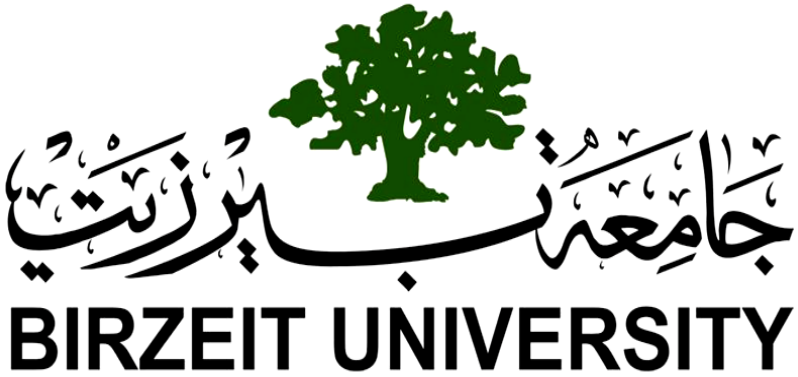


Faculty of Engineering & Technology
Electrical & Computer Engineering Department

Artificial Intelligence-ENCS3340

Project Report

Give Life: Predict Blood Donations

Instructor:
Dr. Adnan Yahia

Prepared By:
Abd Khuffash-1200970

Rana Deek-1201724

Section:
1

Date:
8-1-2023

The project aimed to predict blood donation behavior using various machine learning algorithms. The dataset contained information on past donation history, which was used to predict whether an individual would donate blood in March 2007. The project's primary goal was to compare different algorithms' performance in classifying individuals into donors and non-donors.

The Dataset was found on: drivendata.org / attached.

Tools and Libraries Used

The following tools and libraries were integral to the project:

- Python: The primary programming language.
- Pandas: For data manipulation and analysis.
- NumPy: For numerical operations on arrays.
- Scikit-learn: Provided machine learning algorithms and utilities for model training, evaluation, and hyperparameter tuning.
- Matplotlib and Seaborn: For data visualization.
- Jupyter Notebook: For interactive development and documentation.
- Google Colab uses Jupyter Notebooks, which are documents that can contain both code and rich text elements, like paragraphs, equations, and charts.

The Project was done via google Colab; u can find the code in the link below:(you don't need to download Python on your local machine to use Google Colab. Google Colab is a cloud-based service that provides a Python programming environment. It runs in your web browser and allows you to write and execute Python code without needing to install Python or any other programming tools on your computer.)

https://drive.google.com/drive/folders/1W8cFOWfYli2wDSoDtxdwp0_zlNCbICns?usp=drive_link

The dataset was loaded and preprocessed using Pandas. Missing values were removed, and the dataset was described to understand its characteristics. The target variable was renamed for clarity. The data was then split into training and testing sets, ensuring stratification based on the target variable.

Scaler was used to normalize the feature set to ensure that each feature contributes equally to the distance calculations and optimization algorithms, potentially improving the performance and speed of the machine learning models. However, the choice of scaler also depends on the specific characteristics of the dataset and the requirements of the machine learning model being used.

MinMaxScaler was used to scale the features, ensuring that all features contributed equally to the model training process.

Three different models were trained and evaluated:

1. Decision Tree Classifier: A non-linear model good for capturing complex patterns.
2. Naive Bayes: A probabilistic model, often effective in classification tasks.
3. Support Vector Classifier (SVC): Effective in high-dimensional spaces and with different kernel functions.
4. Artificial Neural Network(ANN): Good for learning from complex data and used for a variety of tasks.

Each model's hyperparameters were tuned using GridSearchCV to find the best-performing combination.

The models were evaluated based on their accuracy, precision, recall, and F1-score. Confusion matrices were plotted to visualize the true positives, true negatives, false positives, and false negatives. Receiver Operating Characteristic (ROC) curves were also plotted to assess the models' performance across different threshold settings.

Results:

- The best hyperparameters for each model were determined.
- The performance of each model was compared using various metrics.
- Decision Trees, Naive Bayes, and SVC showed varying levels of effectiveness in predicting blood donation.

Starting with each model:

1. Decision Tree:
   Classification Report before Cross-validation/hyperparameters found:

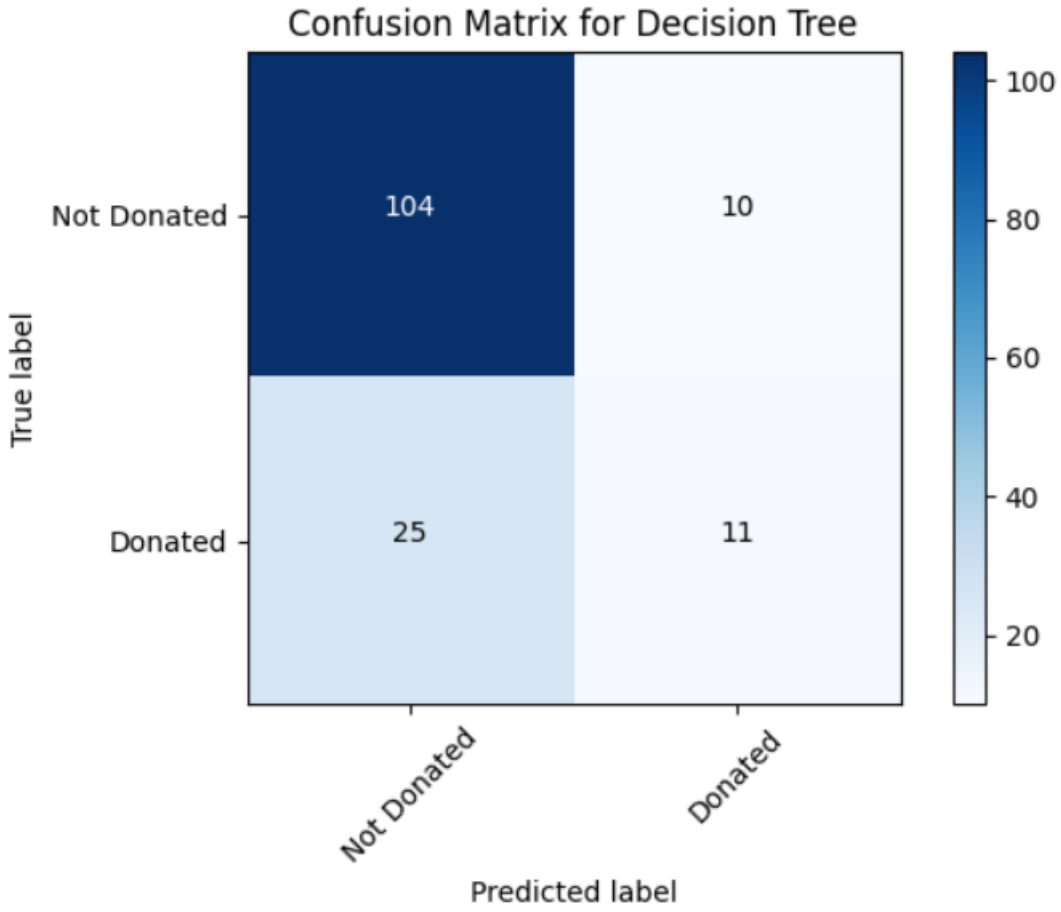
```
Accuracy: 0.7133333333333334

Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.83      0.82       114
           1       0.39      0.33      0.36        36

    accuracy                           0.71       150
   macro avg       0.59      0.58      0.59       150
weighted avg       0.70      0.71      0.71       150
```

   After finding the best hyperparameters: (runtime=1s)

```
Best Hyperparameters:
{'max_depth': None, 'min_samples_leaf': 4, 'min_samples_split': 2}

Best Estimator:
DecisionTreeClassifier(min_samples_leaf=4, random_state=1)

Accuracy with Best Estimator: 0.7666666666666667

Classification Report with Best Estimator:
              precision    recall  f1-score   support

           0       0.81      0.91      0.86       114
           1       0.52      0.31      0.39        36

    accuracy                           0.77       150
   macro avg       0.67      0.61      0.62       150
weighted avg       0.74      0.77      0.74       150
```

Accuracy was improved to (0.77)

Confusion Matrix:



2. Naïve Bias:

   Before doing Cross Validation:

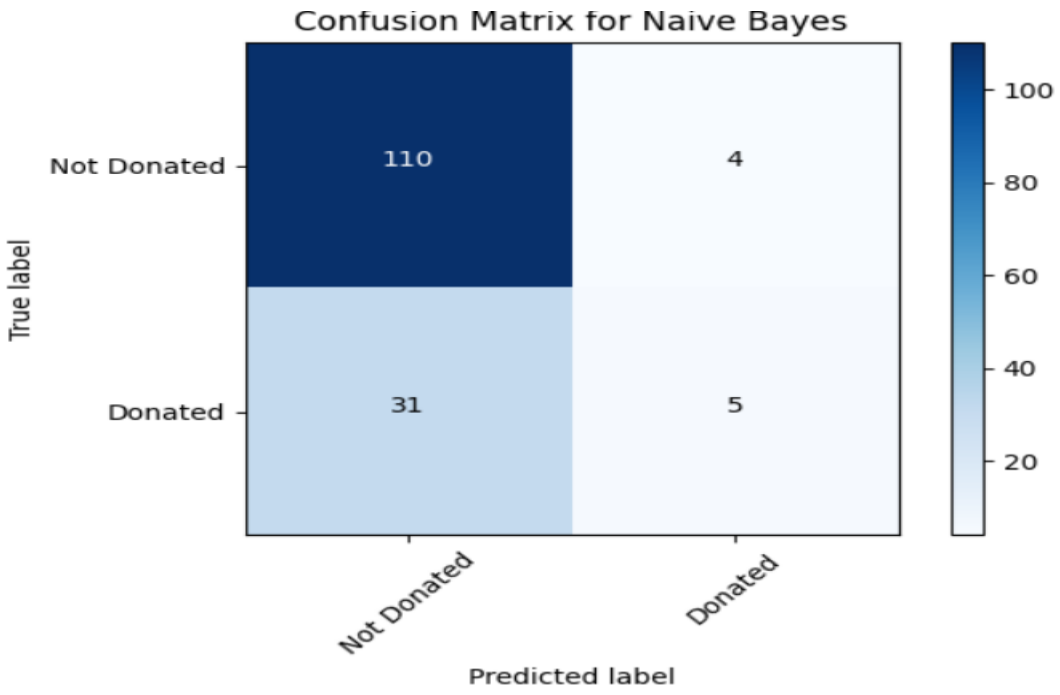
```
Accuracy: 0.76

Classification Report:
              precision    recall  f1-score   support

           0       0.79      0.94      0.86       114
           1       0.50      0.19      0.28        36

    accuracy                           0.76       150
   macro avg       0.64      0.57      0.57       150
weighted avg       0.72      0.76      0.72       150
```

After: (runtime=4s)

```
Accuracy: 0.7666666666666667

Accuracy with Best Estimator: 0.7666666666666667

Classification Report with Best Estimator:
              precision    recall  f1-score   support

           0       0.81      0.91      0.86       114
           1       0.52      0.31      0.39        36

    accuracy                           0.77       150
   macro avg       0.67      0.61      0.62       150
weighted avg       0.74      0.77      0.74       150
```

Accuracy was improved to (0.77)

Confusion Matrix:



3. Support Vector Classification:
   Before doing Cross Validation:

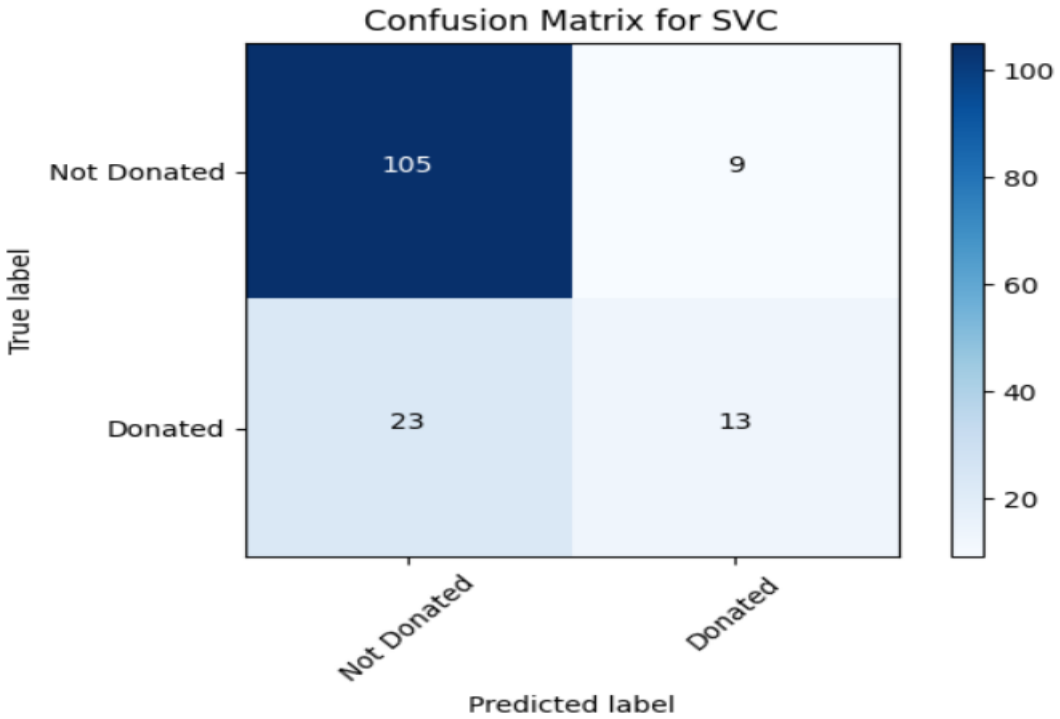
```
Accuracy: 0.78

Classification Report:
              precision    recall  f1-score   support

           0       0.78      0.98      0.87       114
           1       0.71      0.14      0.23        36

    accuracy                           0.78       150
   macro avg       0.75      0.56      0.55       150
weighted avg       0.77      0.78      0.72       150
```

   After: (runtime=2s)

```
Accuracy: 0.7866666666666666

Classification Report:
              precision    recall  f1-score   support

           0       0.82      0.92      0.87       114
           1       0.59      0.36      0.45        36

    accuracy                           0.79       150
   macro avg       0.71      0.64      0.66       150
weighted avg       0.77      0.79      0.77       150
```

   Accuracy was improved to (0.79)

   Confusion Matrix:



4. Artificial Neural Network:

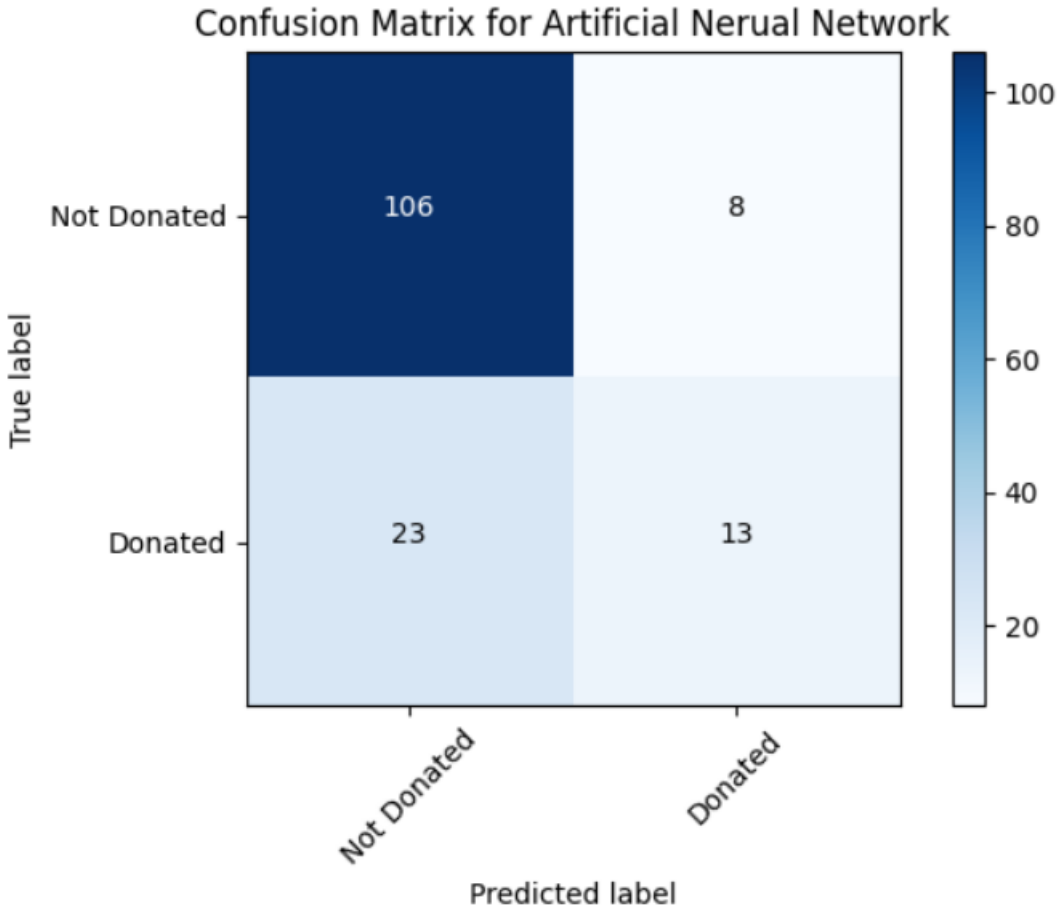
Before doing Cross validation:

```
Accuracy: 0.7866666666666666
Classification Report:
              precision    recall  f1-score   support

           0       0.79      0.97      0.87       114
           1       0.70      0.19      0.30        36

    accuracy                           0.79       150
   macro avg       0.75      0.58      0.59       150
weighted avg       0.77      0.79      0.74       150
```

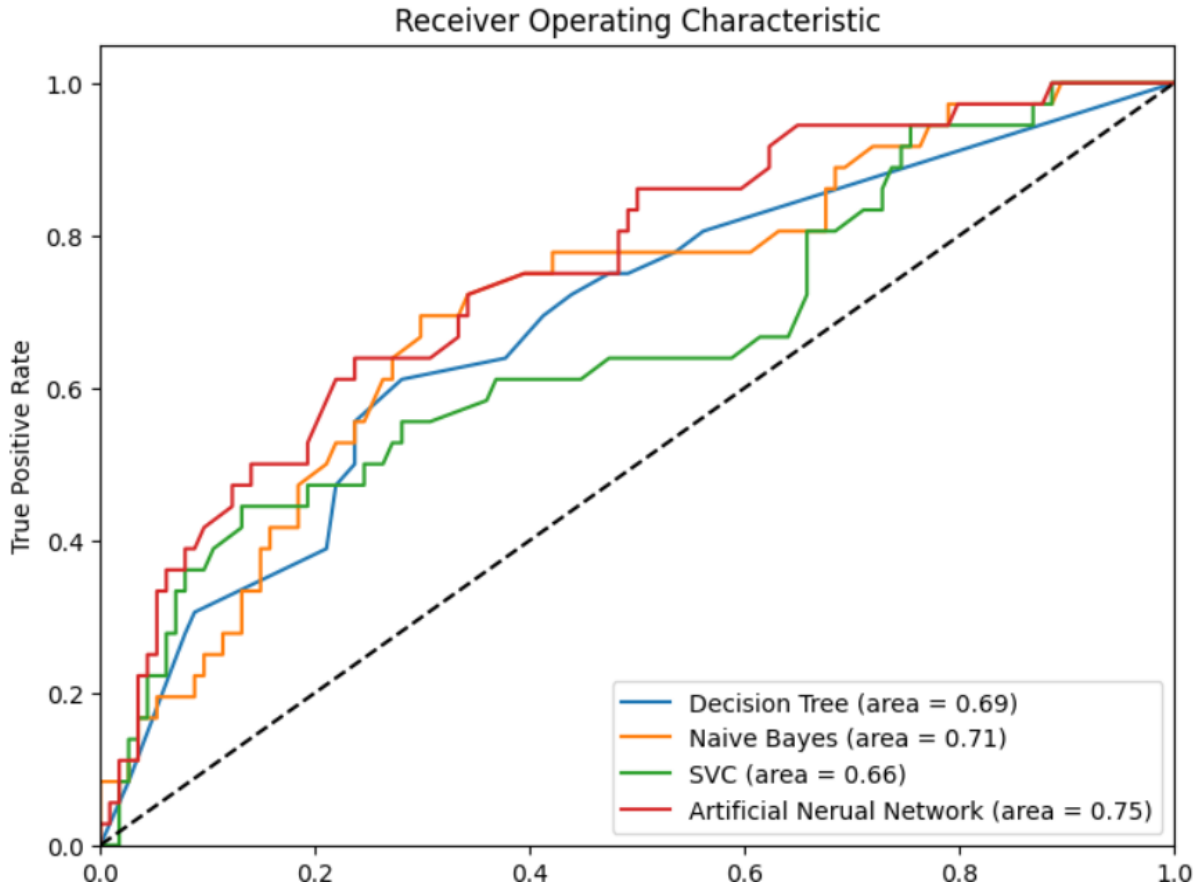
After doing the cross validation: (runtime=1m):

```
Classification Report:
              precision    recall  f1-score   support

           0       0.82      0.93      0.87       114
           1       0.62      0.36      0.46        36

    accuracy                           0.79       150
   macro avg       0.72      0.65      0.66       150
weighted avg       0.77      0.79      0.77       150
```

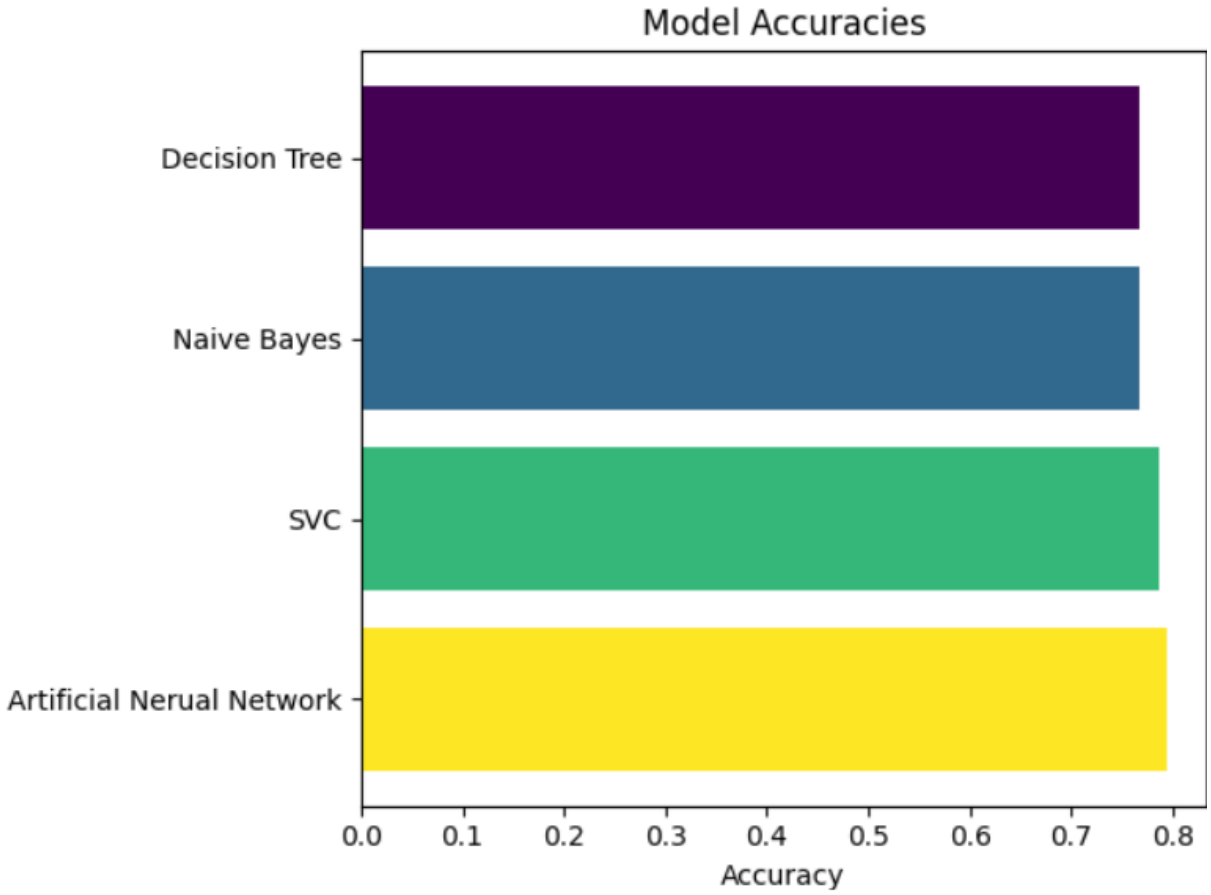
Confusion Matrix:



ROC curve:



Model Accuracies:



The project showcased the utility of different machine learning algorithms in solving classification problems. It highlighted the importance of data preprocessing, feature scaling, model selection, hyperparameter tuning, and evaluation metrics in building effective predictive models.

This project demonstrated the practical application of machine learning techniques in solving real-world problems, specifically in the healthcare domain.