CSE422 Project Report

Project Topic: Financial Institution Subscriber Prediction

Section: 23

Group Number: 3

Members:

1. **Name:** Mohd. Shadman Ahmed Razeen
2. **Name:** Muktadirul Alam Sowad

Table of Contents

1.Introduction:

This project aims to predict whether a certain individual would sign up for a term deposit based on personal and banking information. Here, the Bank Marketing Dataset has been to predict the outcomes. Banks and different financial establishments organize various campaigns to advertise their term deposits. These initiatives are often expensive and time consuming. The main challenge is to determine ahead of time which customers are likely to sign up for a term deposit which will enable the bank to target its marketing strategy more effectively. To solve this problem, the project has used machine learning models which are trained on historical data campaigns data including the clients profile and economic indicators. The aim is to compare the predictive performance and select the most effective model.

The motivation behind this project is to accurately predict customer responses which will reduce the overall marketing costs of the financial institutions.

2.Dataset Description:

Dataset Description:

Dataset: Bank Marketing.csv

The bank marketing dataset contains 20 features excluding the target variable ‘y’. This is a binary classification problem as the target variable ‘y’ consists of two categorical instances ‘yes’ or ‘no’. Again, the dataset contains a total of 41,188 Data Points.

The dataset features have both categorical and numerical types, which reflects client information, previous campaign interactions, and macroeconomic indicators.

Input Features:

1. age *(*numerical*)*:
The age of the client in years. Older clients may have different financial actions than younger ones.

2. job *(*categorical*)*:
Type of job (e.g., admin., technician, blue-collar, retired). Indicates economic status.

3. marital *(*categorical*)*:
Marital status of the client (e.g., married, single, divorced). May influence financial priorities.

4. education (categorical):
Education level (e.g., primary, secondary, tertiary, unknown). Affects income and financial decisions.

5. default (categorical):
Whether the client has credit in default (yes/no/unknown). Indicates creditworthiness.

6. housing (categorical):

If the client has a housing loan. Useful in understanding existing liabilities.

7. loan (categorical):
Tells if the client has a personal loan. Like housing, shows financial burden.

8. contact (categorical):
Type of communication used (cellular/telephone). Some contact methods may yield better responses.

9. month (categorical):
Month of the last contact during the campaign. May show seasonal effects on success rates.

10. day_of_week (categorical):
Day of the week of the last contact. Some days may result in better outcomes.

11. duration (numerical):
Duration of the last contact in seconds. Longer conversations typically lead to more successful outcomes.

12. campaign (numerical):
Number of contacts performed during this campaign for the client. Can reflect persistence or annoyance.

13. pdays (numerical):
Number of days since the client was last contacted in a previous campaign. A value of 999 means never contacted.

14. previous (numerical):
Number of contacts performed before this campaign. Higher values may suggest interest or over-contacting.

15. poutcome *(categorical)*:
Outcome of the previous marketing campaign (success, failure, nonexistent). Strong indicator of client responsiveness.

16. emp.var.rate *(numerical)*:
Employment variation rate — an economic indicator that may reflect market confidence.

17. cons.price.idx *(numerical)*:
Consumer price index — a macroeconomic indicator tied to inflation.

18. cons.conf.idx *(numerical)*:
Consumer confidence index — measures the confidence of consumers in the economic outlook.

19. euribor3m *(numerical)*:
3-month Euribor rate — an important benchmark for interest rates in Europe.

20. nr.employed *(numerical)*:
Number of employees — another macroeconomic indicator that may reflect the health of the economy.

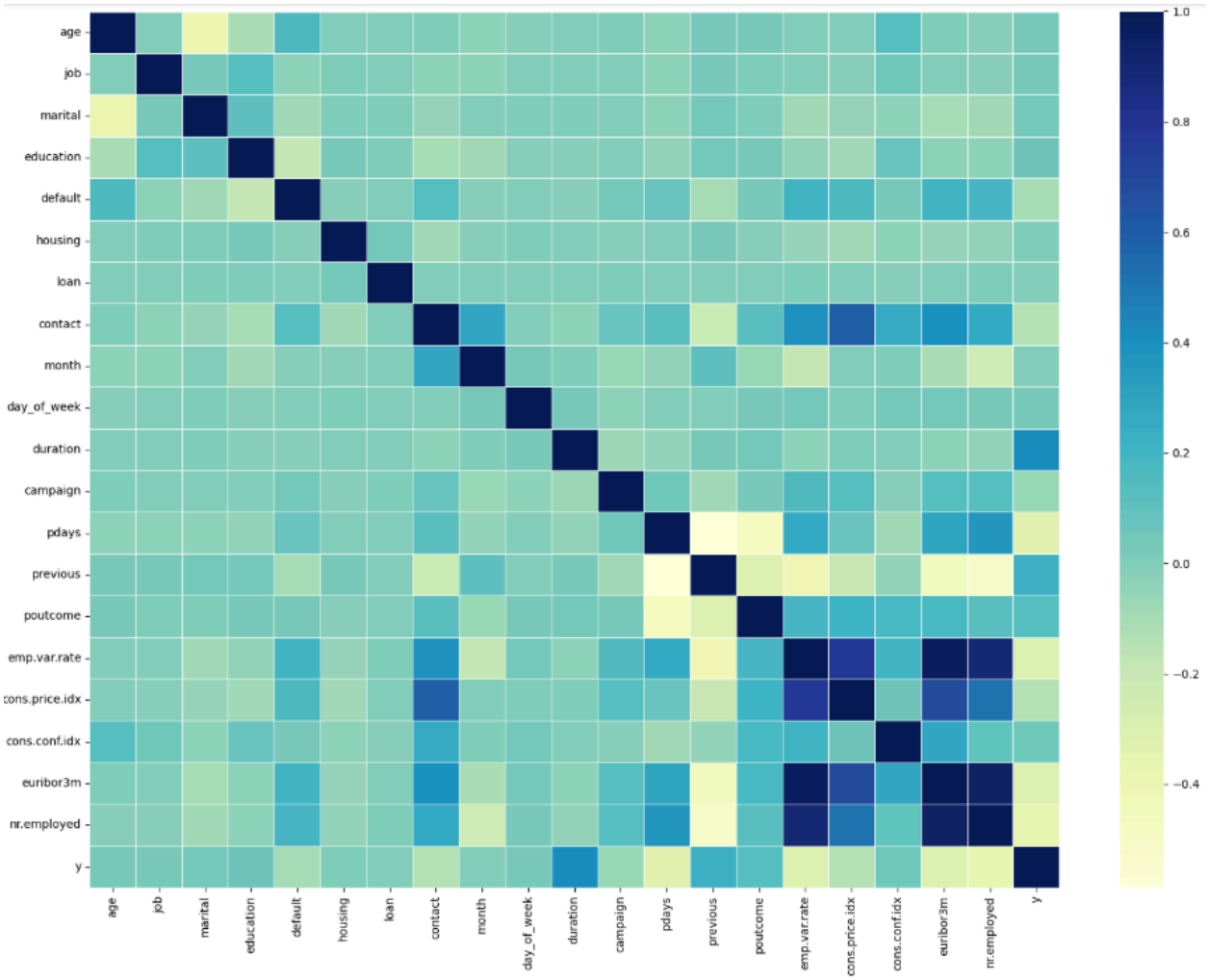
Target Feature:

21. y (binary: yes/no):

Indicates whether the client subscribed to a term deposit.

Correlation Analysis:

The heatmap of the correlated features are given below:-



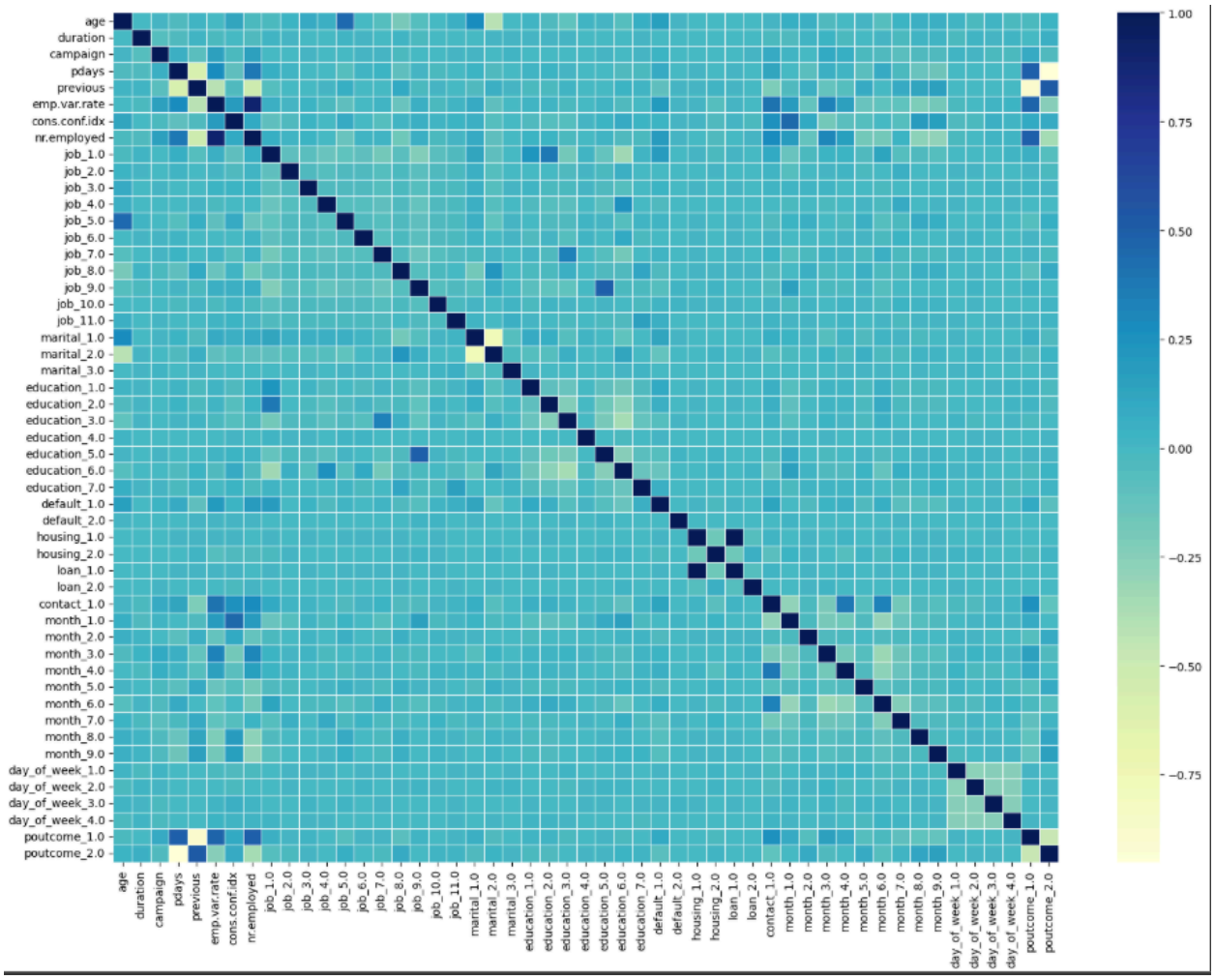
A correlation heatmap reveals relationships between features and the target variable. The correlated columns are:

- Correlation between cons.price.idx and emp.var.rate: 0.7753
- Correlation between emp.var.rate and euribor3m: 0.94515.

Here, the column where the correlation value is greater than 0.75 has been dropped.

Dropped Column: cons.price.idx, euribor3m

After One-Hot Encoding the generated Heatmap:



The correlated columns are:

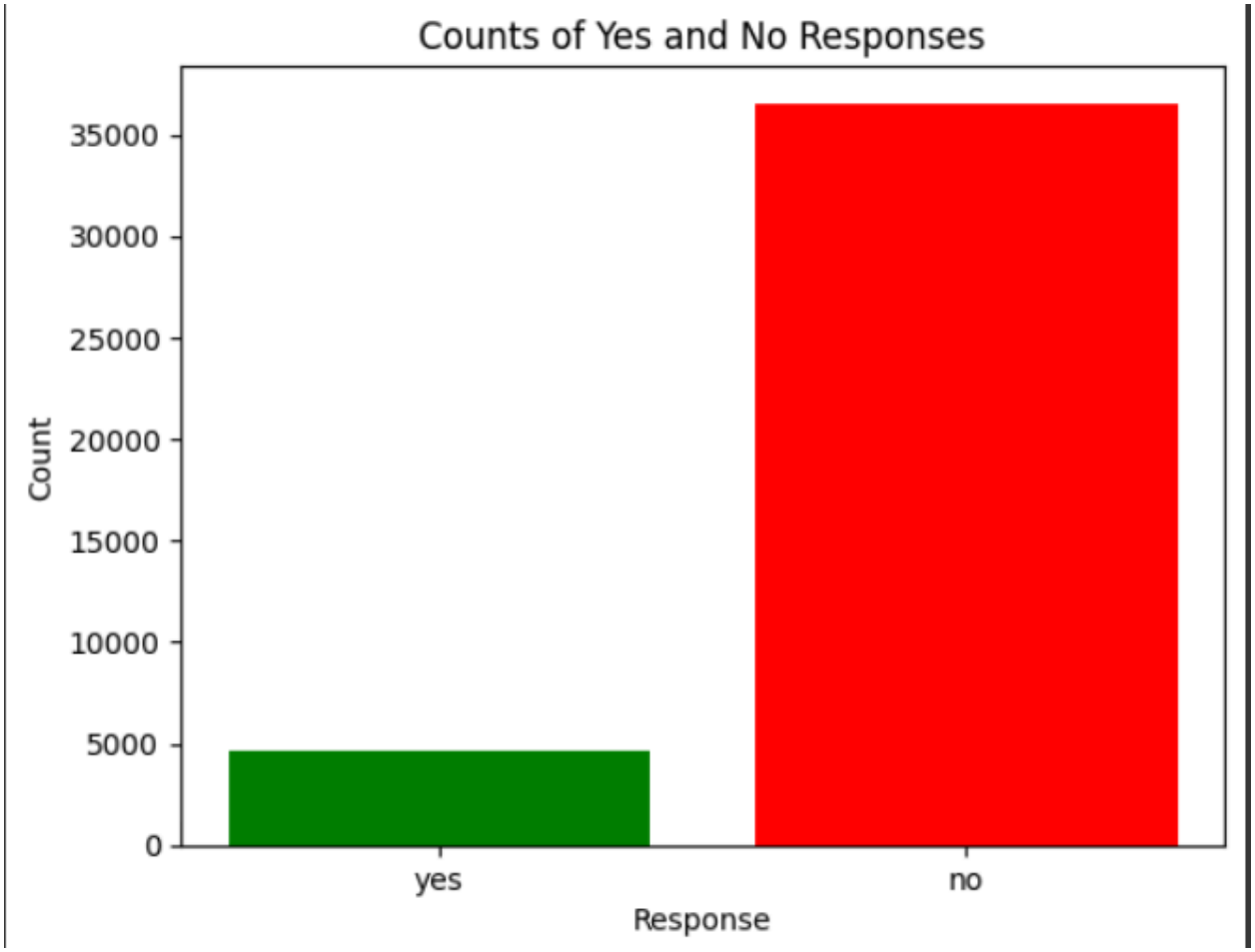
- Correlation between nr.employed and emp.var.rate: 0.906970

Here, the column where the correlation value is greater than 0.75 has been dropped.

Dropped Column: nr.employed

Imbalanced Dataset:

The target feature ‘y’ have two instances ‘yes’ and ‘no’. The number of ‘no’ count is 36548 and the number of ‘yes’ count is 4640, which clearly shows that the dataset is highly imbalanced.



3.Dataset Preprocessing:

Null/Missing Values:

In the entire Bank Marketing dataset, Null or missing values were absent. Still, the dataset was preprocessed for any sorts of null values. As the dataset was imbalanced, the rows had not been removed rather a value was imputed on required places. Because, removed rows would heavily affect the instances of ‘yes’ as it was only of 4640 rows.

Categorical Values:

There are 10 categorical columns on the entire dataset. The categorical values has been encoded to integer, as the model is unable to train using strings. One Hot Encoding has been used as an encoding method. Because, One-hot encoding prevents neural network model from assuming false category order and considers all categories are equally important in feature space.

Feature Scaling:

To reduce bias and ensure that all features contribute equally to a machine learning model model we perform feature scaling. To prevent dominance by large scale features the MinMax scaler puts all inputs within the same range of 0 and 1 and allows the optimizer to move in a more direct path. This type of situation is more suitable while training a neural network model.

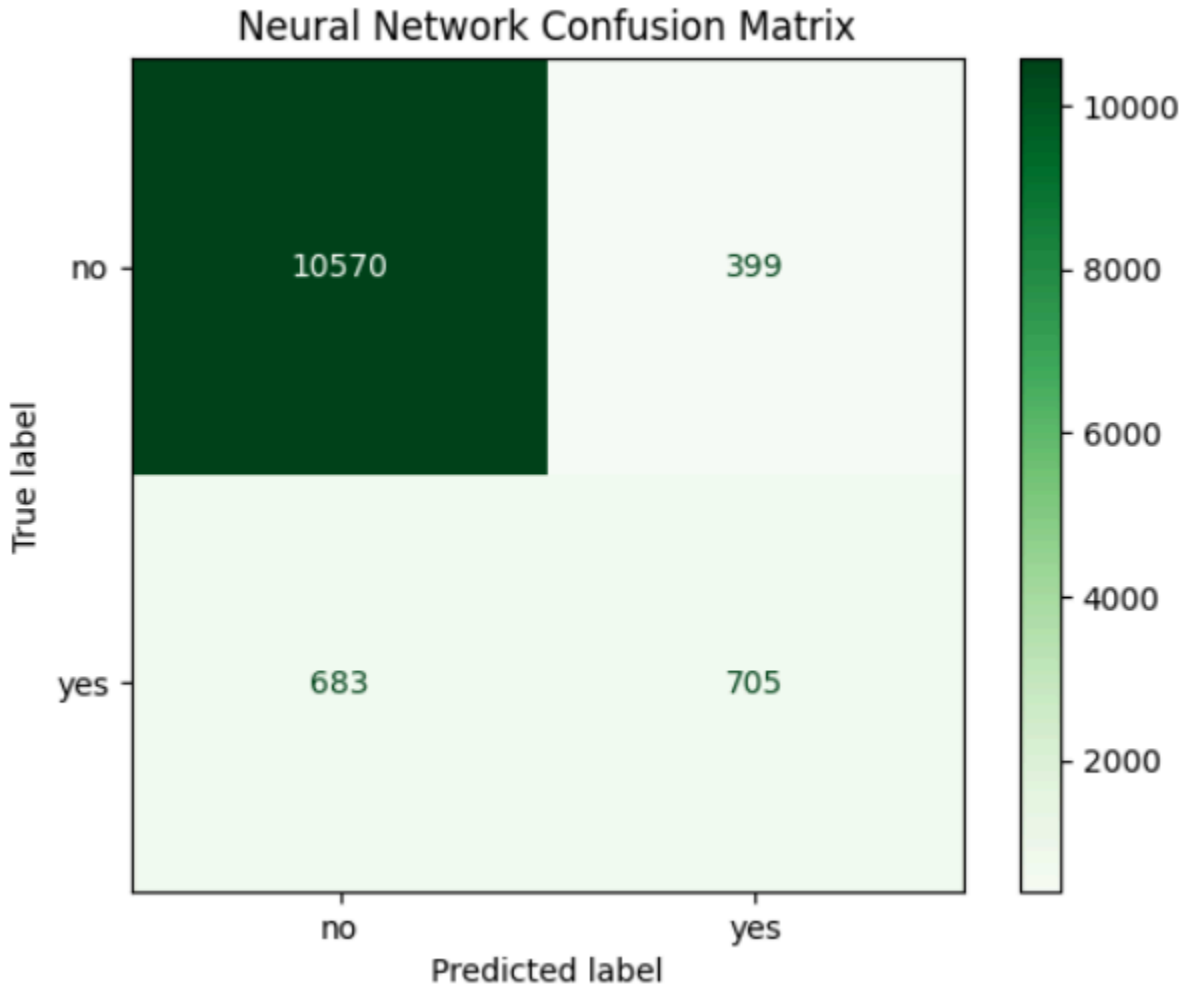
4.Dataset Splitting:

We split our dataset into a ratio of 70 to 30, where 70% of the data is the training set and 30% of the data is the test set. For the random_state=0 has been set for the dataset which ensures a stratify split. It means every time the same train/test split of the data is performed.

5.Model Training & Testing:

Neural Network:

The neural network is formed with multiple layers of neurons where weights are adjusted during each epoch. It can extract the complex nonlinear relationships between input features and targeted variables. It can learn from large and high dimensional datasets and tuned for better performance. Using sufficient amounts of data and tuning, neural networks can easily outperform traditional machine learning models. The learning rate was tuned to 0.0001, epoch =100 and batch size = 64, accuracy was found to be higher than other cases. The final accuracy score of this model was 0.912.

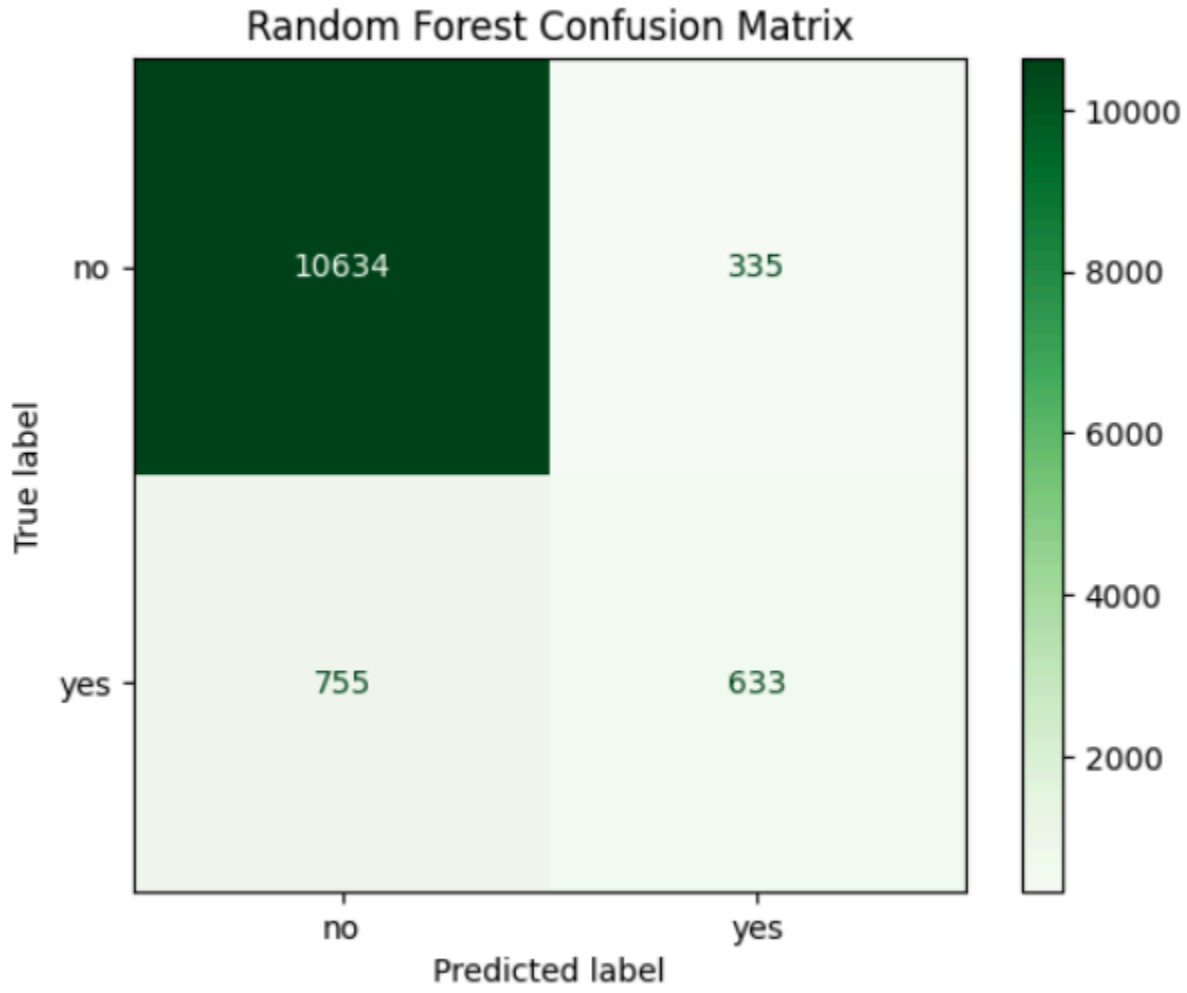


Neural Network:

	precision	recall	f1-score	support
no	0.94	0.96	0.95	10969
yes	0.64	0.51	0.57	1388
accuracy			0.91	12357
macro avg	0.79	0.74	0.76	12357
weighted avg	0.91	0.91	0.91	12357

Random Forest Classifier:

Random forest Classifier is an ensemble learning model which builds multiple decision trees to reduce overfitting and increase the overall accuracy. It is suitable for classification problems and can handle both linear and complex/non-linear relationships of the dataset. As the dataset contains both categorical and numerical data, random forest can handle both without much processing. The final accuracy of this model during testing was 0.91.



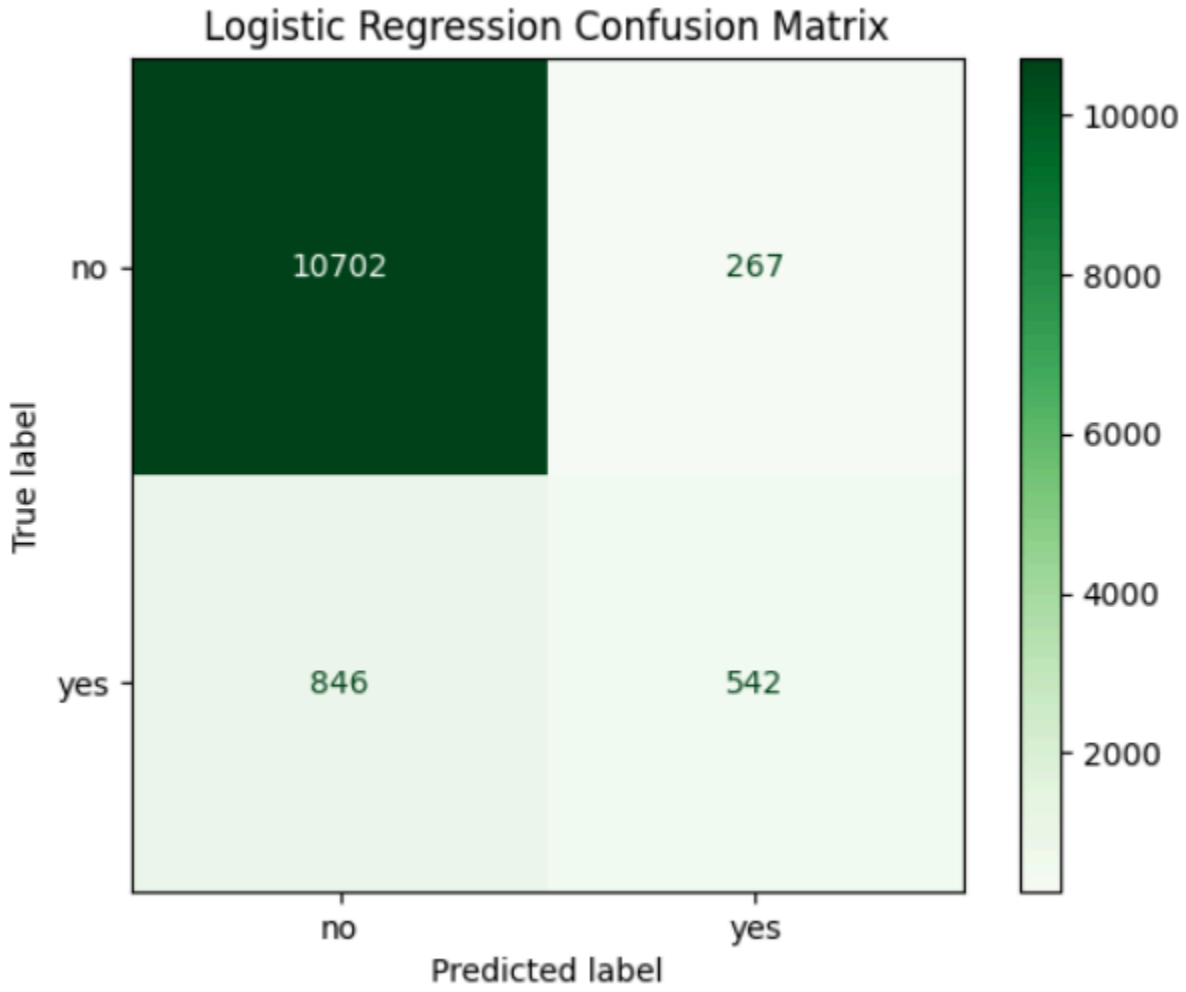
```
Random Forest:
              precision    recall  f1-score   support

          no       0.93      0.97      0.95     10969
         yes       0.65      0.46      0.54      1388

    accuracy                           0.91     12357
   macro avg       0.79      0.71      0.74     12357
weighted avg       0.90      0.91      0.90     12357
```

Logistic Regression:

Logistic regression is a linear model that estimates the probability of a binary outcome applying the sigmoid function. It is a strong and works well for binary classification and large datasets. The final output provides a values that ranges between 0 to 1. The final accuracy from this model was 0.909.



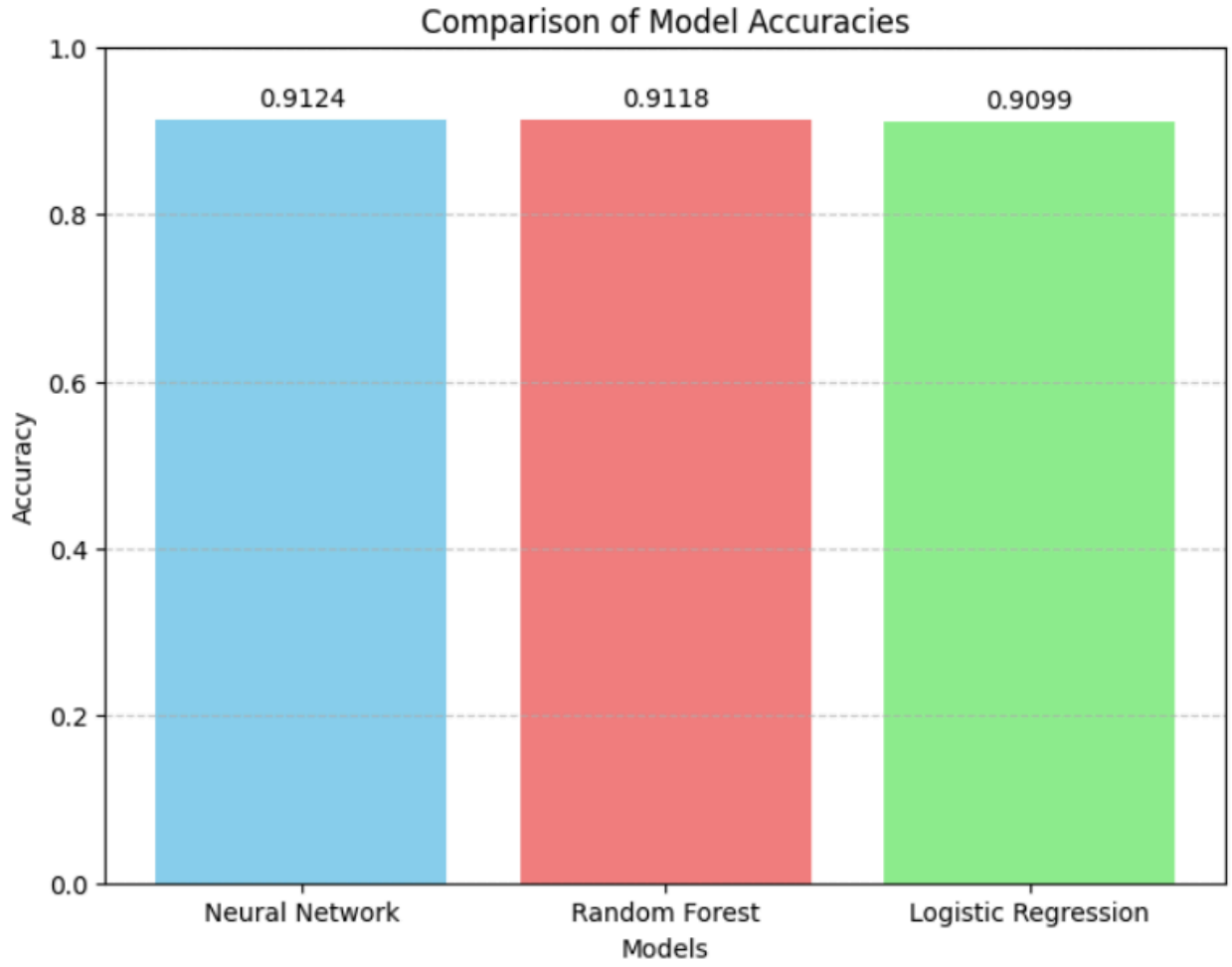
```
Logistic Regression:
              precision    recall  f1-score   support

          no       0.93      0.98      0.95     10969
         yes       0.67      0.39      0.49      1388

    accuracy                           0.91     12357
   macro avg       0.80      0.68      0.72     12357
weighted avg       0.90      0.91      0.90     12357
```

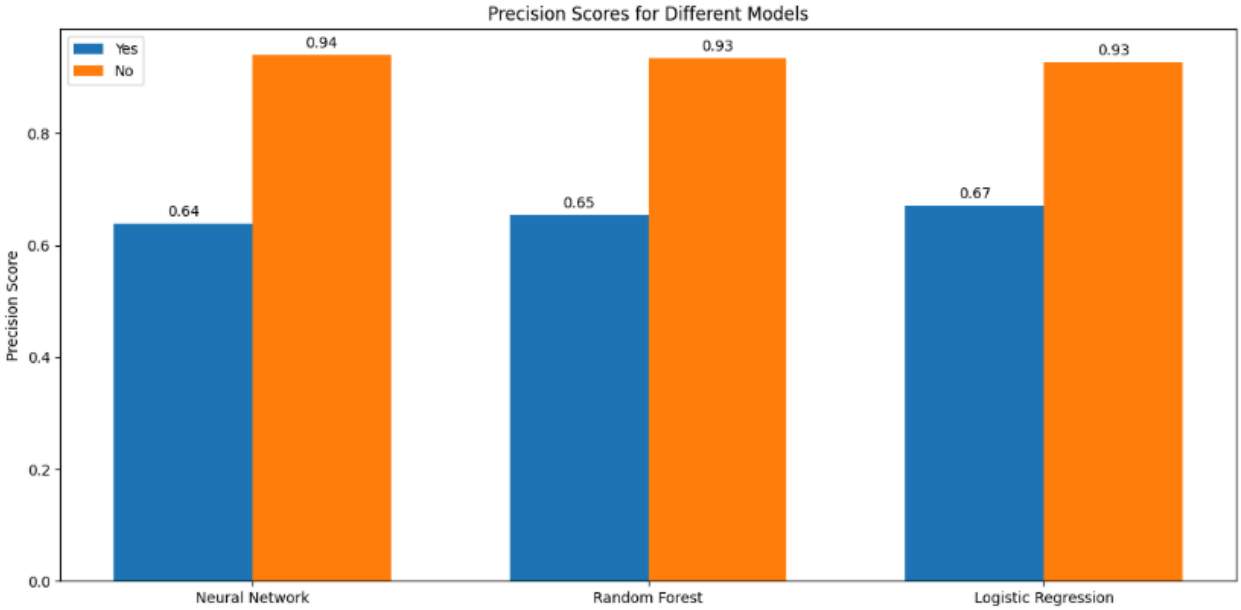
6.Model Comparison Analysis:

In this project we applied Neural network, Random forest Classifier and Logistic regression models to our dataset. The comparison bar chart of accuracy score are shown below:-

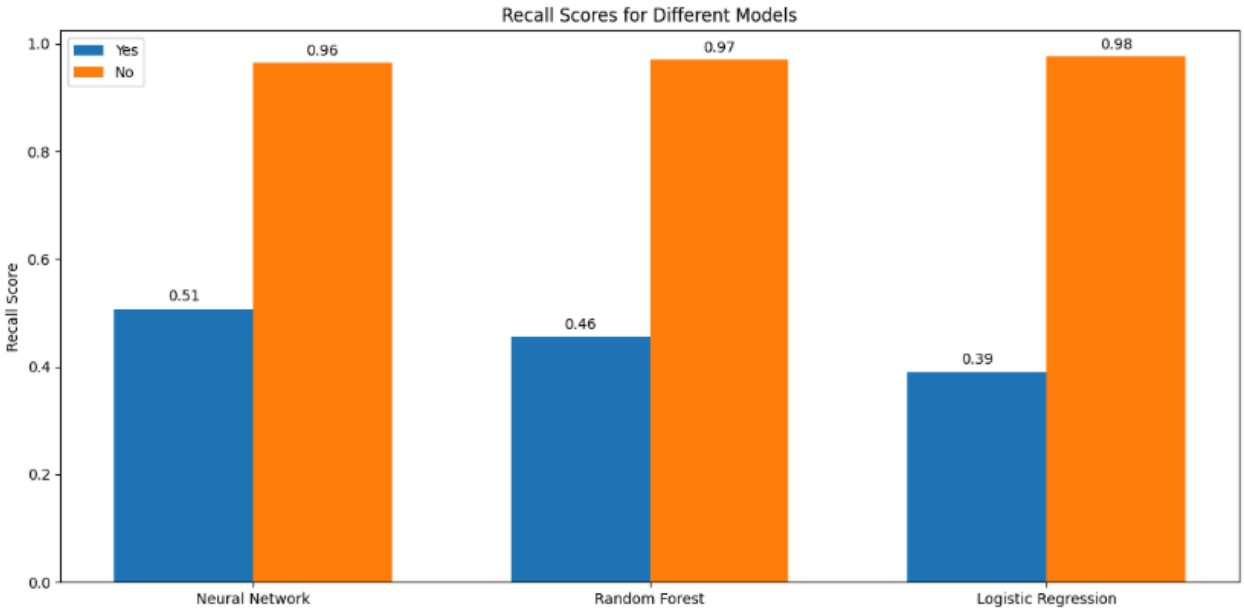


In our comparison of model performance on the Bank Marketing dataset, the Neural Network model yielded the highest accuracy among the three models. On the other hand, logistic regression had the least accuracy of 0.909 among all the models. Again, Random forest classifier provided a middle-ground performance with an accuracy of 0.9118. This comparison indicates that the Neural Network model outperformed the other models which came out to be a better candidate for selection.

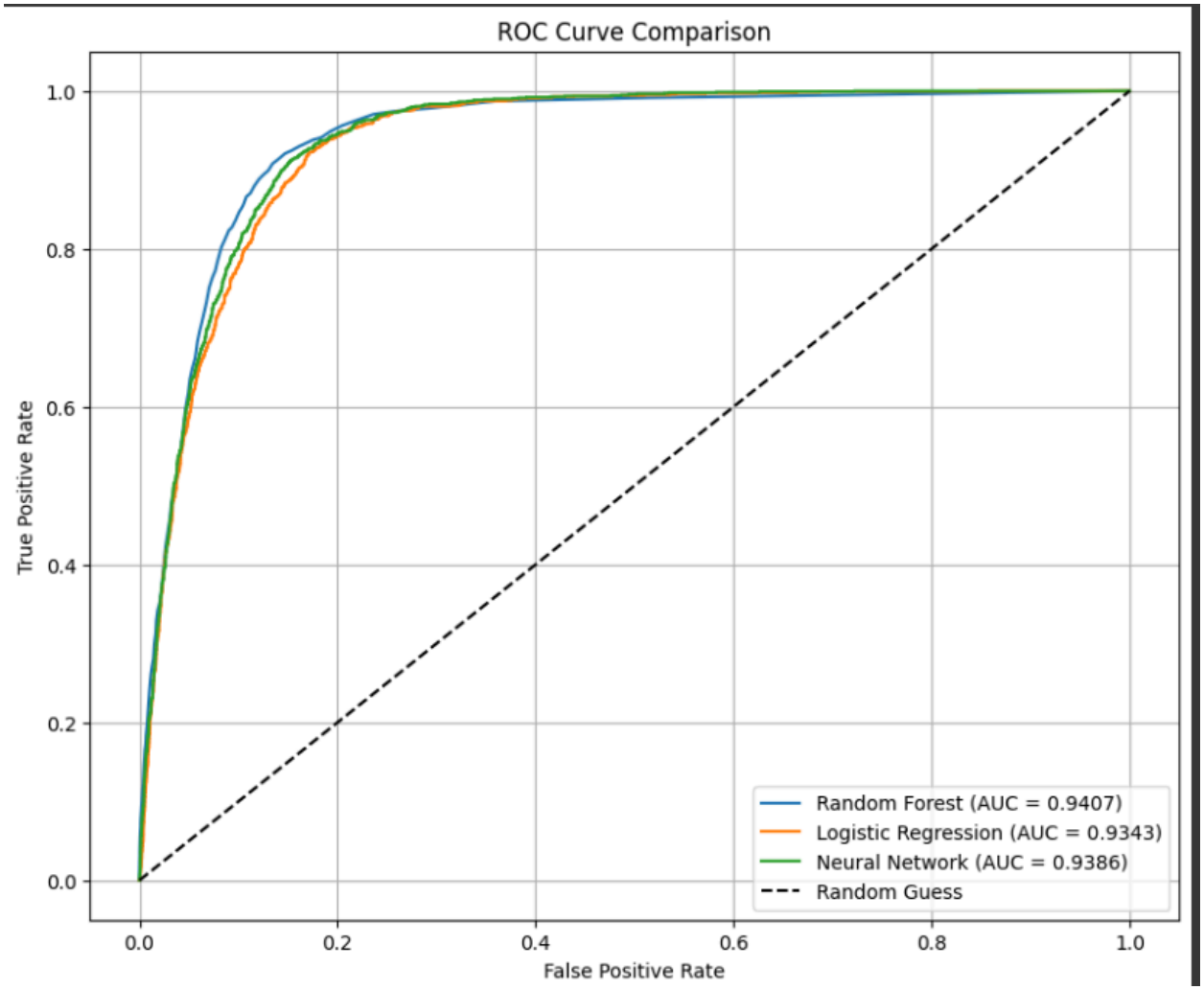
The comparison bar chart of precision score are shown below:-



The comparison bar chart of recall score are shown below:-



The AUC Score and ROC curve of the three model are shown below:-



7.Conclusion:

The three models Neural Network, Random forest classifier, Logistic regression have achieved overall high accuracy on the test set. However, looking deeper into the precision and recall score we can observe some important differences. The ‘yes’ class, precision and recall score are significantly lower than the ‘no’ class. It was observed before that the dataset is highly imbalanced which causes the models to favor predicting ‘no’ instances, which leads to better performance with high precision and recall and significantly low performance for ‘yes’ instances.

Neural network achieved the highest accuracy among the three models and slightly better for the precision and recall for the ‘yes’ instances. On the other hand, Logistic regression had the least overall accuracy and the lowest recall for the ‘yes’ instances. Also, the accuracy of random forest is close to the neural network and slightly better in precision for ‘yes’ instances. Neural network is able to extract complex and nonlinear relationships between the features of the dataset which helps the model to perform slightly better than other models. Again, Neural Network provides the flexibility to be tuned for higher accuracy and can learn from large datasets. On the contrary, Logistic regression is a linear model and may not extract complex patterns between the features of the dataset which leads to the least performance.

The core challenges we have faced on this project was on the data preprocessing stage. The severe imbalance between the two instances ‘yes’ and ‘no’ classes made some decisions of the preprocessing a bit challenging. Also, it made the model to predict the ‘yes’ instances difficult. While generating the heatmap we faced some difficulties to accurately select the features which were strongly correlated. On the other hand, we had to go through a number of trial and error while tuning the neural network model to successfully arrive at a higher accuracy.

To conclude, the three models performed well in overall accuracy but Neural network was slightly better at identifying potential subscribers. However, all models struggled to correctly identify the ‘yes’ class due to highly class imbalanced dataset. Improving future model performance would require addressing this issue with focused approaches.