Term Deposit (FixedDeposit) Prediction -Banking Case Study

Submitted by: Harsha KG
Roll No:2248035

Problem Statement:

- A major bank in Middle East came to NeoStats with help in analysing its current customer base and its marketing campaign. It wants to understand which customers are most likely to take a term deposit (fixed deposit), and then send this list to their call centre.

Objective:

1. • **Data Analysis & Visualization:**
2. . **Modelling:** Term Deposit and Related Variables:Identify variables strongly related to Term Deposit. Discuss your approach when the variable is categorical. Which tests or metrics will you employ?

- **Predictive Model Building:**Train a prediction model of your choice to estimate the probability that a customer will opt for a term deposit. Adhere to an 80:20 train:test split. Report and present the model's performance metrics on both the train and test datasets.

- **Model Improvement Strategies:** Discuss potential methods or approaches to enhance model performance. This could include feature engineering, different algorithms, or refining the data preprocessing steps.

1. Understand the Problem Statement

- **Goal: Using the collected data from the existing customers, build a model that will help the marketing team identify potential customers who are relatively more likely to subscribe term deposits and thus increase their hit ratio.**

- **Data:**

	Variable	Description	Data Type
0	Customer_number	Unique Customer Identification number	ID
1	Acc_creation_date	Account opening date	Date
2	Insurance	Has the customer taken insurance?	Categorical
3	balance	NaN	Numeric
4	housing	Has the customer taken housing loan?	Categorical
5	loan	Has the customer taken personal loan?	Categorical
6	contact	Contact communication type	Categorical
7	duration	Duration of call with the customer for Term loan	Numeric
8	campaign	Number of contacts performed during this campa...	Numeric
9	last_contact_day	Number of days that passed by after the client...	Numeric
10	previous	Number of contacts performed before this campa...	Categorical
11	poutcome	Outcome of the previous marketing campaign	Categorical
12	Term Deposit	Has the customer subscribed a term deposit (fi...	Target (categorical)
13	Count_Txn	Number of Transactions Done by the customer	Numeric
14	NaN	NaN	NaN
15	NaN	NaN	NaN
16	Customer_number	Unique Customer Identification number	ID
17	age	Age of customer in years	Numeric
18	job	Type of job the customer has	Categorical
19	marital	Marital status of the customer	Categorical
20	education	Highest education level of customer	Categorical
21	Annual Income	Annual income of the customer	Numeric
22	Gender	Gender of the customer	Categorical

Combine the dataframes transaction_data and Customer_Demographics dataframe using customer number

	Sno	Customer_number	Insurance	balance	housing	loan	contact	duration	campaign	last_contact_day	previous	poutcome	Term Deposit	Count_Txn
0	0	1001	no	2143	yes	no	NaN	261.0	1	2	0	unknown	no	351.0
1	1	1002	no	29	yes	no	unknown	151.0	1	2	0	unknown	no	326.0
2	2	1003	no	2	yes	yes	unknown	76.0	1	2	0	NaN	no	422.0
3	3	1004	no	1506	yes	no	unknown	92.0	1	2	0	unknown	no	113.0
4	4	1005	no	1	no	no	unknown	198.0	1	2	0	unknown	no	342.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
45206	45206	46207	no	825	no	no	cellular	0.0	0	-1	0	unknown	yes	152.0
45207	45207	46208	no	1729	no	no	cellular	0.0	0	-1	0	unknown	yes	334.0
45208	45208	46209	no	5715	no	no	cellular	1127.0	5	184	3	success	yes	381.0
45209	45209	46210	no	668	no	no	telephone	0.0	0	-1	0	unknown	no	211.0
45210	45210	46211	no	2971	no	no	cellular	361.0	2	188	11	other	no	331.0

45211 rows × 20 columns

2.Data Preprocessing

2.1. Header correction for data_dictionary dataframe

2.2, remove the column Unnamed: 0

2.3. Understanding Attributes

- **Shape of the DataFrames**

```
Shape of df_data: (23, 3)
Shape of df_trans_data: (45211, 14)
Shape of df_cust_data: (45211, 7)
```

- **Info()**

```
Info of df_trans_data: None
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45211 entries, 0 to 45210
Data columns (total 7 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Customer_number  45211 non-null  int64
 1   age              45211 non-null  int64
 2   job              45198 non-null  object
 3   marital          45193 non-null  object
 4   education        45190 non-null  object
 5   Annual Income    45194 non-null  object
 6   Gender           45211 non-null  object
dtypes: int64(2), object(5)
memory usage: 2.4+ MB
Info of df_cust_data: None
```

```
Info of df_data: None
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45211 entries, 0 to 45210
Data columns (total 14 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Sno              45211 non-null  int64
 1   Customer_number  45211 non-null  int64
 2   Insurance        45211 non-null  object
 3   balance          45156 non-null  object
 4   housing          45211 non-null  object
 5   loan             45203 non-null  object
 6   contact          45168 non-null  object
 7   duration         45172 non-null  float64
 8   campaign         45211 non-null  int64
 9   last_contact_day 45211 non-null  int64
 10  previous         45211 non-null  int64
 11  poutcome         45196 non-null  object
 12  Term Deposit     45203 non-null  object
 13  Count_Txn        45210 non-null  float64
```

There are 9 integer(Numeric) type attribute and 12 categorical attribute

- Describe ()

Describe() statitics method here only serve numerical attribute and attibutes which are categorical in nature here is ignored.So we use include='object', provides categorical data information.

	Sno	Customer_number	duration	campaign	last_contact_day	previous	Count_Txn	age
count	45211.000000	45211.000000	45172.000000	45211.000000	45211.000000	45211.000000	45210.000000	45211.000000
mean	22605.000000	23606.000000	258.139511	2.762182	41.832253	0.580323	299.614952	40.980005
std	13051.435847	13051.435847	257.631452	3.087291	99.457030	2.303441	115.721788	10.838273
min	0.000000	1001.000000	-167.000000	0.000000	-9.000000	0.000000	-423.000000	18.000000
25%	11302.500000	12303.500000	103.000000	1.000000	1.000000	0.000000	200.000000	33.000000
50%	22605.000000	23606.000000	180.000000	2.000000	1.000000	0.000000	300.000000	39.000000
75%	33907.500000	34908.500000	319.000000	3.000000	1.000000	0.000000	400.000000	48.000000
max	45210.000000	46211.000000	4918.000000	63.000000	871.000000	275.000000	499.000000	121.000000

	Insurance	balance	housing	loan	contact	poutcome	Term Deposit	job	marital	education	Annual Income	Gender
count	45211	45156	45211	45203	45168	45196	45203	45198	45193	45190	45194	45211
unique	2	6268	2	2	6	7	2	13	3	6	44972	2
top	no	0	yes	no	cellular	unknown	no	blue-collar	married	secondary	1380371	M
freq	44396	3514	25130	37959	29282	36884	39914	9623	27202	23187	3	27168

- Mean Age is aproximately 41 years old. (Minimum: 18 years old and Maximum: 95 years old.)
- Age has mean and median almost equal to 40, it shows that the aga data is normally distributed

- Averege bank balance is 1,362
- Standard Deviation (std) is a high number so we can understand through this that the balance is heavily distributed across the dataset.
- Mean & Median value of the balance attribute has lot of difference which means you will find high level of data skewness and outlier in its distribution

  - **isnull ().sum()**: Give the count of null values in each column of the dataframe

```
Sno                0
Customer_number    0
Insurance          0
balance           55
housing            0
loan               8
contact           43
duration          39
campaign           0
last_contact_day   0
previous           0
poutcome          15
Term Deposit       8
Count_Txn          1
age                0
job               13
marital           18
education         21
Annual Income     17
Gender             0
dtype: int64
```

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Impuatation)

To replace null values, it's essential to understand the distribution of the data.

- **Mean/Median Imputation:** Replace null values (numeric data) with the mean (for normal distribution) or median (for skewed distribution/extreme points) of the column.
- **Mode Imputation:** Replace null values(categorical data) with the mode (most frequent value) of the column.

Test to identify the distribution and presence of outliers for mean/mode Imputation.

- **Shapiro-Wilk test:** Check the normality of a sample distribution
- **QQ Plot:** Compares the quantiles of the observed data against the quantiles expected from a theoretical distribution (normal distribution).
- **Histogram Plot:** Check type of distribution (skewness).
- **Boxplot:** Helps to identify the outliers.

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Impuatation)

To replace null values, it's essential to understand the distribution of the data.

- **Mean/Median Imputation:** Replace null values (numeric data) with the mean (for normal distribution) or median (for skewed distribution/extreme points) of the column.
- **Mode Imputation:** Replace null values(categorical data) with the mode (most frequent value) of the column.

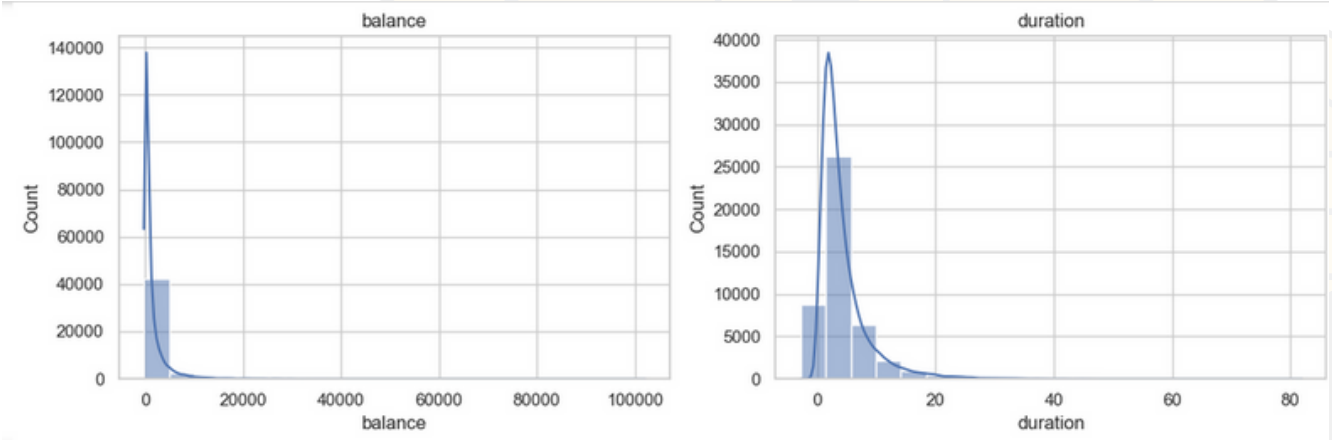
Test to identify the distribution and presence of outliers for mean/mode Imputation.

- **Shapiro-Wilk test:** Check the normality of a sample distribution
- **QQ Plot:** Compares the quantiles of the observed data against the quantiles expected from a theoretical distribution (normal distribution).
- **Histogram Plot:** Check type of distribution (skewness).
- **Boxplot:** Helps to identify the outliers.

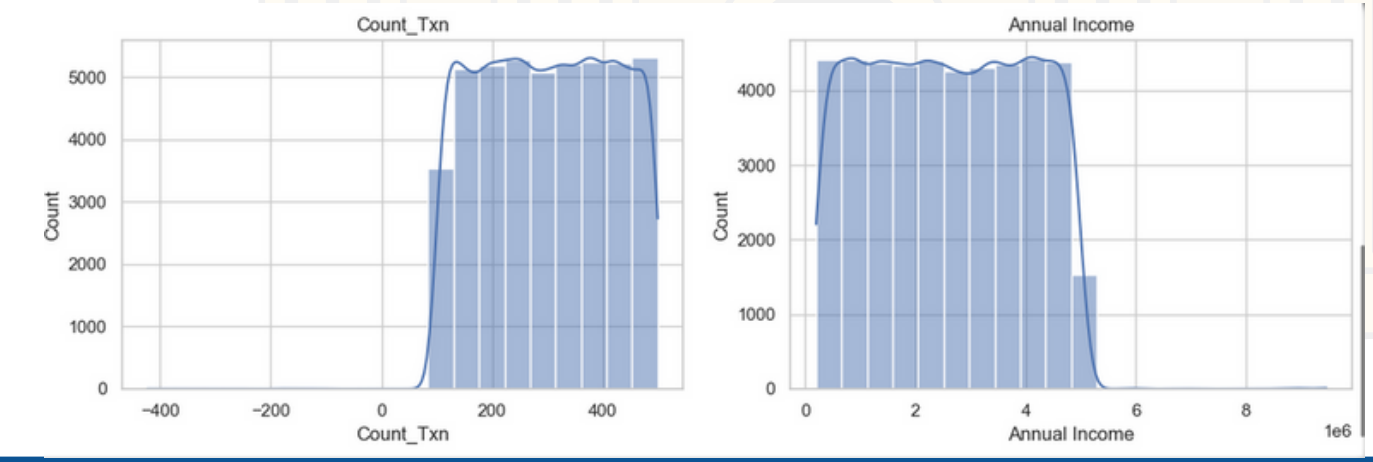
2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation) ['balance', 'duration', 'Count_Txn', 'Annual Income']

Numerical Attributes:

1. **Histogram Plot**



Positively Skewed Distribution



Negatively Skewed Distribution

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Impuatation)

Numerical Attributes:

1. Shapiro Wilk test for Normality

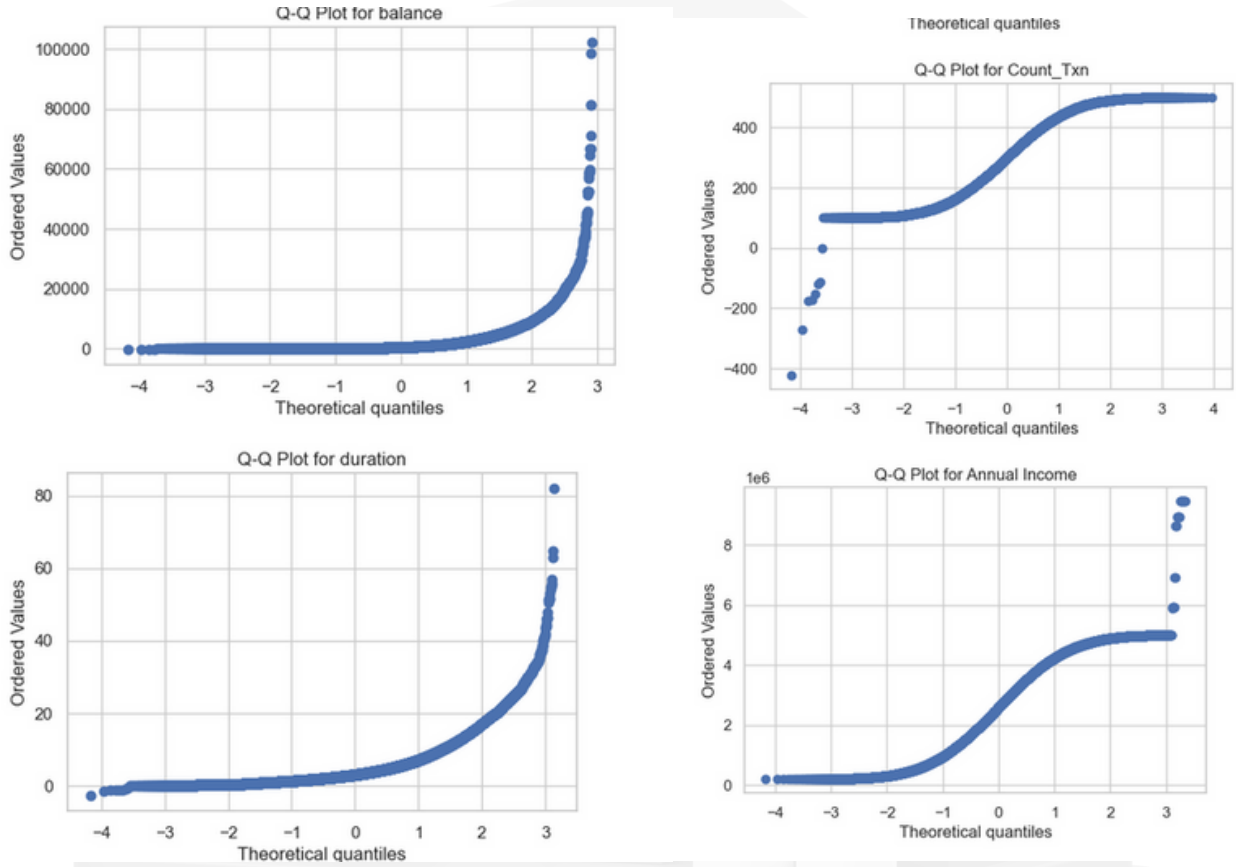
```
Shapiro-Wilk test for balance:
Statistic: nan, p-value: 1.0
The data for balance looks normally distributed (fail to reject the null hypothesis)

Shapiro-Wilk test for duration:
Statistic: nan, p-value: 1.0
The data for duration looks normally distributed (fail to reject the null hypothesis)

Shapiro-Wilk test for Count_Txn:
Statistic: nan, p-value: 1.0
The data for Count_Txn looks normally distributed (fail to reject the null hypothesis)

Shapiro-Wilk test for Annual Income:
Statistic: nan, p-value: 1.0
The data for Annual Income looks normally distributed (fail to reject the null hypothesis)
```

Shapiro wilks test is not valid because of the presence of outliers.The test is sensitive to deviations from normality, and in the presence of outliers. Thus, proceed with visuals plots like QQ plot and histograma and box plot.

2.4 Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Impuatation)

Numerical Attributes:

3. QQ plot

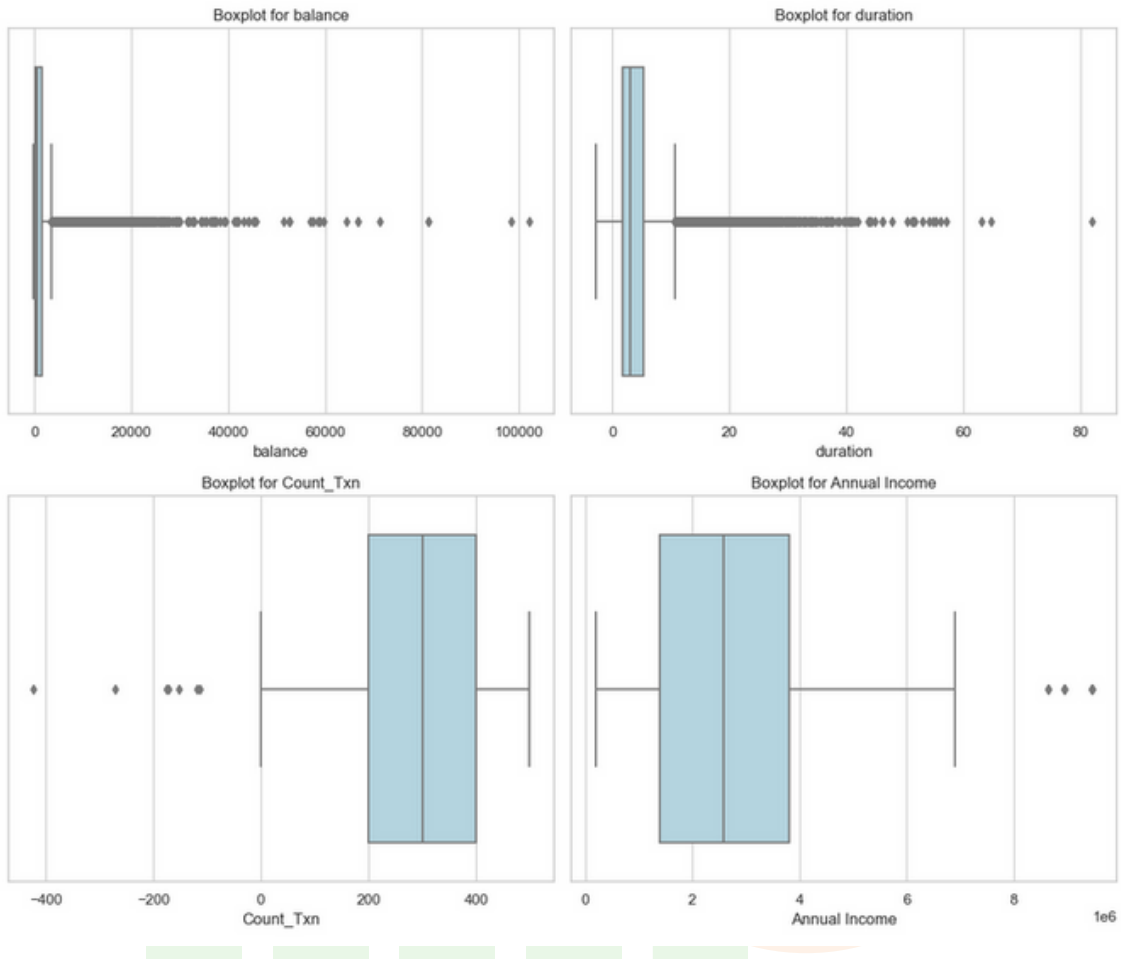


The upward curve and scattered points in the tail suggest that data has a non-normal distribution, with skewness and heavier tails than the normal distribution.

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Impuatation)

Numerical Attributes:

3. Box Plot



Presence of Outlier in 'balance' and 'duration' variable are high

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Impuatation)

Balance

- Balance data are not normally distributed and are highly positively skewed (right tail distribution) and have lots of outliers (boxplot).
- The extreme outliers are between 60k to 100k euro.
- The distribution of balance has a huge standard deviation relative to the mean, thus large variabilities in customer balance levels (which need to be look deeper upon).
- Outliers need to be treated.

Duration

- Converted the column from seconds to minutes for better clarity and insights.
- Duration data are not normally distributed and is positively skewed and have lots of outlier (boxplot)
- The right skewed boxplot indicate that most calls are relatively short.
- There is a large no of outliers ranging from 10 mins to 40 mins which also need to look furthur upon.

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Impuatation)

Count_Txn

- Even though only one value is missing, it is better not to ignore it because the term deposit corresponding to his value is 'yes', which is very crucial for us (very few no of values)
- Count_Txn is not normally distributed and is negatively skewed (left tail distribution) and have few outliers.
- The plot indicate that most of the customers are highly engaged (interacting) with the account.

Annual Income

- Annual Income is not normally distributed and is negatively skewed (left tail distribution) and have few outliers.
- This plot indicate that annual income of the customers range from 1 to 4 million.
- Thre are 3 customers whose income fall above this range.

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation)

Median Imputation

Conclusion: There seems to be a high level of skewed distribution with extreme outliers; the median could be a more robust measure of central tendency because it is less sensitive to extreme values.

```
Median value for balance: 486.5
Median value for duration: 3.0
Median value for Count_Txn: 300.0
Median value for Annual Income: 2586789.0
```

Sno	0
Customer_number	0
Insurance	0
balance	0
housing	0
loan	8
contact	43
duration	0
campaign	0
last_contact_day	0
previous	0
poutcome	15
Term Deposit	8
Count_Txn	0
age	0
job	13
marital	18
education	21
Annual Income	0
Gender	0

dtype: int64

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation)

Categorical Attributes- ['contact', 'poutcome', 'job', 'marital', 'education']

```
Unique values and counts for contact:
cellular     29282
unknown      12970
telephone     2850
Mobile          29
Tel             20
?               17
Name: contact, dtype: int64

Unique values and counts for poutcome:
unknown    36884
failure     4901
other       1840
success     1511
pending       55
?              4
????           1
Name: poutcome, dtype: int64

Unique values and counts for job:
blue-collar      9623
management       9455
technician       7595
admin.           5171
services         4153
retired          2264
self-employed    1579
entrepreneur     1486
unemployed       1302
housemaid        1240
student           938
unknown           288
blue collar       104
Name: job, dtype: int64

Unique values and counts for marital:
married     27202
single      12787
divorced     5204
Name: marital, dtype: int64

Unique values and counts for education:
secondary    23187
tertiary     13296
primary       6845
unknown       1857
Primary          3
ter tiary        2
Name: education, dtype: int64
```

```python
# Replace specified values with 'tertiary' in the 'education' column
merged_df['education'].replace(['ter tiary'], 'tertiary', inplace=True)
merged_df['education'].replace(['Primary'], 'primary', inplace=True)
merged_df['contact'].replace(['Mobile'], 'cellular', inplace=True)
merged_df['contact'].replace(['Tel'], 'telephone', inplace=True)
merged_df['job'].replace(['blue collar'], 'blue-collar', inplace=True)
```

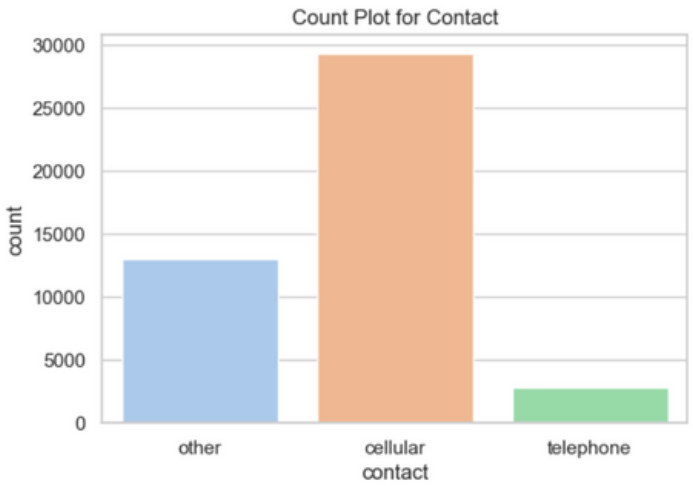
Why Not Mode?

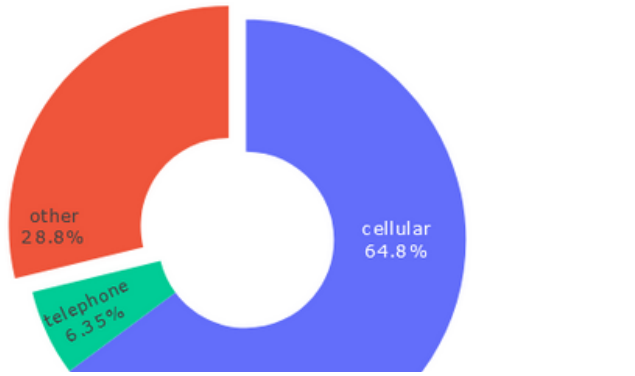
Maintains transparancy and prevents bias. Thus, labelled as ‘Unknown’ instead of mode replacement

```python
# Replace blank values with 'unknown' in each selected column
for column in selected_columns:
    merged_df[column].replace('', 'unknown', inplace=True)
    # Uncomment the line below if you also want to handle NaN values
    merged_df[column].fillna('unknown', inplace=True)
```

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation)

Categorical Attributes- ['contact', 'poutcome', 'job', 'marital', 'education']



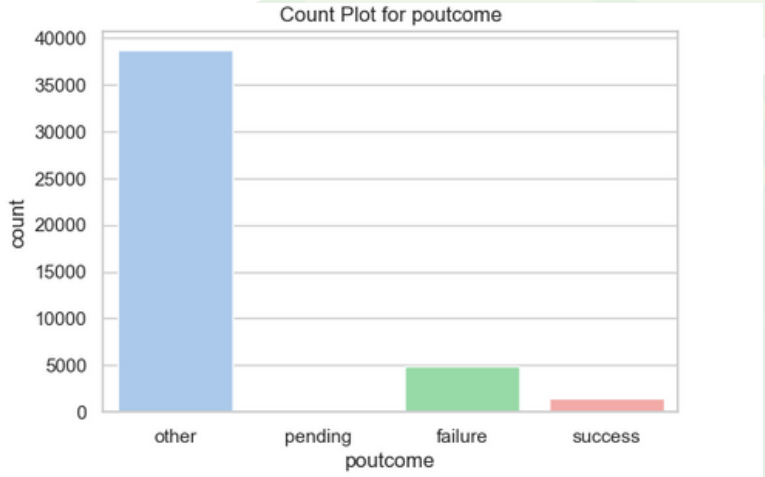


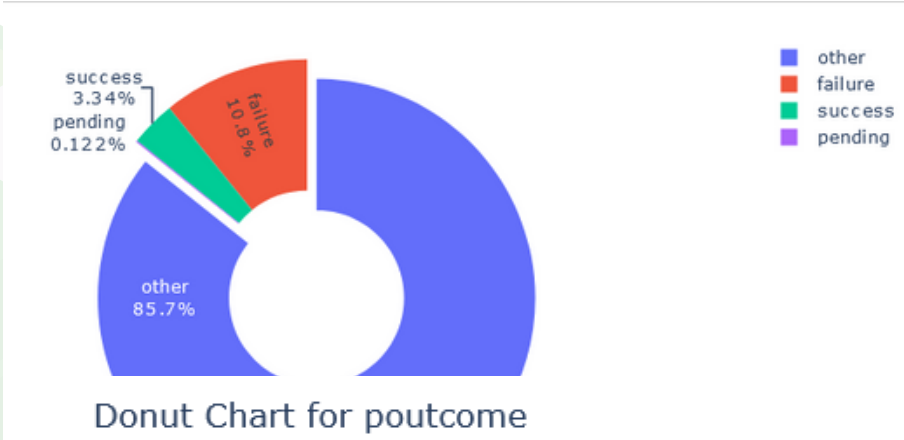
Donut Chart for contact

- Around 64% of mode of client communication was using mobile phone. Understanding the call duration along with this provide a significant impact on conversaion rate.(ie, higher conversion rate generally means a higher percentage of successful outcomes, such as turning potential customers into actual customers.)
- landline usage exhibits a notably low share as a means of client communication. This observation underscores the dominance of mobile communication, surpassing all other modes.
- Around 28% of people's mode of communication is not captured here thus, we cannot infer from this.We need to resample the data to gain insights from this.

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation)

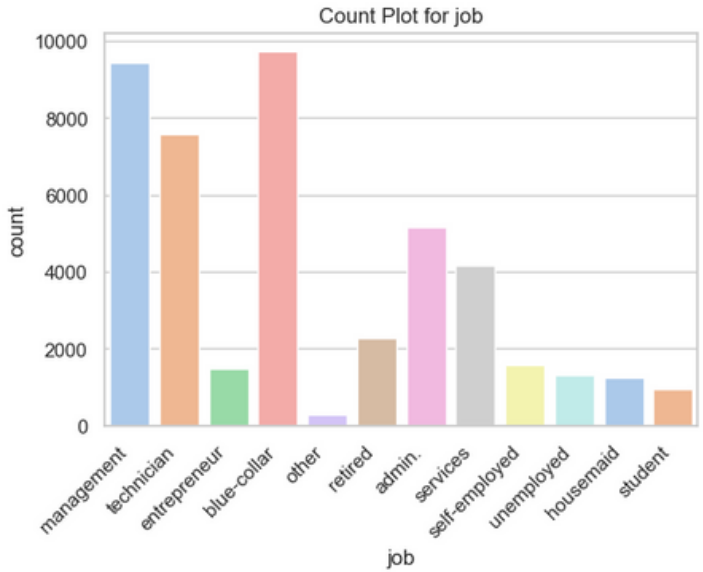
Categorical Attributes- ['contact', 'poutcome', 'job', 'marital', 'education']

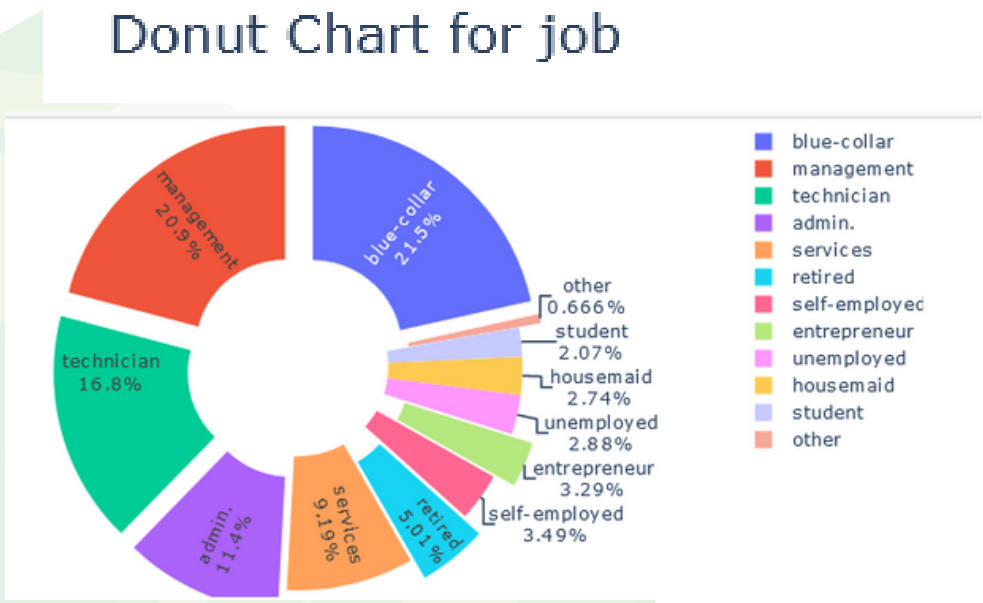




2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation)

Categorical Attributes- ['contact', 'poutcome', 'job', 'marital', 'education']

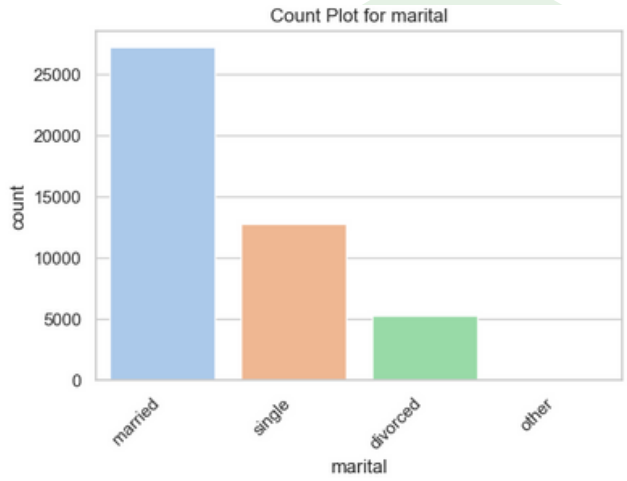


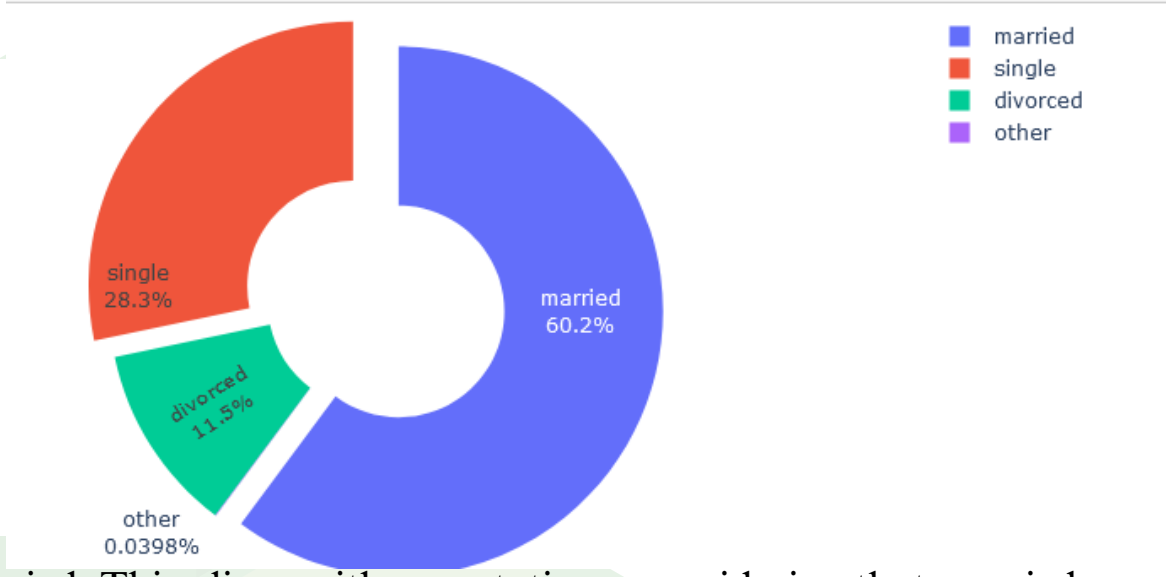


- The bank predominantly targeted individuals in blue-collar, management, and technician professions.
- Customers classified as unemployed, housemaids, and students. This decision, though these groups represent a minority, may not be optimal, as they are less likely to convert into fixed deposit customers.
-

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation)

Categorical Attributes- ['contact', 'poutcome', 'job', 'marital', 'education']

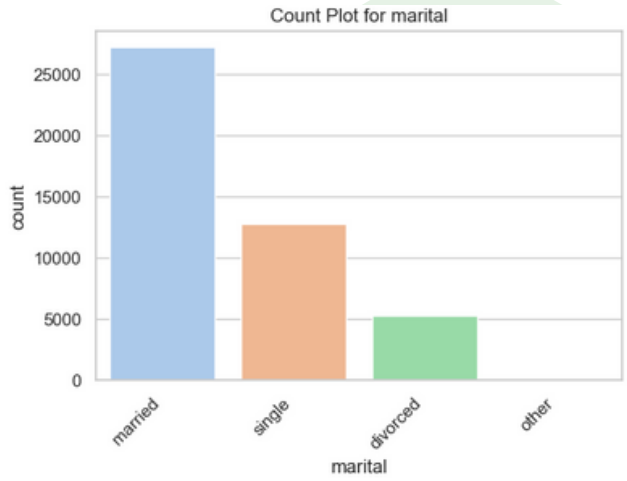


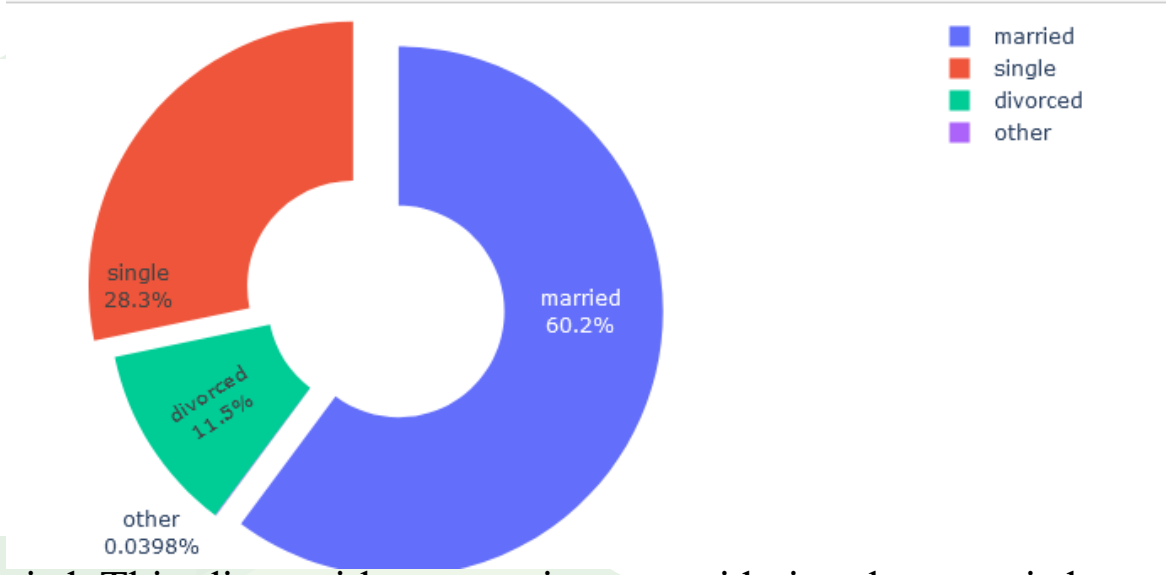


- Approximately half of the customers, are married. This aligns with expectations, considering that married individuals often prioritize savings.
- About 28% of the customers are classified as single. Anticipating a potentially higher conversion rate from this group, especially among young, single working professionals.
- Accounting for 11%, there are customers labeled as divorced. It is not anticipated that this group will exhibit a high likelihood of converting to fixed deposit customers.

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation)

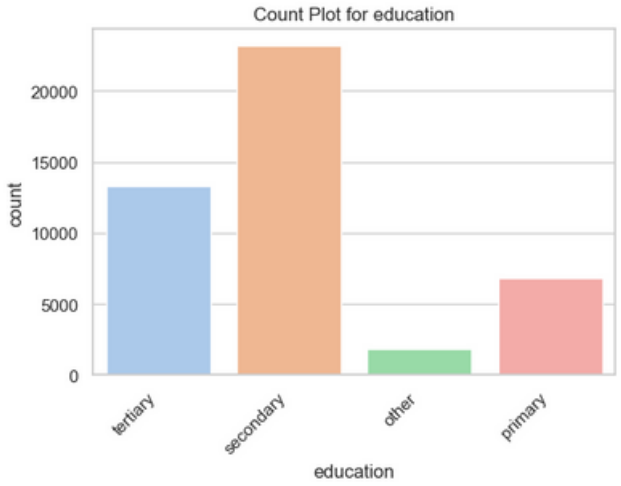
Categorical Attributes- ['contact', 'poutcome', 'job', 'marital', 'education']

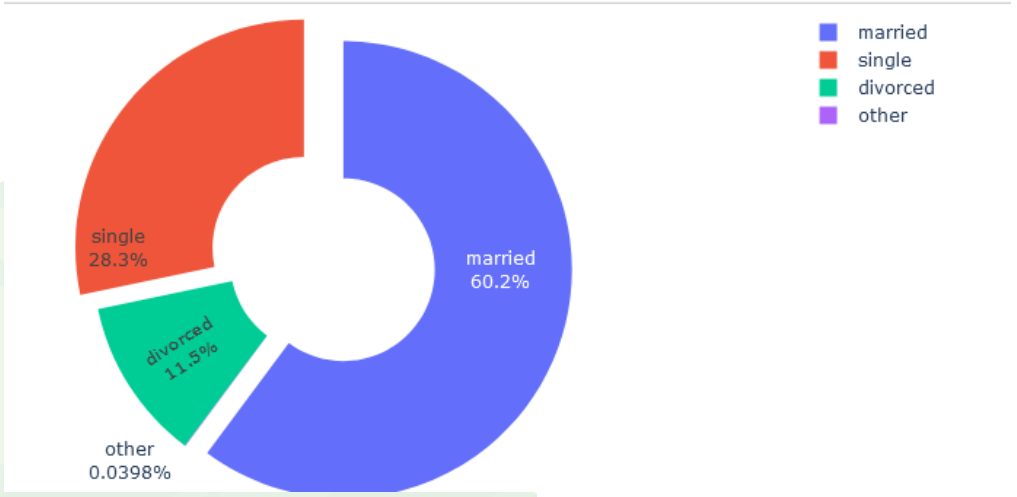




- Approximately half of the customers, are married. This aligns with expectations, considering that married individuals often prioritize savings.
- About 28% of the customers are classified as single. Anticipating a potentially higher conversion rate from this group, especially among young, single working professionals.
- Accounting for 11%, there are customers labeled as divorced. It is not anticipated that this group will exhibit a high likelihood of converting to fixed deposit customers.

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation)





- With a higher education level, individuals may have a better understanding of various investment options, including fixed deposits.
- There are chances that that a significant portion of tertiary-educated customers might consider fixed deposits as part of their investment portfolio. So our focus should be on this category. we need to a bivariate analysis to furthur validate this point.
- if the secondary group prioritize stability and security in their investments chance can be move from moderate to high in the fixed deposit investment. For that campaign should be focused on this section.
- customers are still studying and have completed only primary education should not be our target audience. They are very less likely to take fd option.

2.4. Replace Missing Value with appropriate Statistical Values(Mean/Median/Mode Imputation)

Conclusion:

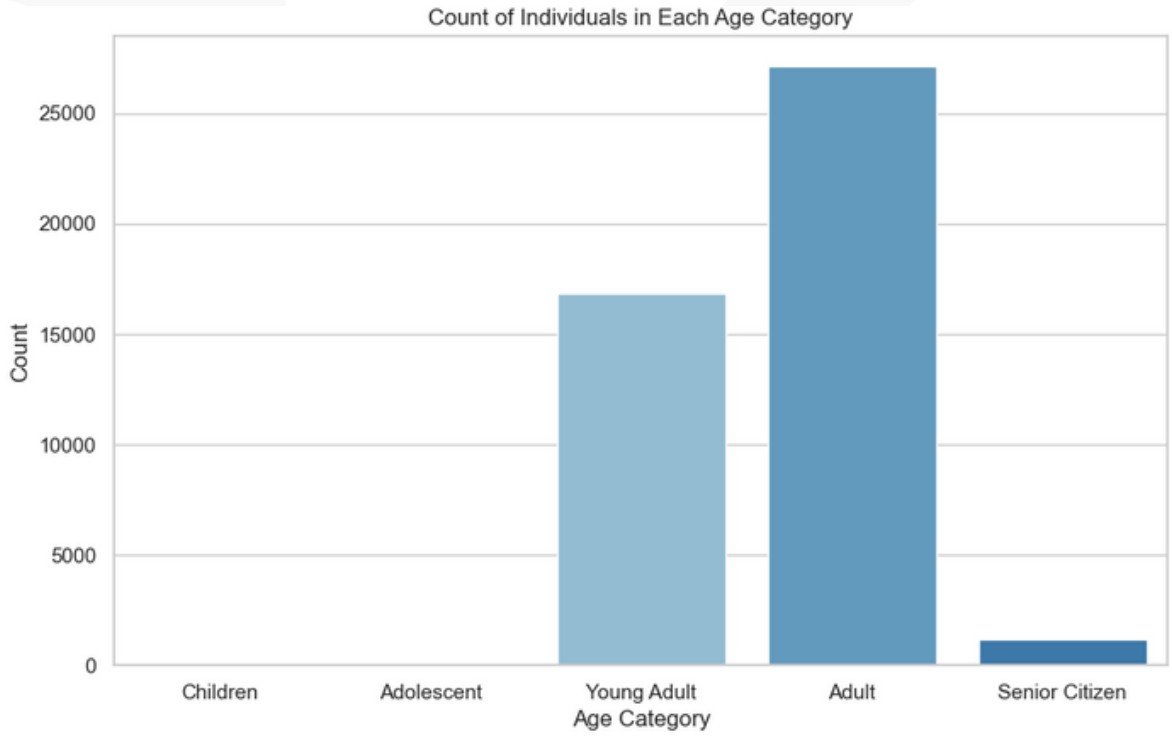
```
Percentage of "unknown" values in contact: 28.82%
Percentage of "unknown" values in poutcome: 85.70%
Percentage of "unknown" values in job: 0.67%
Percentage of "unknown" values in marital: 0.04%
Percentage of "unknown" values in education: 4.15%
```

After replacement of missing value we can see that for outcome more half of the per of data is unknown. Thus, It can be ignored. (Validated using feature selection).

28 per of contact variable is also unknown. This variable doesn't provide any significance for the Target variable theoretically also.

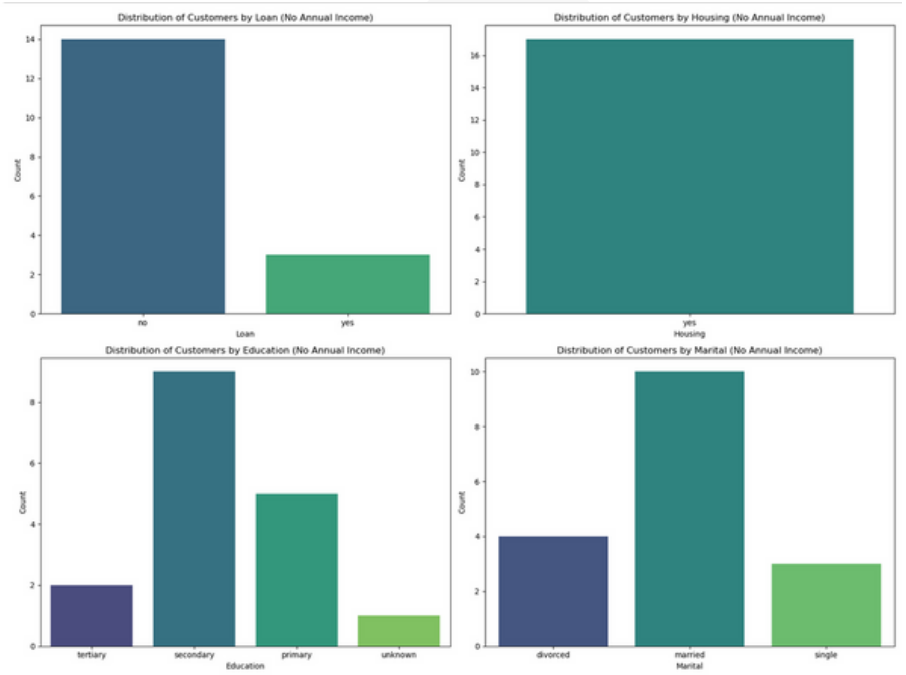
3. Exploratory Analysis

'Children': (0, 12),
'Adolescent': (13, 19),
'Young Adult': (20, 35),
'Adult': (36, 60),
'Senior Citizen': (61, 120)



3. Exploratory Analysis

1. Income Insights:
• How many customers have no annual income? Plot and present the data distribution of these customers.

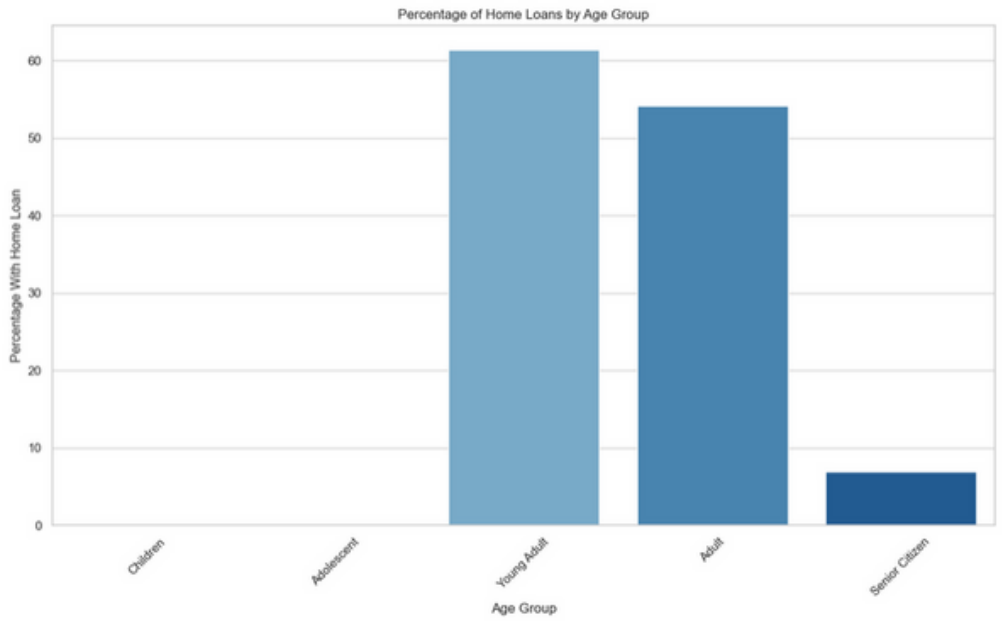


Total of 17 customers has no annual income

3. Exploratory Analysis

- **Age and Home Loans:**

•Determine which age group has the highest percentage of home loans. Present this data visually and discuss possible reasons.

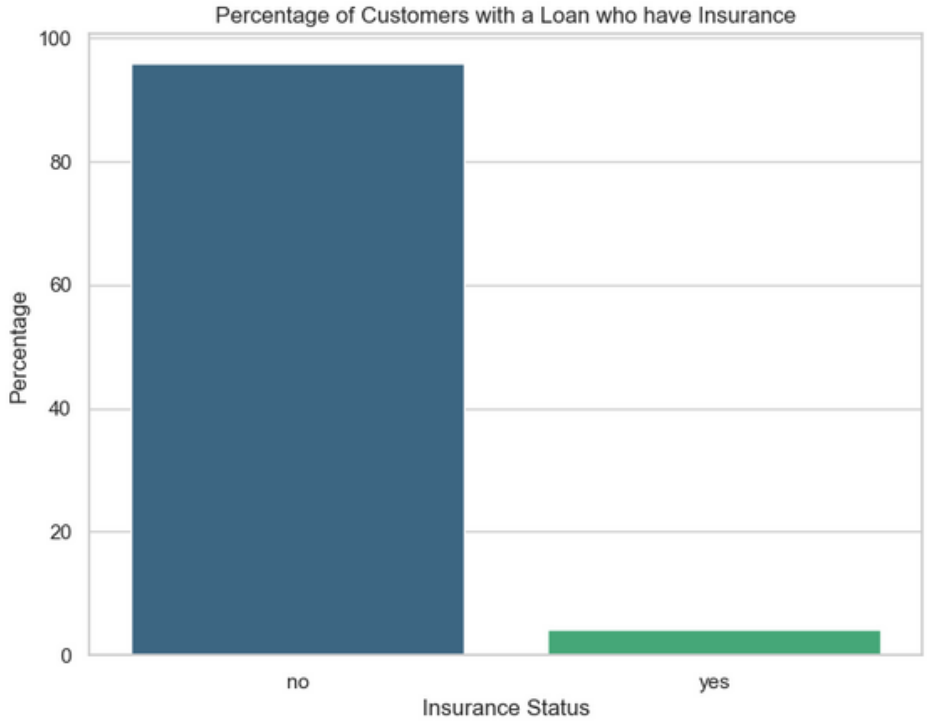


- People in the 'Young Adult' and 'Adult' age groups may be at stages in their lives where they are more likely to make significant financial decisions, such as purchasing a home.
- Young adults might be entering the housing market for the first time, and adults may be upgrading to larger homes for family needs.
- **Stability and Income**. Increased financial stability can make it easier to qualify for a mortgage and handle the financial responsibilities of homeownership.
- **Family Considerations:** The desire to provide a stable home environment for a growing family could be a motivating factor for individuals in these age groups to purchase a home.
- **Investment and Wealth Building**: Homeownership is often viewed as a long-term investment and a means of building wealth/

3. Exploratory Analysis

****3. Loan and Insurance Analysis:**

• Calculate the percentage of customers with a loan who have taken out insurance. Visualize this data and discuss potential implications.




```
Percentage of customers with a loan who have insurance:
no     95.844837
yes     4.155163
Name: Insurance, dtype: float64
```

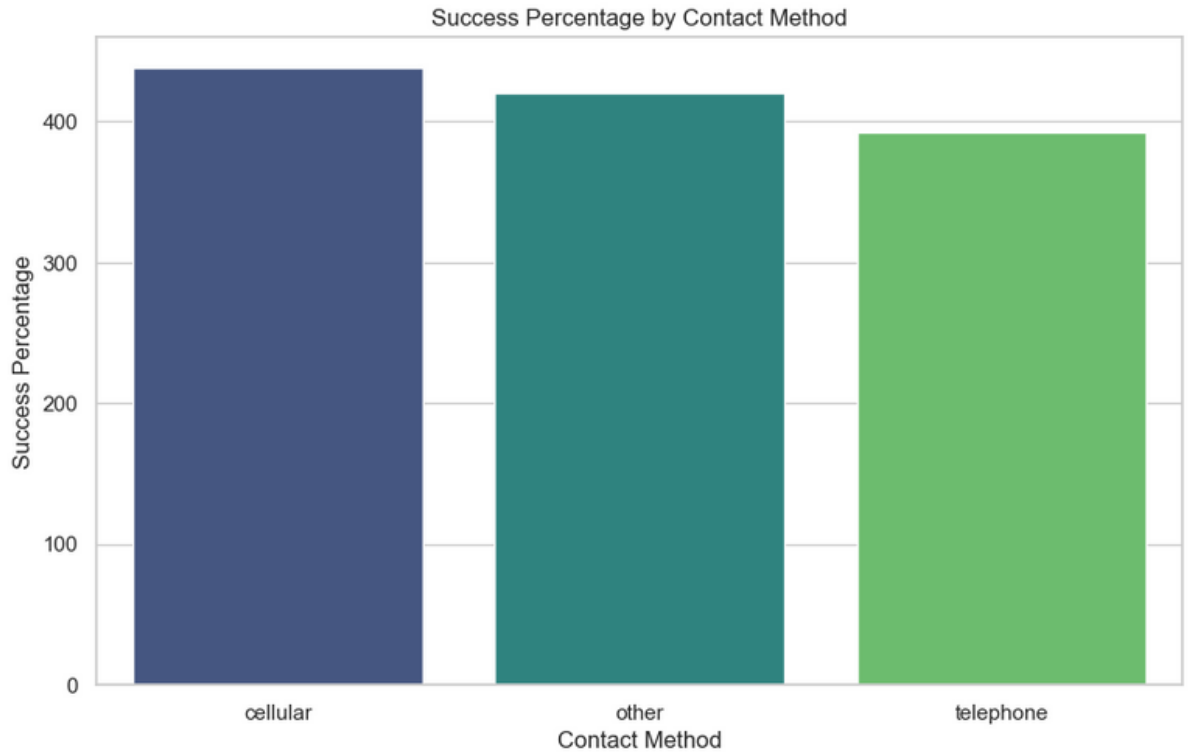
- Customers with loans might be more financially constrained, leading them to prioritize spending on essential items rather than optional ones like insurance.

- The way insurance products are marketed or communicated to customers with loans could impact their decision. If there's a lack of effective communication or targeted marketing, customers may not see the value in obtaining insuranc

3. Exploratory Analysis

. Communication Strategy Insights:

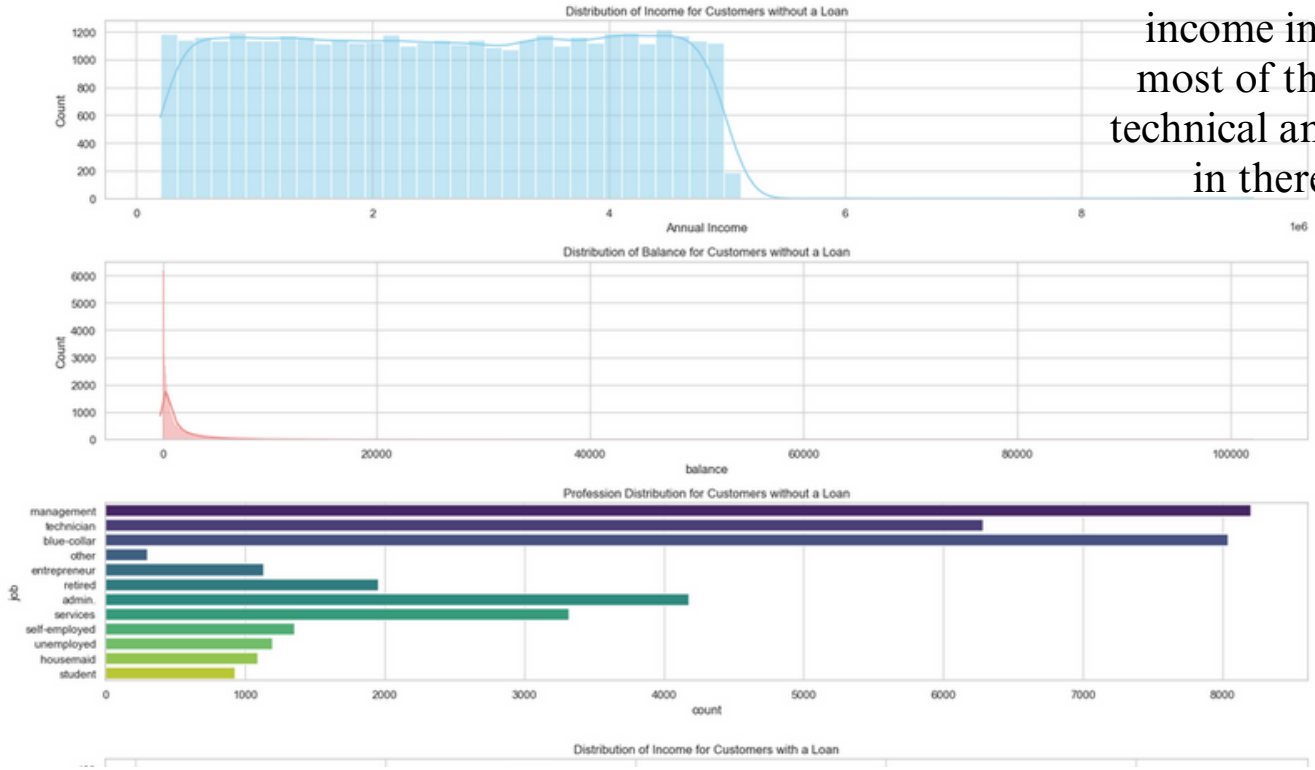
• Analyse and summarize the best Contact method (with the highest success percentage) to contact people to ascertain the status of term deposit subscriptions.



3. Exploratory Analysis

2. Loan-less Customers Profile:

• Filter out customers who don't have any type of loan. Plot the distribution of their Income, balance, and profession. How do these metrics differ from those with loans?

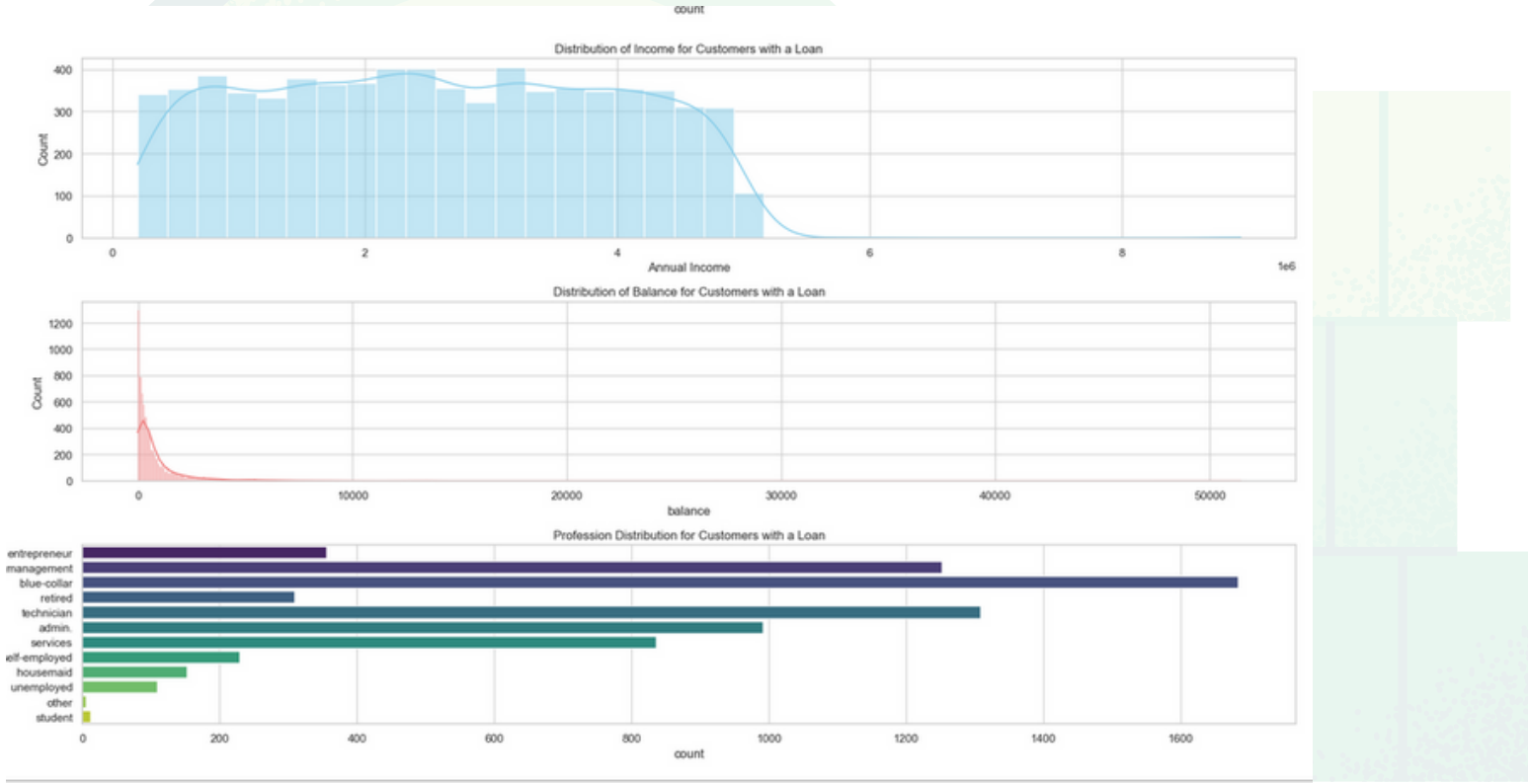


- Customers who donhave any loan have income in the range 1 to 5 million and most of them belong to blue collar job/ technical and self employeed. The balance in there account is also very low.

3. Exploratory Analysis

2. Loan-less Customers Profile:

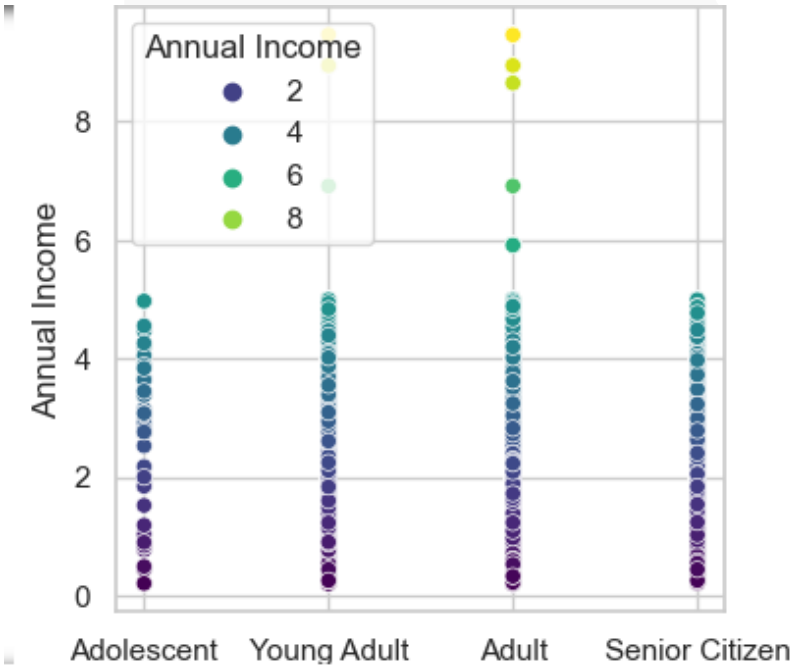
• Filter out customers who don't have any type of loan. Plot the distribution of their Income, balance, and profession. How do these metrics differ from those with loans?

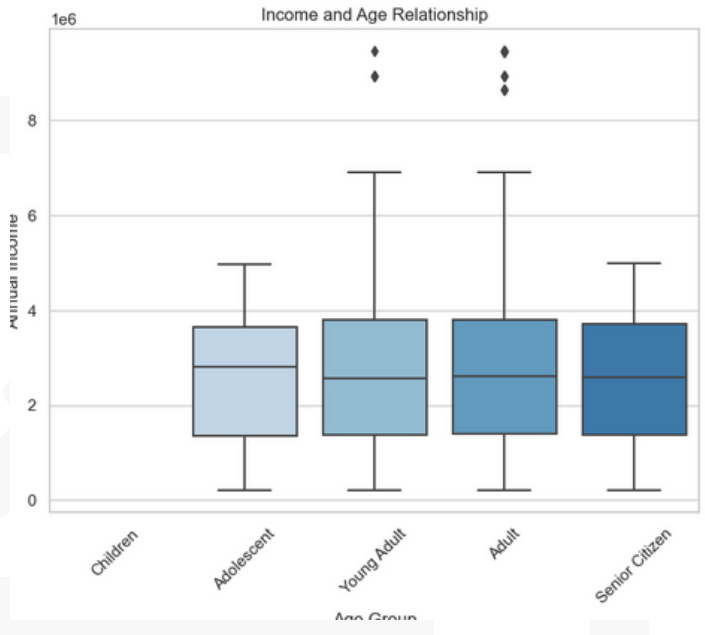


3. Exploratory Analysis

6. Income and Age Relationship:

Investigate any relationships between annual income and age group. Use appropriate plots and statistics to present the findings.





4. Encoding the Categorical Variable

The machine doesn't understand the categorical value, so we encode the categorical variables depending on the nature of the data.

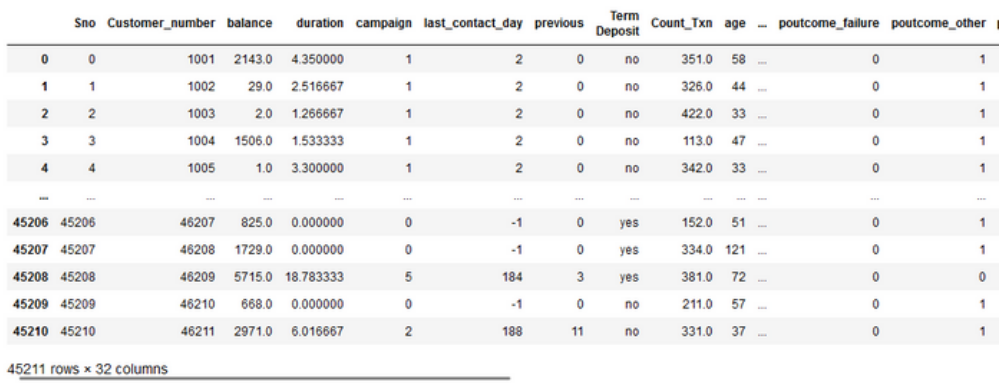
1. **One hot Encoding**
- **Columns= ['Insurance', 'housing', 'loan', 'contact', 'poutcome', 'marital', 'Gender'] performed one hot encoding**
- **Reason: Nominal data(no order)**

ɔn	campaign	last_contact_day	previous	Term Deposit	Count_Txn	age	...	poutcome_failure	poutcome_other	poutcome_pending	poutcome_success	marital_divorce
)0	1	2	0	no	351.0	58	...	0	1	0	0	
37	1	2	0	no	326.0	44	...	0	1	0	0	
37	1	2	0	no	422.0	33	...	0	1	0	0	
33	1	2	0	no	113.0	47	...	0	1	0	0	
)0	1	2	0	no	342.0	33	...	0	1	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	
)0	0	-1	0	yes	152.0	51	...	0	1	0	0	
)0	0	-1	0	yes	334.0	121	...	0	1	0	0	
33	5	184	3	yes	381.0	72	...	0	0	0	1	
)0	0	-1	0	no	211.0	57	...	0	1	0	0	
57	2	188	11	no	331.0	37	...	0	1	0	0	

4. Encoding the Categorical Variable

2. Label Encoding

- Columns= ['education']
- Reason: Ordinal data(order matter, Teritary>Secondary> Primary)



	Sno	Customer_number	balance	duration	campaign	last_contact_day	previous	Term Deposit	Count_Txn	age	...	poutcome_failure	poutcome_other
0	0	1001	2143.0	4.350000	1	2	0	no	351.0	58	...	0	1
1	1	1002	29.0	2.516667	1	2	0	no	326.0	44	...	0	1
2	2	1003	2.0	1.266667	1	2	0	no	422.0	33	...	0	1
3	3	1004	1506.0	1.533333	1	2	0	no	113.0	47	...	0	1
4	4	1005	1.0	3.300000	1	2	0	no	342.0	33	...	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...
45206	45206	46207	825.0	0.000000	0	-1	0	yes	152.0	51	...	0	1
45207	45207	46208	1729.0	0.000000	0	-1	0	yes	334.0	121	...	0	1
45208	45208	46209	5715.0	18.783333	5	184	3	yes	381.0	72	...	0	0
45209	45209	46210	668.0	0.000000	0	-1	0	no	211.0	57	...	0	1
45210	45210	46211	2971.0	6.016667	2	188	11	no	331.0	37	...	0	1

45211 rows × 32 columns

3. Frequency Encoding

- Columns= ['job']
- Reason: High number of unique categories (high cardinality), one-hot encoding can lead to a high-dimensional and sparse feature space.

```
df_encoded['job']

0         1
1         2
2         7
3         0
4        11
        ..
45206     2
45207     5
45208     5
45209     0
45210     7
Name: job, Length: 45211, dtype: int64
```

5. Feature Scaling

First, we check the performance of the model without outlier diagnosis. This is because in the real world problem outliers could be a potential data point.

- **Standard Scaler()-** Features will be rescaled so that they'll have the properties of a standard normal distribution with **μ=0 and σ=1**. They are centered around 0 with a standard deviation of 1 is not only important if we are comparing measurements that have different units, but it is also a general requirement for many machine learning algorithms.

	balance	duration	campaign	last_contact_day	previous	Term Deposit	Count_Txn	age	job	education	...	poutcome_failure	poutcome_other
0	0.240839	0.011362	-0.570769	-0.400545	0	0.0	0.444332	1.570241	1	3	...	0	1
1	-0.459038	-0.415770	-0.570769	-0.400545	0	0.0	0.228273	0.278572	2	2	...	0	1
2	-0.467976	-0.706997	-0.570769	-0.400545	0	0.0	1.057938	-0.736310	7	0	...	0	1
3	0.029949	-0.644869	-0.570769	-0.400545	0	0.0	-1.612545	0.555358	0	0	...	0	1
4	-0.468308	-0.233268	-0.570769	-0.400545	0	0.0	0.366551	-0.736310	11	0	...	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...
45206	-0.195508	-1.002106	-0.894689	-0.430707	0	1.0	-1.275494	0.924406	2	3	...	0	1
45207	0.103777	-1.002106	-0.894689	-0.430707	0	1.0	0.297412	7.382749	5	1	...	0	1
45208	1.423411	3.374056	0.724907	1.429277	3	1.0	0.703602	2.861909	5	2	...	0	0
45209	-0.247486	-1.002106	-0.894689	-0.430707	0	0.0	-0.765596	1.477979	0	2	...	0	1
45210	0.514962	0.399664	-0.246850	1.469493	11	0.0	0.271485	-0.367262	7	2	...	0	1

45203 rows × 30 columns

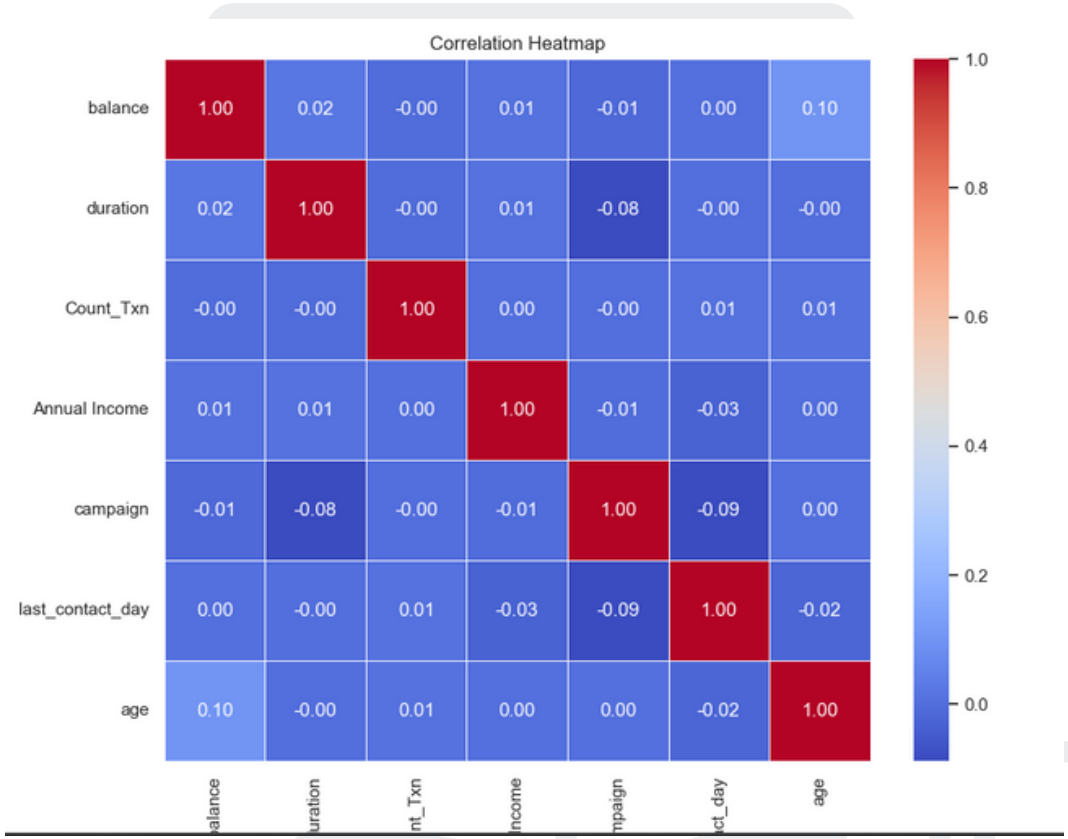
6. Split the Dependent and Independent Features

7. Split Train and Test Data

- **Split train and test in the ratio 80:20 (train:test)**

```
X_train shape: (36162, 29)
X_test shape: (9041, 29)
y_train shape: (36162,)
y_test shape: (9041,)
```

Heat Map identify the correlation



*****Observation:"campaign " has a strong correlation with "duration"and "last_day_contacts"*****

8. Define the Model

```
# Models to Fit & Evaluate
models =
    LogisticRegression(),#common for binary classification prob
    KNeighborsClassifier()#capture complex relationship
    GaussianNB()#independence feature, suitable for large
dataset, binary prob
    DecisionTreeClassifier(),#capture non linear relationship
    RandomForestClassifier(),#aids in feature selection
    AdaBoostClassifier(),#focus on misclassified instances
    XGBClassifier()#reduce overfitting

]
```

8. Model Evaluation

- Confusion Matrix
- Classification Report based on precision, recall , f1 and accuracy score

Because the target column is imbalanced, the accuracy score is not a valid score to evaluate the model

- ROC AUC Curve
- Pre Recall Curve

Train and Test Accuracy (check for overfitting)

```
Cross-Validation Accuracy: 0.9010839385057915
Training Accuracy Score: 0.9013605442176871

C:\Users\hp\anaconda3\lib\site-packages\sklearn\linear_model\_l

lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver o
    https://scikit-learn.org/stable/modules/linear_model.html#l

Testing Accuracy Score: 0.8962504147771264
```

```
Results for KNeighborsClassifier:

Cross-Validation Accuracy: 0.8893591082966589
Training Accuracy Score: 0.9164869199712405
Testing Accuracy Score: 0.8872912288463666
```

```
Results for DecisionTreeClassifier:

Cross-Validation Accuracy: 0.8587469290212386
Training Accuracy Score: 1.0
Testing Accuracy Score: 0.8576484902112598
```

```
Results for GaussianNB:

Cross-Validation Accuracy: 0.4910404271107474
Training Accuracy Score: 0.49043194513577787
Testing Accuracy Score: 0.4897688388153966
```

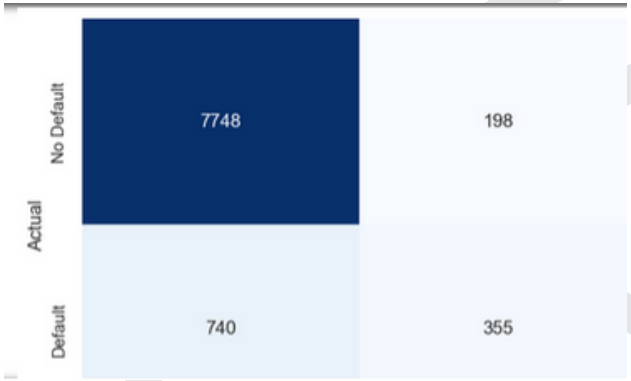
```
Results for AdaBoostClassifier:

Cross-Validation Accuracy: 0.8998395000666808
Training Accuracy Score: 0.9021348376749073
Testing Accuracy Score: 0.896029203097002
```

```
Results for RandomForestClassifier:

Cross-Validation Accuracy: 0.901609373336808
Training Accuracy Score: 1.0
Testing Accuracy Score: 0.8965822364782656
```

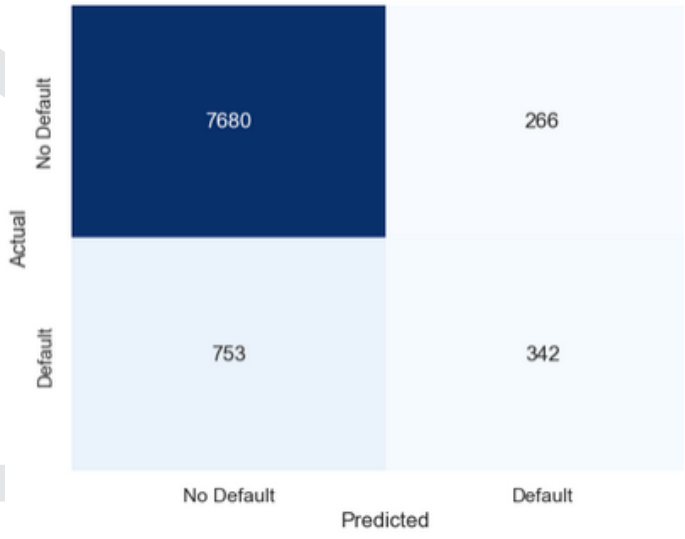
```
Results for XGBClassifier:

Cross-Validation Accuracy: 0.89809742313071
Training Accuracy Score: 0.9494496985786184
Testing Accuracy Score: 0.894038270102864 7
```

Confusion Matrix

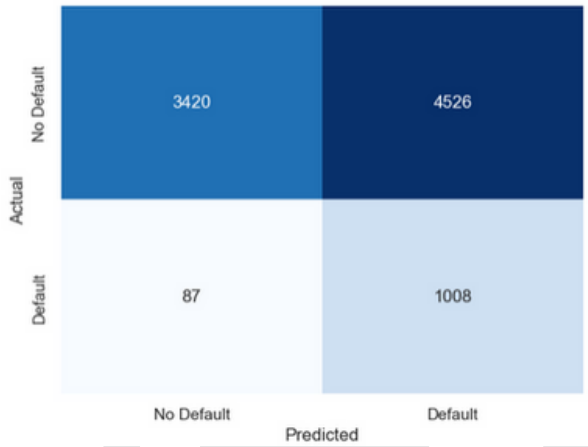
Logistic Regression



Knearest Neigbour

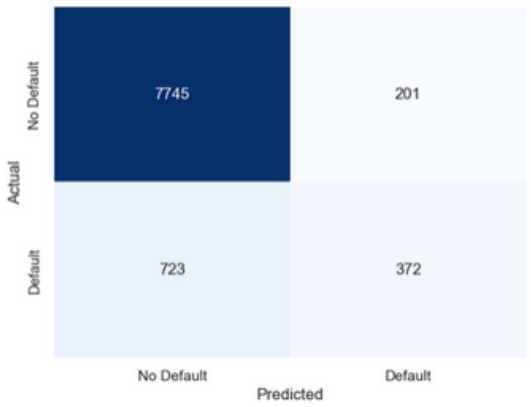


Guassian

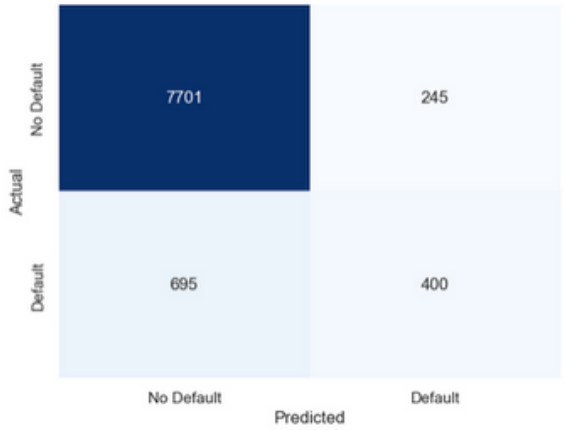


Decision Tree

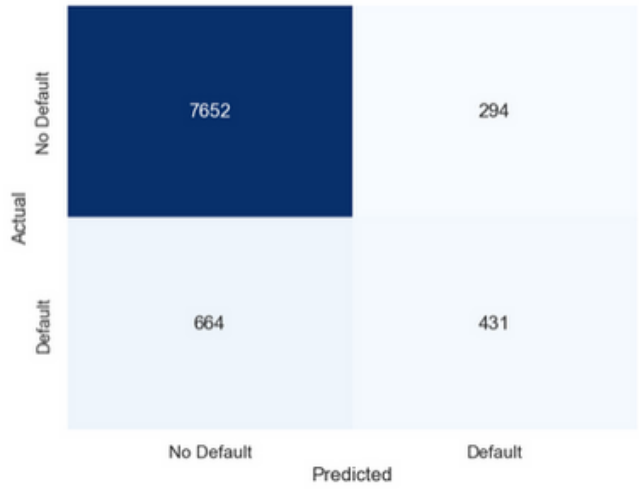
Confusion Matrix



Random Forest



Ada Boost



XGBoost

Classfication Report

Classification Report:

	precision	recall	f1-score	support
0.0	0.91	0.98	0.94	7946
1.0	0.64	0.32	0.43	1095
accuracy			0.90	9041
macro avg	0.78	0.65	0.69	9041
weighted avg	0.88	0.90	0.88	9041

Logistic Regression

	precision	recall	f1-score	support
0.0	0.91	0.97	0.94	7946
1.0	0.56	0.31	0.40	1095
accuracy			0.89	9041
macro avg	0.74	0.64	0.67	9041
eighted avg	0.87	0.89	0.87	9041

Knearest Neigbour

Classification Report:

	precision	recall	f1-score	support
0.0	0.98	0.43	0.60	7946
1.0	0.18	0.92	0.30	1095
accuracy			0.49	9041
macro avg	0.58	0.68	0.45	9041
weighted avg	0.88	0.49	0.56	9041

Guassian

Classification Report:

	precision	recall	f1-score	support
0.0	0.92	0.91	0.92	7946
1.0	0.41	0.44	0.43	1095
accuracy			0.86	9041
macro avg	0.67	0.68	0.67	9041
weighted avg	0.86	0.86	0.86	9041

Decision Tree

Classfication Report

- **Precision:** Ratio of true positives to the total items labeled as positive.
- **Recall:** Ratio of true positives to the total items actually belonging to the positive class.
- **Objective:** Prioritize high recall for interested customers, even at the cost of precision.
- **Reasoning:** Higher recall captures more interested customers but may misclassify some not-interested
- **Imbalanced Data:** The majority class is 'NO,' so we need to consider the F1 score.
- **F1 Score:** Harmonic mean of precision and recall.
- Need to consider f1 score which is a kind of trade off betweeb precision & recall being the harmonic mean of both
- **Trade-off:** F1 score balances precision and recall in an imbalanced dataset.

Classification Report:

	precision	recall	f1-score	support
0.0	0.92	0.97	0.94	7946
1.0	0.65	0.35	0.46	1095
accuracy			0.90	9041
macro avg	0.78	0.66	0.70	9041
weighted avg	0.88	0.90	0.88	9041

Random Forest

Classification Report:

	precision	recall	f1-score	support
0.0	0.92	0.97	0.94	7946
1.0	0.62	0.37	0.46	1095
accuracy			0.90	9041
macro avg	0.77	0.67	0.70	9041
weighted avg	0.88	0.90	0.88	9041

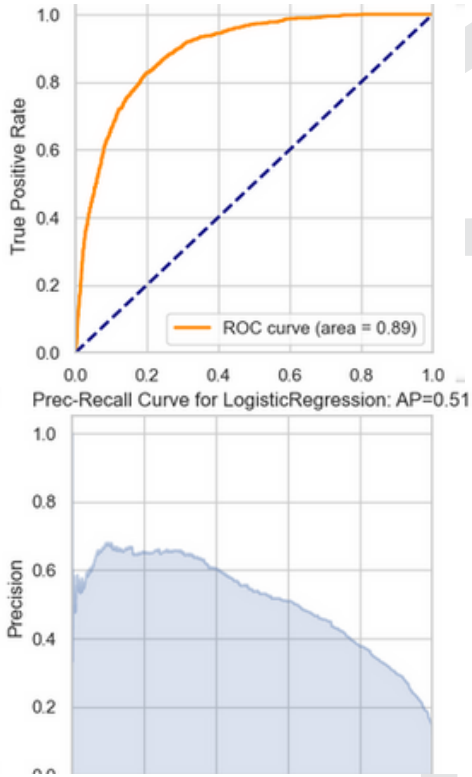
Ada Boost

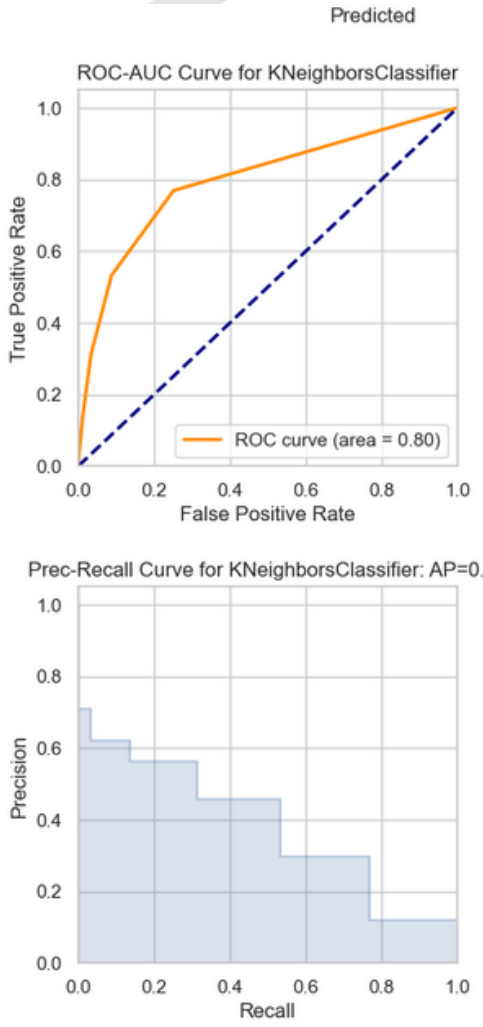
Classification Report:

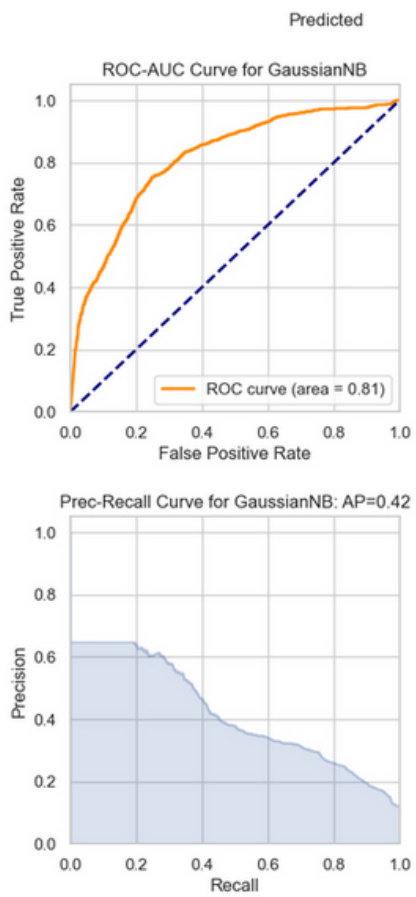
	precision	recall	f1-score	support
0.0	0.92	0.96	0.94	7946
1.0	0.59	0.39	0.47	1095
accuracy			0.89	9041
macro avg	0.76	0.68	0.71	9041
weighted avg	0.88	0.89	0.88	9041

XGBoost

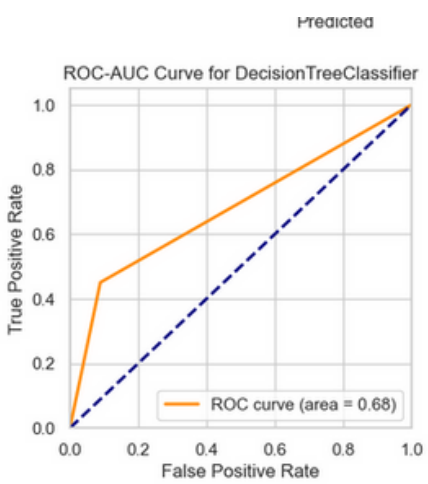
ROC-AUC and Pre-Recall Curve

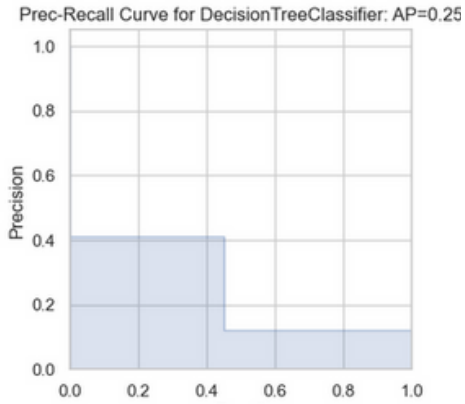


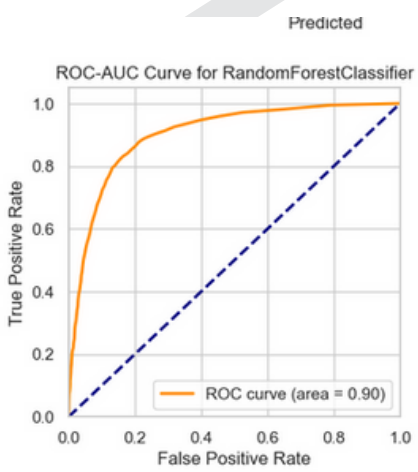


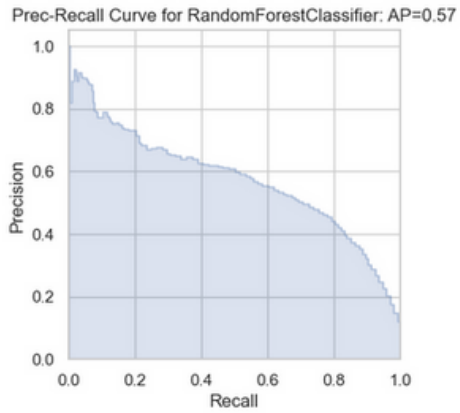


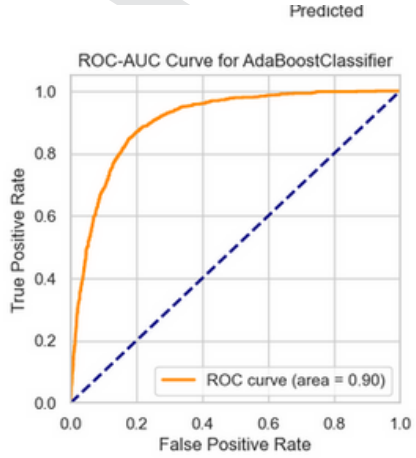
ROC-AUC and Pre-Recall Curve

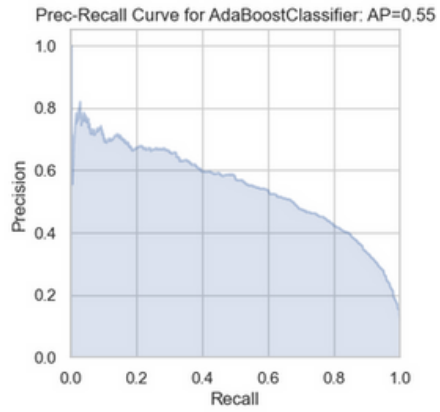


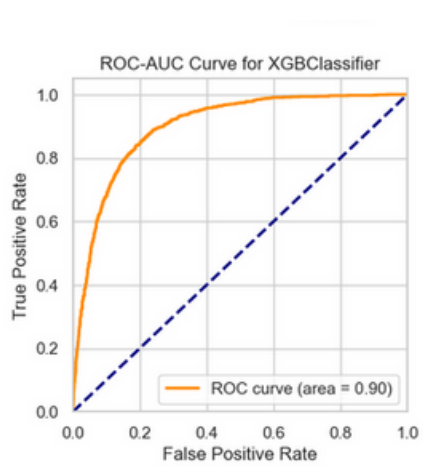


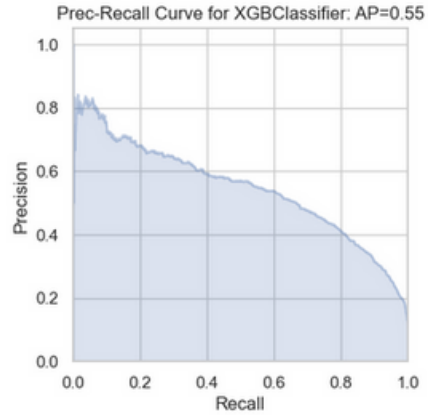












Observations

- Closely observing the roc curve, decision tree, logistic, Random Forest, NB, XG, and Adaboost all faired well with a score of 90 % while KNN scored 80 %, but the roc curve can be misleading as data is highly imbalanced.

- Let's see the precision-recall curve, which is more balanced in its approach for this kind of data sets. It will give more clear picture of auc score and average precision score along with f1 score.

- Here we can see decision tree has fared well compared to other models in terms of f1 score and auc score trade off.

- Decision Tree and Random Forest are Overfitting also

9. SMOTE- Balance the Target Data

- SMOTE is specifically designed to tackle imbalanced datasets by generating synthetic samples for the minority class.
- Upsample Technique

```
Original class distribution:
0.0    31968
1.0     4194
Name: Term Deposit, dtype: int64

Class distribution after SMOTE:
0.0    31968
1.0    31968
Name: Term Deposit, dtype: int64
```

10. Outlier Diagnosis

- Reduce Overfitting
- Reduce Noises

```
Skewness Values:
balance              8.524954
duration             3.145427
Count_Txn           -0.015093
Annual Income        0.038263
campaign             4.824712
last_contact_day     2.627820
age                  0.952016
dtype: float64
```

- If skewness is less than −1 or greater than +1, the distribution is highly skewed.
- If skewness is between −1 and −½ or between +½ and +1, the distribution is moderately skewed. If skewness is between −½ and +½, the distribution is approximately symmetric.
- **age,Count_Txn and annual_income & seems to be doing ok in terms of skewness while balance , duration, campaign, last_contact_day, are highly skewed and seems to have lot of outlier**

10. Feature Selection

Focused on two technques:

- **Weight of Evidence (WOE) and Information Value (IV) :They have been used as a benchmark to screen variables in the credit risk modeling projects such as probability of default**

```
1. job: 1.813615
2. education: 0.210222
3. ins_no: 0.014258
4. ins_yes: 0.014258
5. house_no: 0.365726
6. house_yes: 0.365726
7. loan_no: 0.092944
8. loan_yes: 0.092583
9. contact_cellular: 0.412392
10. contact_other: 0.587397
11. contact_telephone: 0.004230
12. poutcome_failure: 0.002925
13. poutcome_other: 0.416219
14. poutcome_success: 0.961266
15. marital_divorced: 0.000157
16. marital_married: 0.075774
17. marital_other: 0.001889
18. marital_single: 0.079493
19. gender_F: 0.000482
20. gender_M: 0.000482
```

- **Performed only on Categorical Values**

10. Feature Selection

Focused on two techniques:

- **Weight of Evidence (WOE) and Information Value (IV) : They have been used as a benchmark to screen variables in the credit risk modeling projects such as probability of default**

```
1. job: 1.813615
2. education: 0.210222
3. ins_no: 0.014258
4. ins_yes: 0.014258
5. house_no: 0.365726
6. house_yes: 0.365726
7. loan_no: 0.092944
8. loan_yes: 0.092583
9. contact_cellular: 0.412392
10. contact_other: 0.587397
11. contact_telephone: 0.004230
12. poutcome_failure: 0.002925
13. poutcome_other: 0.416219
14. poutcome_success: 0.961266
15. marital_divorced: 0.000157
16. marital_married: 0.075774
17. marital_other: 0.001889
18. marital_single: 0.079493
19. gender_F: 0.000482
20. gender_M: 0.000482
```

- **Performed only on Categorical Values**

Focused on two techniques:

- **Recursive Feature Elimination (RFE) using RandomClassifier**
  - The importance of features is determined by their contribution to the model's predictive performance.
  - Feature importance is calculated based on how much each feature contributes to reducing impurity or increasing information gain during the construction of individual decision trees within the random forest ensemble.

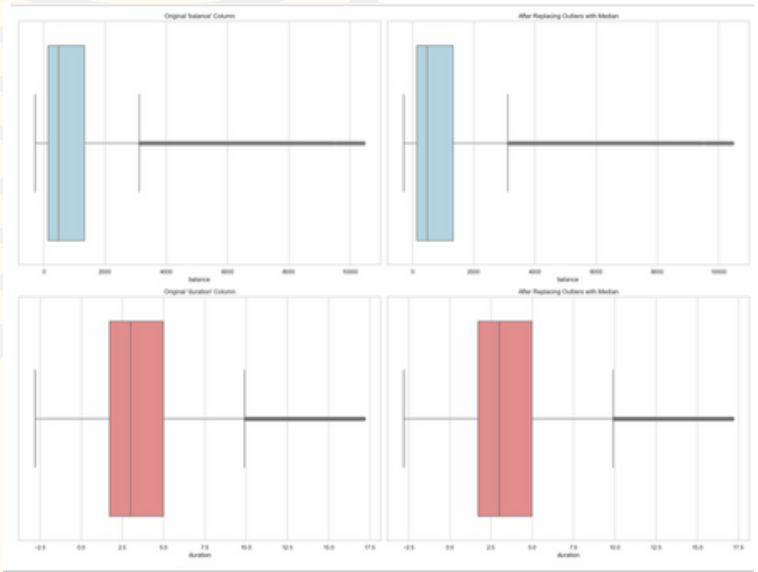
```
Selected Features:
Index(['balance', 'duration', 'campaign', 'last_contact_day', 'previous',
       'Count_Txn', 'age', 'job', 'Annual Income', 'poutcome_success'],
      dtype='object')
```

From the above techniques, we choose the common ones

9. Outlier Diagnosis

Imputate the outlier with Median

- Applied Z-score-based outlier handling for 'balance' and 'duration' columns.
- Outliers were identified and replaced with the median value.
- Z-score threshold set to 3 for outlier detection.
- Visualization through boxplots illustrates the impact of outlier handling



11. Model Evaluation after Feature Selection and Outlier Diagnosis

- **Random Forest**
- **After Outlier Treatment and feature selection random forest Training accuracy is 99% and Test accuracy is 89% (able to reduce overfitting not completely)**

```
==================================================

Results for RandomForestClassifier:

Cross-Validation Accuracy: 0.9021625547973949
Training Accuracy Score: 0.9999446933244842
Accuracy Score: 0.8977989160491097
```

Classification Report:

	precision	recall	f1-score	support
0.0	0.92	0.97	0.94	7946
1.0	0.64	0.35	0.45	1095
accuracy			0.90	9041
macro avg	0.78	0.66	0.70	9041
weighted avg	0.88	0.90	0.88	9041

Before Feature selection and Outlier detection

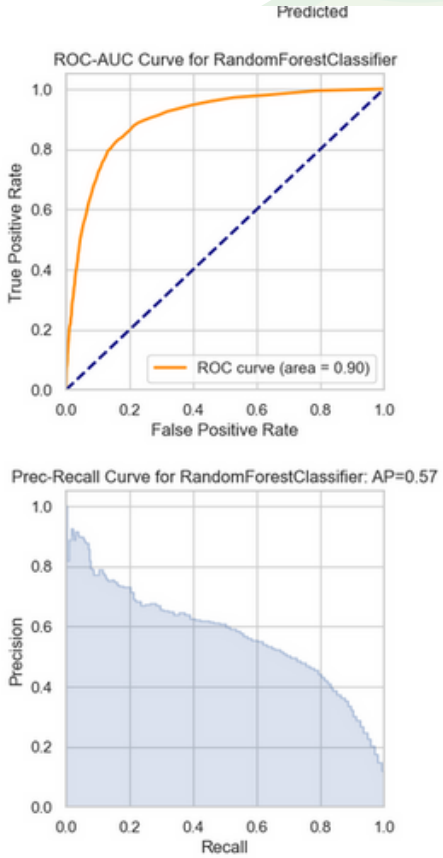
Classification Report:

	precision	recall	f1-score	support
0.0	0.92	0.97	0.94	7946
1.0	0.62	0.38	0.47	1095
accuracy			0.90	9041
macro avg	0.77	0.67	0.71	9041
weighted avg	0.88	0.90	0.89	9041

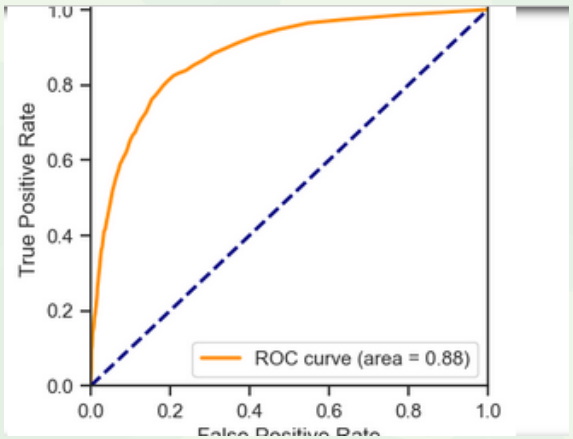
AfterFeature selection and Outlier detection

Little bit improvement in f1 score and recall score with a value of 45 % and 47 % which fared to be the best choice among individual classification models.

11. Model Evaluation after Feature Selection and Outlier Diagnosis



Before Feature selection and Outlier detection



After Feature selection and Outlier detection

Random forest performance is decrease in terms of area covered under the curve with an auc score of 88 %, from 90 per

11. Model Evaluation after Feature Selection and Outlier Diagnosis

- **XGBoost**
- **The gap between training and test accuracy is smaller than the Random Forest model. This suggests that the XGBoost model is performing better in terms of generalization to the test set.**
-

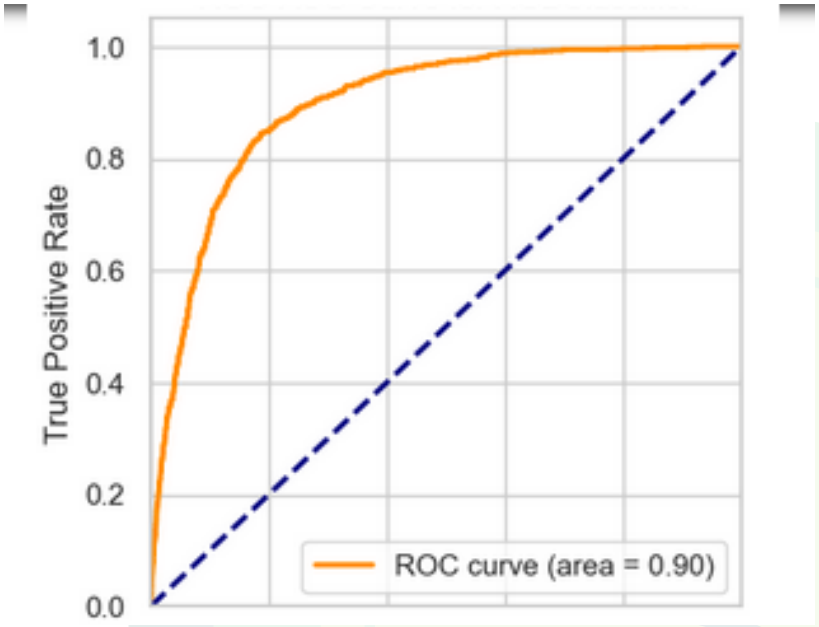
```
Training Accuracy Score: 0.9452740445771805
Accuracy Score: 0.8968034509456918
```

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

         0.0       0.92      0.96      0.94      7946
         1.0       0.59      0.39      0.47      1095

    accuracy                           0.89      9041
   macro avg       0.76      0.68      0.71      9041
weighted avg       0.88      0.89      0.88      9041
```

Before Feature selection and Outlier detection

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

         0.0       0.92      0.97      0.94      7946
         1.0       0.62      0.38      0.47      1095

    accuracy                           0.90      9041
   macro avg       0.77      0.67      0.71      9041
weighted avg       0.88      0.90      0.89      9041
```

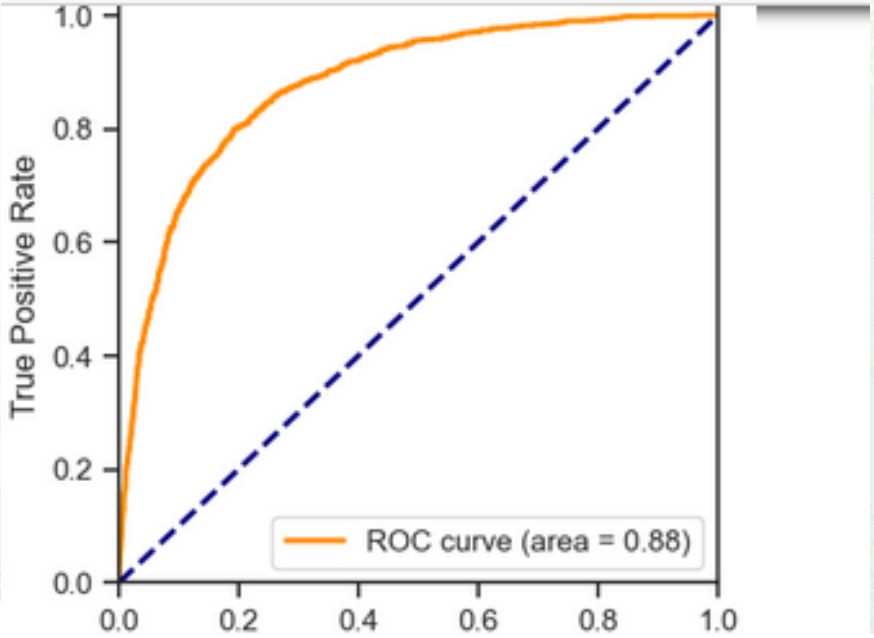
After Feature selection and Outlier detection



Before Feature selection and Outlier detection

XG boost is also decrease in terms of area covered under the curve with an auc score of 88 %, from 90 per

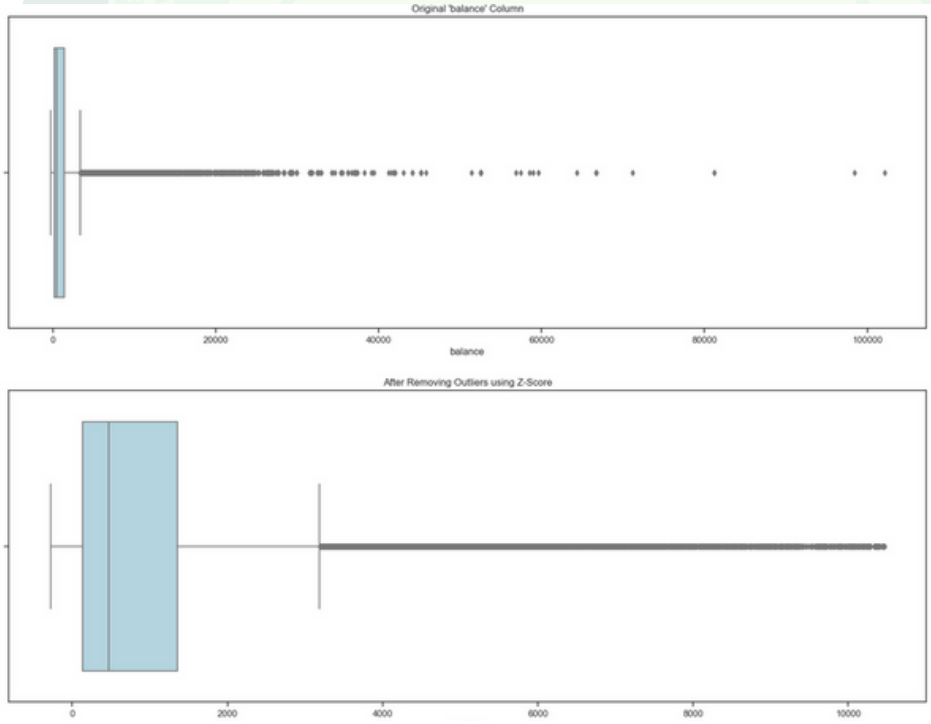
After Feature selection and Outlier detection



Clearly gradientboost clasifier outperforms random forrestor,

12. Model Evaluation Without Feature Selection, Outlier Removal and without one hot and label encoding

Outliers are removed completely

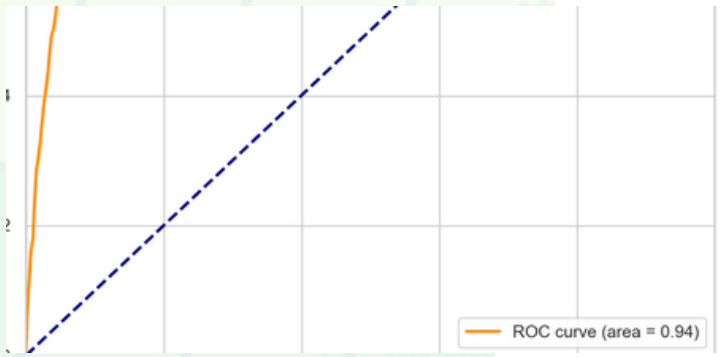


Manual Labelling

```
Unique values in 'Insurance' after replacement: [0 1]
Unique values in 'housing' after replacement: [1 0]
Unique values in 'contact' after replacement: [0 1 2]
Unique values in 'poutcome' after replacement: [0 1 2 3]
Unique values in 'job' after replacement: [ 0  1  2  3  4  5  6  7  8  9 10 1
Unique values in 'marital' after replacement: [0 1 2 3]
Unique values in 'education' after replacement: [0 1 2 3]
Unique values in 'Gender' after replacement: [0 1]
```

12. Model Evaluation Without Feature Selection, Outlier Removal (balance column) and without one hot and label encoding

```
Cross-Validation Accuracy: 0.9129506210855951
Training Accuracy Score: 0.9999718832592925
Accuracy Score: 0.9093567251461988
Precision Score: 0.632051282051282
Recall Score (Class 0): 0.9643129286258573
Recall Score (Class 1): 0.4950884086444008
F1 Score (Class 0): 0.9495366892061107
F1 Score (Class 1): 0.5525885558583106
Classification Report:
              precision    recall  f1-score   support

         0.0       0.94      0.96      0.95      7874
         1.0       0.63      0.50      0.55      1018

    accuracy                           0.91      8892
   macro avg       0.78      0.73      0.75      8892
weighted avg       0.90      0.91      0.90      8892
```



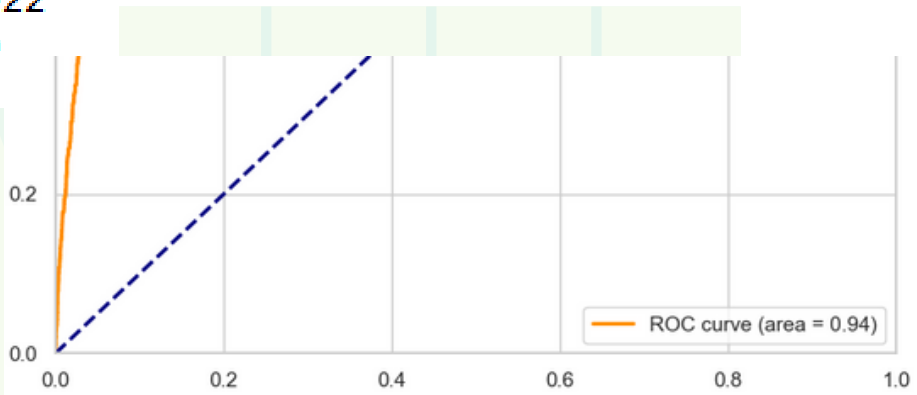
improvement in accuracy score, f1 score and recall score with a value of 47% to 52%which is far better than with outliers and encoding (one hot encoding)

Random Forest inecrease in terms of area covered under the curve with an auc score of 88 %, from 94 %

12. Model Evaluation Without Feature Selection, Outlier Removal and without one hot and label encoding

```
Cross-Validation Accuracy: 0.9092673628225922
Training Accuracy Score: 0.9683124332227409
Accuracy Score: 0.9073324336482231
```

improvement in accuracy score, f1 score and recall score with a value of 47% to 52%which fared to be the best choice among individual classification models.



Classification Report:

	precision	recall	f1-score	support
0.0	0.94	0.96	0.95	7874
1.0	0.61	0.52	0.56	1018
accuracy			0.91	8892
macro avg	0.78	0.74	0.76	8892
weighted avg	0.90	0.91	0.90	8892

XG boost inecrease in terms of area covered under the curve with an auc score of 88 %, from 94 %

Results

- **Performance of Random Forest Classifier and XG boost was better than others.**
- **Without feature selection, Outlier removal and upsampling reduced the problem of overfitting was detected.**
- **Able to remove (not completely) overfitting in Random Forest.**
- **Model with outlier removal and manual encoding performs better when compared with all of the metrics for XGboost.**
- **Thus, in this the removal of the balance column is a better solution than imputating it.**
- **Oversampling to balance out the data sample w,e can see a significant increase in the model accuracy score and recall score of Guassian NB, XG and Random forest. Thus, in the case of oversampling best choice to consider.**