

**Daffodil International University**

Project Report

Course Name: Big Data and IOT Lab

Course Code: CSE413

Submitted to

**Amir Sohel**

Senior Lecturer

Department of Computer Science and Engineering

Daffodil International University

Submitted by

|  |  |
| --- | --- |
| **Name** | **ID** |
| Tahmid Shihab (Report) | 201-15-13667 |
| Md Mahamud Mredha (Code) | 201-15-13756 |

**Title:** Risk Prediction of Cardiovascular Disease Using Machine Learning Classifiers.

**Introduction**

Health is a crucial part of everyone’s life. Nonetheless, due to a variety of causes including unhealthy lifestyles, work stress, psychological strain, and external factors such as pollution, hazardous work environment, and lack of proper health services, a significant number of individuals across the globe succumb to enduring illnesses such as cardiovascular disorders (CVD), which impact both the cardiac organ and the circulatory system, leading to fatality or incapacitation. In recent years, it was reported that the majority of human deaths were due to CVD [[1](Cardiovascular%20diseases%20(CVDs).%20http:/www.who.int/newsroom/factsheets/detail/cardiovascular-diseases-(cvds%20accessed%20on%2030/9/2018.),2]. The associated conditions are hypertension, thromboembolism, hyperlipidemia, and coronary heart disease, which culminate in heart failure. Hypertension is the primary cause of CVD [3]. In 2012, 7.4 million people were reported to have died from coronary heart disease, while 6.7 million people died from stroke [4]. The World Health Organization estimates that nearly 17 million people die every year from CVDs, which accounts for approximately 31% of global deaths. Early diagnosis of CVD can potentially cure patients and save innumerable lives. Diagnosis and treatment of patients at early stages by cardiologists remain a challenge. Every traditional CVD risk-assessment model implicitly assumes each risk factor related to CVD outcome linearly. Such models tend to oversimplify complex relationships, including several risk factors with non-linear interactions. The goal of this study is to determine if ML can enhance cardiovascular risk prediction accuracy in population primary care at large and find out which ML algorithm result had fairly high brevity. In recent years, multiple ML-based CVD detection models have been proposed. A review of previous studies is presented to identify the research problem and objective of each study. ML helps a cardiologist to predict diseases at an early stage and treat the patient accordingly. There are many ML techniques such as support vector machines [5], artificial neural networks, decision trees [6], and K-Nearest Neighbor (K-NN) [7], each with its strengths and weaknesses. These methods have been applied in broader areas like predicting liver [8,9], human heart (echocardiogram signals) [10,11], and skin diseases [12,13,14]. The results of each technique differ owing to several constraints. This study contains an in-depth statistical analysis of input data sets to understand the effects of data range on the CVD predictions. It includes a correlation study of categorical and continuous features of patients. In addition, data visualization and scatter plots for pairs of important features were obtained to understand the significance of the correlation between important features. These are discussed and analyzed in the results section.

**Methodology**

Cardiovascular disease (CVD) makes our heart and blood vessels dysfunctional and often leads to death or physical paralysis. Therefore, early and automatic detection of CVD can save many human lives.

Prediction of cardiovascular disease is regarded as one of the most important subjects in the section of clinical data analysis. Data mining turns the large collection of raw healthcare data into information that can help to make informed decisions and predictions. Some existing studies applied data mining techniques in heart disease prediction.

We aim to identify significant features and data mining techniques that can improve the accuracy of predicting cardiovascular disease. Prediction models were developed using different combinations of features, and four classification techniques: Decision Tree, Naive Bayes, Random Forest, and Logistic Regression.

**1. Proposed Model**

**The following steps are carried out to build our model to predict CVD:**

**Step 1:** At first, the CVD data set is collected from Kaggle

**Step 2:** Data samples are pre-processed by eliminating null values, range value handling with mid-point, label encoding, and one hot encoding.

**Step 3:** Feature selection of the column on which the target value is more dependent.

**Step 4:** Multiple effective algorithms are chosen to classify the selected features based on Figure 1.

**Step 5:** Various performance measures are evaluated to compare and find the better method.

**Data**

**Data**

**Preprocessing**

**Feature Selection**

**Predicted**

**Class**

**Training of Model**

**Desired**

**Class**

**+**

**-**

**Training Algorithms**

Figure 1: The training phase

**Predicted**

**Class**

**Trained Model**

**Features for Testing**

Figure 2: The testing phase

The methodology employed in developing the CVD detection model:

(1) The training phases.

(2) The testing phase.

**2. Dataset Description**

**Dataset:** Cardiovascular Diseases Risk Prediction Dataset.

The dataset that we are using includes various features related to patients’ health and lifestyle. Each row represents a unique patient and includes attributes such as age, sex, general health, checkup frequency, exercise habits, and smoking history.

We have collected this dataset of the “Cardiovascular Diseases Risk Prediction Dataset” from “Kaggle” which provides comprehensive insights into the lifestyle factors of a person that can contribute to being at risk with any form of cardiovascular disease.

Our dataset has a total of **19 columns** containing **308855** data having various features related to patients. The target variable is “Heart\_Disease”.

**Dataset source:** <https://www.kaggle.com/datasets/alphiree/cardiovascular-diseases-risk-prediction-dataset>

**Data provider:** <https://www.kaggle.com/alphiree>

**2.2 Column-wise Dataset Description:**

|  |  |  |
| --- | --- | --- |
| **SL** | **Column Name** | **Description** |
| 1 | General\_Health | This column likely represents the general health status of each individual. It seems to have categories such as "Poor" and "Very Good" indicating different levels of health. |
| 2 | Checkup | This column indicates the timing of the last general health checkup for each individual. It has values like "Within the past year" and "Within the past 2 years." |
| 3 | Exercise | This column appears to indicate whether the individual engages in regular exercise. It has values like "Yes" and "No." |
| 4 | Heart\_Disease | Indicates whether an individual has a history of heart disease (values "Yes" or "No"). |
| 5 | Skin\_Cancer | Indicates whether an individual has a history of skin cancer (values "Yes" or "No"). |
| 6 | Other\_Cancer | Indicates whether an individual has a history of other types of cancer (values "Yes" or "No"). |
| 7 | Depression | Indicates whether an individual has a history of depression (values "Yes" or "No"). |
| 8 | Diabetes | Indicates whether an individual has a history of diabetes (values "Yes" or "No"). |
| 9 | Arthritis | Indicates whether an individual has a history of arthritis (values "Yes" or "No"). |
| 10 | Sex | Gender of the individual (values "Female" or "Male"). |
| 11 | Age\_Category | Age group or category to which the individual belongs (e.g., "70-74," "60-64," "80+"). |
| 12 | Height\_(cm) | Height of the individual in centimeters. |
| 13 | Weight\_(kg) | Weight of the individual in kilograms. |
| 14 | BMI | Body Mass Index (BMI) of the individual. BMI is typically calculated using height and weight and is a measure of body fat. |
| 15 | Smoking\_History | Indicates the individual's smoking history (values like "Yes" or "No"). |
| 16 | Alcohol\_Consumption | Indicates the individual's alcohol consumption status (values like "0" for no consumption or other values indicating the frequency or quantity). |
| 17 | Fruit\_Consumption | Indicates the individual's fruit consumption habits (likely in some quantifiable measure). |
| 18 | Green\_Vegetables\_Consumption | Indicates the individual's consumption of green vegetables. |
| 19 | FriedPotato\_Consumption | Indicates the individual's consumption of fried potatoes (likely in some quantifiable measure). |

**3. Data Pre-Processing**

Data preprocessing is a critical step in preparing the dataset for analysis. It involves cleaning, transforming, and organizing the data to ensure that it is suitable for modeling and analysis. Here's a step-by-step process of data processing for this project:

**3.1 Null Value Identification**

The null value identification process identifies and locates missing or null values of a dataset using methods for checking null entries with functions such as **isnull()** and visualizing missing value patterns.

**3.2 Null Value Handling**

Null value handling is the removal of affected rows/columns or imputations with estimated values such as mean and median values to replace missing data, ensuring the dataset is suitable for analysis or machine learning.

However, in our dataset, there are no missing or null values so we do not need to handle the missing or null values.

**3.3 Encoding**

Encoding converts categorical data into a numerical format so that computers can understand and process them. It helps convert words or labels into a format that machine learning models can use for analysis.

There are many encoding techniques and among them, we used **One-Hot Encoding** and **Label Encoding** as these are required for our dataset.

**3.3.1 Label Encoding**

Label Encoding is the method of converting categorical data into numeric form by assigning a unique numerical label to each category, allowing machine learning models to process and analyze the information. It provides an ordinal representation where the numeric values indicate the order or ranking of the categories.

In our dataset, the following columns have categorical data and those columns are:

**'Exercise', 'Heart\_Disease', 'Skin\_Cancer', 'Other\_Cancer', 'Depression', 'Diabetes', 'Arthritis', 'Sex' and 'Smoking\_History'**.

So, to convert them into machine-readable numerical data we did Label Encoding for these columns.

**3.3.2 OneHot Encoding**

OneHot encoding is performed to represent categorical data with multiple categories in a binary format. This technique ensures that each category is distinctly captured, preventing the model from misinterpreting ordinal relationships.

In our dataset, the following columns have categorical data with multiple categories and those columns are: **'General\_Health', and 'Checkup'**.

So, to convert them into machine-readable numerical data we did OneHotEncoding for these two columns.

**3.4 Correlation Matrix**

A correlation matrix is a square matrix that quantifies the strength and direction of linear relationships between pairs of variables in a dataset. Each entry in the matrix represents the correlation coefficient between two variables, with values ranging from -1 (perfect negative correlation) to 1 (perfect positive correlation), and 0 indicating no correlation. Feature Engineering

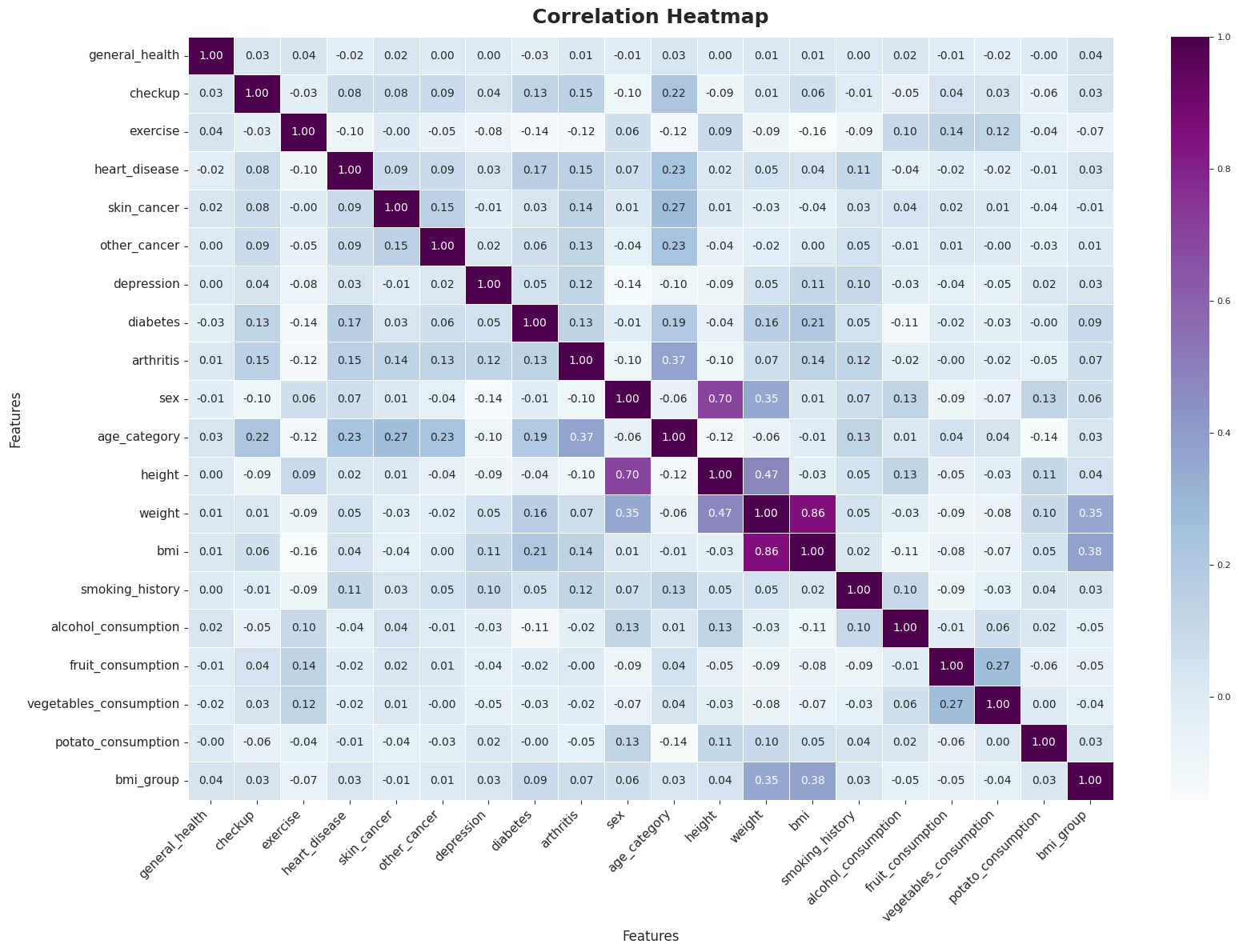


Figure 3: Correlation Heatmap

**4. Dataset Splitting**

Dataset splitting is the process of dividing a dataset into multiple subsets for training, validation, and testing purposes. The typical splits include:

**Training Set:** The largest portion of the dataset is used to train the machine learning model. The model learns patterns and relationships from this set.

**Validation Set:** A subset used to fine-tune the model's hyperparameters and evaluate its performance during training. It helps in preventing overfitting of the training data.

**Test Set:** A separate portion of the dataset that the model has never seen during training or validation. It is used to assess the model's generalization performance on new, unseen data.

For our dataset, we have split the dataset into an 80-20 ratio where 80% of the dataset is for training and the rest 20% is for validation and testing.

**5. Standard Scaler**

Standard Scaler is a technique that transforms the values of features to have a mean of 0 and a standard deviation of 1, ensuring that all features are on a comparable scale. It helps improve the performance and convergence of certain algorithms sensitive to feature scales.

**6. Model Selection**

Model selection is the process of choosing the best machine-learning model for a specific task or dataset to make accurate predictions.

For our dataset, we have used the **Logistic Regression Model, Decision Tree Model Random Forest Classifier Model,** and **Naive Bayes Model** to make more accurate and reliable predictions.

**6.1 Decision Tree Model**

A Decision Tree model is a predictive algorithm that recursively splits the data into subsets based on the most significant features, forming a tree-like structure. It is widely used for classification and regression tasks in machine learning.

**6.2 Naive Bayes Model**

The Naive Bayes model is a probabilistic classification algorithm based on Bayes' theorem. It assumes independence between features, making it computationally efficient and often used for text classification and spam filtering.

**6.3** **Random Forest Classifier Model**

The Random Forest Classifier is an ensemble machine learning model that constructs multiple decision trees during training and outputs the mode of the classes for classification tasks, offering high accuracy and robustness by combining predictions from multiple trees.

**6.4 Logistic Regression Model**

Logistic Regression is a statistical model used for binary classification, predicting the probability of an instance belonging to a particular class based on input features. It employs the logistic function to transform a linear combination of features into a probability score.

**7. Result Discussion and Analysis**

Result Discussion and Analysis involve interpreting and elaborating on the outcomes obtained from experiments or analyses, emphasizing their significance, patterns, and implications to draw meaningful conclusions or insights.

**7.1 Performance Measurement Matrix**

Performance measurement metrics in the context of datasets refer to quantitative measures used to evaluate the quality, characteristics, and effectiveness of a dataset. Here are some common performance measurement metrics that are being used for our datasets:

**7.1.1 Accuracy:**

Accuracy is commonly used as an initial performance metric to assess the overall assessment of a model's performance, with higher accuracy indicating better performance.

The formula for accuracy is: **Accuracy =**

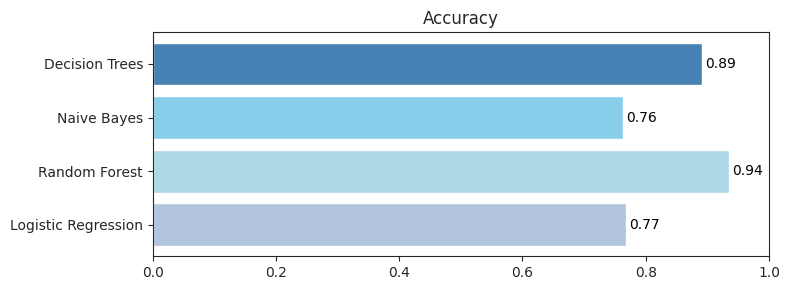


Figure 3: Accuracy Comparison

For each algorithm in this model, the accuracy score is shown in Figure 3 and in Table 11.

Random Forest got the highest accuracy score of 0.94 (94%) among the four algorithms evaluated for accuracy, followed by Decision Tree and Logistic Regression with accuracy scores of 0.89 (89%) and 0.77 (77%) respectively. While Naive Bayes obtained the lowest accuracy score of 0.76 (76%).

**7.1.2 Precision:**

Precision measures the ratio of correctly predicted positive instances (TP) to all instances predicted as positive (TP + FP). It quantifies how many of the positive predictions made by the model were correct.

The formula for precision is: **Precision =**

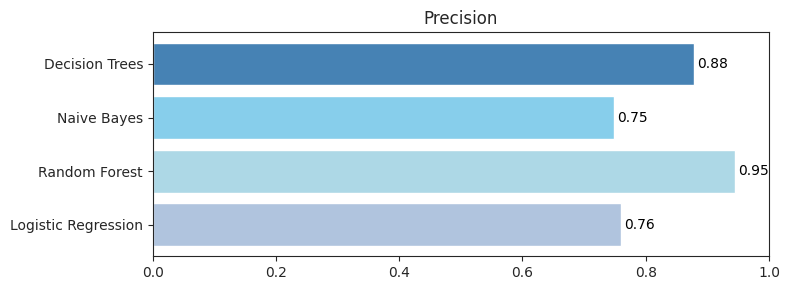


Figure 4: Precision Comparison

For each algorithm in this model, the precision score is shown in Figure 4 and in Table 11.

Random Forest got the highest accuracy score of 0.95 (95%) among the four algorithms evaluated for accuracy, followed by Decision Tree and Logistic Regression with accuracy scores of 0.88 (88%) and 0.76 (76%) respectively. While Naive Bayes obtained the lowest precision score of 0.75 (75%).

**7.1.3 Recall (Sensitivity):**

It is also similar to the Precision metric; however, it aims to calculate the proportion of actual positive that was identified incorrectly Recall measures the ratio of correctly predicted positive instances (TP) to all actual positive instances (TP + FN). It quantifies the model's ability to identify all instances of the positive class.

The formula for recall is: **Recall =**

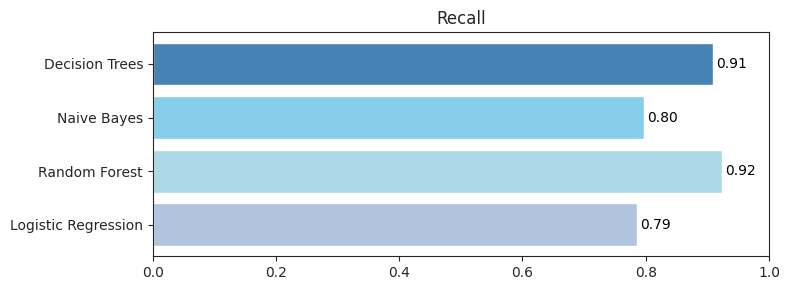


Figure 5: Sensitivity Comparison

For each algorithm in this model, the precision score is shown in Figure 5 and in Table 11. Random Forest got the highest accuracy score of 0.92 (92%) among the four algorithms evaluated for accuracy, followed by Decision Tree and Naive Bayes with accuracy scores of 0.91 (91%) and 0.80 (80%) respectively. While this time Logistic Regression obtained the lowest recall score of 0.79 (79%).

**7.1.4 F1 Score:**

F1 Score can is the harmonic mean of both precision and Recall. The formula for recall is: **F1 Score =**

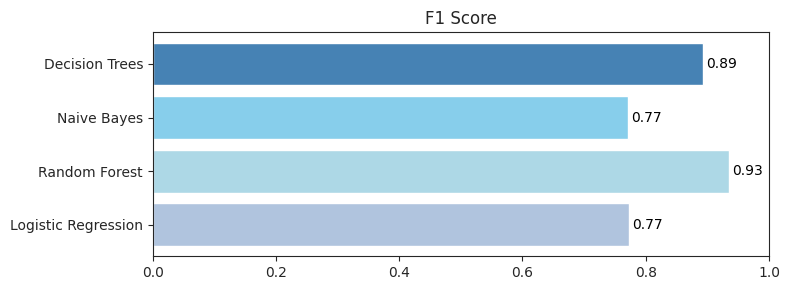


Figure 6: F1 Score Comparison

For each algorithm in this model, the F1 score is shown in Figure 6 and in Table 11.

Random Forest got the highest F1 score of 0.93 (93%) among the four algorithms evaluated for accuracy, followed by Decision Tree with accuracy scores of 0.89 (89%). While both Naive Bayes and Logistic Regression obtained the lowest F1 score of 0.77 (77%).

**7.1.5 MSE (Mean Squared Error)**

MSE is a regression metric that measures the average squared difference between predicted and actual values. Lower MSE indicates better model performance.

The formula for MSE is: **MSE** **=**

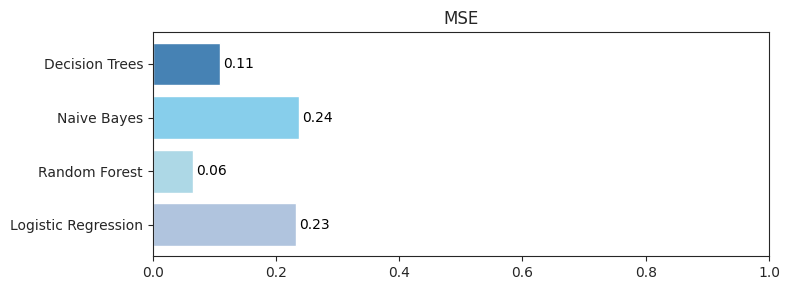


Figure 7: MSE Score Comparison

For each algorithm in this model, the MSE score is shown in Figure 7.

Random Forest got the best MSE score of 0.06 among the four algorithms evaluated for accuracy, followed by Decision Tree with MSE scores of 0.11. On the other hand Decision Tree and Logistic Regression obtained scores of 0.24 and 0.23 respectfully.

**7.1.6 RMSE (Root Mean Squared Error)**

RMSE is the square root of MSE, providing a measure of the average magnitude of the errors in the same units as the target variable.

The formula for RMSE is: **RMSE** **=**

For each algorithm in this model, the RMSE score is shown in Figure 8.

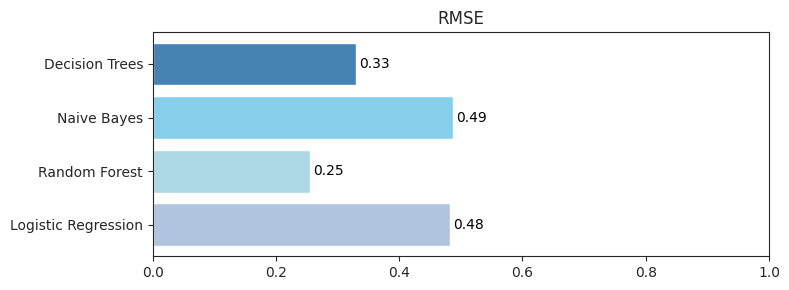


Figure 8: RMSE Score Comparison

For each algorithm in this model, the RMSE score is shown in Figure 7.

Random Forest got the best RMSE score of 0.25 among the four algorithms evaluated for accuracy, followed by Decision Tree with MSE scores of 0.33. But Decision Tree and Logistic Regression comparatively obtained high RMSE scores of 0.49 and 0.48 respectively.

**7.1.7 Jaccard Score**

The Jaccard score is used in classification tasks to measure the similarity between predicted and actual sets of labels. It ranges from 0 to 1, where 1 indicates perfect overlap. The formula for the Jaccard Score is:

**Jaccard Score** **=**

|  |  |
| --- | --- |
| **Model** | **Jaccard Score** |
| Decision Tree | 0.81 |
| Naïve Bayes | 0.63 |
| Random Forest | 0.88 |
| Logistic Regression | 0.63 |

Figure 8: Jaccard Score Comparison

For each algorithm in this model, the Jaccard score is shown in Figure 8.

Random Forest got the best Jaccard score of 0.88 among the four algorithms evaluated for accuracy, followed by Decision Tree with Jaccard scores of 0.81. But Naïve Bayes and Logistic Regression both obtained the same Jaccard score of 0.63 which is comparatively low from the standard.

**7.1.8 ROC (Receiver Operating Characteristic) Curve**

ROC curve is a graphical representation of the true positive rate against the false positive rate at various threshold settings.

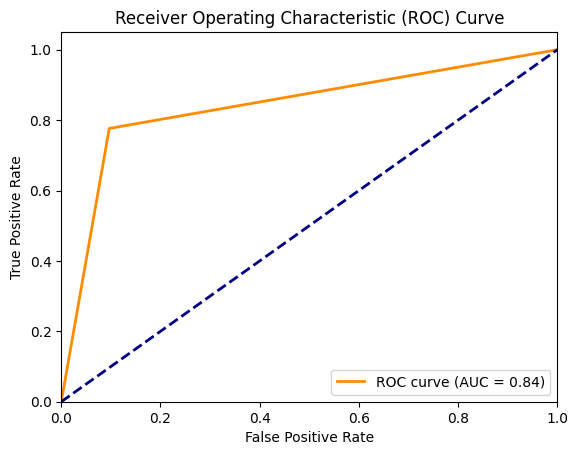
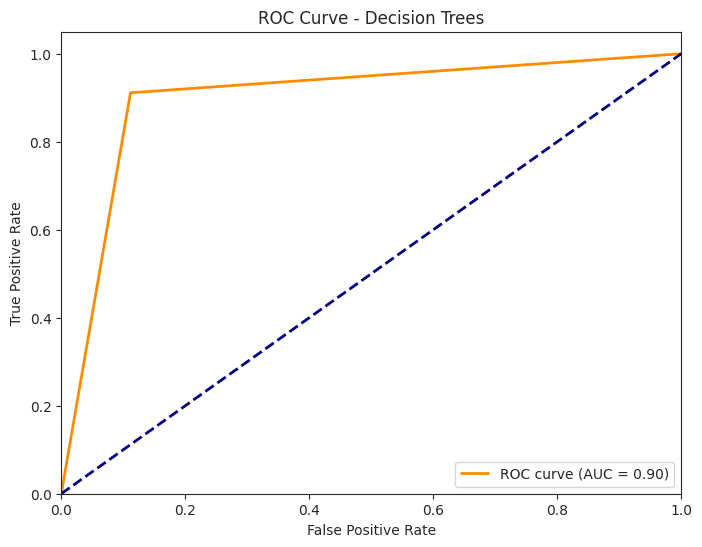


Figure 9.1: Decision Trees Figure 9.2: Naïve Bayes

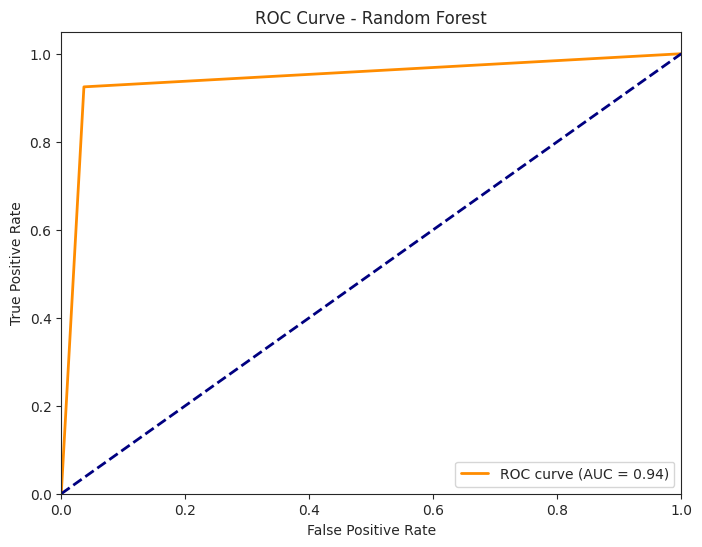
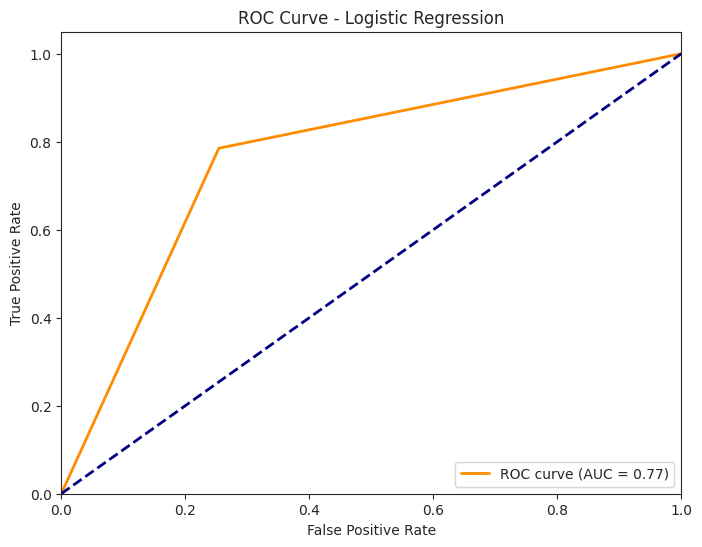


Figure 9.3: Random Forest Figure 9.4: Logistic Regression

For each algorithm in this model, the ROC curve is graphically represented.

Random Forest got the best ROC\_AUC score of 0.94 among the four algorithms evaluated for accuracy.

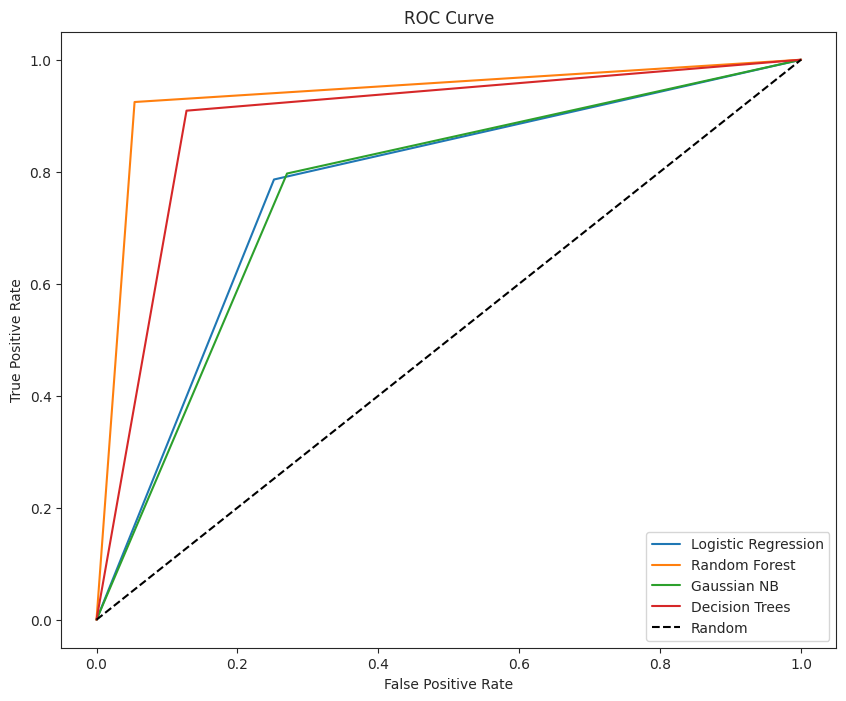


Figure 9.5: ROC Curve

For each algorithm in this model, the ROC comparison is graphically shown in Figure 9.5.

**7.1.9 AUC (Area Under the Curve)**

AUC is a metric that quantifies the area under the ROC curve. AUC ranges from 0 to 1, with higher values indicating better model performance.

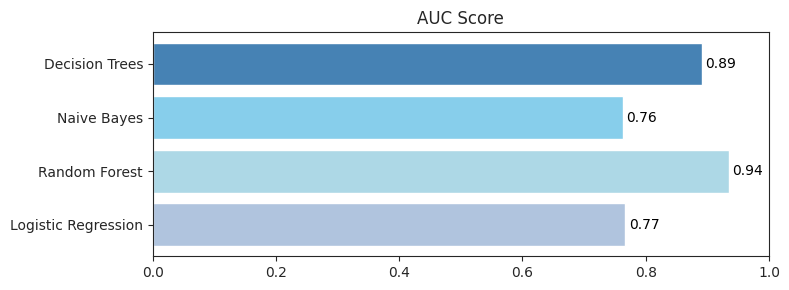


Figure 10: AUC Score Comparison

For each algorithm in this model, the AUC score comparison is graphically shown in Figure 8.

Random Forest got the best AUC score of 0.94 among the four algorithms evaluated for accuracy, followed by Decision Tree with an AUC score of 0.89. Naive Bayes and Logistic Regression both obtained almost the same AUC scores which are 0.76 and 0.77 respectively.

**8. Result Discussion**

Performance Measurement Table summarizes the evaluation metrics of the model, providing insights into its accuracy, precision, recall, F1-score, and other relevant metrics across different classes or categories

**8.1 Performance Measurement Matrix Result Comparison**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Sensitivity** | **F1** |
| Decision Tree | 0.89 | 0.88 | 0.91 | 0.89 |
| Naïve Bayes | 0.76 | 0.75 | 0.80 | 0.77 |
| Random Forest | 0.94 | 0.95 | 0.92 | 0.93 |
| Logistic Regression | 0.77 | 0.76 | 0.79 | 0.77 |

Table11: Result Comparison

After analyzing all the performance measurement matrix results for all four algorithms we found out that Random Forest got the best Accuracy, Precision, Sensitivity, and F1 score among the four algorithms evaluated for our project.

**8.2 Cross-Validation Score**

The cross-validation score is a performance metric indicating how well a model generalizes to new data by averaging evaluation scores across multiple train-test splits, enhancing reliability in assessing its predictive capability.

|  |  |
| --- | --- |
| **Model** | **Cross-Validation Score**  **n = 5** |
| Decision Tree | [0.89606571 0.89448011 0.89428191 0.89522887 0.89437984] |
| Naïve Bayes | [0.76266558 0.76047436 0.7624894 0.7613002 0.76136364] |
| Random Forest | [0.94179504 0.93875596 0.94080403 0.94060583 0.93905259] |
| Logistic Regression | [0.76747745 0.76132222 0.76703701 0.76542938 0.76427061] |

Table12: Cross-Validation Score Comparison

**8.3 Misclassification Error**

Misclassification Error, also known as classification error, measures the rate of incorrectly classified observations in a model compared to the total observations evaluated, often expressed as a percentage or proportion.

|  |  |
| --- | --- |
| **Model** | **Misclassification Error** |
| Decision Tree | 0.10014798893606526 |
| Naïve Bayes | 0.23697609273973327 |
| Random Forest | 0.05601557407374782 |
| Logistic Regression | 0.23435985976286536 |

Table13: Misclassification Error

The algorithm with the lowest Misclassification Error rate performs better as it indicates fewer incorrectly classified observations compared to the total evaluated. Random Forest got the lowest Misclassification Error score of 0.05601557407374782 among the four algorithms evaluated for accuracy.

**8.4 Confusion matrix**

We used Confusion Matrix to generate and display the performance of Logistic Regression, Naive Bayes, Decision Tree, and Random Forest model's predictions on the test dataset using the sklearn.metrics.confusion\_matrix and ConfusionMatrixDisplay.

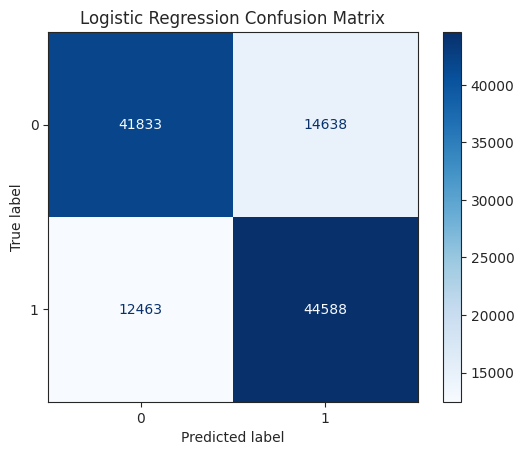
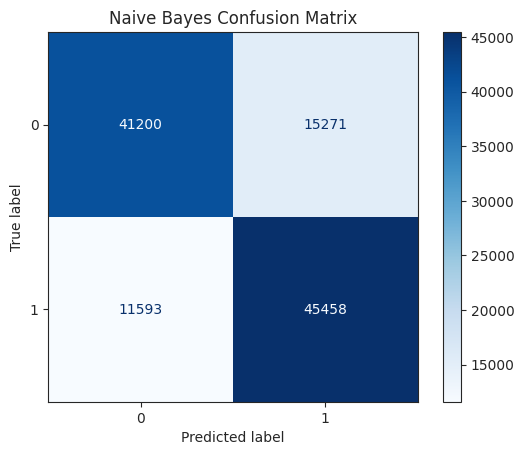
****

Figure 14.1: Logistic Regression Figure 14.2: Naïve Bayes

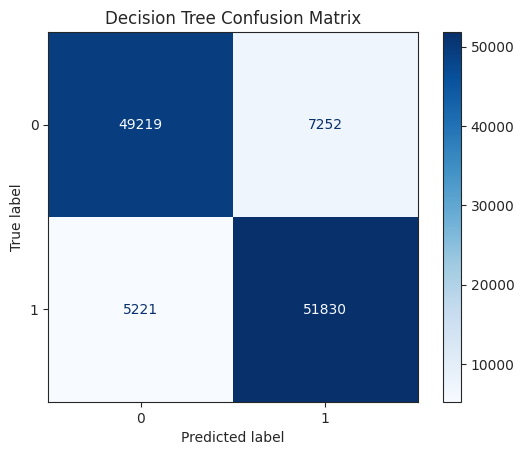
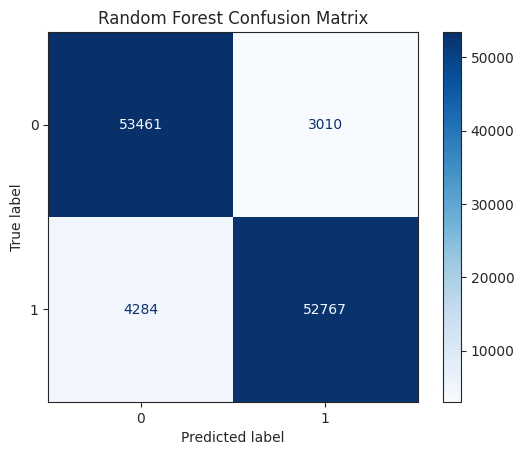
****

Figure 14.3: Decision Tree Figure 14.4: Random Forest

In the confusion matrix the rows represent the true labels and the columns represent the predicted labels. The diagonal elements represent the number of correct predictions made by the model, while the off-diagonal elements represent the number of incorrect predictions made by the model. The x-axis is labeled “Predicted label” and the y-axis is labeled “True label”.

**8.5 Comparative Analysis**

|  |  |  |
| --- | --- | --- |
| **Paper** | **Model** | **Accuracy** |
| [1](file:///D:\Documents\Fall%2023\IOT\Lab\Singh,%20N.%20and%20Singh,%20P.,%202019.%20Cardiac%20arrhythmia%20classification%20using%20machine%20learning%20techniques.%20In%20Engineering%20Vibration,%20Communication%20and%20Information%20Processing:%20ICoEVCI%202018,%20India%20(pp.%20469-480).%20Springer%20Singapore) | Random Forest | 85.58% |
| [2](file:///D:\Documents\Fall%2023\IOT\Lab\Niazi,%20K.A.K.,%20Khan,%20S.A.,%20Shaukat,%20A.%20and%20Akhtar,%20M.,%202015,%20July.%20Identifying%20best%20feature%20subset%20for%20cardiac%20arrhythmia%20classification.%20In%202015%20Science%20and%20Information%20Conference%20(SAI)%20(pp.%20494-499).%20IEEE) | KNN | 73.8% |
| [3](https://arxiv.org/abs/1504.04646) | Naïve Bayes | 82.3% |
| [4](https://arxiv.org/abs/1504.04646) | Logistic Regression | 85% |
| [5](https://arxiv.org/abs/1504.04646) | Hybrid Random Forest | 88.7% |
| Our Paper | Random Forest | 94% |

Table: 15

From the comparative analysis with some existing studies, we see that our proposed model outperformed them comprehensively. Hence, our model will be more accurate in predicting cardiovascular disease.

**9. Conclusion**

Heart disease is one of the main causes of death in the world. It is crucial to detect it in patients as soon as possible to prevent fatality. The proposed system can help clinicians in their accurate and reliable decision-making of CAD based on significant features of the patient.

**10. Reference**

[1] cardiovascular diseases (CVDs). http://www.who.int/newsroom/factsheets/detail/cardiovascular-diseases-(cvds accessed on 30/9/2018. [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Cardiovascular+diseases+(CVDs)&)]

[2] Patel B, Sengupta P. Machine learning for predicting cardiac events: what does the future hold? Expert Rev Cardiovasc Ther. 2020;18(2):77–84. [[PMC free article](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7962010/)] [[PubMed](https://pubmed.ncbi.nlm.nih.gov/32066289)] [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Expert+Rev+Cardiovasc+Ther&title=Machine+learning+for+predicting+cardiac+events:+what+does+the+future+hold?&author=B+Patel&author=P+Sengupta&volume=18&issue=2&publication_year=2020&pages=77-84&pmid=32066289&)]

[3] Baharvand-Ahmadi B, Bahmani M, Zargaran A. A brief report of Rhazes manuscripts in the field of cardiology and cardiovascular diseases. Int J Cardiol. 2016;207:190–1. [[PubMed](https://pubmed.ncbi.nlm.nih.gov/26803240)] [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Int+J+Cardiol&title=A+brief+report+of+Rhazes+manuscripts+in+the+field+of+cardiology+and+cardiovascular+diseases&author=B+Baharvand-Ahmadi&author=M+Bahmani&author=A+Zargaran&volume=207&publication_year=2016&pages=190-1&pmid=26803240&)]

[4] Weng SF, Reps J, Kai J, Garibaldi JM, Qureshi N. Can machine learning improve cardiovascular risk prediction using routine clinical data? PLOS ONE. 2017;12(4):e0174944. [[PMC free article](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5380334/)] [[PubMed](https://pubmed.ncbi.nlm.nih.gov/28376093)] [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=PLOS+ONE&title=Can+machine-learning+improve+cardiovascular+risk+prediction+using+routine+clinical+data?&author=SF+Weng&author=J+Reps&author=J+Kai&author=JM+Garibaldi&author=N+Qureshi&volume=12&issue=4&publication_year=2017&pages=e0174944&pmid=28376093&)]

[5] Yan H, Ye Q, Zhang T, Yu D-J, Yuan X, Xu Y, et al. Least squares twin bounded support vector machines based on L1-norm distance metric for classification. Pattern Recogn. 2018; 74:434–47. [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Pattern+Recogn&title=Least+squares+twin+bounded+support+vector+machines+based+on+L1-norm+distance+metric+for+classification&author=H+Yan&author=Q+Ye&author=T+Zhang&author=D-J+Yu&author=X+Yuan&volume=74&publication_year=2018&pages=434-47&)]

[6] Jaworski M, Duda P, Rutkowski L. New splitting criteria for decision trees in stationary data streams. IEEE Trans Neural Netw Learn Syst. 2018;29:2516–29. [[PubMed](https://pubmed.ncbi.nlm.nih.gov/28500013)] [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=IEEE+Trans+Neural+Netw+Learn+Syst&title=New+splitting+criteria+for+decision+trees+in+stationary+data+streams&author=M+Jaworski&author=P+Duda&author=L+Rutkowski&volume=29&publication_year=2018&pages=2516-29&pmid=28500013&)]

[7] Zhang S, Cheng D, Deng Z, Zong M, Deng X. A novel K-NN algorithm with data-driven k parameter computation. Pattern Recogn Lett. 2018;109:44–54. [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Pattern+Recogn+Lett&title=A+novel+K-NN+algorithm+with+data+driven+k+parameter+computation&author=S+Zhang&author=D+Cheng&author=Z+Deng&author=M+Zong&author=X+Deng&volume=109&publication_year=2018&pages=44-54&)]

[8] Abdar M, Zomorodi-Moghadam M, Das R, Ting IH. Performance analysis of classification algorithms on early detection of liver disease. Expert Syst Appl. 2017;67:239–51. [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Expert+Syst+Appl&title=Performance+analysis+of+classification+algorithms+on+early+detection+of+liver+disease&author=M+Abdar&author=M+Zomorodi-Moghadam&author=R+Das&author=IH+Ting&volume=67&publication_year=2017&pages=239-51&)]

[9] Abdar M, Yen NY, Hung JC-S. Improving the diagnosis of liver disease using multilayer perceptron neural network and boosted decision trees. J Med Biol Eng. 2017;10:1–13. [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=J+Med+Biol+Eng&title=Improving+the+diagnosis+of+liver+disease+using+multilayer+perceptron+neural+network+and+boosted+decision+trees&author=M+Abdar&author=NY+Yen&author=JC-S+Hung&volume=10&publication_year=2017&pages=1-13&)]

[10] Pławiak P. Novel genetic ensembles of classifiers applied to myocardium dysfunction recognition based on ECG signals, Swarm. Evol Comput. 2018;39:192–208. [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Evol+Comput&title=Novel+genetic+ensembles+of+classifiers+applied+to+myocardium+dysfunction+recognition+based+on+ECG+signals,+Swarm&author=P+P%C5%82awiak&volume=39&publication_year=2018&pages=192-208&)]

[11] Pławiak P. Novel methodology of cardiac health recognition based on ECG signals and evolutionary-neural system. Expert Syst Appl. 2018;92:334–49. [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Expert+Syst+Appl&title=Novel+methodology+of+cardiac+health+recognition+based+on+ECG+signals+and+evolutionary-neural+system&author=P+P%C5%82awiak&volume=92&publication_year=2018&pages=334-49&)]

[12] Khozeimeh F, Alizadehsani R, Roshanzamir M, Khosravi A, Layegh P, Nahavandi S. An expert system for selecting wart treatment method. Comput Biol Med. 2017;81:167–75. [[PubMed](https://pubmed.ncbi.nlm.nih.gov/28086200)] [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Comput+Biol+Med&title=An+expert+system+for+selecting+wart+treatment+method&author=F+Khozeimeh&author=R+Alizadehsani&author=M+Roshanzamir&author=A+Khosravi&author=P+Layegh&volume=81&publication_year=2017&pages=167-75&pmid=28086200&)]

[13] Khozeimeh F, Azad FJ, Oskouei YM, Jafari M, Tehranian S, Alizadehsani R, et al. Intralesional immunotherapy compared to cryotherapy in the treatment of warts. Int J Dermatology. 2017;56:474–8. [[PubMed](https://pubmed.ncbi.nlm.nih.gov/28108992)] [[Google Scholar](https://scholar.google.com/scholar_lookup?journal=Int+J+Dermatology&title=Intralesional+immunotherapy+compared+to+cryotherapy+in+the+treatment+of+warts&author=F+Khozeimeh&author=FJ+Azad&author=YM+Oskouei&author=M+Jafari&author=S+Tehranian&volume=56&publication_year=2017&pages=474-8&)]

[14] Alizadehsani R, Abdar M, Jalali SMJ, Roshanzamir M, Khosravi A, Nahavandi S. Comparing the performance of feature selection algorithms for wart treatment selection. Proc. Int. Workshop Future Technol; 2018. p. 6–18. [[Google Scholar](https://scholar.google.com/scholar_lookup?title=Proc.+Int.+Workshop+Future+Technol&author=R+Alizadehsani&author=M+Abdar&author=SMJ+Jalali&author=M+Roshanzamir&author=A+Khosravi&publication_year=2018&)]