Capstone Project

Predicting The Effectiveness of Bank Marketing Campaigns

1. Vilas Sonawane
2. Bhavika Gaurkar
3. Soumya Ranjan Dash

Point for Discussion

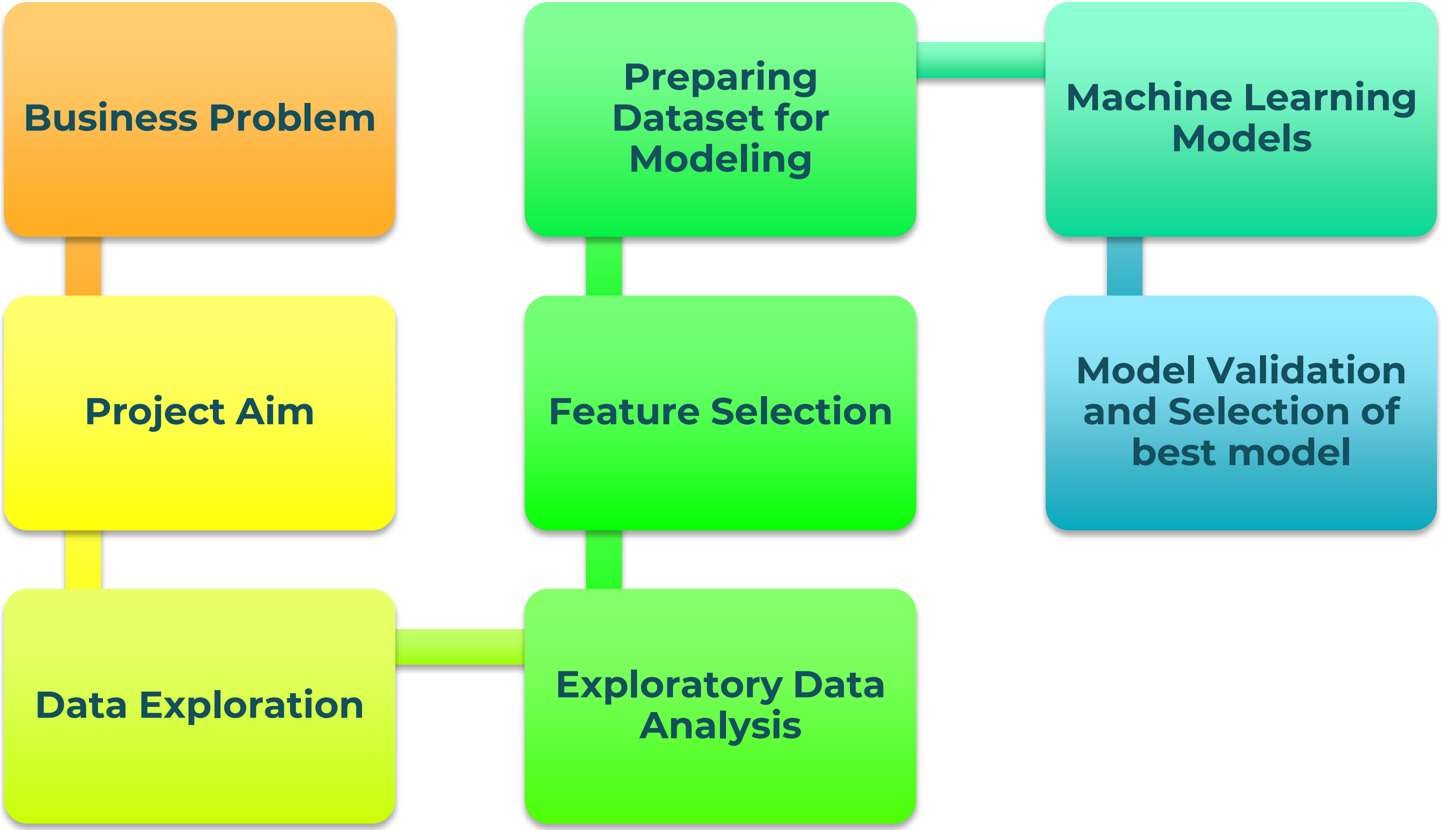
- About Portugal Banking Institution
- Project Road Map
- Business Problem
- Purpose of the Project
- Data Pipeline
- Data Summary
- Data Cleaning
- Feature Engineering
- Exploratory Data Analysis
  - i) Univariate Analysis
  - ii) Bivariate Analysis
- Feature Selection
- Preparing Dataset for Modelling
- Machine Learning Models
- Model Validation and Selection
- Conclusion

Portugal Banking Institution

Portugal has a modern banking system that includes one of the most advanced inter-bank networks in the world through Multibanco. There are currently over 150 banks in Portugal.

The majority of banks in Portugal belong to the Portuguese Banking Association. The central bank in Portugal is the Banco de Portugal, which also serves as the regulatory authority for Portuguese banks. It is a full member of the Euro system and the European System of Central Banks.

Project Road Map



Business Problems

The Portuguese banking institution want to predict the Effectiveness of their Marketing Campaign (Subscribe a Term Deposit) ?

Purpose of The Project

The main purpose of Our project is to build Machine Learning classification model which can predict the Effectiveness of Marketing Campaign to subscribe a term deposit of one of the Portuguese banking institution.

Data Pipeline

- **Data Processing:** In this part we have explore dataset and identified inconsistency in dataset if any and take necessary action on it wherever it was necessary. Since there were some columns which were not directly important so we have converted them into proper format , So we have also created some new features based on exited feature in dataset.

- **EDA:** In this part we explored the data and identified outlier and inconsistent data and removed outlier and modified data whenever required and obtained some useful insights and trends from the data.

- **Data Preparation:** After cleaning data we have prepared data for implementation of classification models by creation of some new features and removing features which are having high multicollinearity between each other. Then selected dependent features and normalized data and make it ready for application of machine learning model.

- **Machine Learning Classification Modelling**: After preparation of data we have applied different classification model on the dataset. Applying the model is not an easy task. It's also an iterative process. We have started with simple classification model, then slowly used complex models for better performance.

Data Summary

age	Age of the customers
Job	Types of jobs of customers
marital	Marital status
education	Type of education
housing loan	Customer has housing loan or not
loan	Personal loan or not
contact	Communication type cellular or telephone
month	Last contact month of year
day	Last contact day of a week
duration	Last contact duration
default	Has credit in default?
age	Number of contacts performed during this campaign and for this client
campaign	Number of contacts performed during this campaign and for this client
pdays	Number of days that passed by after the client was last contacted from a previous
Previous	Number of contacts performed before this campaign and for this client
poutcome	Outcome of the previous marketing campaign
y	has the client subscribed a term deposit? (binary: yes /no)

Data Cleaning

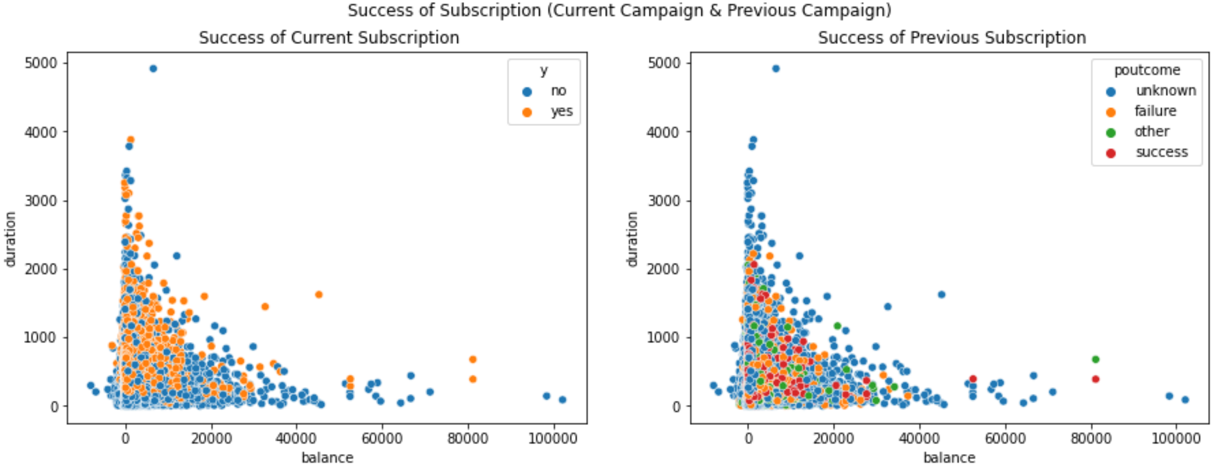
Dealing with Missing Values /Null /Nan & Outliers

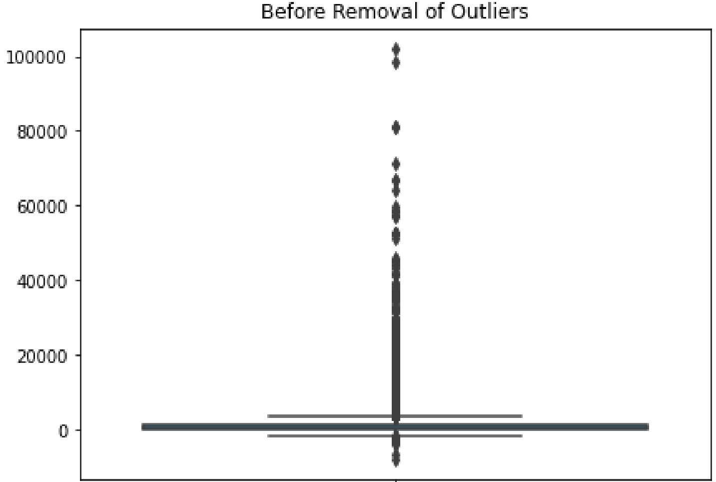
The data also has some outliers in following attributes Balance , Duration, Campaign, Previous column. Here in the below table, we can easily identify the presence of outliers (highlighted) Balance , Duration, Campaign, Previous column.

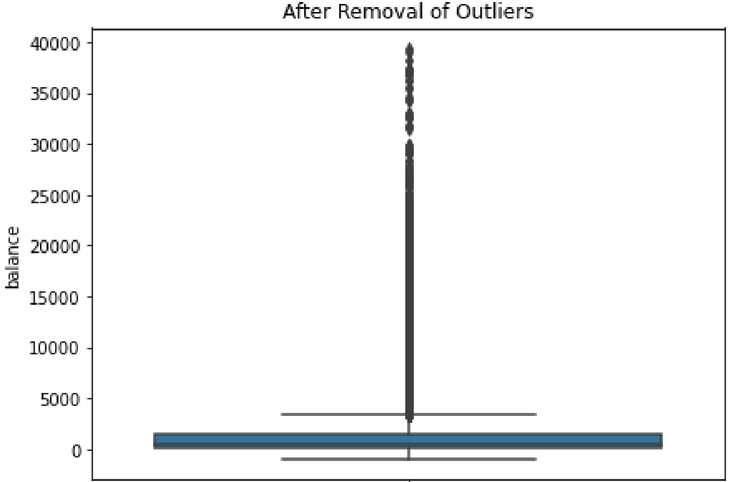
	age	balance	day	duration	campaign	pdays	previous	Outcome_y
count	45211.000000	45211.000000	45211.000000	45211.000000	45211.000000	45211.000000	45211.000000	45211.000000
mean	40.936210	1362.272058	15.806419	258.163080	2.763841	40.197828	0.580323	0.116985
std	10.618762	3044.765829	8.322476	257.527812	3.098021	100.128746	2.303441	0.321406
min	18.000000	-8019.000000	1.000000	0.000000	1.000000	-1.000000	0.000000	0.000000
25%	33.000000	72.000000	8.000000	103.000000	1.000000	-1.000000	0.000000	0.000000
50%	39.000000	448.000000	16.000000	180.000000	2.000000	-1.000000	0.000000	0.000000
75%	48.000000	1428.000000	21.000000	319.000000	3.000000	-1.000000	0.000000	0.000000
max	95.000000	102127.000000	31.000000	4918.000000	63.000000	871.000000	275.000000	1.000000

Dealing with Outliers

1. Balance

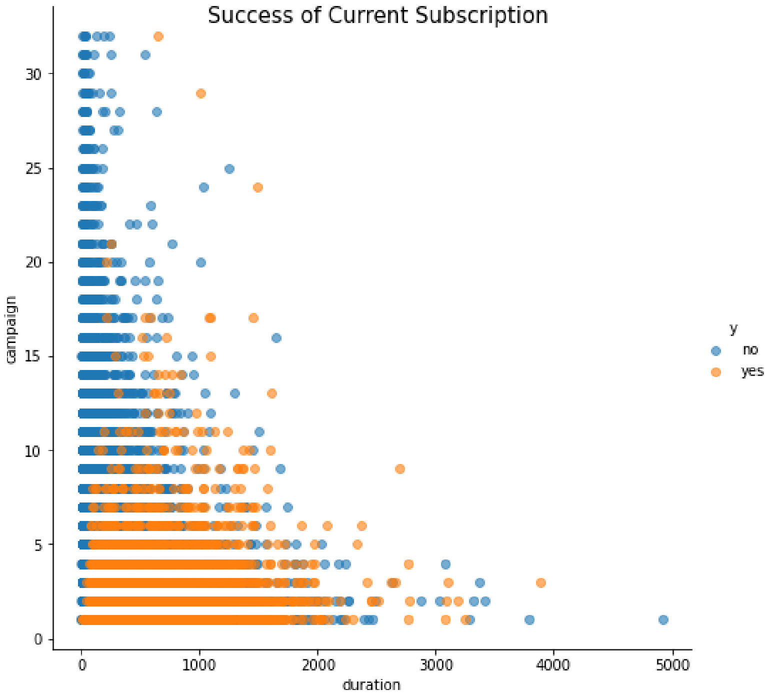


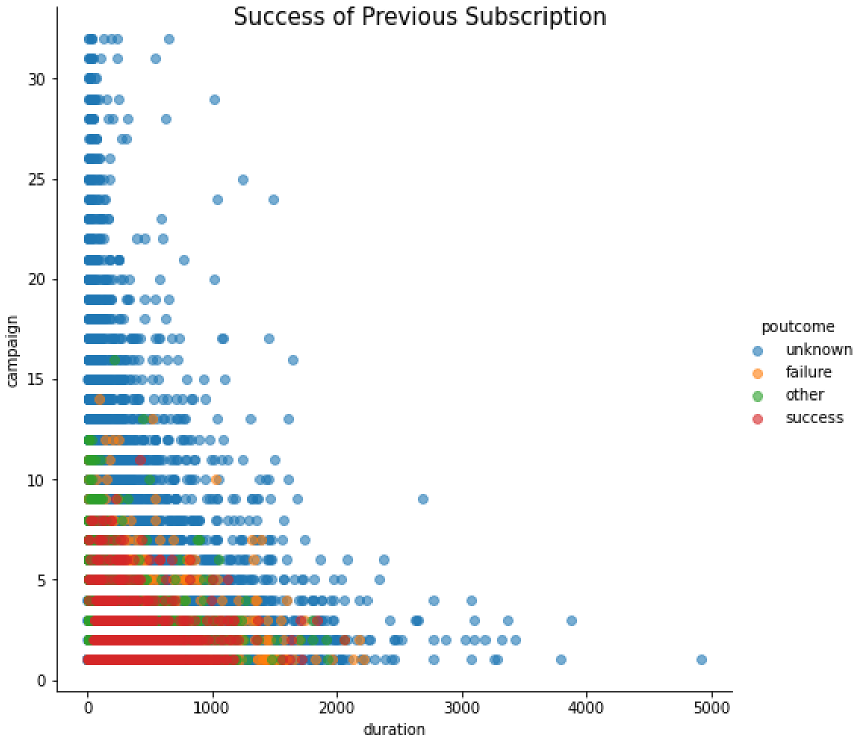




Dealing with Outliers

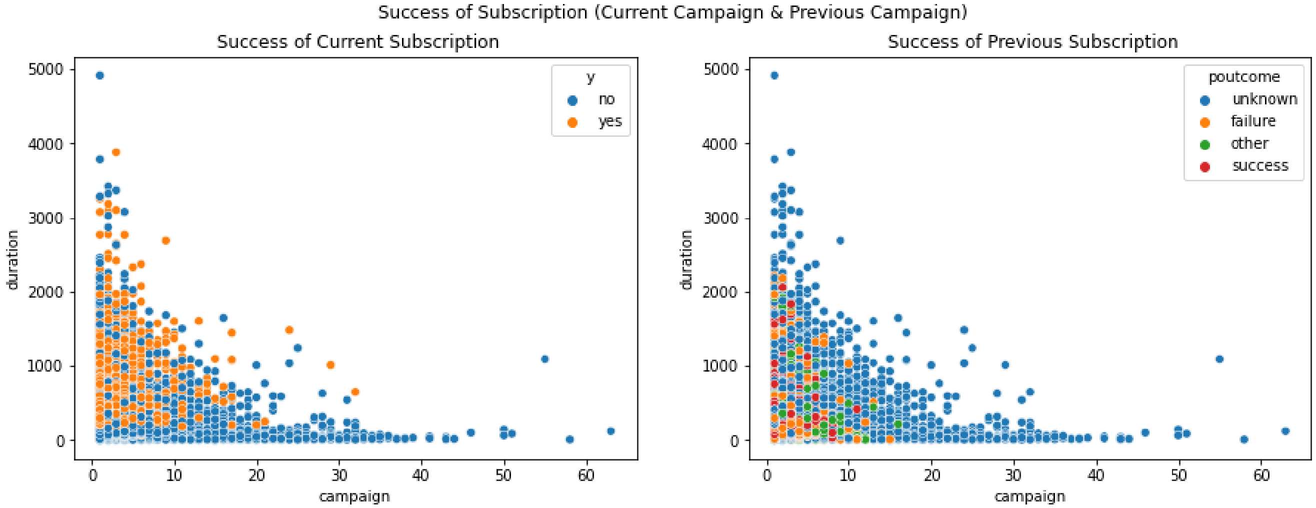
2 . Duration

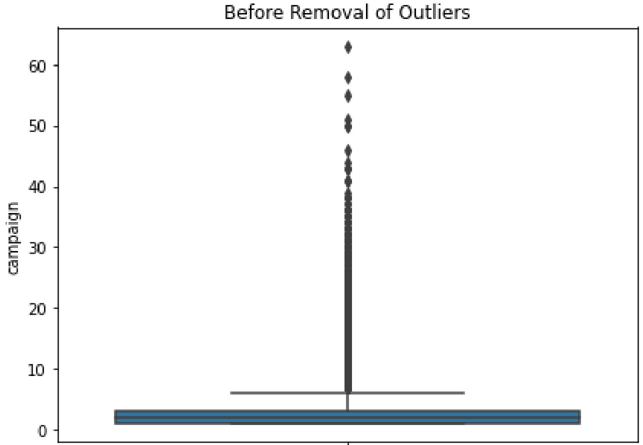


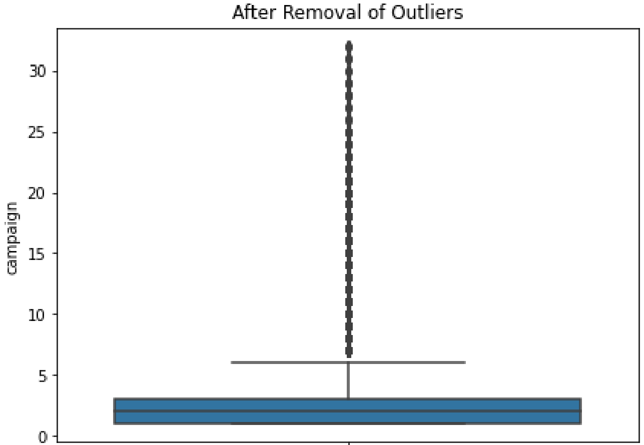


Dealing with Outliers

3. Campaign

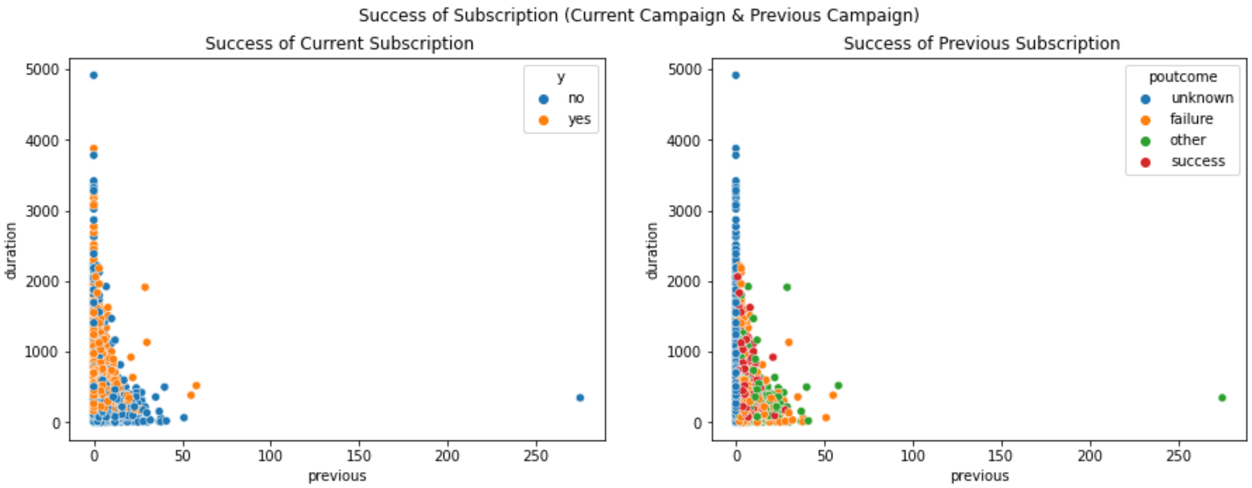


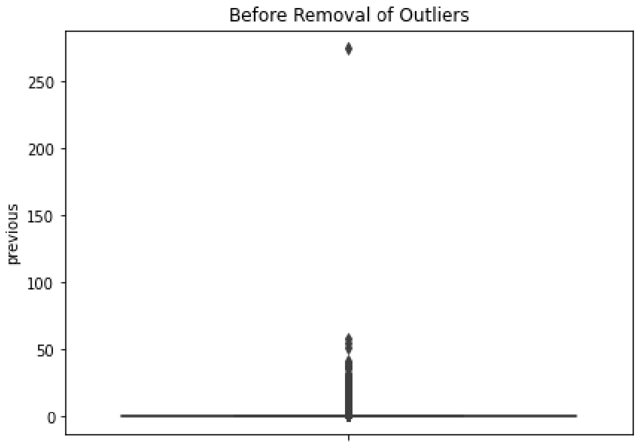


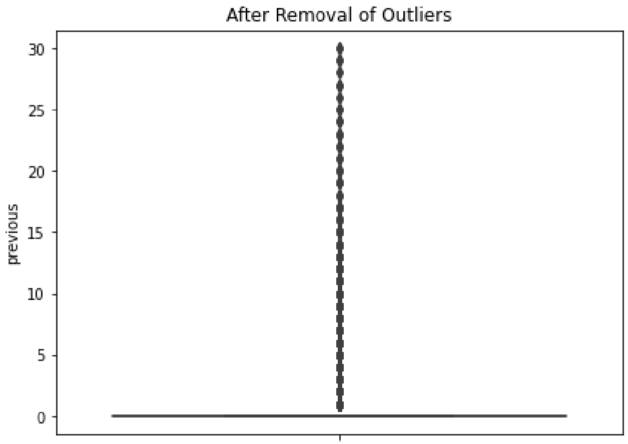


Dealing with Outliers

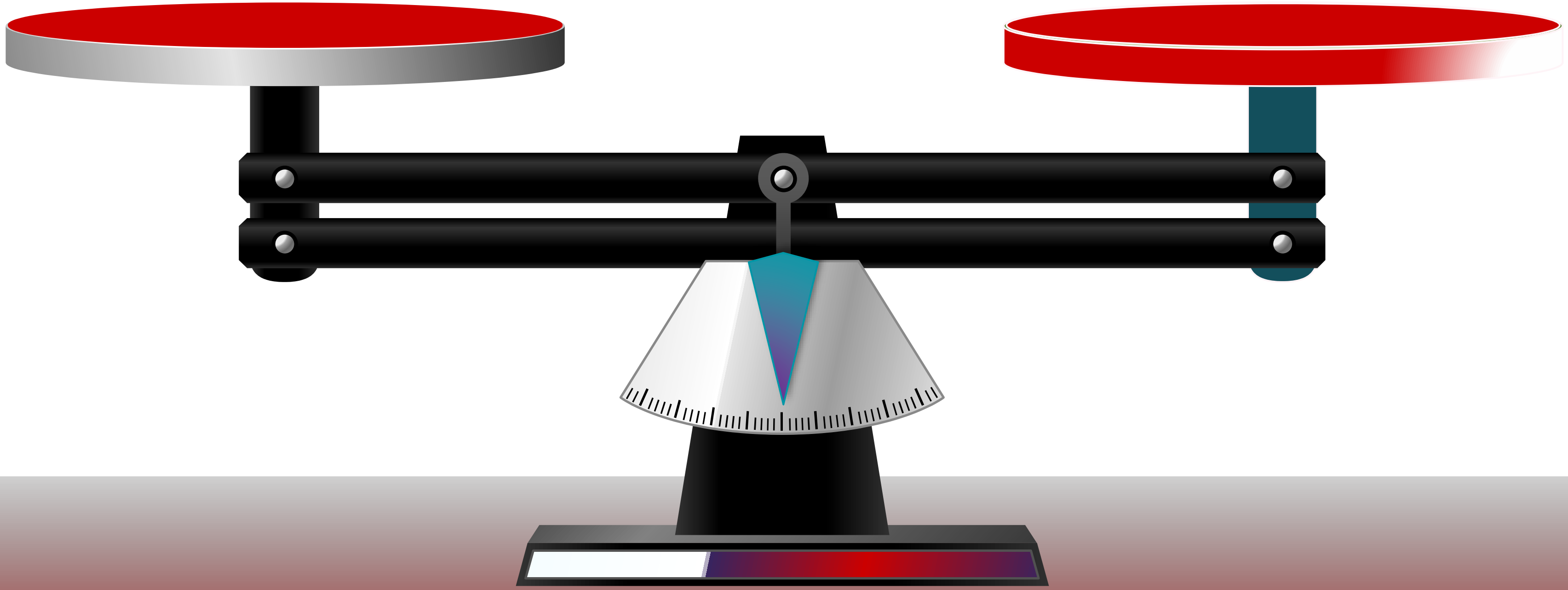
4 . Previous





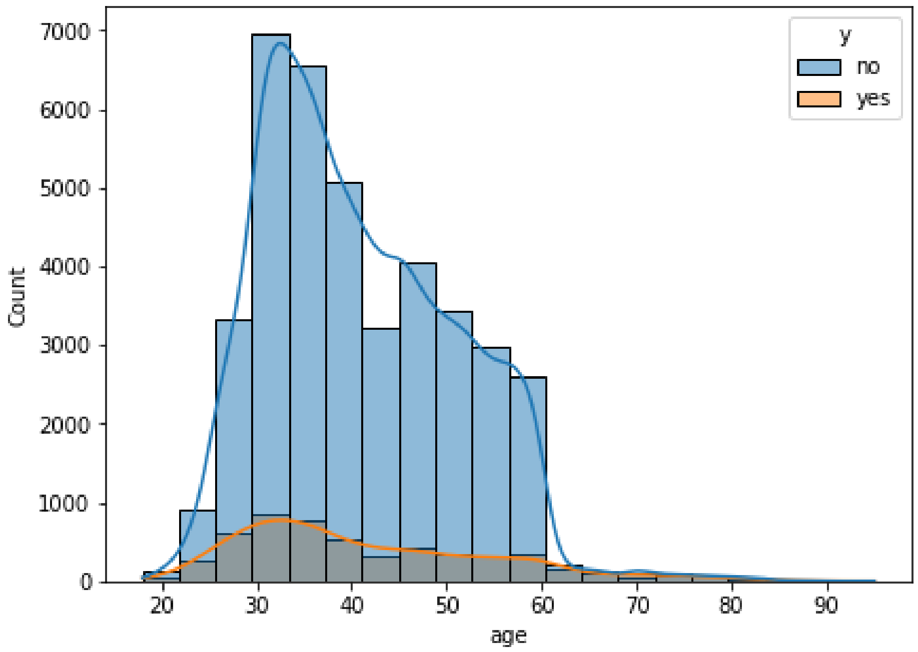


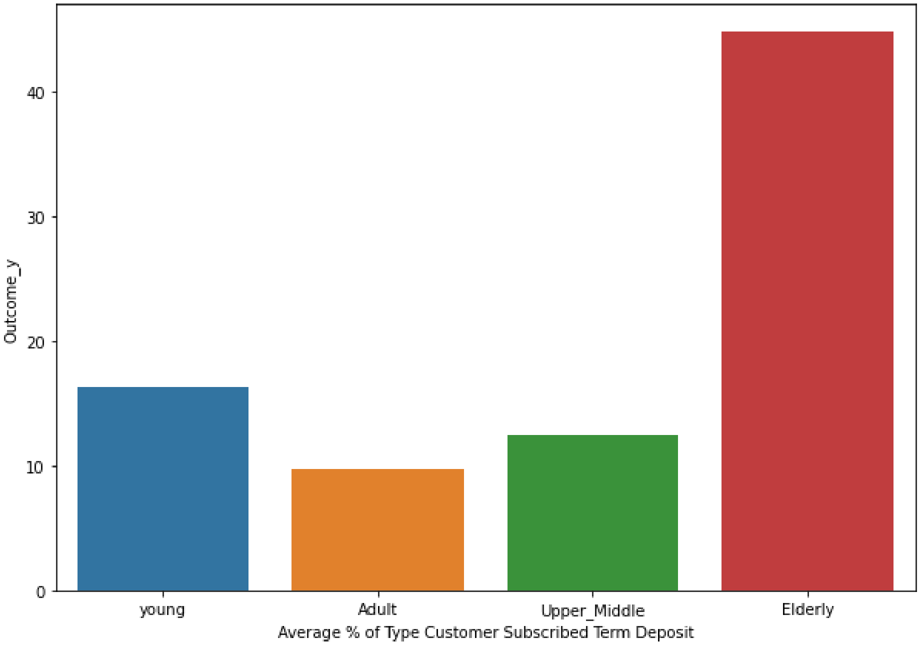
Univariate Analysis **Bivariate Analysis**



EDA

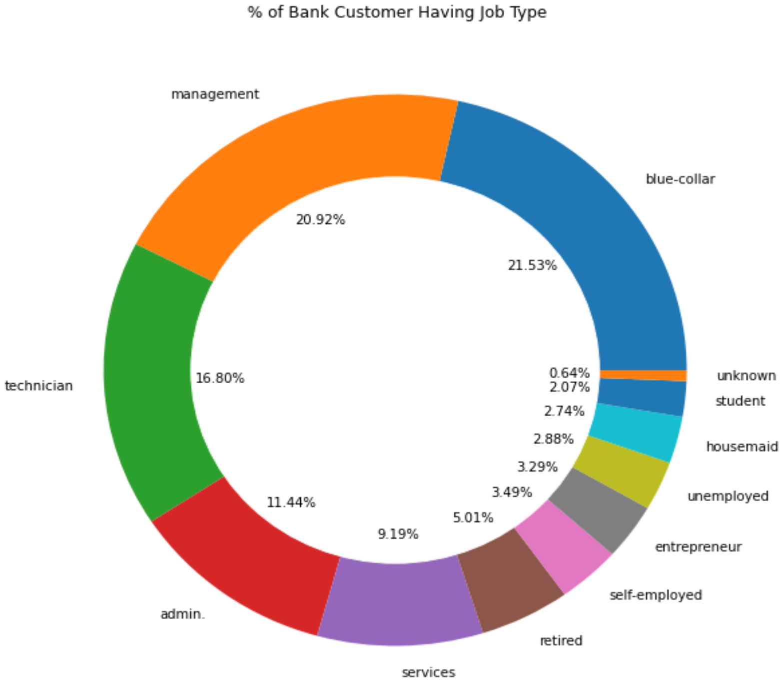
Based on Age Group

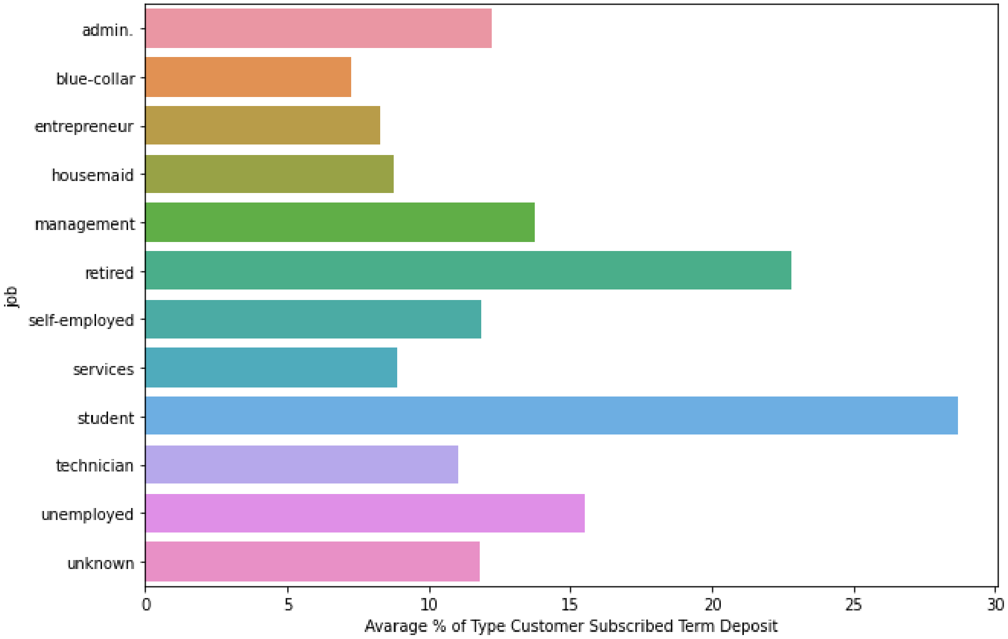




Clients of age above 60 years and under 30 years have a higher probability of subscribed term deposit.

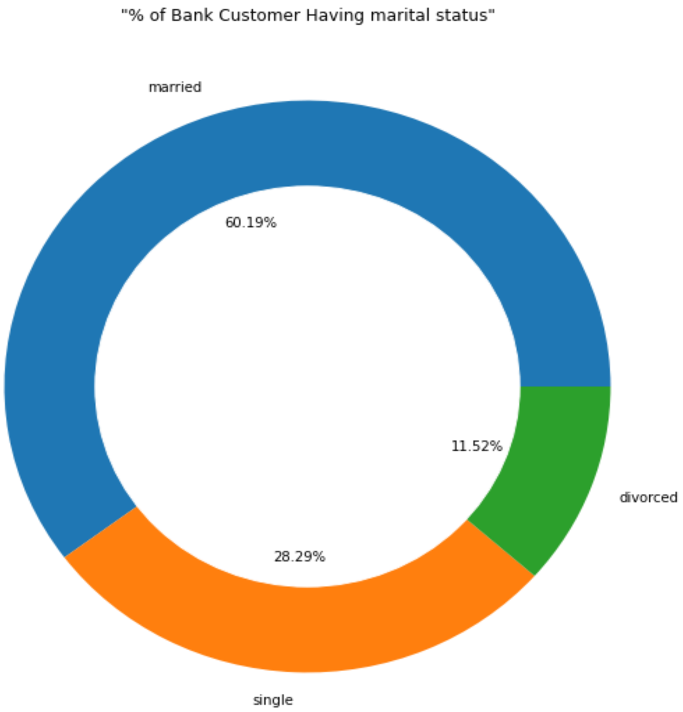
Types of Job

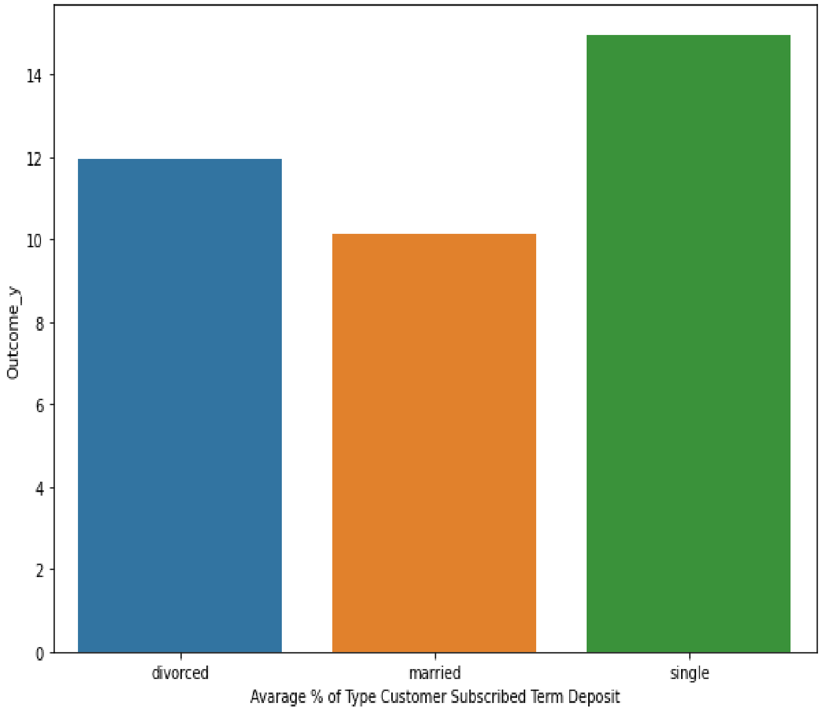




Students and retired clients account are having average more than 50% of subscription rate.

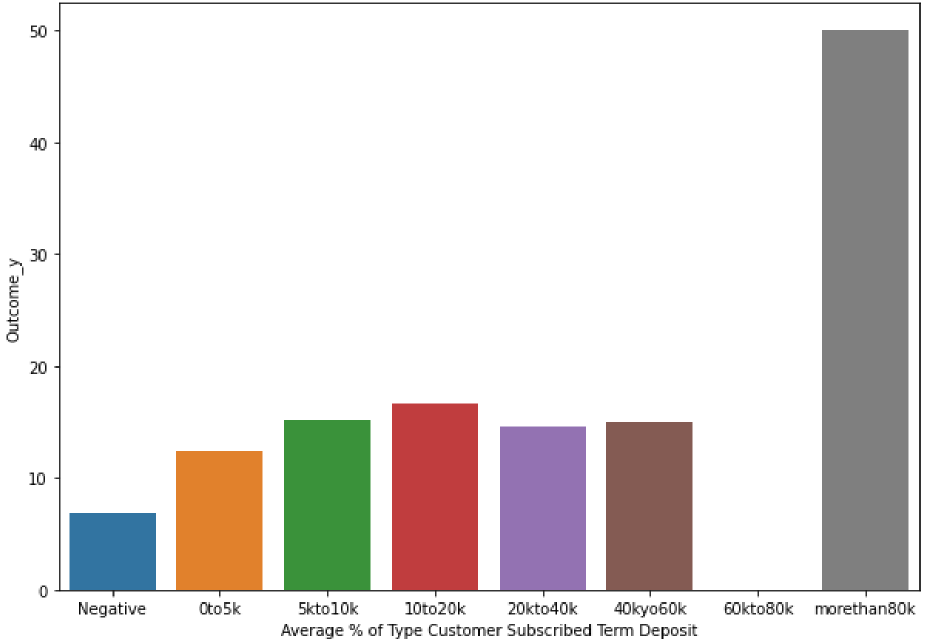
Marital Status

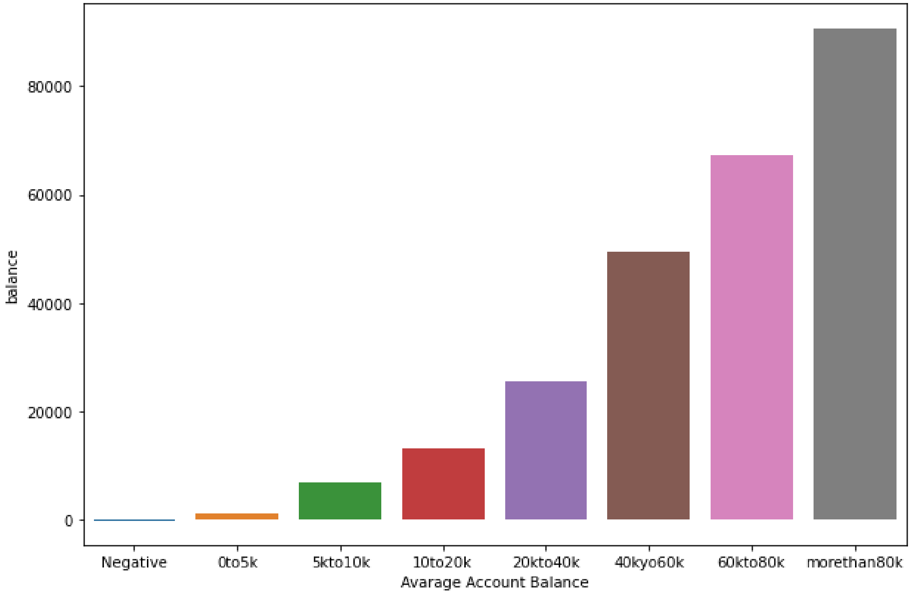




Single clients account are having average more than 14 % of subscription rate.

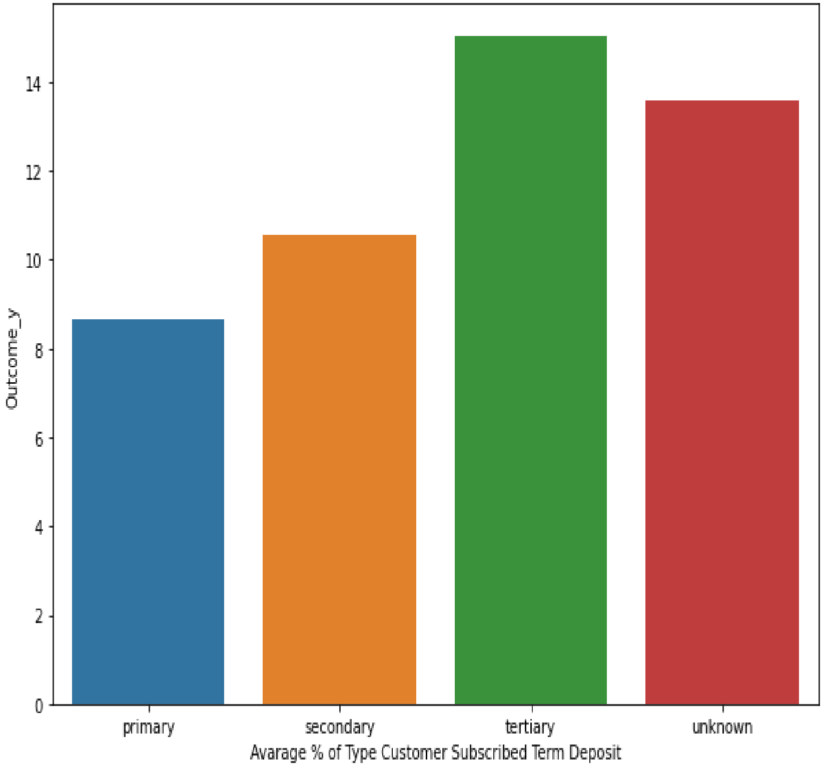
Based on the Account Balance

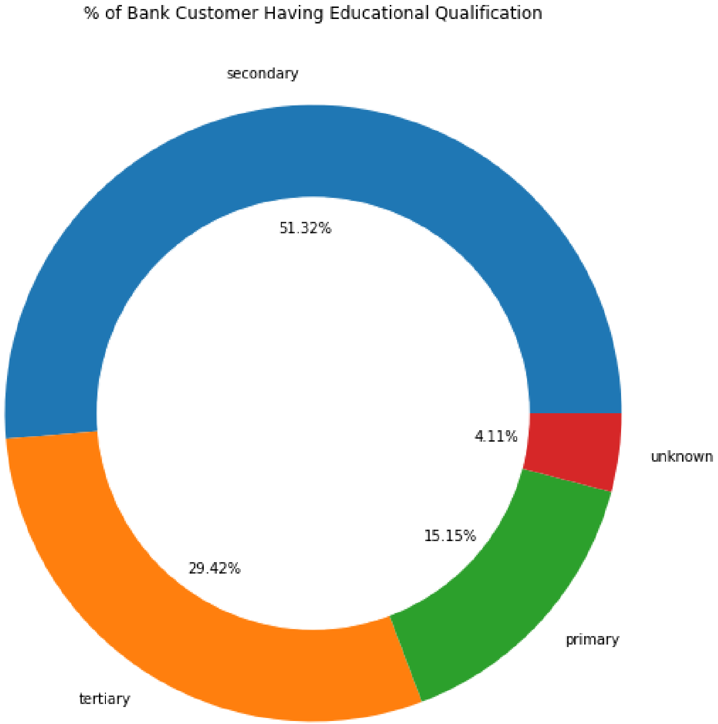




Clients with negative balances only returned a subscription rate of 6.9% while **clients with average or high balances had significantly higher subscription rates** of average 15%.

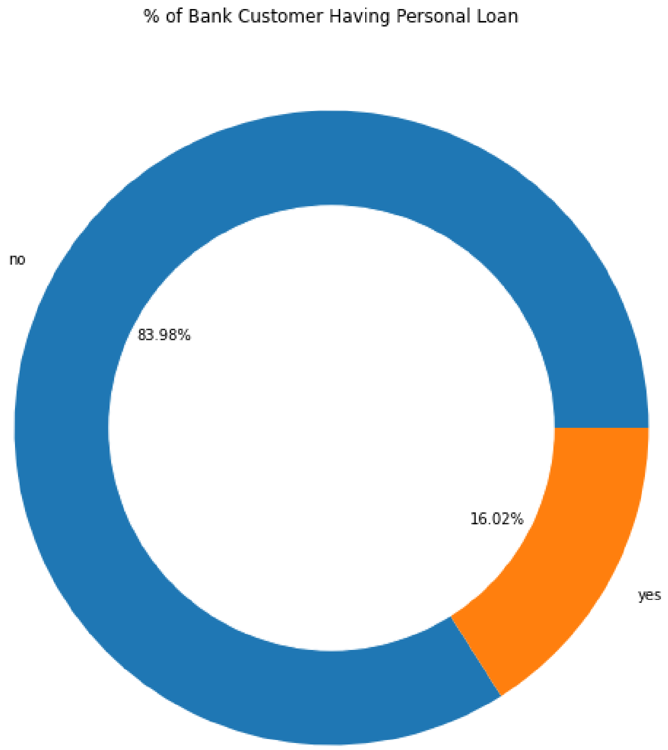
Education Qualification

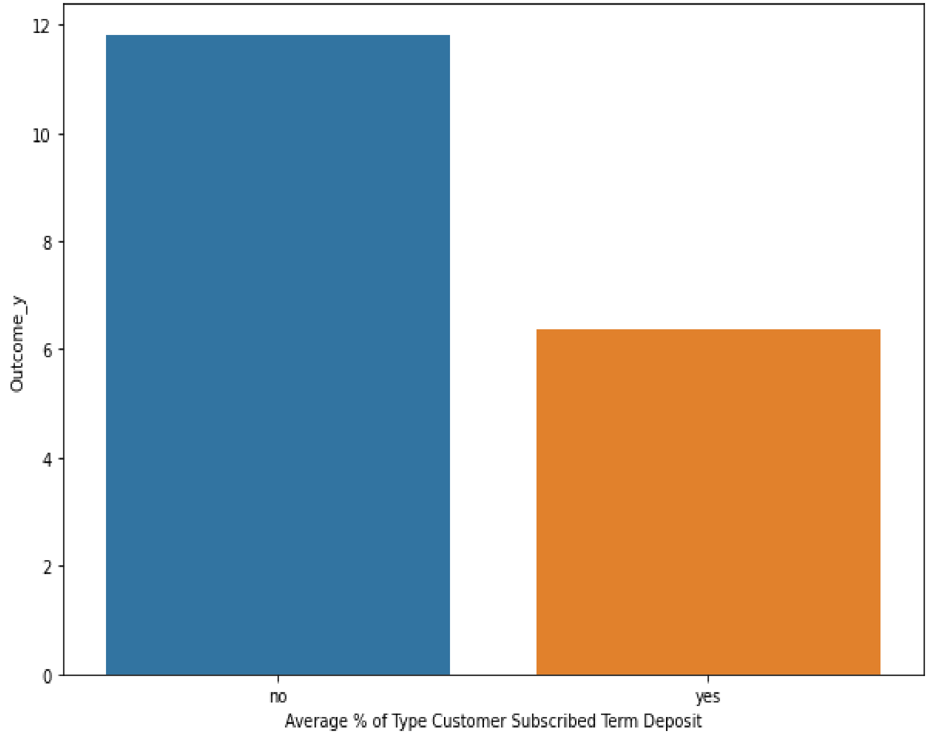




Customer who are **highly educated subscribed more** to term deposit plan.

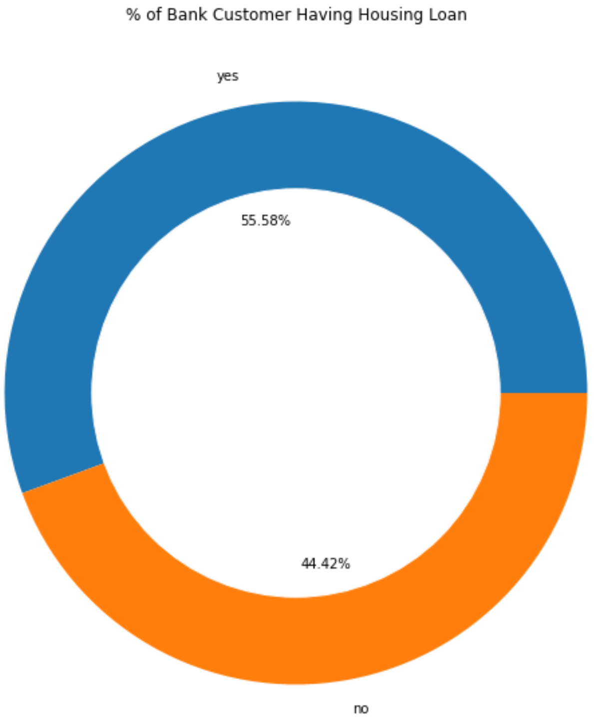
Default in loan payment

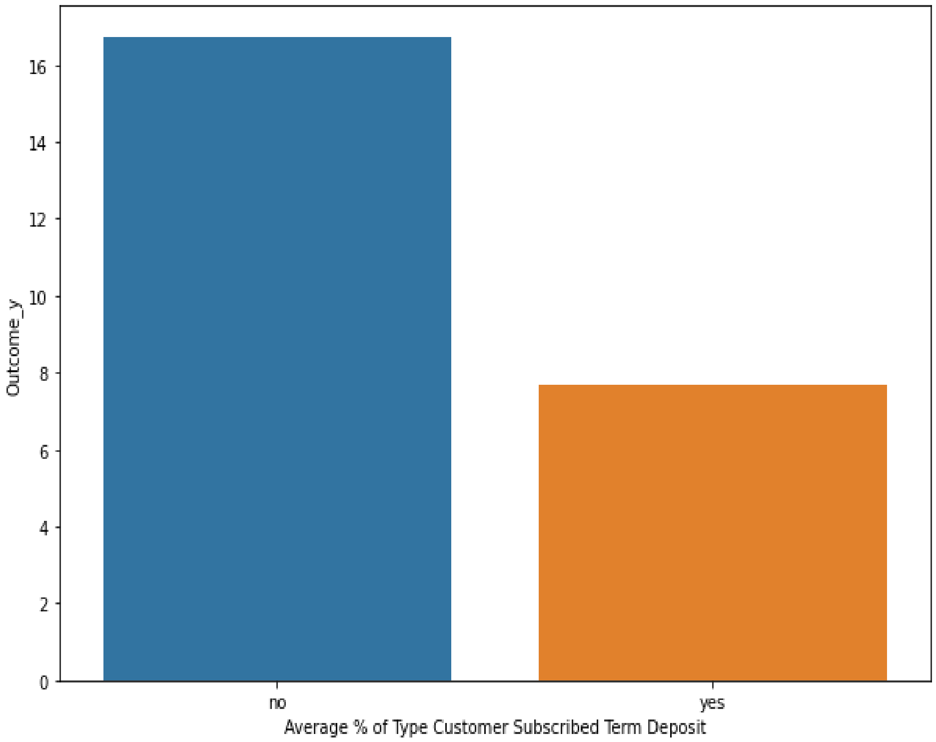




The Customer **who haven't default loan account** are having high subscription rate.

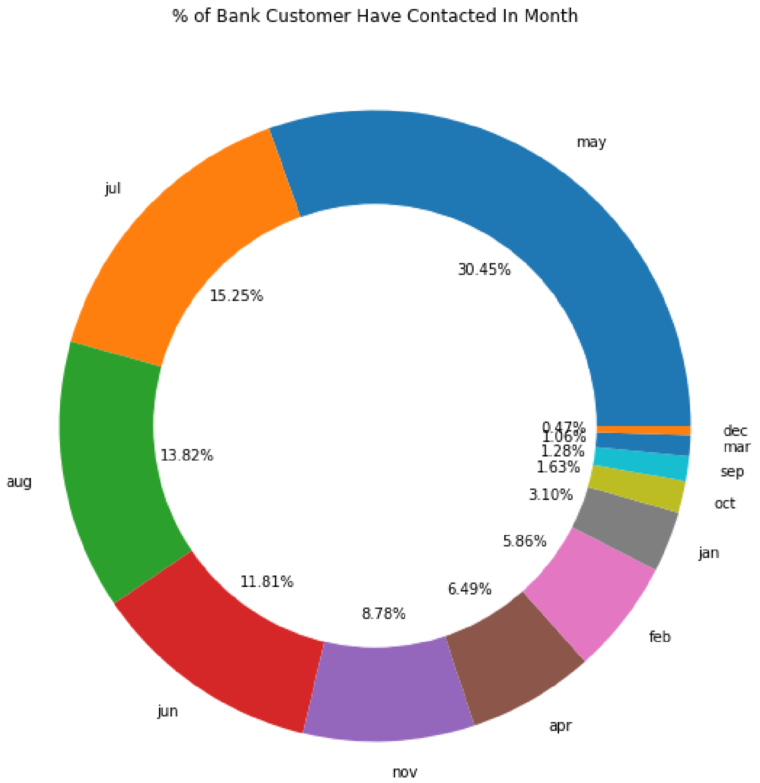
Housing loan

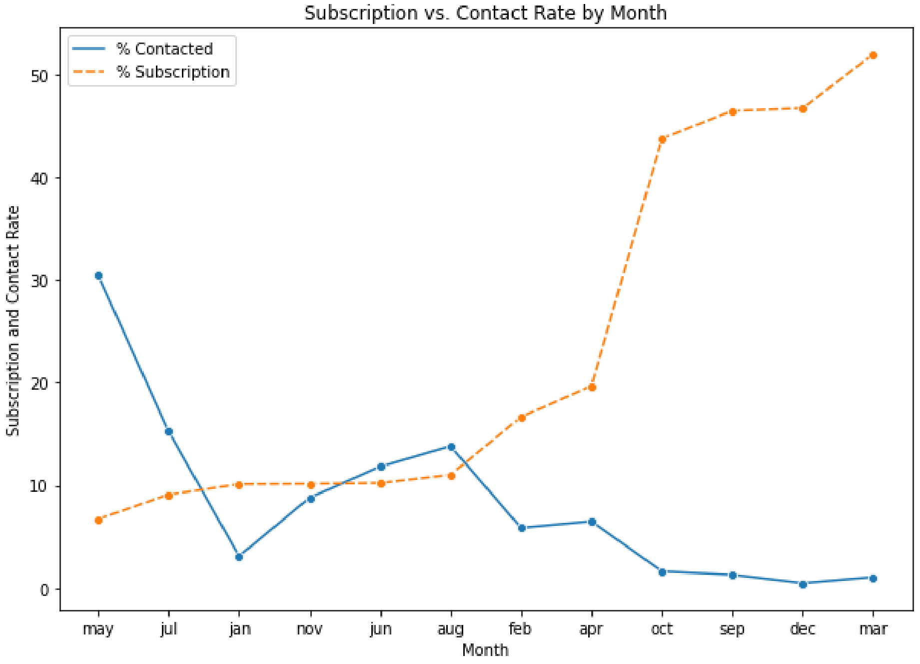




The Customer **who haven’t active housing/loan** are having high subscription rate.

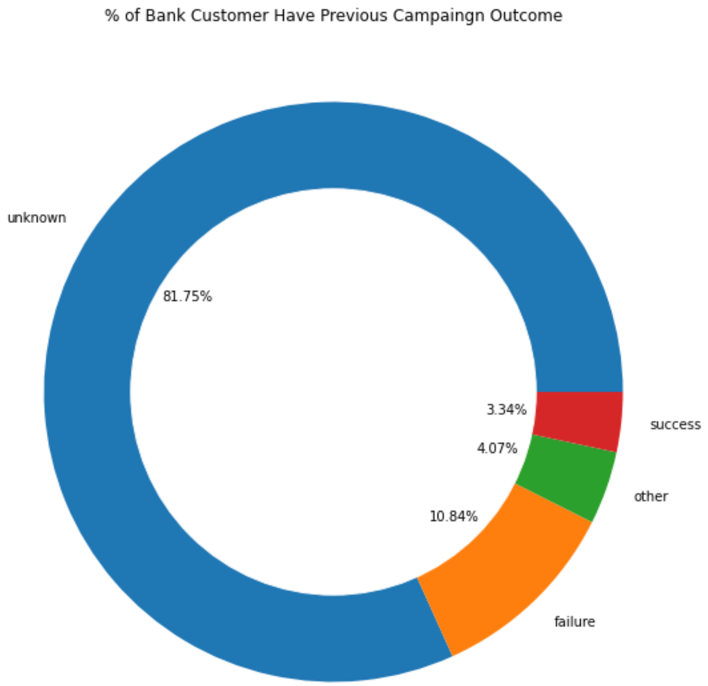
Month

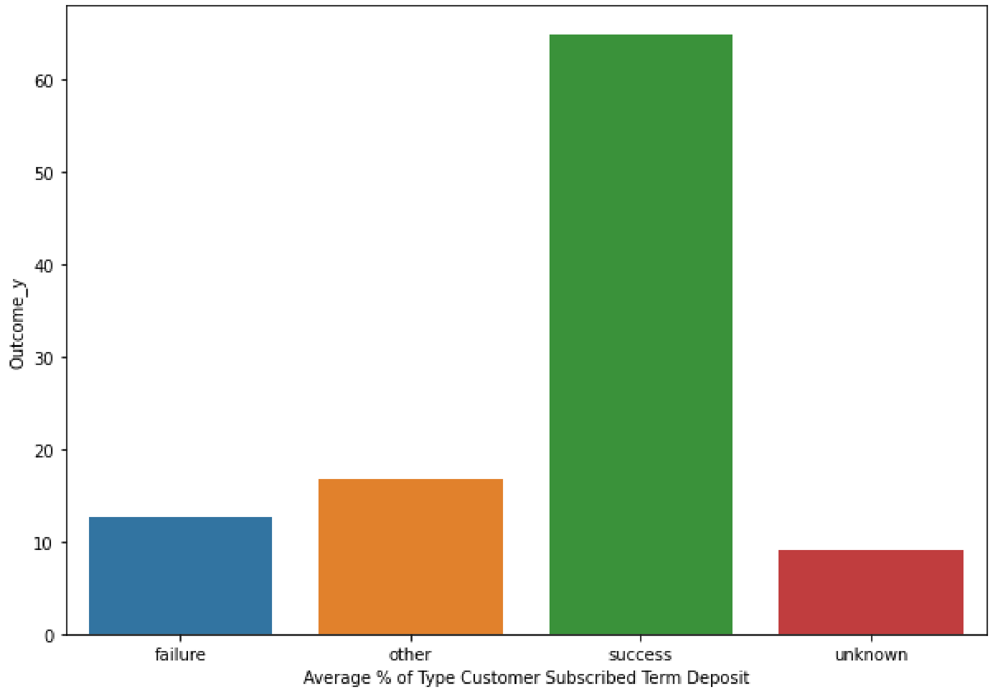




- The bank **contacted most clients between May and August** & **least in March, September, October, and December.**
- The **highest subscription rate occurred in March, which is over 50%**, and all subscription rates **in September, October, and December are over 40%.**

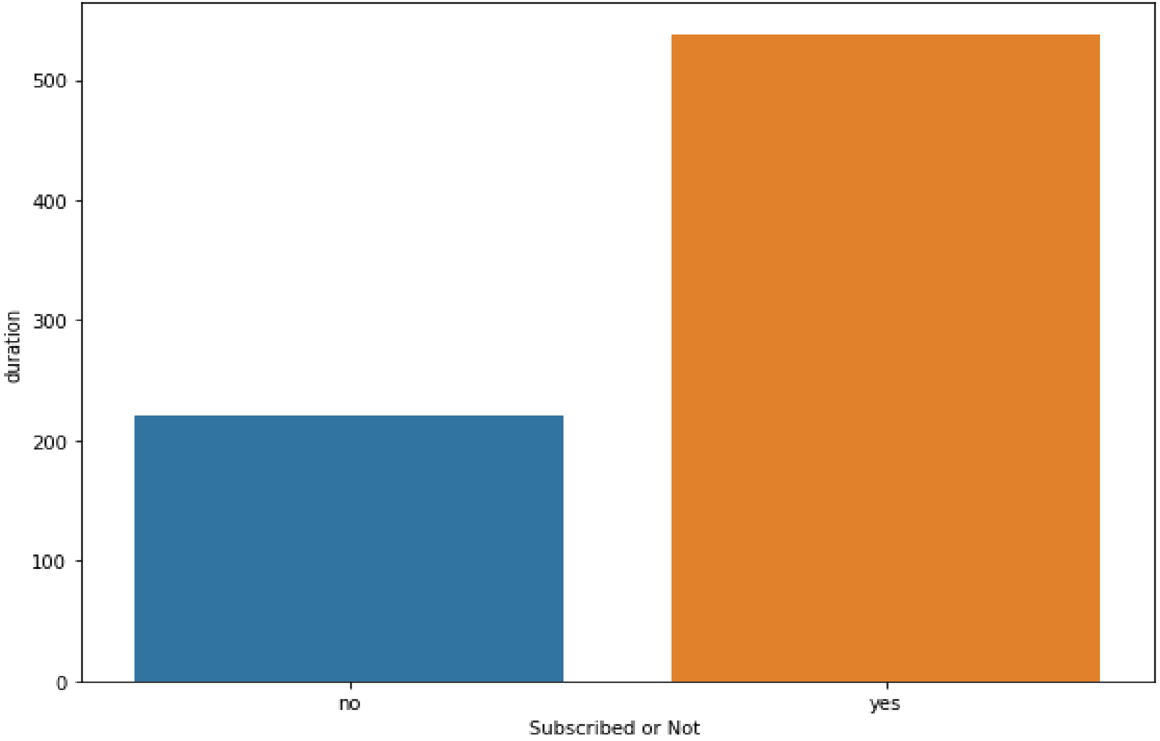
Outcome of previous campaign





The Clients **who have subscribed in previous campaign** also have higher probability to subscribed current product as well.

Total call duration



The **duration of the last call in seconds, is more than twice** for the customers who subscribed the products than for customers who didn’t.

Feature Engineering

AI

Creation of New Feature by Label Encoding & One hot Encoding

I. Convert Categorical Feature which are having two class output (Yes/No)

1. **default**: has credit in default? (Categorical: Yes/No)
2. **housing**: has housing loan? (Categorical: Yes/No)
3. **loan**: has personal loan? (Categorical: Yes/No)
4. **y** : Has the client subscribed a term deposit? (Binary: : Yes/No)

II. By Using One hot Encoding create Dummy Variable of following Multiclass Features

4. job: type of job (categorical: admin, blue-collar, entrepreneur, housemaid etc.)

5. marital : marital status (categorical: divorced, married ,single)

6.education : education (categorical: primary , secondary , tertiary, unknown)

7.contact: contact communication type (categorical: cellular, telephone, unknown)

8. poutcome: outcome of the previous campaign (categorical: failure, nonexistent, success)

III. Convert Categorical Feature into Numerical Features by label Encoding

9. **month**: last contact month of year (categorical: jan, feb, mar, ..., nov, dec)

Preparing Dataset for Modeling

1. Normalization of Dataset

- **Normalizing the Dataset using MinMaxScaler Technique.**
- **MinMaxScaler scales all the data features in the range [0, 1].**

2. Dealing With Class Imbalance

Synthetic Minority Oversampling Technique (SMOTE)

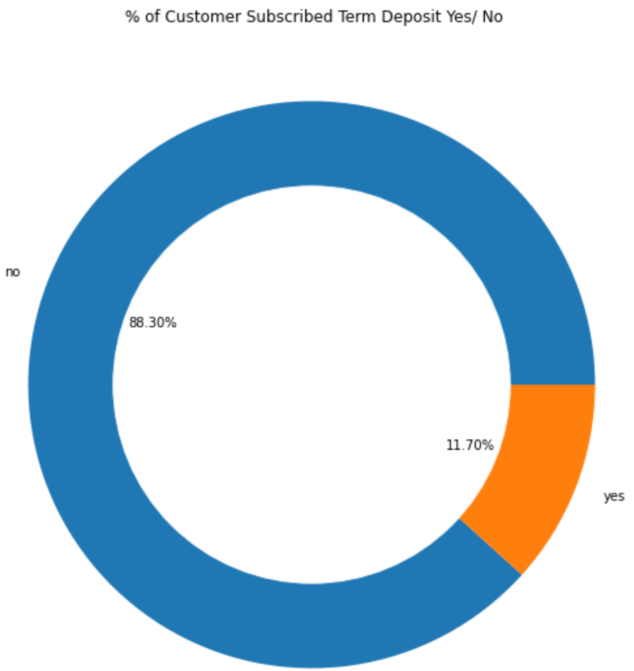
```
from imblearn.over_sampling import SMOTE
smote = SMOTE()
# fit predictor and target variable
x_smote, y_smote = smote.fit_resample(scaled_df.iloc[:,0:-1], scaled_df['y'])

print('Original dataset shape', len(scaled_df))
print('Resampled dataset shape', len(y_smote))
```

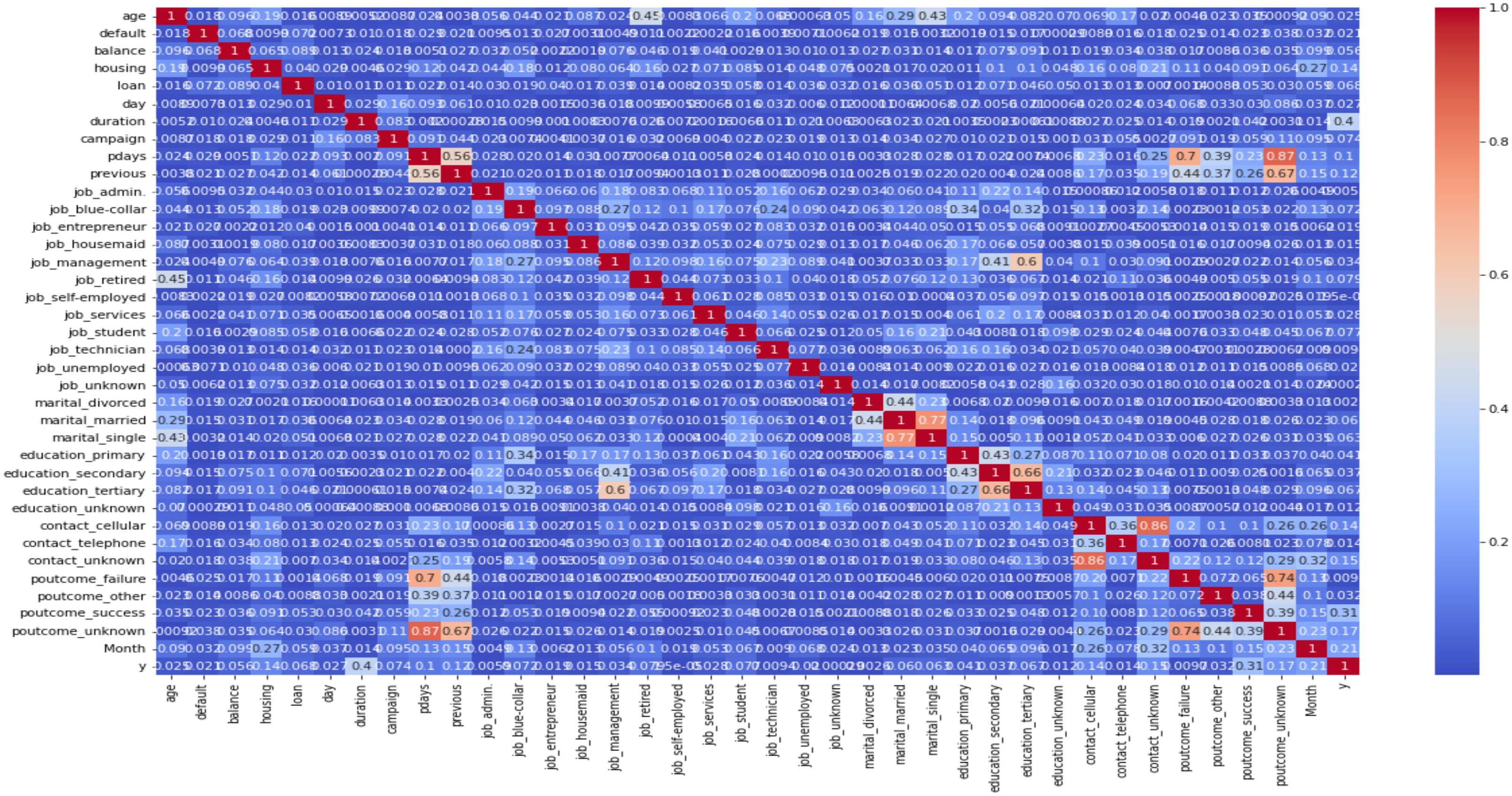
```
Original dataset shape 44988
Resampled dataset shape 79422
```

```
# So now class is balanced
y_smote.value_counts()
```

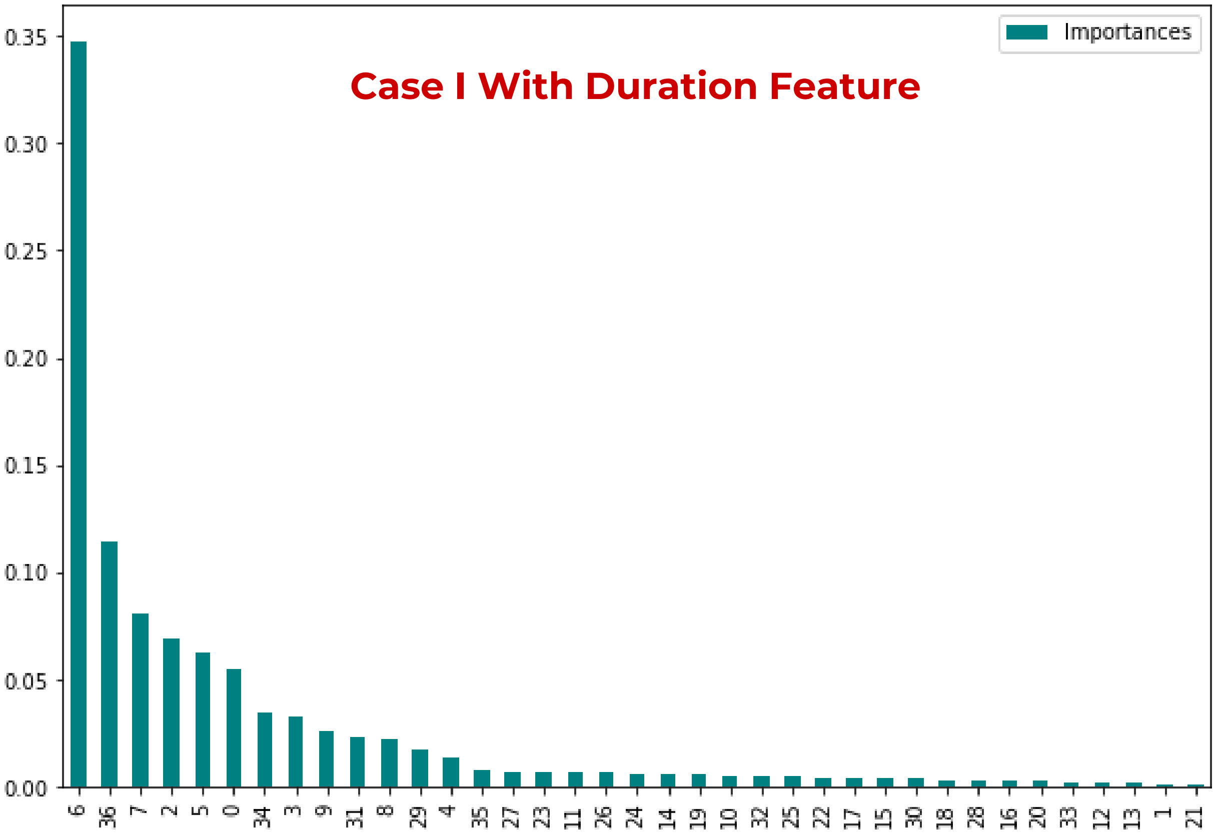
```
0.0    39711
1.0    39711
Name: y, dtype: int64
```



Feature Correlation

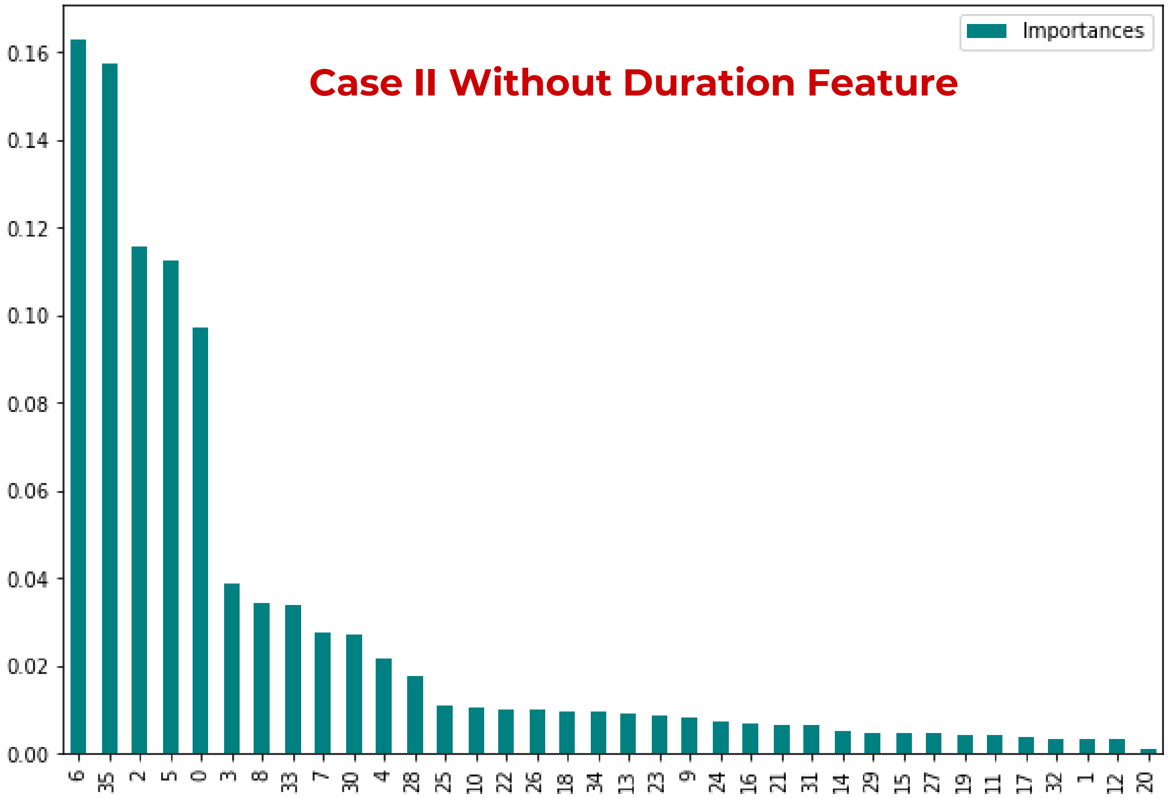


Feature Importance



	Features	Importances
6	duration	0.347661
36	Month	0.114472
7	campaign	0.080380
2	balance	0.068840
5	day	0.061938
0	age	0.054817
34	poutcome_success	0.034522
3	housing	0.032668
9	previous	0.025494
31	contact_unknown	0.023071
8	pdays	0.022013
29	contact_cellular	0.017281
4	loan	0.013376
35	poutcome_unknown	0.007648

Feature Importance



	Features	Importances
6	campaign	0.162973
35	Month	0.157483
2	balance	0.115728
5	day	0.112326
0	age	0.097296
3	housing	0.038552
8	previous	0.034235
33	poutcome_success	0.033878
7	pdays	0.027634
30	contact_unknown	0.026863
4	loan	0.021565
28	contact_cellular	0.017665
25	education_secondary	0.010642
10	job_blue-collar	0.010546
22	marital_married	0.009996

Feature Selection

Independent variables

Dependent variables

CASE I

- Duration
- campaign
- month
- balance
- day
- age
- housing
- poutcome success
- previous
- contact unknown
- loan

CASE II

- campaign
- month
- balance
- day
- age
- housing
- poutcome success
- previous
- contact unknown
- loan
- education secondary
- marital married
- job blue collar

Feature Importance > 0.01

Preparing Dataset (Train Test Split)

```
independent_variables= final_df[final_df.Importances>0.005].Features.to_list()
print(f'independent_variables are {independent_variables}')

dependent_variables = 'y'
print(f'independent_variables are {dependent_variables}')
```

```
independent_variables are ['duration', 'Month', 'campaign', 'balance', 'day', 'age', 'poutcome_success', 'housing', 'previous', 'contact_unknown', 'pdays', 'contact_c
independent_variables are y
```

```
# Creating the dataset with all independent variables
X = x_smote[independent_variables]

# Creating the dataset with the dependent variable
Y = y_smote
```

```
print(X.shape,Y.shape)
```

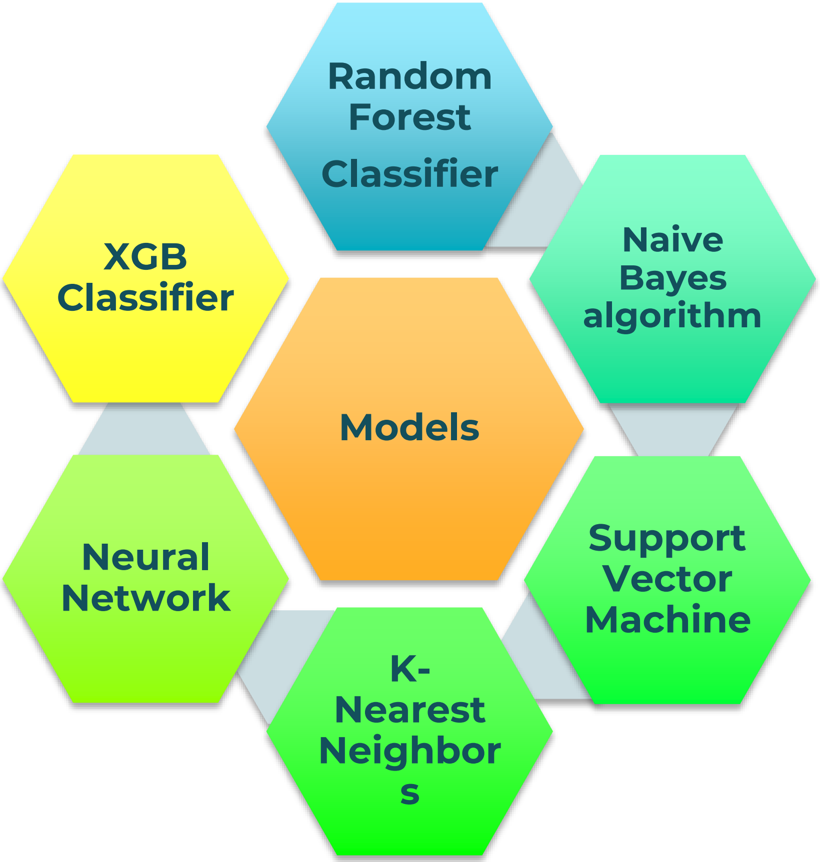
```
(79422, 22) (79422,)
```

```
#Lets Split The dataset Into Test & Train dataset
from sklearn.model_selection import train_test_split

# Splitting the dataset into the Training set and Test set
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.25, random_state=0,stratify= Y)
```

Train Size 75 % & Test Size 25%

Machine Learning Models



1. Random Forest Classifier

AI

Evaluation Metrics	Class	With Duration		Without Duration	
Score		Train Data	Test Data	Train Data	Test Data
Accuracy		0.9181	0.9012	0.8664	0.8624
Roc_Auc_Score		0.9180	0.9012	0.8664	0.8624
Precision	0	0.95	0.94	0.83	0.82
	1	0.89	0.87	0.90	0.89
Recall	0	0.88	0.86	0.91	0.90
	1	0.96	0.95	0.82	0.80
F1 Score	0	0.91	0.90	0.87	0.86
	1	0.92	0.91	0.86	0.84

2. Naive Bayes Algorithm

Evaluation Metrics	Class	With Duration		Without Duration	
Score		Train Data	Test Data	Train Data	Test Data
Accuracy		0.7839	0.7850	0.6893	0.6945
Roc_Auc_Score		0.7839	0.7850	0.6893	0.6945
Precision	0	0.75	0.75	0.67	0.67
	1	0.83	0.83	0.72	0.71
Recall	0	0.86	0.85	0.75	0.75
	1	0.71	0.71	0.63	0.63
F1 Score	0	0.80	0.80	0.71	0.71
	1	0.77	0.77	0.67	0.67

3. Support vector classifier

Evaluation Metrics	Class	With Duration		Without Duration	
Score		Train Data	Test Data	Train Data	Test Data
Accuracy		0.8711	0.8668	0.7310	0.7327
Roc_Auc_Score		0.8711	0.8668	0.7310	0.7327
Precision	0	0.90	0.90	0.71	0.71
	1	0.85	0.84	0.77	0.76
Recall	0	0.84	0.83	0.79	0.78
	1	0.91	0.90	0.68	0.68
F1 Score	0	0.87	0.86	0.75	0.75
	1	0.88	0.87	0.72	0.72

4. K-Neighbours Classifier

Evaluation Metrics	Class	With Duration		Without Duration	
Score		Train Data	Test Data	Train Data	Test Data
Accuracy		1.0	0.9207	0.9999	0.8681
Roc_Auc_Score		1.0	0.9207	0.9999	0.8681
Precision	0	1.00	0.95	1.00	0.89
	1	1.00	0.89	1.00	0.85
Recall	0	1.00	0.89	1.00	0.84
	1	1.00	0.95	1.00	0.89
F1 Score	0	1.00	0.82	1.00	0.86
	1	1.00	0.92	1.00	0.87

5. Neural Network

Evaluation Metrics	Class	With Duration		Without Duration	
Score		Train Data	Test Data	Train Data	Test Data
Accuracy Score		0.8637	0.8593	0.7333	0.7327
Roc_Auc_Score		0.8637	0.8593	0.7333	0.7327
Precision	0	0.90	0.90	0.72	0.72
	1	0.85	0.84	0.78	0.77
Recall	0	0.83	0.83	0.80	0.80
	1	0.91	0.91	0.69	0.69
F1 Score	0	0.87	0.86	0.76	0.75
	1	0.88	0.87	0.73	0.73

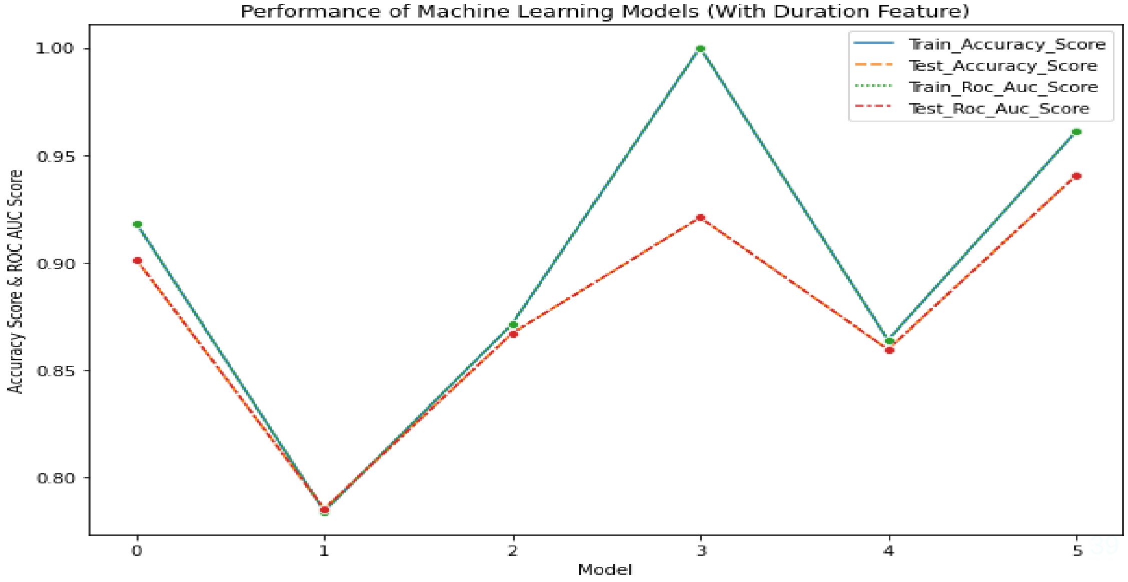
6. XGB Classifier

Evaluation Metrics	Class	With Duration		Without Duration	
Score		Train Data	Test Data	Train Data	Test Data
Accuracy		0.9607	0.9403	0.9411	0.9282
Roc_Auc_Score		0.9607	0.9403	0.9411	0.9282
Precision	0	0.96	0.94	0.91	0.90
	1	0.96	0.94	0.98	0.97
Recall	0	0.96	0.94	0.98	0.80
	1	0.96	0.94	0.90	0.89
F1 Score	0	0.96	0.94	0.94	0.93
	1	0.96	0.94	0.94	0.93

Model Validation and Selection

	Model Name	Train_Accuracy_Score	Test_Accuracy_Score	Train_Roc_Auc_Score	Test_Roc_Auc_Score
0	Random Forest Classifier	0.918091	0.901239	0.918091	0.901239
1	Naive Bays Algorithem	0.783988	0.785052	0.783988	0.785052
2	Support Vector Machine	0.871168	0.866841	0.871168	0.866841
3	KNeighborsClassifier	1.000000	0.920780	1.000000	0.920780
4	Sequential Neural Network	0.863731	0.859337	0.863731	0.859337
5	XGBoost Classifier	0.960783	0.940320	0.960783	0.940320

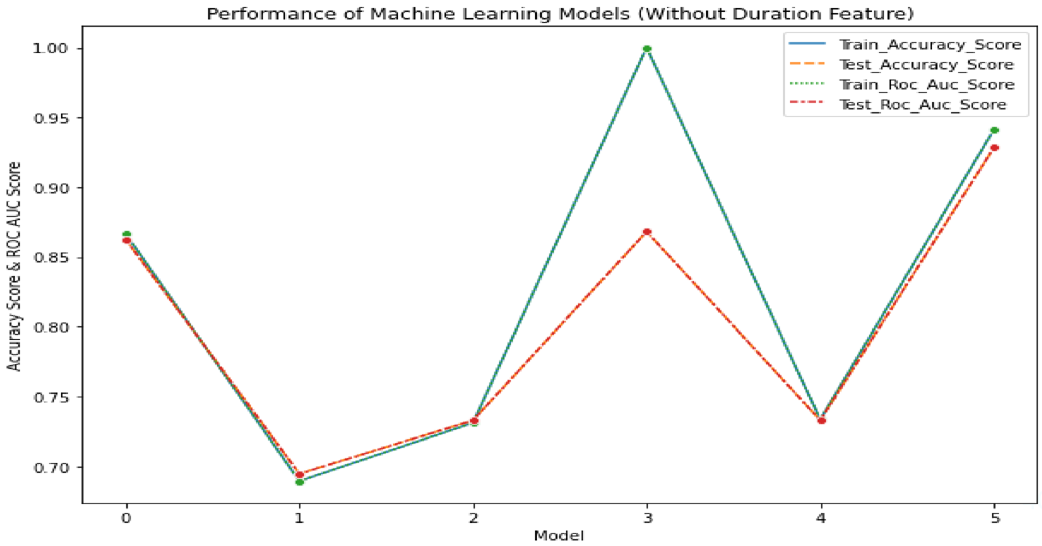
Case I
With Duration Feature



Model Validation and Selection

	Model Name	Train_Accuracy_Score	Test_Accuracy_Score	Train_Roc_Auc_Score	Test_Roc_Auc_Score
0	Random Forest Classifier	0.866451	0.862460	0.866451	0.862460
1	Naive Bays Algorithem	0.689319	0.694551	0.689319	0.694551
2	Support Vector Machine	0.731004	0.732726	0.731004	0.732726
3	KNeighborsClassifier	0.999966	0.868100	0.999966	0.868100
4	Sequential Neural Network	0.733388	0.732776	0.733388	0.732776
5	XGBoost Classifier	0.941141	0.928233	0.941141	0.928233

Case II
Without Duration Feature



Conclusion

Clients of age above 60 years and under 30 years have a higher probability of subscribed term deposit.

The Customer **who haven't active housing/loan** are having high subscription rate.

The Clients **who have subscribed in previous campaign** also have higher probability to subscribed current product as well.

The **duration of the last call in seconds, is more than twice** for the customers who subscribed the products than for customers who didn't.

Clients with negative balances only returned a subscription rate of 6.9% while **clients with average or high balances had significantly higher subscription rates** of average 15%.

However, in this campaign, more than 50% of clients contacted those who have a low balance. So in the future, the bank **should have to shift its marketing focus on high-balance customers to secure more term deposits.**

Continue...

The **bank contacted most clients between May and August.** The highest contact rate is around 30%, which happened in May, while the **contact rate is low in March, September, October, and December.**

However, the subscription rate showed a different trend. **The highest subscription rate occurred in March**, which **is over 50%**, and **all subscription rates in September, October, and December are over 40%.**

To improve the marketing campaign, the **bank should consider initiating the telemarketing campaign in fall and spring when the subscription rate tends to be higher.**
By applying **Random Forest Classifier , KNN Neighbours classifiers & XGB Classifier** classification model were successfully built with descent results. With help of these three models, the bank **will be able to predict a customer's response to its telemarketing campaign before calling this customer**.

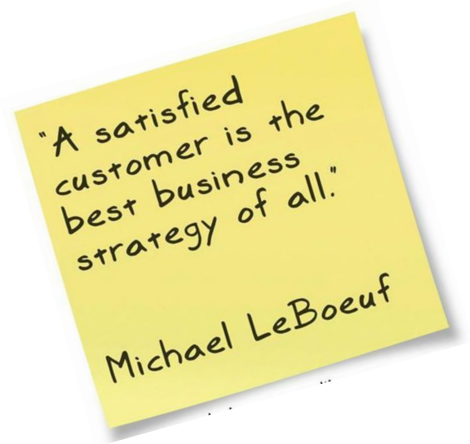
The **best machine learning model is XGBoost Classifier**, which resulted in **best AUC score & one of the best Precision & Recall Value** among the all classification model in both the cases.

Continue…

In this way, the **bank can allocate more marketing efforts to the clients who are classified as highly likely to accept term deposits,** and call less to those who are unlikely to make term deposits.

Increase the **efficiency of the bank's telemarketing campaign**, **saving time and efforts** , prevents some clients from receiving undesirable advertisements, raising **customer satisfaction**.

With the aid of above Machine Learning models, the bank **can enter a virtuous cycle of effective marketing, more investments and happier customers.**




thank
YOU