**KONGU ENGINEERING COLLEGE**

**(Autonomous) PERUNDURAI, ERODE- 638 060**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**MACHINE LEARNING – 22ITT52**

**MINI PROJECT REPORT**

# HEART FAILURE PREDICTION

**TEAM MEMBERS**

**AKHIL ARAVINDH P N – 22ITR004**

**DINESHKUMAR V - 22ITR023**

**GUNA D – 22ITR027**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **SNO** | **TITLE** | **PAGE NO** |
| 1 | Classification | 3 |
| 1.1 | K Nearest Neighboror | 5 |
| 1.2 | Naïve Bayes | 6 |
| 1.3 | Decision Tree | 7 |
| 1.4 | Random Forest | 8 |
| 1.5 | Support Vector Machine | 9 |
| 1.6 | Code Snippets | 10 |
|  | Conclusion(Classification) | 16 |
| 2 | Regression | 17 |
| 2.1 | Types of Regression | 17 |
| 2.1.1 | Simple Linear Regression | 18 |
| 2.1.2 | Multiple Linear Regression | 18 |
| 2.1.3 | Polynomial Regression | 19 |
| 2.1.4 | Logistic Regression | 19 |
| 2.2 | Code Snippets | 20 |
|  | Conclusion(Regression) | 24 |

# CLASSIFICATION

The Classification algorithm is a Supervised Learning technique that is used to identify the category of new observations on the basis of training data. In Classification, a program learns from the given dataset or observations and then classifies new observation into a number of classes or groups. Such as, **Yes or No, 0 or 1, Spam or Not Spam, cat or dog,** etc. Classes can be called as targets/labels or categories.

## LEARNERS IN CLASSIFICATION

**I. LAZY LEARNERS**

Lazy Learner firstly stores the training dataset and wait until it receives the test dataset. In Lazy learner case, classification is done on the basis of the most related data stored in the training dataset. It takes less time in training but more time for predictions. The main advantage of lazy learners is their simplicity and the fact that they can adapt to new data without the need to retrain the entire model. This makes them suitable for applications where the training data may change over time. **Example:** K-NN algorithm.

**II**.**EAGER LEARNERS**

Eager Learners develop a classification model based on a training dataset before receiving a test dataset. Opposite to Lazy learners, Eager Learner takes more time in learning, and less time in prediction. The main advantage of eager learners is their ability to make fast predictions once the model is trained. This makes them suitable for applications where the response time is critical, such as real-time decision-making or high-volume predictions.**Example:** Decision Trees, Naive Bayes, ANN.

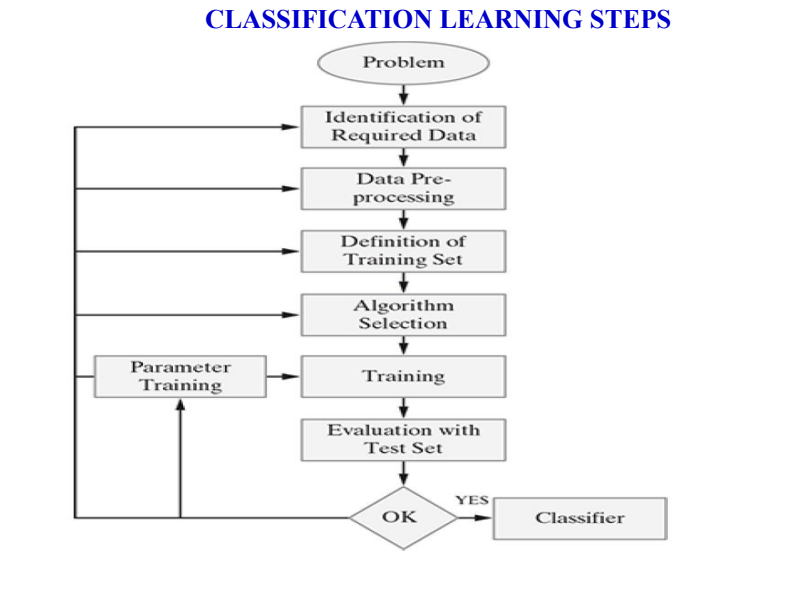
**CLASSIFIER**

**I.BINARY CLASSIFIER**

If the classification problem has only two possible outcomes, then it is called as Binary Classifier.**Examples:** YES or NO, MALE or FEMALE, SPAM or NOT SPAM, etc.

**II.MULTI-CLASS CLASSIFIER**

If a classification problem has more than two outcomes, then it is called as Multi-class Classifier.**Example:** Classifications of types of crops, Classification of types of music.



# 1.1 K-NEAREST NEIGHBOR

K-Nearest Neighbors (K-NN) is a lazy learning algorithm used for classification. It is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions).

* It is a lazy learning algorithm, meaning it does not have an explicit training phase. The model is built during the prediction/classification stage.
* K-NN classifies a new data point by finding the K nearest neighbors from the training data, and then assigns the class that is most common among those K neighbors.
* The value of K (the number of nearest neighbors to consider) is a hyperparameter that needs to be tuned based on the problem.
* K-NN is highly effective for multi-modal classes and does not make any assumptions about the underlying data distribution.
* It is commonly used in applications such as handwriting recognition, image classification, recommendation systems, and anomaly detection.
* The main advantages of K-NN are its simplicity, flexibility, and ability to perform well on non-linear data. However, it can be computationally expensive for large datasets, as it needs to calculate distances between the new data point and all training data points.

**HYPERPARAMETERS OF KNN**

 **Number of neighbors (K):** This determines how many of the closest data points will be used to make the classification or regression prediction.

 **Distance metric**: This determines the way the distance between the query point and the training data points is calculated, which affects the identification of the nearest neighbors.

 **Weighting Scheme**: This determines how the votes of the K nearest neighbors are combined, with different weighting schemes giving more or less importance to the closer neighbors.

# ALGORITHM

# 1. Find distance( Euclidean or Manhatten )

# 2. Find Rank

# 3. Find the nearest neighbor

# 1.2 NAÏVE BAYAS

* Naïve Bayes algorithm is a supervised learning algorithm, which is based on **Bayes theorem** and used for solving classification problems.
* It is mainly used in *text classification* that includes a high-dimensional training dataset.
* Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.
* It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.
* Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

# BAYES THEOREM

* Bayes' theorem is also known as **Bayes' Rule** or **Bayes' law**, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.

# The formula for Bayes' theorem is given as:

# Naïve Bayes Classifier Algorithm

**Where,**

* **P(A|B) is Posterior probability**: Probability of hypothesis A on the observed event B.
* **P(B|A) is Likelihood probability**: Probability of the evidence given that the probability of a hypothesis is true.
* **P(A) is Prior Probability**: Probability of hypothesis before observing the evidence.
* **P(B) is Marginal Probability**: Probability of Evidence.

**1.3 DECISION TREE**

Decision Trees are a popular machine learning algorithm used for both classification and regression tasks. They work by recursively partitioning the input space based on feature values, creating a tree-like model of decisions and their possible consequences.

* Decision Trees are a type of supervised learning algorithm, meaning they learn a model from labeled training data.
* They work by repeatedly splitting the data based on the feature that provides the most "information gain" or "Gini impurity reduction" at each node of the tree.
* The tree is built top-down, with the root node representing the entire dataset, and leaf nodes representing the final predictions.
* Decision Trees are highly interpretable, as the logic behind the predictions can be easily visualized and understood from the tree structure.
* They can handle both numerical and categorical features, and can learn complex non-linear relationships in the data.
* Decision Trees are relatively fast to train and make predictions, and they are also robust to outliers and missing values.
* The main hyperparameters to tune in Decision Trees are the maximum depth of the tree, the minimum number of samples required to split a node, and the minimum number of samples in a leaf node.
* Decision Trees are often used as the base estimator in ensemble methods like Random Forests and Gradient Boosting, which can significantly improve their performance.
* Decision Trees are widely used in a variety of applications, such as credit risk assessment, customer churn prediction, medical diagnosis, and image recognition.

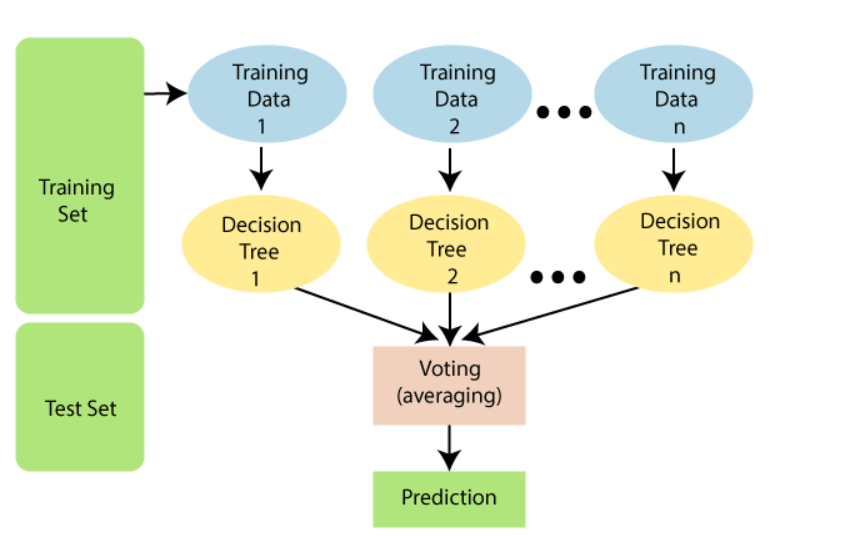
**DECISION TREE TERMINOLOGIES**

* **Root Node:** Root node is from where the decision tree starts. It represents the entire dataset, which further gets divided into two or more homogeneous sets.
* **Leaf Node:** Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.
* **Splitting:** Splitting is the process of dividing the decision node/root node into sub-nodes according to the given conditions.
* **Branch/Sub Tree:** A tree formed by splitting the tree.
* **Pruning:** Pruning is the process of removing the unwanted branches from the tree.
* **Parent/Child node:** The root node of the tree is called the parent node, and other nodes are called the child nodes.

**1.4 RANDOM FOREST**

Random Forest is an ensemble learning algorithm used for both classification and regression tasks. It is a collection of decision tree models, where each tree is trained on a random subset of the features and a random subset of the training data.

* Random Forest is an ensemble method that combines multiple decision trees to improve the overall performance and stability of the model.
* Each individual decision tree in the forest is trained on a random subset of the features and a random subset of the training data, using the technique of bagging (bootstrap aggregating).
* The final prediction is made by aggregating the predictions from all the decision trees in the forest, either through majority voting (for classification) or averaging (for regression).
* The randomness introduced in the training of each decision tree helps to reduce overfitting and improve the generalization of the overall model.
* Key hyperparameters to tune in Random Forest include the number of trees in the forest, the maximum depth of each tree, the minimum number of samples required to split a node, and the number of features to consider at each split.
* Random Forest is capable of handling both numerical and categorical features, and it can automatically handle missing values and outliers.
* It is highly scalable and can be parallelized, making it efficient for large-scale datasets.
* Random Forest is widely used in a variety of applications, such as image recognition, natural language processing, bioinformatics, and financial modeling.



**1.5 SUPPORT VECTOR MACHINE**

Support Vector Machines (SVM) are a powerful and versatile machine learning algorithm used for both classification and regression tasks. SVMs work by finding the optimal hyperplane that best separates different classes of data in a high-dimensional feature space.

**OPTIMAL SEPARATING HYPERPLANE**

There are three hyperplanes: A, B, and C to classifies the triangles and circles in the best possible way. The optimal separating plane not only best for working in training data , also best for both test data and new data.

A diagram of a diagram of a triangle

Description automatically generated with medium confidence

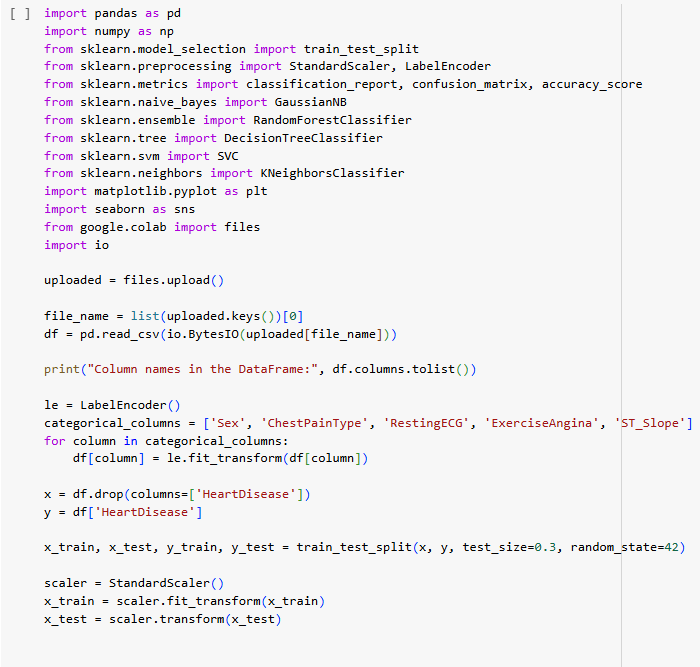
**SUPPORT VECTORS**

The data points that lie close to the Maximum Margin Hyperplane(MMH) are called support vectors. There should be atleast one support vector from each class.

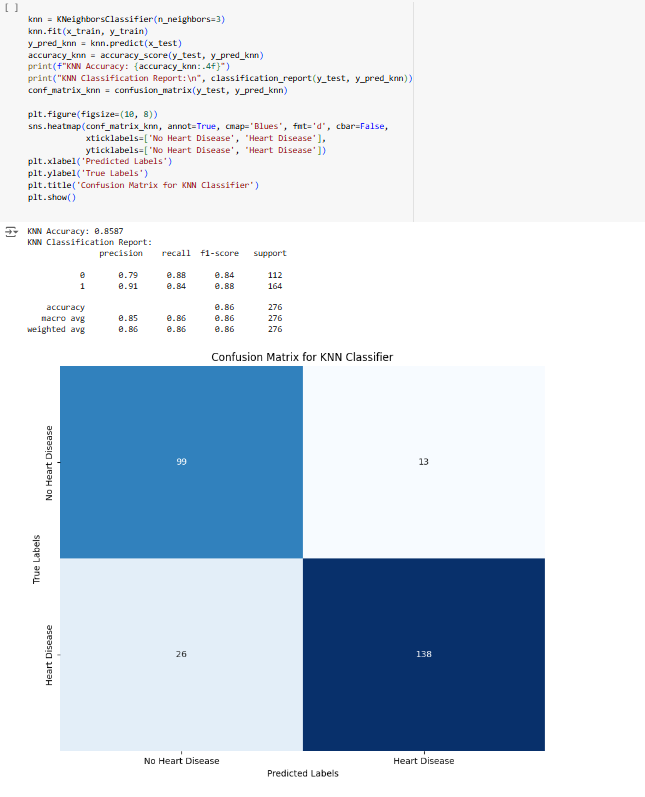
A diagram of support vector

Description automatically generated

**1.6 CODE SNIPPETS**



**K-NEAREST NEIGHBOR**

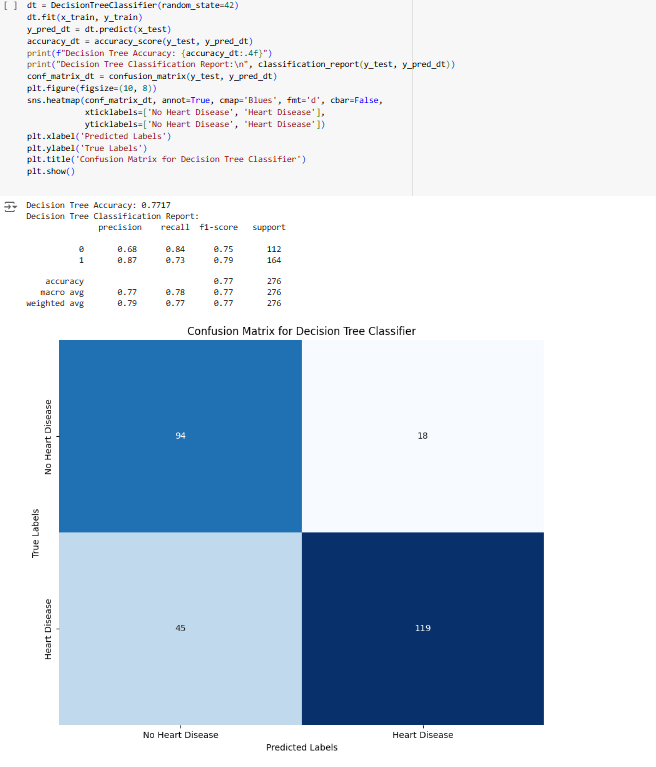
****

**NAÏVE BAYES**

**A screenshot of a computer

Description automatically generated**

**DECISION TREE**

****

**RANDOM FOREST**

A screenshot of a computer

Description automatically generated

**SUPPORT VECTOR MACHINE**

A screenshot of a computer

Description automatically generated

**ACCURACY TABLE**

|  |  |
| --- | --- |
| **CLASSIFIER** | **ACCURACY** |
| K-Nearest Neighbhor | 85.87% |
| Naïve Bayes | 87.32% |
| Decision Tree | 77.17% |
| Random Forest | 89.49% |
| Support Vector Machine | 85.87% |

**CONCLUSION**

The main aim of the system is to classify heart failure risk in patients based on various health indicators. For classifying the risk of heart failure, the "Heart Failure Prediction dataset" from Kaggle has been used. The dataset has been visualized and any missing values have been addressed. The dataset has been used to train models such as K-Nearest Neighbour, Support Vector Machine, Decision Tree, Naïve Bayes, and Random Forest. Evaluation metrics, including accuracy, recall, and the confusion matrix, have been utilized. Among these models, the Random Forest classifier achieved the highest accuracy. This approach aims to assist the healthcare sector in predicting heart failure risk and improving patient outcomes.

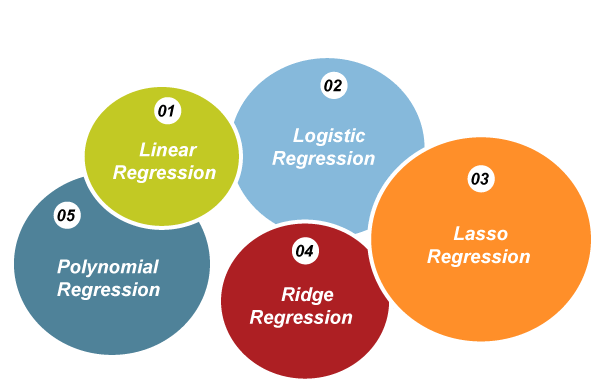
**REGRESSION**

Regression is a supervised learning technique which helps in finding the correlation between variables and enables us to predict the continuous output variable based on the one or more predictor variables. It is mainly used for **prediction, forecasting, time series modeling, and determining the causal-effect relationship between variables.**

## REGRESSION TERMINOLOGIES

* **Dependent Variable:** The main factor in Regression analysis which we want to predict or understand is called the dependent variable. It is also called **target variable**.
* **Independent Variable:** The factors which affect the dependent variables or which are used to predict the values of the dependent variables are called independent variable, also called as a **predictor.**
* **Outliers:** Outlier is an observation which contains either very low value or very high value in comparison to other observed values. An outlier may hamper the result, so it should be avoided.
* **Multicollinearity:** If the independent variables are highly correlated with each other than other variables, then such condition is called Multicollinearity. It should not be present in the dataset, because it creates problem while ranking the most affecting variable.
* **Underfitting and Overfitting:** If our algorithm works well with the training dataset but not well with test dataset, then such problem is called **Overfitting**. And if our algorithm does not perform well even with training dataset, then such problem is called **underfitting.**

# 2.1 TYPES OF REGRESSION

****

* + 1. **SIMPLE LINEAR REGRESSION**

Simple Linear Regression is a statistical method used to model the relationship between two variables:

* **Dependent variable (y)**: This is the outcome or the variable we are trying to predict or explain. It is also called the response variable.
* **Independent variable (x)**: This is the predictor or feature. It is the variable used to explain the variation in the dependent variable.

The relationship between these two variables is assumed to be linear, meaning that the change in the dependent variable can be explained by a straight line equation. The general form of this equation is:

**y=a0+a1x+ε**

Where:

* Y - Dependent variable (the variable we want to predict or explain).
* X - Independent variable (the variable used for prediction).
* a0 - Intercept of the regression line (the value of y when x=0).
* a1 - Slope of the regression line (how much y changes for a one-unit change in x).
* ε - Error term (accounts for the difference between the predicted and actual values of y).

**2.1.2 MULTIPLE LINEAR REGRESSION**

**Multiple Linear Regression** is an extension of Simple Linear Regression that models the relationship between a dependent variable and **multiple** independent variables. It is used when there is more than one predictor variable.

Equation:

The general form of the equation in Multiple Linear Regression is:

**y=a0+a1x1+a2x2+⋯+anxn+ε**

Where:

* Y **-**  Dependent variable (the outcome or response variable we are trying to predict).
* x1​,x2​,…,xn​ - Independent variables (the predictors or features).
* a0 **-** Intercept of the regression line (the value of y when all xi=0).
* a1​,a2​,…,an​ - Coefficients of the independent variables. These represent the change in the dependent variable for a one-unit change in the corresponding independent variable, assuming other variables remain constant.
* **ε** = Error term (captures the deviation between the observed and predicted values of y).

**2.1.3 POLYNOMIAL REGRESSION**

**Polynomial Regression** is a type of regression that models the relationship between the dependent variable and independent variable as an **nth-degree polynomial**. It is used when the data shows a non-linear relationship between the variables, but a straight line can't capture that complexity.

Equation:

The general form of the equation in Polynomial Regression is:



Where:

* Y - Dependent variable.
* X - Independent variable.
* a0**​** - Intercept of the regression.
* a1​,a2​,…,an**​** - Coefficients for each term in the polynomial.
* x^n - Polynomial term (for example, x^2 for quadratic regression, x^3 for cubic regression, etc.).
* **ε** - Error term.

**2.1.4 LOGISTIC REGRESSION**

Logistic regression is a type of regression analysis used to predict the outcome of a categorical dependent variable based on one or more independent variables. It is commonly used when the dependent variable is binary (i.e., it can only take two values, such as 0 or 1, true or false, yes or no).

The equation for logistic regression is:

P(Y=1|X) = 1 / (1 + e^-(a0 + a1X))

Where:

* P(Y=1|X) is the probability of the dependent variable Y being 1 (or "true"), given the values of the independent variables X.
* a0 is the y-intercept (the value of the log-odds when all the independent variables are 0).
* a1, a2, ..., an are the regression coefficients, which represent the change in the log-odds of the dependent variable for a one-unit change in the corresponding independent variable, holding all other variables constant.
* e is the base of the natural logarithm(approximately 2.718).
  1. **CODE SNIPPETS**

**SIMPLE LINEAR REGRESSION**

A screenshot of a computer program

Description automatically generated

A graph with blue dots and a line

Description automatically generated

**MULTIPLE LINEAR REGRESSION**

A screen shot of a computer screen

Description automatically generated

**POLYNOMIAL REGRESSION**

A screen shot of a computer screen

Description automatically generated

**LOGISTIC REGRESSION**

A screenshot of a computer

Description automatically generated

**CONCLUSION**

The main aim of the system is to predict heart failure. For predicting heart failure, the “Heart Failure Prediction Dataset” has been used, which has been provided by Kaggle. The dataset has been visualized and analyzed to understand the relationships between various features and the risk of heart failure. Key features have been retained after feature selection to enhance model performance. The dataset has been used to train several machine learning models, including Linear Regression, Multiple Linear Regression,Logistic Regression, and Polynomial Regression . Scatter plots and other visualizations have been used to assess the performance of these models. Hopefully, this project will help in predicting heart failure risk, potentially assisting healthcare professionals in early detection and treatment planning.

**GOOGLE COLAB LINK:**

[**https://colab.research.google.com/drive/1LQWin8fIZdX0Oy83D4tij\_ZwsNkezrXB?usp=sharing**](https://colab.research.google.com/drive/1LQWin8fIZdX0Oy83D4tij_ZwsNkezrXB?usp=sharing)