Multiple Disease Prediction Using ML

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology
In
Computer Science and Engineering

School of Engineering and Sciences

Submitted by

Bhavana Mallesh Gadde AP21110010551
Sai Dhanush Gavini AP21110011214
Vasavi Maddula AP21110010530
Nagasai Chowdary AP21110010558

Under the Guidance of

(Dr.Sowmyajyothi Biswas)

SRM University–AP

Neerukonda, Mangalagiri, Guntur

Andhra Pradesh

May, 2024

This project stands out for its innovative approach to disease diagnosis, utilizing a blend of machine learning algorithms. By integrating diverse algorithms like Support Vector Machine (SVM), Naive Bayes, and Random Forest, it explores various avenues for predicting common chronic illnesses based on patient symptoms and profiles. Furthermore, the project is divided into two distinct branches: one focusing on symptom-based prediction using SVM, Naive Bayes, and Random Forest classifiers, while the other relies on clinical measurements. For the latter, SVM is employed for Parkinson's disease diagnosis, while logistic regression is utilized for heart disease prediction. The incorporation of real-world patient data enhances the authenticity and practicality of the analysis, ensuring relevance to clinical settings. Moreover, the project's emphasis on evaluating the performance of different models, identifying their strengths and weaknesses, and providing actionable insights for healthcare professionals sets it apart from existing research in the field. In summary, the project's novelty lies in its comprehensive strategy to address the challenge of disease diagnosis through the integration of machine learning techniques across diverse domains.

Abstract

Disease diagnosis constitutes a pivotal aspect of healthcare, yet conventional methods often rely on manual examination and may not consistently deliver accurate results. In this project, we propose a novel machine learning-based disease diagnosis system to overcome the limitations of traditional diagnostic approaches. By harnessing the power of Support Vector Machine (SVM), Naive Bayes, and Random Forest algorithms, we endeavor to automate the process of identifying common chronic illnesses based on patient symptoms and profiles. Our dataset comprises symptom data collected from patients, alongside their corresponding diagnoses. Leveraging this dataset, we preprocess the data and train our machine learning models to predict the probability of specific diseases given a set of symptoms.

We introduce two distinct branches within our project: one focuses on symptom-based prediction utilizing SVM, Naive Bayes, and Random Forest classifiers, while the other employs clinical measurements. Specifically, SVM is employed for Parkinson's disease diagnosis, whereas logistic regression is utilized for heart disease prediction. The incorporation of real-world patient data enhances the authenticity and practicality of our analysis, ensuring relevance to clinical settings. Moreover, we evaluate the performance of each model using key metrics such as accuracy, precision, and recall.

Overall, our project contributes to the advancement of disease diagnosis by demonstrating the efficacy of machine learning techniques in automating the identification of common chronic illnesses. By providing a reliable and efficient diagnostic tool, our system has the potential to improve healthcare outcomes and enhance patient care.

Introduction:

In today's healthcare landscape, disease diagnosis is a critical aspect of patient care, with timely and accurate identification being essential for effective treatment and management. However, traditional diagnostic methods often rely on manual examination and may not always yield accurate results, leading to delays in treatment and potential misdiagnoses. To address these challenges, the integration of machine learning techniques in disease diagnosis has emerged as a promising approach to automate the process and improve diagnostic accuracy.

This project embarks on a pioneering journey, leveraging machine learning algorithms such as Support Vector Machine (SVM), Naive Bayes, and Random Forest, to develop a sophisticated disease diagnosis system adept at accurately discerning common chronic illnesses based on patient symptoms and clinical measurements. By harnessing the collective power of these algorithms, our aim is to automate and streamline the diagnostic workflow, thereby elevating healthcare outcomes and augmenting patient care.

The motivation behind this project stems from the need to address the shortcomings of traditional diagnostic methods and provide a more efficient and reliable approach to disease diagnosis. By exploring the capabilities of machine learning algorithms in this context, we seek to contribute to the advancement of healthcare and improve the quality of patient care.

Through a comprehensive exploration of machine learning techniques and their application in disease diagnosis, this project aims to provide valuable insights into the potential of automated diagnostic systems to revolutionize healthcare delivery. By developing a robust and accurate disease diagnosis system, we aim to empower healthcare professionals with the tools they need to make informed decisions and provide timely interventions for patients.

The highlights of this project are:

1.**Comprehensive Exploration:** The project encompasses a comprehensive exploration of machine learning algorithms for disease diagnosis, including SVM, Naive Bayes, and Random Forest, logistic regression providing a thorough analysis of their effectiveness in identifying common chronic illnesses.

2.Automation of Diagnosis: By leveraging machine learning techniques, we aim to automate the process of disease diagnosis, reducing the reliance on manual examination and improving the accuracy of diagnoses.

3.Improved Healthcare Outcomes: The development of a machine learning-based disease diagnosis system has the potential to significantly improve healthcare outcomes by enabling early and accurate identification of common chronic illnesses.

4.Symptom-based and Clinical Measurements Analysis: Our project addresses disease diagnosis through two distinct approaches. Firstly, we employ SVM, Naive Bayes, and Random Forest classifiers for symptom-based analysis, utilizing patient-reported symptoms to predict disease likelihood. Secondly, we utilize Support Vector Machine for diagnosing Parkinson's disease and logistic regression for heart disease based on clinical measurements, ensuring a comprehensive assessment of patient health.

Background:

Disease diagnosis is a critical aspect of healthcare, with accurate and timely identification of illnesses playing a pivotal role in patient outcomes. However, traditional diagnostic methods often rely on manual assessments and symptom-based evaluations, which can be time-consuming and prone to errors. Additionally, the complexity and variability of diseases pose challenges for healthcare professionals in accurately diagnosing patients.

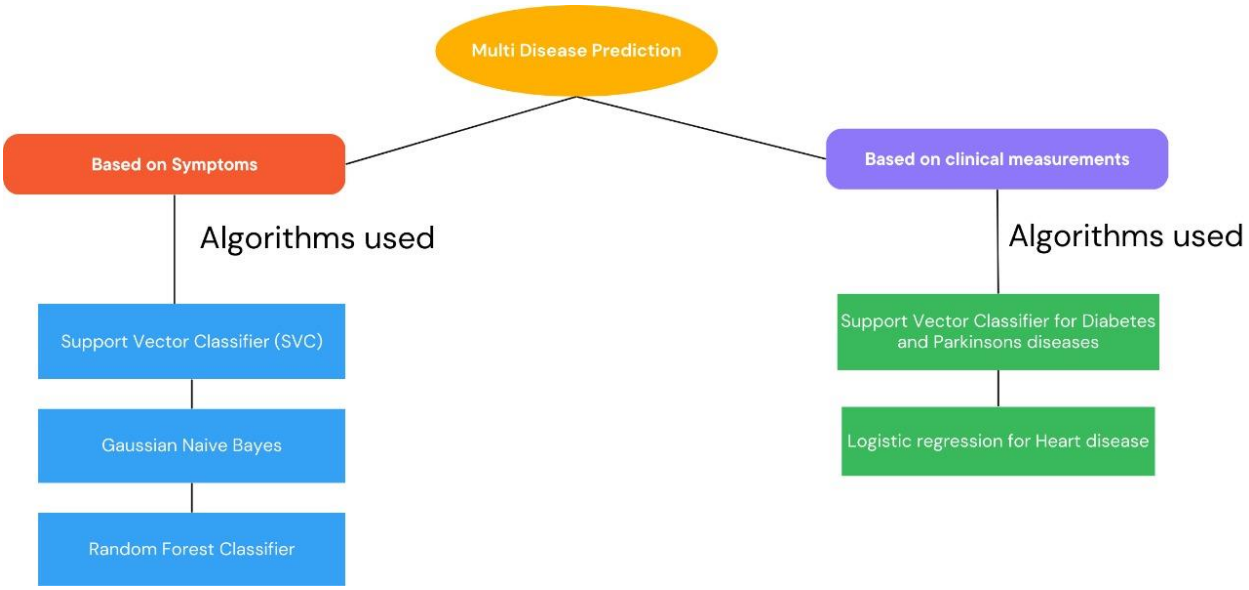
In recent years, the advent of machine learning has revolutionized disease diagnosis by leveraging advanced algorithms to analyze patient data and identify patterns indicative of various illnesses. By harnessing large datasets containing clinical measurements, symptoms, and patient profiles, machine learning models can assist healthcare providers in making more accurate and efficient diagnoses.

The goal of this project is to explore the application of machine learning techniques in disease diagnosis and evaluate their effectiveness in predicting and identifying common illnesses. Specifically, we aim to analyze the performance of machine learning models, such as Support Vector Machine (SVM), Naive Bayes, and Random Forest, logistic regression in diagnosing diseases based on patient symptoms and clinical measurements.

Methodology:

1.**Data Collection:** Gather a dataset containing information about symptoms and corresponding diseases. This dataset can be sourced from reputable medical sources or collected through surveys and patient records.
2.**Data Preprocessing**: Clean the dataset by handling missing values, removing duplicates, and standardizing data formats. This step may also involve encoding categorical variables and normalizing numerical features.
3.**Model Selection:** Choose appropriate machine learning algorithms for disease prediction. Consider algorithms like Support Vector Machine (SVM), Naive Bayes, and Random Forest, logistic regression ,which are commonly used for classification tasks.
4.**Model Training:** Train the selected models using the preprocessed dataset. Split the data into training and validation sets to evaluate model performance during training.

5.**Model Evaluation:** Assess the performance of trained models using evaluation metrics such as accuracy, precision, recall, and F1-score. Use techniques like cross-validation to ensure robustness and generalization of the models.
6.**Deployment:** We've developed a website for disease prediction using Flask for the backend. We deployed this website on Render. It takes clinical measurements as input from the user and provides the result indicating whether the person has the disease or not.



Random forest algorithm:

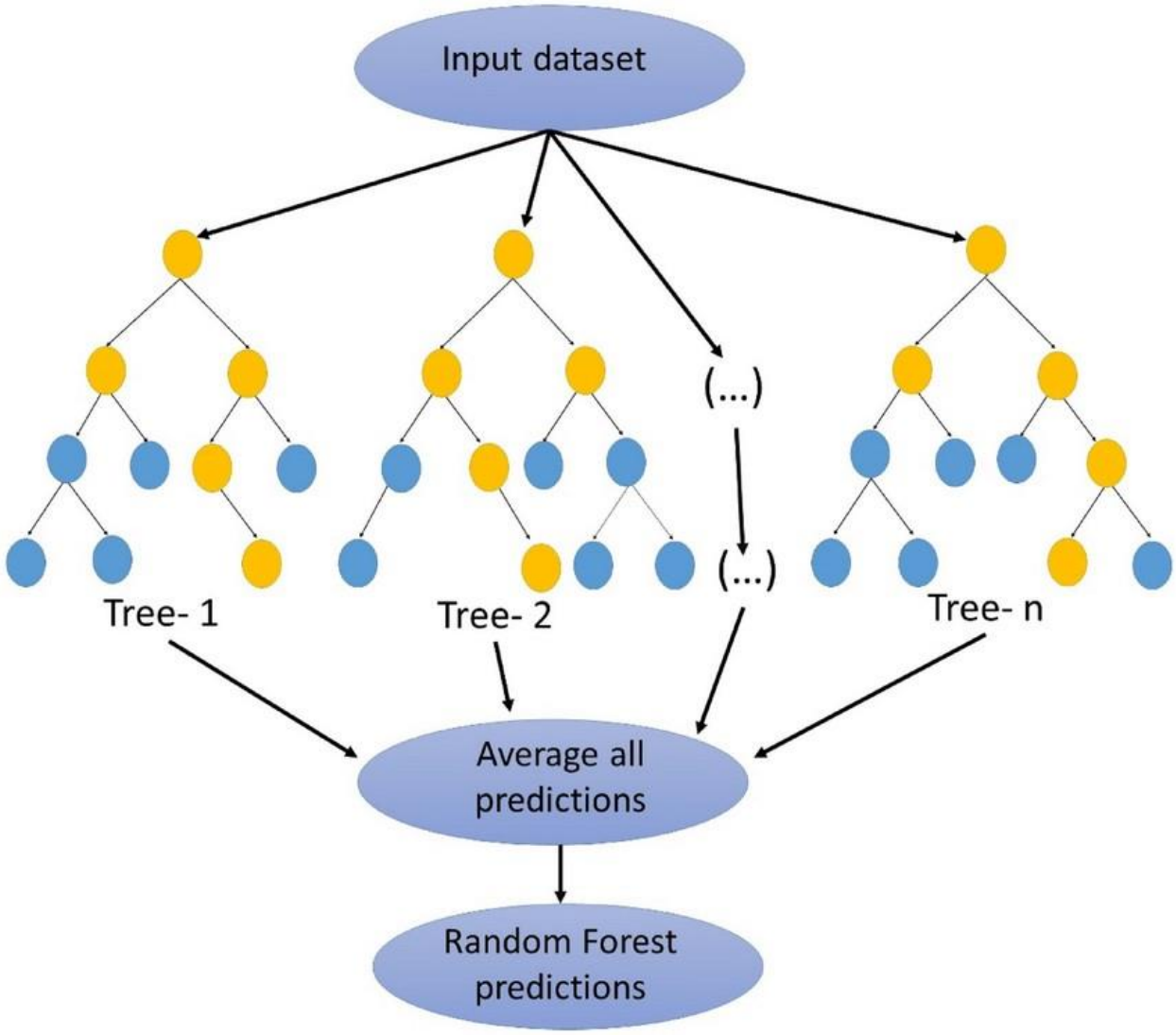
In this project, Random Forest serves as a potent tool to enhance the accuracy and reliability of disease prediction. As a supervised learning algorithm, Random Forest harnesses the collective power of multiple decision trees, leveraging their combined strength to make precise predictions. The process begins with the partitioning of the dataset through random sampling of both rows and features, where subsets of data are randomly selected and allocated to individual decision trees. This random sampling, conducted with replacement, ensures diversity among the trees, guarding against the influence of specific patterns or outliers in the data.

Each decision tree undergoes independent training on its subset of data to classify patients into different disease categories based on their symptoms and other relevant features. By considering various subsets of characteristics and data points, each tree uncovers unique patterns and correlations within the dataset. Following training, a majority voting mechanism enables the ensemble of decision trees to collectively contribute to the prediction process. This approach enhances the overall accuracy of the model and reduces the likelihood of individual errors by aggregating predictions from multiple trees.

One of the notable advantages of Random Forest is its adeptness at handling high-dimensional data and intricate relationships effectively. By amalgamating the predictions of numerous

decision trees, Random Forest can capture diverse patterns and anomalies present in the dataset, facilitating accurate disease predictions. Moreover, the ensemble nature of Random Forest mitigates the risk of overfitting, ensuring robust generalization to unseen data.

In summary, Random Forest emerges as a robust and versatile tool for disease diagnosis, offering improved accuracy and reliability compared to individual decision trees or alternative machine learning algorithms. Its capacity to harness the collective intelligence of multiple trees makes it well-suited for identifying diseases accurately amidst the complexity of real-world medical data.



Naive Bayes:

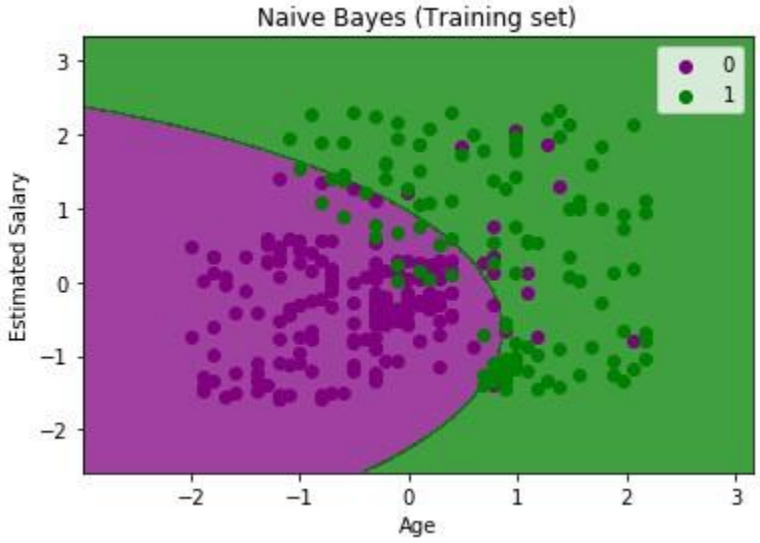
The Naive Bayes algorithm is a simple yet effective probabilistic classifier utilized in this project for disease diagnosis. Despite its simplistic assumptions, Naive Bayes often performs remarkably well in practice, particularly in situations with limited training data and high-dimensional feature spaces.

In disease diagnosis, Naive Bayes operates by applying Bayes' theorem to calculate the probability that a patient has a particular disease given their observed symptoms. It assumes that all features (symptoms) are conditionally independent given the class label (disease), which allows for efficient computation of the posterior probability using the joint probability distribution of the features.

Despite its "naive" assumption of feature independence, Naive Bayes can still produce reliable predictions, especially when applied to datasets with categorical or binary features. It is particularly well-suited for tasks involving text classification or medical diagnosis, where the presence or absence of specific symptoms contributes to the likelihood of a particular disease.

Naive Bayes offers several advantages, including its simplicity, scalability, and fast training speed. It requires minimal tuning of hyperparameters and can handle missing data gracefully. Moreover, Naive Bayes performs well even with small training datasets and is robust to irrelevant features.

In summary, the Naive Bayes algorithm utilizes Bayes' theorem and the assumption of feature independence to probabilistically predict the likelihood of various classes (in this case, diseases) based on the observed features (such as symptoms).



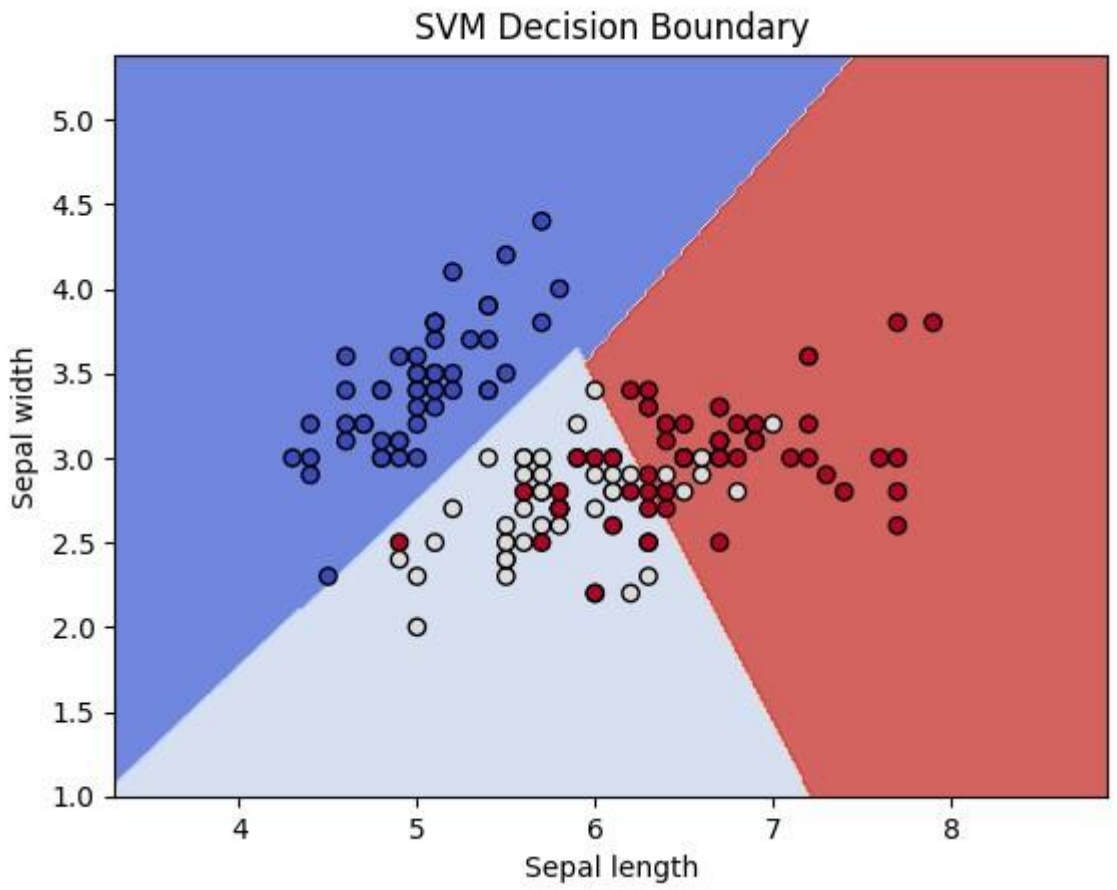
Support Vector Machine

The Support Vector Machine (SVM) algorithm is a powerful supervised learning technique employed in this project for disease diagnosis. SVM is particularly well-suited for binary classification tasks, where it aims to find the optimal hyperplane that separates data points belonging to different classes with the maximum margin.

In the context of disease diagnosis, SVM works by mapping patient symptoms to a high-dimensional feature space and identifying the optimal boundary (hyperplane) that best separates patients with different diseases. By maximizing the margin between classes, SVM can effectively classify patients into distinct disease categories based on their symptom profiles.

One of the key strengths of SVM is its ability to handle non-linear relationships between symptoms and diseases through the use of kernel functions, such as radial basis function (RBF) or polynomial kernels. These kernels enable SVM to capture complex patterns in the data and achieve high classification accuracy.

Moreover, SVM exhibits robustness to overfitting and performs well in scenarios with high-dimensional feature spaces, making it suitable for disease diagnosis tasks involving a large number of symptoms and patients.



Logistic regression

Logistic Regression, a supervised learning algorithm, plays a vital role in our project's methodology by offering a contrasting approach to disease diagnosis. Unlike other algorithms such as Support Vector Machine (SVM) and Random Forest, Logistic Regression relies on labeled data and is proficient in modeling the probability of a binary outcome.

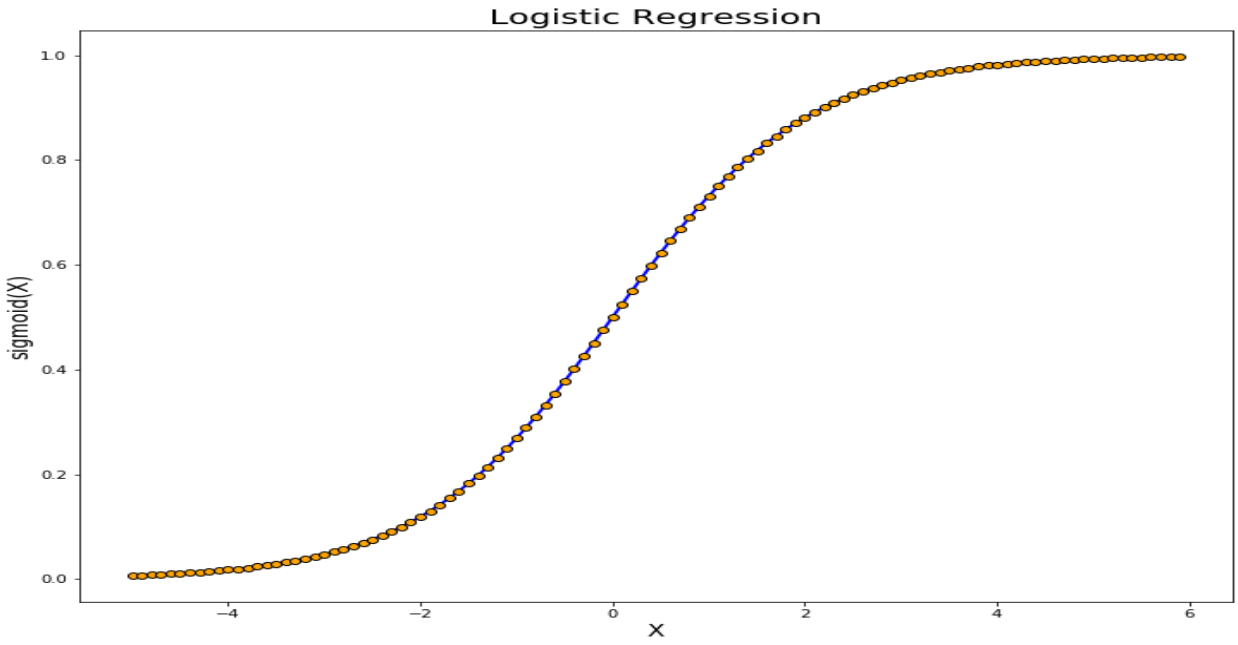
In our research, Logistic Regression is applied to predict, based on available patient data, the probability of a particular disease occurrence. The algorithm works by fitting the input data to a logistic function, which correlates the input features to the likelihood of disease presence.

Several key steps are involved in the Logistic Regression approach. Firstly, the dataset is split into training and testing sets using the train-test split technique to ensure the model's generalizability. Subsequently, the training data is utilized to train the Logistic Regression model, enabling it to predict the likelihood of disease occurrence based on patient profiles.

During the training phase, the logistic loss function is minimized by iteratively optimizing the

model's parameters, typically using techniques like gradient descent. Following training, the model's accuracy in classifying disease occurrences is evaluated using the testing data.

Evaluation metrics such as accuracy, precision, recall, and F1-score are commonly utilized to assess the Logistic Regression model's effectiveness. These metrics provide valuable insights into the model's predictive power and its ability to accurately identify disease occurrences while minimizing false diagnoses.



Predicted Diseases:

1.(vertigo) Paroymsal Positional Vertigo
2.AIDS
3.Acne
4.Alcoholic hepatiti
5.Allerg
6.Arthritis
7.Bronchial Asthma
8.Cervical spondylosis
9. Chicken pox
10.Chronic cholestasis
11.Common Cold
12.Dengue
13.Diabetes
14.Dimorphic hemmorhoids(piles)
15.Drug Reaction
16.Fungal infection
17.GERD
18.Gastroenteritis
19.Heart attack
20.Hepatitis B
21.Hepatitis C
22.Hepatitis D
23.Hepatitis E
24.Hypertension
25.Hyperthyroidism
26.Hypoglycemia
27.Hypothyroidism
28.Impetigo
29.Jaundice
30.Malaria
31.Migraine
32.Osteoarthristis
33.Paralysis (brain hemorrhage)
34.Peptic ulcer diseae
35.PneumoniaPsoriasis
36.Tuberculosis
37.Typhoid
38.Urinary tract infection
39.Varicose Veins
40. veinshepatitis A
41. Parkinson’s disease

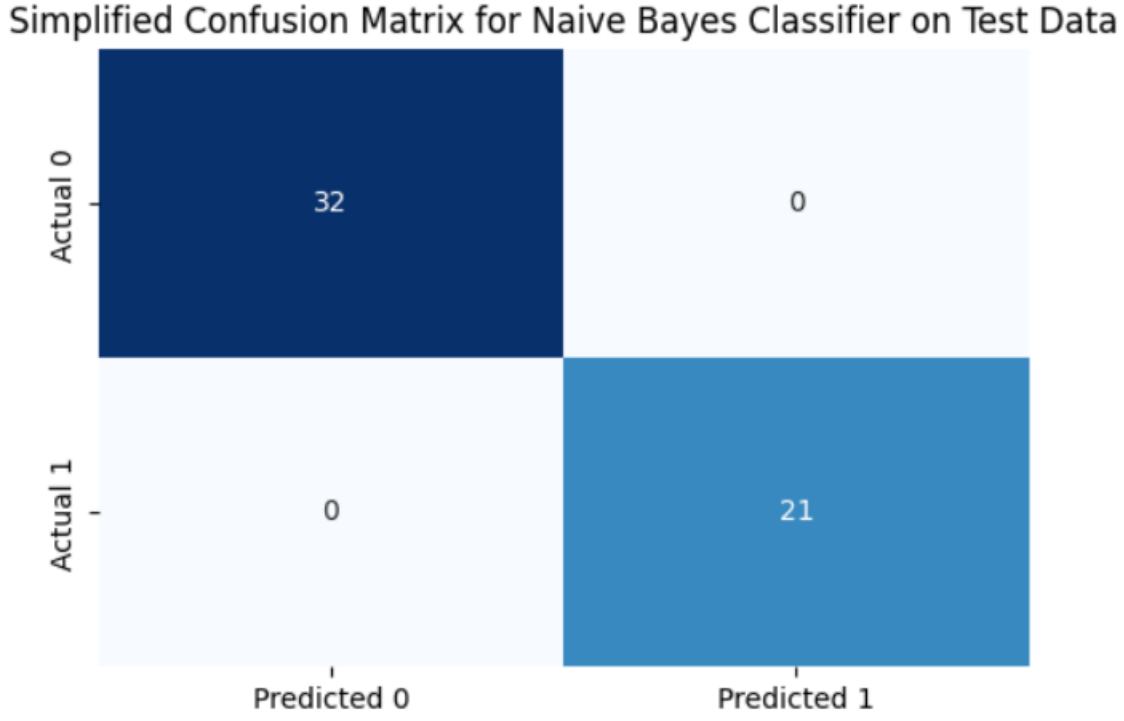
Disease Prediction Based On Symptoms

Code and Result:

Naive Bayes Algorithm:

```
# Naive Bayes Classifier
print("Naive Bayes Classifier")
nbModel = GaussianNB()
nbModel.fit(x_train, y_train)
nbModelPredict = nbModel.predict(x_test)

accuracy = accuracy_score(y_test, nbModelPredict) * 100
print(f"Accuracy on test data: {accuracy:.2f}%")

print("Classification Report :")
print(classification_report(y_test, nbModelPredict, labels=[0, 1]))

# Calculating confusion matrix
cf_matrix = confusion_matrix(y_test, nbModelPredict)

# Extracting TP, TN, FP, FN
TP = cf_matrix[1, 1]
TN = cf_matrix[0, 0]
FP = cf_matrix[0, 1]
FN = cf_matrix[1, 0]


# Plotting simplified confusion matrix
plt.figure(figsize=(6, 4))
sns.heatmap([[TN, FP], [FN, TP]], annot=True, fmt='d', cmap='Blues', cbar=False,
            xticklabels=['Predicted 0', 'Predicted 1'],
            yticklabels=['Actual 0', 'Actual 1'])
plt.title("Simplified Confusion Matrix for Naive Bayes Classifier on Test Data")
plt.show()
```

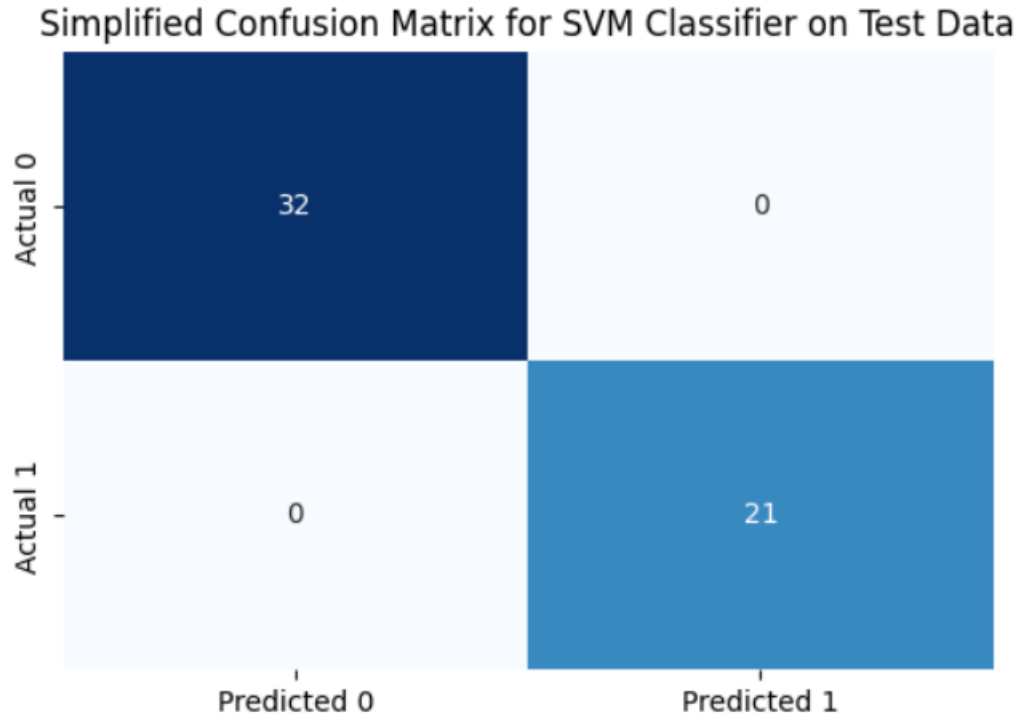
```
Naive Bayes Classifier
Accuracy on test data: 100.00%
Classification Report :
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        32
           1       1.00      1.00      1.00        21

   micro avg       1.00      1.00      1.00        53
   macro avg       1.00      1.00      1.00        53
weighted avg       1.00      1.00      1.00        53
```



Support Vector Machine

```
# SVM Classifier
svmModel = SVC()
svmModel.fit(x_train, y_train)
preds = svmModel.predict(x_test)

accuracy = accuracy_score(y_test, preds) * 100
print("Support Vector Machine (SVM) Classifier")
print(f"Accuracy on test data: {accuracy:.2f}%")

from sklearn.metrics import classification_report, confusion_matrix
print("Classification Report for Classes :")
print(classification_report(y_test, preds, labels=[0, 1]))

# Calculating confusion matrix
cf_matrix = confusion_matrix(y_test, preds)

# Extracting TP, TN, FP, FN
TP = cf_matrix[1, 1]
TN = cf_matrix[0, 0]
FP = cf_matrix[0, 1]
FN = cf_matrix[1, 0]



# Plotting simplified confusion matrix
plt.figure(figsize=(6, 4))
sns.heatmap([[TN, FP], [FN, TP]], annot=True, fmt='d', cmap='Blues', cbar=False,
            xticklabels=['Predicted 0', 'Predicted 1'],
            yticklabels=['Actual 0', 'Actual 1'])
plt.title("Simplified Confusion Matrix for SVM Classifier on Test Data")
plt.show()
```

```
Support Vector Machine (SVM) Classifier
Accuracy on test data: 100.00%
Classification Report for Classes 0 and 1:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        32
           1       1.00      1.00      1.00        21

   micro avg       1.00      1.00      1.00        53
   macro avg       1.00      1.00      1.00        53
weighted avg       1.00      1.00      1.00        53
```

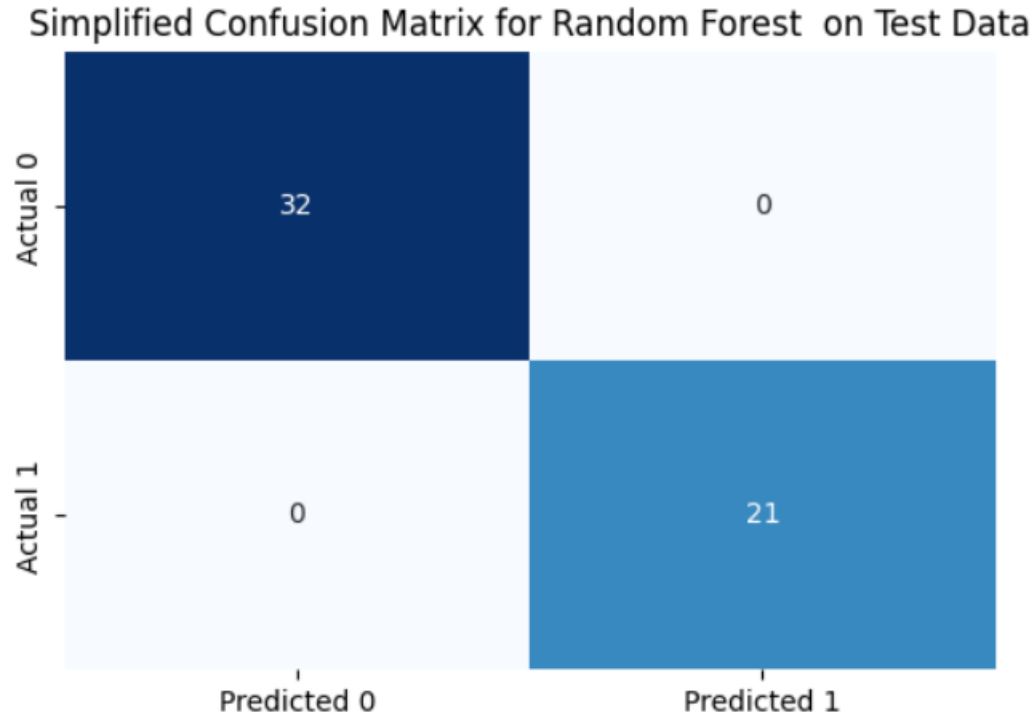


Random Forest Algorithm

```
from sklearn.metrics import precision_recall_curve

# Random Forest Classifier
print("Random Forest Classifier")
randomForestModel = RandomForestClassifier(random_state=18)
randomForestModel.fit(x_train, y_train)
randomForestModelPredict = randomForestModel.predict(x_test)

accuracy = accuracy_score(y_test, randomForestModelPredict) * 100
print(f"Accuracy on test data: {accuracy:.2f}%")

print("Classification Report :")
print(classification_report(y_test, randomForestModelPredict, labels=[0, 1]))

# Calculating confusion matrix
cf_matrix = confusion_matrix(y_test, randomForestModelPredict)

# Extracting TP, TN, FP, FN
TP = cf_matrix[1, 1]
TN = cf_matrix[0, 0]
FP = cf_matrix[0, 1]
FN = cf_matrix[1, 0]


# Plotting simplified confusion matrix
plt.figure(figsize=(6, 4))
sns.heatmap([[TN, FP], [FN, TP]], annot=True, fmt='d', cmap='Blues', cbar=False,
            xticklabels=['Predicted 0', 'Predicted 1'],
            yticklabels=['Actual 0', 'Actual 1'])
plt.title("Simplified Confusion Matrix for Random Forest  on Test Data")
plt.show()
```

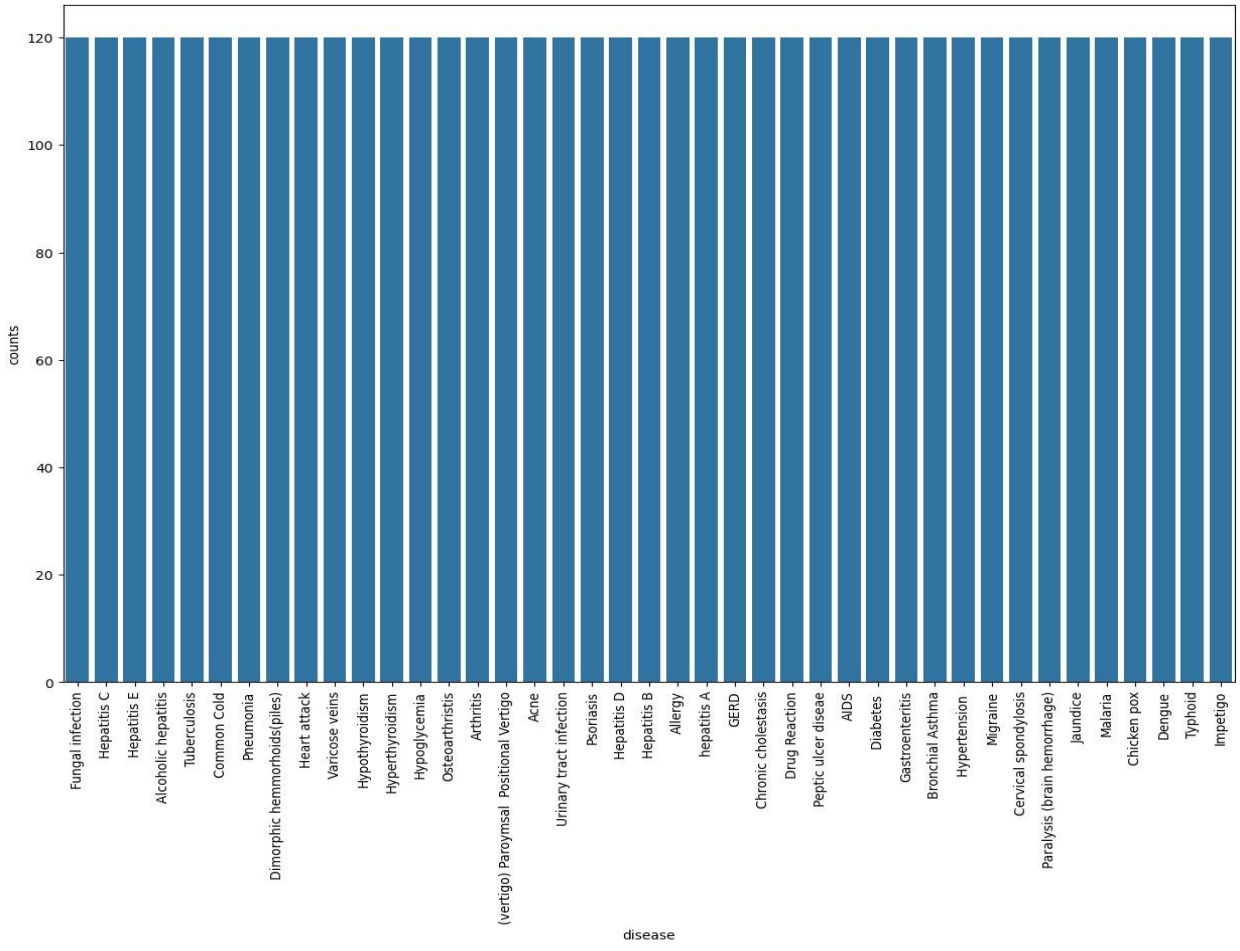
Random Forest Classifier
Accuracy on test data: 100.00%
Classification Report :

	precision	recall	f1-score	support
0	1.00	1.00	1.00	32
1	1.00	1.00	1.00	21
micro avg	1.00	1.00	1.00	53
macro avg	1.00	1.00	1.00	53
weighted avg	1.00	1.00	1.00	53



```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load dataset
train_data_path = "/content/Training.csv"
train_data = pd.read_csv(train_data_path).dropna(axis=1)

# Visualize disease counts
disease_counts = train_data['prognosis'].value_counts()
temp_dataframe = pd.DataFrame({
    'disease': disease_counts.index,
    'counts': disease_counts.values
})

plt.figure(figsize=(15, 10))
sns.barplot(x='disease', y='counts', data=temp_dataframe)
plt.xticks(rotation=90)
plt.show()
```

Counts of Each Disease in the Dataset



Final Result:

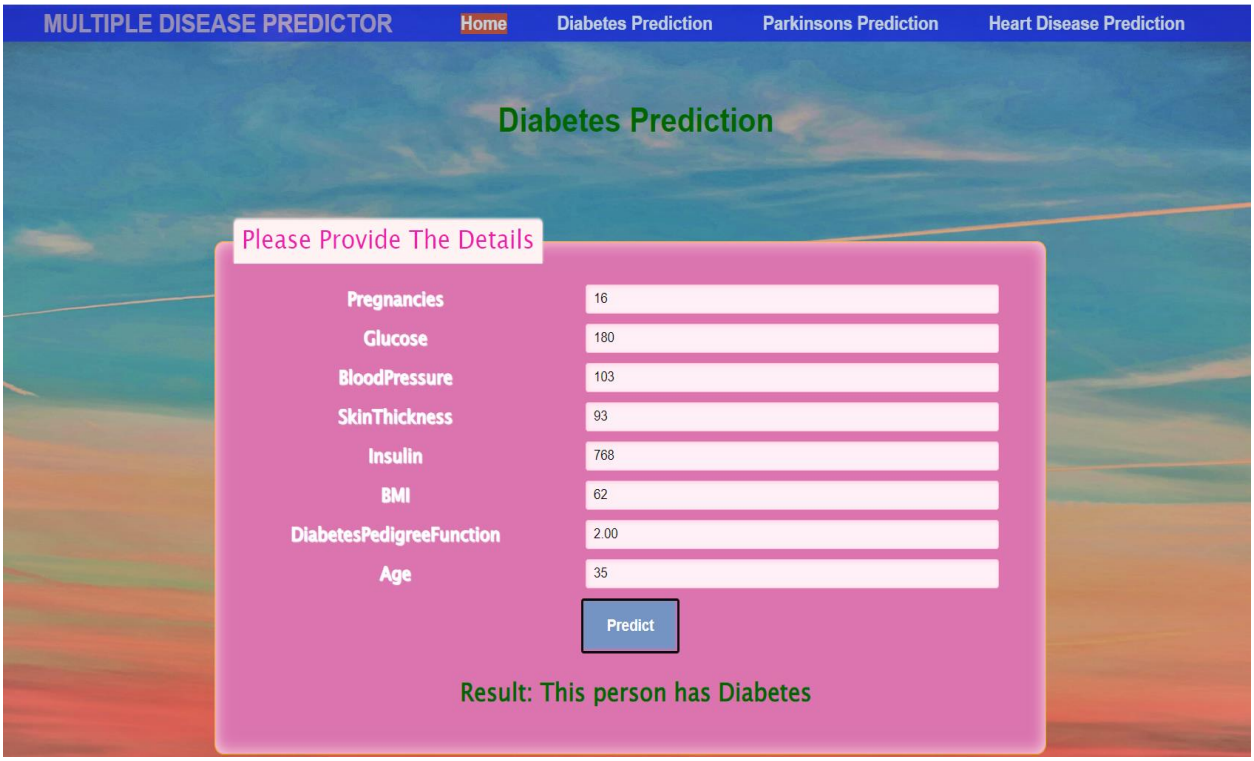
```
def predictDisease(symptoms):
  symptoms = symptoms.split(",")

  input_data = [0] * len(data_dict["symptom_index"])
  for symptom in symptoms:
    index = data_dict["symptom_index"][symptom]
    input_data[index] = 1


  input_data = np.array(input_data).reshape(1,-1)

  # generating individual outputs
  rf_prediction = data_dict["predictions_classes"][rfModelFit.predict(input_data)[0]]
  nb_prediction = data_dict["predictions_classes"][nbModelFit.predict(input_data)[0]]
  svm_prediction = data_dict["predictions_classes"][svmModelFit.predict(input_data)[0]]

  # making final prediction by taking mode of all predictions
  final_prediction = mode([rf_prediction, nb_prediction, svm_prediction])[0][0]
  predictions = {
    "rf_model_prediction": rf_prediction,
    "naive_bayes_prediction": nb_prediction,
    "svm_model_prediction": svm_prediction,
    "final_prediction":final_prediction
  }
  return predictions

# Testing the function
test_symptoms = "Vomiting,Breathlessness,Sweating,Chest Pain"
test_predictions = predictDisease(test_symptoms)
print(test_predictions)
```

```
{'rf_model_prediction': 'Heart attack', 'naive_bayes_prediction': 'Heart attack', 'svm_model_prediction': 'Heart attack', 'final_prediction': 'H'}
```

Disease Prediction Based On Clinical Measurements

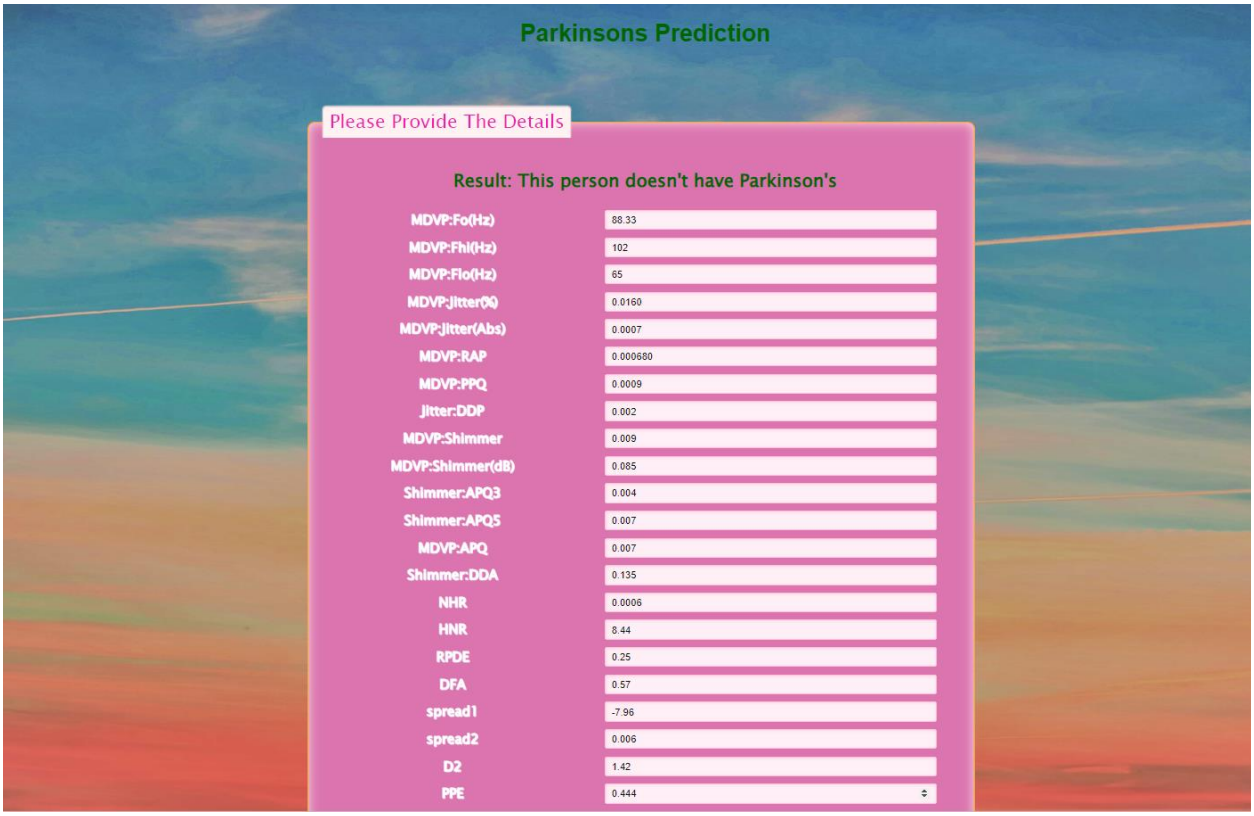
Normal Range Table For Diabetes Prediction:

Feature	Normal Range
Pregnancies	Varies widely among individuals, typically 0-5 over a lifetime
Glucose	Fasting: 70-140 mg/dL; 2 hours after eating: < 180 mg/dL
Blood Pressure	Approximately 120/80 mmHg
Skin Thickness	Varies by location, typically 1.5-4 mm
Insulin	Fasting: 2.6-24.9 µU/mL
BMI (Body Mass Index)	Normal: 18.5-24.9; Overweight: 25-29.9; Obese: ≥ 30
Diabetes Pedigree Function	No specific range, lower values indicate lower diabetes risk
Age	Varies widely, typically 18-100+ years old



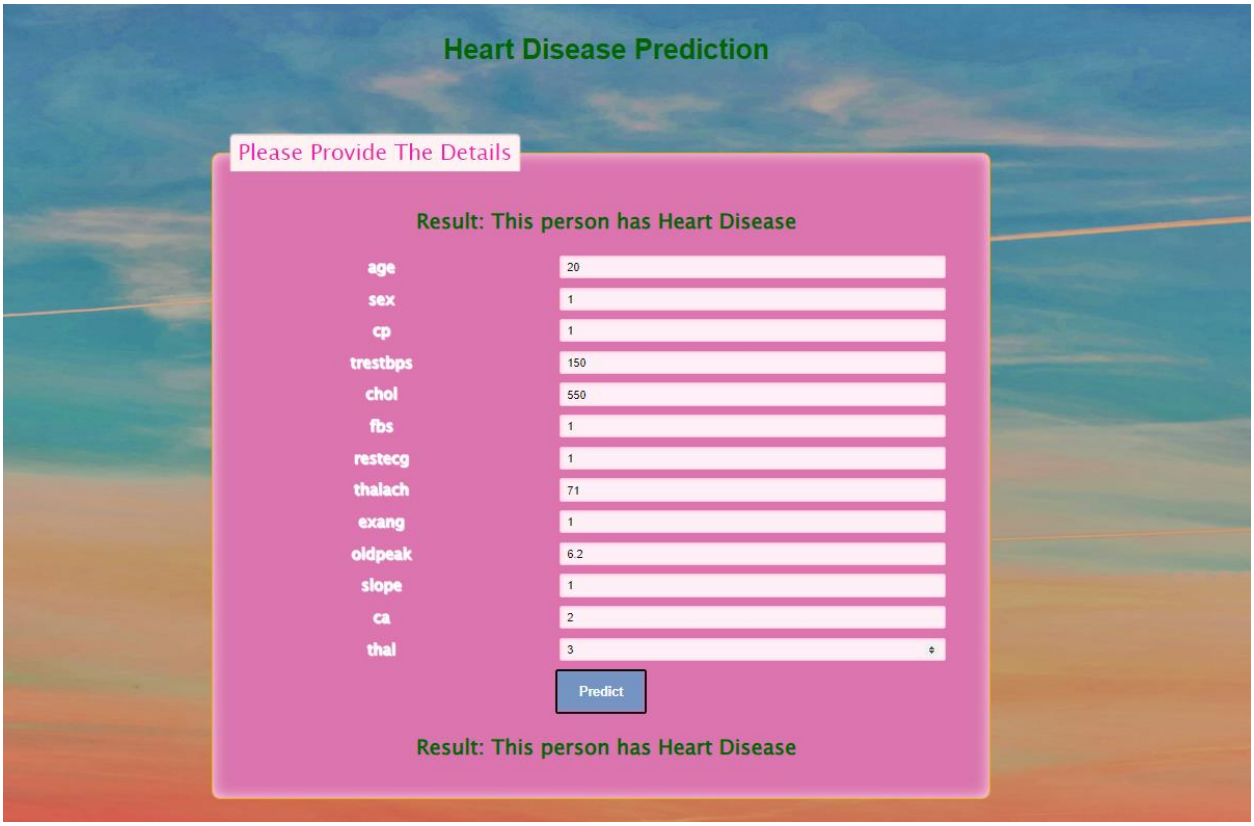
Normal Range Table For Parkinson's Prediction:

Feature	Full Form	Normal Range
MDVP:Fo(Hz)	Minimum and maximum vocal fundamental frequency (in Hz)	88.333 - 260.105 Hz
MDVP:Fhi(Hz)	Maximum vocal fundamental frequency variation (in Hz)	102.145 - 592.030 Hz
MDVP:Flo(Hz)	Minimum vocal fundamental frequency variation (in Hz)	65.476 - 239.170 Hz
MDVP:Jitter(%)	Cycle-to-cycle variation in vocal fold oscillation period (in %)	0.001680 - 0.033160
MDVP:Jitter(Abs)	Absolute jitter (in ms)	0.000007 - 0.000260
MDVP:RAP	Relative amplitude perturbation (in ms)	0.000680 - 0.021440
MDVP:PPQ	Five-point period perturbation quotient (in ms)	0.000920 - 0.019580
Jitter:DDP	Average absolute difference between jitter measures (in ms)	0.002040 - 0.064330
MDVP:Shimmer	Cycle-to-cycle variation in vocal fold amplitude (in dB)	0.009540 - 0.119080
MDVP:Shimmer(dB)	Amplitude perturbation quotient (in dB)	0.085000 - 1.302000 dB
Shimmer:APQ3	Three-point amplitude perturbation quotient (in dB)	0.004550 - 0.056470
Shimmer:APQ5	Five-point amplitude perturbation quotient (in dB)	0.005700 - 0.079400
MDVP:APQ	Amplitude perturbation quotient (in dB)	0.007190 - 0.137780
Shimmer:DDA	Average absolute differences between consecutive amplitude measures (in dB)	0.013640 - 0.169420
NHR	Noise-to-harmonics ratio	0.000650 - 0.314820
HNR	Harmonics-to-noise ratio	8.441000 - 33.047000
RPDE	Recurrence period density entropy	0.256570 - 0.685151
DFA	Detrended fluctuation analysis	0.574282 - 0.825288
spread1	Nonlinear measures of fundamental frequency variation	-7.964984 to -2.434031
spread2	Nonlinear measures of fundamental frequency variation	0.006274 - 0.450493
D2	Correlation dimension	1.423287 - 3.671155
PPE	Pitch period entropy	Normal range may vary, consult data


Parkinsons Prediction
Please Provide The Details
Result: This person doesn't have Parkinson's
MDVP:Fo(Hz)
88.33
MDVP:Fhi(Hz)
102
MDVP:Flo(Hz)
65
MDVP:Jitter(%)
0.0160
MDVP:Jitter(Abs)
0.0007
MDVP:RAP
0.000680
MDVP:PPQ
0.0009
Jitter:DDP
0.002
MDVP:Shimmer
0.009
MDVP:Shimmer(dB)
0.085
Shimmer:APQ3
0.004
Shimmer:APQ5
0.007
MDVP:APQ
0.007
Shimmer:DDA
0.135
NHR
0.0006
HNR
8.44
RPDE
0.25
DFA
0.57
spread1
-7.96
spread2
0.006
D2
1.42
PPE
0.444

Normal Range Table For Heart Disease Prediction:

Feature	Description	Normal Range
age	Age of the patient	29-77 years
sex	Gender of the patient	0 (female), 1 (male)
cp	Chest pain type	0, 1, 2, 3
trestbps	Resting blood pressure (in mm Hg)	94-200 mm Hg
chol	Serum cholesterol (in mg/dl)	126-564 mg/dl
fbs	Fasting blood sugar > 120 mg/dl	0 (false), 1 (true)
restecg	Resting electrocardiographic results	0, 1, 2
thalach	Maximum heart rate achieved (in beats per minute)	71-202 bpm
exang	Exercise induced angina	0 (no), 1 (yes)
oldpeak	ST depression induced by exercise relative to rest	0-6.2
slope	Slope of the peak exercise ST segment	0, 1, 2
ca	Number of major vessels (0-3) colored by flourosopy	0, 1, 2, 3, 4



Conclusion:

In our project, we delve into the realm of disease diagnosis, utilizing machine learning algorithms to enhance the accuracy and efficiency of the diagnostic process. We focus on two distinct aspects: symptom-based diagnosis and clinical measurement-based diagnosis.

For the symptom-based diagnosis, we explore the effectiveness of Support Vector Machine (SVM), Naive Bayes, and Random Forest algorithms. By analyzing patient symptoms and profiles, our aim is to develop a robust disease diagnosis system capable of accurately identifying common chronic illnesses.

In parallel, we delve into clinical measurement-based diagnosis, focusing specifically on Parkinson's disease and heart disease. For Parkinson's disease, we employ Support Vector Machine (SVM), leveraging clinical measurements to predict the likelihood of disease occurrence. Conversely, for heart disease diagnosis, we utilize Logistic Regression, harnessing clinical measurements to assess the risk of cardiovascular ailments.

Through a meticulous analysis of these machine learning algorithms, we aim to provide valuable insights into their effectiveness in disease diagnosis. By evaluating metrics such as accuracy, precision, and recall, we seek to identify the most suitable algorithms for each diagnostic scenario.

Our project's ultimate goal is to develop reliable and efficient disease diagnosis systems that empower healthcare professionals with the tools they need to make informed decisions and provide timely interventions for patients. By integrating machine learning techniques into disease diagnosis, we aim to revolutionize healthcare delivery and improve patient outcomes.

Future Scope:

1.Real-time Monitoring Systems: Develop real-time disease monitoring systems that continuously analyze patient data from wearable devices, electronic health records, and other sources to detect early warning signs of diseases and provide timely interventions.

2.Precision Medicine: Embrace the concept of precision medicine, which involves tailoring treatment strategies to individual patients based on their unique genetic makeup, lifestyle factors, and environmental influences. Machine learning algorithms can help identify optimal treatment plans and predict patient responses to specific therapies.

3. Automated Disease Diagnosis from Cell Images: One promising future scope in healthcare is the development of automated disease diagnosis systems from cell images. By leveraging advanced image analysis techniques and machine learning algorithms, these systems have the potential to revolutionize healthcare by enabling early detection, accurate diagnosis, and personalized treatment strategies.