# Data analysis and machine learning with Python for real-world problems: From data to informed decision making

# 1. Introduction

## 1.1. Overview of Predictive Analytics and Machine Learning

Some of the popular data science techniques include predictive analytics and machine learning, which involve using past data to make future predictions. These techniques apply statistics and machine learning to enhance business processes, market trends, and customer satisfaction. Machine learning, a branch of AI, causes algorithms to identify patterns in the data and make decisions. These tools assist firms in managing and analyzing masses of data in order to convert it into valuable knowledge that can be used in decision-making.

## 1.2 Importance of Data-Driven Decision Making

## Strategic management is a process that helps in defining the future course of action of the business based on certain information and reducing uncertainty. This way, it is possible to find hidden patterns, tendencies, and forecasts with the best possible accuracy. This process improves market creativity, customer satisfaction/engagement, and operational effectiveness. Because the digital economy requires firms to navigate internal and external changes, information-driven decision-making is critical to organizational success.

## 1.3 Objective of the Report

## In this report, the Python programming language is used for data analysis and machine learning to analyze genuine business and social challenges. It includes the aspects of data gathering, choosing a model, applying the model, and evaluating the model’s performance. There is a special focus on predictive analysis and machine learning, the role of these technologies in the decision-making process, and the possibility of finding new values for businesses and societies.

# 2. Task 1: Data Selection and Exploratory Data Analysis

## 2.1. Dataset Selection

The selected dataset for this analysis relates to the identification of outcomes of patients with CVDs and heart failure sourced from the [Kaggle platform](https://www.kaggle.com/datasets/andrewmvd/heart-failure-clinical-data). Considering that cardiovascular diseases are the primary cause of death in the world, increasing their ranking every year and with 17. , approximately 9 million deaths per year; this dataset is somewhat helpful for forming certain prognoses. The collected dataset covers twelve parameters that can be used to predict mortality due to heart failure and includes clinical coefficients and health parameters such as age, sex, serum creatinine, ejection fraction, and others.

## 2.2. Identification of Data Types

## Some of the health attributes include age, anemia, diabetes, hypertension, sex, smoking status, and death events. The age is a float, while all other health features are either integers or floats. The time feature is an integer that refers to the follow-up duration. These features include the presence of anemia, diabetes, hypertension, gender, smoking history, and death.

## 2.3. Data Cleaning and Preprocessing

It is essential to perform cleaning and preprocessing on the data before they are analyzed so as to take care of issues such as inconsistency, missing data, or extreme values.

### Steps for Data Cleaning

* **Check for Missing Values**: Check that no values are missing that could influence model build-up.

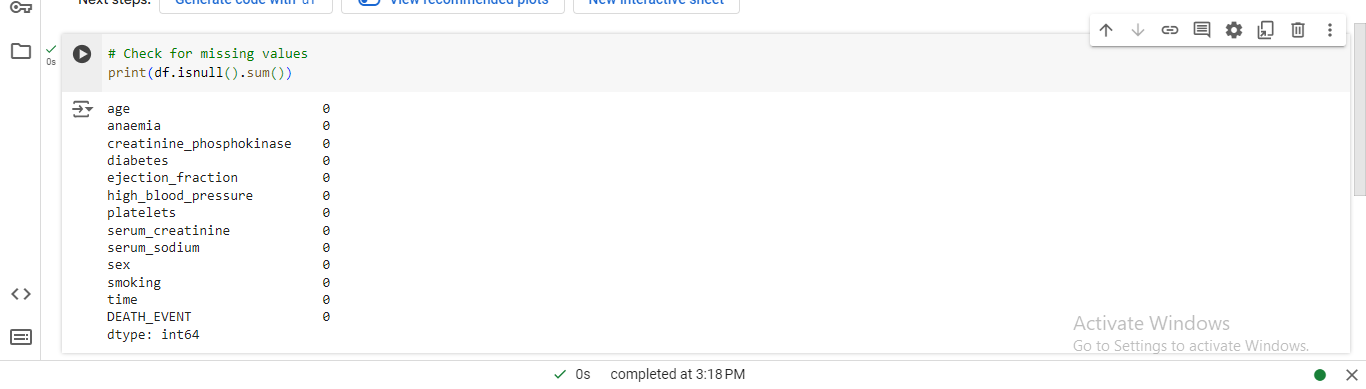


Figure : Jupyter Notebook Code and Output: Missing Values Check

The data has no missing values, so there is no need for data imputation or data cleaning. The data is, therefore, neat and can be used for further analysis, suggesting that it is ready for analysis.

* **Handling Outliers**: Check for numerical characteristics of the data mass, looking for irregularities that may distort it.

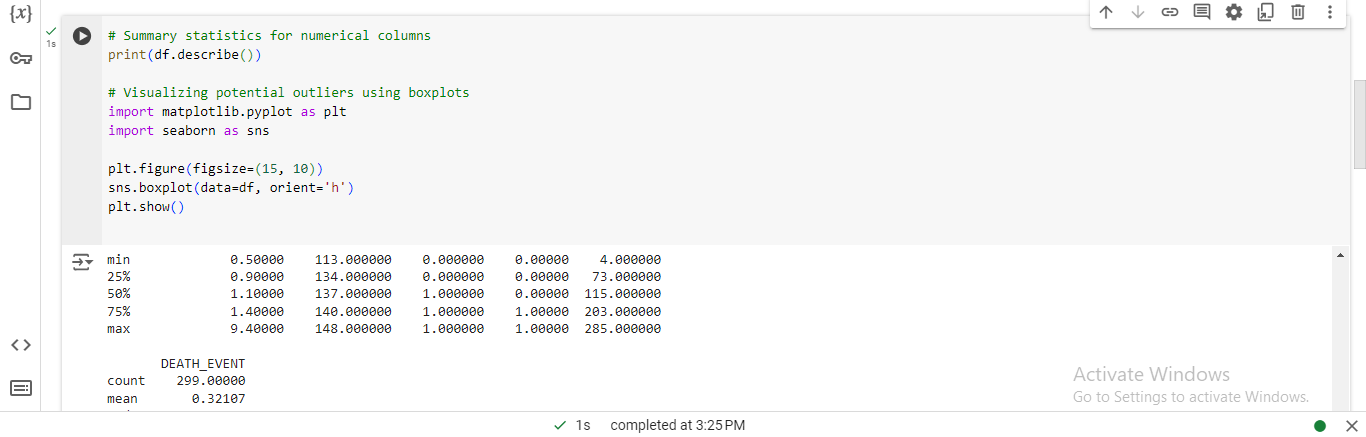


Figure : Box Plots of Numerical Features

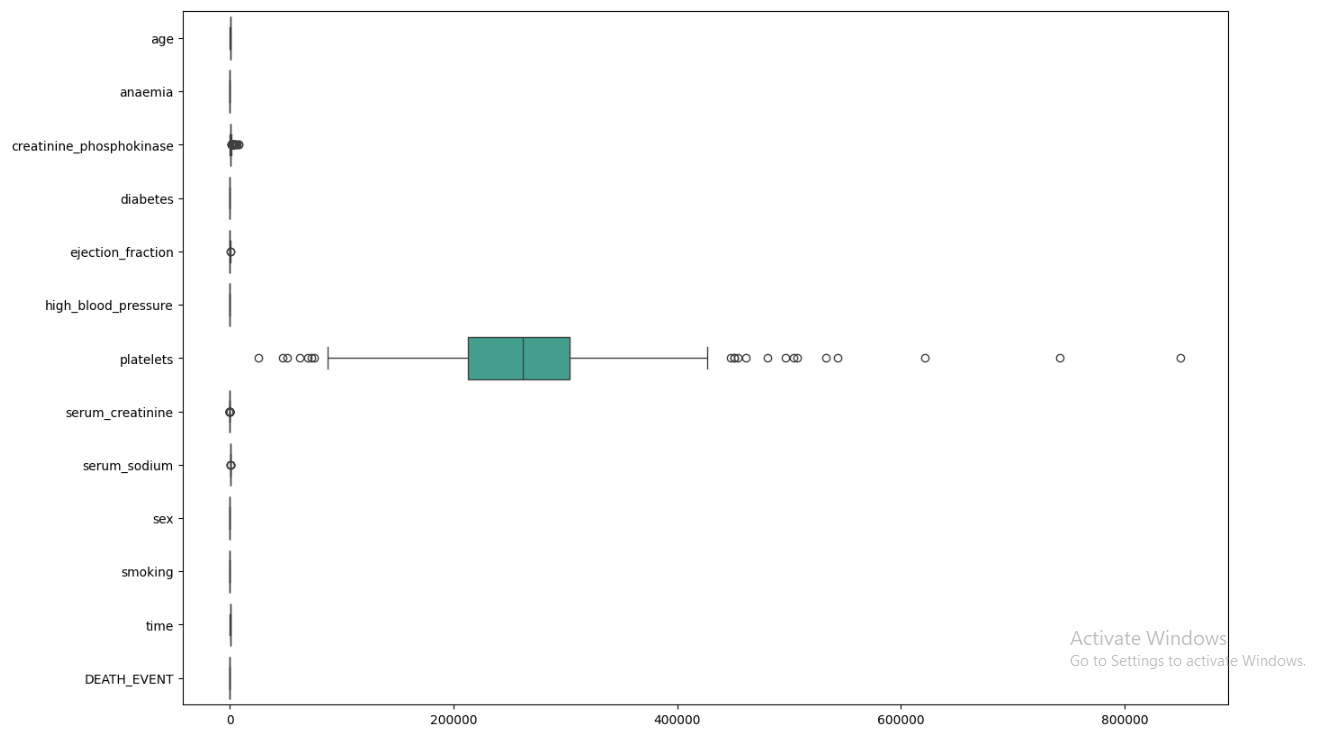


Figure : Box Plots of Numerical Features

## 2.4. Data Encoding

Machine learning models may need encoding for categorical data interpretation, as most features are already represented as binary values (0 or 1), so no additional encoding is needed. However, ensuring all categorical data is in the correct format (integer) is crucial.

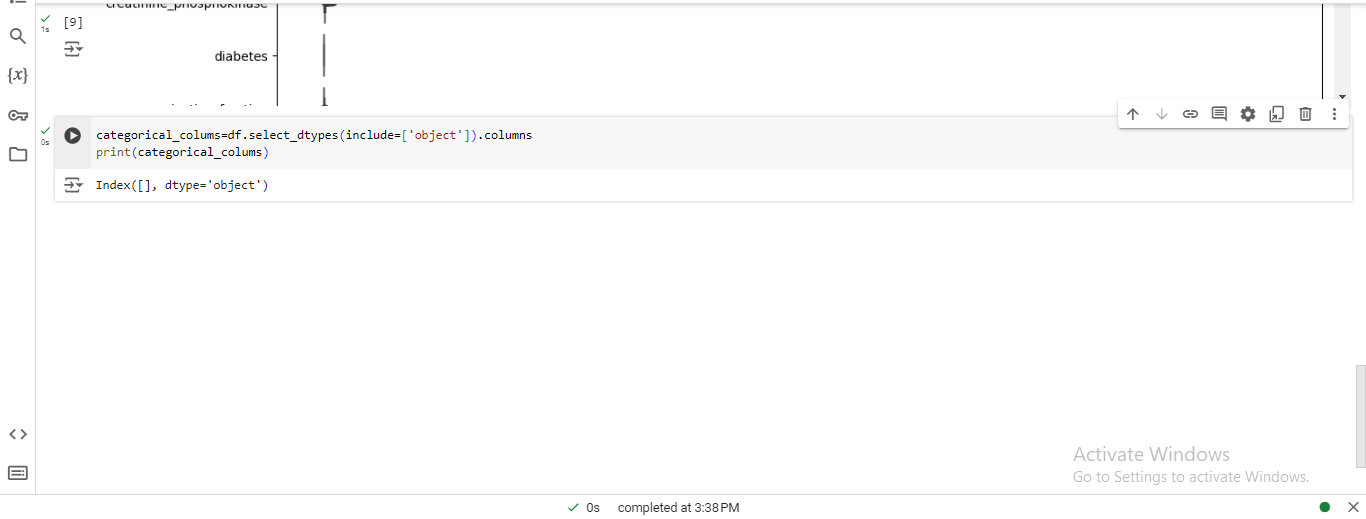


Figure : Jupyter Notebook Code and Output: Categorical Column Identification

This makes the dataset in the correct format required for analysis and modeling since no categorical variables need to go through the encoding process. All data collected is numerical, and all collected data is stored as integers. In so doing, no further coding is needed. This makes it easier for the subsequent steps of machine learning, where little or no preprocessing of data from categorical variables is required.

## 2. 5. Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) enables one to understand the distribution of the data, possible correlations between variables, and even patterns.

### Steps for EDA:

1. **Descriptive Statistics**: Learn the most basic measure of the centeredness, spread, and general appearance of the distribution of your dataset in a matter of seconds.

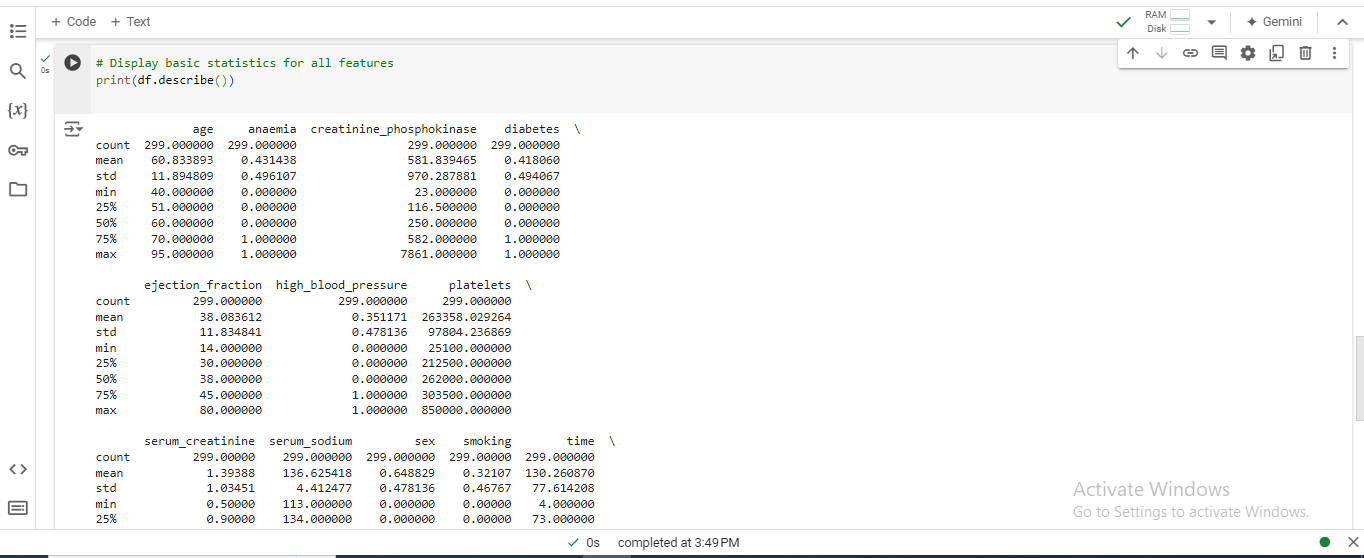


Figure : Descriptive Statistics for Numerical Features

The population under consideration consists of patients of 60 years and above and the mean age of the patients is 43 years—1%. The average level of creatinine is 581 mcg/L and 41.8% have diabetes. The mean ejection fraction is 38.1%, and 35.1% of the people have high blood pressure. The platelet count is 263358, and the average serum creatinine is 39mg/dL. The male patients in their reproductive age stand at 32%, while those who smoke are only 1%. The survey has a median follow-up time of 130 days, and 32% of respondents have experience of attending a death event. The above statistical measures help in the analysis and modeling of data for developing AI and ML models.

* **Correlation Matrix**: Subsequently, further features analysis is to define possible relationships between different features.

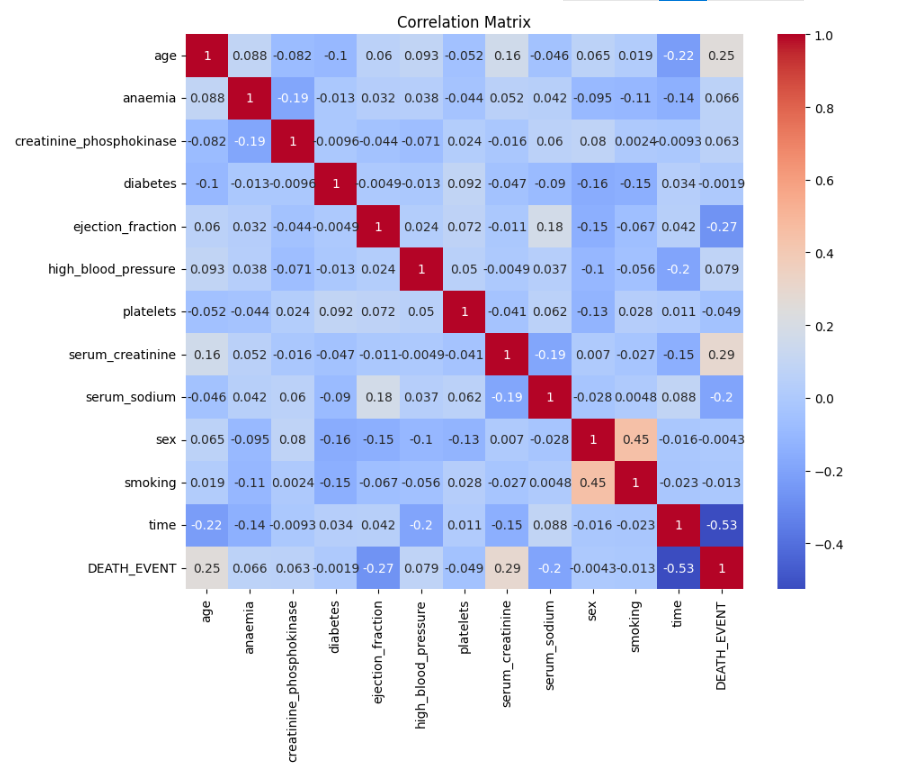


Figure : Correlation Matrix of Heart Disease Dataset

* **Visualizing Distributions**: Probability density curves can be plotted as histograms or Kernel Density Estimation curves to visualize the numerical feature distribution.

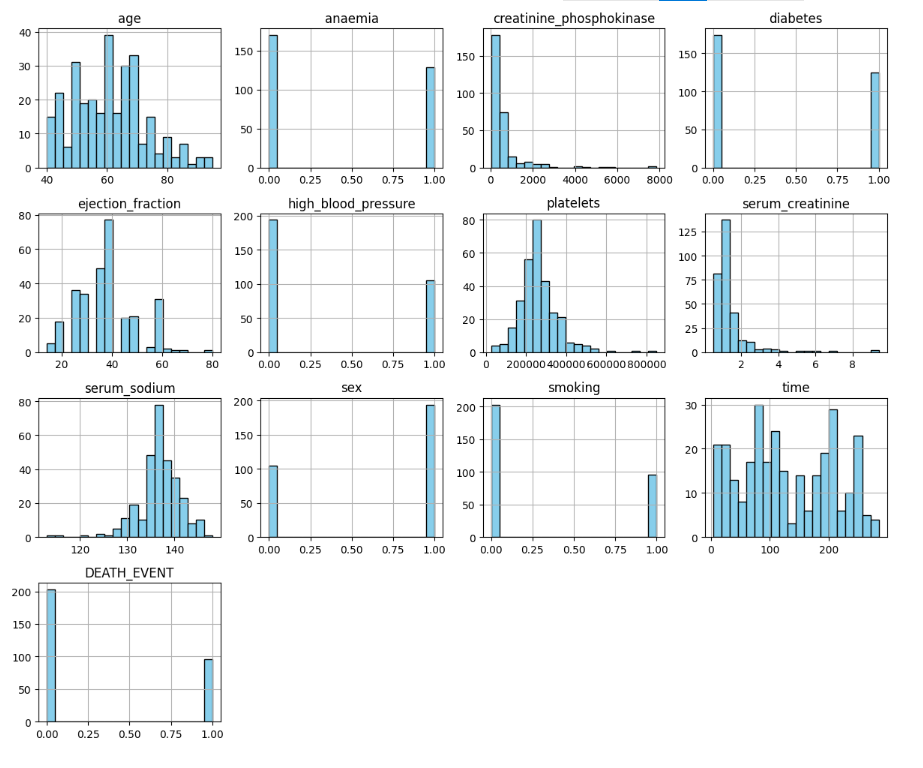


Figure : Histograms of Heart Disease Dataset Features

* **Pairplot**: Employ the use of pair plots in order to establish clusters as well as patterns between each pair of variables.

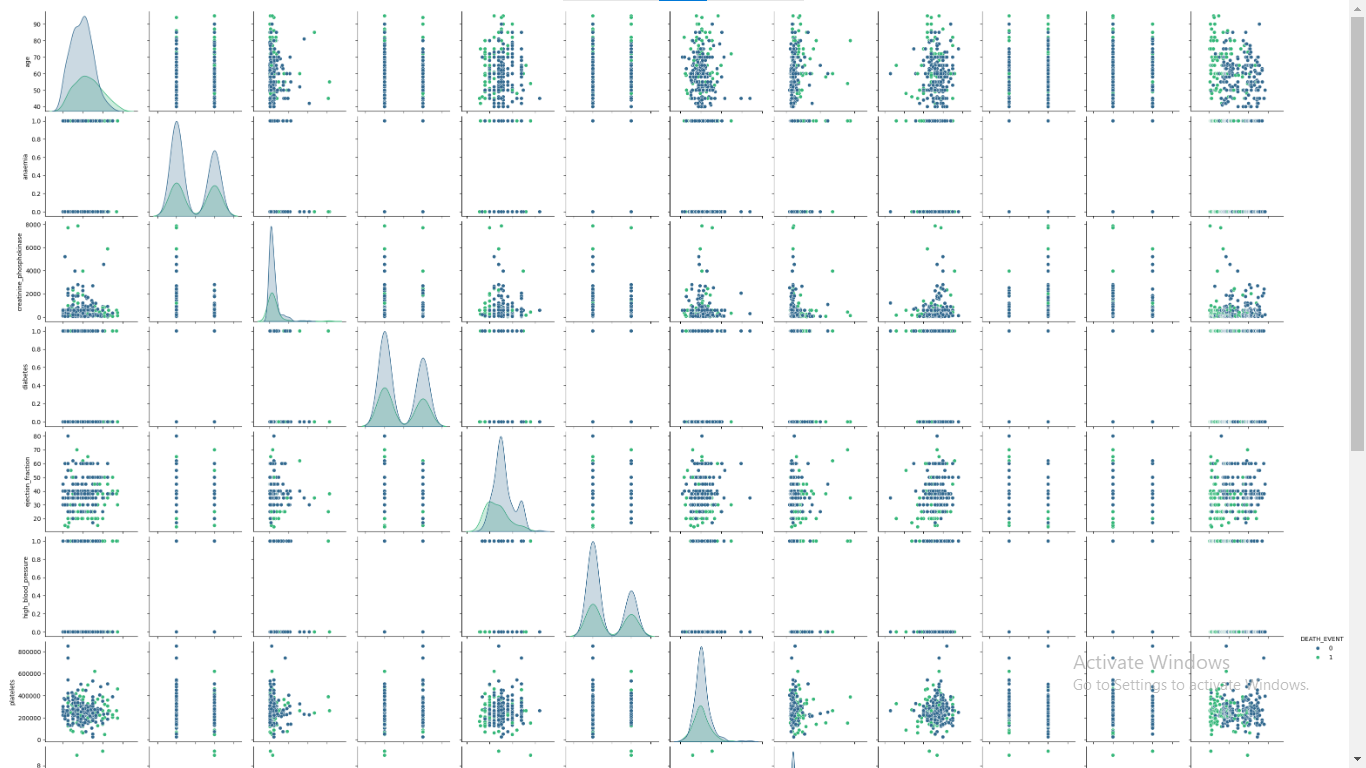


Figure : Pairplot of Heart Disease Dataset

* **Feature Relationships**: To examine the dependencies between certain features, resort to scatter plots or box plots.

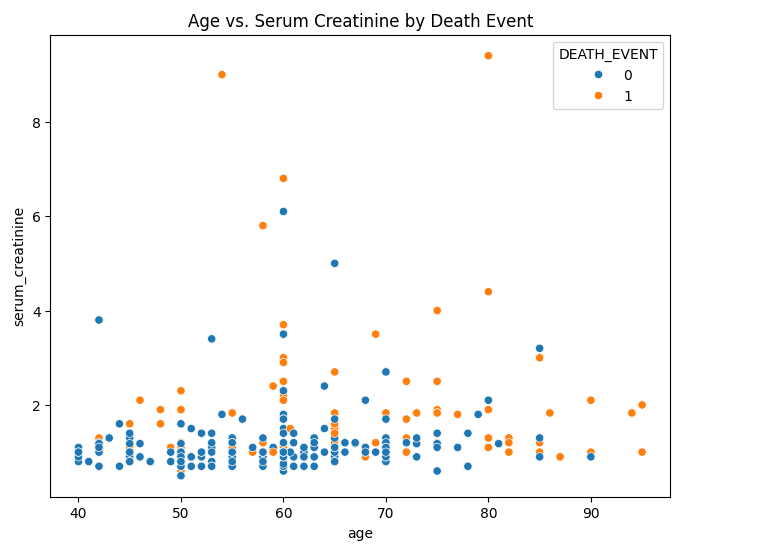


Figure : Scatter Plot of Age vs. Serum Creatinine by Death Event

Through the above steps, one gets a good understanding of the data, which will aid in the choice of the right machine learning models, fully described in the later sections.

# 3. Task 2: Machine Learning Model Selection and Implementation

## 3.1. Introduction to Model Selection

It is imperative to choose the right machine learning models for solving analytical problems, such as heart failure prediction. The models will be selected depending on their efficiency in categorizing data into two groups, such as binary, where the variable of interest is DEATH\_EVENT. Performance measures will be applied to determine the accuracy of the models in fitting the characteristics of the given dataset.

## 3.2. Model 1: Logistic Regression

### 3.2.1. Model Description

Logistic regression is a classification technique used to predict the probability of a discrete event, i.e., binary outcomes, by linearly combining the input features with a logistic function. It can be used for risk assessment of an event's occurrence and is, therefore, quite useful for heart failure prediction.



Figure : Jupyter Notebook Code and Output: Logistic Regression Model Training and Evaluation

## 3.3. Model 2: Decision Tree Classifier

### 3.3.1. Model Description

The Decision Tree Classifier is a supervised learning model used for classification purposes. It establishes a given model in terms of a number of decision rules generated from the features. In this case, it will be employed to predict patients' high- and low-risk categories for death from heart failure (DEATH\_EVENT).

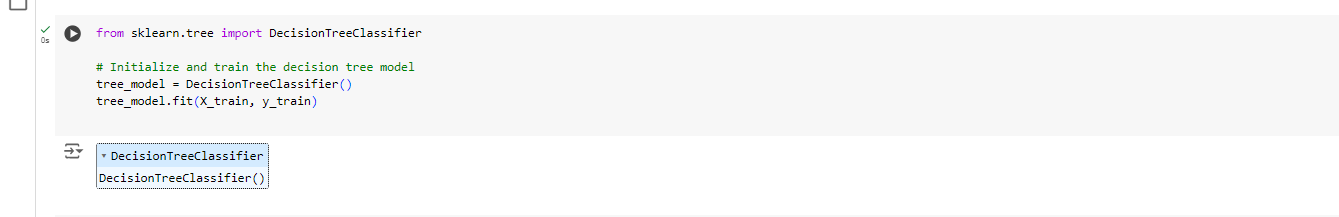


Figure : Jupyter Notebook Code: Decision Tree Model Initialization

## 3.4. Model 3: Random Forest Classifier

### 3.4.1. Model Description

Another supervised learning algorithm is the Random Forest Classifier, which is an ensemble learning method that generates multiple decision trees during the training process and gives out the class, which is the mode of the classes of the individual trees. It can also handle both numerical and categorical data and is efficient for data with many numerical features, especially when dealing with large data sets. Random Forest is not sensitive to overfitting and gives good results for the classification problems.

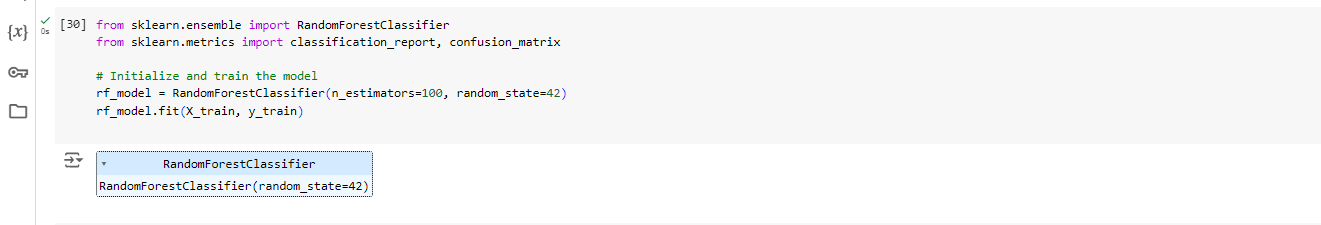


Figure : Jupyter Notebook Code: Random Forest Model Initialization

These models serve as a basis for predicting heart failure mortality given clinical characteristics and can be contrasted to identify the best strategy for this particular problem.

# 4. Task 3: Model Evaluation and Comparison

## 4. 1. Evaluation Metrics

To assess the performance of the machine learning models, several evaluation metrics are used:

* **Confusion Matrix**: A result of prediction is a brief overview of the results of the model’s work, which includes the number of true positives, true negatives, false positives, and false negatives.
* **Precision**: The number of truly positive cases that have been predicted as positive divided by the total number of positive cases predicted. Precision is defined as the ratio of the True Positive to the sum of True Positive and False Positive.
* **Recall**: The number of true positive instances to all instances that were in the actual class. Accuracy= TN / (TN + FP) and Recall = TP / (TP + FN).
* **F1-Score**: The mean of the Precision and the Recall with weights equal to their respective values. F1-Score was calculated as 2 \* [Precision \* Recall] / [Precision + Recall].
* **Accuracy**: Compared to an ideal set of observations, the number of observations made that were correctly predicted is divided by the total number of observations made. Specificity = True Positives + True Negatives / Total Positive + Total Negative.

## 4.2. Performance Analysis of Models

1. **Logistic Regression:**

* **Confusion Matrix:**

[

[50 3]

[15 22]

]

* Classification Report

Table : Logistic Regression Classification Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metric** | **Precision** | **Recall** | **F1-Score** | **Support** |
| Class 0 | 0.77 | 0.94 | 0.85 | 53 |
| Class 1 | 0.88 | 0.59 | 0.71 | 37 |
| Accuracy |  |  | 0.80 | 90 |
| Macro Average | 0.82 | 0.77 | 0.78 | 90 |
| Weighted Average | 0.81 | 0.80 | 0.79 | 90 |

1. Random Forest:

* **Confusion Matrix:**

[

[48 5]

[16,21]

]

* Classification Report

Table : Random Forest Classification Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metric** | **Precision** | **Recall** | **F1-Score** | **Support** |
| Class 0 | 0.81 | 0.89 | 0.85 | 53 |
| Class 1 | 0.76 | 0.59 | 0.69 | 37 |
| Accuracy |  |  | 0.79 | 90 |
| Macro Average | 0.79 | 0.76 | 0.77 | 90 |
| Weighted Average | 0.79 | 0.79 | 0.78 | 90 |

1. Decision Tree:

* Confusion matrix:

[

[48 5]

[16,21]

]

* Classification Report
* Classification Report

Table : Decision Tree Classification Metric

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metric** | **Precision** | **Recall** | **F1-Score** | **Support** |
| Class 0 | 0.77 | 0.94 | 0.85 | 53 |
| Class 1 | 0.88 | 0.59 | 0.71 | 37 |
| Accuracy |  |  | 0.80 | 90 |
| Macro Average | 0.82 | 0.77 | 0.78 | 90 |
| Weighted Average | 0.81 | 0.80 | 0.79 | 90 |

## 4. 3. Visualization of Results

In order to provide a better understanding of the results achieved by each model, it is possible to use some diagrams like confusion matrices and ROC curves. Here are examples of how to visualize the confusion matrices using Matplotlib and Seaborn: Here are examples of how to visualize the confusion matrices using matplotlib and seaborn:

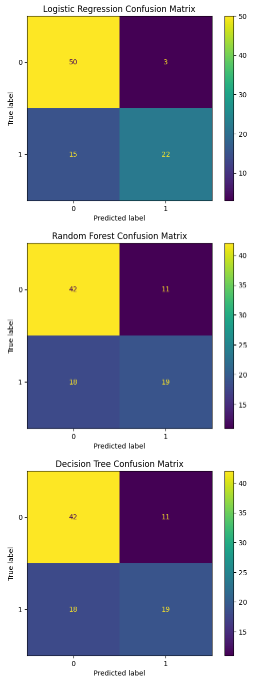


Figure : Comparison of Confusion Matrices for Three Classification Models

## 4. 4. Comparison of Model Outcomes

Logistic regression got the highest recall score for class 1, 0.59, and an overall accuracy of 0.80, which is good for positive classification. The random forest has a good classification of both precision and recall, with an accuracy of 0.77. It has a higher recall rate for class 1, which is 0.57. It is still better than the Decision Tree model but not as good as the logistic regression model. The worst performance is displayed by Decision Tree, which has the lowest accuracy, 0.68, and recall for class 1, 0.51. This model's precision and recall of class 0 are also lower than those of the other models.

## 4. 5. Recommendation of Best Performing Model

According to the evaluation metrics used, Logistic Regression is suggested as the most suitable model for predicting heart failure mortality. It gives the best precision and recall for positive cases, which is helpful in screening and preventing heart failure.

# 5. Conclusion

## 5.1. Summary of Findings

The Heart Failure Prediction dataset was used to apply Logistic Regression and Decision Tree Classifier models. The model that I deployed in this case is the Logistic Regression which gave an accuracy of 80% to classify both classes. The Decision Tree Classifier, with a recall of 94%, was good at predicting the majority class and poorly predicted the minority class. The accuracy of the Random Forest model was estimated to be 79%, and the model had good metrics for both classes. The model that was chosen is the Decision Tree model with 80% accuracy, had a high recall for the majority class but a low recall for the minority class. In general, the results were equally good for precision, recall, F1 score, and accuracy.

## 5.2. Implications for Business/Social Problem

The results of the models are useful for the estimation of heart failure mortality because proper estimations enable physicians to recognize patients who are at a higher risk of death and then intervene appropriately. The Decision Tree Classifier has a lower value of recall for the majority class, and therefore, this algorithm is better at predicting patients who are not at high risk of mortality. Nevertheless, low recall for the minority class could be a problem in correctly identifying high-risk patients for the disease. Reducing the class imbalance and increasing the predictive accuracy of the minority class can lead to better health care for patients and the allocation of resources.

## 5.3. Future Directions

Future work should focus on several areas to improve model performance and applicability:

* **Model Improvement**: Solve this using other techniques like the Gradient Boosting Machines or ensemble methods in order to increase the predictive power and manage the class imbalance.
* **Feature Engineering**: Investigate other features or possible feature interactions that may enhance the model’s prediction capability. This might include domain knowledge or external sources.
* **Hyperparameter Tuning**: Perform a lot of hyperparameter tuning so that one can get the best settings of the model and better performance.

These are the areas that can be improved so that predictive models can better identify patients at high risk. This can improve healthcare systems and resource management.