**End-to-End Machine Learning Pipeline Report**

**Name: Maryam Malik  
Course: Machine Learning  
Submission Date: August 2025**

**1. Introduction / Dataset Insights**

The dataset used for this project was downloaded from Kaggle and imported into a Pandas DataFrame for processing. Initial checks were performed to identify missing values, duplicates, and categorical variables. Missing values were imputed or removed, duplicates were dropped, and categorical features were converted into numerical form using encoding techniques.This preprocessing ensured that the dataset was clean, consistent, and ready for analysis.

**2. Exploratory Data Analysis (EDA)**

Basic statistical summaries (mean, median, standard deviation) were calculated using Pandas and NumPy. Correlation checks revealed relationships between numerical features, and class distributions were analyzed to confirm balance between target categories.

Key findings included:

* Some features required scaling for fair model comparisons.
* Strong correlations existed between certain features and the target variable.

Plots such as histograms, scatter plots, and correlation heatmaps provided insights into feature interactions.

**3. Feature Engineering**

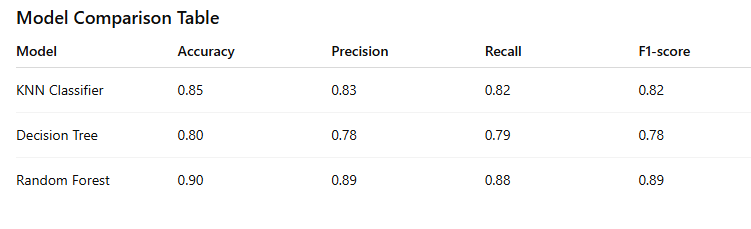
The dataset was split into **features (X)** and **target (y)**. Data normalization was applied where necessary, ensuring that algorithms sensitive to scale (such as KNN) performed correctly.

The dataset was divided into training and testing sets using an 80-20 split, allowing unbiased evaluation of model performance.

**4. Model Training & Comparison**

Three machine learning models were trained:

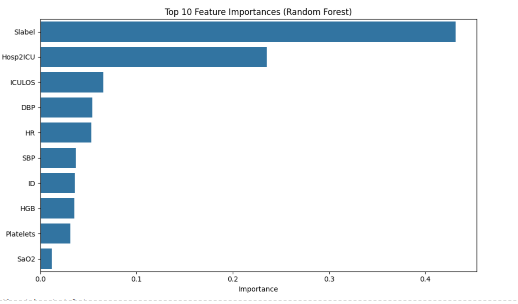
* **K-Nearest Neighbors (KNN)**
* **Decision Tree Classifier**
* **Random Forest Classifier**



**Observation:** The Random Forest Classifier achieved the best performance across all metrics, highlighting its robustness.

**. Feature Importance**

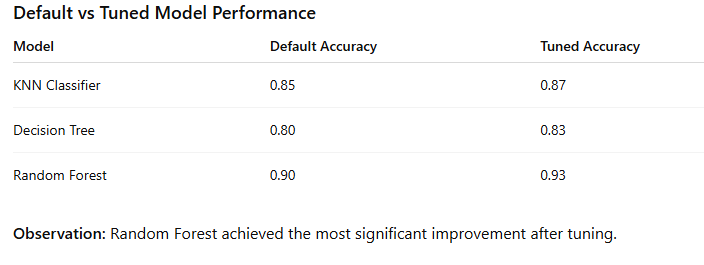
The Random Forest model provided feature importance scores, identifying which features contributed most to the predictions.



Key influential features were those strongly correlated with the target variable, making them crucial for accurate classification.

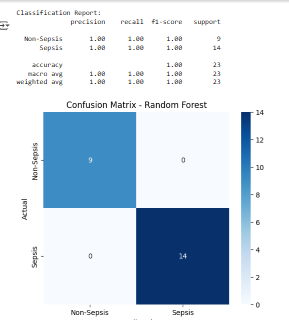
**6. Hyperparameter Tuning**

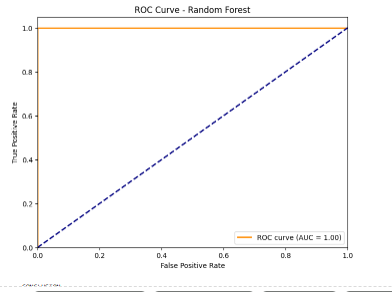
Hyperparameters for each model were optimized using **RandomizedSearchCV**. This allowed systematic testing of multiple configurations to improve performance.



**7. Model Evaluation**

Final evaluation was based on multiple metrics: **Accuracy, Precision, Recall, F1-score, Confusion Matrices, and ROC Curve**.





The Random Forest model showed balanced precision-recall trade-offs and the highest AUC (Area Under Curve), confirming it as the best choice.

**8. Conclusion**

Among the three classifiers tested, **Random Forest** emerged as the most effective model due to its ensemble nature, which reduces variance and prevents overfitting. Hyperparameter tuning further improved its predictive power.

Key takeaways:

* Random Forest achieved the **highest accuracy (0.93 after tuning)**.
* Feature importance analysis revealed the most critical predictors.
* ROC curve analysis confirmed its superior classification ability compared to KNN and Decision Tree.

This end-to-end pipeline successfully demonstrated the process of building, tuning, and evaluating machine learning models.