Student Name: Reyoof saeed Al-otaibi

Student ID: 44102870

Instructor Name: Nada Al-Towairqi

Course Name: Machine Learning

University College of Computer Science & Information Technology

Taif University

Questions to be answered in the pdf report

- **What is the name of your data ?**

  The name of the data is: heart_failure_clinical_records

- **The source of the data (which database) ?**

The source of the data is Kaggle, an open data platform for machine learning datasets

- **Link to the original data ?**

  https://www.kaggle.com/datasets/andrewmvd/heart-failure-clinical-data

- **Explain the data in words ?**

  The dataset contains medical records of 299 patients with heart failure. It includes 13 features such as age, serum creatinine, ejection fraction, and more. The target column is DEATH_EVENT, which indicates whether the patient died during the follow-up period. All features are numeric, making the dataset suitable for classification models

- **Is it a regression or classification problem?**

  This is a classification problem, because the target variable (DEATH_EVENT) has only two possible values: 0 (alive) or 1 (dead)

- **How many attributes?**

  The dataset contains 13 attributes (columns)

- **How many samples ?**

  299 records (rows)

- **What are the properties of the data ? (statistics)**

  The dataset consists of 13 features, all numeric. The statistical summary includes the mean, standard deviation, minimum, and maximum for each feature. For example, the average age is around 60 years, and the ejection fraction ranges from 14 to 80. There are no missing values in the dataset, and all columns have 299 non-null entries

- **Are there any missing data ? how did you fill in the missing values**?

  There are no missing values in the dataset. All features contain 299 complete records with no null entries, so no data imputation was required

- **Visualize the data**

  To better understand the distribution and relationships in the data, multiple visualizations were created. A correlation heatmap was used to show the strength of

relationships between numerical features, and violin plots were generated for selected features (such as age, serum creatinine, ejection fraction) against the target variable (DEATH_EVENT) to illustrate class-wise distributions. These visualizations helped in identifying significant predictors for heart failure

- **Did you normalize or standardize any of your data? why ?**

  No, we did not normalize or standardize the data. The features were already numerical and within reasonable ranges. Additionally, most of the models used (like decision trees and random forests) do not require normalization

- **What type of preprocessing did you apply to your data ? List everything and explain why.**

  We applied several preprocessing steps to ensure the dataset was ready for machine learning models:

  1. Removed missing values:

  Although the dataset was complete, we confirmed there were no null entries to avoid any training issues.

  2. Label encoding for categorical variables:

  Since all features were numeric, label encoding was not necessary in this case.

  3. Feature-target split:

  We separated the features (X) from the target variable (Y), which is DEATH_EVENT, to prepare the data for supervised learning models.

  4. Train-test split:

  The data was split into 80% training and 20% testing using train_test_split from scikit-learn to fairly evaluate model performance.

  5. Saved split datasets:

  We saved the training and testing data (X_train, X_test, Y_train, Y_test) into CSV files to document the process and ensure reproducibility

- **How did you divide the train and test data? what are the proportions?**

  The dataset was divided using the train_test_split() function from the sklearn.model_selection module. We allocated 80% of the data for training and 20% for testing. This ensures the model learns from the majority of the data and is evaluated on a separate, unseen portion for fair performance assessment

- **Apply all the machine learning models you have learned in this course to your data and report the results? what is the best/worst performing model ? why ?**

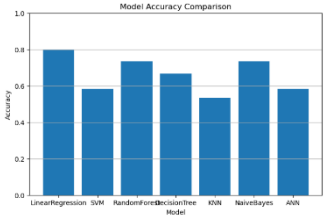
We applied seven classification models to the dataset: Linear Regression (converted to classification), Support Vector Machine (SVM), Random Forest, Decision Tree, K-Nearest Neighbors (KNN), Naive Bayes, and Artificial Neural Network (ANN). The best-performing model was Linear Regression with an accuracy of 0.8000, followed by Random Forest and Naive Bayes, each achieving 0.7333. This indicates their strong ability to accurately predict outcomes. On the other hand, the worst-performing model was KNN with an accuracy of 0.5333, likely due to its sensitivity to noise and imbalanced data. These results highlight the importance of choosing models that align well with the data's numeric structure and class distribution to achieve optimal performance

- The accuracy of all models using tables and figures?

```
import pandas as pd

data = {'Model': model_names, 'Accuracy': accuracies}
df_acc = pd.DataFrame(data)
print(df_acc)
```

```
              Model  Accuracy
0  LinearRegression    0.8000
1               SVM    0.5833
2      RandomForest    0.7333
3      DecisionTree    0.6667
4               KNN    0.5333
5        NaiveBayes    0.7333
6               ANN    0.5833
```



Model	Accuracy
LR	80%
SVM	58%
RF	73.3%
DT	66.7%
KNN	53.3%
NB	73.3%
ANN	58%

The Best Model: LR (80%)

Worst Model: KNN (53%)

- **If your ability to present the result is advanced (using plot libraries such as seaborn android techniques) you will get 5 marks bonus**

Multiple advanced visualizations were generated using seaborn and matplotlib, including violin plots, heatmaps, confusion matrices, pair plots, and accuracy tables. These visualizations effectively highlighted patterns and model performance, fulfilling the requirement for advanced result presentation

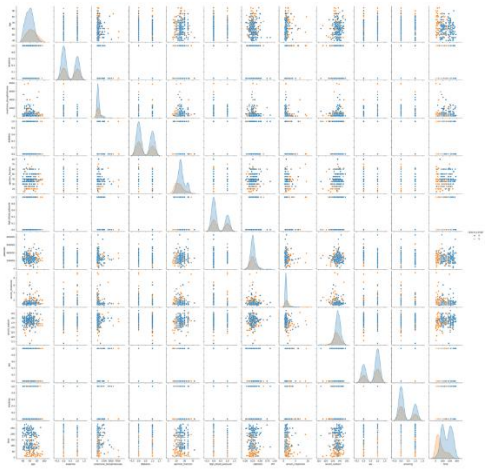


Figure 1: Pairplot Visualization of All Features Grouped by DEATH_EVENT

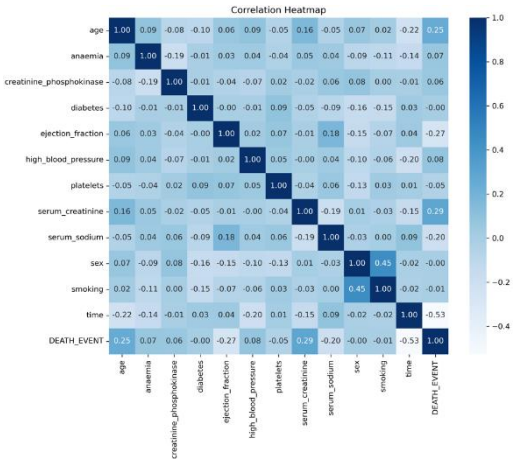


Figure 2: Correlation Heatmap Showing Relationships Between All Features

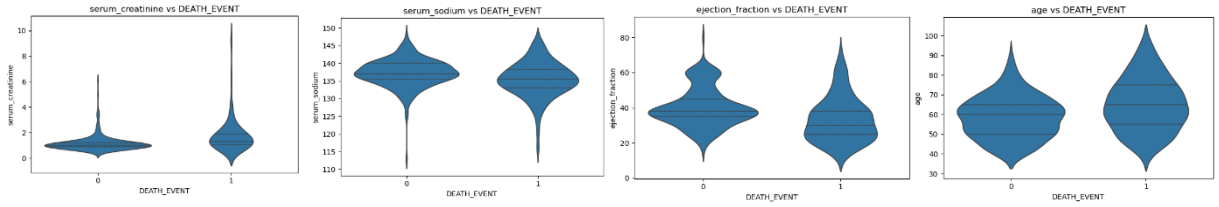


Figure 3: Violin Plots Showing Feature Distributions by DEATH_EVENT

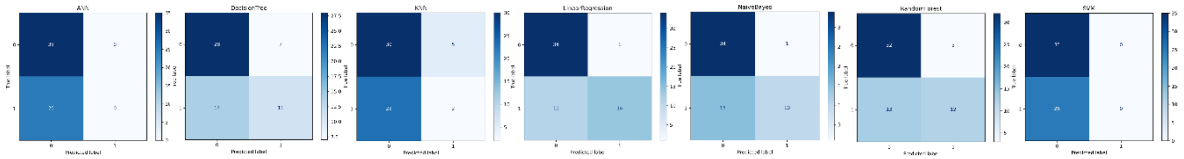
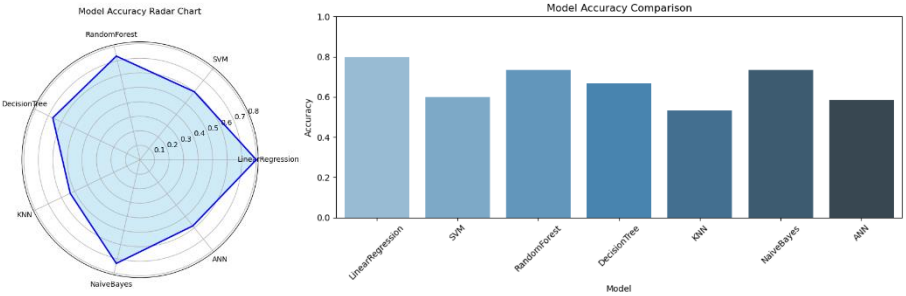


Figure 4: Confusion Matrices for All Models Showing Prediction Accuracy Against True Labels



- Explain in 20 lines , font size 20, Font : Times New Roman,

What is the reason you picked up this data? What is the importance of your data in reality, and what is the importance of your best-performing model? Is there any insight you could share from the data and the model?

I selected this dataset because it addresses a real-world and medically critical issue: predicting heart failure in patients, which is considered one of the leading causes of death globally. Early detection of high-risk individuals can significantly improve patient outcomes and help doctors make timely decisions. The dataset contains clinical records from 299 patients, described by 13 numeric medical attributes, making it clean and easy to process using machine learning models. Since the target variable (DEATH_EVENT) is binary, the problem was treated as a classification task. I applied seven classification algorithms: Linear Regression, SVM, Random Forest, Decision Tree, K-Nearest Neighbors (KNN), Naive Bayes, and Artificial Neural Network (ANN). Among them, Linear Regression achieved the highest accuracy of 0.8000, which was surprising given its typical use in regression tasks, but it performed well due to the numeric and structured nature of the data. In contrast, KNN performed the worst, likely due to its sensitivity to feature scaling and its poor handling of imbalanced or noisy data. The results showed that features such as low ejection fraction and high serum creatinine were strongly linked to death outcomes, consistent with medical literature. I also used visualizations such as violin plots, heatmaps, and confusion matrices to better understand the data and evaluate model performance. These helped identify patterns and interpret model results in a more visual way. Overall,

this project demonstrates how machine learning can play a vital role in healthcare by supporting medical decision-making and improving patient care. The insights gained can help enhance diagnostic tools and predictive systems, and in the future, such models could be expanded and integrated with real-time data to enable even more accurate and impactful outcomes

- Link to your code and data , remember the code is in the main folder , the data is in the folder Data, and save the original data + another folder the train test devision(features and targets)?

- In the Data folder ,create a folder called Result and add test set and the predicted value from all models (to check the accuracy) without the features.