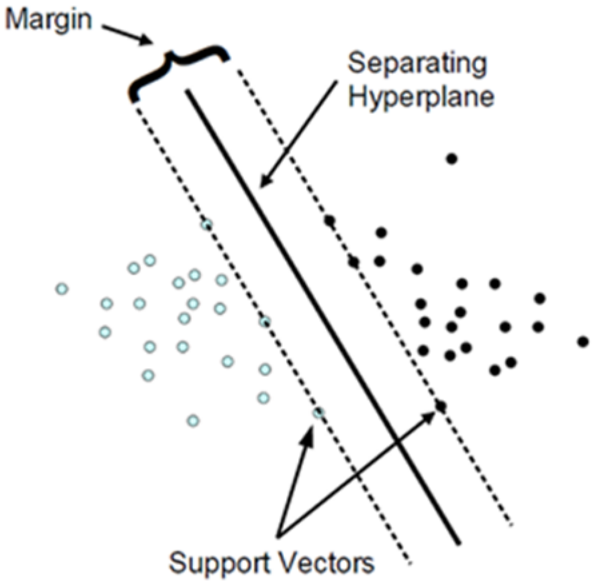


Figure 2.4: Working of Support Vector Machine (Dey 2016)

(Montaa et al. 2018) a database of 27 Parkinson's disease patients and 27 healthy controls and achieved 92.2 percent using SVM algorithm. The study obtained 92.2 percent using 10-fold cross-validation and 94.4 percent in the case of leave-one-out method. The cross-validation methods will be discussed in the coming sections.

SVM's main limitations are its computational complexity, especially for large datasets, and the challenge of selecting appropriate kernel functions and tuning hyperparameters. While SVM often exhibits impressive performance, its models can be less interpretable compared to logistic regression.

SVM has demonstrated its efficacy in various applications. In image recognition, SVM has been employed for facial recognition and object detection. In text classification, it has been utilized for sentiment analysis and spam email detection. SVM has also made significant contributions to the field of bioinformatics, aiding in protein structure prediction.

For example, Cortes and Vapnik (1995) presented SVM as a novel approach for character recognition, achieving superior performance on handwritten digit recognition tasks compared to traditional methods.

Logistic Regression

Logistic regression is a mathematical modeling approach that can be used to describe the relationship of several variables to a single dependent variable. For example, consider a scenario where we need to classify whether an email is a spam or not. There is a need for setting up a threshold value in this case and hence the classification will not be perfect. But in logistic regression, as it uses a sigmoid function their value strictly ranges from 0 to 1 and the classification can be done without any unbound situations.

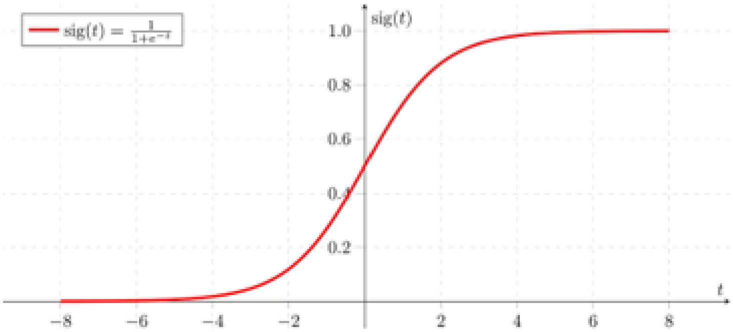


Figure 2.5: Sigmoid Activation Function

As shown in the figure above the sigmoid activation function gives output as either 0 or 1 irrespective of any input given. Thus logistic regression is a popularly used classification model in machine learning.

Despite its advantages, logistic regression assumes a linear relationship between predictors and the log-odds of the outcome. Consequently, its performance can be limited when dealing with data exhibiting complex nonlinear patterns. Additionally, logistic regression is susceptible to the influence of outliers and multicollinearity among predictors.

Logistic regression has found extensive applications in various fields. In medical research, it has been utilized for disease diagnosis, such as predicting the likelihood of a patient having a certain medical condition based on diagnostic tests. In finance, it aids in credit risk assessment by predicting the probability of default for loan applicants. Marketing professionals leverage logistic regression to forecast customer churn, enabling proactive retention strategies.

In a study by Hosmer Jr. et al. (2013), logistic regression was applied to predict the likelihood of post-surgery complications based on patient characteristics, yielding a model with an area under the ROC curve of 0.80.

The choice between logistic regression and SVM depends on several factors, including the nature of the data and the specific problem at hand. Logistic regression excels when the relationship between predictors and the outcome is approximately linear, and when interpretability is crucial. SVM, on the other hand, is particularly adept at handling nonlinear relationships through its kernel trick, making it a suitable choice when the data is more complex.

2.5.2 Unsupervised Learning

As new data is supplied, the unsupervised learning algorithm employs previously learned features to identify the class of the data. It learns a few characteristics from the dataset. It is mostly utilised for feature reduction and the clustering of the data process. The two primary unsupervised learning techniques that are used for clustering and dimensionality reduction are mentioned here.

K-Means Clustering

Clustering is a kind of unsupervised learning approach that, when used, automatically forms groups or clusters. The objects or members that share the same qualities are grouped or clustered together. The number of different clusters produced by the algorithm is indicated by the letter K in the K-means clustering formula. The center of a cluster is determined by the mean of its values. All of the members are divided into several groups or clusters because those who have values that are close to the mean value are grouped. The data are depicted below in a dispersed layout both before and after the clustering technique has been applied (Shalev-Shwartz et al. [no date]).

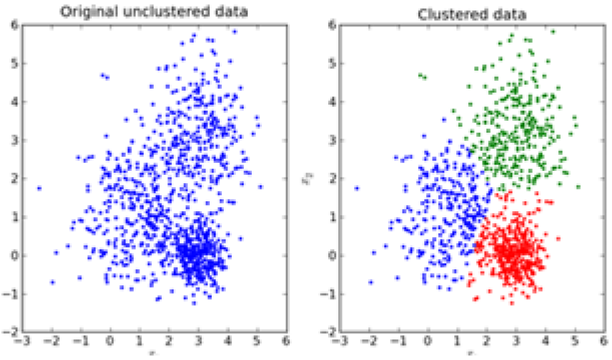


Figure 2.6: K-Means Clustering (Dey, 2016)

Principal Component Analysis

The Principal Component Analysis or PCA is used as a dimension reduction algorithm to make computations faster and easier. As explained in (Harrington [no date]). To Understand the working an example of 2D data is taken. 2D data when plotted in a graph takes two axes. After PCA is applied to the data the dimensionality of the data is reduced and the data becomes 2D. This is done by choosing a coordinate system that covers most of the data by interpreting the direction with the most variance and largest variance of the data.

The figure below shows the data representation before and after applying PCA.

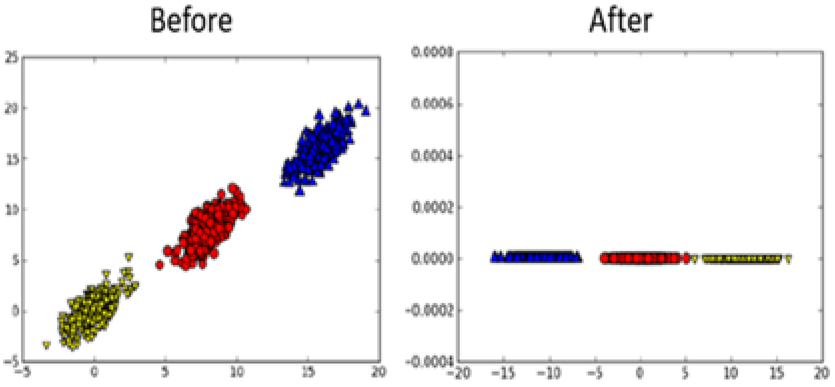


Figure 2.7: Visualisation of Data before and after applying PCA (Dey, 2016)

2.6 Evaluation of Machine Learning Algorithms

There are a variety of things to consider while evaluating classifications of machine learning algorithms. High accuracy does not necessarily indicate that the algorithm is performing as intended or anticipated. For a testing dataset, let's say 90 percent of instances belong to a specific class, and running the algorithm it will give us an accuracy of 90 percent, however, this classifier would likely perform poorly when presented with data other than the test data. When the test set is too small to be statistically significant, confidence boundaries are calculated using statistical techniques (Azuaje 2006). Below are some of the methods used to evaluate machine learning algorithms, in this study we will be using these methods to evaluate the outcome of the study.

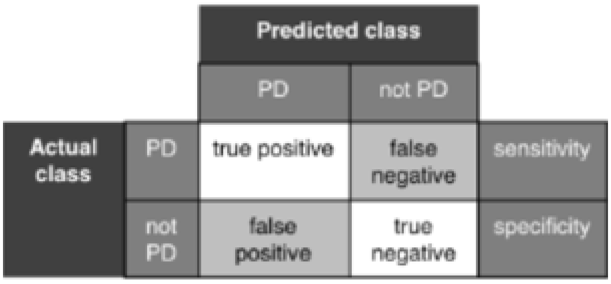
Cross-validation

Cross-validation is one of the widely used data sampling methods to estimate the prediction error and to tune the parameters of the model. The basic concept of Cross-validation is to split the given data set into different groups of equal size. The data division is done randomly. Suppose the data is split into n different groups then the algorithm used is also trained and tested n times. The algorithm is trained using n-1 of the groups and then it's tested on the remaining one. The remaining one which is used for testing is a different one in each of the n times. The accuracy is then calculated as an average of all the n executions (Kohavi, 1995).

The method is easy to use when the amount of data is limited. The perfect or standard method is to hold one-third of the data for testing and use the rest for training. Extensive tests on many datasets have shown that the amount in which the data is to be split and run is 10 as it is the right number to get the best estimate of error(Azuaje 2006).

Confusion Matrix

The confusion matrix is a type of matrix that is used in machine learning algorithms to calculate the sensitivity, specificity, and error rates of the algorithm. It's an n*n matrix that shows the predicted and actual classification values of the algorithm. The figure below shows this matrix and the equations to calculate sensitivity and specificity are given below.



		Predicted class		
		PD	not PD	
Actual class	PD	true positive	false negative	sensitivity
	not PD	false positive	true negative	specificity

Figure 2.8: Confusion Matrix

Sensitivity and specificity are two main values that are considered for checking the accuracy and error rate of an algorithm model.

Sensitivity

The sensitivity of a model is defined as the ability of the model to correctly identify or classify the positive class or group which is of interest to the study.

In this case, the ability to detect people with PD properly.

$$sensitivity = \frac{true\ positives}{true\ positives + false\ negatives}$$

Specificity

The specificity of a model is defined as the ability of the model to correctly identify or classify the negative classes or groups which is not of interest to the study.

In this case, the ability to detect people without PD or healthy people.

$$specificity = \frac{true\ negatives}{true\ negatives + false\ positives}$$

Quality Measures for Models

Many people believe that the ratio of correctly categorised samples to total number of samples is the most appropriate performance statistic. This is just due to the model's accuracy. Accuracy, on the other hand, cannot be deemed dependable since it produces an overly optimistic estimate when the dataset has an imbalanced amount of data (Chicco and Jurman 2020)

The Matthews Correlation Coefficient is a prominent statistic for dealing with this problem (MCC). It returns a value between -1 and 1, with -1 indicating that all predictions are exactly the opposite of what should be provided and 1 indicating that all predictions are correctly returned.

Using the equation below, MCC may be calculated straight from the confusion matrix.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

Another technique used to find the performance of classifiers is the ROC curves. ROC curves show how well a model performed without considering the class distribution of the data set. It's a graphical representation that shows the trade-off between the true positive values (vertical axis) and false positives (horizontal axis). Plotting ROC curves for different curves makes it easy to compare them visually (Witten et al. [no date])

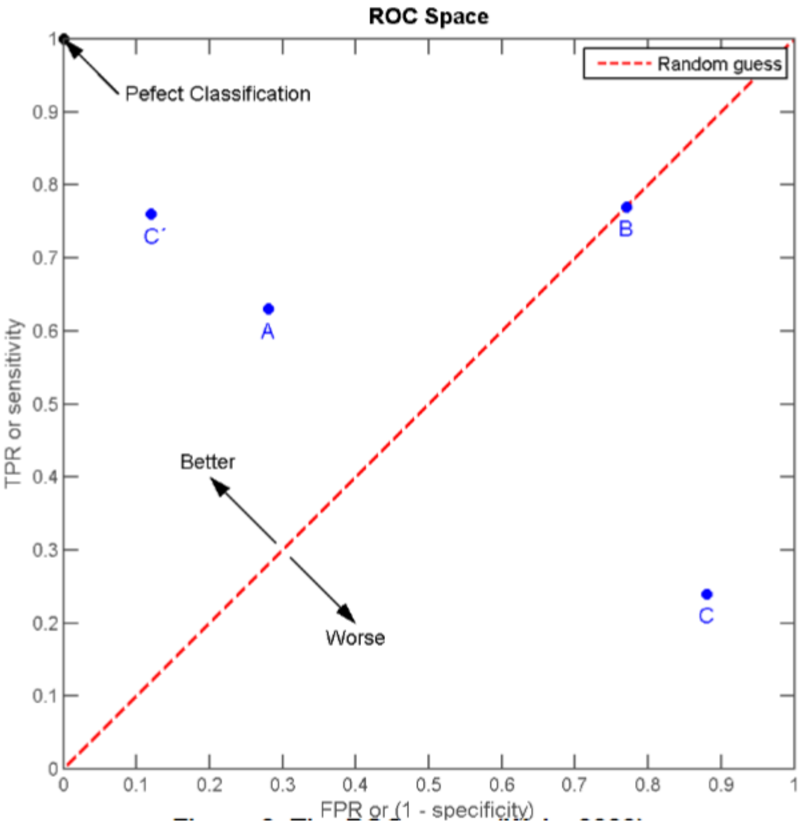


Figure 2.9: ROC Space Curve (Dey, 2016)

2.7 Selecting Features

When using machine learning algorithms for classifying a dataset, selecting the fields or data which are needed for the purpose is very important. If enough fields or data are not used, then the model will not have enough information to work. But using all the available fields or data is not a great idea. Adding unwanted data to the model will affect its accuracy as noise is added. The number of fields or data to be used depends on each problem and it needs to be calculated approximately. Many methods have been discussed and analyzed in (Čehovin and Bosnić 2010). Some of the feature selection methods discussed are ReliefF, Random Forest, Sequential backward selection, Sequential forward selection, and feature shuffling.

- RelefF is a type of feature selection algorithm which aims in finding the quality of features or fields by finding how well their values separate the classes by distance in the problem space.

- The random forest as a feature quality estimation method is based on the difference in performance between the original data set and the changed data set, in which the model randomly permutes observed feature values across cases. The algorithm estimates the relevance of the modification by monitoring performance before and after the change.

- Backward elimination is another term for sequential backward selection. Backward selection begins with the initial data set. Following each model iteration, the least significant feature is identified as the one that produced the least decline or the greatest gain in the classifier's performance. This feature is then removed from the data set, and the method is continued until the minimum number of features necessary is met, or specific criteria are met.

- Sequential forward selection is the inverse of sequential backward selection. Sequential forward selection begins with an empty data set and proceeds in each iteration by extending the data set with the feature that improves model performance the greatest.

- Feature shuffling selects important features, if a random permutation of their values decreases the model performance. If the feature is predictive, a random shuffle of the values across the rows, should return predictions that are off the truth. If the feature is not predictive, their values should have a minimal impact on the prediction.

2.8 Related Work

Several research on the early identification of Parkinson's disease have been conducted utilising various data sets. PD can be recognised in its early stages, according to research done by (Almeida et al. 2019).

(Montaa et al. 2018) investigated the articulatory properties of plosive consonants in a database of 27 Parkinson's disease patients and 27 healthy controls and achieved 92.2 percent using 10-fold cross-validation and 94.4 percent in the case of leave-one-out method. To discover the optimum feature subset, they employed Support Vector Machines (SVM) for classification and a sequential backward feature selection. (Montaa et al. 2018) used a voice data set for their study and had obtained pretty good accuracy output. They have used VOT extraction algorithm in extracting features and sequential backward feature elimination method for feature analysis.

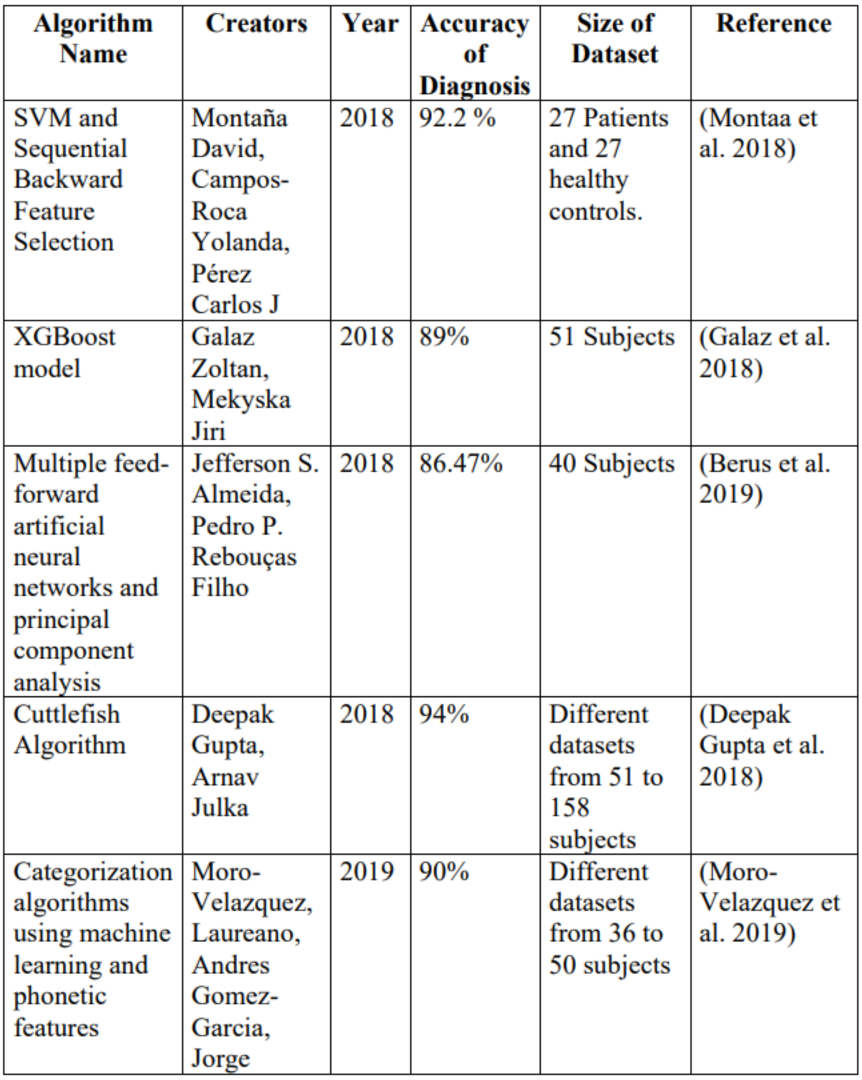
In contrast to that (Galaz et al. 2018) trained XGBoost models to predict changes in clinical ratings using phonation data from 51 Parkinson's disease patients and was able to produce a results with error range from 11 to 26 percent.

(Berus et al. 2019) achieved an accuracy of 86.47 by using feature selection algorithms based on Pearson's and Kendall's correlation coefficients, principal component analysis (PCA), self-organizing maps (SOM), and multiple feed-forward artificial neural networks.

Another technique includes improving the cuttlefish algorithm for illness detection, but with only approximately 94 percent accuracy (Deepak Gupta et al. 2018). This study was done on different data sets like speech and voice datasets with subjects ranging from 31 to 158.The results obtained from this study is really impressive considering the dataset size.

(Moro-Velazquez et al. 2019) studied unique categorization algorithms that combined machine learning with a variety of phonetic features. Although the accuracy rate remains below 90 percent.

A summary of comparison of all the above mentioned algorithms and study done on this field are given in the below table.



Algorithm Name	Creators	Year	Accuracy of Diagnosis	Size of Dataset	Reference
SVM and Sequential Backward Feature Selection	Montaña David, Campos-Roca Yolanda, Pérez Carlos J	2018	92.2 %	27 Patients and 27 healthy controls.	(Montaa et al. 2018)
XGBoost model	Galaz Zoltan, Mekyska Jiri	2018	89%	51 Subjects	(Galaz et al. 2018)
Multiple feed-forward artificial neural networks and principal component analysis	Jefferson S. Almeida, Pedro P. Rebouças Filho	2018	86.47%	40 Subjects	(Berus et al. 2019)
Cuttlefish Algorithm	Deepak Gupta, Arnav Julka	2018	94%	Different datasets from 51 to 158 subjects	(Deepak Gupta et al. 2018)
Categorization algorithms using machine learning and phonetic features	Moro-Velazquez, Laureano, Andres Gomez-Garcia, Jorge	2019	90%	Different datasets from 36 to 50 subjects	(Moro-Velazquez et al. 2019)

Table 2.1: Comparison Table of most prominent research studies and algorithms used for the diagnosis of Parkinson's Disease

Summary

Many works have been done on detecting Parkinson's disease using SVM, Multi Layer Perceptron and many other algorithms as discussed above. Many have used phonetic features in their study and have not explored the physical characteristics of the patients. In this project, I am analysing and trying to study the effects and results of using logistic regression as a model using image datasets obtained by asking patients to trace a pentagon spiral and explore the prediction of Parkinson's disease. I believe logistic regression is something which is less explored in this field and thus decided to analyse the topic using logistic regression and I aim to obtain accuracy and results better than the models discussed above. As (Montaa et al. 2018) had shown a better results the given data set will be also used in a SVM model to compare the results obtained and processing time.

Chapter 3

Methodology

This chapter describes the approach that will be adopted to investigate the effectiveness of this proposed method. The general outline is to develop the proposed model and then compare the outcomes with other models available. We will develop the model in iterations, which will help us in validating the model in each stage and we will add or remove features or other aspects according to the outcomes obtained.

3.1 Software

I will be using Python as the base language in developing the proposed model. We will be using some Python libraries to implement the model. The libraries which will be used are listed below along with a small summary.

NumPy

NumPy library brings computational power to programming languages like Python. It's used to implement dimensional arrays, random number generation, and other algebra routines. NumPy's high-level syntax helps programmers as it is easy to use and implement.

Scikit-learn

Scikit-learn is an open-source machine-learning library that enables the user to implement supervised and unsupervised learning. It also helps in implementing different other tools for model selection, model evaluation, and feature selection.

3.2 Data

Leeds General Infirmary provided the information which is being used for this project. It is the property of 41 control individuals and 48 patients.

Participants were asked to follow the template of the pentagon spiral as shown in figure below.

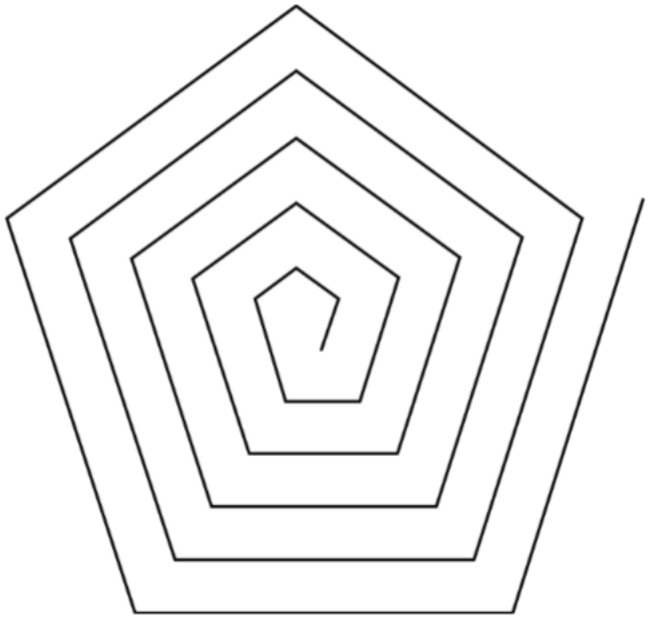
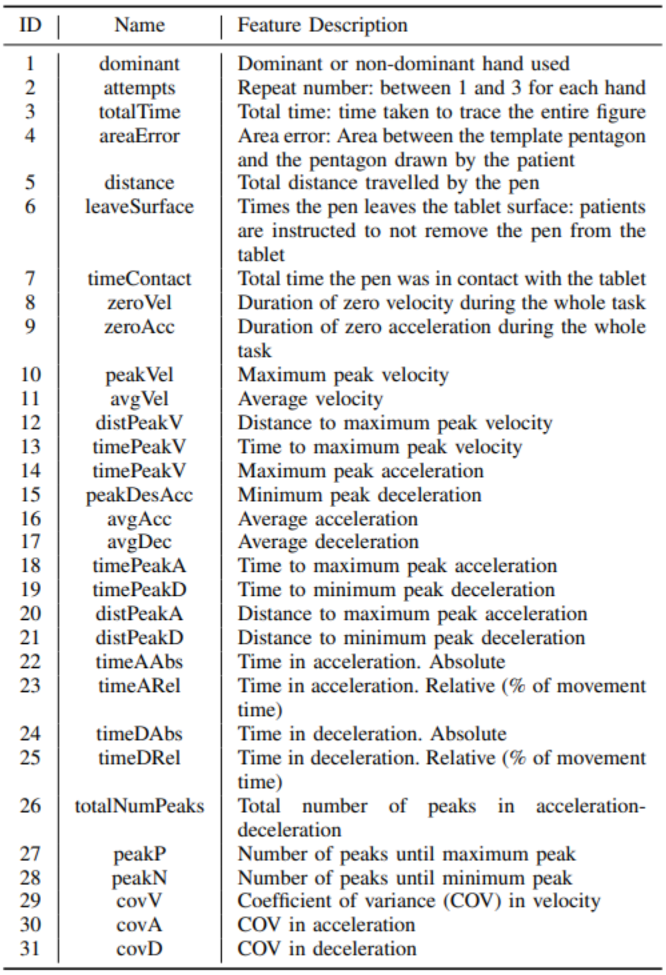


Figure 3.1: Pentagon spiral template used by participants (Vallejo et al. 2017)

When they worked on the design on an A4-sized Wacom digitising tablet, their motions were digitised and recorded to a disc file as a time series of <x,y>coordinates, together with accompanying pen pressure and orientation information. The sampling frequency of the tablet was set to 200Hz. Each person was told to trace three times in succession using their dominant and non-dominant hands.

Figure below summarises the list of characteristics extracted from each drawing. With the exception of 1 and 2, the majority of the characteristics were retrieved automatically from tablet recordings.



ID	Name	Feature Description
1	dominant	Dominant or non-dominant hand used
2	attempts	Repeat number: between 1 and 3 for each hand
3	totalTime	Total time: time taken to trace the entire figure
4	areaError	Area error: Area between the template pentagon and the pentagon drawn by the patient
5	distance	Total distance travelled by the pen
6	leaveSurface	Times the pen leaves the tablet surface: patients are instructed to not remove the pen from the tablet
7	timeContact	Total time the pen was in contact with the tablet
8	zeroVel	Duration of zero velocity during the whole task
9	zeroAcc	Duration of zero acceleration during the whole task
10	peakVel	Maximum peak velocity
11	avgVel	Average velocity
12	distPeakV	Distance to maximum peak velocity
13	timePeakV	Time to maximum peak velocity
14	timePeakV	Maximum peak acceleration
15	peakDesAcc	Minimum peak deceleration
16	avgAcc	Average acceleration
17	avgDec	Average deceleration
18	timePeakA	Time to maximum peak acceleration
19	timePeakD	Time to minimum peak deceleration
20	distPeakA	Distance to maximum peak acceleration
21	distPeakD	Distance to minimum peak deceleration
22	timeAAbs	Time in acceleration. Absolute
23	timeARel	Time in acceleration. Relative (% of movement time)
24	timeDAbs	Time in deceleration. Absolute
25	timeDRel	Time in deceleration. Relative (% of movement time)
26	totalNumPeaks	Total number of peaks in acceleration-deceleration
27	peakP	Number of peaks until maximum peak
28	peakN	Number of peaks until minimum peak
29	covV	Coefficient of variance (COV) in velocity
30	covA	COV in acceleration
31	covD	COV in deceleration

Figure 3.2: Features extracted from the digitised drawings (Vallejo et al. 2017)

3.3 Development

The proposed model will be developed following the basic software development criteria. Initially the data is analysed to know which features are more important for the study. Then Recursive feature shuffling algorithm is used to shuffle between the features extracted and run the logistic regression model developed.

The model will be developed in three phases: First phase will be strictly allocated for analysing the data set to obtain insights and idea on what data we have and how those data can be invoked to obtain the results we are looking for.

The second phase is the development phase where the data after analysing is used on a logistic regression model and the results are analysed and the model is fine tuned until satisfied results are obtained.

The third and final phase is the evaluation of the model. The model is evaluated using different evaluation methods and if the evaluation is not up to the mark then the progress is rolled back to phase 2 to readjust the parameters of the model.

3.4 Evaluation

The developed model will be evaluated using evaluation methods mentioned in the literature review section. Confusion matrix and ROC curve is used to evaluate the model. Also values for accuracy, sensitivity and specificity of the model is obtained. The model will be run for n number of times and an average value will be taken as results of the model. This is done as I am using feature shuffling technique as our selection algorithm and running the model for n number of times helps in obtaining better results and taking an average helps to generalise the result of the model.

Chapter 4

Requirement Analysis

The basic requirement of this project is to find a more accurate and effective way to diagnose Parkinson's disease. Machine learning models are used to try and achieve this.

4.1 Functional Requirements

Must Have:

- The model must be able to accurately distinguish between patients and control subjects.
- The model must provide details on how predictions are made.
- The developed model must have accuracy as near as possible to 100 percent.
- The model should be sensitive enough to detect the disease even in the early stages.
- The model must enable the user or doctor to understand the results, including the symptoms and how they are related to the disease.

Should Have:

- The model should be able to differentiate between different types of diseases, if applicable.
- The model should have a user-friendly interface that can be easily navigated by users.
- The model should be able to provide feedback on how to improve accuracy if the predictions are incorrect.

Could Have:

- The model could incorporate additional features or data sources to improve accuracy.

- The model could have a visualization tool that helps to identify patterns in the data.
- The model could provide insights into the disease that could lead to better treatment options.

Won't Have:

- The model won't provide medical advice or treatment recommendations.
- The model won't be able to predict outcomes for individual patients.

4.2 Non-Functional Requirements

- Speed: The model developed should run faster than the other comparing models.
- Usability: The developed model should be easy to use and configure. The results obtained from the model should be also easy to understand and use.

The assessment of the model is another key part of the non-functional need. The accuracy and speed of the produced model will be the primary assessment criterion for determining its efficacy.

The number of successfully categorised occurrences divided by the total number of instances is the accuracy. The model will be run ten times with various training and test data to determine the average, minimum, and maximum accuracy. These accuracy levels will be compared to the other models to assess the relevance of any performance improvement.

The created model's run time is recorded. The average, lowest, and maximum run duration will be recorded and compared to other available models, much like the accuracy.There are two types of run time: training time and prediction time. The goal is to keep the forecast time as short as feasible.

Chapter 5

Professional, Legal, and Ethical Issues

This section explains different professional, legal, and ethical issues in doing this project and how are these issues mitigated. Professional and legal issues include using third-party software or using any copyrighted content etc. Ethical issues focus on sensitive datasets which do not contain the personal information of users etc. All the practices done in this project is by following the standards of BCS (British Computer Society).

5.1 Professional Issues

Any papers, code, pictures, and libraries used in this project will be cited and used in accordance with the requirements of the publisher licences. The model will be recorded and evaluated in accordance with professional standards. The code will be created with coding standards and design principles in mind. Any code obtained from legitimate third-party sources will be appropriately credited.

5.2 Legal Issues

The model developed will strictly compile with the data security policies. The model will not indulge in any privacy breaches or unauthorized access. The project will not store any derived or inferred information of the user's sensitive characteristics and will not breach any patents.

5.3 Ethical Issues

This research-based project uses data collected from Leeds teaching hospitals from 87 participants, including 29 healthy patients and 58 sick patients. The data only contains the data related to the pentagon drawing exercise done by the patients and no other sensitive data of users are included or used in this project. Thus, there is no ethical issue violation regarding the project.

Chapter 6

Implementation

In this part, I go through the specifics of how I have used feature shuffling and logistic regression to investigate Parkinson's disease.A detailed breakdown of the essential components and strategies employed in this research is explained.

6.1 Data Preparation

PD data set was obtained from Leeds General Infirmary comprising clinical information of 29 control subjects and 58 patients. The controls were friends and relatives of the patients of similar ages,and were only included if they had no neurological disorder. The subjects were asked to draw an Archimedean spiral pentagon, using an inking stylus on a pressure-sensitive tablet. The data set contains both numerical and category variables.

The dataset was preprocessed as needed, dealing with missing values, scaling numerical features, and encoding categorical variables. I made certain that the dataset was suitably divided into features (X) and the target variable (y).

6.1.1 Formatting Data

The Data set obtained had following fields as below:

- ID : Id number for both patients and control subjects
- Dominant : Indicates whether its dominant or non dominant hand (0 - non dominant, 1 - dominant)
- Attempts : Number of attempts
- Duration : Time taken for the entire experiment
- Time : Time taken to complete the spiral
- AreaError : Error in area between the pentagon drawn and the typical pentagon spiral.

- TimeTriangles 1 : Time taken to complete first pentagon spiral
- TimeTriangles 2 : Time taken to complete second pentagon spiral
- TimeTriangles 3 : Time taken to complete third pentagon spiral
- TimeTriangles 4 : Time taken to complete fourth pentagon spiral
- TimeTriangles 5 : Time taken to complete fifth pentagon spiral
- Distance : Distance travelled by the pen in drawing the pentagon spiral
- LeaveSurface : Number of times the pen had left the surface while drawing
- TimeContact : Time in which the pen was in contact with the surface
- ZeroVel : Time in which there was zero velocity by the pen
- ZeroAcc : Time in which there was zero accelaration by the pen
- PC : Indicates whether the user is a patient or control subject (0 - control subject , 1 - patient)

Each ID had 4 rows of data which consists of 2 attempts each on both dominant and non-dominant hands.

The data was then split into two files for for both first attempt and second attempt to study and analyse what difference it makes. The files are saved as '1st attempt.csv' and '2nd attempt.csv' in the folder.

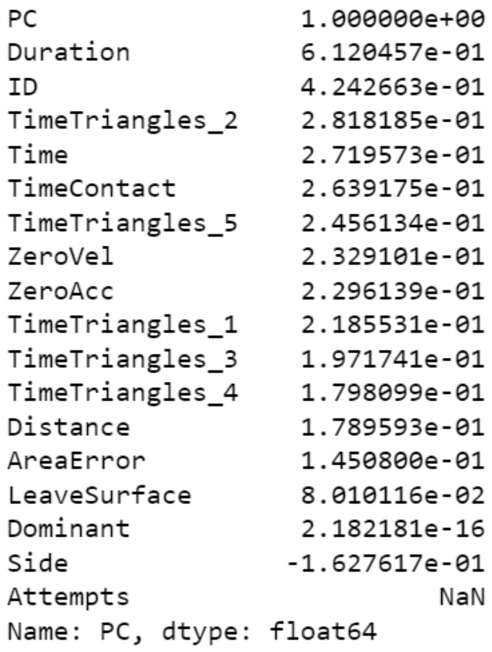
Both these files are then used to build a model using feature shuffling and logistic regression to obatain the accuracy of prediction and other metrics like sensitivity, specificity,Precision, recall and ROC curves.

6.1.2 Data Analysing

The data set after splitting into two was then analysed to find any trends within the data set.

The target variable here is the column PC which shows whether the data belongs to a control subject or patient. So correlation matrix was generated with correlation coefficients of every pair of features with the target feature.

The below image shows the correlation values of the features with the target feature.



```
PC                  1.000000e+00
Duration            6.120457e-01
ID                  4.242663e-01
TimeTriangles_2     2.818185e-01
Time                2.719573e-01
TimeContact         2.639175e-01
TimeTriangles_5     2.456134e-01
ZeroVel             2.329101e-01
ZeroAcc             2.296139e-01
TimeTriangles_1     2.185531e-01
TimeTriangles_3     1.971741e-01
TimeTriangles_4     1.798099e-01
Distance            1.789593e-01
AreaError           1.450800e-01
LeaveSurface        8.010116e-02
Dominant            2.182181e-16
Side               -1.627617e-01
Attempts                     NaN
Name: PC, dtype: float64
```

Figure 6.1: Correlation matrix of the features with target variable ’PC’

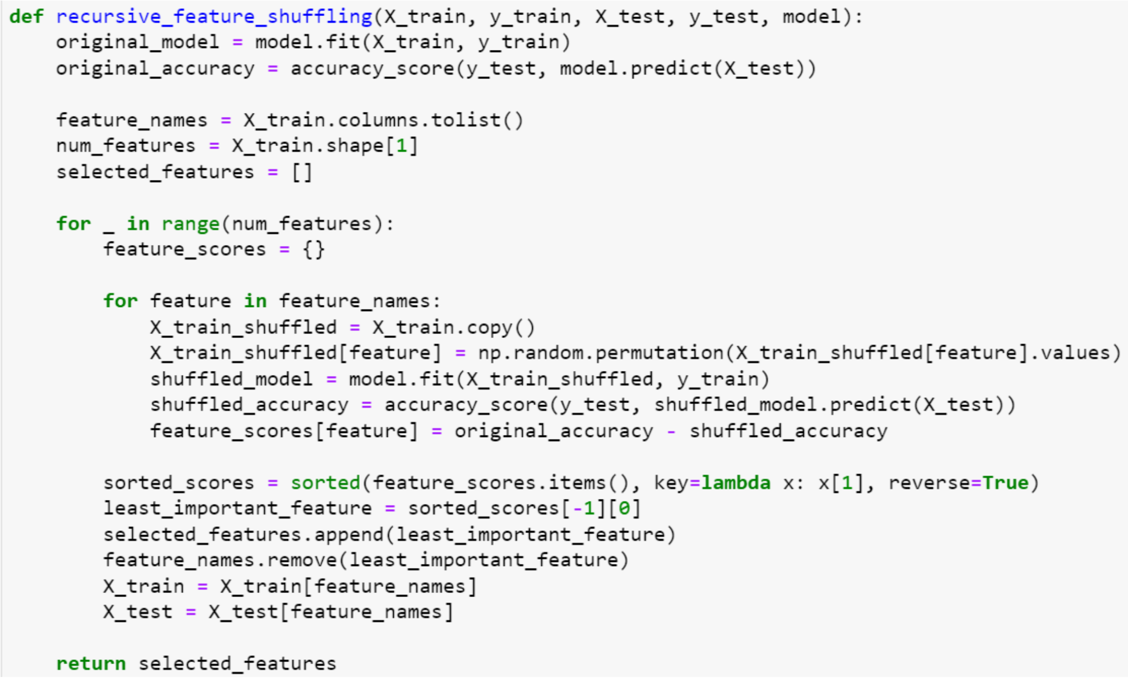
The feature ’ID’ can neglected in this as it only shows the ID of the person who has participated in the experiment and has nothing to do with any diagnosis of the disease.

From the image It can be seen that the features ’Duration’, ’TimeTriangles 2’, ’Time’, ’TimeContact’, and ’TimeTriangles 5’ shows some positive correlation and the feature ’Side’ shows a negative correlation. The data was then split into 3:2 ratio where 60 percent of data was used for training the model and the rest 40 percent for testing the trained model.

6.2 Feature Shuffling

The feature shuffling approach was used to examine the significance of characteristics in predicting Parkinson’s disease. The purpose was to quantify the influence of each feature on model performance by randomly permuting the values of a single feature while leaving others unchanged.

The feature shuffling was implemented follows:



```
def recursive_feature_shuffling(X_train, y_train, X_test, y_test, model):
    original_model = model.fit(X_train, y_train)
    original_accuracy = accuracy_score(y_test, model.predict(X_test))

    feature_names = X_train.columns.tolist()
    num_features = X_train.shape[1]
    selected_features = []

    for _ in range(num_features):
        feature_scores = {}

        for feature in feature_names:
            X_train_shuffled = X_train.copy()
            X_train_shuffled[feature] = np.random.permutation(X_train_shuffled[feature].values)
            shuffled_model = model.fit(X_train_shuffled, y_train)
            shuffled_accuracy = accuracy_score(y_test, shuffled_model.predict(X_test))
            feature_scores[feature] = original_accuracy - shuffled_accuracy

        sorted_scores = sorted(feature_scores.items(), key=lambda x: x[1], reverse=True)
        least_important_feature = sorted_scores[-1][0]
        selected_features.append(least_important_feature)
        feature_names.remove(least_important_feature)
        X_train = X_train[feature_names]
        X_test = X_test[feature_names]

    return selected_features
```

Figure 6.2: Recursive Feature Shuffling Implementation

- The recursive feature shuffling function takes the training and testing datasets (X train, y train, X test, y test) and a model object as input. It aims to estimate the importance of features by recursively shuffling and evaluating them.

- The original model is fitted on the training data, and the accuracy is calculated on the test data to establish a baseline accuracy for comparison.

- feature names is a list of column names (feature names) from the training data, and num features stores the total number of features.

- The outer loop iterates num features times to select one feature at each iteration.

- For each feature in feature names, the algorithm shuffles the values of that feature in the training data by using np.random.permutation. This creates a new dataset X train shuffled with the shuffled feature.

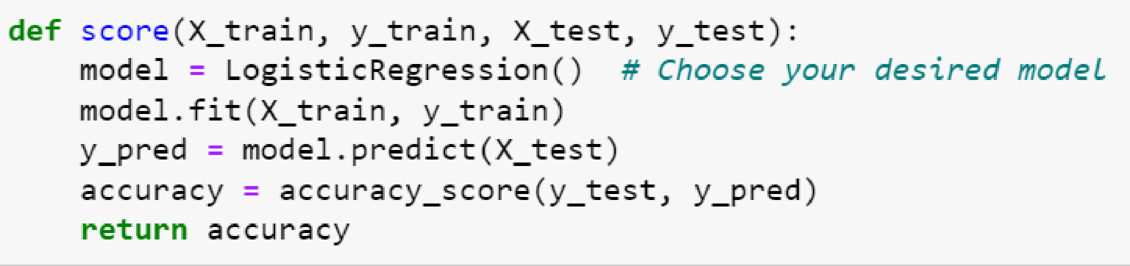
- The shuffled data is used to fit a new model (shuffled model) using the same classifier as the original model.

- The accuracy of the shuffled model is calculated on the test data, and the difference between the original accuracy and shuffled accuracy is computed. This difference represents the importance score for the particular feature.
- The feature scores are stored in the feature scores dictionary with the feature name as the key.
- The feature scores dictionary is sorted in descending order based on the importance scores, and the least important feature is selected (least important feature).
- The least important feature is appended to the selected features list, and the feature name is removed from feature names and the corresponding columns are removed from X train and X test to update the feature set for the next iteration.
- After iterating through all features, the selected features list containing the feature names in the order of their importance is returned as the final result.

6.3 Logistic Regression

The Logistic Regression model was used to examine the accuracy in predicting Parkinson's disease. The purpose was to obtain accuracy, sensitivity and specificity in predicting Parkinson's disease from the data set provided.

Logistic regression was implemented as follows:



```
def score(X_train, y_train, X_test, y_test):
    model = LogisticRegression()  # Choose your desired model
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    return accuracy
```

Figure 6.3: Logistic Regression Implementation

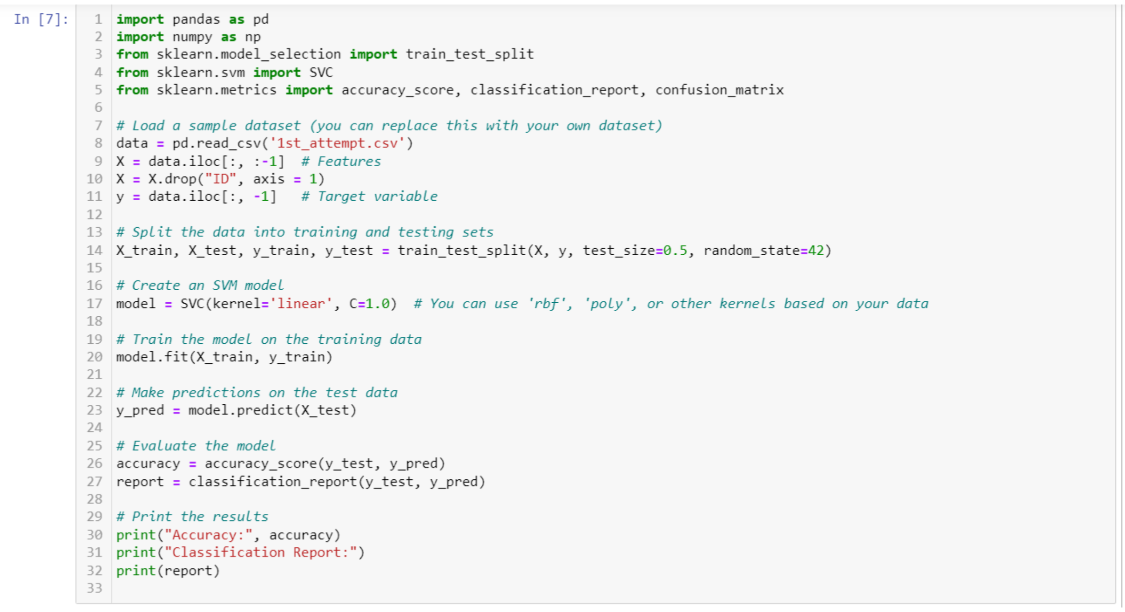
- The score function takes X train, y train, X test, and y test.These parameters represent the training features, training labels, testing features, and testing labels, respectively.
- An instance of the logistic regression model is then created.

- The logistic regression model is then trained using the training data.The fit method fits the model to the training data, allowing it to learn the patterns and relationships between the features (X train) and the corresponding target labels (y train).
- The trained model is then used to make predictions on the testing data. The predict method takes the testing features (X test) as input and returns the predicted labels based on the learned patterns from the training phase.
- The accuracy of the model's predictions is then obtained by comparing the predicted labels (y pred) with the true labels from the testing data (y test).The accuracy score function, provided by scikit-learn, computes the accuracy as the fraction of correctly classified instances.
- The calculated accuracy value is then obtained as the output of the score function.

6.4 Support Vector Machine (SVM)

The Support Vector Machine model was used to analyse the same data set as it is one of the prominent machine learning model used in the industry for binary classifications. In this project I am planning to do a comparison study on the results obtained using both logistic regression model as well as SVM.

The Support Vector Machine model was implemented as follows:



```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix

# Load a sample dataset (you can replace this with your own dataset)
data = pd.read_csv('1st_attempt.csv')
X = data.iloc[:, :-1]  # Features
X = X.drop("ID", axis = 1)
y = data.iloc[:, -1]   # Target variable

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.5, random_state=42)

# Create an SVM model
model = SVC(kernel='linear', C=1.0)  # You can use 'rbf', 'poly', or other kernels based on your data

# Train the model on the training data
model.fit(X_train, y_train)

# Make predictions on the test data
y_pred = model.predict(X_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)

# Print the results
print("Accuracy:", accuracy)
print("Classification Report:")
print(report)

```

Figure 6.4: Support Vector Machine Implementation

- Necessary libraries are imported to use SVM model which includes: 'pandas','numpy','sklearn' etc.
- A SVM model instance is then created using svc function in which we have used kernel type as 'linear' to have a linear model SVM.
- The data set is then split to training and testing parts
- The SVM model is then trained using the training part of the data set.
- This trained model is then used to predict output using the testing part of input data.
- accuracy and other evaluation metrics are then obtained by comapring the predicted outputs with the test outputs of the data set.

6.5 Evaluation of Models

The models developed where evaluated using confusion matrix and ROC curves.

The confusion matrix provides the predicted and actual classification values of an algorithm.

The ROC curve shows how well a classifier works and it is a plot between the True positive and False positive values predicted by the model. It can help users to understand the performance of the model visually.

The models were evaluated mainly using these two methods and some evaluation metrics are also calculated from the values obtained from the confusion matrix to analyse more on the results obtained.

Evaluation Metrics Calculated:

- Sensitivity: The sensitivity of a model is defined as the ability of the model to correctly identify or classify the positive class or group which is of interest to the study.In this case, the ability to detect people with PD properly.

  Sensitivity = TP/(TP+FN), where TP is true positives and FN is false negatives.

- Specificity: The specificity of a model is defined as the ability of the model to correctly identify or classify the negative classes or groups which is not of interest to the study.In this case, the ability to detect people without PD or healthy people.

  Specificity = TN/(TN+FP), where TN is true negatives and FP is false positives.

- Precision: Precision is the quality of positive predictions made by the model.

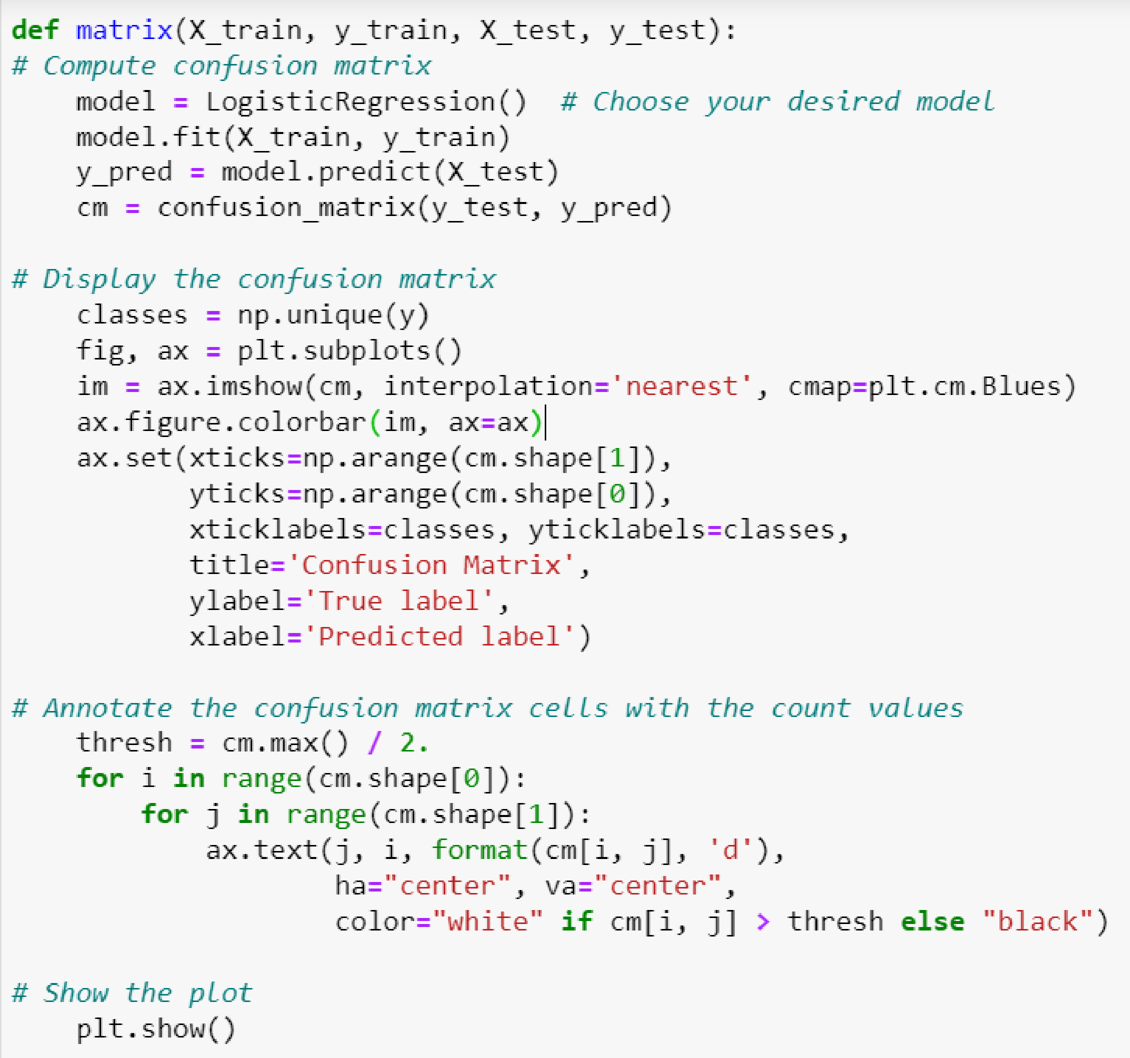
  Precision = TP/(TP+FP), where TP is true positives and FP is false positives.

- Recall: The ratio between number of positive samples correctly classified to the total number of positive samples.

  Recall = TP/(TP+FN), where TP is true positives and FN is false negatives.

6.5.1 *Confusion Matrix*

The confusion matrix evaluation was implemented as explained below:



```
def matrix(X_train, y_train, X_test, y_test):
# Compute confusion matrix
    model = LogisticRegression()  # Choose your desired model
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    cm = confusion_matrix(y_test, y_pred)

# Display the confusion matrix
    classes = np.unique(y)
    fig, ax = plt.subplots()
    im = ax.imshow(cm, interpolation='nearest', cmap=plt.cm.Blues)
    ax.figure.colorbar(im, ax=ax)
    ax.set(xticks=np.arange(cm.shape[1]),
           yticks=np.arange(cm.shape[0]),
           xticklabels=classes, yticklabels=classes,
           title='Confusion Matrix',
           ylabel='True label',
           xlabel='Predicted label')

# Annotate the confusion matrix cells with the count values
    thresh = cm.max() / 2.
    for i in range(cm.shape[0]):
        for j in range(cm.shape[1]):
            ax.text(j, i, format(cm[i, j], 'd'),
                    ha="center", va="center",
                    color="white" if cm[i, j] > thresh else "black")

# Show the plot
    plt.show()
```

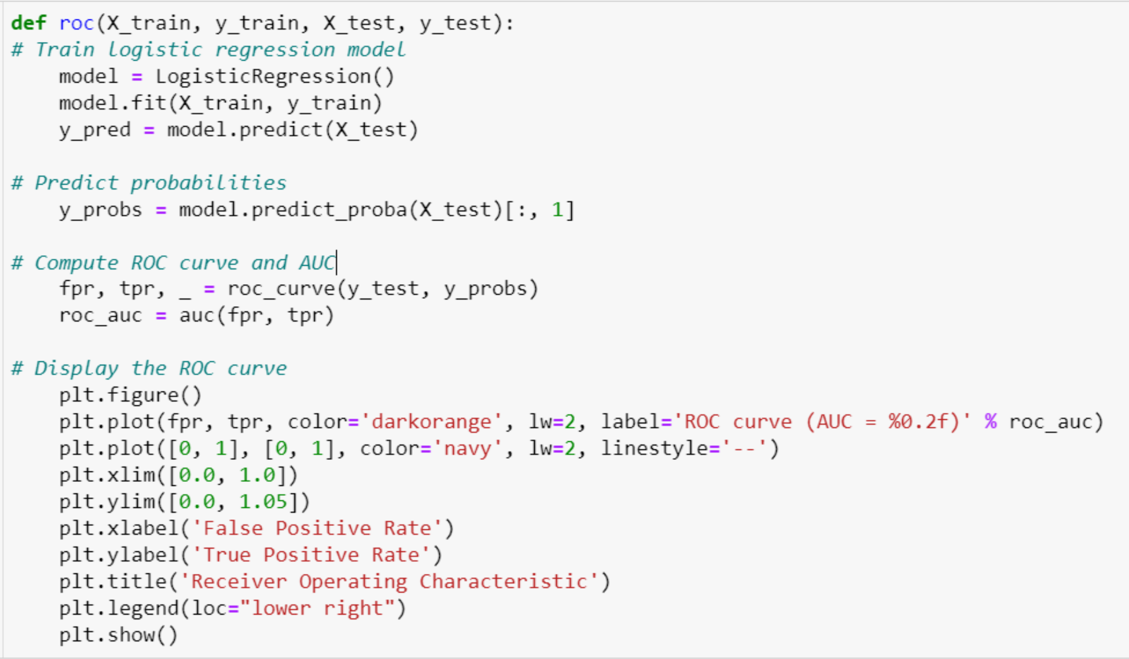
Figure 6.5: Confusion Matrix Implementation

- The function matrix X train, y train, X test, and y test.These parameters represent the training features, training labels, testing features, and testing labels, respectively.

- A logical regression model instance is then created.
- The model is then trained with training data X train.
- Predictions (y pred) are then made by using the test data X test on the trained model.
- Confusion matrix values are then obtained by using the function from scikit-learn library. It represents performance of the model by comparing actual labels (y test) with predicted labels (y pred).
- The confusion matrix is then displayed by taking unique classes from the target variable and plotting the count of instances along with color bar to the plot which helps in interpreting the confusion matrix.

6.5.2 ROC Curve

The ROC curve evaluation was implemented as explained below:



```
def roc(X_train, y_train, X_test, y_test):
# Train logistic regression model
    model = LogisticRegression()
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)

# Predict probabilities
    y_probs = model.predict_proba(X_test)[:, 1]

# Compute ROC curve and AUC
    fpr, tpr, _ = roc_curve(y_test, y_probs)
    roc_auc = auc(fpr, tpr)

# Display the ROC curve
    plt.figure()
    plt.plot(fpr, tpr, color='darkorange', lw=2, label='ROC curve (AUC = %0.2f)' % roc_auc)
    plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title('Receiver Operating Characteristic')
    plt.legend(loc="lower right")
    plt.show()
```

Figure 6.6: ROC Curve Implementation

- The function roc X train, y train, X test, and y test.These parameters represent the training features, training labels, testing features, and testing labels, respectively.

- A logical regression model instance is then created.
- The model is then trained with training data X train.
- Predictions (y pred) are then made by using the test data X test on the trained model.
- Using the predict proba() function, it computes the anticipated probability of the positive class (class 1). The [:, 1] indexing chooses the positive class probability.
- The Receiver Operating Characteristic (ROC) curve is computed by computing the False Positive Rate (FPR) and True Positive Rate (TPR) at various categorization thresholds. It makes use of scikit-learn's 'roc curve()' method.
- 'roc auc' computes the ROC curve's Area Under the Curve (AUC). The AUC shows the classifier's overall performance; a higher AUC number implies greater performance.
- The ROC curve is then plotted using the False Positive Rate (FPR) on the x-axis and True Positive Rate (TPR) on the y-axis.

The other evaluation metrics are then calculated from the values obtained from the confusion matrix using the formula's mentioned in chapter 2.

The results and evaluation after running the data set using these developed model and algorithm are discussed in the next chapter.

Chapter 7

Results and Evaluation

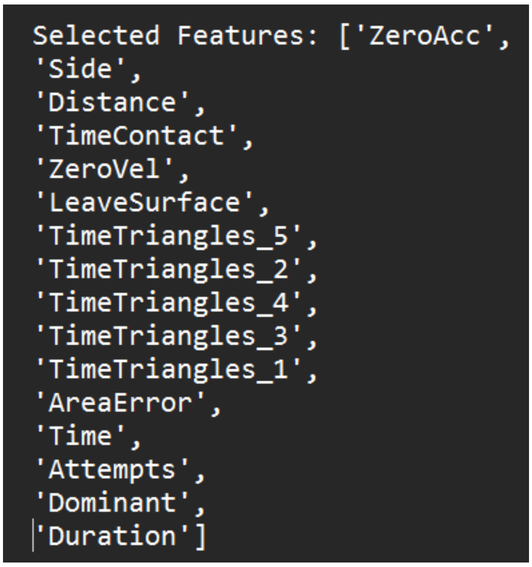
The two data sets '1st attempt.csv' and '2nd attempt.csv' respectively was used in both the developed algortihms explained in the previous chapter to obtain the results.

In this chapter first the outputs from feature shuffling algorithm is analysed and then discuss on the results obtained on the whole data set as well on the data set with significant features. Finally, I will summarise and analyse on the results obtained.

7.1 Feature Shuffling

Feature shuffling algorithm was developed and run on the data set to obtain the top features from the data set. The features are printed on the console from least significant to the most significant feature.

The below image shows the list of features in order of their significance.



```
Selected Features: ['ZeroAcc',
'Side',
'Distance',
'TimeContact',
'ZeroVel',
'LeaveSurface',
'TimeTriangles_5',
'TimeTriangles_2',
'TimeTriangles_4',
'TimeTriangles_3',
'TimeTriangles_1',
'AreaError',
'Time',
'Attempts',
'Dominant',
'Duration']
```

Figure 7.1: Feature list after running feature shuffling algorithm

The top significant features are then taken in to account for the further classification and evaluation. I am neglecting that attempts feature as I have already split the data set into two for 2 files for each attempts made by the control subjects and patients. So, I have taken next most significant feature into account for further study of classification and evaluation of the models.

Two new data sets are then made for 1st attemopt and 2nd attempt which are '1st attempt significant features.csv' and '2nd attempt significant features.csv'.

7.2 Using Full Data Set

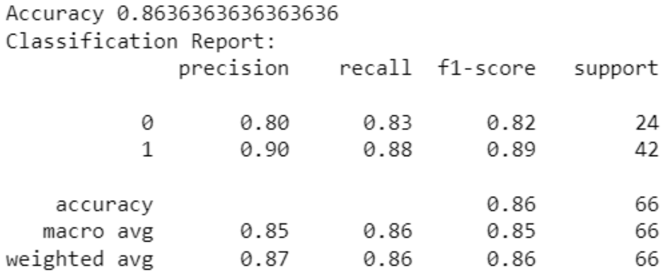
Initially, the full data set is used in the models to obtain the results as well as evaluation metrics.

7.2.1 Logistic Regression Model

The two data sets which are '1st attempt.csv' and '2nd attempt.csv' are used in the logistic regression model to obatin the results as below:

First Attempt Data

On running the first attempt data using logistic regression model below results are obtained.



```
Accuracy 0.8636363636363636
Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.83      0.82        24
           1       0.90      0.88      0.89        42

    accuracy                           0.86        66
   macro avg       0.85      0.86      0.85        66
weighted avg       0.87      0.86      0.86        66
```

Figure 7.2: Output Obtained using Logistic Regression on Full first attempt data set

An accuracy of 86.36 percent is obtained in this case. For class 0 (Control Subjects) precision value is 0.80 which suggests 80 percent of control subjects which were classified as control subjects were correct.The recall value of 0.83 suggests only 83 percent of actual class 0 instances were predicted correctly.F1 score of 0.82 means that the model achieves a reasonable balance between precision and recall for class 0 which is control subjects.

The model obtains high accuracy, recall, and F1-score values for class 1 (Patients).

With a precision of 0.90, 90 percent of the examples identified as class 1 were right. A recall of 0.88 implies that the model successfully predicted 88 percent of real class 1 (Patients) cases. For class 1, the F1-score of 0.89 suggests a solid combination of precision and recall.

The weighted-averaged metrics take the weighted mean of precision, recall, and F1-score for each class, weighted by the number of samples in each class (support). Here, the weighted-averaged precision is 0.87, weighted-averaged recall is 0.86, and weighted-averaged F1-score is 0.86. It represents the average performance across all classes, considering the class imbalance.

The model shows good precision, recall, and F1-score for both classes, indicating that it can effectively distinguish between the positive and negative instances in the test data. The high accuracy and balanced performance metrics suggest that the model is effective and provides accurate predictions for the given test data set.

Below Image shows the Confusion Matrix output:

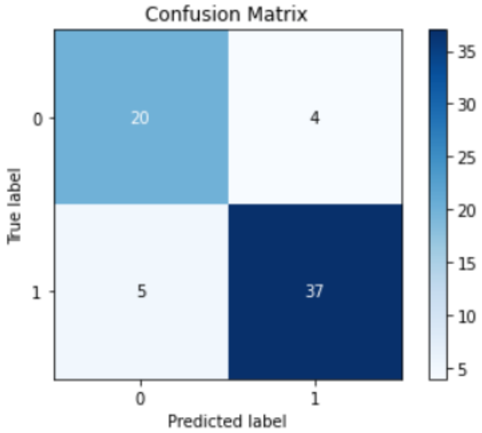


Figure 7.3: Confusion Matrix for Logistic Regression Model using the first attempt full data set

From the matrix It can be seen that out of 24 control subject instances 20 were correctly predicted and out of 42 patient subject instances 37 were correctly predicted by the model. From the matrix, sensitivity is obtained as 0.833 and specificity as 0.902.The model appears to perform well in terms of both sensitivity and specificity, indicating its effectiveness in correctly identifying positive and negative instances.

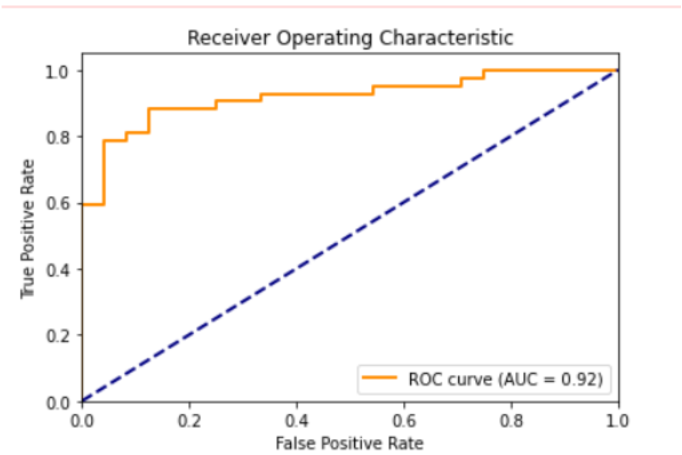
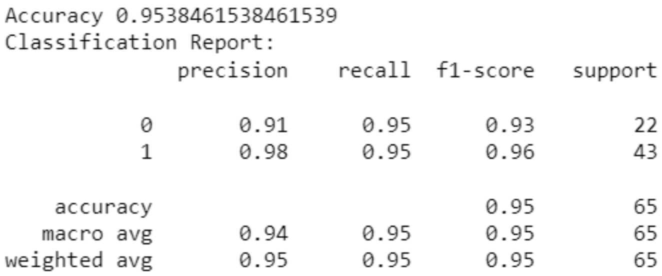


Figure 7.4: ROC curve for Logistic Regression Model using the first attempt full data set

The ROC curve is generated as in the above image and the AUC value is obtained to be 0.92, which shows that the model has good discrimination between the two classes. It suggests that the model is capable of distinguishing between positive and negative instances with a relatively high degree of accuracy.

Second Attempt Data

On running the second attempt data using logistic regression model below results are obtained.



```
Accuracy 0.9538461538461539
Classification Report:
              precision    recall  f1-score   support

           0       0.91      0.95      0.93        22
           1       0.98      0.95      0.96        43

    accuracy                           0.95        65
   macro avg       0.94      0.95      0.95        65
weighted avg       0.95      0.95      0.95        65
```

Figure 7.5: Output Obtained using Logistic Regression on Full second attempt data set

An accuracy of 95.38 percent is obtained in this case.For class 0 (Control Subjects) precision value is 0.91 which suggests 91 percent of control subjects which were classified as control subjects were correct.The recall value of 0.95 suggests 95 percent of actual class 0 instances were predicted correctly. F1 score of 0.93 means that the model achieves a very high balance between precision and recall for class 0 which is control subjects.

The model obtains high accuracy, recall, and F1-score values for class 1 (Patients). With a precision of 0.98, 98 percent of the examples identified as class 1 were right. A recall of 0.95 implies that the model successfully predicted 95 percent of real class 1 (Patients) cases. For class 1, the F1-score of 0.96 suggests a good balance of precision and recall.

The weighted-averaged metrics take the weighted mean of precision, recall, and F1-score for each class, weighted by the number of samples in each class (support). Here, the weighted-averaged precision is 0.95, weighted-averaged recall is 0.95, and weighted-averaged F1-score is 0.95. It gives a better results than the first attempt data with high accuracy and other metrics values.

the model is performing well on both classes, achieving high precision, recall, and F1-score values. The high accuracy and balanced performance metrics suggest that the model is effective and provides accurate predictions for the given test data set.

Below Image shows the Confusion Matrix output:

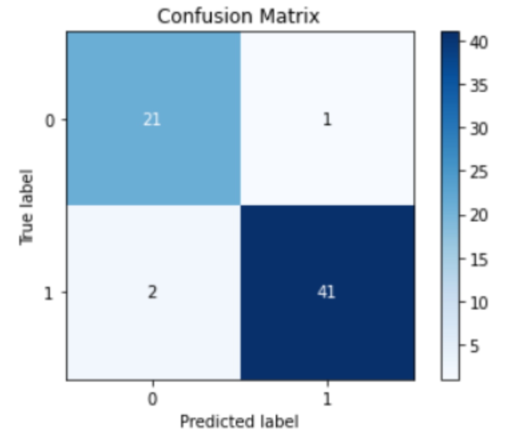


Figure 7.6: Confusion Matrix for Logistic Regression Model using the second attempt full data set

From the matrix It can be seen that out of 22 control subject instances 21 were correctly predicted and out of 43 patient subject instances 41 were correctly predicted by the model. From the matrix, sensitivity is obtained as 0.913 and specificity as 0.97.The model appears to perform well in terms of both sensitivity and specificity, indicating its effectiveness in correctly identifying positive and negative instances slightly better than the first attempt data set.

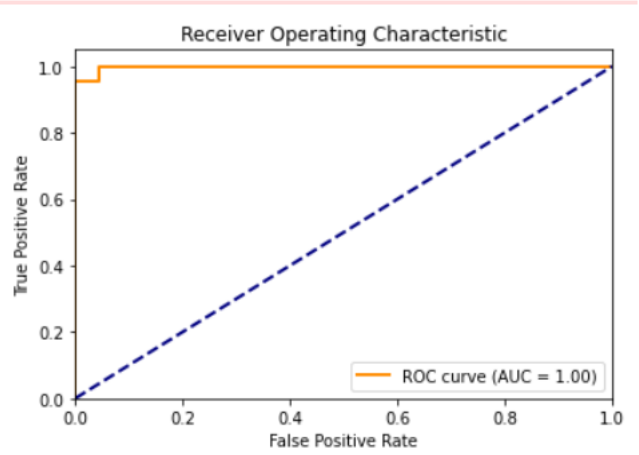


Figure 7.7: ROC curve for second attempt data using Logistic Regression Model

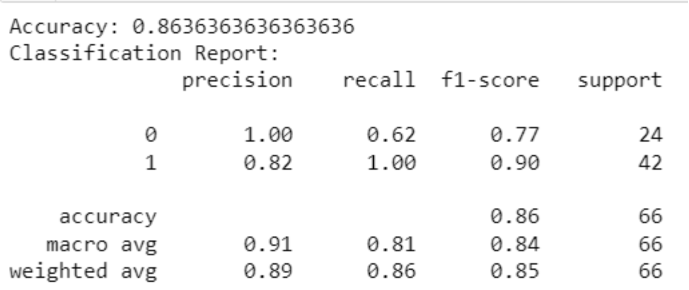
The ROC curve is generated as in the below image and the AUC value is obtained to be 1 which shows that the model shows a perfect working, but it may be due to the fact that the data set contains 70 percent of patient data and only 30 percent of control subject data. So the ROC curve data may be misleading due to data over fitting.

7.2.2 SVM Model

The two data sets ’1st attempt.csv’ and ’2nd attempt.csv’ was used in SVM Model developed and results were analysed similarly as how it was done using logical regression model.

First Attempt Data

On running first attempt data using the SVM Model developed, below results are obtained.



```
Accuracy: 0.8636363636363636
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.62      0.77        24
           1       0.82      1.00      0.90        42

    accuracy                           0.86        66
   macro avg       0.91      0.81      0.84        66
weighted avg       0.89      0.86      0.85        66
```

Figure 7.8: Output Obtained using SVM Model on Full first attempt data se

An accuracy of 86.36 percent is obtained in this case.For class 0 (Control Subjects) precision value is 1 which suggests 100 percent of control subjects who were predicted as control subjects were correct.The recall value of 0.62 suggests that only 62 percent of actual class 0 instances were predicted correctly by the model. F1 score of 0.77 means that the model achieves a reasonable balance between precision and recall for class 0 which is control subjects.

The model obtains high accuracy, recall, and F1-score values for class 1 (Patients). With a precision of 0.82, 78 percent of the examples identified as class 1 were right. A recall of 1 implies that the model successfully predicted 100 percent of real class 1 (Patients) cases. For class 1, the F1-score of 0.90 suggests a strong balance of precision and recall.

The model shows good precision, recall, and F1-score for class 1 (Patients), indicating that it can effectively distinguish between the positive and negative instances in the test data. However, for class 0 (Control Subjects), the precision is perfect, but the recall is relatively low, indicating that the model may have difficulty correctly identifying some instances of class 0.

Below Image shows the Confusion Matrix output:

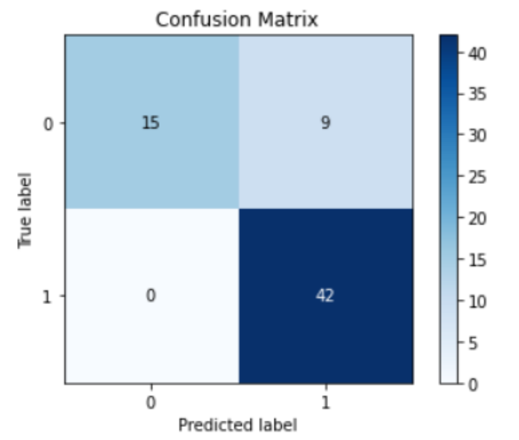


Figure 7.9: Confusion Matrix for SVM Model using the first attempt full data set

From the matrix it can be seen that out of 24 control subject instances 15 were correctly predicted and out of 42 patient subject instances all were correctly predicted by the model.From the matrix, sensitivity is obtained as 1 and specificity as 0.82. The model demonstrates a high level of sensitivity, correctly identifying all positive instances, which is desirable in many applications. However, there is room for improvement in specificity, as the model is miss classifying some negative instances.

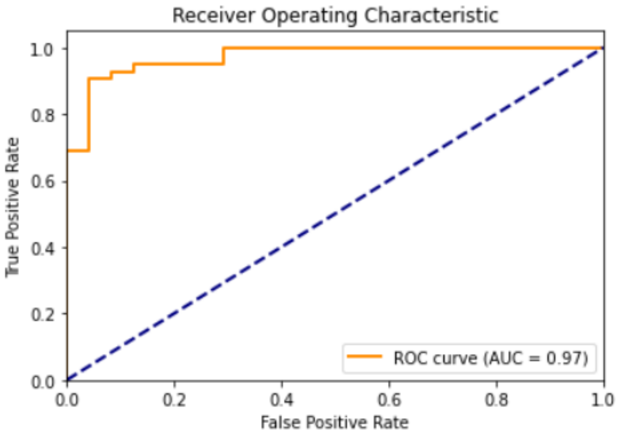
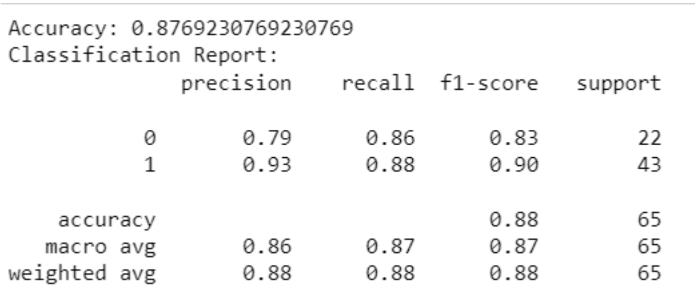


Figure 7.10: ROC curve for first attempt full data using SVM Model

The ROC curve is generated as in the below image and the AUC value is obtained to be 0.97 which indicates that the model can effectively distinguish between patients and control subjects in the test data. It means that the model has a high probability of ranking a randomly chosen patient instance higher than a randomly chosen control subject instance.

Second Attempt Data

On running the second attempt full data on SVM Model the following results are obtained.



```
Accuracy: 0.8769230769230769
Classification Report:
              precision    recall  f1-score   support

           0       0.79      0.86      0.83        22
           1       0.93      0.88      0.90        43

    accuracy                           0.88        65
   macro avg       0.86      0.87      0.87        65
weighted avg       0.88      0.88      0.88        65
```

Figure 7.11: Output Obtained using SVM Model on Full second attempt data set

An accuracy of 87.69 percent is obtained in this case.For class 0 (Control Subjects) precision value is 0.79 which suggests 79 percent of control subjects who were predicted as control subjects were correct.The recall value of 0.86 suggests that only 79 percent of actual class 0 (Control Subjects) instances were predicted correctly by the model.F1 score of 0.83 means that the model achieves a reasonable balance between precision and recall for class 0 which is control subjects.

The model obtains reasonable accuracy, recall, and F1-score values for class 1 (Patients). With a precision of 0.93, 88 percent of the examples identified as class 1 were right. A recall of 0.88 implies that the model successfully predicted 88 percent of real class 1 (Patients) cases. For class 1, the F1-score of 0.9 suggests a solid balance of precision and recall.

The model shows good precision, recall, and F1-score for both classes, indicating that it can effectively distinguish between the positive and negative instances in the test data. The high accuracy and balanced performance metrics suggest that the model is highly effective and provides accurate predictions for the given test data set.

Below Image shows the Confusion Matrix output:

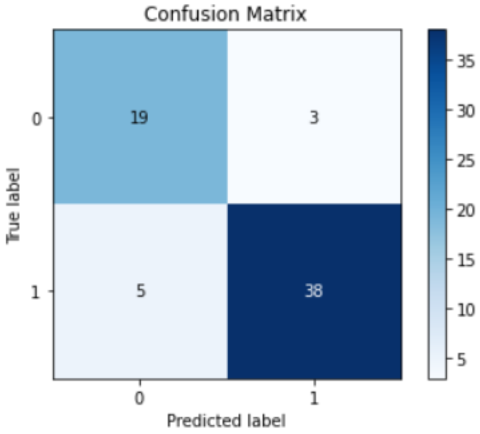


Figure 7.12: Confusion Matrix for SVM Model using the second attempt full data set

From the matrix it can be seen that out of 22 control subject instances 19 were correctly predicted and out of 43 patient subject instances 38 were correctly predicted by the model. From the matrix, sensitivity is obatined as 0.93 and specificity as 0.79. the model demonstrates a high level of sensitivity, correctly identifying most positive instances, which is desirable in many applications. However, specificity might be improved because the model misclassifies certain negative occurrences.

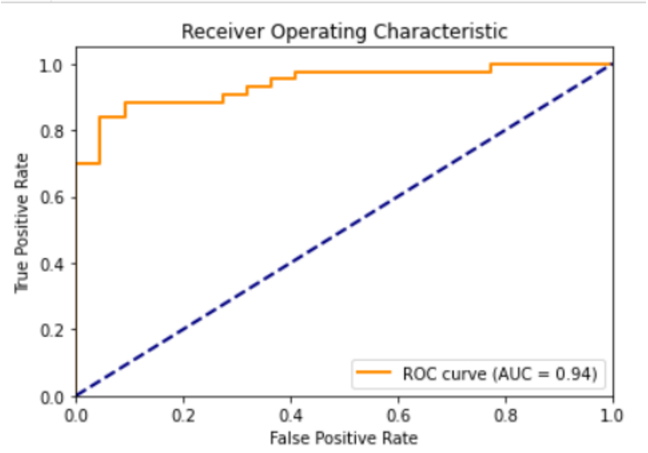


Figure 7.13: ROC curve for second attempt full data set using SVM Model

The ROC curve is generated as in the below image and the AUC value is obtained to be 0.94. Ahigh AUC value of 0.94 suggests that the model is performing well in distinguishing between the positive and negative instances, making it a promising model for binary classification tasks.

7.3 Using Data Set With Only Significant Features

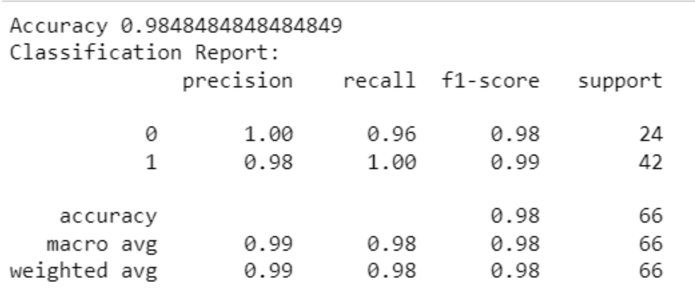
After running the full data set on the developed models and obtaining results and evaluation metrics the data set with only significant features which are '1st attempt significant features.csv' and '2nd attempt significant features.csv' are used in the models to analyse the changes in results and evaluation metrics values.

7.3.1 Logistic Regression Model

The two data sets with only significant features '1st attempt significant features.csv' and '2nd attempt significant features.csv' are used in the logistic regression model and results are obtained.

First Attempt Data

On running the first attempt data set with only significant features on logistic regression model below results are obtained.



```
Accuracy 0.9848484848484849
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.96      0.98        24
           1       0.98      1.00      0.99        42

    accuracy                           0.98        66
   macro avg       0.99      0.98      0.98        66
weighted avg       0.99      0.98      0.98        66
```

Figure 7.14: Output Obtained using Logistic Regression on Significant features first attempt data set

A high accuracy of 98.84 percent is obtained. For class 0 (Control Subjects) precision value is 1 which suggests that all the control subjects who were predicted as control subjects were correct.The recall value of 0.96 suggests that 96 percent of actual class 0 (Control Subjects) instances were predicted correctly by the model.F1 score of 0.98 means that the model achieves a high balance between precision and recall for class 0 which is control subjects.

The model obtains high precision, recall, and F1-score values for class 1 (Patients). With a precision of 0.98, 98 percent of the examples identified as class 1 were right. A recall of 1 implies that the model successfully predicted all the real class 1 (Patients) cases. For class 1, the F1-score of 0.99 which again suggests a high balance of precision and recall.

For all classes, the model has great accuracy, recall, and F1-score, showing that it can successfully differentiate between positive and negative occurrences in the test data. The model's high accuracy and balanced performance metrics indicate that it is extremely effective and gives correct predictions for the provided test data set.

Below Image shows the Confusion Matrix output:

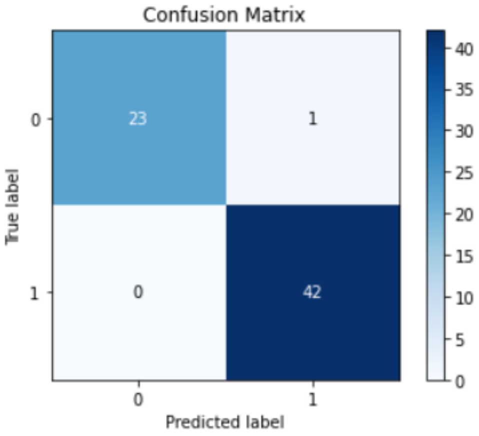


Figure 7.15: Confusion Matrix for Logistic Regression Model using the first attempt significant features data set

From the matrix it can be seen that out of 24 control subject instances 23 were correctly predicted and out of 42 patient subject instances all were correctly predicted by the model.From the matrix, sensitivity is obtained as 0.97 and specificity as 1. The model appears to be highly accurate and reliable, with near-perfect sensitivity and specificity values. It demonstrates robust performance in correctly identifying both positive and negative instances, making it a promising model for binary classification tasks.

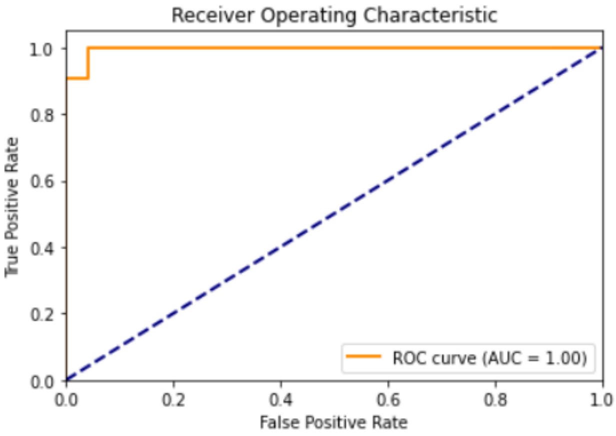
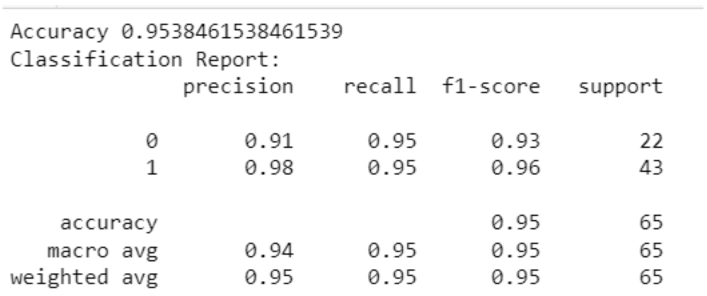


Figure 7.16: ROC curve for first attempt significant features data set using logistic regression model

The ROC curve is generated as in the below image and the AUC value is obtained to be 1 which indicates that the model perfectly distinguishes between positive and negative instances in the test data. It means that the model has a high probability of ranking a randomly chosen positive instance (Patients) higher than a randomly chosen negative instance (Control Subjects) in all cases.

Second Attempt Data

On running the first attempt data set with only significant features on logistic regression model below results are obtained.



```
Accuracy 0.9538461538461539
Classification Report:
              precision    recall  f1-score   support

           0       0.91      0.95      0.93        22
           1       0.98      0.95      0.96        43

    accuracy                           0.95        65
   macro avg       0.94      0.95      0.95        65
weighted avg       0.95      0.95      0.95        65
```

Figure 7.17: Output Obtained using Logistic Regression on Significant features second attempt data set

A high accuracy of 95.38 percent is obtained. For class 0 (Control Subjects) precision value is 0.91 which suggests that 91 percent of the control subjects who were predicted as control subjects were correct.The recall value of 0.95 suggests that 95 percent of actual class 0 (Control Subjects) instances were predicted correctly by the model.F1 score of 0.93 means that the model achieves a high balance between precision and recall for class 0 which is control subjects.

The model obtains high precision, recall, and F1-score values for class 1 (Patients). With a precision of 0.98, 98 percent of the examples identified as class 1 were right. A recall of 0.95 implies that the model successfully predicted 95 percent all the real class 1 (Patients) cases. For class 1, the F1-score of 0.96 which again suggests a high balance of precision and recall.

The model shows good precision, recall, and F1-score for both classes, indicating that it can effectively distinguish between the positive and negative instances in the test data. The high accuracy and balanced performance metrics suggest that the model is highly effective and provides accurate predictions for the given test data set.

Below Image shows the Confusion Matrix output:

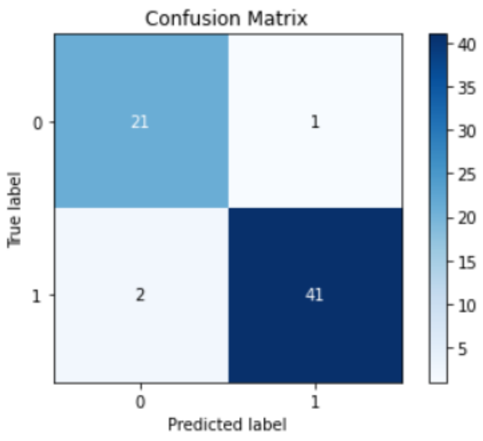


Figure 7.18: Confusion Matrix for Logistic Regression Model using the second attempt significant features data set

From the matrix it can be seen that out of 22 control subject instances 21 were correctly predicted and out of 43 patient subject instances 41 were correctly predicted by the model.From the matrix, sensitivity is obtained as 0.97 and specificity as 0.913. The model appears to be highly accurate and reliable, with near-perfect sensitivity and specificity values. It is critical to achieve a balance between sensitivity and specificity, based on the problem's unique needs and the costs of false positives and false negatives.

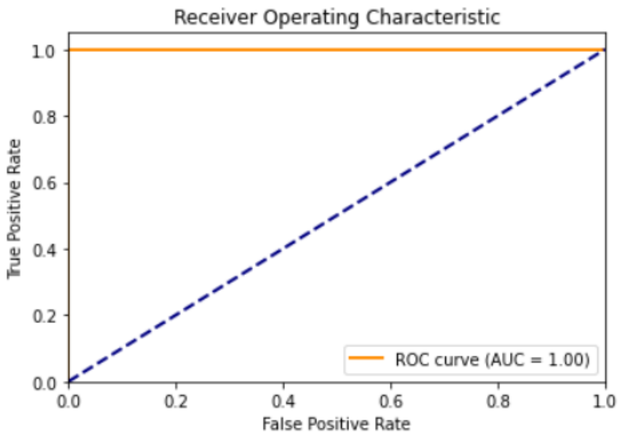


Figure 7.19: ROC curve for second attempt significant features data set using logistic regression model

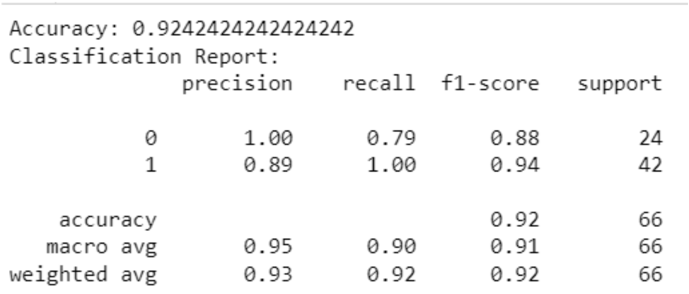
The ROC curve is generated as in the below image and the AUC value is obtained to be 1 Having an AUC value of 1 is a desirable and ideal scenario for a classification model, as it demonstrates a highly accurate and effective model in distinguishing between the positive and negative classes. However, it is essential to ensure that the high AUC value is not a result of over fitting to the training data. Which in this case may be an issue as we have 70 percent of the data belonging to the patients.

7.3.2 SVM Model

The two data sets with only significant features ’1st attempt significant features.csv’ and ’2nd attempt significant features.csv’ are used in the logistic regression model and results are obtained.

First Attempt Data

On running the first attempt data set with only significant features on logistic regression model below results are obtained.



```
Accuracy: 0.9242424242424242
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.79      0.88        24
           1       0.89      1.00      0.94        42

    accuracy                           0.92        66
   macro avg       0.95      0.90      0.91        66
weighted avg       0.93      0.92      0.92        66
```

Figure 7.20: Output Obtained using SVM Model on Significant features first attempt data set

A high accuracy of 92.42 percent is obtained. For class 0 (Control Subjects) precision value is 1 which suggests that all the control subjects who were predicted as control subjects were correct.The recall value of 0.79 suggests that 79 percent of actual class 0 (Control Subjects) instances were predicted correctly by the model.F1 score of 0.88 means that the model achieves a high balance between precision and recall for class 0 which is control subjects.

The model obtains high precision, recall, and F1-score values for class 1 (Patients). With a precision of 0.89, 89 percent of the examples identified as class 1 were right. A recall of 1 implies that the model successfully predicted all the real class 1 (Patients) cases. For class 1, the F1-score of 0.94 which again suggests a high balance of precision and recall.

For all classes, the model has great accuracy, recall, and F1-score, showing that it can successfully differentiate between positive and negative occurrences in the test data. The model's high accuracy and balanced performance metrics indicate that it is extremely effective and gives correct predictions for the provided test data set.

Below Image shows the Confusion Matrix output:

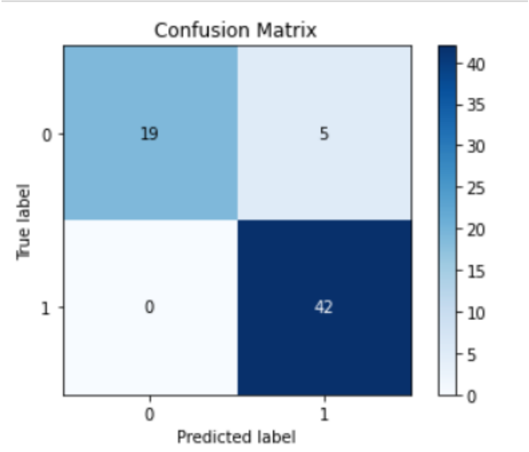


Figure 7.21: Confusion Matrix for SVM Model using the first attempt significant features data set

From the matrix it can be seen that out of 24 control subject instances 19 were correctly predicted and out of 42 patient subject instances 42 were correctly predicted by the model.From the matrix,sensitivity is obatined as 0.89 and specificity as 1. Having a specificity of 1.00 is ideal, as it means the model is highly accurate in identifying negative cases. The sensitivity of 0.89 also indicates good performance in detecting positive cases, although it may benefit from further improvement to increase the true positive rate.

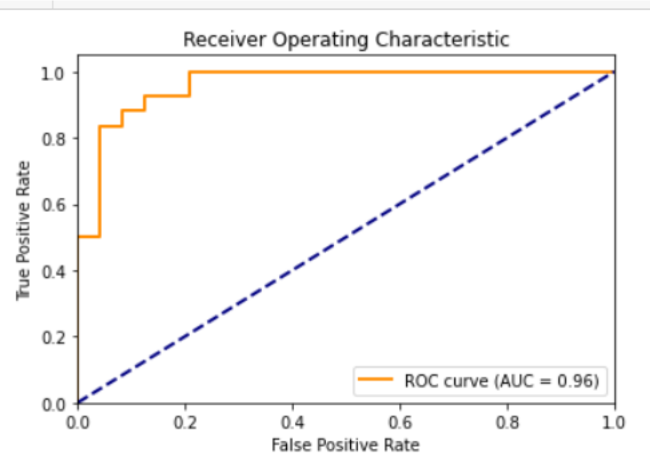
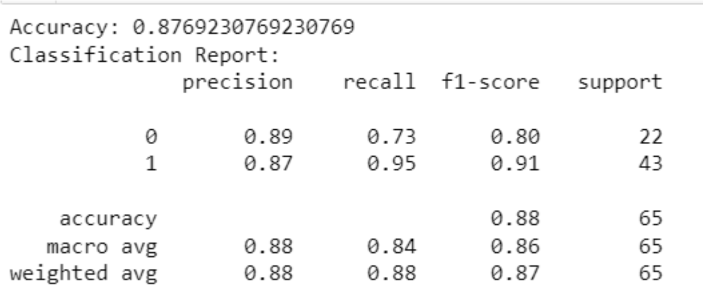


Figure 7.22: ROC curve for first attempt significant features data set using SVM model

The ROC curve is generated as in the below image and the AUC value is obtained to be 0.96. AUC = 0.96 means that the model's predictions are highly accurate, with minimal overlap between the true positive and true negative rates. The model is effective in distinguishing between the positive and negative classes, achieving a high true positive rate (sensitivity) and a high true negative rate (specificity).

Second Attempt Data

On running the first attempt data set with only significant features on logistic regression model below results are obtained.



```
Accuracy: 0.8769230769230769
Classification Report:
              precision    recall  f1-score   support

           0       0.89      0.73      0.80        22
           1       0.87      0.95      0.91        43

    accuracy                           0.88        65
   macro avg       0.88      0.84      0.86        65
weighted avg       0.88      0.88      0.87        65
```

Figure 7.23: Output Obtained using SVM Model on Significant features second attempt data set

An accuracy of 87.69 percent is obtained. For class 0 (Control Subjects) precision value is 0.89 which suggests that 89 percent of the control subjects who were predicted as control subjects were correct.The recall value of 0.73 suggests that 73 percent of actual class 0 (Control Subjects) instances were predicted correctly by the model.F1 score of 0.80 means that the model achieves a relative good balance between precision and recall for class 0 which is control subjects.

The model obtains high precision, recall, and F1-score values for class 1 (Patients). With a precision of 0.87, 87 percent of the examples identified as class 1 were right. A recall of 0.95 implies that the model successfully predicted 95 percent of all the real class 1 (Patients) cases. For class 1, the F1-score of 0.91 which again suggests a high balance of precision and recall.

The model demonstrates a reasonably good ability to correctly classify instances into their respective classes, with higher performance in class 1 (Patients) compared to class 0 (Control Subjects). It may be beneficial to explore ways to improve the model's performance further, depending on the specific requirements of the problem and the costs associated with false positives and false negatives.

Below Image shows the Confusion Matrix output:

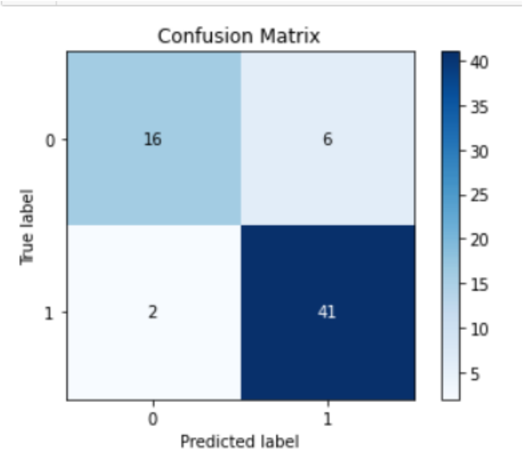


Figure 7.24: Confusion Matrix for SVM Model using the second attempt significant features data set

From the matrix it can be seen that out of 22 control subject instances 16 were correctly predicted and out of 43 patient subject instances 41 were correctly predicted by the model.From the matrix, sensitivity is obtained as 0.87 and specificity as 0.88. Having both sensitivity and specificity values close to 1 is desirable, as it indicates the model's effectiveness in detecting both positive and negative cases accurately.

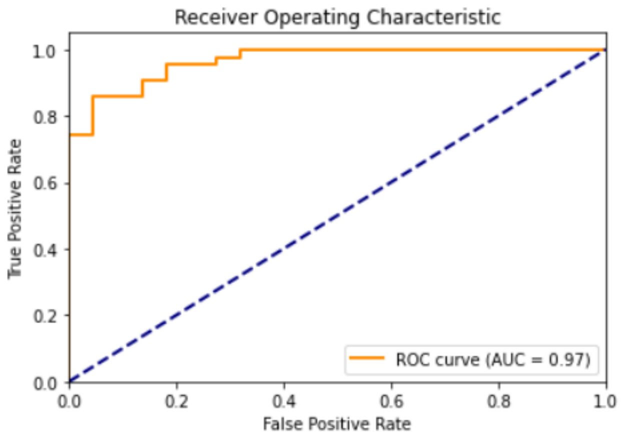


Figure 7.25: ROC curve for second attempt significant features data set using SVM model

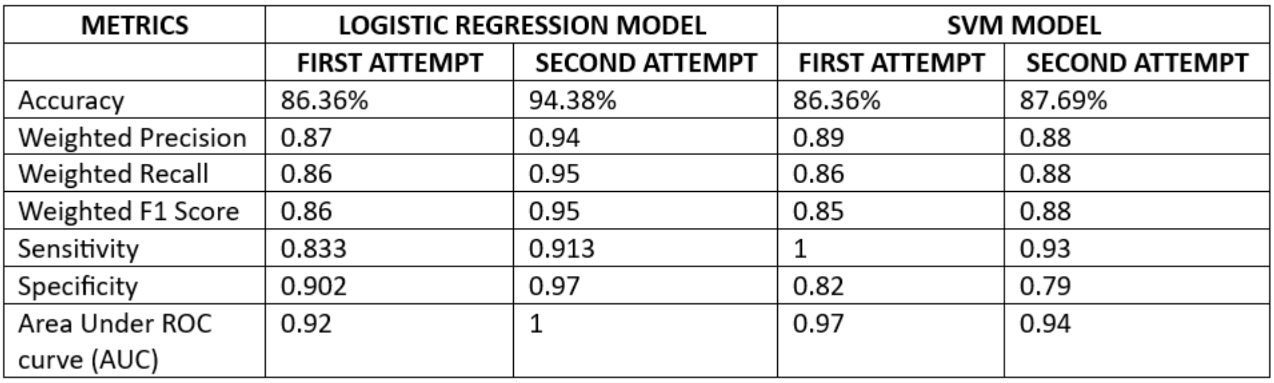
The ROC curve is generated as in the below image and the AUC value is obtained to be 0.97 which is an indication of the model's strong discriminatory power. It suggests that the model is highly effective in distinguishing between the positive and negative instances in the test data.

7.4 Summary and Analysis

The below tables show a summary of different metrics values obtained for different cases during this research project. A detailed analysis on this is also done and explained in this section.

Results of using Full Data Set

The results obtained by using the full data set in the models developed are summarised below in the table.



METRICS	LOGISTIC REGRESSION MODEL		SVM MODEL	
	FIRST ATTEMPT	SECOND ATTEMPT	FIRST ATTEMPT	SECOND ATTEMPT
Accuracy	86.36%	94.38%	86.36%	87.69%
Weighted Precision	0.87	0.94	0.89	0.88
Weighted Recall	0.86	0.95	0.86	0.88
Weighted F1 Score	0.86	0.95	0.85	0.88
Sensitivity	0.833	0.913	1	0.93
Specificity	0.902	0.97	0.82	0.79
Area Under ROC curve (AUC)	0.92	1	0.97	0.94

Table 7.1: Summary Table of Results using Full Data set in models

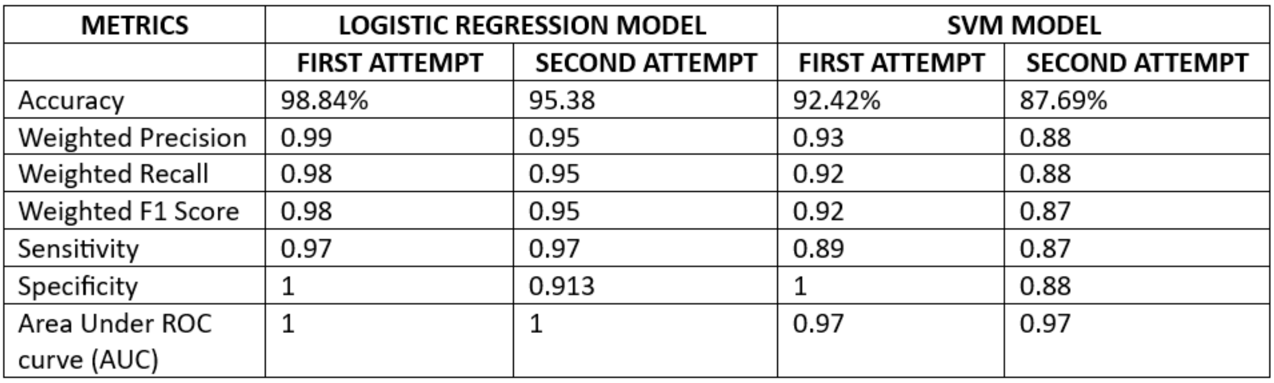
Both the Logistic Regression and SVM models proved to be highly predictive, with excellent accuracy, sensitivity, and AUC values. The Logistic Regression model performed somewhat worse in the first try but much better in the second attempt. The SVM model, on the other hand, performed exceptionally well in the first try but somewhat worse in the second.

While the SVM model performed somewhat better in both efforts in terms of sensitivity, the Logistic Regression model displayed a better balance between precision and recall for the unbalanced data set, resulting in greater Weighted Precision, Recall, and F1 Score. Furthermore, the Logistic Regression model performed more consistently in Specificity across tries.

Both models had strong discriminatory power, as seen by high AUC values, implying their ability to successfully differentiate between positive and negative cases.

Results of using Data Set with significant features

The results obtained by using the data set with significant features in the models developed are summarised below in the table.



METRICS	LOGISTIC REGRESSION MODEL		SVM MODEL	
	FIRST ATTEMPT	SECOND ATTEMPT	FIRST ATTEMPT	SECOND ATTEMPT
Accuracy	98.84%	95.38	92.42%	87.69%
Weighted Precision	0.99	0.95	0.93	0.88
Weighted Recall	0.98	0.95	0.92	0.88
Weighted F1 Score	0.98	0.95	0.92	0.87
Sensitivity	0.97	0.97	0.89	0.87
Specificity	1	0.913	1	0.88
Area Under ROC curve (AUC)	1	1	0.97	0.97

Table 7.2: Summary Table of Results using Data set with significant features in models

Both the Logistic Regression and SVM models proved to be highly predictive, with excellent accuracy, sensitivity, and AUC values. In terms of weighted accuracy, recall, and F1 score, the Logistic Regression model consistently beat the SVM model, making it better appropriate for dealing with the unbalanced data set. Furthermore, the Logistic Regression model attained 100 percent specificity, indicating that it is extremely trustworthy in categorising negative cases.

It is important to remember, however, that model performance may vary depending on the data set and individual use case. More experimentation, cross-validation, and testing on independent data sets would be beneficial in validating both models' generalisation capabilities. Overall, the results give useful insights into the models' performance in identifying occurrences and can help drive decision-making in selecting the best model for the job.

Here the results obtained are much better than what has been achieved using the full data set. This shows that the feature shuffling algorithm in fact helps in achieving better results by removing unwanted features for the study.

Also the high accuracy o and AUC values obtained using the logistic regression may be due to the fact that the dataset provided is imbalanced as 70 percent data belongs to one class and only 30 percent belongs to the other.

Chapter 8

Conclusion And Future Works

8.1 Conclusion

The main objective of this project was to explore the possibilities of using Logistic Regression and Feature Shuffling Algorithm in predicting Parkinson's disease from a given data set. Two models were developed for this Study.

- Logistic Regression Model
- SVM Model

The developed model was used in analysing the given data set and the models were evaluated using different methods discussed in Chapter 2.

The results discussed in Chapter 7 provide evidence that the models perform well in some aspects and the evaluation metrics calculated supports this. I'd like to point out that the performance of the classifiers using the second attempt data set which results in outputs compared to the first attempt data set. The results obtained are far superior to what I expected at the start of the research. I'd also want to emphasise the logistic regression model which achieved 98.84 percent accuracy using the data set with only significant features. The high accuracy and AUC values would be misleading due to the data imbalance in the data set, but having a high weighted precision and recall values shows the ability of the model to classify classes especially in a binary classification problems.

The models developed also succeeded in achieving the mandatory requirements discussed in chapter 4.

Specifically, It succeeded in

- Model Should be accurately distinguish between patients and control subjects : The models were able to predict between patients and control subjects accurately in all the cases.

- The developed model must have accuracy as near as possible to 100 percent: The models developed showed more than 85 percent accuracy is predicting patients and control subjects in all the cases.
- The model should have a user-friendly interface that can be easily navigated by users: The models were developed in Jupyter Notebooks using prominent python libraries, which makes it easier for the user to understand the interface and working of the models.

I lost a lot of time since I didn't have a straight relationship between the data and the attribute names. However, it was also the source of several problems during the project.I also struggled with time management. I spent too much time gathering data, experimenting with various classifiers, filters, and so on. I had issues with data imbalance in the data set as well which leaded to misleading results and had to tune the training and testing data split percentage to obtain somewhat valid results for this research.

8.2 Future Works

Although this project provides basic implementation of logistic regression and SVM models in predicting the Parkinson's disease from a given data set, more research could be done by making the problem more complex.

Some suggestions for future works are listed below:

- The problem could be made more complex by analysing the pentagon spiral image drawn by the patients and control subjects rather than the data obtained from the drawings.
- Using another feature selection methods or model would have made a change or improved accuracy.
- Analysing the data in a much efficient manner or extracting new features from the existing data set can be done in future to know how it affects the model performances and prediction accuracy.
- Adding in some random control subject data to make the data set more balanced would also help in achieving a more balanced model.

Chapter 9

Bibliography

1. Abudi, S., Bar-Tal, Y., Ziv, L. and Fish, M. 1997. Parkinson's disease symptoms - Patients' perceptions. Journal of Advanced Nursing 25(1), pp. 54–59. doi: 10.1046/j.1365-2648.1997.1997025054.x.

2. Ali, L., Zhu, C., Zhou, M. and Liu, Y. 2019. Early Diagnosis of Parkinson's Disease from Multiple Voice Recordings by Simultaneous Sample and Feature Selection. Expert Syst. Appl. 137(C), pp. 22–28. Available at: https://doi.org/10.1016/j.eswa.2019.06.052.

3. Aich, S., Younga, K., Hui, K.L., Al-Absi, A.A. and Sain, M. 2018. A nonlinear decision tree based classification approach to predict the Parkinson's disease using different feature sets of voice data. In: International Conference on Advanced Communication Technology, ICACT. Institute of Electrical and Electronics Engineers Inc., pp. 638–642. doi: 10.23919/ICACT.2018.8323864.

4. Almeida, J.S., Rebouças Filho, P.P., Carneiro, T., Wei, W., Damaševičius, R., Maskeli¯ Unas D , Victor, R. and De Albuquerque, H.C. [2019]. Detecting Parkinson's Disease with Sustained Phonation and Speech Signals using Machine Learning Techniques.

5. Azuaje, F. 2006. Witten IH, Frank E: Data Mining: Practical Machine Learning Tools and Techniques 2nd edition. BioMedical Engineering OnLine 5(1). doi: 10.1186/1475-925x-5-51.

6. Berus, L., Klancnik, S., Brezocnik, M. and Ficko, M. 2019. Classifying parkinson's disease based on acoustic measures using artificial neural networks. Sensors (Switzerland) 19(1). doi: 10.3390/s19010016.

7. Cai, Z. et al. 2018. An Intelligent Parkinson's Disease Diagnostic System Based on a Chaotic Bacterial Foraging Optimization Enhanced Fuzzy KNN Approach. Computational and Mathematical Methods in Medicine 2018. doi: 10.1155/2018/2396952.

8. Čehovin, L. and Bosnić, Z. 2010. Empirical evaluation of feature selection methods in classification. Intelligent Data Analysis 14(3), pp. 265–281. doi: 10.3233/IDA-2010-0421.

9. Chicco, D. and Jurman, G. 2020. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC Genomics 21(1). doi: 10.1186/s12864-019-6413-7.

10. Cortes, C., Vapnik, V. (1995). "Support-vector networks." Machine learning, 20(3), 273-297.

11. Deb, R., Bhat, G., An, S., Shill, H., Ogras, U.Y. and Ali, M. [no date]. Disease Assessment: A Systematic Review This review focuses on the use of modern wearable and mobile equipment for PD applications in the last decade. Four. Available at: https://doi.org/10.1101/2021.02.01.21250939.

12. Deepak Gupta, Arnav Julka, Sanchit Jain, Tushar Aggarwal, Ashish Khanna, N Arunkumar and Victor Hugo C de Albuquerque 2018. Optimized cuttlefish algorithm for diagnosis of Parkinson's disease. Cognitive systems research 52, pp. 36–48.

13. Dey, A. 2016. Machine Learning Algorithms: A Review. International Journal of Computer Science and Information Technologies 7(3)

14. Gagliano, M., Pham, J., Tang, B., Kashif, H. and Ban, J. 2017. Applications of Machine Learning in Medical Diagnosis.

15. Ghazali, R., Mat, M., Nazri, D., Nawi, M. and Abawajy Editors, J.H.2018. Recent Advances on Soft Computing and Data Mining. Available at: http://www.springer.com/series/111

16. Godinho, C. et al. 2016. A systematic review of the characteristics and validity of monitoring technologies to assess Parkinson's disease. Journal of NeuroEngineering and Rehabilitation 13(1). doi: 10.1186/s12984-016-0136-7.

17. Goetz, C.G., LeWitt, P.A. and Weidenman, M. 2003. Standardized training tools for the UPDRS activities of daily living scale: Newly available teaching program. Movement Disorders 18(12), pp. 1455–1458. doi: 10.1002/mds.10591.

18. Harrington, P. [no date]. Machine Learning in Action.

19. Hosmer Jr, D. W., Lemeshow, S., and Sturdivant, R. X. (2013). "Applied Logistic Regression." John Wiley and Sons.

20. Jankovic, J. 2008. Parkinson's disease: clinical features and diagnosis. Journal of Neurology, Neurosurgery and amp;amp; Psychiatry 79(4), p. 368. Available at: http://jnnp.bmj.com/content/79/4/368.abstract.

21. Lill, C.M. and Klein, C. 2017. Epidemiologie und Ursachen der Parkinson-Erkrankung. Der Nervenarzt 88(4), pp. 345–355. Available at: https://doi.org/10.1007/s00115-017-0288-0.

22. Lukowicz, P. 2004. Wearable systems for health care applications Collaborative Interactive Learning View project TRAINWEAR: a Real-Time Assisted Training Feedback System with Fabric Wearable Sensors View project. Article in Methods of Information in Medicine . Available at: https://www.researchgate.net/publication/8480044.

23. Montaña, D., Campos-Roca, Y. and Pérez, C.J. 2018. A Diadochokinesis-based expert system considering articulatory features of plosive consonants for early detection of Parkinson's disease. Computer Methods and Programs in Biomedicine 154, pp. 89–97. doi: 10.1016/j.cmpb.2017.11.010.

24. Moro-Velazquez, L., Andres Gomez-Garcia, J., Godino-Llorente, J.I., Villalba, J., Rusz, J., Shattuck-Hufnagel, S. and Dehak, N. [2019]. A forced gaussians based methodology for the differential evaluation of Parkinson's Disease by means of speech processing.

25. Racette, B.A., Good, L.M., Kissel, A.M., Criswell, S.R. and Perlmutter, J.S. 2009. A Population-Based Study of Parkinsonism in an Amish Community. Neuroepidemiology 33(3), pp. 225–230. Available at: https://www.karger.com/DOI/10.1159/000229776.

26. Rovini, E., Maremmani, C. and Cavallo, F. 2017. How wearable sensors can support parkinson's disease diagnosis and treatment: A systematic review. Frontiers in Neuroscience 11(OCT). doi: 10.3389/fnins.2017.00555.

27. Saravanan, S., Ramkumar, K., Adalarasu, K., Sivanandam, V., Kumar, S.R., Stalin, S. and Amirtharajan, R. 2022. A Systematic Review of Artificial Intelligence (AI) Based Approaches for the Diagnosis of Parkinson's Disease. Archives of Computa-

tional Methods in Engineering 29(6), pp. 3639–3653. Available at: https://doi.org/10.1007/s1183
022-09710-1.

28. Sharma, P. and S.S. and S.M. and S.A. and G.D. 2019. Diagnosis of Parkinson's disease using modified grey wolf optimization. Cognitive Systems Research 54, pp. 100–115.

29. Thomas, B. and Beal, M.F. 2007. Parkinson's disease. Human Molecular Genetics 16(R2), pp. R183–R194. Available at: https://doi.org/10.1093/hmg/ddm159.

30. Vallejo, M., Jamieson, S., Cosgrove, J., Smith, S.L., Lones, M.A., Alty, J.E. and Corne, D.W. 2017. Exploring diagnostic models of Parkinson's disease with multi-objective regression. In: 2016 IEEE Symposium Series on Computational Intelligence, SSCI 2016. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/SSCI.2016.7849884.

31. Witten, Frank and Eibe [no date]. Data Mining: Practical Machine Learning Tools and Techniques, Second Edition.

Chapter 10

Appendix

10.1 Appendix A : Project Plan

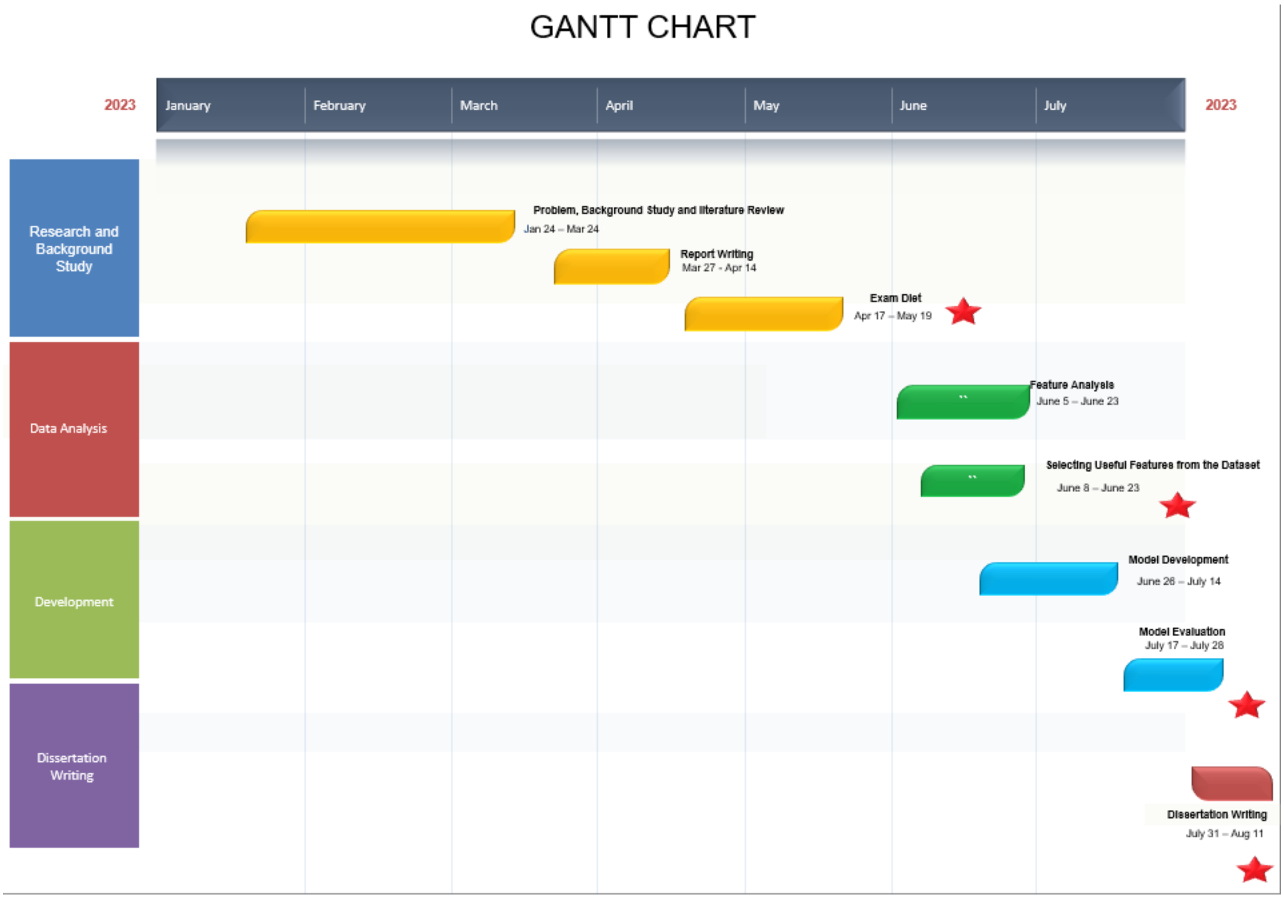


Figure 10.1: Gantt Chart for the proposed Project Plan

10.2 Appendix B : Risk Analysis

The risk connected with this project is shown in the table below. Each risk has been identified and has been assigned an effect and likelihood level (low, medium, or high). The table also includes the mitigating procedures associated with each risk in the event that it occurs.

Risk	Probablity	Impact	Mitigation
Change in scope of the Project	High	High; There may be changes in project scope as adding new features or data	Do a thorough research to find if the available features will be enough for the project aim.
Unavailability of Supervisor	Medium	High;lack of communication in obtaining feedback	Email, phone, or Skype.
Loss of Code or Hardware issue	Low	High; Redo all the development from the scratch	Keep backups.
Model Over fitting	Medium	High; Difficulty in obtaining acceptable results	Use more complex or different validation technique.
Model Under fitting	Medium	Medium; Difficulty in obtaining acceptable results	Increase complexity of the model by adding more features.
Project falls behind plan	Medium	Medium; Make changes in the project plan	Allocate more time for the project.
Sickness	Low	Medium; Project Development gets affected	Request for deadline extension.

Table 10.1: Risk Analysis Table