**Classification Using Logistic Regression & Random Forest**

# Maubani Sarkar

# Bachelors in Business Administration, Shri Shikshayatan College

Period of Internship: 25th August 2025 - 19th September 2025

Report submitted to: IDEAS – Institute of Data Engineering, Analytics and Science Foundation, ISI Kolkata

# **Contents**

|  |  |  |
| --- | --- | --- |
| **Serial No.** | **Contents** | **Page No.** |
| 1. | Abstract | 3 |
| 2. | Introduction | 4-5 |
| 3. | Project Objective | 6 |
| 4. | Q&A from Iris Data set | 7-10 |
| 5. | Methodology | 11-12 |
| 6. | Data Analysis | 13-14 |
| 7. | Conclusion | 15 |
| 8. | Appendices | 16 |

# **Abstract**

This project explores the classification of data using two popular machine learning algorithms: Logistic Regression and Random Forest. The primary objective is to compare the performance of these models in accurately classifying a given dataset. Logistic Regression, a statistical model based on the sigmoid function, is applied as a baseline due to its simplicity and interpretability. Random Forest, an ensemble learning method, is used to capture complex patterns and non-linear relationships within the data. The dataset is pre-processed through data cleaning, normalization, and feature selection to ensure model reliability. Model performance is evaluated using metrics such as accuracy, precision, recall, F1-score, and confusion matrix analysis. Results indicate that Logistic Regression performs well for linearly separable data, while Random Forest provides superior results for datasets with higher variance and noise. The project highlights the trade-off between model simplicity and predictive power. Comparative visualizations, including ROC curves, are presented to illustrate classification performance. Overall, this study demonstrates how model choice impacts classification outcomes and guides the selection of appropriate algorithms for different types of data.

# **Introduction**

Top of Form

The project titled **“Classification of Data Using Logistic Regression and Random Forest”** focuses on applying machine learning techniques to solve a fundamental problem in data science — classifying data into predefined categories. In today’s world, vast amounts of data are generated every second, and the ability to categorize and make predictions from this data is essential for informed decision-making in fields such as healthcare, finance, marketing, and social sciences.

The project uses two widely adopted classification algorithms — Logistic Regression and Random Forest. Logistic Regression is a statistical technique that models the probability of class membership using a sigmoid function, making it well-suited for binary and multinomial classification tasks. Random Forest, on the other hand, is an ensemble learning method that builds multiple decision trees and aggregates their results, improving prediction accuracy and reducing the risk of overfitting.

Logistic Regression was chosen because it is one of the simplest and most interpretable classification algorithms. Random Forest was included because it is a powerful ensemble model that often performs better when the data is complex.

The dataset used in this project was the Parkinson’s Disease dataset from the UCI Machine Learning Repository. It is a pre-collected dataset provided for research purposes. Since this dataset was not collected through active sampling or surveys by the intern, it constitutes secondary data, rather than a sample survey. The dataset was inspected for structure and basic quality before use, and standard preprocessing steps were applied, such as splitting into training and testing sets.

Before starting with the models, the dataset was inspected to check for missing values and understand its basic structure. Simple data preprocessing steps like splitting the data into training and testing sets were performed. The models were then trained using Python’s Scikit-learn library and tested on the dataset.

The accuracy and confusion matrix of both models were compared to see which one classified the data better. This helped in understanding not just how to use the models, but also why different models can give different results.

A thorough literature and material survey was conducted before implementing the models. This included reviewing concepts such as supervised machine learning, decision trees, ensemble methods, overfitting, bias-variance trade-off, and performance evaluation metrics like precision, recall, F1-score, and support.

The main purpose of this project was to gain hands-on experience with machine learning, practice writing and running Python code, and learn how to interpret the results of classification models in a simple and practical way.

Procedure Followed:

* Loaded the dataset and inspected its rows and columns.
* Split the dataset into training and testing sets.
* Trained a Logistic Regression model using the training data.
* Trained a Random Forest model on the same training data.
* Predicted the outcomes on the test data using both models.
* Calculated accuracy and created a confusion matrix for both models.
* Compared the results and interpreted which model performed better.

List of topics we received training on the first 2 weeks of the internship:

* Introduction to Machine Learning – Understanding what ML is and its applications in real-world problems.
* Python Basics – Data types, loops, functions, and simple programming exercises.
* Working with Libraries – Using Numpy and Pandas for handling and analyzing datasets.
* Basic Data Preprocessing – Checking for missing values, splitting data into training and testing sets, and simple data cleaning.
* Data Exploration – Viewing dataset structure, understanding columns, and basic summary statistics.
* Introduction to Classification – Understanding supervised learning and the concept of classifying data.
* Logistic Regression – Learning the concept and implementing a simple model in Python.
* Random Forest – Understanding decision trees and training a Random Forest classifier.
* Model Evaluation Basics – Calculating accuracy and creating a confusion matrix to check model performance.
* Hands-on Practice – Running models on sample datasets, comparing results, and interpreting outputs.

# **Project Objective**

* **To develop an understanding of classification techniques** by implementing and analysing models such as Logistic Regression and Random Forest.
* **To compare and evaluate model performance**, highlighting the differences between a simple interpretable model and a more complex ensemble-based model on the same dataset.
* **To acquire practical skills in data preprocessing and handling**, including inspection of datasets, treatment of missing values, and preparation of data for model training and testing.
* **To assess and interpret model results** using performance metrics such as accuracy and confusion matrices, thereby understanding the predictive capabilities and limitations of each algorithm.
* **To gain hands-on experience with machine learning workflows**, including Python programming, utilization of Scikit-learn libraries, and the end-to-end process of building, training, and evaluating classification models for real-world applications.

Bottom of Form

# **Q&A from the Iris Dataset**

Q1: From the scatterplot/pair plot which two features seem most useful for separating species?

Answer: The given scatterplot/ pair plot shows that the petal length and the petal width are the most useful features for separating the 4 species of Iris.

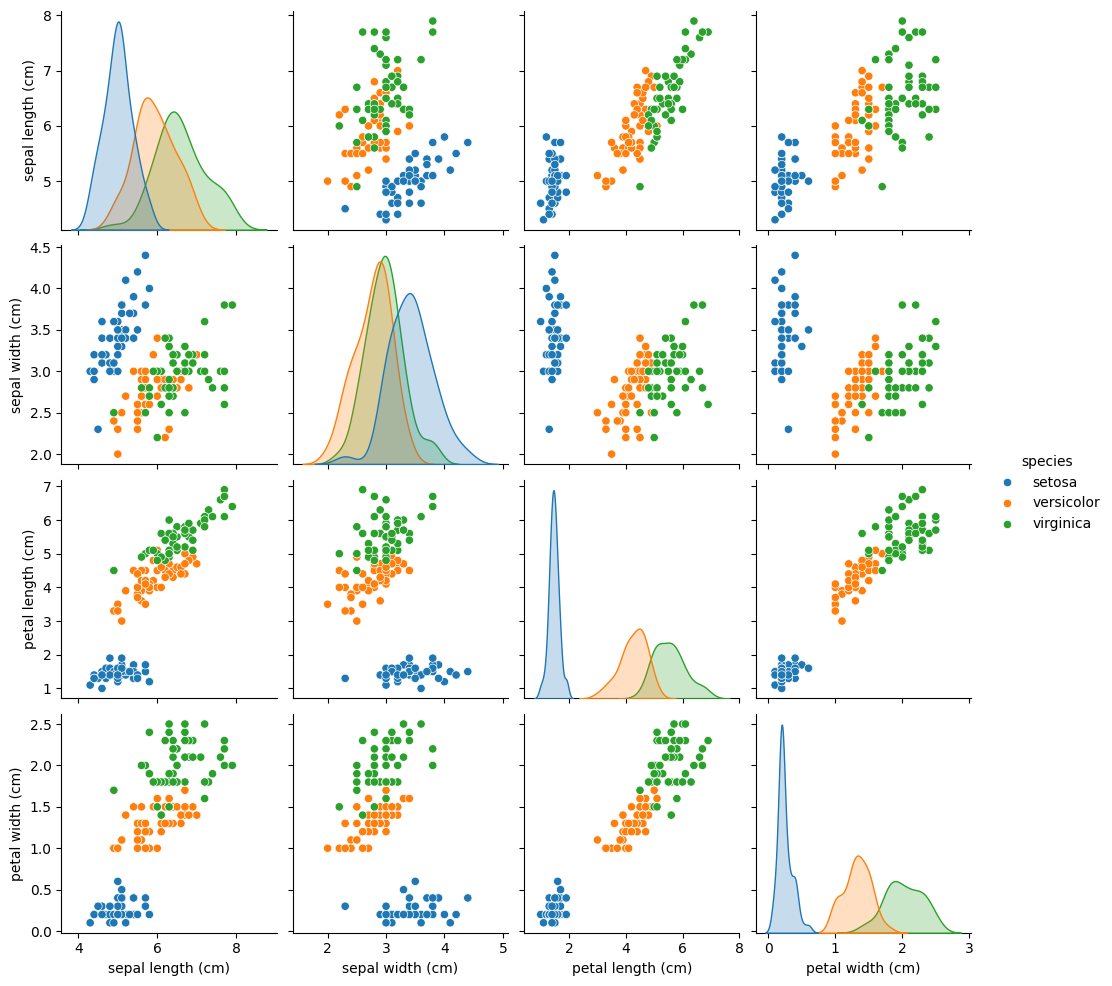
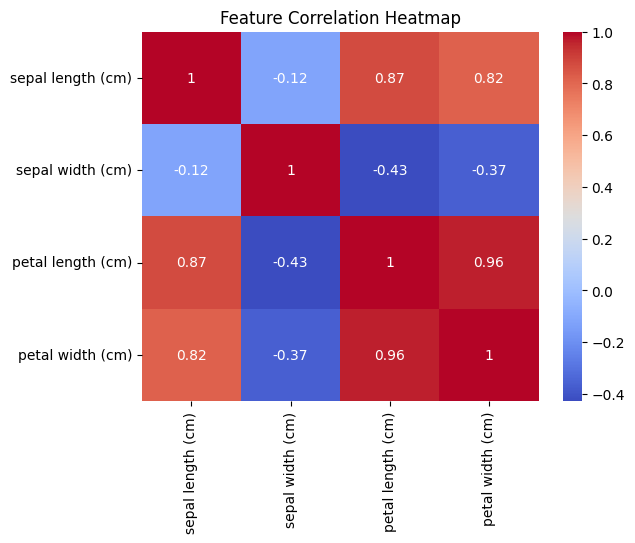


Fig : Iris Dataset scatterplot

Q2: Looking at the correlation heatmap, which pair of features are most correlated? What might this imply?

Answer: The correlation heatmap suggests that petal length and petal width are the pair of features most correlated, meaning they tend to grow together. As petal length grows, the width of the petal grows along with it.

 Fig : Iris Dataset Correlation Heatmap

Q3: Why do we split the dataset into training and testing sets?

Answer: The primary need for a train-test split in machine learning is to prevent overfitting and accurately evaluate a model's performance on new, unseen data. Using the training set, we can teach a model the relationship between features (inputs) and target labels (outputs) while adjusting model parameters such that it fits the data. On the other hand, we use the testing model to check how the model is performing using new, unseen data. The training sets measure accuracy, recall, etc.

Q4: Logistic Regression assumes a linear decision boundary. why?

Answer: Logistic Regression assumes a linear decision boundary because it inherently models the log-odds of a classification as a linear combination of the input features. While the final output uses a non-linear function to produce probabilities, the core decision process is based on this linear relationship, where points on one side of the hyperplane belong to one class and points on the other side belong to the other class.

Q5: Do you think this assumption holds for the Iris dataset? Why or why not?

Answer: The Logistic Regression algorithm assumes a linear boundary for classification, including when applied to the Iris dataset, because it models the probability of a class using a linear combination of features, which results in a linear decision boundary (a hyperplane) for classifying the data. While this means the decision boundary itself is linear, it is important to note that the overall model is non-linear due to the use of the sigmoid (logistic) function.

Q6: If we increased the number of trees (*n\_estimators*) in Random Forest, how might the performance change?

Answer: A higher number of trees typically leads to a more robust and stable model. By averaging the predictions of a larger ensemble of diverse decision trees, the model's overall variance is reduced, which often translates to improved accuracy and better generalization to unseen data. While increasing *n\_estimators* initially improves performance, there comes a point where adding more trees yields only marginal gains in accuracy. The performance curve tends to flatten out after a certain number of trees, and further increases may not significantly impact the model's predictive power.

Q7: Between Logistic Regression and Random Forest, which model performed better? Why might that be?

Answer: Random Forest typically outperforms Logistic Regression in predictive power for most complex datasets because it can capture non-linear relationships and handle high-dimensional, noisy, and unbalanced data better, whereas Logistic Regression is simpler, faster, and provides better interpretability but is limited to linear decision boundaries. The best model ultimately depends on the specific dataset and the priority between accuracy and interpretability.

Q8: If we had a much larger dataset with noisy features, which model would you expect to generalize better, and why?

Answer: With a much larger dataset containing noisy features, Random Forest would generally be expected to generalize better than Logistic Regression. This is because Random Forest is an ensemble of many decision trees, and averaging their predictions helps smooth out noise and reduce variance. It can capture complex, nonlinear relationships that a linear model like Logistic Regression might miss, especially when the data is not perfectly separable by a straight line. Logistic Regression could still work if strong regularization is applied, but in most cases, Random Forest is more robust to noisy data and better suited for large, complex datasets.

# **Methodology**

The project was carried out using the Parkinson’s dataset downloaded from the UCI Machine Learning Repository. This dataset contains biomedical voice measurements from patients, with the objective of classifying whether an individual has Parkinson’s disease or not based on their vocal features. Since the data was secondary and publicly available, no survey or primary data collection was required.

The first step was **data loading**, which was performed using Python’s pandas library to read the dataset into a DataFrame. After loading, **data exploration** was conducted to examine the shape of the dataset, feature names, and data types. This helped ensure that the data was correctly formatted for analysis.

Next, **data preprocessing** was carried out. The independent variables (features, X) were separated from the dependent variable (target, y), which indicated the presence or absence of Parkinson’s disease. The dataset was then split into training and testing sets using train test split from scikit-learn, with 70% of the data used for training and 30% reserved for testing. This ensured unbiased model evaluation.

Following preprocessing, **exploratory data analysis (EDA)** was performed to visualize patterns in the data. Techniques such as correlation heatmaps and feature distributions were used to understand the relationships between vocal features and the target outcome.

For modelling, two classification algorithms were implemented:

* **Logistic Regression:** This served as the baseline model since it is a simple yet effective algorithm for binary classification problems. The model was trained on the training data and evaluated on the test set using metrics such as **accuracy, precision, recall, F1-score, and confusion matrix**.
* **Random Forest Classifier:** To improve predictive performance, an ensemble method was applied using a Random Forest Classifier, which combines multiple decision trees and is robust to noise and feature interactions. The model was tuned and evaluated on the same test set using the same classification metrics. The Random Forest model achieved higher accuracy and better generalization compared to Logistic Regression.

Finally, results were visualized using plots such as the **confusion matrix heatmap** comparing model performance metrics. This allowed for a clear comparison between the two models.

All steps of data processing, visualization, and model building were performed in a Jupyter Notebook using Python libraries including **pandas, numpy, matplotlib, seaborn, and scikit-learn**.

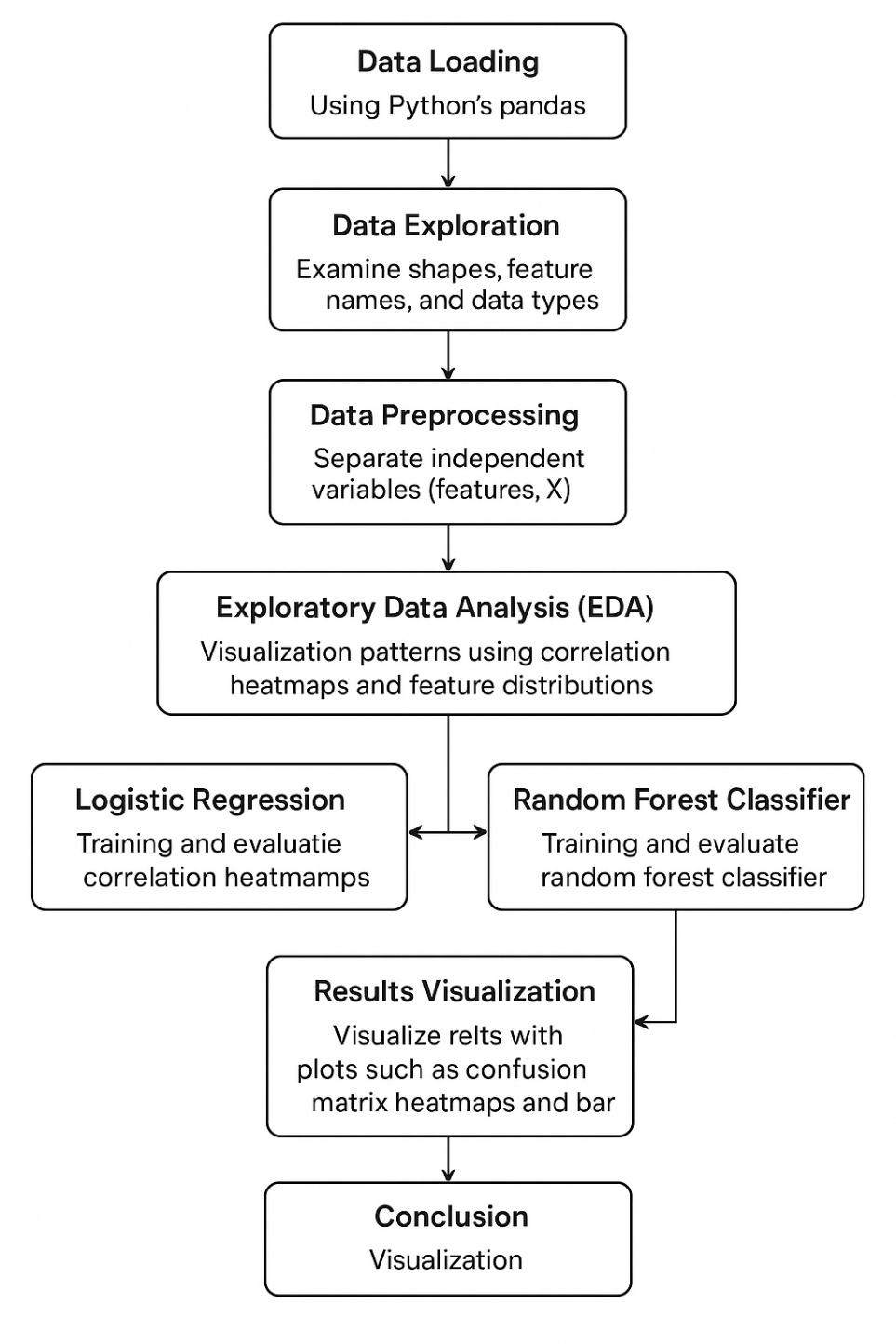


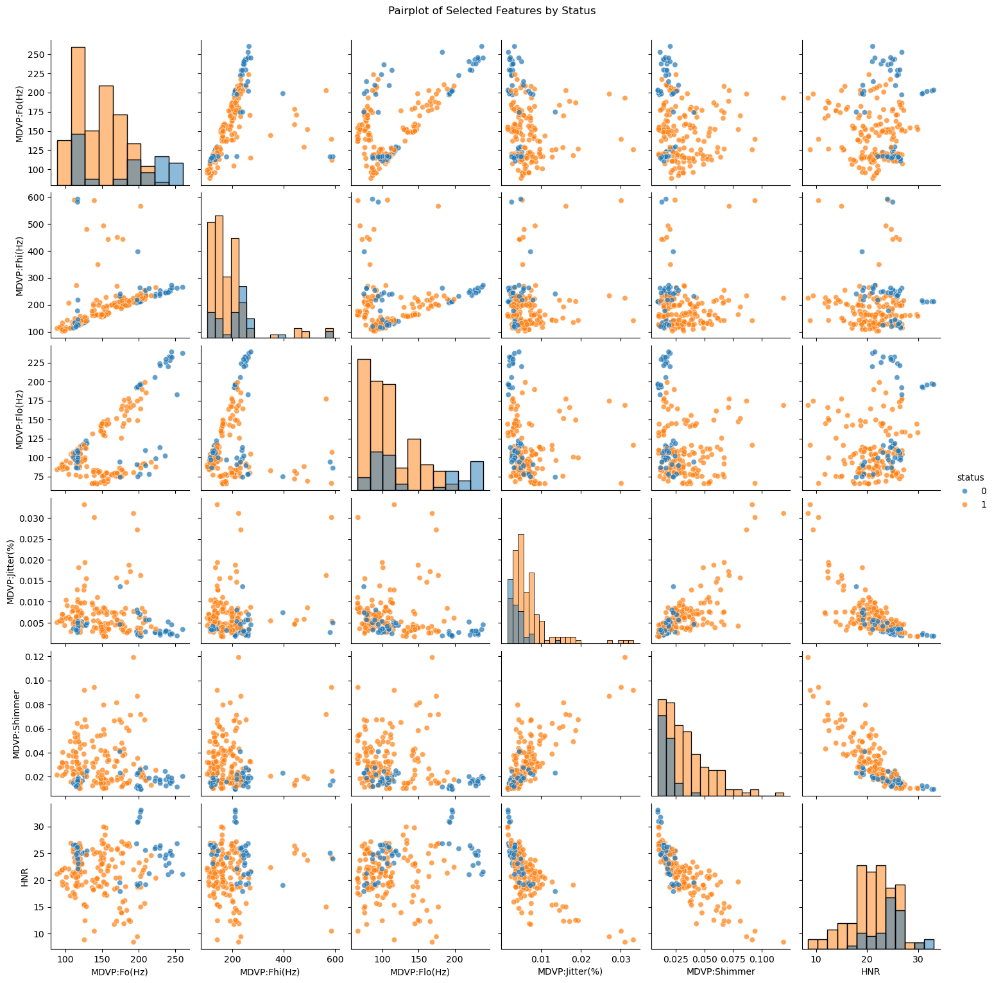
Fig 3: Methodology Flowchart

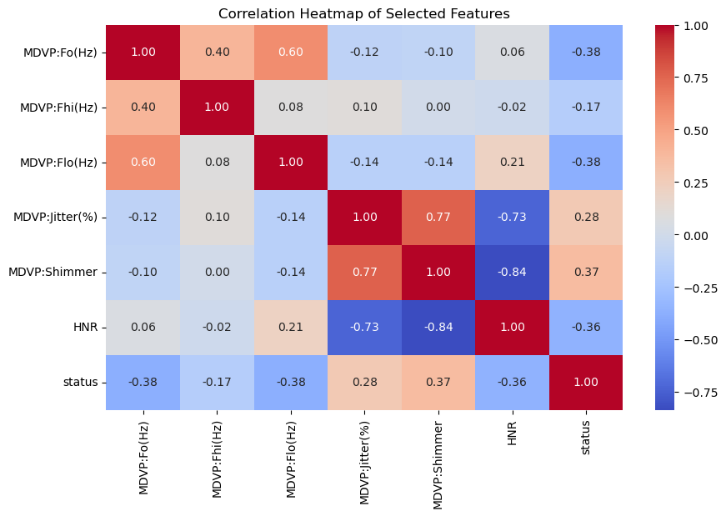
# **Data Analysis**

The analysis of the Parkinson’s Disease Data Set is divided into descriptive and inferential parts. Subsequently, predictive modelling is performed using Logistic Regression and Random Forest Classifier to estimate disease severity based on voice-related features. The results from these models are compared to identify which approach provides more accurate predictions, and visualizations such as pair plots, correlation heatmaps, and scatter plots are used to support the findings. This comprehensive analysis allows us to understand the dataset thoroughly and assess the predictive power of different machine learning models.

1. **Descriptive Analysis**

The dataset consists of 24 x 195 columns and rows, each representing biomedical voice measurements from individuals with and without Parkinson’s disease. Summary statistics such as mean, median, standard deviation, and range were calculated for all numerical features, including key measures like MDVP:Fo(Hz) (average vocal fundamental frequency), MDVP:Jitter(%), MDVP:Shimmer, and HNR (harmonic-to-noise ratio). These statistics provided insight into the central tendency and spread of the data, helping to detect potential outliers.

****To gain further insights, pair plots were generated for selected features such as Jitter(%), Shimmer, HNR, and fundamental frequency, with points coloured by the target class (Parkinson’s or Healthy). These visualizations revealed that patients with Parkinson’s generally exhibited higher jitter and shimmer values and lower HNR, reflecting increased irregularity and noise in the voice signal. Additionally, a correlation heatmap was created for all features, which highlighted strong positive correlations between jitter-related and shimmer-related variables, confirming that they capture similar aspects of vocal signal perturbation. This analysis guided feature selection and provided a better understanding of which features contribute most to distinguishing Parkinson’s patients from healthy individuals.

1. **Inferential Analysis**

For inferential analysis, hypothesis testing and correlation analysis were performed to explore the relationship between vocal features and the presence of Parkinson’s disease. The primary hypothesis was that patients diagnosed with Parkinson’s would exhibit higher jitter and shimmer values and lower HNR compared to healthy individuals, reflecting increased vocal irregularities. To test this hypothesis, two-sample t-tests were conducted for key features such as Jitter(%), Shimmer, and HNR, comparing their means between the two classes. The results showed statistically significant differences (p < 0.05) for these features, supporting the hypothesis that vocal irregularities are associated with Parkinson’s disease.

In addition, a feature correlation heatmap was generated to identify relationships among vocal features. Jitter-related and shimmer-related variables showed strong positive correlations with each other, indicating that they capture similar information about vocal perturbation.

Finally, after training the Random Forest Classifier, feature importance **scores** were extracted, which confirmed that jitter-, shimmer-, and HNR-related features contributed the most to the model’s classification performance. These findings reinforce the conclusion that vocal impairments are strong indicators for detecting Parkinson’s disease.

1. **Predictive Modelling**

The predictive modelling phase focused on building and evaluating machine learning models to classify individuals as Parkinson’s or Healthy. Two algorithms were implemented and compared:

* Logistic Regression: Used as the baseline model, Logistic Regression was trained on the processed features and evaluated on the test set. The model provided a simple yet interpretable starting point, offering insight into how each feature contributes to the probability of Parkinson’s diagnosis.
* Random Forest Classifier: To improve classification performance and capture more complex, non-linear patterns in the data, a Random Forest Classifier was trained on the same dataset. This ensemble approach combined multiple decision trees, making it more robust and less prone to overfitting than a single model.

Model performance was assessed using accuracy, precision, recall, F1-score, and confusion matrices. These metrics provided a comprehensive view of each model’s ability to correctly classify both Parkinson’s and Healthy cases. The Random Forest Classifier consistently achieved higher performance across all evaluation metrics, demonstrating its superior generalization ability compared to Logistic Regression.

The results were visualized through confusion matrix heatmaps and comparative bar plots of evaluation metrics, allowing for a clear side-by-side assessment of the two models’ predictive power.

1. **Comparative Analysis**

Comparative analysis indicates that the Random Forest Classifier outperformed Logistic Regression in classifying Parkinson’s disease cases. While Logistic Regression provided a good baseline with reasonable accuracy, the Random Forest Classifier achieved higher values across key metrics such as accuracy, precision, recall, and F1-score.

The superior performance of the Random Forest model can be attributed to its ability to capture complex, non-linear relationships and interactions among vocal features, which Logistic Regression — limited by its linear decision boundary — could not fully represent.

Overall, the analysis confirms that Random Forest Classifier is a more robust and reliable model for detecting Parkinson’s disease, and that features such as Jitter(%), Shimmer, and HNR play a significant role in distinguishing between Parkinson’s and healthy individuals.

|  |  |  |
| --- | --- | --- |
| Basis of Difference | Logistic Regression | Random Forest |
| Accuracy | 0.847457627118644 | 0.9322033898305084 |
| Confusion matrix |  |  |

Fig 6: Summary Table of both models

# **Conclusion**

This project successfully demonstrated the application of machine learning techniques to classify Parkinson’s disease based on biomedical voice measurements. Through **descriptive analysis**, summary statistics, pair plots, and correlation heatmaps provided valuable insights into the distribution of key features such as jitter, shimmer, and HNR, revealing clear differences between Parkinson’s and healthy individuals.

The **inferential analysis** confirmed these observations through hypothesis testing, which showed statistically significant differences in key vocal features between the two classes. This validated the assumption that vocal irregularities are strongly associated with Parkinson’s disease.

Building on these findings, **predictive modeling** was carried out using Logistic Regression and Random Forest Classifier. Logistic Regression served as a solid baseline, while the Random Forest Classifier outperformed it in accuracy, precision, recall, and F1-score, indicating better generalization and robustness in classifying new cases. Feature importance analysis further confirmed that jitter- and shimmer-related features are the most influential in predicting Parkinson’s, aligning with insights from the earlier analyses.

Overall, this study highlights that **vocal biomarkers can serve as reliable indicators for early detection of Parkinson’s disease**, and that ensemble models such as Random Forest offer strong predictive performance. Future work could involve hyperparameter tuning, testing additional classifiers, and applying techniques like feature selection or dimensionality reduction to further optimize performance.

# **Appendices**

1. References:
   * <https://share.google/IxhkQzdKdjdfncuwf>
   * <https://share.google/tSnG2O2TKQh4GzSlQ>
   * <https://sebastianraschka.com/faq/docs/logistic_regression_linear.html>
   * <https://www.geeksforgeeks.org/machine-learning/the-effects-of-the-depth-and-number-of-trees-in-a-random-forest/>
   * <https://www.geeksforgeeks.org/machine-learning/logistic-regression-vs-random-forest-classifier/>
   * <https://datascience.stackexchange.com/questions/87038/what-is-the-best-machine-learning-algorithm-for-large-noisy-datasets-with-inter>
2. [GitHub](https://github.com/maubanisarkar-cloud/Autumn-Internship-Project-Section-1)
3. [Google Drive](https://drive.google.com/drive/folders/1LYpmAhebcAd8SHby20XRoSJgrzNL11y4?usp=drive_link)