

# **Dengue Disease prediction using Random Forest algorithm**

A Thesis based project, submitted for the fulfillment of the requirement  
for the degree of Master's in Information Technology



Institute of Information and Communication Technology  
Shahjalal University of Science and Technology  
Kumargaon, Sylhet-3114, Bangladesh  
12th November 2024

# **Dengue Disease prediction using Random Forest algorithm**

## **Submitted by**

Poritosh Modak (2022822023)

Roushni Rafa Majumder (2022822033)

Mahbub Hussain (2022822029)

Bipro Bhowmik (2022822025)

Md. Nafiul Hasan Chowdhury (2022822007)

Afruja Akter Rumpa (2022822012)

## **Supervised by**

Dr. Ahsan Habib  
Associate Professor  
IICT, SUST

A Thesis based project, submitted for partial fulfillment of the requirement  
for the degree of Master's in Information Technology



Institute of Information and Communication Technology  
Shahjalal University of Science and Technology  
Kumargaon, Sylhet-3114, Bangladesh  
12th November 2024

## **Declaration**

We hereby declare that the project is our original work, and it has been written by us in its entirety.

We have duly acknowledged all the sources of information which have been used in the report.

The report has also not been submitted for any degree in any other university previously.

Poritosh Modak

Roushni Rafa  
Majumder

Mahbub  
Hussain

Bipro Bhowmik

Md. Nafiul Hasan  
Chowdhury

Afruja Akter  
Rumpa

## **Recommendation Letter from Project Supervisor**

The students, Poritosh Modak, Roushni Rafa Majumder, Mahbub Hussain, Bipro Bhowmik, Md. Nafiul Hasan Chowdhury and Afruja Akter Rumpa whose project report entitled “Dengue Disease prediction using Random Forest algorithm”, is under my supervision and agrees to submit for examination

Supervisor

.....

Dr. Ahsan habib

Associate Professor

Institute of Information and Communication Technology  
Shahjalal University of Science and Technology

## **Acknowledgments**

We are grateful to our supervisor Dr. Ahsan Habib, Associate Professor, IICT, SUST for his vigilant guidelines and continuous support on our project work. We also thankful to DataSoft System Bangladesh Ltd and his resource personnel Khaled Salah Uddin, who acted as a project manager of this project as industrial attachments. A special thanks to Dr. Tuhin Barua Tamal (Child disease Specialist) and Dr. Benjamin (Dengue Focal Point, Sylhet M.A.G Osmani Medical College Hospital) for clarifying our need by arranging a lot of meeting and provided a clear and concise idea on our research work especially for data collection.

# Qualification Form of Master's Degree

## Students Name:

Poritosh Modak(2022822023)

Roushni Rafa Majumder (2022822033)

Mahbub Hussain (2022822029)

Bipro Bhowmik (2022822025)

Md. Nafiul Hasan Chowdhury (2022822007)

Afruja Akter Rumpa (2022822012)

**Research Title:** Dengue Disease prediction using Random Forest algorithm.

This is to certify that the research-based project report submitted by the students named above in November 2024 are qualified and approved by the Examination Committee.

.....  
Director, IICT

.....  
Chairman, Examination Committee

.....  
Supervisor

Dengue Disease prediction using Random Forest algorithm

## **Abstract**

This project tackles a pressing challenge in global health: the early and accurate prediction of dengue fever, particularly in high-risk regions like Bangladesh. Leveraging the power of the Random Forest algorithm, we developed a machine learning model that predicts dengue fever based on symptom profiles and clinical data. This model aims to support healthcare providers by offering reliable predictions even when laboratory results are unavailable or inconclusive. Implemented in Python, our solution is designed for flexibility and integration, using cloud infrastructure and IoT devices to facilitate timely, remote access for healthcare practitioners.

The project's methodology includes extensive data preprocessing, feature selection, and hyper-parameter tuning to ensure optimal model performance. Drawing on the strengths of Random Forest for complex classification tasks, our approach balances predictive accuracy with interpretability, essential for clinical use. Through consultations with medical professionals and analysis of real-world data, we tailored the model to reflect the nuanced presentation of dengue symptoms across cases.

Our results, measured through metrics like accuracy, precision, recall, F1 score, and AUC, demonstrate the model's strong predictive capability. This tool offers a significant step forward in the fight against dengue, providing a resource-efficient, machine learning-driven approach to disease management. By integrating machine learning with health data analysis, this project highlights the transformative role of technology in enhancing diagnostic accuracy and supporting proactive healthcare solutions.

## List of Tables

Table 1: Dengue Cases, Deaths and Case Fatality rate .....	14
Table 2: Preprocessing of Data .....	22
Table 3: Accuracy based of different types of Models.....	29
Table 4: Hyper-Parameters settings.....	34



## List of Figures

Figure 1: Deaths per 1,000,000 inhabitants.....	11
Figure 2: Model Architecture.....	15
Figure 3: Bagging.....	16
Figure 4: Tree Construction.....	17
Figure 5: Features Selection.....	21
Figure 6: Count of Dengue Positive Cases.....	22
Figure 7: Count of Dengue cases of Male and Female.....	23
Figure 8: Dengue Patient with age.....	23
Figure 9: Frequency of Occurrence.....	24
Figure 10: Day Count Symptoms.....	26
Figure 11: Evaluation Matrices.....	27

## Table of Contents

Acknowledgments .....	iii
List of Tables .....	iv
1 Introduction .....	12
2 Related Work.....	14
3 Model Analysis.....	18
3.1 Random Forest Overview .....	14
3.1.1 Random forest: .....	18
3.1.2 Decision Trees: .....	18
3.2 Model Architecture .....	19
3.2.1 Bagging (Bootstrap Aggregating): .....	20
3.2.2 Random Feature Selection: .....	21
3.2.3 Tree Construction: .....	21
4.1 Methodology .....	22
4.1.1 Data Preparation: .....	22
4.1.2 Exploratory Data Analysis.....	25
4.1.3 Decision making.....	28
4.1.4 Data Normalization .....	30
4.1.5 Data Split: .....	30
4.2 Evaluation Metrics .....	31
4.3 Machine Learning Methods .....	34
4.4 Model implementation and UI interaction .....	35

5 Conclusion and Future work .....	38
6 Reference.....	39

## **Chapter 1: Introduction**

Dengue fever is a significant public health challenge, particularly in tropical and subtropical regions like Brazil and Bangladesh. It is a viral illness spread by mosquitoes and can cause symptoms ranging from mild fever, joint pain, and headaches to more severe conditions like hemorrhagic fever or shock syndrome, which can be fatal if not treated in time. One of the biggest difficulties in managing dengue is its ability to present with symptoms similar to other illnesses, making it challenging to diagnose quickly and accurately. In many cases, the standard laboratory tests for dengue—such as the NS1 antigen or IgM assays—can be negative, especially in atypical cases, leaving healthcare providers unsure of the diagnosis. This creates a gap in timely diagnosis and treatment, which is critical, particularly in regions with high dengue incidence.

This project aims to address this diagnostic challenge by applying machine learning techniques, specifically the Random Forest algorithm, to predict dengue based on a combination of clinical symptoms and laboratory data. The goal is to provide healthcare workers with a more reliable tool to diagnose dengue early, even in cases where traditional tests may not give clear results. By using a large dataset of dengue cases from Brazil, we have developed a model that can analyze patient symptoms—like fever, rash, pain, and fatigue—along with laboratory findings to predict whether a person has dengue fever.

The Brazil dataset is particularly valuable for this project because it contains a range of real-world cases, showing different symptoms and severity levels. This gives the model the opportunity to learn from a variety of examples and build a more accurate prediction tool. The Random Forest algorithm is ideal for this kind of task because it can handle large datasets with many variables, like symptoms, lab results, and patient demographics. It works by creating multiple decision trees and combining their results to make a final prediction, which helps improve accuracy and avoid overfitting, a common problem in machine learning.

The first step of our project involves data preprocessing—cleaning and preparing the data to ensure it's accurate and complete. Then, we apply feature selection to focus on the most important symptoms and laboratory values that contribute to dengue diagnosis. Once we've

trained the model, we evaluate its performance using several important metrics, including accuracy, precision, recall, and F1 score, to measure how well it distinguishes between dengue and non-dengue cases. We also calculate the area under the curve (AUC) to assess how effectively the model ranks cases based on the likelihood of being dengue-positive.

In addition to the core machine learning model, this project considers how the predictive tool can be used in real-world healthcare settings. We explore the potential for integrating the model with modern technologies such as Internet of Things (IoT) devices and cloud platforms, which could allow doctors and healthcare providers to access the tool remotely, in real-time. This integration would enable better management of patient data, quicker diagnoses, and the ability to monitor patients from a distance—especially in areas with limited access to hospital resources or where dengue outbreaks are occurring rapidly.

By combining machine learning with healthcare technology, we aim to develop a tool that can support healthcare workers in diagnosing dengue more accurately and efficiently. This project has the potential to significantly improve the way dengue is diagnosed in regions where the disease is endemic, helping to reduce diagnostic delays, improve patient outcomes, and alleviate pressure on healthcare systems that are often overwhelmed during outbreaks.

Ultimately, this project not only demonstrates the potential of machine learning in healthcare but also offers a practical solution to a longstanding challenge. By improving the speed and accuracy of dengue diagnosis, we hope to help healthcare providers make more informed decisions, ultimately saving lives and reducing the burden of dengue fever on communities globally.

## Chapter 2: Related Work

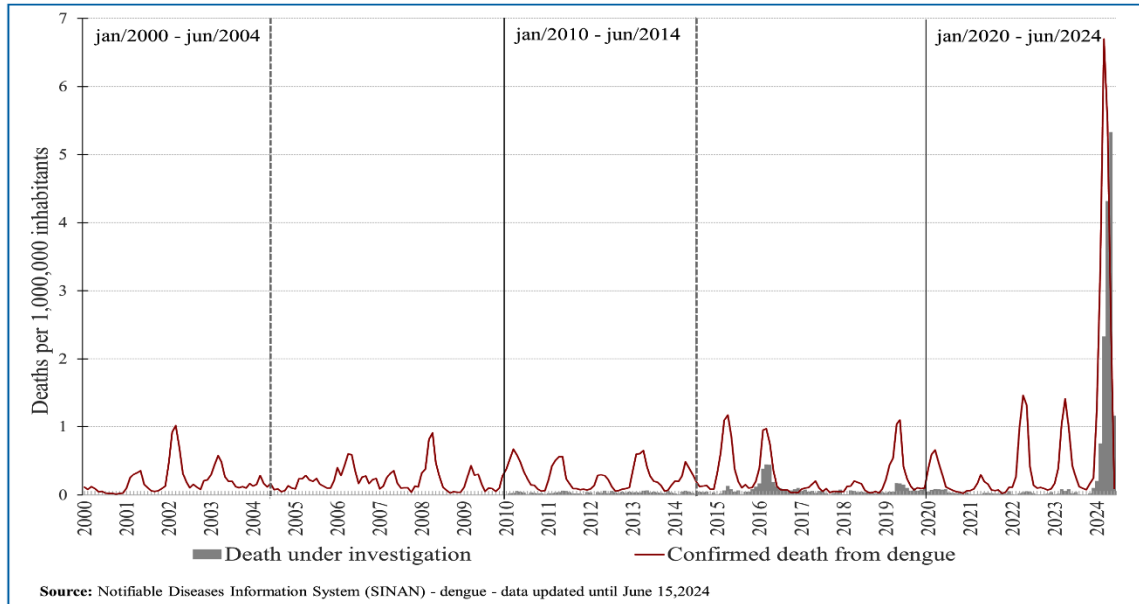
Generally, every year when the temperature and rainfall increase, the mosquitoes which carry dengue viruses, expand their range. The World Health Organization (WHO) declared Dengue as endemic in Bangladesh and is one of the major public health concerns of the country. Dengue virus has the potential to cause epidemics resulting in high morbidity and mortality. From 1 January to 7 August 2023, the Ministry of Health and Family Welfare of Bangladesh reported a total of 69483 laboratory-confirmed dengue cases and 327 related deaths among which 63% of cases and 62% of the deaths were reported in the month of July 2023.

Year	Cases	Deaths	Case Fatality Rate (CFR)
2018	10 148	26	0.26%
2019	101 354	164	0.16%
2021	28 429	105	0.37%
2022	62 382	281	0.45%
2023 (as of 7 August)	69 483	327	0.47%

Data source: DGHS (\*data for 2020 is limited due to COVID-19)

**Table 1:** Dengue cases, deaths, and case fatality rate in Bangladesh for 2018-23

The graph below displays the trend in dengue-related deaths in Brazil from 2000 to mid-2024. The red line represents confirmed deaths from dengue, which show periodic peaks, indicating recurring outbreaks over the years. Starting in 2023, there is a dramatic increase in confirmed deaths, reaching a peak in early 2024, along with a sharp rise in deaths under investigation (gray bars). This recent spike may be linked to climate conditions favoring mosquito breeding, healthcare system strain, or the emergence of new dengue strains. Overall, this pattern highlights the urgent need for strengthened public health efforts, including mosquito control, rapid response systems, and preventive measures to manage and reduce the impact of future dengue outbreaks.



**Figure 1: Deaths per 1,000,000 inhabitants**

Researchers have been studying the dengue viruses to understand the factors that are responsible for transmitting the virus from mosquitoes to humans. The literature on dengue fever viewed as three aspects viz. Biological, computational and bio-computational [1]. This project is focused on the computational aspect.

There are several computational techniques in Machine Learning which are used to predict and diagnose dengue fever each of which contribute to improved accuracy and efficiency in diagnostic processes. A range of algorithms has been explored in recent research endeavors, shaping the landscape of Dengue prediction.

Marimuthu et al. [1] proposed a bio-computational methodology for mapping gene sequences to construct dengue viral association. It achieved 96.74% accuracy by establishing classification and association rules using standard tools. Dengue had already been endemic in India. Bangladesh shares 94% of its land border with India and in the eastern Indian states Aedes mosquitoes breed and circulate most. Several studies have been conducted to predict and diagnose the dengue patients of India.

In a study, 97% of the prediction accuracy was achieved employing the decision tree classification model by analyzing the features of affected patients in India [2]. Likewise, P. Manivannan et al. [3] developed a classification and clustering model to detect dengue viruses using patient data from several Indian states. Shaukat, K., et al. [4] used the DBSCAN algorithm for dengue fever clustering to illustrate the overall behavior of dengue in the district Jhelum and evaluated several clustering algorithms (such as k-means, K-medoids, DBSCAN, and OPTICS) using graphs based on the dataset. N. A. Husin et al. [5] proposed a model based on environmental features using the support vector machine. For feature selection, the model used PCA, and for model execution, it used c-SVM with the Gaussian kernel. The model outperformed their earlier efforts in terms of accuracy. Subitha et al. [6] presented a mining model for dengue fever. In this paper, they implemented KNN, and mining performance was improved. The classification result was analyzed using a back-propagation network. It has an accuracy of 98%. Decision trees and ensemble methods, particularly Random Forests, have gained popularity due to their interpretability and proficiency in handling diverse datasets. These algorithms exhibit promise in predicting Dengue outbreaks and providing valuable assistance in patient diagnosis. The Dengue outbreak got severe in Bangladesh in recent years yet study on the Dengue patients based particularly on the case studies of this country is very slim in number. In a study based on the real-time raw data samples of various types of dengue fever patients from the Medicine Department of Chittagong Medical College Hospital and Dhaka Medical College Hospital, Bangladesh, a new machine learning approach was taken to predict dengue fever. They had splitted the dataset into 70:30 ratios using 70% for training and 30% for test purposes. They had applied the Decision tree as the classification model to predict the fever and it achieved an average accuracy of 79% [7]. In the year of 2021, Dourjoy et al. [8] conducted a survey and collected data of 600 dengue patients of Bangladesh. Based on this dataset, they have tried to predict Dengue fever using SVM and Random Forest approach. In both the models, they have achieved approximately 70% accuracy. Time period of the data source is not defined. In 2021, Paul et al. [9] researched through the Climate data of Bangladesh Meteorological Department and found that Dengue transmission season could eventually extend to all-year-round. In 2022, Hossain et al. [10] used Dengue data from the DGHS to experiment through the Quasi Poisson Model to successfully predict Dengue



incidence using seasonal climate data. In that same year, Dey Sk. et al. [11] created a DengueBD dataset and employed two regression models and found out that the number of dengue cases reduces throughout the winter season in the country and increases mainly during the rainy season in the next ten months, from August 2021 to May 2022. Different practices and procedures have been done to detect Dengue Fever using various other data sources containing patient symptoms and other related environment measures. Tanner et al. [12] ] have used pathological features to detect Dengue Fever and thereby make a distinction with other chronic diseases. Ahmed et al. [13] have given a survey of various systems to detect the Dengue Fever successfully, including the comparisons, advantages, and also limitations of the expert system. Husin et al. [14] have used dengue related-data data from a hospital to design a self-notified model to detect Dengue. In this model, the diagnosis rules are generated by interviewing doctors, and finally, an expert in a system using fuzzy logic will use those rules to detect the Dengue Fever. Phakhounthong et al. [15] have projected a model to predict the fever severity of Dengue by Decision Tree (DT) and the importance of the individual feature by logistic regression. Boruah et al. [16] have proposed a way to find the risk factor of Parkinson's disease by using DT. The rules generated from DT are processed to find the important factor, which is the main cause of the disease. Gangula et al. [17] have used Ensemble Machine Learning technique to identify characteristics responsible for Dengue. Three modeling approaches of dengue review have been given by Hoyos et al. [18] and have concluded that Logistic models are the most used for the diagnosis of dengue. The performance of the Artificial Neural Network (ANN) classifier and Random Forest are evaluated for prediction of Dengue Fever by Silitonga et al. [19]. Hasan et al. [20] proposed an ensemble model and showed that the preprocessing plays a vital role in prediction.

## **Chapter 3: Model Analysis**

### **3.1 Random Forest Overview**

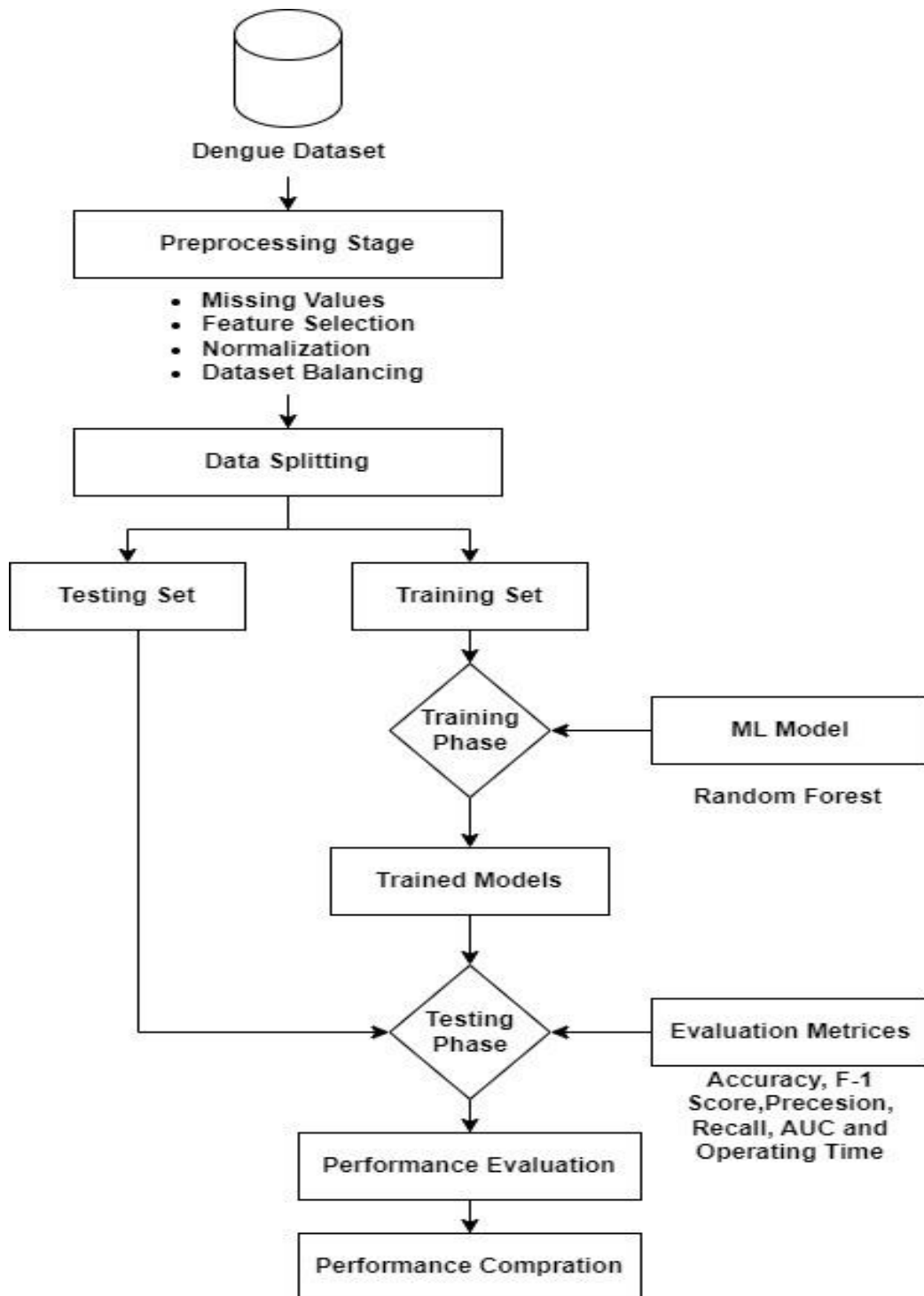
#### **3.1.1 Random Forest:**

The Random Forest Algorithm is highly popular because it's easy to use and can handle different types of problems like classification and regression. It's like a versatile tool in machine learning that's good at understanding and making predictions from complex datasets. One of its strong points is that it's skilled at preventing overfitting, which is when a model learns too much from its training data and struggles with new, unseen data. Because of these qualities, Random Forest is widely used in machine learning for various prediction tasks.

#### **3.1.2 Decision Trees:**

The basic building blocks of Random Forest are decision trees. A decision tree is a flowchart-like structure where each internal node represents a decision based on a specific feature, each branch represents an outcome of that decision, and each leaf node represents the final predicted label or value. Decision trees are prone to overfitting, meaning they may capture noise in the training data and perform poorly on new, unseen data.

### 3.2 Model Architecture



**Figure 2:** Model Architecture

### 3.2.1 Bagging (Bootstrap Aggregating):

Random Forest employs an ensemble learning technique called bagging to reduce overfitting and increase the model's robustness.

Bagging involves creating multiple subsets of the training data through random sampling with replacement (bootstrap sampling). Each subset is used to train a separate decision tree. Here are the steps involved in Bagging:

**Selection of Subset:** Bagging starts by choosing a random sample, or subset, from the entire dataset.

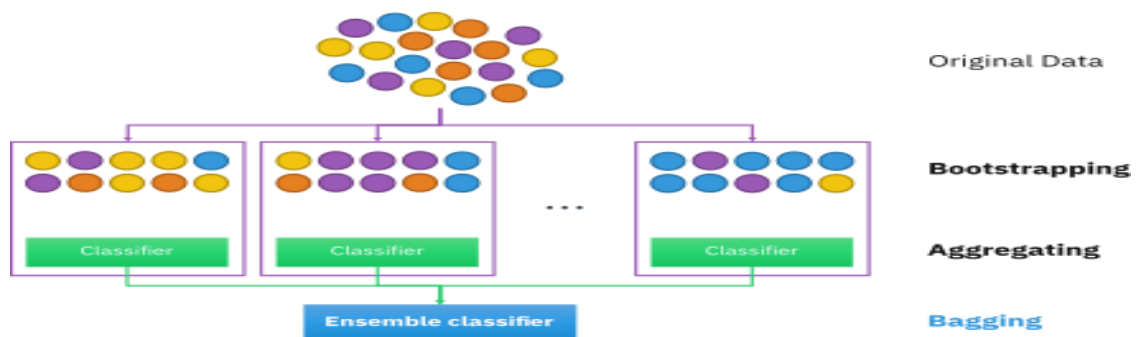
**Bootstrap Sampling:** Each model is then created from these samples, called Bootstrap Samples, which are taken from the original data with replacement. This process is known as row sampling.

**Bootstrapping:** The step of row sampling with replacement is referred to as bootstrapping.

**Independent Model Training:** Each model is trained independently on its corresponding Bootstrap Sample. This training process generates results for each model.

**Majority Voting:** The final output is determined by combining the results of all models through majority voting. The most commonly predicted outcome among the models is selected.

**Aggregation:** This step, which involves combining all the results and generating the final output based on majority voting, is known as aggregation.



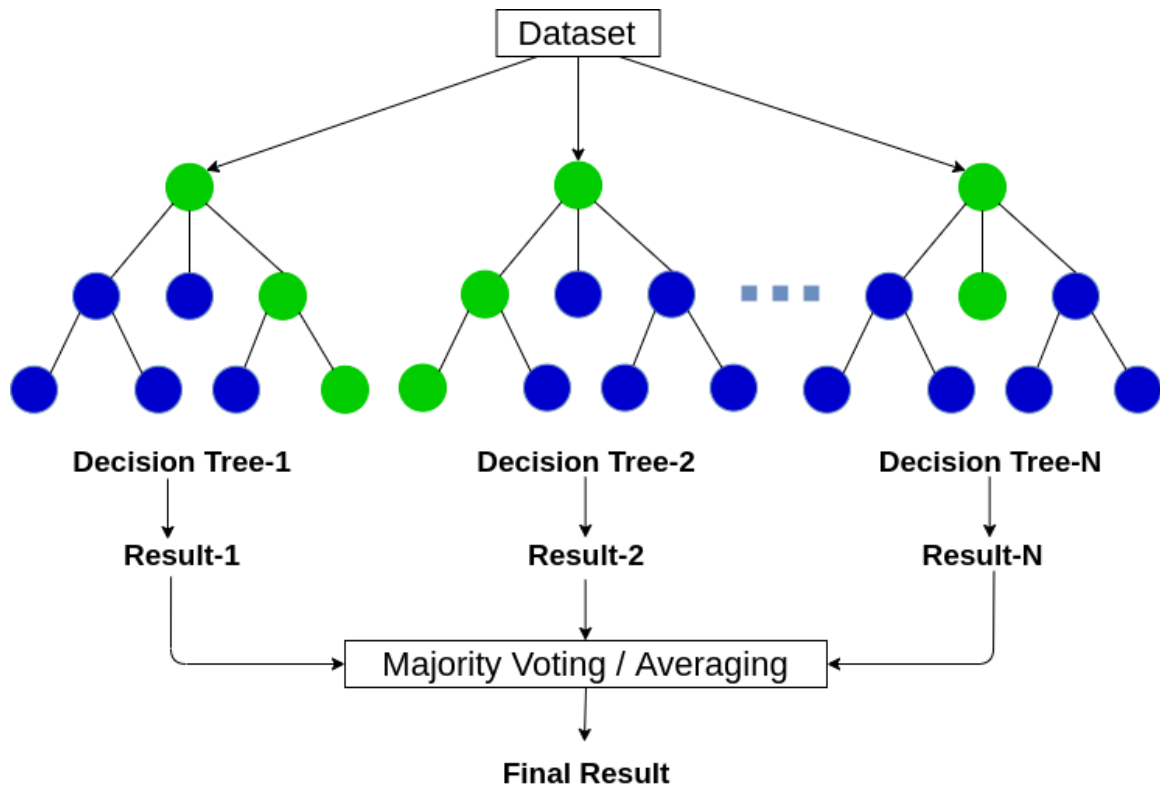
**Figure 3:** Bagging

### 3.2.2 Random Feature Selection:

Random Forest introduces randomness by considering only a random subset of features for each split in each decision tree. This ensures diversity among individual trees and prevents a single dominant feature from exerting excessive influence on the entire forest.

### 3.2.3 Tree Construction:

Building multiple decision trees in a Random Forest involves constructing each tree independently. This is achieved by using bootstrapped subsets of the original data and selecting random feature subsets for each tree. The construction continues until a specified depth is reached or until nodes have a minimum number of samples.



**Figure 4:** Tree Construction

## Chapter 4: Methodology and Exploratory Data Analysis

### 4.1 Methodology

The project consists of five phases: i) Data Preparation, ii) Exploratory Data Analysis, iii) Decision Making iv) Evaluation metrics & v) Model implementation & UI interaction.

#### 4.1.1 Data Preparation:

Data Preparation is a very essential step before proceeding to ML model building. We have used the dataset of confirmed Dengue patients in the state of Amazonas and the city of Recife, Pernambuco from the year 2015 to 2020 [21]. Regarding the state of Amazonas, data was retrieved from the SINAN. SINAN is the official system for disease reporting in Brazil. The data set for Recife was retrieved from an open data set named Portal de Dados Abertos do Recife, maintained by the Recife Health Department, whose primary source is also the SINAN, and therefore it follows the same dictionary pattern, and allows integration without further issues. The dataset was prepared following all the ethical guidelines.

Sex	Race	Residence_Area	Fever	Myalgia	Headache	Rash	Vomiting	Nausea	Back_Pain	Conjunctivitis	Arthritis	Artralgia	Petechiae	Tourniquet_Test	Retroorbital_Pain	Diabetes
1	9	1	1	2	2	2	1	1	2	2	1	1	2	2	2	2
0	9	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2
1	9	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2
1	9	1	1	1	1	2	2	1	2	2	2	1	2	2	2	2
1	4	1	1	1	2	2	1	2	2	2	2	1	2	2	2	2
1	9	1	1	2	2	2	2	2	2	2	2	1	2	2	2	2
1	4	1	1	2	1	2	2	1	2	2	2	2	2	2	2	2
1	9	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2
0	9	1	1	1	2	2	2	2	2	2	2	1	2	2	2	2
0	9	1	1	2	1	1	2	2	2	2	2	1	2	2	2	2

Hematological_Disease	Liver_Disease	Kidney_Disease	Hypertension	Peptic_Acid_Disease	Auto_Immune_Disease	Day_Count_of_Symptoms	Patient_Age_Years	Classification
2	2	2	2	2	2	8	9	0
2	2	2	2	2	2	10	42	1
2	2	2	2	2	2	4	11	0
2	2	2	2	2	2	9	22	1
2	2	2	2	2	2	9	17	0
2	2	2	2	2	2	0	43	0
2	2	2	2	2	2	0	35	1
2	2	2	2	2	2	1	15	1
2	2	2	2	2	2	8	39	0
2	2	2	2	2	2	56	17	0

**Table 2:** Preprocessing of Data

In the realm of machine learning applications, the key to success lies in effective data processing. This essential procedure works like magic, turning raw and chaotic data into

a refined and usable form. Prior to initiating any analysis, it is imperative to give the data a thorough cleaning, eliminating impurities, ensuring consistency, and eliminating unwanted noise. This entire transformational process is referred to as text preprocessing. Here data is preprocessed keeping in mind the problem statement as well as the type and size of data. Some of the major steps of data preparation are:

1) **Removal of Irrelevant Features:** We have removed the unnecessary column from the dataset that was analyzed. These irrelevant features didn't have any impact on the final prediction. Id, race and gestational age columns were completely removed.

2) **Data Wrangling:** In this experiment, we had to convert the header of the columns of the dataset from Portuguese to English. We had to clean the unnecessary data. Approximately 1000 of unnecessary data were removed from this dataset.

3) **Handling Missing Data:** The dataset had numerous null values. We had to carefully clean it maintaining the purity of the dataset.

Our study delves into the pre-processing of a real-world dataset for machine learning applications. One of the primary challenges encountered was the presence of missing values in specific features, as depicted in Figure. To address this issue, we adopted the mean imputation technique, enabling robust data preparation for subsequent analysis.

The `isnull()` method in pandas can be applied to the entire dataset, generating a Boolean mask where True represents missing values

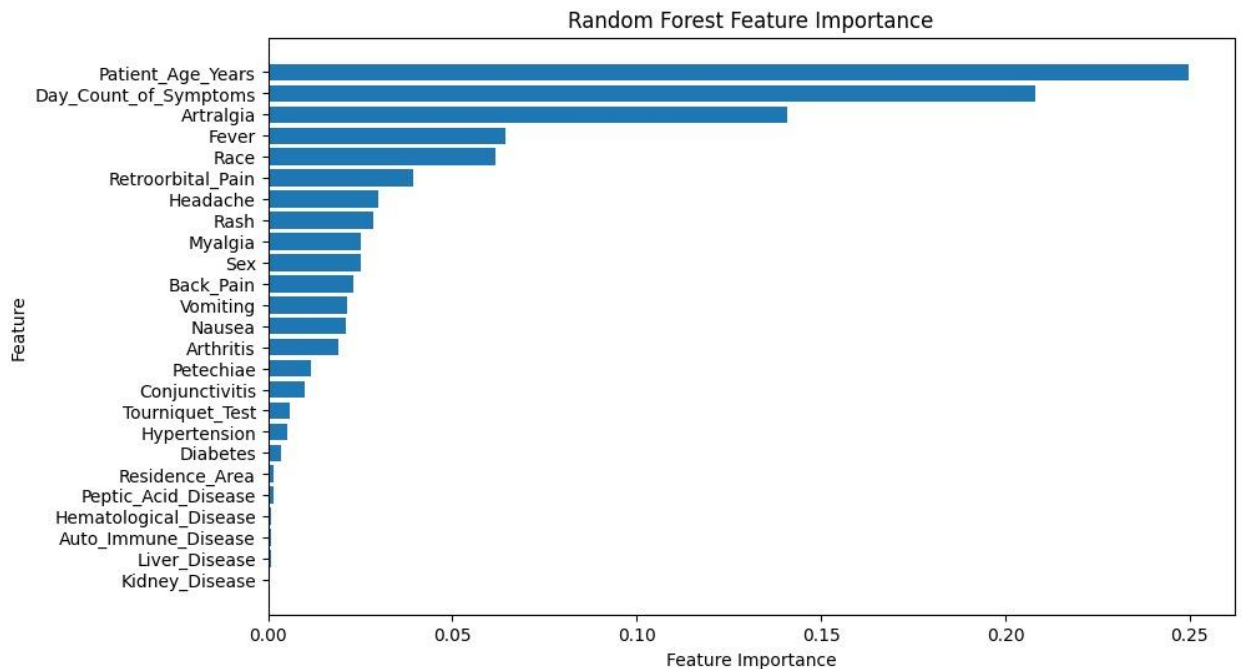
After identifying missing values, we used to impute them with the mean, median, or other strategies using the `fillna()` method.

4) **Label Encoding of the features:** As ML models expect numerical input to train the model, it's necessary to convert the features to a numerical format. Most of the features here are binary in nature so are encoded to [0,1] using Label Encoder.

5) **Class Balancing:** In Machine Learning, when there is a bias or skewness towards the majority class present in the dependent variable, it is considered a class imbalance

problem. Hence, to predict whether a person is having DF based on the given attributes (independent variables) class imbalance must be corrected. There is an assumption of even data distribution within classes in ML algorithms, especially with DT and RF. The extensive issue in the class imbalance problem is that the algorithm will be more biased towards predicting the majority class. The algorithm will not learn the patterns in the minority class as it does not have enough data points for the minority class. That is why there will be high classification errors for the minority class. We tried to solve this class imbalance problem.

6) **Feature Selection:** For feature selection, the Extra Tree method was utilized to conduct feature ranking. Figure 3 illustrates the importance of features as predicted by the Extra Tree method. Utilizing this approach, the Extra Tree (ET) method identified 20 features deemed the most pertinent for this study.

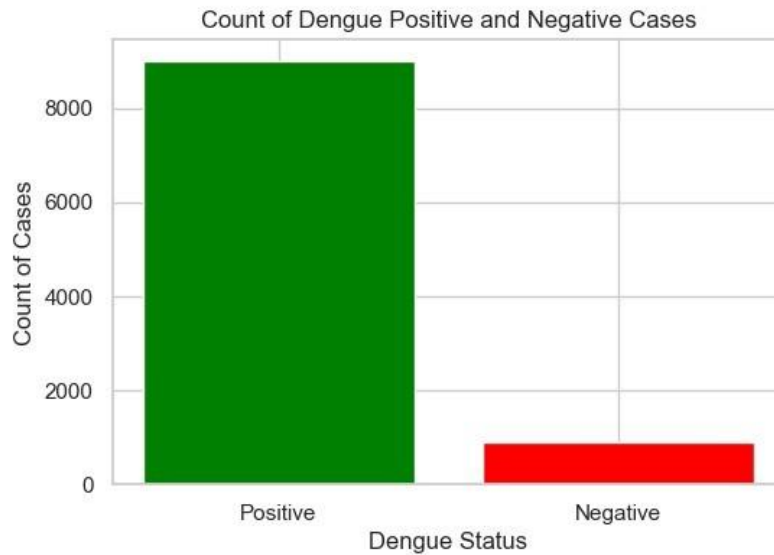


**Figure 5: Features Selection**

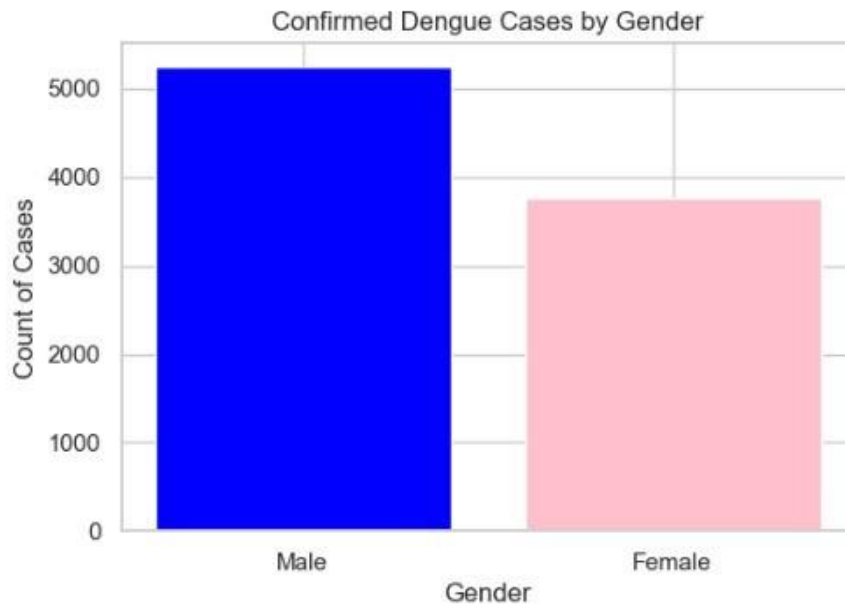


#### 4.1.2 Exploratory Data Analysis:

In the dataset we have analyzed, we had 9006 patient's data with dengue positive and 886 patient's data with dengue negative. Amongst the dengue positive case, a little more than 5000 patients were male and close to 4000 patients were female.

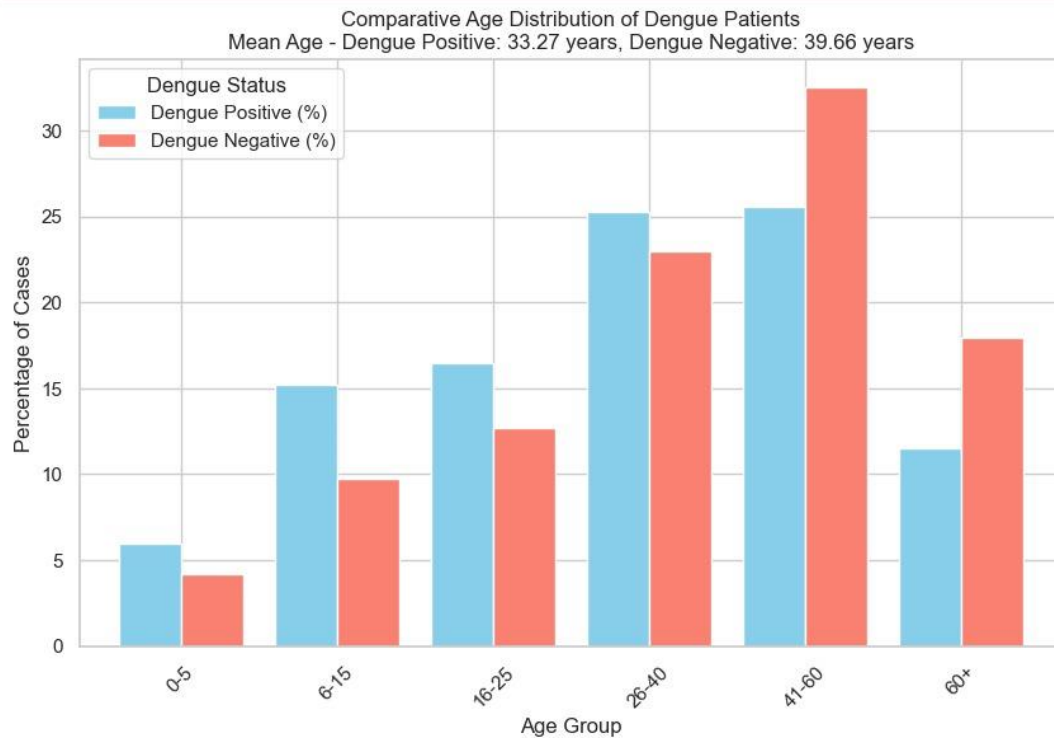


**Figure 6:** Count of Dengue Positive Cases



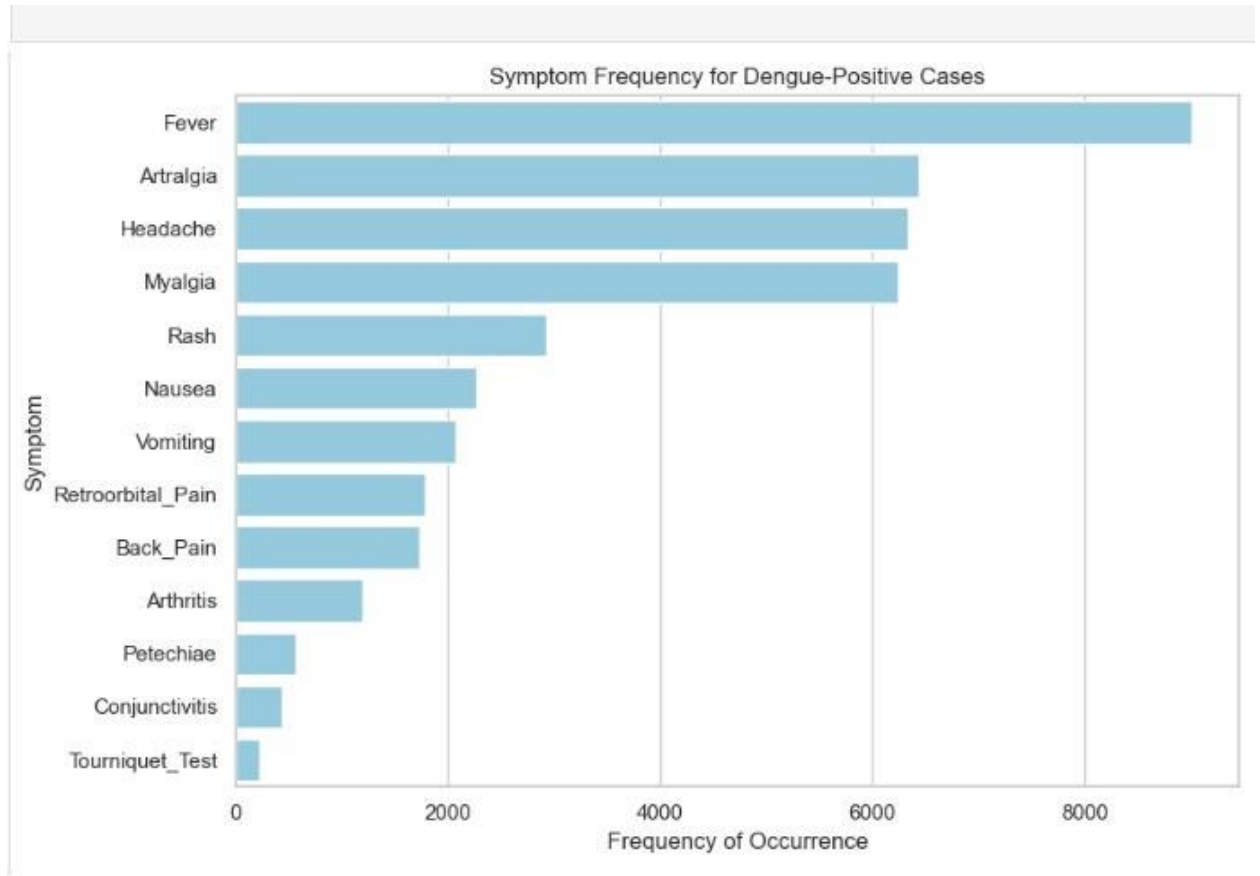
**Figure 7:** Count of Dengue cases of Male and Female

We have seen that the highest number of patients found in the 26-60 years range where the mean age of dengue patients is 33 years. It is proven that children and teenagers are less likely to be affected by Dengue.



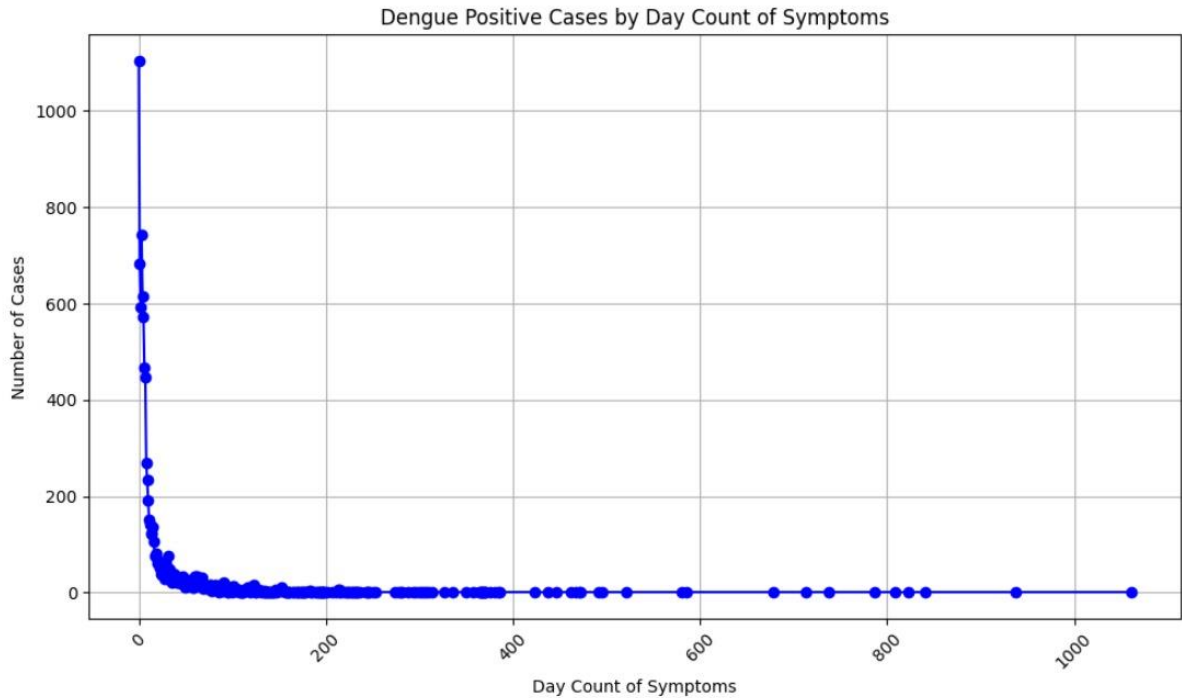
**Figure 8:** Age variation of Dengue patients

After analyzing the symptoms for the dengue positive cases, we have seen that 91% patients were affected with fever. 65% patients were impacted with Arthralgia which means pain in several joints on the body. 64% patients were diagnosed with headache where a similar number of patients were affected with Myalgia (muscle pain). Rashes, nausea and vomiting were also found in approximately 20% of the patients. Thus we have come to the conclusion that these are some of the severe symptoms of Dengue.



**Figure 9:** Frequency of Occurrence

As Dengue symptoms change with the passing number of days, the highest number of Dengue patients (1103 patients) were seen on Day 0. After passing quite a several days, the Dengue symptoms start to change their nature, thus it get even more unpredictable to detect the disease.



**Figure 10: Day Count Symptoms**

#### **4.1.3 Decision making:**

We have implemented the model to get the best output with different classifiers, some of them are below:

- 1) Logistic Regression: Logistic regression is used for binary classification where we use sigmoid function that takes input as independent variables and produces a probability value between 0 and 1.
- 2) Support Vector Classifier (SVC): SVC is best suited for classification tasks. The primary objective of the SVM algorithm is to identify the optimal hyperplane in an N-dimensional space that can effectively separate data points into different classes in the feature space.
- 3) Decision Tree (DT): DT is a supervised machine learning algorithm. The features of the dataset are represented by the internal nodes whereas the leaf node represents the outcome, thus the branches of the tree represent the decision rules.
- 4) Gradient Boosting Classifier: This classifier combines several weak learners into strong learners, in which each new model is trained to minimize the loss function.

5) Random Forest (RF): RF is an ensemble learning algorithm that combines multiple DTs from a randomly selected subset of the training set and for prediction it depends on the votes from different DTs.

Classifier Name	Accuracy
Logistic Regression	72.63%
SVC	63.23%
Decision Tree Classifier	67.17%
Gradient Boosting Classifier	75.35%
Random Forest Classifier	76.03%

**Table 3:** Accuracy based of different types of Models

To choose an algorithm which is best fit to the project, we have run all the models mentioned above on the software and we have seen that the Random Forest Classifier gives us the best output with the highest accuracy (76.03% accuracy). The reason for which we have chosen Random Forest Classifier is:

i) **High Accuracy:** Using several decision trees, each trained on a distinct subset of the data, Random Forest aggregates their predictions. Random Forest lessens the variation associated with individual trees, resulting in predictions that are more accurate, by averaging (for regression) or voting (for classification) the predictions of these trees. When using an ensemble approach instead of a single decision tree model, accuracy is typically higher.

ii) **Robustness to Noise:** As Random Forest combines the forecasts of several decision trees, it is resilient to noisy data. Because noisy data points are unlikely to alter the forecasts of every tree in the forest, they have a lower chance of affecting the overall performance of the model. Random Forest works well with datasets that contain outliers or intrinsic noise because of its robustness.

iii) **Estimating Feature Importance:** Random Forest calculates a feature's importance by taking into account the relative contributions of each feature to the overall variance (for regression) or impurity (for classification) reduction of all the trees in the forest. Features are considered more significant when they regularly result in a larger reduction of impurities or variance. This data can direct feature selection or dimensionality reduction efforts and aid in determining which characteristics are most relevant for prediction.

iv) **Missing Data and Outliers Handling:** Random Forest does not require the use of data preprocessing methods like imputation or outlier removal in order to handle missing data and outliers. Every decision tree is trained using a random subset of the input, and the technique naturally handles missing values. As outliers are unlikely to affect the forecasts of every tree in the forest, they have less of an effect on the performance of the model as a whole.

vi) **Handles Both Numerical and Categorical Data:** Random Forest is capable of handling a combination of numerical and categorical characteristics. The method can handle both types of data without bias since it automatically chooses random subsets of features for each decision tree during training.

#### **4.1.4 Data Normalization**

Source data for Dengue disease was analyzed and thus there were no such parameters identified that are to be normalized before started working with it.

#### **4.1.5 Data Split**

After preprocessing, we applied two approaches to split the dataset into a training set and a testing set after preprocessing stage. The first method is Holdout cross-validation, in which we divided the data set into 70% for training and 30% for testing, and the second method is 10-fold cross-validation. The training data is fed into the machine learning model to train the model. The dengue class (Class 1: Positive, Class 0: Negative) feature is used as the target variable in the prediction classifier.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

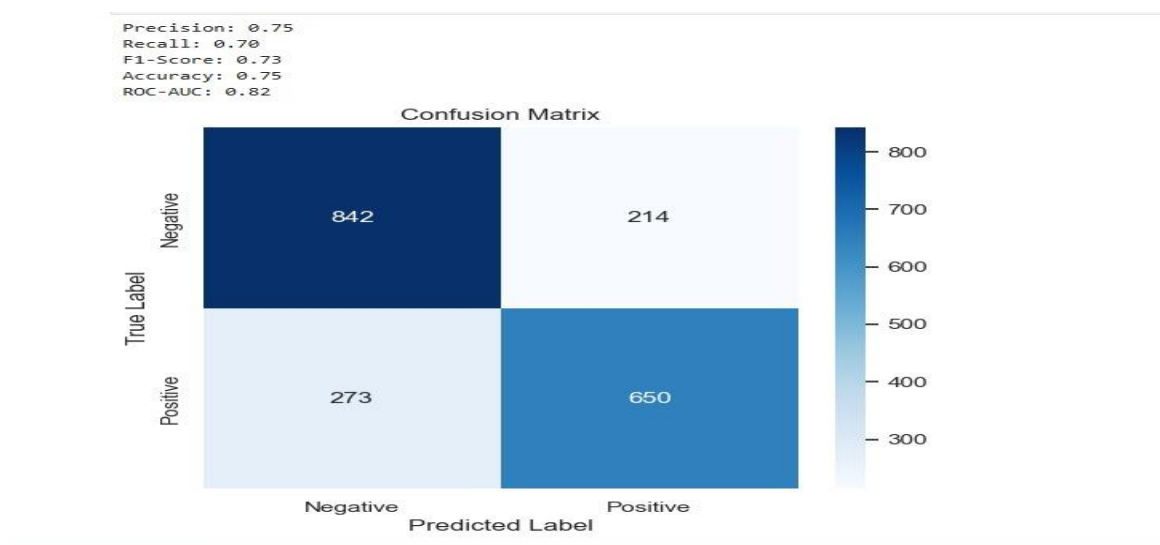
## 4.2 Evaluation metrics

The confusion matrix generally appears as follows for a binary classification (where the classes are “Positive” and “Negative”):

	Predicted Negative	Predicted Positive
Actual Negative	TN	FP
Actual Positive	FN	TP

Where:

- True Negative (TN): Correctly predicted negatives.
- False Positive (FP): Incorrectly predicted positives (actual is Negative, predicted Positive).
- False Negative (FN): Incorrectly predicted negatives (actual is Positive, predicted Negative).
- True Positive (TP): Correctly predicted positives.



**Figure 11:** Evaluation Matrices

Based on the analyzed figure, it seems that there is a mix of values spread across positive and negative labels. Confusion matrix values are as follows:

- True Negative (TN): 842
- False Positive (FP): 214
- False Negative (FN): 273
- True Positive (TP): 650

Thus we can interpret the performance metrics as below:

Precision (0.75):

- Precision measures the proportion of true positive predictions (correctly identified dengue cases) among all positive predictions made by the model.
- With a precision of 0.75, 75% of the predictions the model makes as "dengue positive" are correct. This means the model is relatively good at avoiding false positives, which is crucial for ensuring that people are not incorrectly diagnosed with dengue.

Recall (0.70):

- Recall (or Sensitivity) is the proportion of true positives identified out of all actual positive cases. In this case, it tells us how well the model identifies true dengue cases among all real dengue cases.
- A recall of 0.70 means that the model correctly identifies 70% of actual dengue cases. However, it misses 30% of the cases, which may need improvement if detecting every possible dengue case is crucial.



F1-Score (0.73):

- The F1-Score is the harmonic mean of precision and recall, balancing both metrics to give a single measure of model performance, especially useful when there's an uneven class distribution or when both precision and recall are equally important.
- With an F1-Score of 0.73, the model is reasonably balanced in terms of precision and recall, meaning it performs moderately well at detecting dengue cases without making excessive incorrect positive predictions.

Accuracy (0.75):

- Accuracy measures the proportion of correct predictions (both positive and negative) out of all predictions.
- An accuracy of 0.75 indicates that the model correctly identifies 75% of the cases (both dengue and non-dengue). This is generally a good result but can sometimes be misleading if the dataset is imbalanced. If there are many more non-dengue cases, a high accuracy could result from simply predicting non-dengue more often.

ROC-AUC (0.82):

- ROC-AUC (Receiver Operating Characteristic - Area Under Curve) measures the model's ability to distinguish between the positive and negative classes across all possible classification thresholds. An ROC-AUC score of 0.82 suggests the model has a strong discriminatory ability.
- With a score of 0.82, your model is quite effective at distinguishing between dengue and non-dengue cases, with a high probability of correctly ranking a random positive case higher than a random negative case.

Summary Interpretation

- **High True Negatives (842)** and **True Positives (650)** indicate the model is effective at correctly classifying both negative and positive cases.

- **False Positives (214)** and **False Negatives (273)** suggest that some misclassifications occur, with slightly more false negatives, meaning some actual dengue cases are missed.
- The model's precision and recall are balanced at around 0.75 and 0.70, making it reasonably reliable for both detecting positive cases and minimizing false positives.

## 4.3 Machine Learning Methods

### 4.3.1 Hyper-Parameters settings of classification methods:

No.	Model	Hyper-parameters settings
1	Random Forest Algorithm	<pre>rf_model = RandomForestClassifier(   n_estimators=100,   criterion='gini',   max_depth=None,   min_samples_split=2,   min_samples_leaf=1,   max_features='auto',   bootstrap=True,   random_state=42,   n_jobs=-1,   class_weight=None,   oob_score=False )</pre>

**Table 4:** Hyper-Parameters settings

For the Random Forest algorithm used in predicting dengue disease, we made specific choices and configurations for its hyper-parameters:

**n\_estimators:** The number of trees in the forest.

**criterion:** The function used to measure the quality of a split. 'gini' for Gini impurity or 'entropy' for information gain.

**max\_depth:** The maximum depth of the trees. None means nodes are expanded until they contain less than min\_samples\_split samples.

**min\_samples\_split:** The minimum number of samples required to split an internal node.

**min\_samples\_leaf:** The minimum number of samples required to be at a leaf node.

`max_features`: The number of features to consider when looking for the best split.

`bootstrap`: Whether to bootstrap samples when building trees.

`random_state`: Seed for random number generation.

`n_jobs`: The number of jobs to run in parallel for both fit and predict.

`class_weight`: Weights associated with classes. Useful for handling imbalanced datasets.

`oob_score`: Whether to use out-of-bag samples to estimate the generalization accuracy.

By setting these parameters, we are configuring the behavior of the Random Forest model.

#### **4.4 Model implementation and UI interaction:**

To integrate a machine learning model with a front-end application, here's the full process from training to deployment with an API as an intermediary layer:

##### **1. Train and save the model**

We are saving the model as a .pkl file .

This file Will be loaded in the API backend for inference.

##### **2. Setup and API Backend**

Using Django is common for creating an API that the front-end can interact with .


This API will load the model and exposed endpoints for making predictions.

##### **3. Front end Setup:**

The frontend we built with Vue js to communicate with API by sending data in Json format

**4. Testing the Application:** Enters values for each symptoms and submit the form. The JS sends the input data to the Django API, received the predictions and displays it on the webpage.

## Application Interface:

[Overview](#) [FAQ](#) [Settings](#)

Dengue Disease Prediction Tool (DDPT)

Personal Information

Patient Name:

Gender:  
☐ Male ☒ Female

Age (year):

Patient Race:

Residence Area:

Clinical Symptom

Fever:  
☒ Yes ☐ No

Myalgia:  
☒ Yes ☐ No

Headache:  
☒ Yes ☐ No

Rash:  
☐ Yes ☒ No

Vomiting:  
☐ Yes ☒ No

Nausea:  
☒ Yes ☐ No

Back Pain:  
☐ Yes ☒ No

Conjunctivitis:  
☐ Yes ☒ No

Arthritis:  
☐ Yes ☒ No

Artralgia:  
☒ Yes ☐ No

Petechiae:  
☐ Yes ☒ No

Tourniquet Test:  
☐ Yes ☒ No

Retroorbital Pain:  
☐ Yes ☒ No

Pre-existing Disease

Diabetes:  
☐ Yes ☒ No

Hematological Disease:  
☐ Yes ☒ No

Liver Disease:  
☐ Yes ☒ No

Kidney Disease:  
☐ Yes ☒ No

Hypertension:  
☐ Yes ☒ No

Peptic Acid Disease:  
☐ Yes ☒ No

Auto Immune Disease:  
☐ Yes ☒ No

Other Information

Day Count of Symptoms:

Predict

### Test output for Dengue positive:

The screenshot shows a web application interface for dengue prediction. A modal window titled "Dengue Prediction result" is displayed in the center. The modal contains the following text: "Hello Bipros Bhowmik Joy !!", "Machine predicted Dengue Positive !! 🤖", and "Suggestion for Hospitalization." Below the text are "Cancel" and "OK" buttons. The background shows a form with sections for "Pre-existing Disease" (Diabetes, Hypertension, Hematological Disease, Liver Disease, Kidney Disease) and "Other Information" (Day Count of Symptoms). The "Day Count of Symptoms" field is filled with the value "15". A "Predict" button is visible at the bottom of the form.

### Test output for Dengue Negative:

The screenshot shows the same web application interface as the previous one, but with a different user and symptom count. The modal window titled "Dengue Prediction result" displays: "Hello Sum Majumder !!", "Machine predicted Dengue Negative !! 🤖", and "Cancel" and "OK" buttons. In the background, the "Day Count of Symptoms" field is filled with the value "81". The "Predict" button is at the bottom.

## **Chapter 5: Conclusion and future work**

Dengue infection is a global problem today. The early detection and prevention of dengue can help to avoid complications and save human lives. In this paper, we proposed a framework for dengue prediction and evaluated the performance of five machine learning models for predicting dengue (Logistic Regression, SVC, Decision Tree Classifier, Gradient Boosting Classifier, Random Forest Classifier).

In the initial stage of the work, namely the pre-processing stage, the missing values were processed by the mean method. The selection features technique was applied to select the important features. The data were normalized previously. After that machine learning models were built, and their performance was evaluated using accuracy, F1-score, precision, recall and ROC-AUC scores. The experimental results show that the performance of the dengue prediction system is improved with 76.03% accuracy followed by 73% f1-score, 75% precision, 70% recall and 82% ROC-AUC score.

We have built a software where patients can input their symptoms and the software will produce the result based on the training and testing data. The proposed software is very much essential for early detection to cure Dengue Fever. This software can save both time and money and also the high risk of DF at an early stage. It is for just preliminary checking for the fever along with some other symptoms if a person is having Dengue Fever so that he or she can contact the doctor and undergo pathological testing.

In future work, we plan to adopt the atypical Dengue symptoms and mandatory laboratory test results like NS1 antigen, IgM, IgG test to increase the accuracy of the prediction. Besides this, we will adopt the latest Machine learning algorithms for increasing system performance and techniques for fetching the users' input to this application so that the application will be easy to run and user friendly.

## Chapter 6: References

- [1] T. Marimuthu, V. J. I. J. o. C. E. Balamurugan, and Technology, "A Novel Bio-Computational Model for Mining the Dengue Gene Sequences," *International Journal of Computer Engineering & Technology*, Article vol. 6, no. 10, pp. 17-33, 2015.
- [2] N. K. Rao, G. S. Varma, N. Rao, P. J. I. J. o. R. i. C. Cse, and C. Technology, "Classification rules using decision tree for dengue disease," *International Journal of Research in Computer and Communication Technology*, Article vol. 3, no. 3, pp. 340-343, 2014.
- [3] V. Krishnaiah, G. Narsimha, N. S. J. I. J. o. C. S. Chandra, and I. Technologies, "Diagnosis of lung cancer prediction system using data mining classification techniques," *International Journal of Computer Science and Information Technologies*, Article vol. 4, no. 1, pp. 39-45, 2013.
- [4] K. Shaukat, N. Masood, A. B. Shafaat, K. Jabbar, H. Shabbir, and S. J. a. p. a. Shabbir, "Dengue fever in perspective of clustering algorithms," in *2017 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*, pp. 1-5. IEEE
- [5] N. A. Husin and N. Salim, "Modeling of dengue outbreak prediction in Malaysia: a comparison of neural network and nonlinear regression model," in *2008 International Symposium on Information Technology*, 2008, vol. 3, pp. 1-4: IEEE.
- [6] N. Subitha, A. J. I. J. o. C. T. Padmapriya, and Technology, "Diagnosis for dengue fever using spatial data mining," *International Journal of Computer Trends and Technology*, Article vol. 4, no. 8, pp. 2646-2651, 2013.
- [7] D. Sarma, S. Hossain, T. Mittra, M. A. M. Bhuiya, I. Saha and R. Chakma, "Dengue Prediction using Machine Learning Algorithms," *2020 IEEE 8th R10 Humanitarian Technology Conference (R10-HTC)*, Kuching, Malaysia, 2020, pp. 1-6, doi: 10.1109/R10-HTC49770.2020.9357035.
- [8] Dourjoy, Saif Mahmud & Rafi, Abu Mohammed Golam Rabbani & Tumpa, Zerine & Saifuzzaman, Mohd. (2021). A Comparative Study on Prediction of Dengue Fever Using Machine Learning Algorithm. 10.1007/978-981-15-4218-3\_49.

- [9] Paul KK, Macadam I, Green D, Regan DG, Gray RT. Dengue transmission risk in a changing climate: Bangladesh is likely to experience a longer dengue fever season in the future. *Environ Res Lett*. 2021; 16: 114003. <https://doi.org/10.1088/1748-9326/ac2b60>
- [10] Hossain MP, Zhou W, Ren C, Marshall J, Yuan H-Y. Prediction of dengue annual incidence using seasonal climate variability in Bangladesh between 2000 and 2018. *PLOS Global Public Health*. 2022; 2:e0000047. <https://doi.org/10.1371/journal.pgph.0000047>
- [11] Dey SK, Rahman M.M, Howlader A, Siddiqi UR, Uddin KMM, Borhan R, et al. (2022) Prediction of dengue incidents using hospitalized patients, metrological and socio-economic data in Bangladesh: A machine learning approach. *PLoS ONE* 17(7): e0270933. <https://doi.org/10.1371/journal.pone.0270933>
- [12] L. Tanner, M. Schreiber, J. G. H. Low, A. Ong, T. Tolfvenstam, Y.L.Lai, L. C. Ng, Y. S. Leo, L. T. Puong, S. G. Vasudevan, C. P. Simmons, M. L. Hibberd, and E.E. Ooi, "Decision Tree Algorithms Predict the Diagnosis and Outcome of Dengue Fever in the Early Phase of Illness", *PLoS Negl Trop Dis*, vol. 2(3), 2008.
- [13] N.Ahmed, A. Ishaq, M. Shoaib, and A. Wahab, "Role of Expert Systems in Identification and Overcoming of Dengue Fever", *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 8, no.10, 2017.
- [14] N. A. Husin, A. S. N. Al-Harogi, N. Mustapha, H. Hamdan, and U. A. Husin, "Fuzzy Rules Base System for EarlySelf-Diagnosis of Dengue Symptoms", *International Journal of Engineering & Technology*, vol. 7 (4.19), pp.458-463, 2018.
- [15] K. Phakhounthong, P. Chaovalit, P. Jittamala, S. D.Blacksell, M. J. Carter, P. Turner, K. Chheng, S.Sona, V. Kumar, N. P. J. Day, L. J. White and W. Pan-Ngum, "Predicting the severity of dengue fever in children on admission based on clinical features and laboratory indicators: application of classification tree analysis", *BMC Pediatrics*, vol. 18(1), 2018.
- [16] A. N. Boruah, S. Kumar Biswas, S. Bandyopadhyay and S. Sarkar, "Expert System to Manage Parkinson Disease by Identifying Risk Factors: TD-Rules-PD," 2020 International Conference on Computational Performance Evaluation (ComPE), 2020, pp. 001-006, doi: 10.1109/ComPE49325.2020.9200075.



- [17] R. Gangula, L. Thirupathi, R. Parupati, K. Sreeved and S. Gattoju, “Ensemble machine Learning Based prediction of Dengue Disease with performance and Accuracy Elevation Patterns”, Materialstudy Proceedings , <https://doi.org/10.1016/j.matpr.2021.07.270>
- [18] W. Hoyos, J. Aguilar and M. Toro, “Dengue Models Based on Machine Learning Techniques: A Systematic Literature Review”, Artificial Intelligence in Medicine , Vol. 119, 2021. <https://doi.org/10.1016/j.artmed.2021.102157>
- [19] P. Silitonga, B. E. Dewi, A. Bustamam and H. ShaoriAl-Ash, “Evaluation of Dengue Model Performances Developed Using Artificial Neural Network and Random Forest Classifiers”, Procedia Computer Science , vol. 179, pp. 135-143, 2021. <https://doi.org/10.1016/j.procs.2020.12.018>
- [20] M. K. Hasan, M. A. Alam, D. Das, E. Hossain and M. Hasan, "Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers," in IEEE Access , vol. 8, pp. 76516-76531, 2020, doi: 10.1109/ACCESS.2020.2989857.
- [21] Tabosa, Thomás; Silva Neto, Sebastião; Teixeira, Igor; Oliveira, Samuel; Rodrigues, Maria Gabriela; Sampaio, Vanderson; Endo, Patricia (2021), “Clinical cases of Dengue and Chikungunya”, Mendeley Data, V1, doi: 10.17632/bv26kznkjs.1