A

PROJECT REPORT

ON

Parkinson's Disease Detection System

Submitted in partial fulfillment of the requirements for the degree of Bachelor of Engineering in "**Information Technology**"

By:

Sr.No	Name	ID No.
1	Pranjal Mahajan	VU4F2021061
2	Sanika Pitre	VU4F2021063
3	Shruti Gaikwad	VU4F2021079
4	Shaikh Barirah Saquib	VU4F2021104
5	Hrishikesh Gupta	VU4F2021069

Under the Guidance of

Prof. Ravindra Pande

Department of Information Technology

Vasantdada Patil Pratishthan's College of Engineering &Visual Arts

Vasantdada Patil Educational Complex, Eastern Express Highway, Near Everard Nagar, Sion-Chunabhatti, Mumbai – 400022.

UNIVERSITY OF MUMBAI

A.Y 2023-24

CERTIFICATE

This is to certify that the project entitled **“Parkinson’s Disease Detection System”** is a bonafide work of PranjalMahajan(vu4f2021061), Sanika Pitre(vu4f2021063), Shruti Gaikwad(vu4f2021079), Shaikh Barirah Saquib (vu4f2021104),Hrishikesh Gupta(VU4F2021069) submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology**.

Prof. Ravindra Pande
GUIDE

Dr. PRADIP MANE
HOD(IT)

Dr. ALAM N.SHAIKH
Principal

Project Report Approval for B. E.

This project report entitled **"Parkinson's Disease Detection System"** Pranjal Mahajan(vu4f2021061), Sanika Pitre(vu4f2021063), Shruti Gaikwad(vu4f2021079), Shaikh Barirah Saquib (vu4f2021104),Hrishikesh Gupta(VU4F2021069) is approved for the degree of **Bachelor of Engineering** in **Information Technology**.

Examiners

1. ______________________________

2. ______________________________

Date:

Place: Mumbai – 22

ABSTRACT

People nowadays suffer from a variety of diseases as a result of their living habits and the state of the environment. As a result, predicting sickness at an early stage becomes a crucial task. But the accurate prediction based on symptoms becomes too difficult for the doctor. The correct prediction of disease is the most challenging task. However, the analysis accuracy is reduced when the quality of medical data is incomplete. For processing such a large amount of data, we have proposed a "Disease Prediction System" which predicts whether an individual is a patient or not using all their general information and also the symptoms. In this system, we have rigorously used various supervised ML algorithms and worked with the ones that gave the highest accuracy for that particular disease, to build a reliable model that makes the most accurate predictions. The abstract discusses a "Parkinson's Disease Detection System," highlighting its significance in early disease detection and management. This system employs a comprehensive approach, utilizing data from clinical assessments, medical imaging, and patient-reported symptoms. Machine learning and AI techniques are used to process and analyze the data, ultimately building a predictive model. Key components include data collection, feature extraction, and model development. The system integrates medical tests, brain scans, and lifestyle information, enhancing its accuracy. It represents a promising development in medical technology, with the potential to revolutionize early diagnosis and contribute to Parkinson's disease research and treatment.

Keywords: Artificial Intelligence, Machine Learning, Parkinson's Detection System, Prediction, Disease, Early Detection, Logistic Regression, Decision Trees,Data Preprocessing, Model Selection, Healthcare, , Data Science, Predictive Modeling.

ACKNOWLEDGMENT

With pleasure we take this opportunity to express our fervent & deepest gratitude and commendation to our guide and faculty of IT Engineering Department, Prof Ravindra Pande for his remarkable cooperation, guidance, monitoring and constant encouragement.

The blessings, help and advice's given by him from time to time shall take us to higher echelons of success in the journey that we are embarking upon. We also take pleasure in expressing our extreme gratefulness to our other faculty members for their cordial support, valuable inputs & information and directions which helped us complete this project.

New disease prediction system to leverage the advancement in technologies specially in Healthcare Artificial Intelligence (AI) , Machine Learning (ML) & robotics. We as Technological enthusiasts from engineering background are moving towards better adoptions & implementations day by day in Indian Healthcare Industry.

The co-operation extended by all of them has been commendable. We are grateful to ALMIGHTY, our parents, friends & colleagues for their continuous encouragement, solidarity and faith shown by them in all our endeavors.

LIST OF FIGURE

TABLE OF CONTENT

CHAPTER 1

INTRODUCTION

1.1 Introduction

Parkinson’s disease (PD) is a progressive neurological disorder associated with progressive neuronal loss of the substantia nigra and other brain structures and is characterized by tremors, bradykinesia, rigidity, and postural instability along with other symptoms such as sleep disorders (non-motor features), cardiac arrhythmia, and constipation. Alteration of voice and speech is one of the features of PD. PD is an age-related and the second most common neurodegenerative condition. The prevalence of PD increases in the aging population, thus increasing the economic burden on society. The cardinal motor symptoms of PD are identified relatively late in the pathological process (i.e. when approximately 50% of dopaminergic neurons are lost in the substantia nigra); thus, PD diagnosis is often delayed. Early detection or prediction of PD could make early pharmacological and non- pharmacological management possible, which could slow its progression. Unified Parkinson’s Disease Rating Scale or UPDRS, which shows symptoms’ presence and severity, is mainly used in tracking PD symptom progression. UPDRS is considered the well-validated test and the most widely used clinical rating scale for patients with PD. The benefits of early prediction and management of PD would affect not only the individual (and their families) but also the wider society and research community. It highlights the complex nature of the disease, with motor and non-motor symptoms, and how traditional diagnostic methods can be limited. The proposed system takes a comprehensive approach, incorporating data from clinical assessments, medical imaging, and patient-reported symptoms, which are processed using advanced machine learning techniques. Key components include data collection, feature extraction, and model development. The system's accuracy is rigorously validated, even in the early disease stages. Ultimately, the system offers the potential for more accurate and timely diagnosis, improving the lives of Parkinson's patients and advancing research and treatment options.

1.2 Aim and Objective of the project

Aim:

The aim of our AI/ML project is to develop a highly accurate Parkinson's Disease Detection System that enables early diagnosis and uses diverse data sources. We prioritize non-invasive diagnosis to improve the quality of life for patients. This project supports healthcare professionals, contributes to research, fosters innovation, and empowers patients by providing early health information. Ultimately, our goal is to advance the field of medical technology and enhance the management of Parkinson's disease.

Objectives:

1. **Early Detection**: Develop a system that can identify Parkinson's disease in its early stages, allowing for timely intervention and treatment.

2. **High Accuracy**: Achieve a level of diagnostic accuracy that surpasses traditional methods through the utilization of AI/ML algorithms.

3. **Multimodal Data Integration**: Combine clinical assessments, medical imaging, and patient-reported symptoms to create a comprehensive dataset for analysis.

4. **Non-Invasive Diagnosis**: Prioritize non-invasive diagnostic methods to make the process more patient-friendly and cost-effective.

5. **Decision Support for Healthcare Professionals**: Provide healthcare providers with a tool that aids in informed decision-making and personalized treatment planning.

6. **Quality of Life Improvement**: Enhance the quality of life for individuals with Parkinson's disease by facilitating early diagnosis and intervention.

7. **Research Contribution**: Contribute valuable data for ongoing Parkinson's disease research, supporting the development of innovative treatments and interventions.

8. **Ethical Considerations**: Ensure the project adheres to ethical guidelines and data privacy standards in the development and deployment of the Parkinson's Disease Detection System.

1.3 Scope of the Project

Comprehensive Data Integration: Utilizing AI/ML, it can incorporate a wide range of data sources, including clinical assessments, medical imaging, and patient-reported symptoms, providing a holistic view of the patient's health.

Telemedicine Integration: The system can be integrated with telemedicine platforms, allowing for remote and real-time monitoring of Parkinson's patients, enhancing access to care.

Feature Importance Analysis: The project will include a detailed analysis of feature importance for each algorithm, allowing for a deeper understanding of which attributes contribute significantly to diabetes prediction. This analysis will help in feature selection and refinement, which is crucial for the optimal performance of the Naive Bayes-based model. By comparing feature importance across different algorithms, insights into the dataset's characteristics will be gained.

Model Ensemble Strategies: To enhance prediction accuracy, the project will explore ensemble techniques, such as stacking or voting classifiers. By combining predictions from various models, including Naive Bayes, the system can potentially yield more reliable and robust diabetes predictions. Ensemble approaches will be considered as a part of the scope to ensure that the final prediction model is more resilient and accurate.

The scope for a Parkinson's Disease Detection System using AI/ML extends beyond diagnosis to positively impact patient outcomes, healthcare professionals, and the broader field of medical technology and research. It holds the potential to revolutionize Parkinson's disease management and contribute to better health and well-being for affected individuals.

CHAPTER 2
LITERATURE SURVEY

1. T. J. Wroge, Y. Özkanca, C. Demiroglu, D. Si, D. C. Atkins and R. H. Ghomi, "Parkinson’s Disease Diagnosis Using Machine Learning and Voice," 2018 IEEE Signal Processing in Medicine and Biology Symposium (SPMB), Philadelphia, PA, USA, 2018, pp. 1-7, doi: 10.1109/SPMB.2018.8615607.

Date	Methodology	Result
2018	In a study on Parkinson's Disease (PD) diagnosis, raw audio data was cleaned using a Voice Activation Detection (VAD) algorithm to remove background noise. Two feature extraction methods, including the Minimum Redundancy Maximum Relevance (mRMR) technique, were applied to the audio data. mRMR ranked features by predictive correlation. The study tested varying feature lengths and found that 1200 features offered the best categorical accuracy for PD diagnosis. Additionally, the Geneva Minimalistic Acoustic Parameter Set (GeMaps) was used for feature extraction, yielding 62 features per audio sample. Various machine learning classifiers were employed, including decision trees, support vector machines (SVM), and deep neural networks, to optimize accuracy in PD diagnosis. These models were fine-tuned and evaluated using metrics such as accuracy, F-1, recall, and precision.	The study evaluated various machine learning classifiers for Parkinson's Disease (PD) diagnosis using metrics like recall, precision, and F-1 scores. Models were assessed on AVEC and GeMaps datasets. The Random Forest model achieved high AUC and accuracy but lower recall and F-1 scores. The Artificial Neural Network performed best in overall accuracy and recall but had challenges with the GeMaps dataset. The Decision Tree Classifier showed moderate performance, while the Gradient Boosted Classifier excelled with high accuracy, precision, and F-1 scores. The Extra Tree Classifier achieved high precision and accuracy but lower recall and F-1 scores. The SVM model demonstrated strong accuracy and F-1 scores on the AVEC dataset, with most models performing better with AVEC features.

2. W. Wang, J. Lee, F. Harrou and Y. Sun, "Early Detection of Parkinson’s Disease Using Deep Learning and Machine Learning," in IEEE Access, vol. 8, pp. 147635-147646, 2020, doi: 10.1109/ACCESS.2020.3016062.

Date	**Methodology**	**Result**
2020	This study presents a deep learning framework for early Parkinson's disease (PD) detection, divided into training and testing stages. Data is obtained from the Parkinson's Progression Markers Initiative (PPMI) database, comprising 401 early PD patients and 183 healthy individuals. Thirteen features, including RBDSQ score, UPSIT score, CSF biomarkers, and SPECT imaging, are considered for early PD detection. Exploratory analysis reveals five critical features distinguishing healthy individuals from PD patients. Machine learning methods, including Deep Learning (DEEP), are employed to build a model for early PD diagnosis. Features are log-transformed and scaled to unify their scales. The study focuses on identifying key features and their impact on PD diagnosis.	The study utilizes several metrics for assessing the performance of machine methods in distinguishing Parkinson's patients:Accuracy evaluates the proportion of correctpredictions And Over allperformance. Sensitivity (or recall) measures the ability to correctly detect Parkinson's patients. Specificity quantifies the proportion of correctly predicted normal individuals. Precision assesses the relevance of predicted positives.The F1 score, a harmonic mean of precision and sensitivity, provides an overall evaluation of model performance.

3. S. Raval, R. Balar and V. Patel, "A Comparative Study of Early Detection of Parkinson's Disease using Machine Learning Techniques," 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), Tirunelveli, India, 2020, pp. 509-516, doi: 10.1109/ICOEI48184.2020.9142956.

Date	**Methodology**	**Result**
2020	Two key aspects, finger tapping frequency and tremor at rest, are used for early PD detection. Machine learning models are applied to pre-processed and normalized data from 53 participants, achieving high specificity (95-100%) and sensitivity (92-100%). A mobile touchscreen typing study with 51 subjects detects PD effectively. For tremor at rest, handwriting data from 75 individuals results in 79.4% accuracy, while a multi-modal approach combines tremor and voice changes to enhance accuracy with data from 77 participants.	The study compares machine learning algorithms based on Sensitivity and Specificity, evaluating their ability to predict positive and negative outcomes.Additionally,ensemble approaches, including Random Forest (RF), AdaBoost (AB), and Hard Voting (HV), are implemented across four modalities to enhance accuracy. RF aggregates results from sub-samples, AB assigns higher weights to weak classifiers, and HV combines multiple algorithms for predictions through majority voting. The inclusion of algorithms with accuracy above a threshold improves the results in hard voting.

4. M. Sivakumar, A. H. Christinal and S. Jebasingh, "Parkinson's disease Diagnosis using a Combined Deep Learning Approach," 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 2021, pp. 81-84, doi: 10.1109/ICSPC51351.2021.9451719.

Date	Methodology	Result
2021	A method combines LeNet and LSTM architectures to classify 102 spiral drawings as "Healthy" or "Parkinson's," followed by severity level classification for Parkinson's images. The performance metric is accuracy, and adjustments to hidden layers and epochs are explored to optimize model performance for early Parkinson's disease diagnosis. The approach aims to assist individuals with Parkinson's disease by providing valuable diagnostic information..	The paper addresses the need for early Parkinson's disease diagnosis using deep learning techniques to improve treatment outcomes. The proposed system combines LeNet and LSTM methods for accuracy and severity level assessment, aiming to aid individuals with Parkinson's disease.

5. A. Ouhmida, A. Raihani, B. Cherradi and Y. Lamalem, "Parkinson's disease classification using machine learning algorithms: performance analysis and comparison," 2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), Meknes, Morocco, 2022, pp. 1-6, doi: 10.1109/IRASET52964.2022.9738264.

Date	Methodology	Result
2022	The study evaluates nine Machine Learning Algorithms (MLA) for Parkinson's disease detection, including SVM, Logistic Regression, Discriminant Analysis, KNN, Decision Tree. These algorithms are applied to a dataset of 240 speech measurements with 44 features. KNN achieved the highest accuracy rate at 97.22% and an F1-score of 97.30%.	In the evaluation of nine Machine Learning Algorithms for Parkinson's disease detection, K-Nearest Neighbors (KNN) achieved the highest accuracy rate, scoring 97.22%. Additionally, KNN obtained an impressive F1-score of 97.30%. This underscores the potential of KNN in the assessment of this challenging neurodegenerative disorder.

6. S. Kamoji, D. Koshti, V. V. Dmello, A. A. Kudel and N. R. Vaz, "Prediction of Parkinson's Disease using Machine Learning and Deep Transfer Learning from different Feature Sets," 2021 6th International Conference on Communication and Electronics Systems (ICCES), Coimbatre, India, 2021, pp. 1715-1720, doi: 10.1109/ICCES51350.2021.9488944.

Date	**Methodology**	**Result**
2021	This study focuses on early detection of Parkinson's disease (PD) by analyzing various datasets. It involves the Freezing of Gait dataset to predict leg and trunk symptoms, the Parkinson Clinical speech dataset for audio frequency analysis, and the Parkinson Disease wave and spiral drawing dataset to detect handwriting impairment due to tremors. Convolutional Neural Network with Transfer Learning is applied to the image dataset for efficient diagnosis.	The study focuses on identifying early symptoms of Parkinson's disease. It uses various datasets to analyze symptoms related to gait, speech, and handwriting. Convolutional Neural Network with Transfer Learning is employed to detect handwriting impairment, offering a promising approach for early diagnosis of PD.

CHAPTER 3
EXISTING SYSTEM

Several existing systems and approaches for Parkinson's disease detection and assessment were in use. Here are some existing methods and systems:

Clinical Assessments: The primary method for diagnosing Parkinson's disease is through clinical assessments by neurologists and movement disorder specialists. They evaluate the patient's medical history, conduct physical examinations, and use standardized rating scales like the Unified Parkinson's Disease Rating Scale (UPDRS).

Imaging Techniques: Various medical imaging technologies, such as magnetic resonance imaging (MRI) and functional MRI (fMRI), are used to detect structural and functional changes in the brain associated with Parkinson's disease. DaTscan imaging, a type of single-photon emission computed tomography (SPECT) imaging, can help differentiate Parkinson's disease from other movement disorders.

Biomarker Research: Researchers are actively investigating potential biomarkers for Parkinson's disease in blood, cerebrospinal fluid, and other bodily fluids. Identifying reliable biomarkers could significantly aid in diagnosis and monitoring.

Deep Learning and AI: AI and machine learning techniques are being applied to analyze medical imaging data and sensor data from wearables for more accurate and objective assessment of Parkinson's disease.

Robot-Assisted Assessments: Advanced technologies like robotics are used to assess motor functions in patients and monitor their progress over time.

It's important to note that the diagnosis and management of Parkinson's disease often involve a combination of these methods. Advances in AI, machine learning, and the integration of technology are continuously improving the accuracy and efficiency of diagnostic procedures and disease monitoring. It is advisable to consult with a healthcare professional for the most up-to-date information on Parkinson's disease detection and management.

CHAPTER 4

DESIGN AND IMPLEMENTATION

4.1 PROPOSED SYSTEM

1. Data Collection: Gather data from various sources, including clinical assessments, wearable devices, medical imaging, and patient input through mobile apps or online platforms.

2. Data Integration: Combine and process data using AI/ML algorithms to create a comprehensive patient profile that includes both motor and non-motor symptoms.

3. Diagnostic Model: Develop a predictive model that utilizes the integrated data to assess the likelihood of Parkinson's disease and its stage.

4. User Interface: Create a user-friendly interface for both patients and healthcare professionals to interact with the system. This can include mobile apps, web platforms, or even voice-activated interfaces.

5. Telemedicine Integration: Enable remote consultations, data sharing, and treatment monitoring through telemedicine platforms.

6. Alert System: Implement an alert system that notifies healthcare providers and patients about changes in the patient's condition, ensuring timely intervention.

7. Data Visualization: Provide visual representations of patient data to aid in diagnosis and patient understanding.

8. Machine Learning Updates: Continuously train the AI model with new data to adapt to the evolving understanding of Parkinson's disease.

Abbreviations Feature description Multidimensional Voice Program (MDVP) analysis is a computer program which analyzes various aspects of voice, can detect abnormal voice patterns of patients with upper airway pathology. Different parameters about the voice patterns and characteristics of voice are mentioned below:

MDVP:F0 (Hz)	Average vocal fundamental frequency
MDVP:Fhi (Hz)	Maximum vocal fundamental frequency
MDVP:Flo (Hz)	Minimum vocal fundamental frequency
MDVP:Jitter(%)	MDVP jitter in percentage
MDVP:Jitter(Abs)	MDVP absolute jitter in ms
MDVP:RAP	MDVP relative amplitude perturbation
MDVP:PPQ	MDVP five-point period perturbation quotient

Jitter:DDP	Average absolute difference of differences between jitter cycles
MDVP : Shimmer	MDVP local shimmer
MDVP:Shimmer(dB)	MDVP local shimmer in dB
Shimmer:APQ3	Three-point amplitude perturbation quotient
Shimmer:APQ5	Five-point amplitude perturbation quotient
MDVP:APQ11	MDVP 11-point amplitude perturbation quotient
Shimmer : DDA	Average absolute differences between the amplitudes of consecutive periods
NHR	Noise-to-harmonics ratio
HNR	Harmonics-to-noise ratio
RPDE	Recurrence period density entropy measure
D2	Correlation dimension
DFA	Signal fractal scaling exponent of detrended fluctuation analysis
Spread1	Two nonlinear measures of fundamental
Spread2	Frequency variation
PPE	Pitch period entropy

A vocal software can be used to measure these values by simply recording a sample of voice from the patients. These are all numeric values. These values can be tested to whether they are useful and able to be to used as classifying factors for whether a patient has Parkinson’s disease or not. We can also check for which of the classification models is applicable for this purpose. We can check whether data dimensionality reduction is useful for the model to improve its performance. In this way we can see whether MDVP can be used for classification of Parkinson’s patients.

4.2 DESIGN

4.2.1 System Flow Diagram:

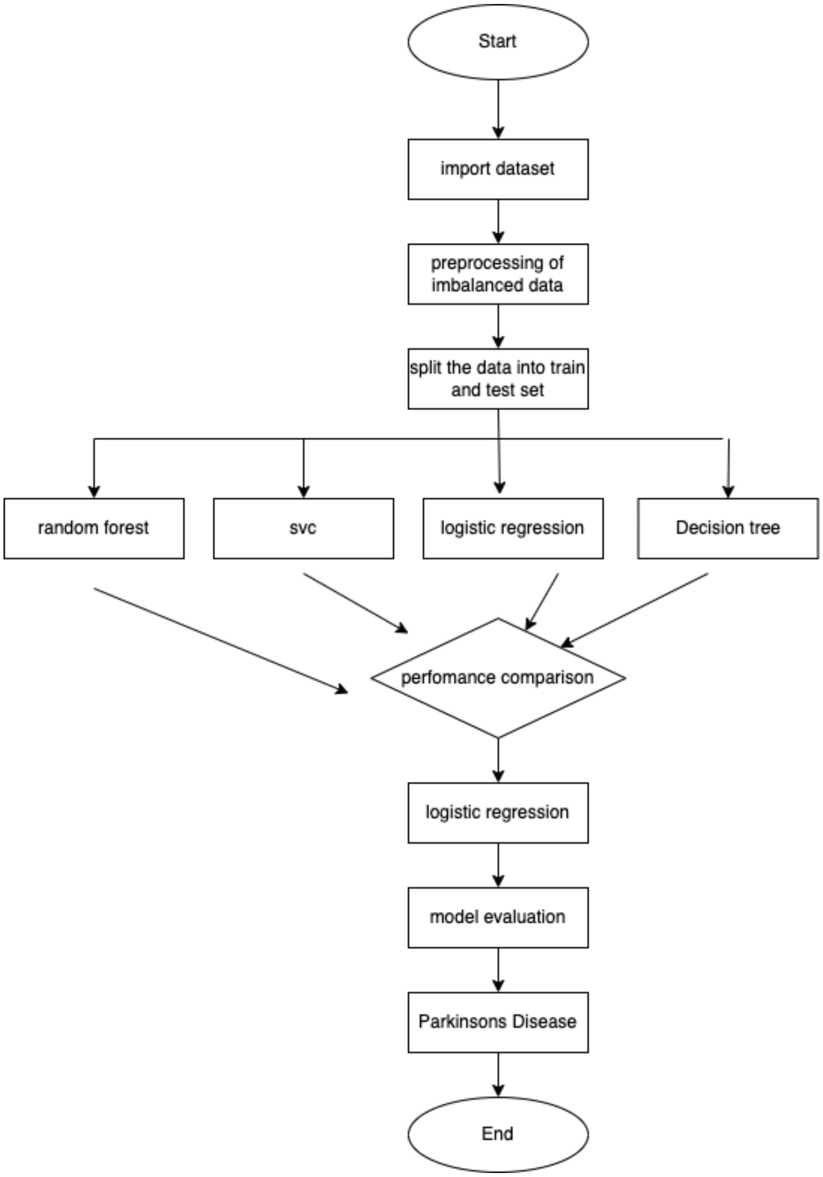


Fig 4.2.1: System Flow Diagram of proposed system

Here's a system flow for a project report on a Parkinsons

detection system:

1. **Start:** Begin the project report by introducing the problem statement and explaining theimportance of detecting Parkinsons early.

2. **Import dataset:** Collect and import the necessary dataset that includes features such as age,weight, height, BMI, and hormone levels.

3. **Preprocessing of imbalanced data:** Check the dataset for imbalanced data, handle missingvalues, and perform necessary feature engineering to prepare the data for analysis.

4. **Split the data into training and testing dataset:** Split the data into a training set and a testing set. The training set is used to train the models, and the testing set is used to evaluate theirperformance.

5. **Used algorithms:** Random Forest, Logistic Regression, SVC, and Decision Tree algorithms areused to train the models. The performance of each model is evaluated using cross-validation.

6. **Compare performances and select algorithm:** Analyze the performance of each model,compare their performances, and select the best performing algorithm for this project. In this case, logistic regression is selected based on its performance.

7. **Model evaluation:** Evaluate the performance of the selected model by calculating metrics suchas accuracy, precision, recall, and F1 score.

8. **Detect Parkinsons Disease:** Finally, use the selected model to predict whether a patient hasParkinsons disease or not based on their input features.

This system flow helps to create a structured approach to the project report and ensures that all necessarysteps are taken to develop a robust Parkinsons detection system.

4.2.2 Flowchart Diagram

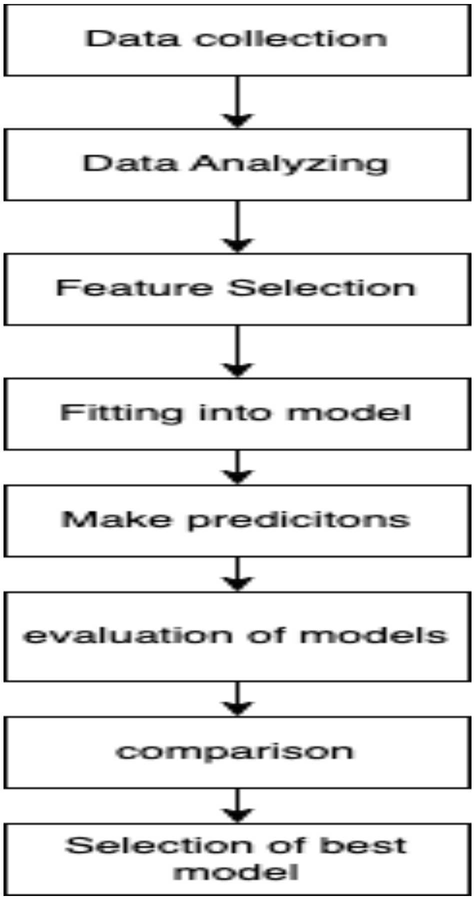


Fig 4.2.3: Flowchart of proposed system

The flowchart for a Parkinsons detection system for a project report with the mentionedcomponents can be summarized as follows:

1. Start by collecting patient data such as age, weight, height, menstrual cycle information,medical history, and family history.

2. Pre-process the data to clean, transform, and normalize it.

3. Analyze the data to identify potential features that can be used for Parkinsons detection.

4. Select the most relevant features for Parkinsons detection using feature selectiontechniques.

5. Split the data into training and testing sets for modeling.

6. Fit the data into different models such as logistic regression, decision trees, and supportvector machines.

7. Use the models to make predictions on the testing set.

8. Evaluate the performance of the models using metrics such as accuracy, sensitivity,specificity, and F1 score.

9. Compare the performance of the different models.

10. Select the best-performing model based on the evaluation metrics.

11. End the flowchart.

CHAPTER 5

5.1 CODE

Parkinson’s_Disease_Detection.ipynb

Importing libararies

```
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
from sklearn.linear_model import LogisticRegression
from sklearn.cluster import KMeans
from sklearn.ensemble import RandomForestClassifier
import matplotlib.pyplot as plt
from sklearn.metrics import confusion_matrix
from sklearn.metrics import ConfusionMatrixDisplay
```

- **Data Collection and Analysis**

```
# loading the data from csv file to a Pandas DataFrame
parkinsons_data = pd.read_csv('parkinsons.csv')
parkinsons_data.info()
```

```
Output exceeds the size limit. Open the full output data in a text
editor
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 195 entries, 0 to 194
Data columns (total 24 columns):
 #    Column            Non-Null Count  Dtype
---   ------            --------------  -----
 0    name              195 non-null    object
 1    MDVP:Fo(Hz)       195 non-null    float64
```

```
  2   MDVP:Fhi(Hz)      195 non-null    float64
  3   MDVP:Flo(Hz)      195 non-null    float64
  4   MDVP:Jitter(%)    195 non-null    float64
  5   MDVP:Jitter(Abs)  195 non-null    float64
  6   MDVP:RAP          195 non-null    float64
  7   MDVP:PPQ          195 non-null    float64
  8   Jitter:DDP        195 non-null    float64
  9   MDVP:Shimmer      195 non-null    float64
  10  MDVP:Shimmer(dB)  195 non-null    float64
  11  Shimmer:APQ3      195 non-null    float64
  12  Shimmer:APQ5      195 non-null    float64
  13  MDVP:APQ          195 non-null    float64
  14  Shimmer:DDA       195 non-null    float64
  15  NHR               195 non-null    float64
  16  HNR               195 non-null    float64
  17  status            195 non-null    int64
  18  RPDE              195 non-null    float64
  19  DFA               195 non-null    float64
...
  22  D2                195 non-null    float64
  23  PPE               195 non-null    float64
dtypes: float64(22), int64(1), object(1)
memory usage: 36.7+ KB

# checking for missing values in each column
parkinsons_data.isnull().sum() name        0
MDVP:Fo(Hz)         0
MDVP:Fhi(Hz)        0
MDVP:Flo(Hz)        0
MDVP:Jitter(%)      0
MDVP:Jitter(Abs)    0
MDVP:RAP            0
```

```
MDVP:PPQ            0
Jitter:DDP          0
MDVP:Shimmer        0
MDVP:Shimmer(dB)    0
Shimmer:APQ3        0
Shimmer:APQ5        0
MDVP:APQ            0
Shimmer:DDA         0
NHR                 0
HNR                 0
status              0
RPDE                0
DFA                 0
spread1             0
spread2             0
D2                  0
PPE                 0
dtype: int64
```

```
# distribution of target Variable
parkinsons_data['status'].value_counts()
1    147
0     48
Name: status, dtype: int64
1  --> Parkinson's Positive

0 --> Healthy
```

Data Pre-Processing

```
X = parkinsons_data.drop(columns=['name','status'], axis=1)
Y = parkinsons_data['status']
print(X)
```

```
Output exceeds the size limit. Open the full output data in a text
editor
     MDVP:Fo(Hz)  MDVP:Fhi(Hz)  MDVP:Flo(Hz)   MDVP:Jitter(%)  \
0        119.992       157.302        74.997         0.00784   
1        122.400       148.650       113.819         0.00968   
2        116.682       131.111       111.555         0.01050   
3        116.676       137.871       111.366         0.00997   
4        116.014       141.781       110.655         0.01284   
..           ...           ...           ...             ...   
190      174.188       230.978        94.261         0.00459   
191      209.516       253.017        89.488         0.00564   
192      174.688       240.005        74.287         0.01360   
193      198.764       396.961        74.904         0.00740   
194      214.289       260.277        77.973         0.00567   

     MDVP:Jitter(Abs)      MDVP:RAP      MDVP:PPQ      Jitter:DDP
MDVP:Shimmer  \
0             0.00007   0.00370   0.00554     0.01109       0.04374   
1             0.00008   0.00465   0.00696     0.01394       0.06134   
2             0.00009   0.00544   0.00781     0.01633       0.05233   
3             0.00009   0.00502   0.00698     0.01505       0.05492   
4             0.00011   0.00655   0.00908     0.01966       0.06425   
..                ...       ...       ...         ...           ...   
190           0.00003   0.00263   0.00259     0.00790       0.04087   
191           0.00003   0.00331   0.00292     0.00994       0.02751   
192           0.00008   0.00624   0.00564     0.01873       0.02308   
193           0.00004   0.00370   0.00390     0.01109       0.02296   
194           0.00003   0.00295   0.00317     0.00885       0.01884   
...
193  0.643956 -6.744577  0.207454  2.138608  0.123306  
194  0.664357 -5.724056  0.190667  2.555477  0.148569  
```

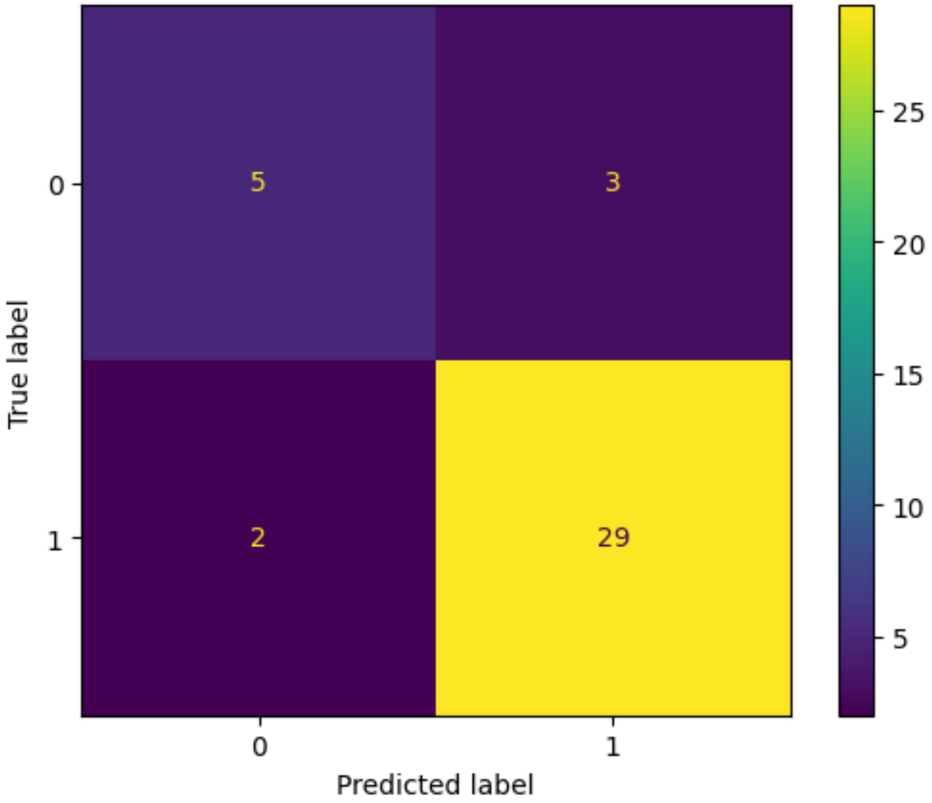
```
[195 rows x 22 columns]
```

```
print(Y)
0      1
1      1
2      1
3      1
4      1
      ..
190    0
191    0
192    0
193    0
194    0
Name: status, Length: 195, dtype: int64
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)
(195, 22) (156, 22) (39, 22)
```

- Support Vector Machine Model

  ```
  model = svm.SVC(kernel='linear')
  # training the SVM model with training data
  model.fit(X_train, Y_train)

  Model Evaluation
  Accuracy Score
  # accuracy score on training data
  X_train_prediction = model.predict(X_train)
  training_data_accuracy = accuracy_score(Y_train,
  X_train_prediction)
  ```

```
print('Accuracy score of training data : ', training_data_accuracy)
 Accuracy score of training data :  0.8846153846153846
```



Saving the model using pickle

```
import pickle
with open('model_svm.pkl','wb') as file:
    pickle.dump(model,file)
with open('scaler.pkl','wb') as file:
    pickle.dump(scaler,file)
```

Loading model using pickle

```
import pickle
with open('model_svm.pkl','rb') as file:
    model_svm = pickle.load(file)
```

Building a Predictive System

```
#input_data = (197.07600,206.89600,192.05500,0.00289,0.00001,0.00166,0.00168
,0.00498,0.01098,0.09700,0.00563,0.00680,0.00802,0.01689,0.003
39,26.77500,0.422229,0.741367,-
7.348300,0.177551,1.743867,0.085569)
input_data                                                    =
(119.992,157.302,74.997,0.00784,0.00007,0.0037,0.00554,0.01109
,0.04374,0.426,0.02182,0.0313,0.02971,0.06545,0.02211,21.033,0
.414783,0.815285,-4.813031,0.266482,2.301442,0.284654)
# changing input data to a 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)

# standardize the data
std_data = scaler.transform(input_data_reshaped)

prediction = model_svm.predict(std_data)
print(prediction)



if (prediction[0] == 0):
  print("The Person does not have Parkinsons Disease")

else:
  print("The Person has Parkinsons")
The Person has Parkinsons

# Trying logistic classifier
parkinsons_sys                                                =
LogisticRegression(random_state=0).fit(X_train, Y_train)
X_test_prediction = parkinsons_sys.predict(X_test)
test_data_accuracy = accuracy_score(Y_test, X_test_prediction)
```

```
   print (test_data_accuracy)
   0.8205128205128205
     cm= confusion_matrix(Y_test, X_test_prediction)
ConfusionMatrixDisplay.from_estimator(parkinsons_sys, X_test,
Y_test)
plt.show()
# Trying kmeans
kmeans = KMeans(n_clusters=4, random_state=0,
n_init="auto").fit(X_train, Y_train)
X_test_prediction = kmeans.predict(X_test)
test_data_accuracy = accuracy_score(Y_test, X_test_prediction)
print (test_data_accuracy)
# Logistics after PCA
from sklearn.decomposition import PCA

pca = PCA(.95)
X_PCA=pca.fit_transform(X_train)
X_test_PCA=pca.transform(X_test)
print(X_train.shape)
print(X_PCA.shape)
print(X_test_PCA.shape)
X_PCA
parkinsons_sys_pca =
LogisticRegression(random_state=0).fit(X_PCA, Y_train)
X_test_prediction = parkinsons_sys_pca.predict(X_test_PCA)
test_data_accuracy = accuracy_score(Y_test, X_test_prediction)
print (test_data_accuracy)
## Implementing Random Forest
classifier= RandomForestClassifier(n_estimators= 10,
criterion="log_loss")
classifier.fit(X_train, Y_train)
X_test_prediction = classifier.predict(X_test)
```

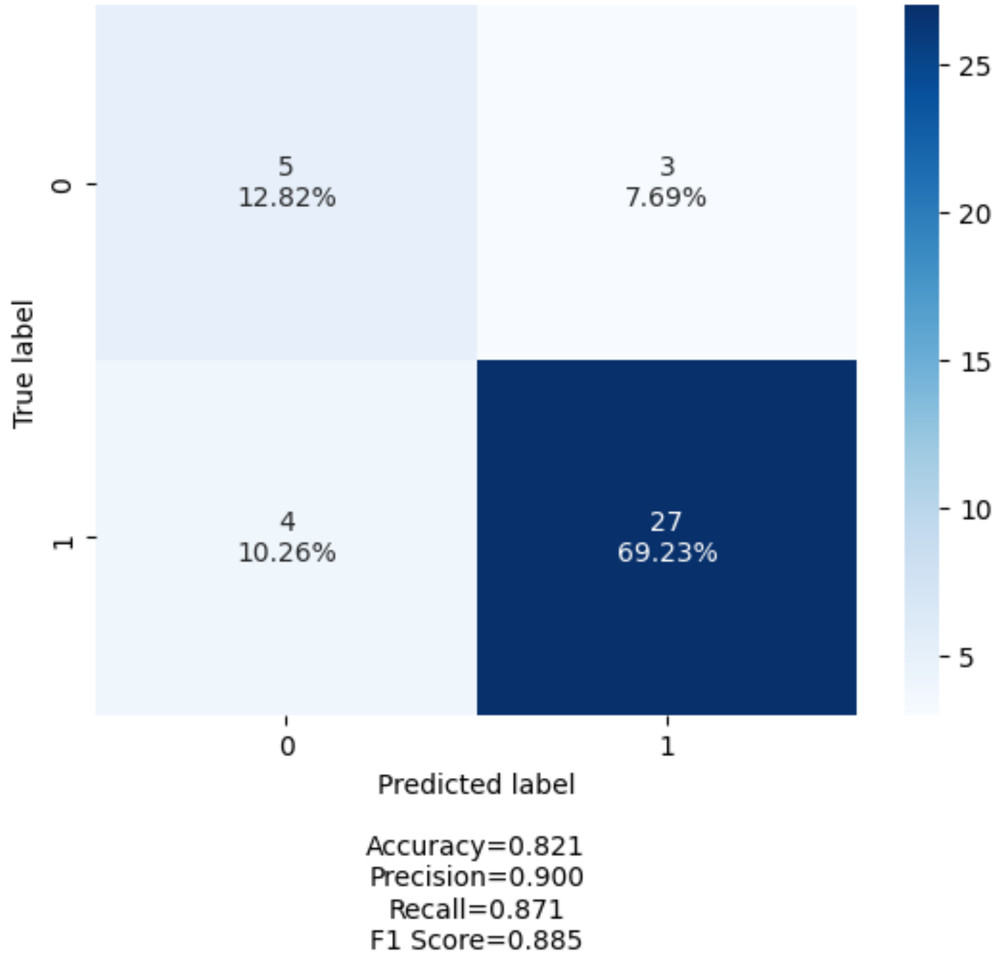
```
test_data_accuracy = accuracy_score(Y_test, X_test_prediction)
print (test_data_accuracy)
avg = 0
for i in range(10):
    classifier=RandomForestClassifier(n_estimators=
10,criterion="log_loss")
    classifier.fit(X_train, Y_train)
    X_test_prediction = classifier.predict(X_test)
    test_data_accuracy =accuracy_score(Y_test, X_test_prediction)
#     print (test_data_accuracy)
    avg+=test_data_accuracy
print(avg/10)
#Creating the Confusion matrix
from sklearn.metrics import confusion_matrix
from sklearn.metrics import ConfusionMatrixDisplay
cm= confusion_matrix(Y_test, X_test_prediction)
ConfusionMatrixDisplay.from_estimator(classifier, X_test, Y_test)
plt.show()
precision = cm[1][1] / (cm[1][1] + cm[0][1])
print(precision)
recall= cm[1][1] / (cm[1][1] + cm[1][0])
print(recall)
```

- **Using various algorithms to select the best one for the data set**

➢ **LOGISTIC REGRESSION**

```
lrc = LogisticRegression(random_state=0)
lrc.fit(X_train,Y_train)
with open("pickle_files/plain/lrc.model","wb") as file:
    pickle.dump(lrc,file)
Y_pred = lrc.predict(X_test)
accuracies["LRC"]=accuracy_score(Y_test,Y_pred)
precisions["LRC"]=precision_score(Y_test,Y_pred)
```

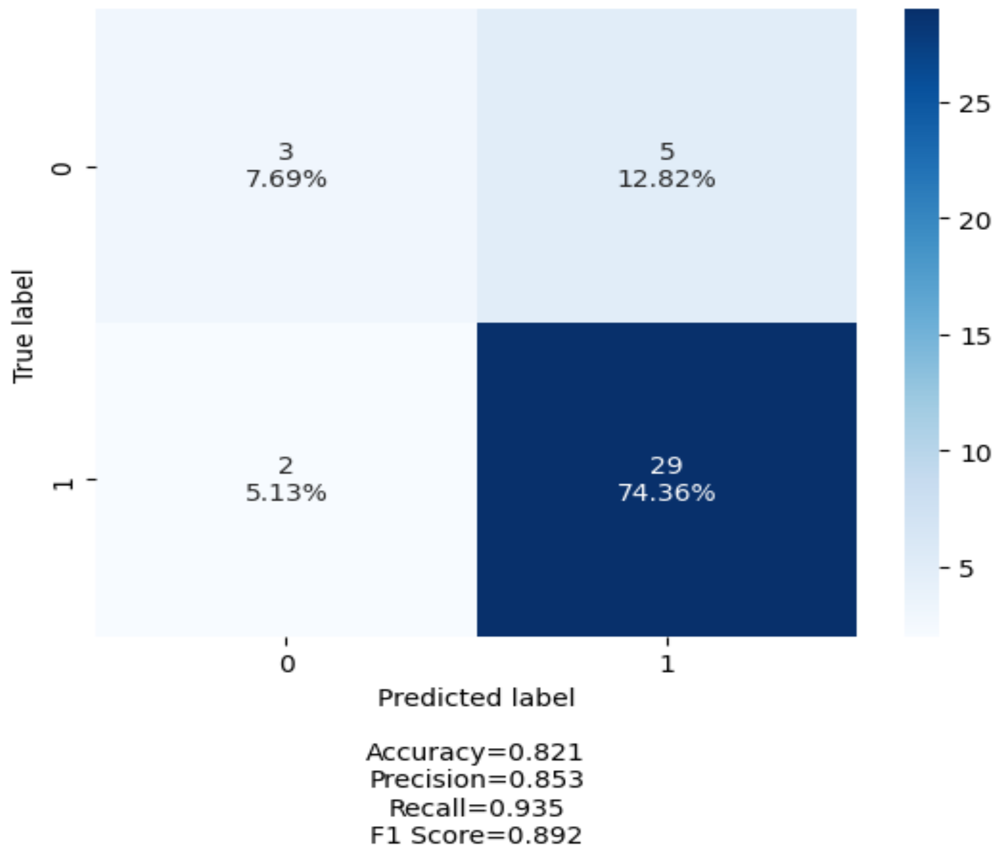
```
cf = confusion_matrix(Y_test,Y_pred)
make_confusion_matrix(cf)
```



```
Accuracy=0.821
Precision=0.900
Recall=0.871
F1 Score=0.885
```

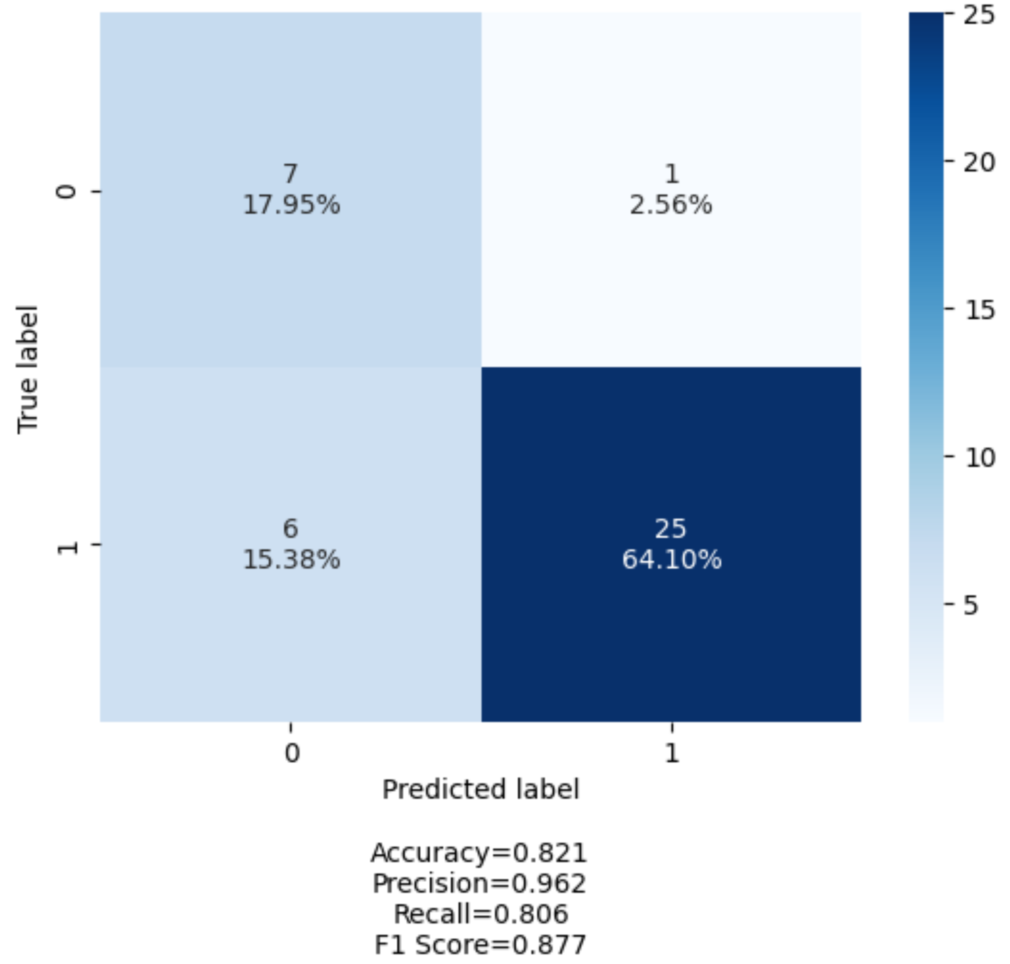
- **DECISION TREE**

```
dtc = DecisionTreeClassifier(max_leaf_nodes = 5, random_state = 0)
dtc.fit(X_train, Y_train)
with open("pickle_files/plain/dtc.model","wb") as file:
    pickle.dump(dtc,file)
Y_pred = dtc.predict(X_test)
accuracies["DTC"]=accuracy_score(Y_test,Y_pred)
precisions["DTC"]=precision_score(Y_test,Y_pred)
cf = confusion_matrix(Y_test,Y_pred)
make_confusion_matrix(cf)
```



- **RANDOM FOREST**

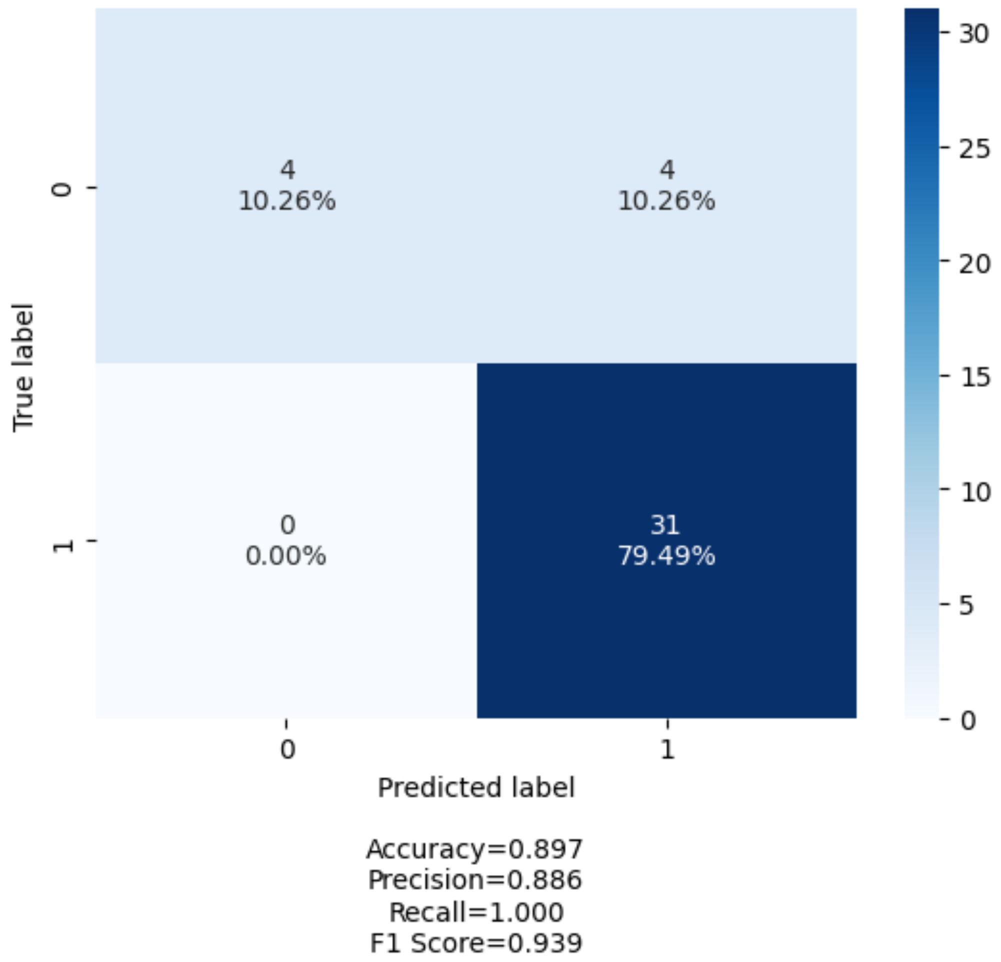
```
rfc = RandomForestClassifier(n_estimators= 10,
criterion="log_loss")
rfc.fit(X_train,Y_train)
with open("pickle_files/plain/rfc.model","wb") as file:
    pickle.dump(rfc,file)
Y_pred = rfc.predict(X_test)
accuracies["RFC"]=accuracy_score(Y_test,Y_pred)
precisions["RFC"]=precision_score(Y_test,Y_pred)
cf = confusion_matrix(Y_test,Y_pred)
make_confusion_matrix(cf)
```



Accuracy=0.821
Precision=0.962
Recall=0.806
F1 Score=0.877

➢ SUPPORT VECTOR MACHINE

```
svc = svm.SVC()
svc.fit(X_train,Y_train)
with open("pickle_files/plain/svm.model","wb") as file:
    pickle.dump(svc,file)
Y_pred = svc.predict(X_test)
accuracies["SVM"]=accuracy_score(Y_test,Y_pred)
precisions["SVM"]=precision_score(Y_test,Y_pred)
cf = confusion_matrix(Y_test,Y_pred)
make_confusion_matrix(cf)
```

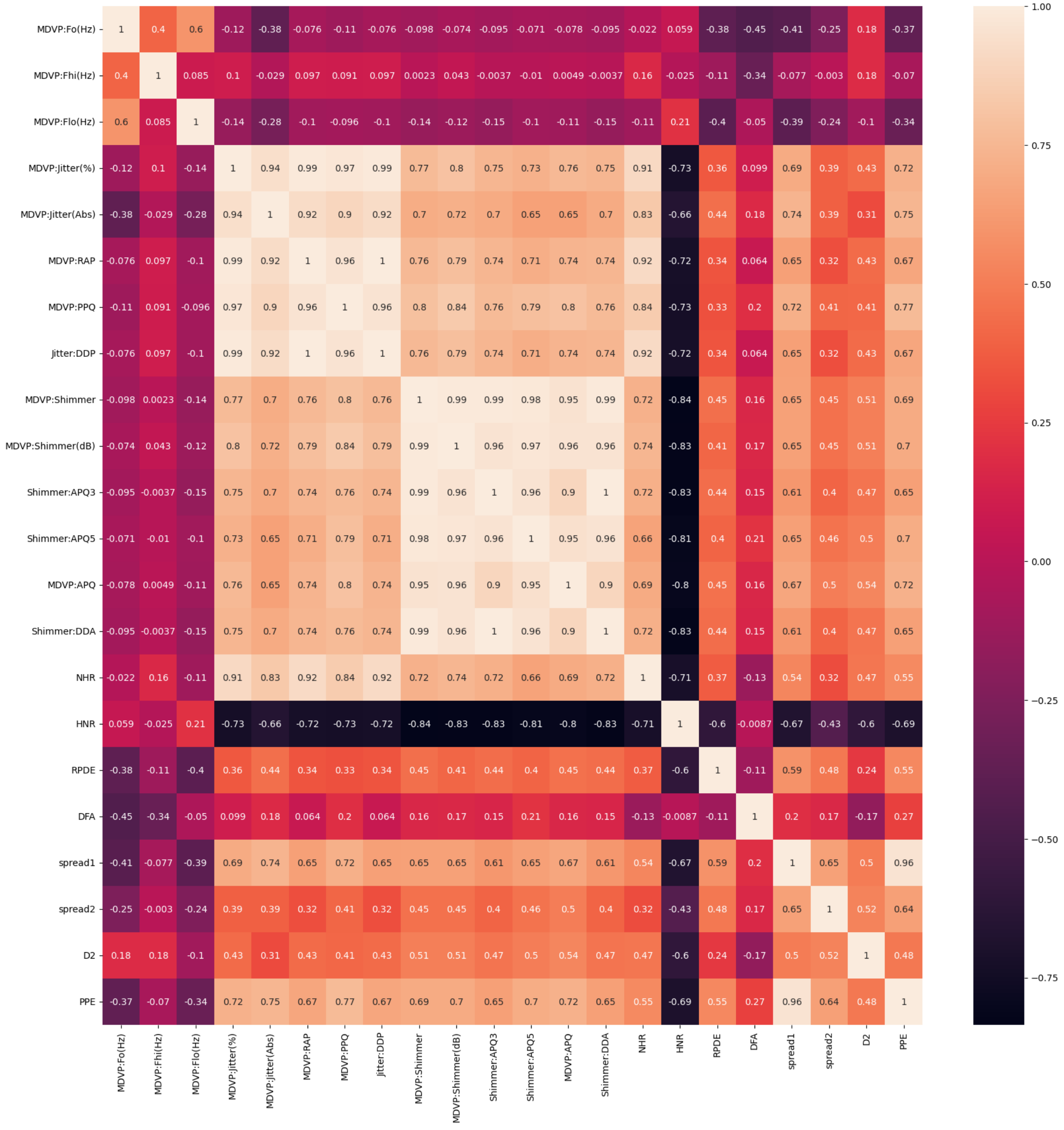


- **Analysisi**

```
df = pd.read_csv('parkinsons.csv')

print(f"Columns : {len(df.columns)}")

print(f"Rows : {len(df)}")

# print(df)

Columns : 24

Rows : 195

# print(df.dtypes)
```

```
df = df.drop(columns=['name','status'])

print(df.dtypes)

MDVP:Fo(Hz)         float64
MDVP:Fhi(Hz)        float64
MDVP:Flo(Hz)        float64
MDVP:Jitter(%)      float64
MDVP:Jitter(Abs)    float64
MDVP:RAP            float64
MDVP:PPQ            float64
Jitter:DDP          float64
MDVP:Shimmer        float64
MDVP:Shimmer(dB)    float64
Shimmer:APQ3        float64
Shimmer:APQ5        float64
MDVP:APQ            float64
Shimmer:DDA         float64
NHR                 float64
HNR                 float64
RPDE                float64
DFA                 float64
spread1             float64
spread2             float64
D2                  float64
PPE                 float64
dtype: object
```

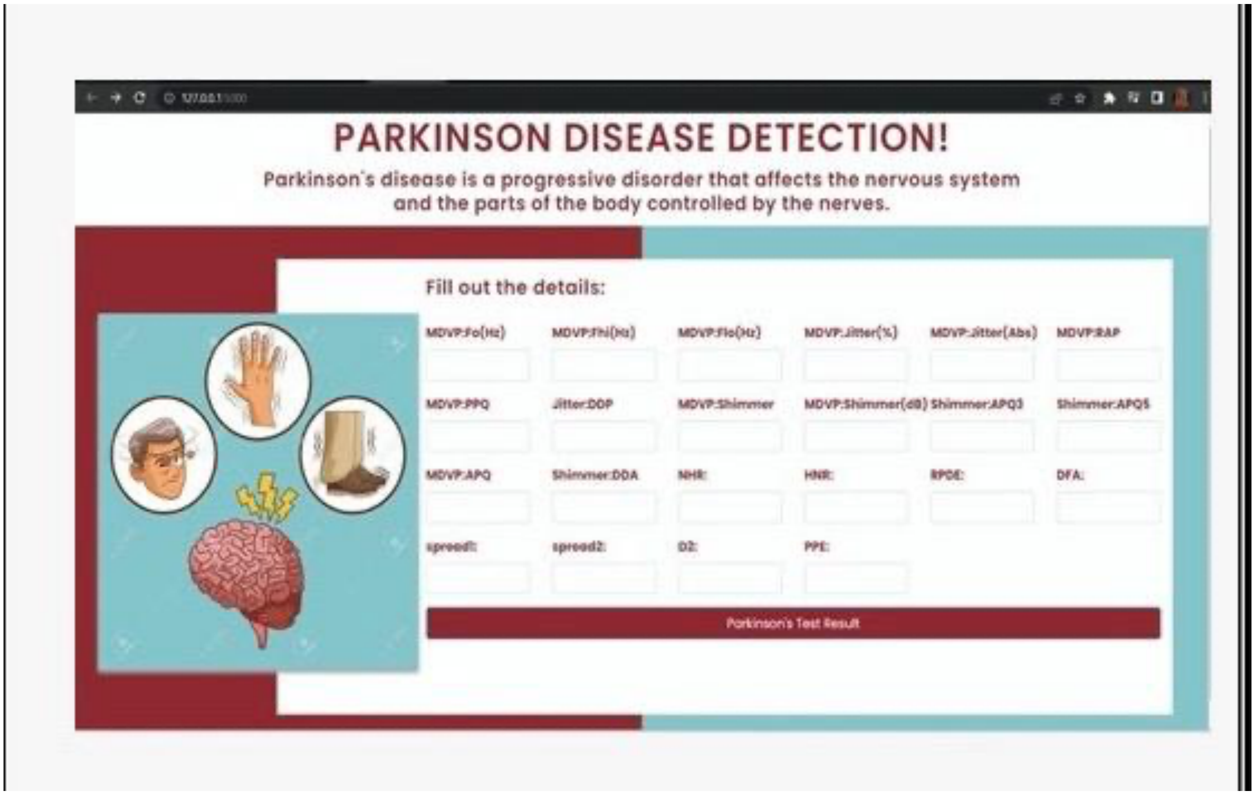
```
plt.figure(figsize=(20,20))

sns.heatmap(df.corr(),annot=True)

plt.show()
```



```
Splitting the data to training data & 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)
(195, 22) (156, 22) (39, 22)
```

5.2 OUTPUT

- **User Interface**



PARKINSON DISEASE DETECTION!

Parkinson's disease is a progressive disorder that affects the nervous system and the parts of the body controlled by the nerves.

Fill out the details:

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:

CHAPTER 6

6.1 CONCLUSION

Parkinson's is the second most neurodegenerative disease which has no cure. It results in difficulty of body movements, anxiety, breathing problems, loss of smell, depression, and speech. In this paper, the three different machine learning algorithms used to measure the performance are KNN, Naïve Bayes, and Logistic Regression applied on the dataset. The author chose the voice features of patients important features that are useful to evaluate the system. The author compared all the three machine learning methods accuracies and based on this one prediction model is generated. Hence, the aim is to use various evaluation metrics like confusion matrix, accuracy, precision, recall, and f1-score which predicts the disease efficiently. The ML algorithms were also compared and contrasted in light of the particular data. We were able to achieve desirable accuracy and predict the UPDRS scores in the expected way. The limitations of the current work would be that no matter how automated the process of Parkinson prediction becomes, there still will be a need for human intervention, intelligence and experience to make the diagnosis an accurate one. For future works, the dataset could be modelled on other more fitting Machine Learning models to improve accuracy of prediction

6.2 FUTURE SCOPE

Early detection of Polycystic Ovary Syndrome (PCOS) holds the potential to significantly enhance the quality of life for those affected. The future of PCOS detection systems shows promising prospects. Firstly, improved accuracy is expected through the application of advanced machine learning algorithms, reducing the occurrence of false positives and false negatives in diagnosis. Furthermore, the shift towards non-invasive diagnostic methods, like saliva or urine tests, can replace current invasive procedures, making PCOS detection more accessible and patient-friendly. Personalized medicine will play a pivotal role by tailoring treatment plans based on individual symptoms and genetic factors. With the increasing popularity of telemedicine, future PCOS detection systems could facilitate remote monitoring and integrate with wearable devices, allowing patients to track their symptoms and progress from the comfort of their homes. Early detection, especially targeting adolescence, can prevent delayed diagnosis, mitigating long-term health consequences. These advancements in technology and healthcare aim to fulfill the ultimate goal of improving the quality of life for individuals with PCOS, with early detection and intervention at its core.

CHAPTER 7

REFERENCES

[1] T. J. Wroge, Y. Özkanca, C. Demiroglu, D. Si, D. C. Atkins and R. H. Ghomi, "Parkinson’s Disease Diagnosis Using Machine Learning and Voice," 2018 IEEE Signal Processing in Medicine and Biology Symposium (SPMB), Philadelphia, PA, USA, 2018, pp. 1-7, doi: 10.1109/SPMB.2018.8615607.

[2] W. Wang, J. Lee, F. Harrou and Y. Sun, "Early Detection of Parkinson’s Disease Using Deep Learning and Machine Learning," in IEEE Access, vol. 8, pp. 147635-147646, 2020, doi: 10.1109/ACCESS.2020.3016062.

[3] S. Raval, R. Balar and V. Patel, "A Comparative Study of Early Detection of Parkinson's Disease using Machine Learning Techniques," 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), Tirunelveli, India, 2020, pp. 509-516, doi: 10.1109/ICOEI48184.2020.9142956.

[4] M. Sivakumar, A. H. Christinal and S. Jebasingh, "Parkinson’s disease Diagnosis using a Combined Deep Learning Approach," 2021 3rd International Conference on Signal Processing and Communication (ICPSC), Coimbatore, India, 2021, pp. 81-84, doi: 10.1109/ICSPC51351.2021.9451719.

[5] A. Ouhmida, A. Raihani, B. Cherradi and Y. Lamalem, "Parkinson's disease classification using machine learning algorithms: performance analysis and comparison," 2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), Meknes, Morocco, 2022, pp. 1-6, doi: 10.1109/IRASET52964.2022.9738264.

[6] S. Kamoji, D. Koshti, V. V. Dmello, A. A. Kudel and N. R. Vaz, "Prediction of Parkinson's Disease using Machine Learning and Deep Transfer Learning from different Feature Sets," 2021 6th International Conference on Communication and Electronics Systems (ICCES), Coimbatre, India, 2021, pp. 1715-1720, doi: 10.1109/ICCES51350.2021.9488944