# Assignment 3
# Muhammad Bilal Naseer
# 202046892

## Logistic Regression:

Logistic regression is a classic statistical method used for binary classification tasks. Logistic regression models the probability that an instance belongs to a particular class. It's a common method due to its simplicity, interpretability, and efficiency in handling large datasets. Logistic regression assumes a linear relationship between the independent variables and the log-odds of the dependent variable, making it ideal for scenarios where the decision boundary is linear or can be approximated as such. It serves as a robust starting point for classification tasks, it establishes a standard for evaluating the effectiveness of more advanced models.

## Support Vector Machine (SVM):

Support Vector Machine (SVM) is a powerful supervised learning algorithm used for both classification and regression tasks. It works by finding the hyperplane that best separates the classes in the feature space while maximizing the margin between them. SVM is chosen for its ability to handle both linear and nonlinear relationships through the use of kernel functions, making it versatile for various types of datasets. It is particularly effective in high-dimensional spaces and when the number of features exceeds the number of samples. SVM's ability to find the optimal decision boundary makes it robust against overfitting and suitable for scenarios where the data may not be linearly separable.

## Random Forest Classifier:

Random Forest Classifier is an ensemble learning method that constructs a multitude of decision trees during training and outputs the mode of the classes (classification) or the mean prediction (regression) of the individual trees. It's chosen for its robustness to overfitting, ease of use, and ability to handle both numerical and categorical data. Random Forest builds diverse trees by randomly selecting subsets of features and bootstrap samples of the data, which helps capture complex relationships and reduce variance. It's particularly effective when dealing with noisy or missing data, and it can handle large datasets with high dimensionality. Random Forest Classifier is known for its high predictive accuracy and resistance to overfitting, making it a popular choice for various classification tasks.

## Reason for Choosing Accuracy

Accuracy measures the proportion of true results (both true positives and true negatives) among the total number of cases examined. It's a straightforward and intuitive metric, making it a popular first choice for classification problems. Using accuracy as a scoring parameter is particularly useful when you are interested in the overall success of the model in assigning the correct labels, without distinguishing between the types of errors made. It provides a quick and easy way to gauge the model's performance, especially in scenarios where every incorrect classification is equally important.

However, it's important to consider that Accuracy can misleadingly suggest high performance on imbalanced datasets by simply predicting the majority class. Hence, the choice of accuracy reflects an initial assumption that the model's overall rate of correct predictions is the most relevant measure of its effectiveness.

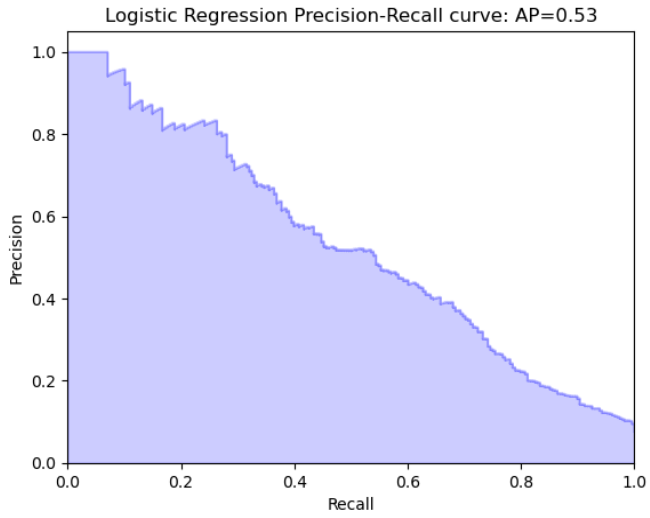
# Justification

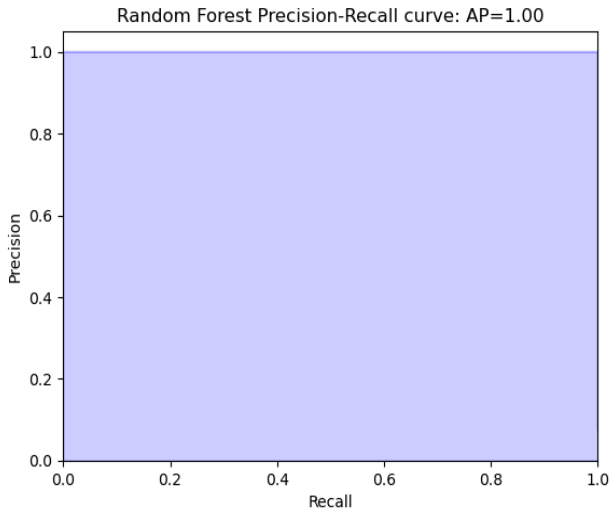
In machine learning the parameters we choose are very important for optimizing the performance. For both the models: Support Vector Machines and Random Forest classifier I used hyperparameter tuning in which a grid of potential parameters are passed and evaluated. This process involves systematically testing different combination of these parameters to determine which one yields the best results. This methodical approach to selecting parameters is based on the understanding that the performance of machine learning models is highly sensitive to the choice of these hyperparameters, and that optimal settings can significantly enhance model efficacy.

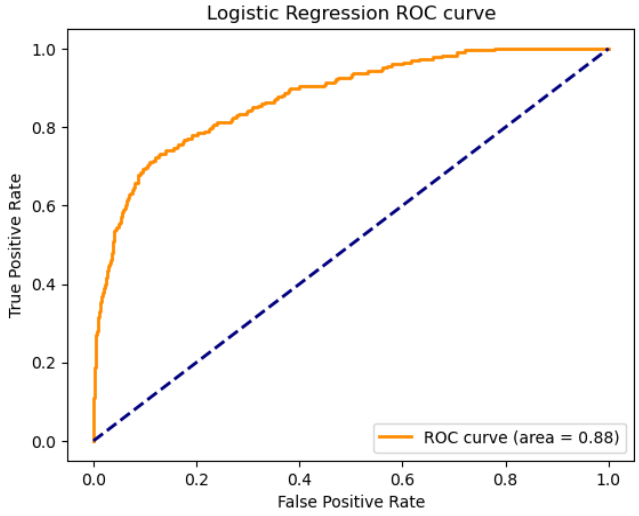
# Table for Best Results and Best Parameters

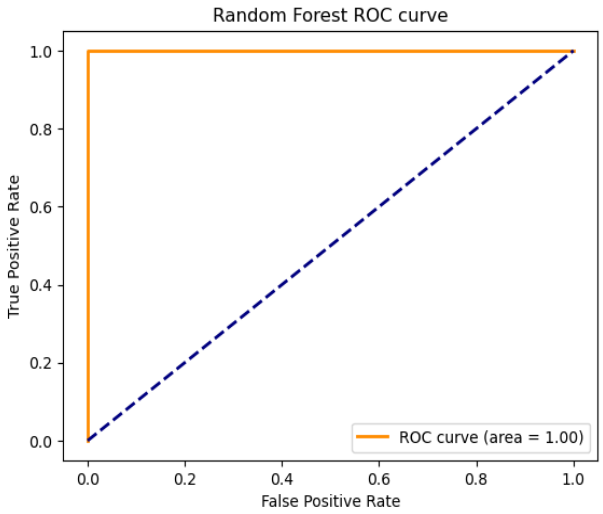
| Models | Best Result | Parameters |
|---|---|---|
| Logistic Regression | 0.908 | max_iter=1000, random_state=42 |
| SVM Model | 0.9246 | 'C': 10, 'gamma': 'scale', 'kernel': 'rbf' |
| Random Forest Model | 0.924 | 'max_depth': 10, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100 |

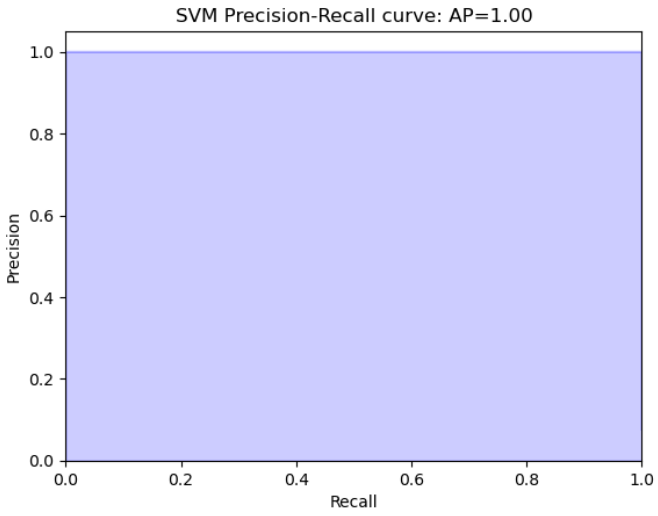
# Figures

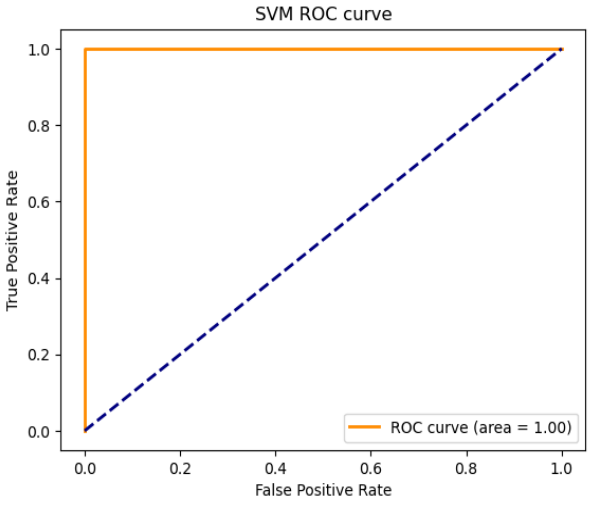

Logistic Regression Precision-Recall curve: AP=0.53
Precision
Recall


Random Forest Precision-Recall curve: AP=1.00
Precision
Recall


Logistic Regression ROC curve
True Positive Rate
False Positive Rate
ROC curve (area = 0.88)


Random Forest ROC curve
True Positive Rate
False Positive Rate
ROC curve (area = 1.00)


SVM Precision-Recall curve: AP=1.00
Precision
Recall


SVM ROC curve
True Positive Rate
False Positive Rate
ROC curve (area = 1.00)

```
Best Model Overall: Random Forest
```

```
RandomForestClassifier(max_depth=10, random_state=42)
```

# Attributions

In this assignment I utilized the following online resources:
• https://scikit-learn.org/stable/tutorial/statistical_inference/model_selection.html
• https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc_crossval.html
• https://towardsdatascience.com/complete-guide-to-pythons-cross-validation-with-examplesa9676b5cac12
• https://scikit-learn.org/stable/modules/cross_validation.html#cross-validation-iterators
• https://machinelearningmastery.com/k-fold-cross-validation/

I also used chatgpt for better understanding some concepts and making my code more efficient.

For this assignment I also discussed my approach and what how I reached the final solutions with my fellow course mates:

1. Syed Basim Ali
2. Sara Hamid
3. Muneeb-ur-Rehman