

<Automation Lab>

<Fall 2024>

Heart

Failure

Data Set

Olivia R., Somto O., Zach G.



## <Step 1>

## <Problem Definition>

### Objective:

To understand what is causing a Death Event. By understanding what causes a Death Event it may help put a focus and emphasis on that specific issue. Understanding what causes Death Events can ultimately help reduce the number of Death Events.

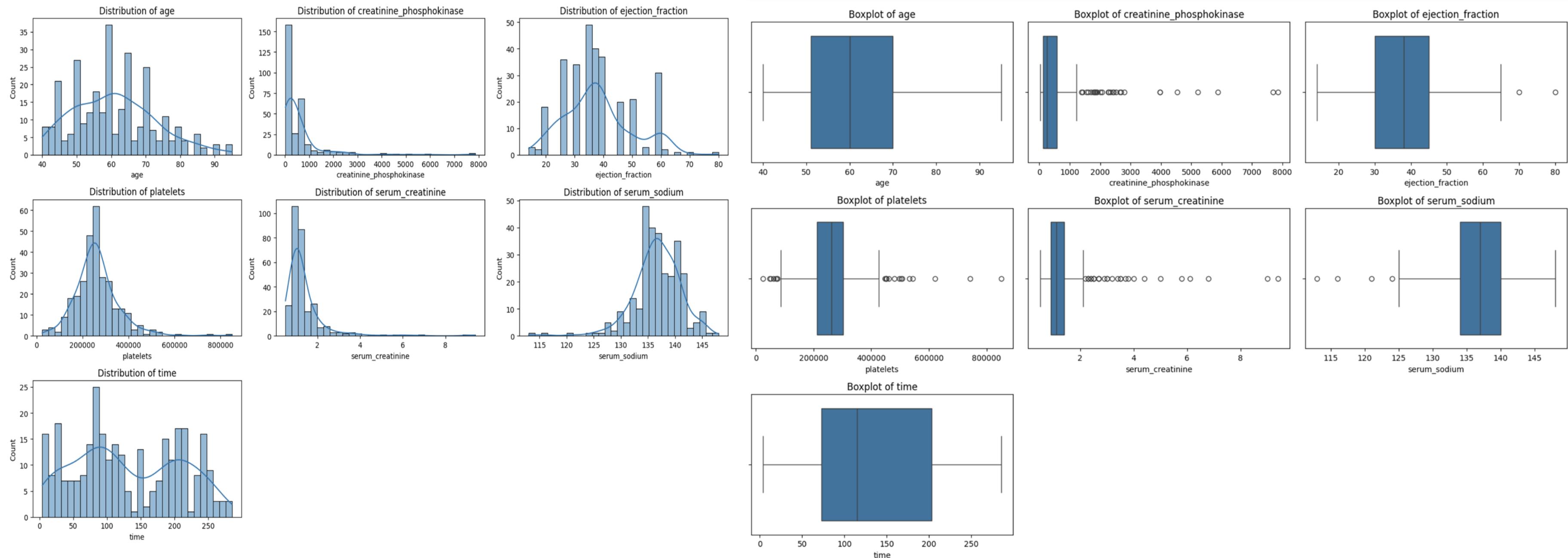
### Task:

We will perform a classification analysis to determine how the independent variables impact the dependent variable Death Event. We are doing this to determine what events determine the change of a death event.

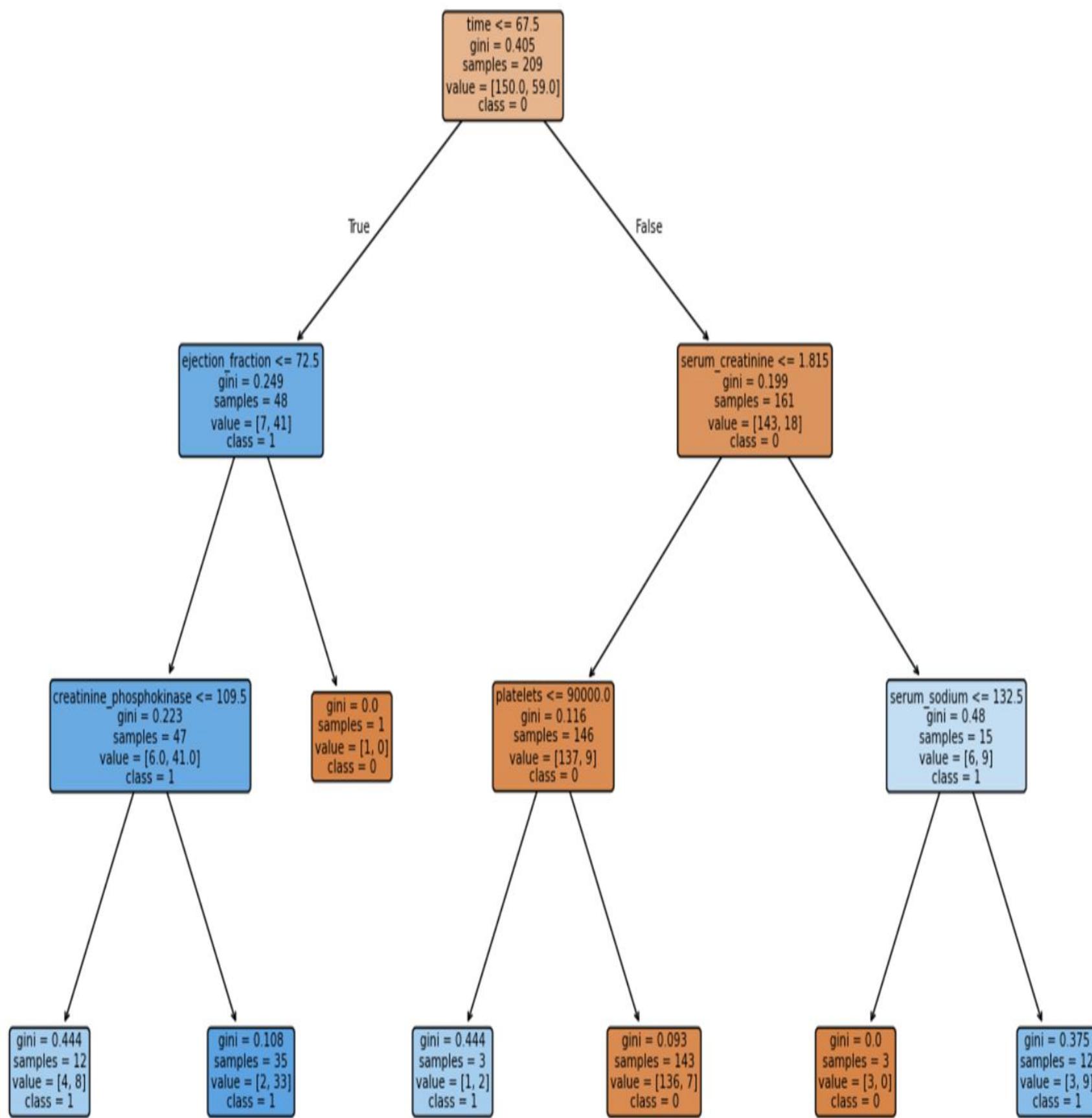
## <Step 2>

## <Exploratory Data Analysis (EDA)>

- Zero missing values detected
- Creatinine Phosphokinase and Serum Creatinine have the most outliers and have a right skewed graph
- Null Hypothesis ( $H_0$ ): There is no significant relationship between the categories (features such as age, anaemia, creatinine phosphokinase, diabetes, etc.) and the likelihood of a Death Event.
- Alternative Hypothesis ( $H_1$ ): At least one of the categories (features) significantly impacts the likelihood of a Death Event.



## <Step 2>



## <Exploratory Data Analysis (EDA)>

If  $\text{time} \leq 67.5$ , the samples are sent to the left branch.  
If  $\text{time} > 67.5$ , the samples are sent to the right branch.

Left branch 1:

If  $\text{ejection fraction} \leq 72.5$ , the samples are further split into the next level.  
If  $\text{ejection fraction} > 72.5$ , only one patient is classified, with no death event (class 0).

Left branch 2:

If  $\text{serum creatinine} \leq 1.815$ , the samples are split further.  
If  $\text{serum creatinine} > 1.815$ , there is a strong majority of no death events (class 0)

Right branch 1:

If  $\text{creatinine phosphokinase} \leq 109.5$ , the majority of patients experience no death events (class 0).

If  $\text{creatinine phosphokinase} > 109.5$ , there's a higher proportion of death events, leading to further splits.

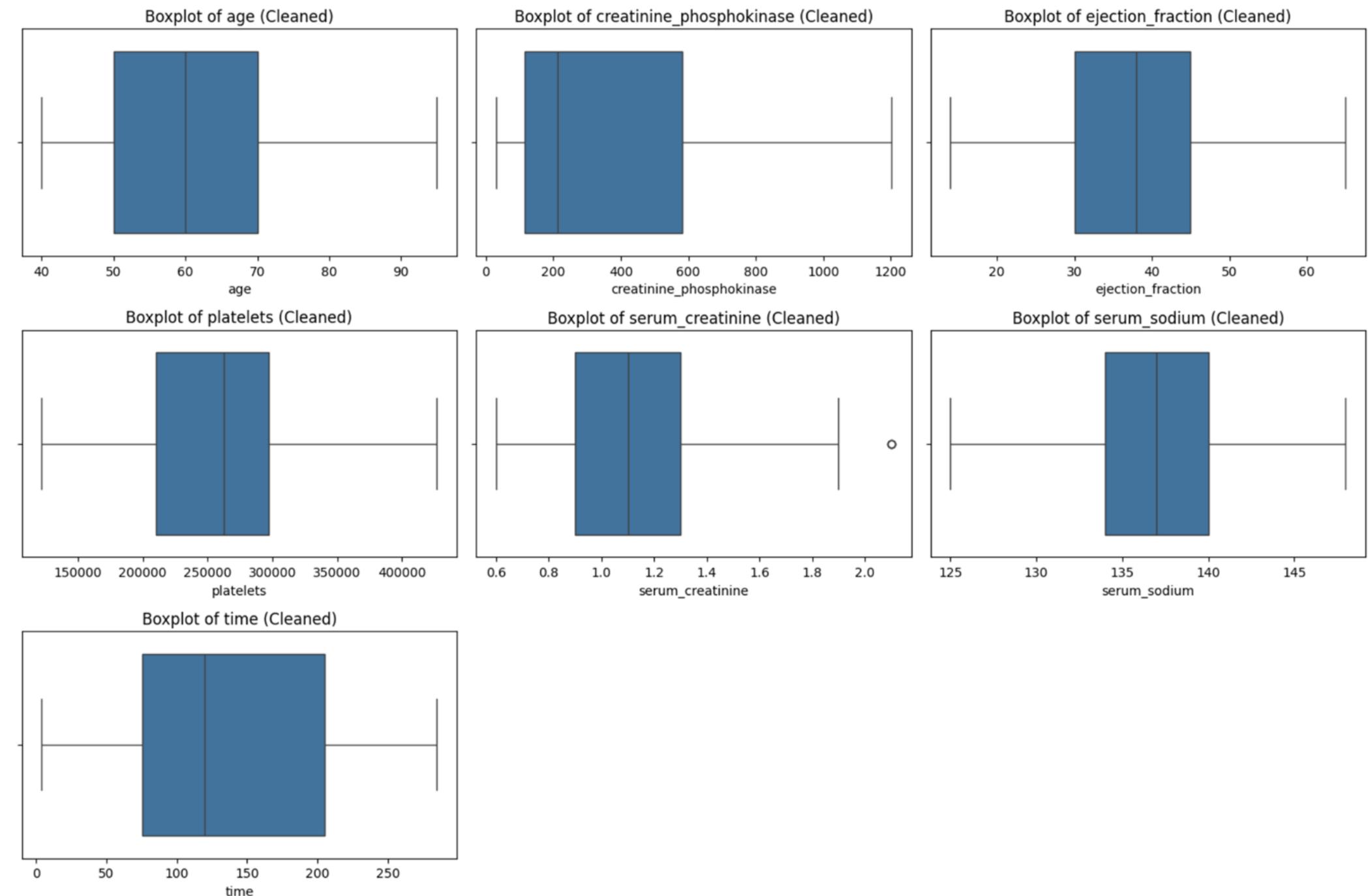
Right branch 2:

If  $\text{serum sodium} \leq 132.5$ , the risk of a death event (class 1) is high.  
If  $\text{serum sodium} > 132.5$ , the risk of a death event is lower.

## <Step 3>

## <Data Preprocessing>

- Dropped any duplicates
- Detected and handled outliers by removing them front the dataset
  - 29 from creatinine\_phosphokinase
  - 2 from ejection\_fraction
  - 18 from platelets
  - 23 from serum\_creatinine
  - 3 from serum\_sodium
- Check for Erroneous Data (negative values where not applicable)
- Create a feature for age group



<Step 4>

<Feature Selection>

```
X = data.drop(['DEATH_EVENT'], axis=1)  
y = data['DEATH_EVENT']
```

```
X_train shape: (209, 12), y_train shape: (209, )  
X_test shape: (90, 12), y_test shape: (90, )
```

### Split Data into Features and Target

- Features (**X**): All columns except DEATH\_EVENT.
- Target (**y**): The DEATH\_EVENT column.

Dropped the Target Variable (**DEATH\_EVENT**) from Features to avoid data leakage.

Used the **train\_test\_split Function** from `sklearn.model_selection` to separate data into training and testing sets.

## <Step 5>

## <Modeling>

### **Data Splitting:**

- Divided the dataset into **70% training** and **30% testing** using `train_test_split`.
- Maintained reproducibility with `random_state=42`.

### **Model Training:**

- Used a **Decision Tree Classifier** with a maximum depth of 3 to avoid overfitting.
- Trained the model on the **training set (`X_train`, `y_train`)**.

### **Model Visualization:**

- Plotted and visualized the **Decision Tree**, showing key splits and decision paths.

### **Prediction:**

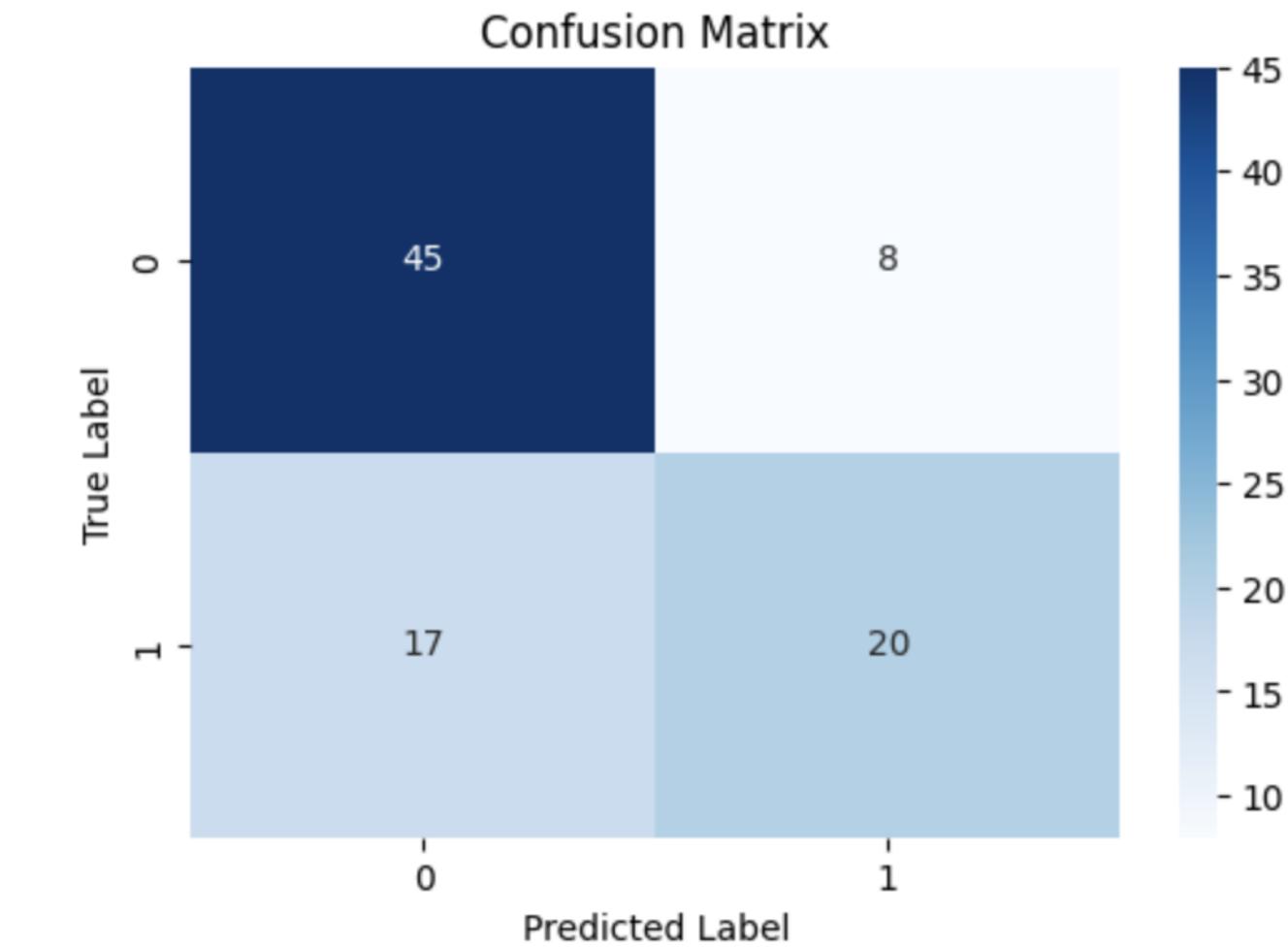
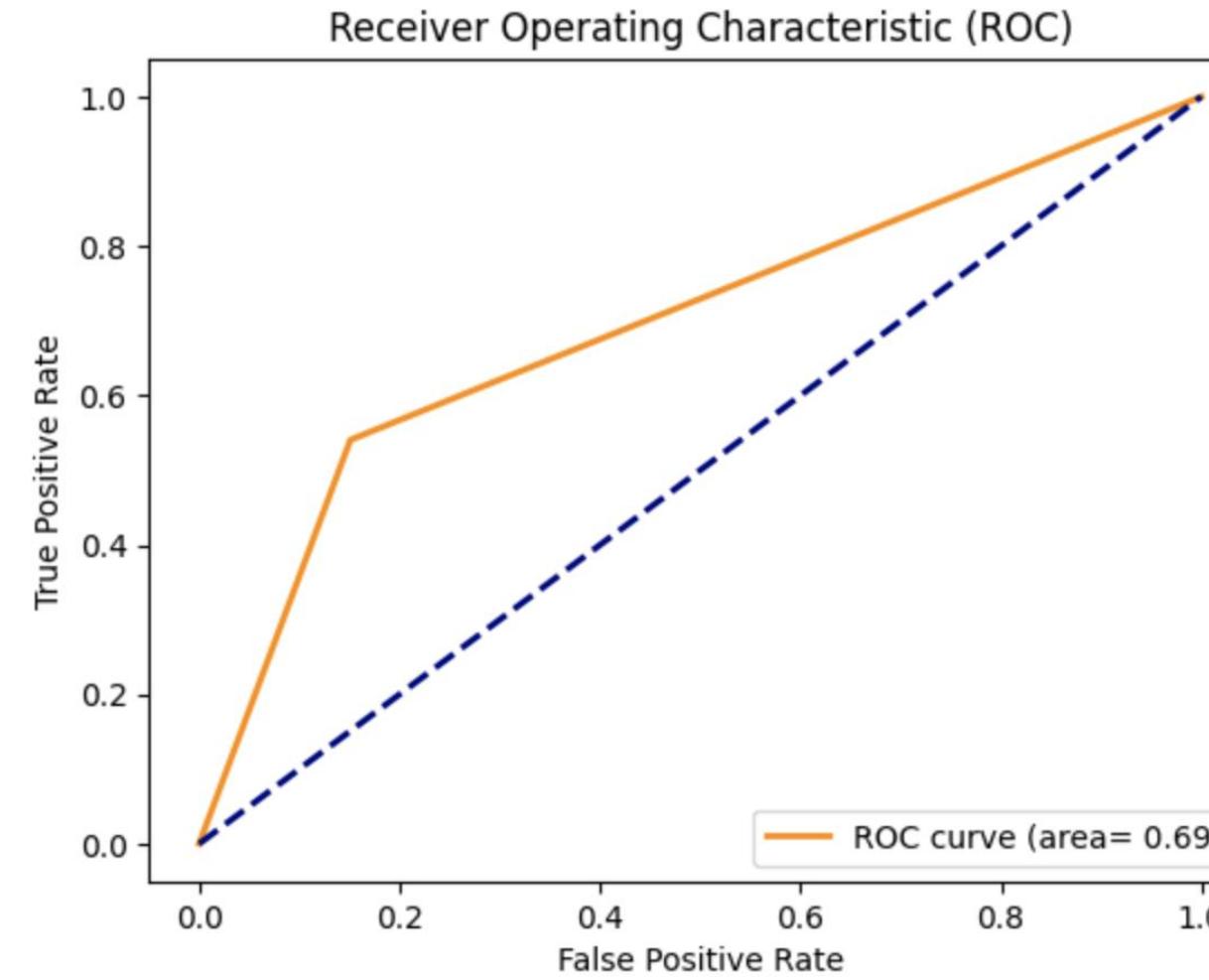
- Predicted outcomes (`y_pred`) on the **test set (`X_test`)**.

### **Model Evaluation:**

- Calculated model accuracy: `{accuracy:.2f}` (replace with the printed accuracy).
- Generated a **classification report** to evaluate precision, recall, F1-score, and support for each class.

## <Step 6>

## <Model Evaluation>



- An ROC curve (area) of 0.69 indicates that the model is better than random guessing but still leaves room for improvement
- For the confusion matrix:
  - **False Negatives:** The model misses 17 "Death" cases, highlighting low recall (54.1%) for the positive class, which is critical in high-stakes contexts.
  - **False Positives:** There are 8 incorrect "Death" predictions; while less critical, they could lead to unnecessary actions.
  - **Class Imbalance:** Likely skewed towards "No Death," affecting the model's ability to detect minority cases.
  - **Model Simplicity:** A max depth of 3 limits the model's complexity, potentially causing misclassifications.

## <Step 7>

## <Model Interpretation and Insights>

### **ROC AUC Score (0.69):**

- Moderate ability to discriminate between death and survival events.
- Not sufficient for reliable predictions in high-stakes healthcare settings.

### **Confusion Matrix Insights:**

- 17 false negatives, indicating missed death events.
- Risks of not identifying patients needing critical intervention.

### **Key Features:**

- **Serum creatinine** and **ejection fraction** align with medical knowledge of heart failure.
- Model captures some high-risk patterns but shows variability in **platelets** and **serum sodium** predictions.

### **Model Context:**

- False negatives are critical in healthcare; missing death events can have severe consequences.
- Model conservatively identifies survival cases but risks overlooking critical patients.

### **Next Steps:**

- Focus on reducing false negatives for better detection.
- Explore resampling or cost-sensitive methods to handle class imbalance.

<Thank You!>  
Any Questions?