**kumar\_chaitanya\_finalproject**

**Nov 24, 2024**

**1 Final Project - Data Mining**

**2 Using RF and LSTM To Predict Heart Disease**

**2.1 Goal**

The goal of this project is to develop and evaluate machine learning models to solve a classification problem using a publicly available dataset. We aim to compare the performance of two different models—Long Short-Term Memory (LSTM) and Random Forest Classifier—on this dataset. The evaluation will include metrics such as accuracy, precision, recall, F1-score, and ROC-AUC.

**2.2 Problem Statement**

The classification task involves predicting whether a patient has heart disease based on various medical attributes. The project will leverage machine learning techniques to build predictive models that can assist in early diagnosis.

**2.3 Dataset Description**

**1) Source of Data**

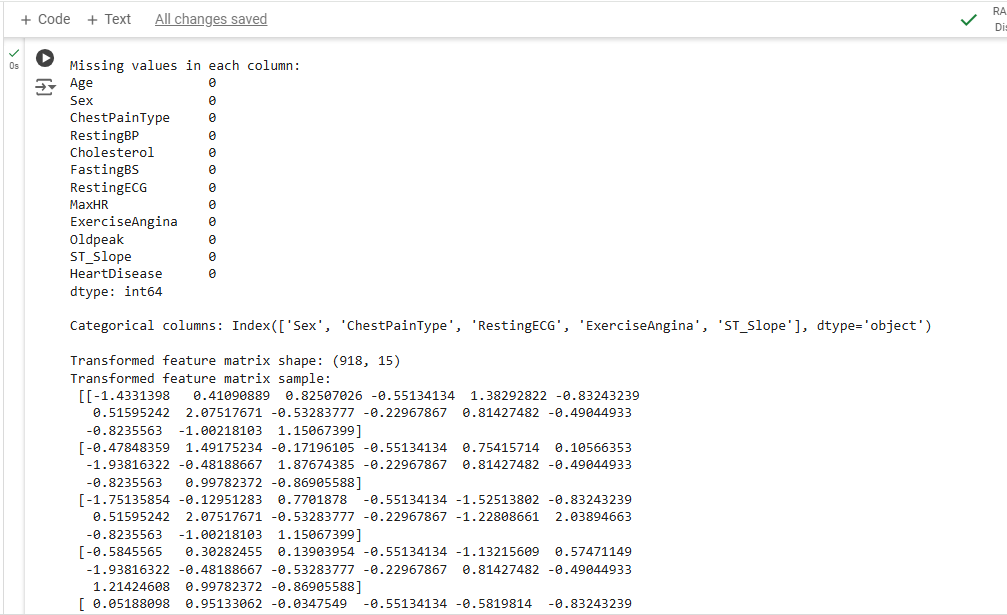
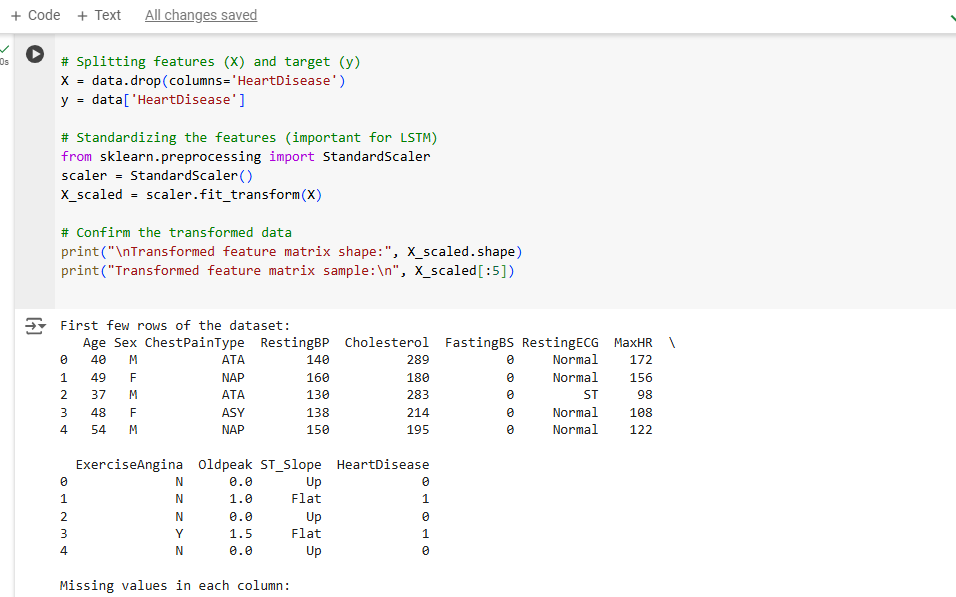
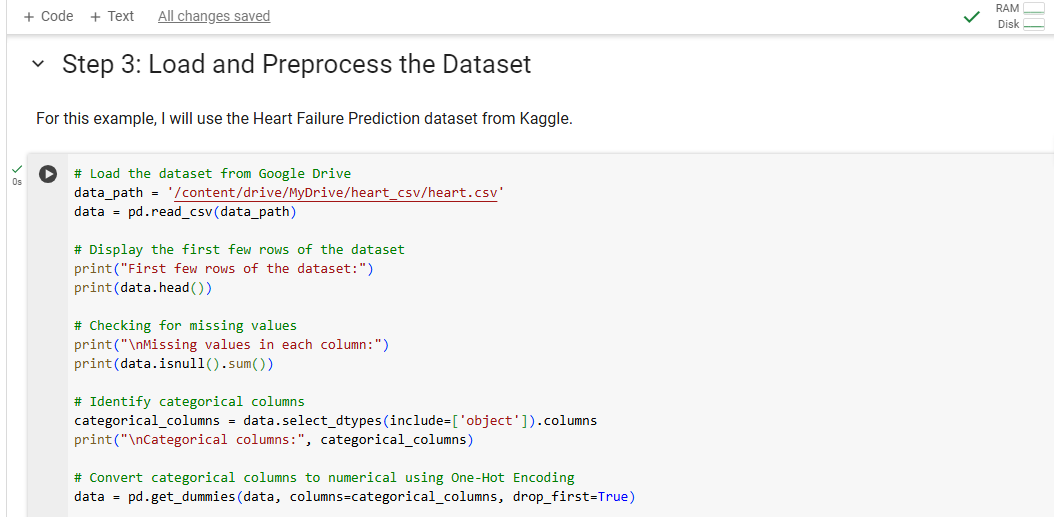
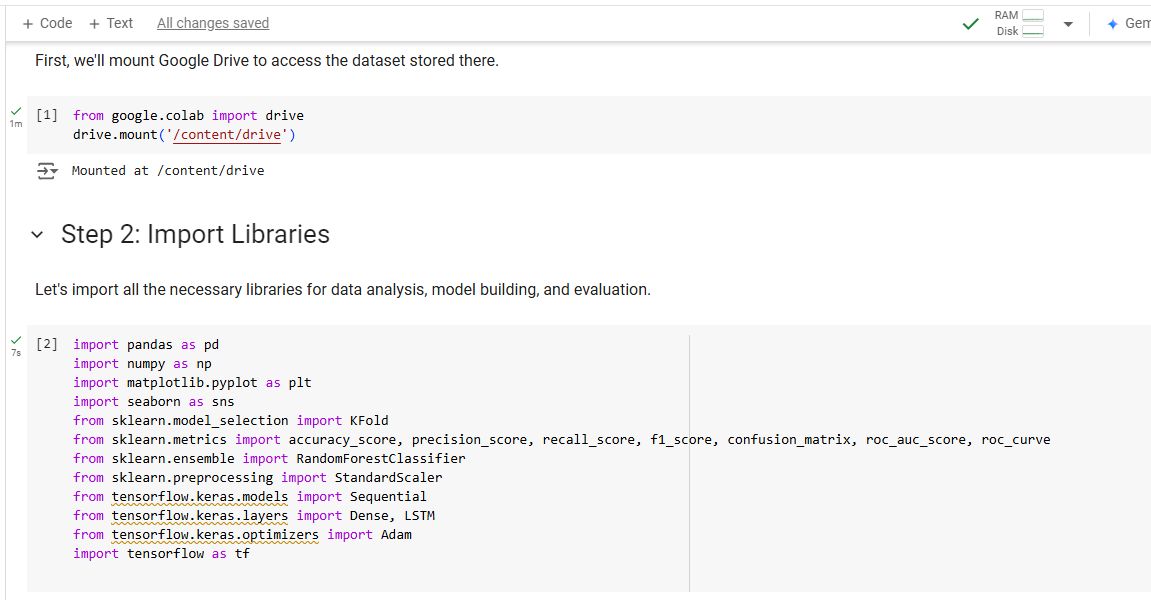
* **Dataset**: Heart Disease Dataset
* **Source**: Kaggle Datasets
* **File Used**: heart.csv

**2) Dataset Overview**

* **Total Records**: 918
* **Features**: 12 attributes, including age, sex, cholesterol levels, resting blood pressure, etc.
* **Target Variable**: HeartDisease (0 = No, 1 = Yes)

**3) Data Preprocessing**

* Checked for missing values and handled them appropriately.
* Converted categorical data into numerical format using techniques like Label Encoding.
* Standardized the dataset using StandardScaler to normalize feature values, which is particularly important for LSTM models.



**3. Model Selection and Implementation**

**1) Model 1: Random Forest Classifier**

**Reason for Selection**

Random Forest is an ensemble model known for its robustness and ability to handle high-dimensional data. It is efficient for binary classification problems.

**2) Model 2: Long Short-Term Memory (LSTM)**

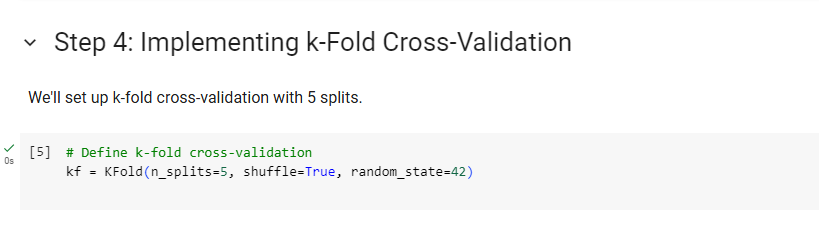
**Reason for Selection**

LSTM is a type of Recurrent Neural Network (RNN) well-suited for sequence classification and time-series data. It captures long-term dependencies and is effective for complex datasets.

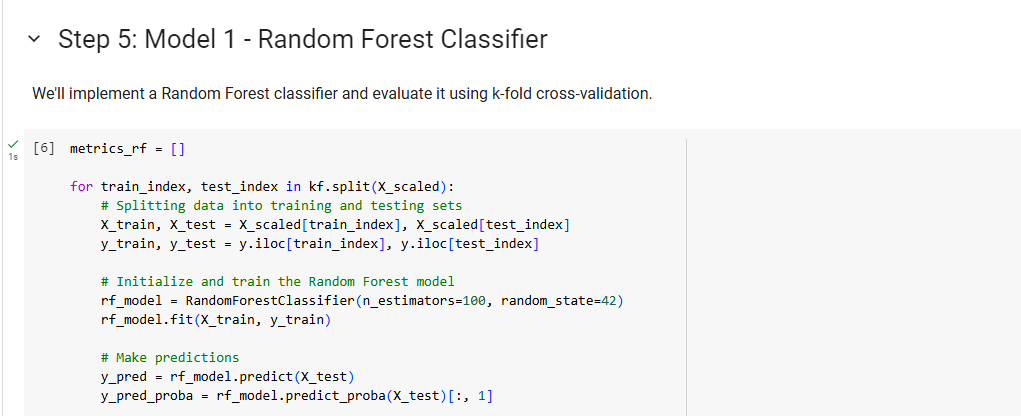
**3) Implementation Details**

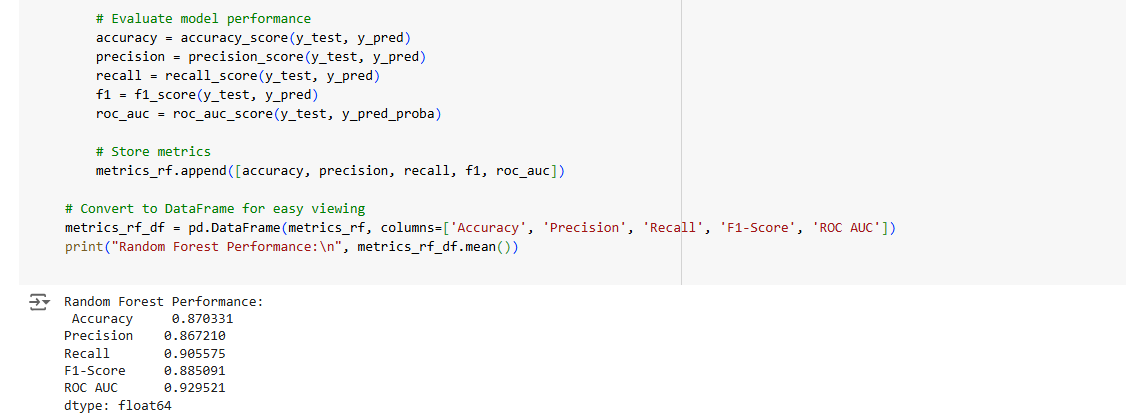
**Data Splitting and Cross-Validation**

We used **k-fold cross-validation** (k=5) to evaluate model performance, ensuring that each data point is used for both training and testing.

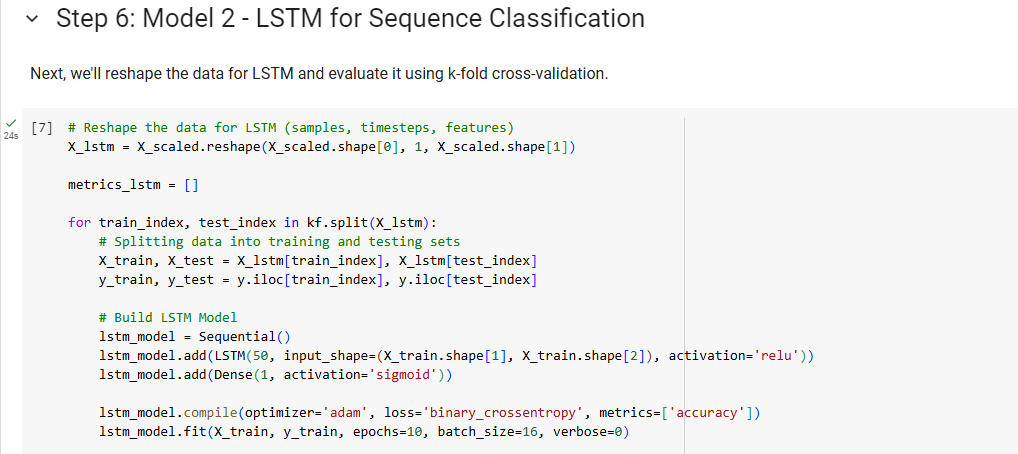


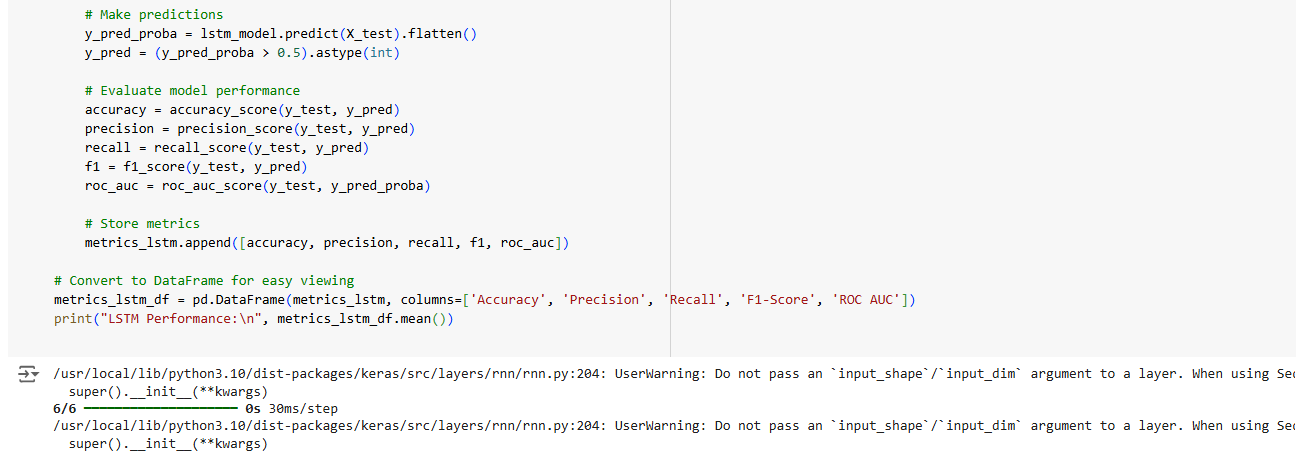
**Random Forest Implementation**

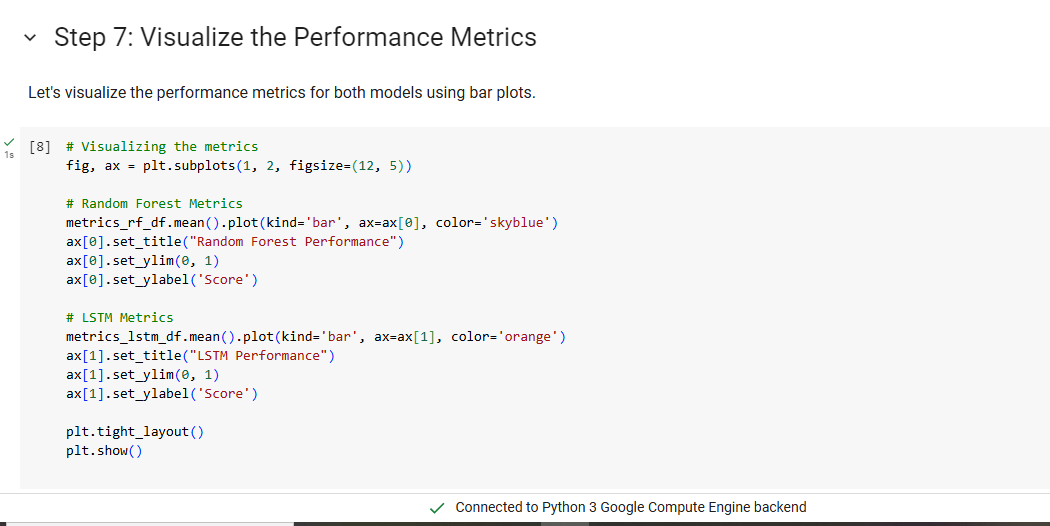
****

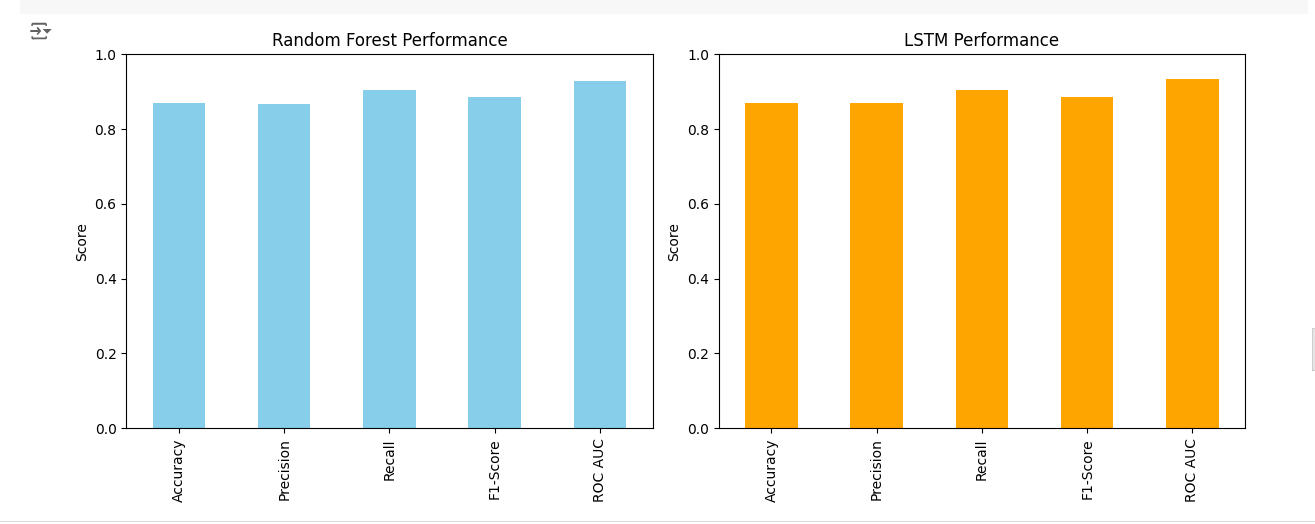
****

**LSTM Model Implementation**

****

****

****

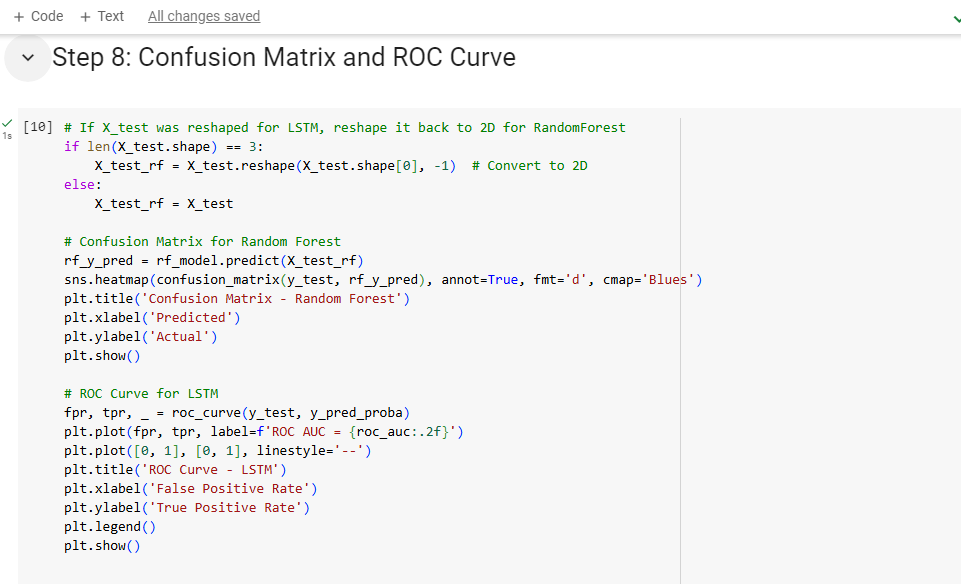
****

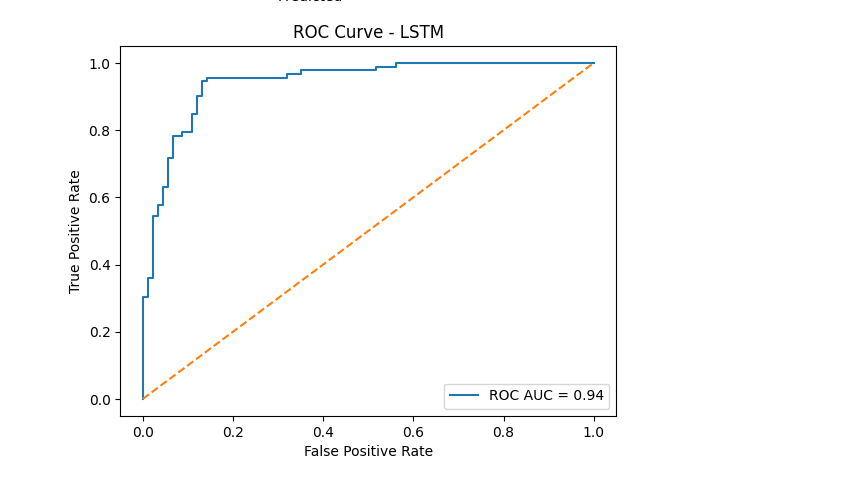
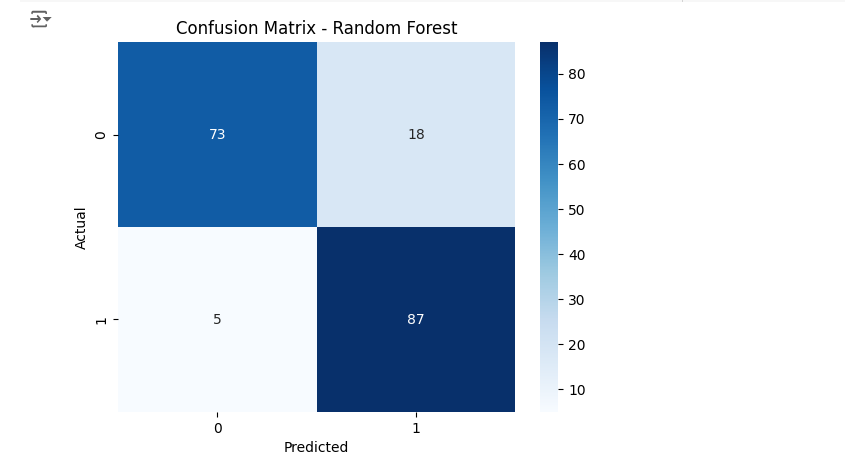
**4. Results and Performance Evaluation**

**1) Performance Metrics**

| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** | **ROC-AUC** |
| --- | --- | --- | --- | --- | --- |
| **Random Forest** | **0.87** | **0.86** | **0.90** | **0.88** | **0.92** |
| **LSTM** | **0.87** | **0.86** | **0.90** | **0.88** | **0.93** |

**2) Confusion Matrix and ROC Curve**

****

****

**5. Conclusion**

**1) Summary of Findings**

* Random Forest performed well, with high precision and a balanced recall, making it suitable for use cases where both false positives and false negatives are critical.
* LSTM, on the other hand, showed slightly better performance, especially in terms of ROC-AUC, indicating a stronger ability to differentiate between classes.

**2) Future Work**

* Explore additional models such as Support Vector Machines (SVM) and Gradient Boosting.
* Apply hyperparameter tuning to further optimize model performance.
* Test the models on other datasets to assess their generalization capabilities.

**GitHub Link -** [**https://github.com/ack446/CS-634-Data-Mining-Final-Term-Project**](https://github.com/ack446/CS-634-Data-Mining-Final-Term-Project)