# TABLE OF CONTENTS

# LIST OF FIGURES

# ABSTRACT

Parkinson's disease is a progressive disorder of the central nervous system affecting movement and inducing tremors and stiffness. It has 5 stages to it and affects more than 1 million individuals every year in India. This is chronic and has no cure yet. It is a neurodegenerative disorder affecting dopamine-producing neurons in the brain.

People aged over 60 years are majorly affected by Parkinson's disease. People with Parkinson's disease suffer from speech impairments like dysphonia (defective use of the voice), hypophonia (reduced volume), monotone (reduced pitch range), and dysarthria(difficulty with the articulation of sounds or syllables). Parkinson's disease sufferers get worse over time as the normal bodily functions, including breathing, balance, movement, and heart function worsen.

Using Machine Learning with python we can develop a program to diagnose Parkinson's disease in a person early depending on certain medical procedures(Voice Parameters) and treat those patients accordingly.

# CHAPTER 1

## INTRODUCTION

### 1.1 Introduction

Parkinson's disease is a progressive disorder of the central nervous system affecting movement and inducing tremors and stiffness. It has 5 stages to it and affects more than 1 million individuals every year in India. This is chronic and has no cure yet. It is a neurodegenerative disorder affecting dopamine-producing neurons in the brain.

Parkinson's disease is due to the Dopamine deficiency. When there is a decrease in the dopamine level it leads to the Parkinson's Disease. Some of the factors responsible for this disease include genes, environment, triggers. According to some sources, there are about 7-10 million people suffering from this disease. People aged over 60 years are majorly affected by Parkinson's disease. People with Parkinson's disease suffer from

- Tremor
- Bradykinesia (Slowness of movement)
- Rigid muscles/Stiff limbs
- Muscle twisting

- Spasms or Cramps
- Stooped posture
- Decreased facial expressions
- Handwriting changes
- Depression and anxiety
- Chewing and swallowing problems
- Drooling
- Urinary problems
- Hallucinations and delusions
- Skin problems
- Loss of smell
- Low blood pressure
- Restless Leg Syndrome

Other symptoms of this disease include speech impairments like dysphonia (defective use of the voice), hypophonia (reduced volume), monotone (reduced pitch range), and dysarthria (difficulty with the articulation of sounds or syllables). Parkinson's disease sufferers get worse over time as the normal bodily functions, including breathing, unsteady walk and balance, and coordination problems, movement, and heart function worsens.

Using Machine Learning with Python we can develop a program to diagnose Parkinson's disease in a person in the early stage depending on certain medical procedures and readings (Biomedical Voice Measurements in our case) and treat those patients accordingly.

## 1.2 Motivation Of The Project

The main objective of this project is to understand what is Parkinson's disease and to detect the early onset of the disease. Machine Learning is one of the important factors that help us in achieving the feat.

This is one of the important projects under Machine Learning because it comes under Health Care Domain and Health Care is one of those fields where Machine Learning can contribute a lot. We can use these Machine Learning techniques to diagnose and find certain diseases early and treatments can be started accordingly and thus the patient's life can be saved.

Parkinson's Disease is a progressive disease and the symptoms of the disease may worsen over time as the condition continues if no proper attention is given on time. Currently, there is no cure found for this disease but the treatments available will only assist in delaying progression and in subdued symptoms. Since these treatments are essential in the early stages of the disease, there is a strong motivation for academic research into improved diagnostic procedures.

Machine Learning provides us with colossal opportunities for computer-aided classification and diagnosis that helps us to reduce unavoidably unreliability and inherent diagnostic variabilities in healthcare, provide guidance and speed up decision making. These methods help us to detect the onset of Parkinson's so that further treatments can be provided.

## 1.3 Problem Statement

In our Python Machine Learning Project**,** we are going to use certain python libraries such as NumPy, pandas**,** and (train-test split, Accuracy score, StandardScaler, and **Support Vector Machine**) from Scikit-learn to build a Machine Learning model to diagnose Parkinson's Disease with the help of a Parkinson's dataset which consists of Bio-Medical Voice Measurements of the different patients.

# CHAPTER 2

## LITERATURE SURVEY

### 2.1 Existing System

Currently, there are no specific tests available to diagnose Parkinson's Disease. Neurologists (The doctor who is trained in the nervous system Conditions) will diagnose the disease manually based on the patient's medical history, signs and symptoms, and also with the help of neurological and physical examinations.

Some of the tests the neurologists may suggest that supports the suspicion of having the Parkinson's Disease include Single-Photon Emission Computerized Tomography(SPECT) scan called a Dopamine Transporter Scan (DaTscan), Imaging tests such as Ultrasound of the brain, (MRI) Magnetic Resonance Imaging and PET scans. However these tests aren't particularly helpful for diagnosing Parkinson's disease. But ultimately it is the symptoms and the neurological tests that helps determine the correct diagnosis.

With these existing systems, Parkinson's is detected in the second stage only and this delayed detection results in some serious medical challenges.

## 2.2 Proposed System

With the help of the Python libraries such as NumPy, pandas and scikit-learn from which the sub-modules such as Accuracy score, StandarScaler, train-test split, and Support Vector Machine (SVM) and also the Parkinson's Dataset to develop a machine learning model to diagnose Parkinson's Disease.

In some cases, the researches on this disease have been dependent on Machine Learning methods. With these Machine Learning methods, the computers are trained to make them capable of detecting the presence of the disease with the help of certain medical data of the particular patient

There are two kinds of Machine Learning Algorithms,

i). Supervise
ii). Unsupervised

i). Supervised Algorithms

These algorithms are first trained on the patient data with the correct diagnosis and then they are tested for accuracy by evaluating them against the data that hasn't been used to train them. By doing this we get to know whether the machine has captured the criteria and patterns in the given data or the machine had just memorized the data

points of training data and with the help of the accuracy score, we can determine how good and accurate our model is.

k-nearest neighbors, Support Vector Machines, Decision trees are some of the methods that come under Supervised Algorithms.

ii). Unsupervised Algorithms

Unlike Supervised Algorithms, Unsupervised Algorithms are not trained with a correct diagnosis, instead, they operate on the patterns, conditions, and criteria of the Patient data. The algorithms are not trained with the status of Parkinson's Disease. These algorithms summarize the patient data by determining the most relevant measurements and also study the internal variations and structure of the data.

**SUPPORT VECTOR MACHINE (SVM):**

With respect to Supervised Machine Learning Algorithm, Support Vector Machine is used in a Machine Learning model with problem statement related to classification and regression. In our case, we are going to use SVM for classification problem. The main aim of Support Vector Machine with respect to classification problems is to

separate the whole data into two classes with a Hyperplane.

Some of the important factors of Support Vector Machine are,

**Support vectors**: The nearest datapoints to the Hyperplane from both the sides. These points help us to differentiate between two classes

**Hyperplane:** A hyperplane is a decision plane which separates a data into two classes based on their characteristic features. The Hyperplane is selected in such a way that the marginal distance must be maximum.

**Margin**: The perpendicular distance from Hyperplane to the Support vectors.

**Working of the Support Vector machine**:

When we train our Support Vector Machine model with our training data, the SVM model will classify the data into two separate classes based on their Characteristics with the help of a Hyperplane

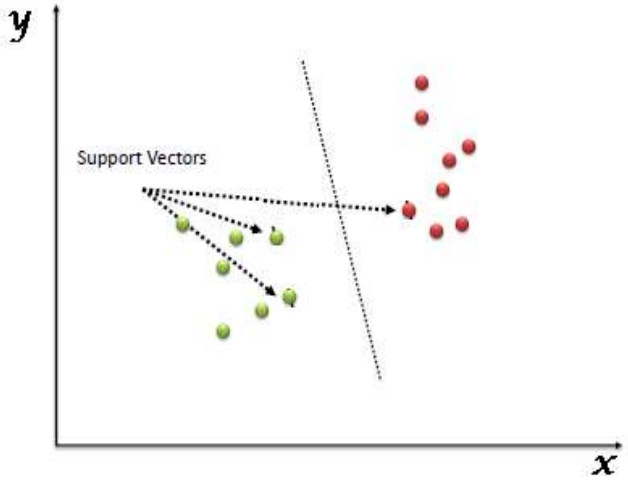


Fig.2.2.1Support Vector Machine

This SVM model is trained with the Dataset and evaluated with the data which hasn't been used to train the model and the accuracy of the model is determined to check whether the designed model is good or not. Once the model is trained, with the help of Voice Measurements as input from the user, presence or absence of the Disease for the particular patient is detected.

**Libraries used:**

1. NumPy
2. Pandas
3. sklearn
    i). train_test_split
    ii). StandardScaler
    iii). svm
    iv). accuracy score

**Parkinson's Dataset**: It is a dataset containing Biomedical voice measurements of 31 people out of which 23 having Parkinson's Disease. The dataset consists of 195 rows and 24 columns which includes 6 recordings per patient with a size of approximately 36.7KB.

**Different attributes used in Parkinson's Dataset:**

MDVP:Fo(Hz) - Average value of vocal fundamental frequency

MDVP:Fhi(Hz) - Maximum value of vocal fundamental frequency

MDVP:Flo(Hz) - Minimum value of vocal fundamental frequency

[MDVP:Jitter(%),MDVP:Jitter(Abs),MDVP:RAP,MDVP:PPQ,Jitter:DDP] are the measures of variation in fundamental frequency

[MDVP:Shimmer,MDVP:Shimmer(dB),Shimmer:APQ3,Shimmer:APQ5,MDVP:APQ,Shimmer:DDA] are the several measures of variation in amplitude.

NHR,HNR - Two measures of ratio of noise to tonal components in the voice

status - Health status of the subject (one) - Parkinson's, (zero) - healthy

RPDE,D2 - Two nonlinear dynamical complexity measures

DFA - Signal fractal scaling exponent

spread1,spread2,PPE - Three nonlinear measures of fundamental frequency variation

| | name | MDVP:Fo(Hz) | MDVP:Fhi(Hz) | MDVP:Flo(Hz) | MDVP:Jitter(%) | MDVP:Jitter(Abs) | MDVP:RAP | MDVP:PPQ | Jitter:DDP | MDVP:Shimmer |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | phon_R01_S01_1 | 119.992 | 157.302 | 74.997 | 0.00784 | 0.00007 | 0.00370 | 0.00554 | 0.01109 | 0.04374 |
| 1 | phon_R01_S01_2 | 122.400 | 148.650 | 113.819 | 0.00968 | 0.00008 | 0.00465 | 0.00696 | 0.01394 | 0.06134 |
| 2 | phon_R01_S01_3 | 116.682 | 131.111 | 111.555 | 0.01050 | 0.00009 | 0.00544 | 0.00781 | 0.01633 | 0.05233 |
| 3 | phon_R01_S01_4 | 116.676 | 137.871 | 111.366 | 0.00997 | 0.00009 | 0.00502 | 0.00698 | 0.01505 | 0.05492 |
| 4 | phon_R01_S01_5 | 116.014 | 141.781 | 110.655 | 0.01284 | 0.00011 | 0.00655 | 0.00908 | 0.01966 | 0.06425 |

...

| /P:Shimmer(dB) | Shimmer:APQ3 | Shimmer:APQ5 | MDVP:APQ | Shimmer:DDA | NHR | HNR | status | RPDE | DFA | spread1 | spread2 | D2 | PPE |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.426 | 0.02182 | 0.03130 | 0.02971 | 0.06545 | 0.02211 | 21.033 | 1 | 0.414783 | 0.815285 | -4.813031 | 0.266482 | 2.301442 | 0.284654 |
| 0.626 | 0.03134 | 0.04518 | 0.04368 | 0.09403 | 0.01929 | 19.085 | 1 | 0.458359 | 0.819521 | -4.075192 | 0.335590 | 2.486855 | 0.368674 |
| 0.482 | 0.02757 | 0.03858 | 0.03590 | 0.08270 | 0.01309 | 20.651 | 1 | 0.429895 | 0.825288 | -4.443179 | 0.311173 | 2.342259 | 0.332634 |
| 0.517 | 0.02924 | 0.04005 | 0.03772 | 0.08771 | 0.01353 | 20.644 | 1 | 0.434969 | 0.819235 | -4.117501 | 0.334147 | 2.405554 | 0.368975 |
| 0.584 | 0.03490 | 0.04825 | 0.04465 | 0.10470 | 0.01767 | 19.649 | 1 | 0.417356 | 0.823484 | -3.747787 | 0.234513 | 2.332180 | 0.410335 |

Fig.2.2.2 Parkinson's Dataset (First 5 lines)

## 2.3 Objectives of the Proposed System

The main objective of this project is to understand what is Parkinson's disease and to detect the early onset of the disease. Machine Learning is one of the important factors that helps us in achieving the feat.

This is one of the important projects under Machine Learning because it comes under Health Care Domain and Health Care is one of those fields where Machine Learning can contribute a lot. We can use these Machine Learning techniques to diagnose and find certain

diseases early and treatments can be started accordingly and thus the patient's life can be saved.

In our Machine Learning Project, we use Support Vector Machine (SVM), an algorithm that is one of the most commonly used algorithms in Machine Learning. Support Vector Machine is used in ML projects having Classifications and Regressions challenges. In our case we are using the classification (Support Vector Classifier) to classify the pre-processed Parkinson's Dataset.

# CHAPTER 3

## SYSTEM REQUIREMENT SPECIFICATIONS

### 3.1 Hardware Requirements

The following are needed to efficiently use the application.

| | | |
|---|---|---|
| Processor | - | Intel Core i3 and above |
| Speed | - | 2.5 GHz |
| RAM | - | 4 GB (min) |
| Hard Disk | - | 16 GB |

### 3.2 Software Requirements

Software requirements define software resource fundamentals that need to be installed on a workstation to provide optimum working of a software. The following are required for optimal development and usage of the application.

| | | |
|---|---|---|
| Operating System | - | Windows 7 and above |
| Programming Language | - | Python 3.7 |
| Compiler | - | PyCharm, Jupyter Notebook |

# CHAPTER 4

## SYSTEM DESIGN

### 4.1 Architecture design

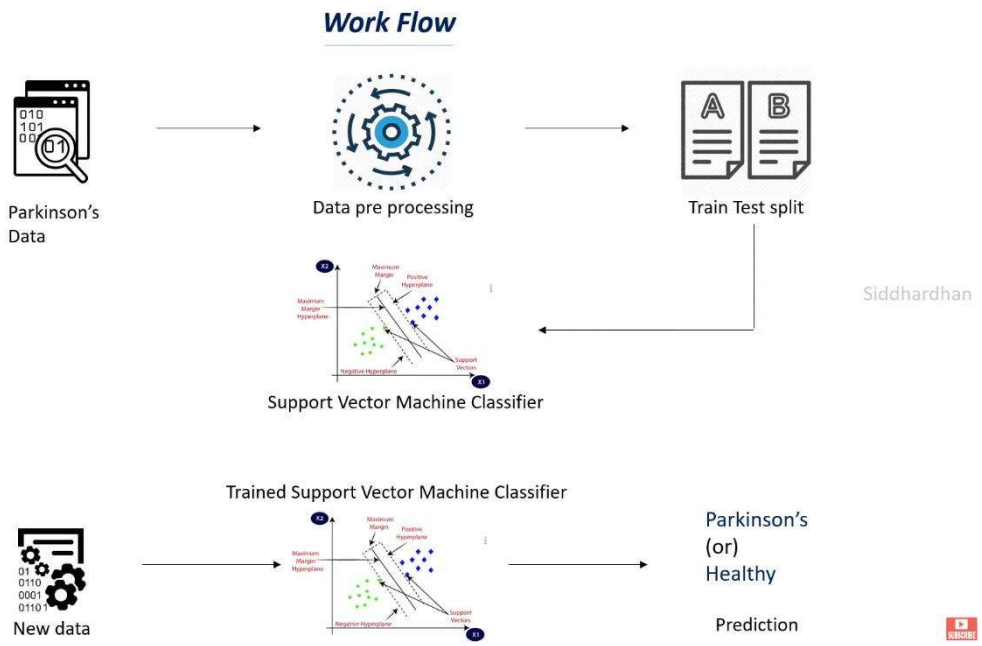


Fig.4.1.1 WorkFlow of the proposed system

**Working of the Model:**

The working of our Machine Learning Model can be divided into 6 steps.

a. Data Collection and Analysis
b. Grouping the data based on the Target variable
c. Splitting the data into Training and Test data
d. Data Standardization
e. Training and Evaluating the Support Vector Machine
f. Building the Predictive System

**a. Data Collection and Analysis**

The data from Parkinson's Dataset is loaded from the CSV file to pandas Dataframe with the help of a python library called pandas. This uploaded data is stored in a variable for further applications.

**b. Grouping the data based on the Target variable**

The status variable present in our Parkinson's Dataset is the target variable which takes the values in either 0s or 1s where the 0s represent people with Parkinson's disease and the 1s represents people without Parkinson's disease. In order to train our Machine Learning model, we have to separate the feature and target variable from the data and store it in two separate variables 'X' and 'Y' where 'X' contains all the data except the 'Name' and 'Status' columns and 'Y' contains only the 'Status' data corresponding to the respective 'X' data.

**c. Splitting the data into Training and Test data**

In order to train the Support Vector Classifier algorithm, the data has to be split into testing data and training data and is done with the help of the train_test_split library taken from the sklearn library. Here we use the training data to train the SVM and once it is trained,

we evaluate the model with the test data (the data which the SVM model is not trained with) to calculate the accuracy score.

**d. Data Standardization**

We cannot feed the raw data to our Machine Learning model. Pre-processing of the data has to be done before using the data. For example, in our case, the Parkinson's Dataset contains the data in different ranges (100s, 10s, decimals, and negatives as well) which makes the model difficult to analyze and interpret. Thus to make it simple, the Standardization of the data has to be done where the entire data is scaled down to the common range. This is done with the help of the StandardScaler library.

**e. Training and Evaluating the Support Vector Machine**

The Support Vector Machine algorithm is trained with the help of training data. The trained model is evaluated with the help of test data to calculate the accuracy score and this is done with the help of a python library called accuracy_score.

### f. Building the Predictive System

In this part, we take input from the user and convert the input data into a NumPy array with the help of the NumPy library for easy, accurate, and efficient calculations. This NumPy array is then reshaped and transformed to a standard form. Once we have the standardized data, with the help of the inbuilt predict function, we predict the Status value for the given input data in the form of either 0 or 1. And then with the help of a simple If and Else conditions we give the final verdict whether the particular person is having Parkinson's Disease or not based on the predicted Status value.

## 4.2 Flowchart of the Proposed System

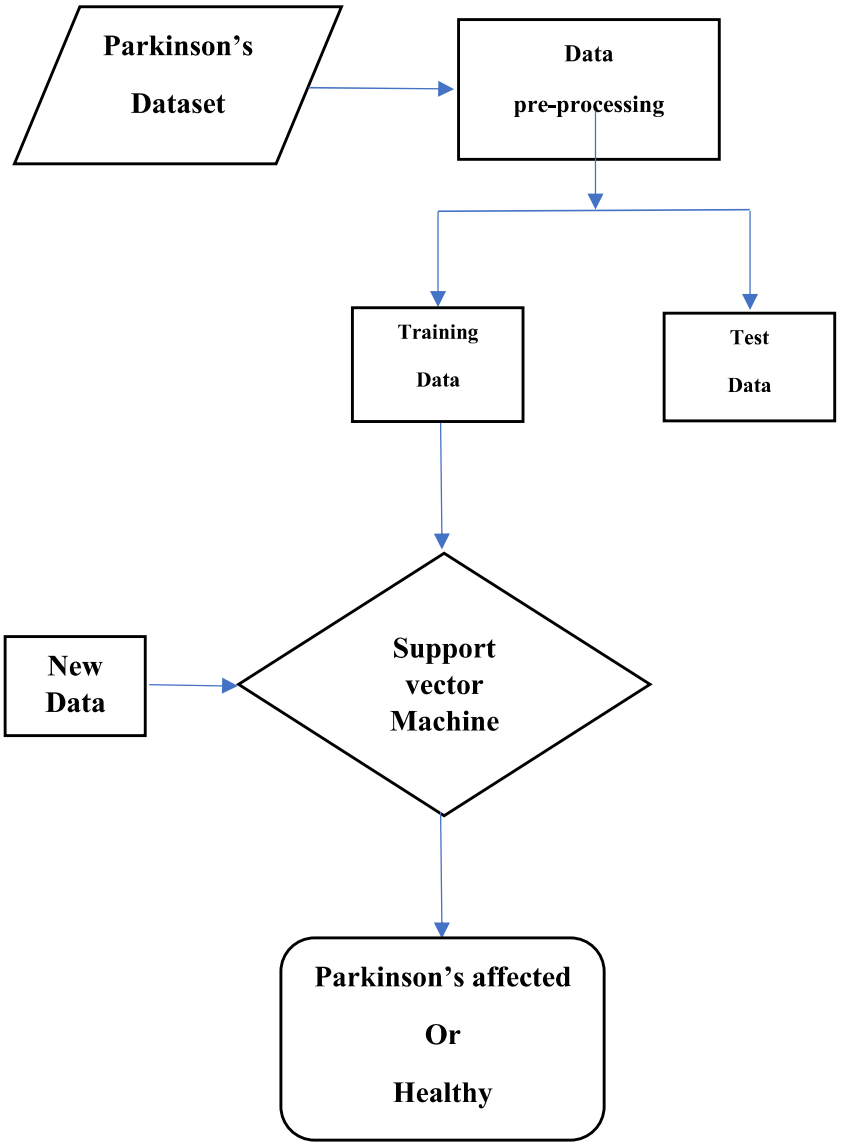


Fig.4.2.1 Flowchart of the Proposed System

# CHAPTER 5

## IMPLEMENTATION

### 5.1 Code Implementation

The code used to implement our Machine Learning Model is given below

CODE:

```
# IMPORTING THE DEPENDENCIES

import numpy as np

import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler

from sklearn import svm

from sklearn.metrics import accuracy_score

import warnings

warnings.filterwarnings('ignore')

pd.set_option('display.max_columns', None)

pd.set_option('display.max_rows', None)

# DATA COLLECTION AND ANALYSIS

# Loading the data from csv file to pandas Dataframe

Parkinsons_Data = pd.read_csv("parkinsons.csv")
```

```
# Printing the first five lines(First 5 by Default)

# print(Parkinsons_Data.head())

# Printing the total number of rows and columns

# print(Parkinsons_Data.shape)

# Prints some more details

# print(Parkinsons_Data.info())

# Prints the STATISTICAL DESCRIPTION

# print(Parkinsons_Data.describe())

# IN OUR CASE STATUS VARIABLE IS OUR TARGET VARIABLE

# Distribution of Target Variable(Status)

# 1=> People with Parkinson's Disease

# 0=> Healthy People

# print(Parkinsons_Data['status'].value_counts())

# Grouping the data based on the target variable

Parkinsons_Data.groupby('status').mean()

# Separating features and targets

X = Parkinsons_Data.drop(columns=['name' , 'status'], axis=1)

Y = Parkinsons_Data['status']

# Splitting the Data into Training Data and Test Data

X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2 ,
random_state=2)

# print(X.shape, X_train.shape, X_test.shape)

# print(X_train,Y_train)
```

```
# Data Standardization

scaler = StandardScaler()

# Training our model

scaler.fit(X_train)

X_train = scaler.transform(X_train)

X_test = scaler.transform(X_test)

# MODEL TRAINING

# SUPPORT VECTOR MACHINE MODEL

model = svm.SVC(kernel='linear')

# Training the SVM Model

model.fit(X_train, Y_train)

# EVALUATION

# Accuracy score of X_train data

X_train_prediction = model.predict(X_train)

training_data_accuracy = accuracy_score(Y_train, X_train_prediction)

print("Accuracy Score of the Training Data is:",training_data_accuracy)

# Accuracy Score of Test Data

X_test_prediction = model.predict(X_test)

test_data_accuracy = accuracy_score(Y_test,
X_test_prediction)

print("Accuracy Score of the Test Data is: ", test_data_accuracy)

# BUILDING A PREDICTIVE SYSTEM

#
MDVP:Fo(Hz),MDVP:Fhi(Hz),MDVP:Flo(Hz),MDVP:Jitter(%),MDVP:Jitter(Abs),MDVP:
```

```
RAP,MDVP:PPQ,Jitter:DDP,MDVP:Shimmer,MDVP:Shimmer(dB),Shimmer:APQ3,Shimmer:
APQ5,MDVP:APQ,Shimmer:DDA,NHR,HNR,RPDE,DFA,spread1,spread2,D2,PPE

report_results = input("Enter the report details separated by comma:")

list1 = report_results.split(',')

list2 = map(float,list1)

input_data = tuple(list2)

# CHANGING INPUT DATA TO NUMPY ARRAY

input_data_as_numpy_array = np.asarray(input_data)

# RESHAPE THE NUMPY ARRAY

input_data_reshaped =
input_data_as_numpy_array.reshape(1,-1)

# STANDARDISE THE DATA

std_data = scaler.transform(input_data_reshaped)

prediction = model.predict(std_data)

print("STATUS:",prediction)

if prediction[0]==0:

    print("The person is HEALTHY and is not having any symptoms of PARKINSON'S DISEASE")

else:

    print("The person is having PARKINSON'S DISEASE")
```

# CHAPTER 6

## RESULT

### 6.1 Output

For the above implemented code, the following output will be produced.

```
D:\venv\Scripts\python.exe D:/pythonProject4/MiniProject.py

Accuracy Score of the Training Data is:  0.8846153846153846
Accuracy Score of the Test Data is:  0.8717948717948718

Enter the report details separated by comma:
```

Here the first and second line gives us the Accuracy score of Training and Test data respectively. It says that the accuracy of our training data is 88.46% and that of test data is around 87.18%.

The third line asks the user to input the report details of a particular person i.e., the readings of Biomedical Voice Measurements of the person separated by comma. Initially, the input data will be converted into a tuple and then it is converted into a NumPy array for further use. The input data must contain 22 values separated by comma. The system will throw an error in case of miscounts

Once the user gives the input, the following output will be produced

```
D:\venv\Scripts\python.exe D:/pythonProject4/MiniProject.py
Accuracy Score of the Training Data is:  0.8846153846153846
Accuracy Score of the Test Data is:  0.8717948717948718
Enter the report details separated by comma:
107.33200,113.84000,104.31500,0.00290,0.00003,0.00144,0.00182,0.00431,0.01567,0.13400,0.00829,0.00946,0.01256,0.02487,0.00344,26.89200,0.637420,0.763262,-6.167603,0.183721,2.064693,0.163755
STATUS: [1]
The person is having PARKINSONS DISEASE
Process finished with exit code 0
```

OR

```
D:\venv\Scripts\python.exe D:/pythonProject4/MiniProject.py
Accuracy Score of the Training Data is:  0.8846153846153846
Accuracy Score of the Test Data is:  0.8717948717948718
Enter the report details separated by comma:
229.40100,252.22100,221.15600,0.00205,0.000009,0.00114,0.00113,0.00342,0.01457,0.12900,0.00769,0.00957,0.01016,0.02308,0.00300,26.41500,0.276850,0.673636,-7.496264,0.056844,2.003032,0.073581
STATUS: [0]
The person is HEALTHY and is not having any symptoms of PARKINSONS DISEASE
Process finished with exit code 0
```

# CHAPTER 7

# CONCLUSION AND FUTURE ENHANCEMENT

## 7.1 Conclusion

Parkinson's Disease is a progressive disorder and has five stages to it. It is chronic and has no cure yet. It worsens the patient's condition slowly in the further stages. Therefore, detecting the disease in the early stages may help us to provide proper treatments and medications so that the patient's life can be saved.

With the help of the existing clinical methods, Parkinson's Disease can be diagnosed in the second stage only which creates medical challenges to the doctors.

Our Machine Learning project uses the Parkinson's dataset to train the model which aims to detect Parkinson's Disease in the early stages possible. This helps the doctors to study the case thoroughly and treat the patient accordingly.

## 7.2 Future Enhancement

In our current project, we use Parkinson's Dataset containing Bio-Medical Voice Measurements of different people to train our model and detect the presence of the disease. This project can be enhanced in future by using some advanced dependencies to take vocal inputs directly from the user and interpret the required data with the help of certain algorithms.

# REFERENCES

[1] http://ijariie.com/AdminUploadPdf/EARLY_DETECTION_OF_PARKINSON_S_DISEASE_USING_MACHINE_LEARNING_ijariie11591

[2] https://www.datacamp.com/community/tutorials/svm-classification-scikit-learn-python

[3] https://www.mayoclinic.org/diseases-conditions/parkinsons-disease/diagnosis-treatment/drc-20376062

[4] https://www.kaggle.com/nidaguler/parkinsons-data-set