Bank Telemarketing

CIND – 820

Omer Mirza

501077861

Supervisor - Ashok Bhowmick

Date of Submission: April 04, 2022

Ryerson University

Table of Contents

1. Abstract

1.1. Project Theme

The theme of this project is to analyze the effectiveness and suggest improving the direct marketing strategy of the firm using predictive analytics. Firms exist to make money. Therefore, they usually attain their objective, e.g. by gaining more new customers and selling more products or reducing losses from fraud in other cases. To achieve their goal, firms employ marketing strategies. These strategies define how the company shapes its product, promotion, pricing, and distribution to provide unique value to its customers and support its broader goals.

Our dataset came from a Portuguese bank that conducted direct marketing campaigns to promote term deposits to their customers. This project aims to analyze the bank's direct telemarketing strategy effectiveness using Predictive Analytics. Predictive analytics is based on patterns extracted from a historical data set, so those predictions then do not solve business problems rather, they provide insights that help the organization make better decisions to solve their business problems.

The common problem could be that the marketing strategy cannot identify potential customers who can be persuaded to make deposits and keeps re-calling the ‘’wrong’’ customers (those who don’t need term deposits). Moreover, it may cause high labour costs, and the strategy is susceptible to harming customer relationships. So, our business goal is to improve marketing effectiveness by targeting the right customers and analyzing the current strategy effectiveness.

Data mining is known as the process of monitoring new and innovative information from a vast amount of data sets by discovering hidden and unknown relationships between features entailed in the data records. Data mining (DM) has been used widely in direct marketing to identify prospective customers for new products by using purchasing data, a predictive model to measure that a customer will respond to the promotion or an offer. Accordingly, DM can be used to aid decision-makers in the banking sector to confront the economic pretense by avoiding risky transactions that cause bank attrition and increasing the customer retention incentives to raise the bank revenues

In this capstone project, the goal is to study different classification methods that can predict and explain the success of bank telemarketing. The data set entails a binary classification problem

related to direct marketing campaigns of a Portuguese banking institution based on phone calls (Moro et al., 2014). Our two classes are “yes” denoting that the customer subscribed to a term deposit, and “no,” denoting that the customer did not subscribe. Therefore, these classification models will help the bank reliably predict future customer subscriptions to secure deposits more effectively. Furthermore, the goal of any classification model is to predict a dependent variable using independent variables. This will increase customer satisfaction by reducing undesirable advertisements to customers who do not incline to such a product, hence reducing marketing costs.

1.2. What is Term Deposit

It is important to understand the Term deposit nature before going into this project.

A term deposit is a fixed-term investment that includes the deposit of money into an account at a financial institution. Term deposit investments usually carry short-term maturities ranging from one month to a few years and will have varying levels of required minimum deposits. As the nature of term deposit is fixed-term with minimum deposit levels Typically, term deposits offer higher interest rates than traditional liquid savings accounts, whereby customers can withdraw their money at any time. The investor must understand when buying a term deposit that they can withdraw their funds only after the term ends. In some cases, the account holder may allow the investor early termination—or withdrawal (Chen, J. (2019))

1.3. Research Question (Goal of the Project)

In this project, I will try to answer the following questions:

- Which data preprocessing techniques and data mining models perform best in predicting the success of bank marketing?
- Which feature has a higher probability of customers being more likely to subscribe to a term deposit?
- What type of customers are more likely to subscribe to term deposit and which feature?

1.4. Data Source Select and Information:

The data selected for this project is direct marketing campaigns of a Portuguese banking institution from UCI Machine Learning Repository.

http://archive.ics.uci.edu/ml/datasets/Bank+Marketing

The bank-additional (41,188 instances and 21 attributes) data set is related to direct marketing campaigns of a Portuguese banking institution based on phone calls. Often, more than one contact to the same client was required to assess the product (bank term deposit). The final outcomes indicate whether success campaigns are included in a binary format (yes/no). A successful campaign suggests the customer has finally subscribed to a term deposit at the end of the campaign.

Data info

Data Set Characteristics:	Multivariate	**Number of Instances:**	41188	**Area:**	Business
Attribute Characteristics:	Real	**Number of Attributes:**	21	**Date Donated**	2012-02-14
Associated Tasks:	Classification	**Missing Values?**	0	**Number of Web Hits:**	1716675

1.5. Tools to be Used:

The tool list in the below list is a representative list of tools to be used. As the project proceeds, more tools, if required to answer research questions, will be used:

1.5.1. Jupyter notebook for Python environment and related libraries including Pandas, NumPy, Seaborn, Plotly, Matplotlib and Sklearn. These library contains a lot of efficient tools for machine learning and statistical modeling

1.5.2. Various graph and plots including, Scatter, bar, and box plots, histograms to visualize the outcome of the results

1.5.3. For some data analysis, I will be using Microsoft excel

1.6. Summary of technics that will be used to answer the questions:

To answer the first question, the DM methods I will use are Logistic Regression (LR), Random Forest (RF), Multi-Layer Perceptron (MPL), Support Vector Machine (SVM), XG Boost (XGB). All data will be split into a training and testing set. Further SMOTE will be used as an oversampling technique to balance the data

Explanatory data analysis (EDA) and feature selection techniques will be used to answer the second and third research questions. EDA will help understand the data category visually, and feature selection such as Filters, Wrappers and embedded methods will be used to analyze the data and identify the contribution of each attribute.

2. Literature Review

2.1. **Predicting term deposit subscription from similar datasets**

The purpose of this literature review is to highlight different machine learning techniques & algorithms used in the previous research and results obtained from these studies, and what approaches to use for further analysis.

Several studies examined predicting the success of bank telemarketing for selling long-term deposits through the application of various machine learning techniques. The prior studies are reviews below:

(Moro et al., 2014) performed the baseline study using Portuguese bank marketing data first time and applied data mining to analyze the direct marketing campaign. The study's goal was to develop a predictive model capable of improving or increasing the efficiency of the direct marketing campaign for long-term deposit subscriptions by reducing the number of contacts to do; that is a reduction in the number of customers to be contacted phone. Using the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology, the authors collected real-world data on bank deposit subscriptions from the bank marketing campaign. In the course of the campaign, which was conducted with customers via telephone, an attractive long-term deposit application with good interest rates was offered. Using Naïve Bayes (NB), Decision Tree (DT) and Support Vector Machines (SVM) classifiers, (Moro et al., 2014) used 29 features and 45,211 instances to model the bank data. After applying classification algorithms, results from the analysis revealed that call duration is the most important feature, followed by the month of contact. In addition, they found the SVM to be the most reliable predictive algorithm with an Area Under Curve (AUC) value of 0.938. In a follow-up study, (Moro et al., 2014b) compare decision trees, logistic regression, neural networks and support vector machines. In this study, (Moro et al., 2014b) show that the neural network method outperforms all other methods in predicting the success of bank telemarketing.

(A.Elsalamony, 2014) used three statistical measures; classification accuracy, sensitivity and specificity on the same bank dataset (17 features and 45,211 instances) collected by (Moro et al., 2014). The goal was to increase the campaign effectiveness by identifying the main characteristics that affect the success (the deposit subscribed by the client). He compared and evaluated the classification performance of four different data mining models; Multilayer

Perception Neural Network (MPLNN), Tree Augmented Naïve-Bayes (TAN), Logistic Regression (LR) and C5.0 Decision Tree Classifier. The data set was partitioned into training and test by the ratio 70% and 30%, respectively. The results showed each model effectiveness. C5.0 has achieved slightly better performance than MLPNN, LR and TAN. Analysis showed that attribute "Duration" in C5.0, LR, and MLPNN models achieved the most important attribute; however, the attribute Age is the only assessed as more important than the other attributes by TAN.

Nachev (2015) applied cross-validation and multiple runs for the partitioning of train and test sets (70% and 30%) for the direct marketing response ta.sk. He found out that the two hidden layers architecture proposed by (A.Elsalamony, 2014) could be simplified into a single layer structure. He performed a comparative analysis of Neural Networks (NN), Logistic Regression, Naïve Bayes, Linear and Quadratic Discriminant Analysis (QDA), taking into account their performance at different levels of data saturation. Results revealed that the NN is the best performer in nearly all saturation levels except poorly saturated data (10-20%), where QDA showed better characteristics, measured by AUC. There was also a comparative ROC analysis of the models.

(Olatunji,2016) attempt to improve the performance of classification algorithms used in the bank customer marketing response prediction using the Random Forest ensemble. According to (Sołtys et al., 2014) ensemble methods are classes of highly successful machine learning algorithms that combines many different models to obtain an ensemble which should be more accurate than its constituent members. Classification algorithms used for modelling were; Logistic Regression, Decision Tree, Naive Bayes and the Random Forest ensemble. These algorithms were applied to both the balanced and original bank data by ten-fold cross-validation method. Results derived from the experiment showed that the performance of the Random Forest improved when the data was balanced. The Decision Tree algorithm returned 76.6% area under Curve (AUC) and Classification Accuracy (CA) compared to Logistic Regression 75.7% and Naive Bayes 75.6%. The Random Forest had an AUC and CA value of 74.2%. There were no found improvements in Random Forest (Olatunji, 2016). Therefore, second experiment was conducted and the results showed that the performance metrics of Random Forest increased with an increase of "n" to 200. (Olatunji, 2016). The second study found that changing the number of trees has no significant effects on mean accuracy of the Random forest. Random Forest and k-Nearest Neighbor are proved to be the best classifiers for any type of dataset. (Singh et al., 2017).

One of the Portuguese bank marketing full bank data set drawbacks is that it is highly imbalanced out of 45,211 instances, 36548 instances are of “no” as response, and 4, 289 responded a “yes”. Thus only 11.7% of the total number of customers contacted during the marketing campaign responded positively to the promotion (11.3% in bank-additional dataset containing 41,188 instances). In subsequent studies, the researcher focused on this particular fact and tried to improve the improve

(Safarkhani & Moro, 2021) focus on data mining classification and the performance of a decision tree algorithm called J48. The study focuses on a combination of re-sampling to reduce the imbalanced data, using feature selection, to reduce the complexity of data computing and dimension reduction of inefficiency data modelling. The goal of the study was to sell long-term deposits by telemarketing. They used 10% of the complete bank-additional-full 41188 actual instances data set by random selection with twenty-one identified attributes and two types of attribute as categorical and numeric. A 10-fold cross-validation scheme was adopted to ensure independence between training and test sets.

The selected dataset was imbalanced, and only 451 samples were related to success, all other samples, 3668, belong to failure samples with twenty-one identified attributes and two types of attribute as categorical and numeric. SMOTE method was applied on the imbalanced dataset. An increment in the minority class was accompanied by no change in the majority class. The results showed that the SMOTE method could solve the imbalanced data problem. WrappedSubsetEval was used to select the useful features and remove the other extra ones; it provided dimension reduction to improve speed, accuracy, efficiency and performance. Attribute evaluator for the J48 model, selected 12 attributes out of the 20 attributes that were input.

The experimental results using J48 decision tree achieved 94.39% accuracy prediction, with 0.975 sensitivity and 0.709 specificity, showing better results when compared to other approaches reported in the existing literature, such as logistic regression (91.79 accuracy; 0.975 sensitivity; 0.495 specificity) and Naive Bayes classifier (90.82% accuracy; 0.961 sensitivity; 0.507 specificity). Furthermore, re-sampling and feature selection approach resulted in improved accuracy (94.39%) compared to a state-of-the-art approach based on a fuzzy algorithm (92.89%).

(Ghatasheh et al., 2020) study to mitigate the effects of highly imbalanced data in realizing an enhanced cost-sensitive prediction model. The research considers the bank full with 10% of the examples (4521) and 17 inputs file for model building and validation.

The study used enhanced Artificial Neural Network models (i.e., cost-sensitive) to mitigate the dramatic effects of highly imbalanced data, without distorting the original data samples. Artificial Neural Network (ANN) models have been used broadly in marketing to predict customers' behaviour. Recent research contributions warn from the limitations and shortcomings accompanying re-sampling approaches. In particular, questionable reliability of produced models, i.e., while under-sampling approach may discard important instances, over-sampling approach may result in generating over-fitting models.

Several classification models including Lib-LINEAR (LL), i.e., an implementation of Support Vector Machines, Decision Table (DT), Very Fast Decision Rules (VFDR), Random Forest Trees (RF), Multilayer Perceptron (MLP), J48, and Deep Learning for MLP Classifier (DL-MLP) were constructed and evaluated in order to capture the added value by adding cost sensitivity analysis to the conventional classification algorithm. In particular, two distinct cost-sensitive methods were used, namely Cost-Sensitive Classifier and Meta-Cost. Performance of conventional machine-learning classifiers in Weka environment against the best cost-sensitive prediction models. The theory behind cost-sensitive classification is to either (a) re-weight training inputs in line with a pre-defined class cost or (b) predict a class with the lowest misclassification cost [1,40].]. In cost-sensitive learning, adding cost sensitivity to the base algorithm is done either by re-sampling input data or by re-weighting misclassification errors.

Meta-Cost over Cost-Sensitive Classification method results better in some cases during research. Results are reported in terms of TPR, TNR, Geometric Mean, Type | Error, Type || Error, and Total Accuracy metrics.

Algorithm	TPR	TNR	Geometric Mean	Type I Error	Type II Error	Accuracy
(A) Our approach						
Meta-Cost-MLP	**0.808**	0.771	**78.93%**	**0.192**	0.229	77.48
CostSensitiveClassifier-MLP	0.614	0.872	73.17%	0.386	0.128	84.18
(B) Conventional machine-learning classifiers						
MLP (Baseline)	0.39	0.95	60.87%	0.61	0.05	88.98
DL-MLP	0.36	0.93	57.86%	0.64	0.07	86.24
J48	0.36	0.96	58.79%	0.64	0.04	88.9
LL	0.47	0.83	62.46%	0.53	0.17	78.61
DT	0.33	0.97	56.58%	0.67	0.03	89.49
VFDR	0.46	0.76	59.13%	0.54	0.24	72.61
RF	0.27	**0.98**	51.44%	0.73	**0.02**	**89.82**
(C) Other cost-sensitive results from related works						
CSDE * [1]	0.705	0.62.2	66.2%	n.a.	n.a.	n.a.
CSDNN ** [1]	0.615	0.542	57.9%	n.a.	n.a.	n.a.
AdaCost [1]	0.89	0.22	44.2%	n.a.	n.a.	n.a.
Meta-Cost [1]	0.35	0.868	55.1%	n.a.	n.a.	n.a.

* CSDE: Cost-Sensitive Deep Neural Network Ensemble, ** CSDNN: Cost-Sensitive Deep Neural Network.

Table A (Ghatasheh et al., 2020)

2.2. A brief descriptive statistic of the selected datasets:

The data is related to direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required to access if the product (bank term deposit) would be ('yes') or not

The public dataset is provided by (Moro et al., 2014), which includes real-world data covering the years from May 2008 to June 2013. 1). For this project, I will be using a bank-additional-full data set, very close to the data analyzed (Moro et al., 2014). There are 20 input variables divided into seven numeric and nine nominal attributes. The target variable named "y" is a binary class indicating whether a client applied for a term deposit or not. All 21 variables are described in Tables 1 to 5.

Data has 41,188 rows and 21 attributes with a highly unbalanced class label. The positive class "Yes" of the target variable "Y" has 4,640 observations (i.e., a client has subscribed for a term deposit), which is the class of interest in this case., which is 11.3% (Table 6).

The negative class "No" of target variable "Y" is 36,548 observations that is 89.7%. There are no missing Values (table 4) in the data and 12 duplicates (table 5)

The data has outliner. There is a high variation between values and abnormal distribution for data (Table 7 &8). Histogram graphs highlight the attribute's distribution or possible outliers (Table 9).

Data can be found here:

https://archive.ics.uci.edu/ml/datasets/bank+marketing

There are four datasets:

bank-additional - Close to the data analyzed in (Moro et al., 2014)

The bank’s additional data sets contain a smaller population, 41188, but more importantly, more social and economic context information features.

1) bank-additional-full.csv with all examples (41188) and 20 inputs, ordered by date (from May 2008 to November 2010), very close to the data analyzed in [Moro et al., 2014]
2) bank-additional.csv with 10% of the examples (4119), randomly selected from 1), and 20 inputs.

bank-full - Older Version with 17 inputs

The standard bank data sets carry 17 variables in their content, no missing values, and refer to clients bank information only.

3) bank-full.csv with all examples (45211) and 17 inputs, ordered by date (older version of this dataset with less inputs).
4) bank.csv with 10% of the examples (4521) and 17 inputs, randomly selected from 3 (older version of this dataset with less inputs).

Table. 1 Data Attributes, Description, Type and Category Details:

S. No	Category	Category Description	Data Type	Category Detail
# Personal Attributes				
1	age	age of customer	numeric	18-95
2	job	type of job	categorical	'admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown'
3	marital	marital status	categorical	'divorced','married','single'
4	education	level of education	categorical	basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown'
5	default	has credit in default?	categorical	'no','yes','unknown'
6	housing	has housing loan?	categorical	'no','yes', 'unknown'
7	loan	has personal loan?	categorical	'no','yes', 'unknown'
# related with the last contact of the current campaign:				
8	contact	contact communication type	categorical	'cellular','telephone'
9	month	last contact month of year	categorical	'jan', 'feb', 'mar', ..., 'nov', 'dec'
10	day_of_week	last contact day of the week	categorical	'mon','tue','wed','thu','fri'
11	duration	last contact duration, in seconds	numeric	0-3643, this attribute highly affects the output target (e.g., if duration=0 then y='no'. Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.

S. No	Category	Category Description	Data Type	Category Detail
# other campaign attributes:				
12	campaign	number of contacts performed during this campaign and for this client	numeric, includes last contact	42 different ones 1-56
13	pdays	number of days that passed by after the client was last contacted from a previous campaign	numeric; 999 means client was not previously contacted	27 different ones 0-999
14	previous	number of contacts performed before this campaign and for this client	numeric	0-7
15	poutcome	outcome of the previous marketing campaign	categorical	'failure','nonexistent','success'
# social and economic context attributes				
16	emp.var.rate	employment variation rate - quarterly indicator	numeric	1.1;1.4;-0.1;-0.2;-1.8;-2.9;-3.4;-3;-1.7;-1.1
17	cons.price.idx	consumer price index - monthly indicator	numeric	92.201 - 94.767
18	cons.conf.idx	consumer confidence index - monthly indicator	numeric	-26.9 , -50.8
19	euribor3m	euribor 3 month rate - daily indicator	numeric	0.634 - 5.045
20	nr.employed	number of employees - quarterly indicator	numeric	4963.6 - 5228.1
Output variable (desired target):				
21	y - has the client subscribed a term deposit?		binary	'yes','no'

Table. 2 Sample Data View:

```
bank.head()
```

	age	job	marital	education	default	housing	loan	contact	month	day_of_week	duration	campaign	pdays
0	56	housemaid	married	basic.4y	no	no	no	telephone	may	mon	261	1	999
1	57	services	married	high.school	unknown	no	no	telephone	may	mon	149	1	999
2	37	services	married	high.school	no	yes	no	telephone	may	mon	226	1	999
3	40	admin.	married	basic.6y	no	no	no	telephone	may	mon	151	1	999
4	56	services	married	high.school	no	no	yes	telephone	may	mon	307	1	999

previous	poutcome	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed	y
0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no
0	nonexistent	1.1	93.994	-36.4	4.857	5191.0	no

Table. 3 Data shows “no” null value

```
Data columns (total 21 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   age             41188 non-null  int64
 1   job             41188 non-null  object
 2   marital         41188 non-null  object
 3   education       41188 non-null  object
 4   default         41188 non-null  object
 5   housing         41188 non-null  object
 6   loan            41188 non-null  object
 7   contact         41188 non-null  object
 8   month           41188 non-null  object
 9   day_of_week     41188 non-null  object
 10  duration        41188 non-null  int64
 11  campaign        41188 non-null  int64
 12  pdays           41188 non-null  int64
 13  previous        41188 non-null  int64
 14  poutcome        41188 non-null  object
 15  emp.var.rate    41188 non-null  float64
 16  cons.price.idx  41188 non-null  float64
 17  cons.conf.idx   41188 non-null  float64
 18  euribor3m       41188 non-null  float64
 19  nr.employed     41188 non-null  float64
 20  y               41188 non-null  object
dtypes: float64(5), int64(5), object(11)
memory usage: 6.6+ MB
```

Table 4. There are ‘0’ missing value and list of each attributes unique Value

```
Missing values :   0

Unique values :
 age                78
job                 12
marital              4
education            8
default              3
housing              3
loan                 3
contact              2
month               10
day_of_week          5
duration          1544
campaign            42
pdays               27
previous             8
poutcome             3
emp.var.rate        10
cons.price.idx      26
cons.conf.idx       26
euribor3m          316
nr.employed         11
target               2
dtype: int64
```

Table. 5 Details of Duplicate

```
Number of clients: ..............  41188
Number of variables: ............  21 incl. target
Number of duplicate entries: ....  12
```

Table 6 Visualization of Imbalance data

```
[0] Number of clients that haven't subscribed the term deposit: ... 36548
[1] Number of clients that have subscribed the term deposit: ....... 4640
```

	clients	%
0	36548	88.73
1	4640	11.27

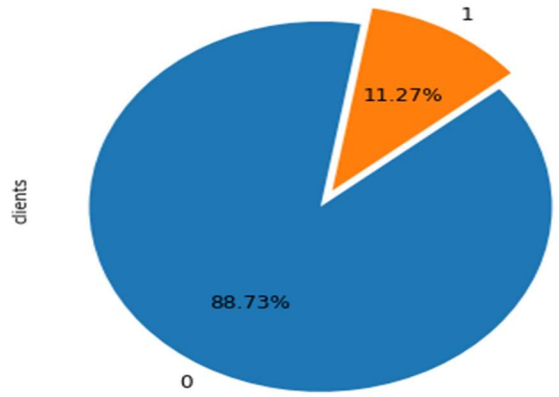


Table 7 – General Stats of Numeric Data

```
# General stats of numeric variables
bank.describe()
describe.append(pd.Series(bank.var(), name='variance'))
```

	age	duration	campaign	pdays	previous	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed	target
count	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000
mean	40.024060	258.285010	2.567593	962.475454	0.172963	0.081886	93.575664	-40.502600	3.621291	5167.035911	0.112654
std	10.421250	259.279249	2.770014	186.910907	0.494901	1.570960	0.578840	4.628198	1.734447	72.251528	0.316173
min	17.000000	0.000000	1.000000	0.000000	0.000000	-3.400000	92.201000	-50.800000	0.634000	4963.600000	0.000000
25%	32.000000	102.000000	1.000000	999.000000	0.000000	-1.800000	93.075000	-42.700000	1.344000	5099.100000	0.000000
50%	38.000000	180.000000	2.000000	999.000000	0.000000	1.100000	93.749000	-41.800000	4.857000	5191.000000	0.000000
75%	47.000000	319.000000	3.000000	999.000000	0.000000	1.400000	93.994000	-36.400000	4.961000	5228.100000	0.000000
max	98.000000	4918.000000	56.000000	999.000000	7.000000	1.400000	94.767000	-26.900000	5.045000	5228.100000	1.000000
variance	108.602451	67225.728877	7.672975	34935.687284	0.244927	2.467915	0.335056	21.420215	3.008308	5220.283250	NaN

smaller standard deviation indicates that more of the data is clustered about the mean while a larger once indicates the data are more spread out.

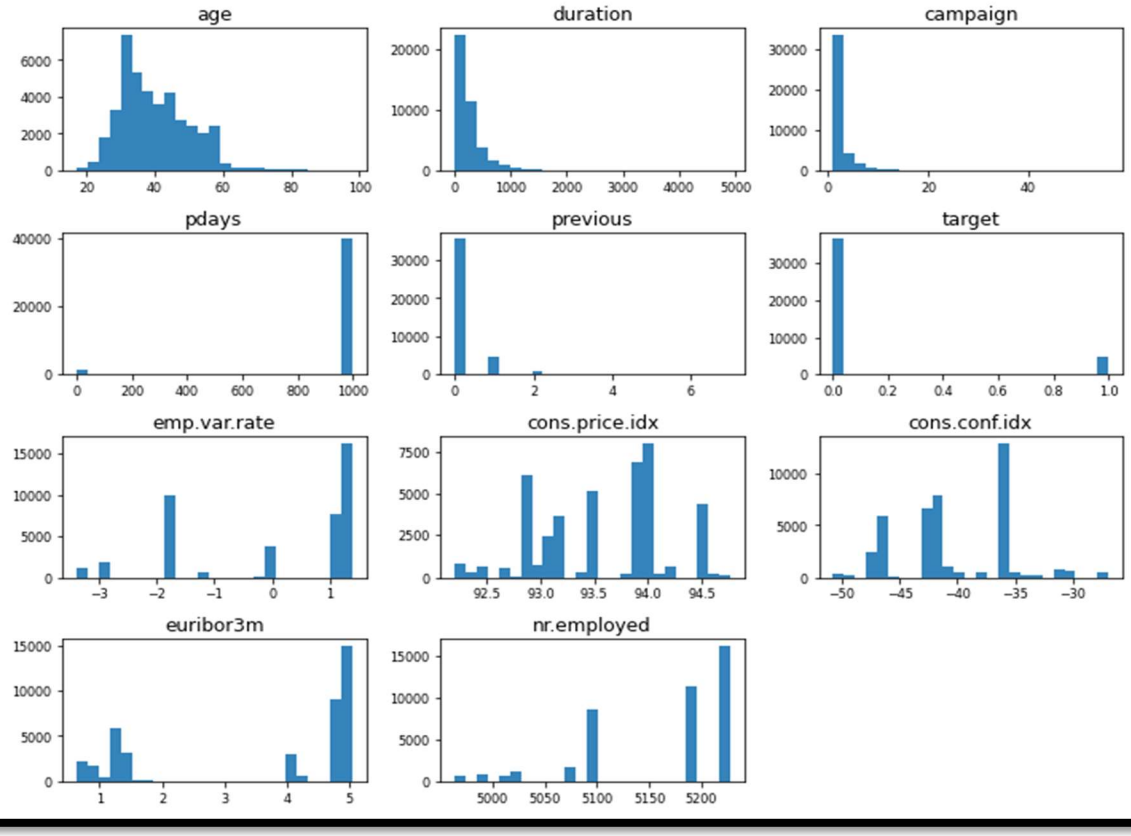
Table 8– General Stats of Category Data

```
# General stats of categoric variables
bank.describe(include=['object'])
```

	job	marital	education	default	housing	loan	contact	month	day_of_week	poutcome	y
count	41188	41188	41188	41188	41188	41188	41188	41188	41188	41188	41188
unique	12	4	8	3	3	3	2	10	5	3	2
top	admin.	married	university.degree	no	yes	no	cellular	may	thu	nonexistent	no
freq	10422	24928	12168	32588	21576	33950	26144	13769	8623	35563	36548

Table 9 – Histogram of the numeric Distribution

```
# Visualization of the numeric distribution
bank[num_features].hist(figsize=(10,8), bins=25, xlabelsize=8, ylabelsize=8, alpha=0.9, grid=False)
plt.tight_layout();
```



2.3. **Link of Code and result for this project to a repository on the GitHub website is below:**

https://github.com/omerbmk/ CIND-820-OmerMirza/blob/main/CIND_820_Bank_Telemarketing.ipynb

2.4. A graph showing the tentative overall methodology

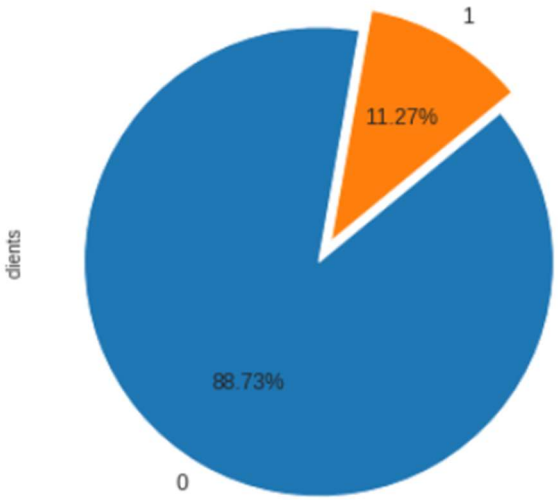
3. Final Results and Project Report

3.1. Exploratory Data Analysis

3.1.1. Some of the characteristics of overall Bank-additional dataset

- Shape of data frame (Cols, rows): (41188, 21)
    - Number of records: 41188
    - Number of variables: 21
- Number of Missing Value: 0
- Number of Duplicate 12
- Type of Data columns
    - Numeric variable 10
    - Categorical variable 11
- Data is highly imbalance
    - Number of clients that haven't subscribed the term deposit: 36548 (88.73%)
    - Number of clients that have subscribed the term deposit: 4640 (11.27%)

This shows that data set is an imbalance data set where the minority class is attributed to successful term deposit subscriptions



3.1.2. Numeric variables Analysis

S. No.	Numerical Atributes
1	age
2	duration
3	campaign
4	pdays
5	previous
6	emp.var.rate
7	cons.price.idx
8	cons.conf.idx
9	euribor3m
10	nr.employed

Table 10: Numeric Values

```
df_bank.describe()
```

	age	duration	campaign	pdays	previous	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed
count	41188.00000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000
mean	40.02406	258.285010	2.567593	962.475454	0.172963	0.081886	93.575664	-40.502600	3.621291	5167.035911
std	10.42125	259.279249	2.770014	186.910907	0.494901	1.570960	0.578840	4.628198	1.734447	72.251528
min	17.00000	0.000000	1.000000	0.000000	0.000000	-3.400000	92.201000	-50.800000	0.634000	4963.600000
25%	32.00000	102.000000	1.000000	999.000000	0.000000	-1.800000	93.075000	-42.700000	1.344000	5099.100000
50%	38.00000	180.000000	2.000000	999.000000	0.000000	1.100000	93.749000	-41.800000	4.857000	5191.000000
75%	47.00000	319.000000	3.000000	999.000000	0.000000	1.400000	93.994000	-36.400000	4.961000	5228.100000
max	98.00000	4918.000000	56.000000	999.000000	7.000000	1.400000	94.767000	-26.900000	5.045000	5228.100000

Table 11: Descriptive statistics

3.1.3. Main statistical features:

1. age: the youngest client has 17 years old and the oldest has 98 years with a median of 38 years whilst the average is 40 years old.

2. pdays: The majority of the clients have the 999 number which indicates that most people did not contact nor were contacted by the bank. Those are considered to be 'out of range' values.

3. previous: The vast majority were never contacted before

4. emp_var_rate: during the period the index varied from [-3.4, 1.4]

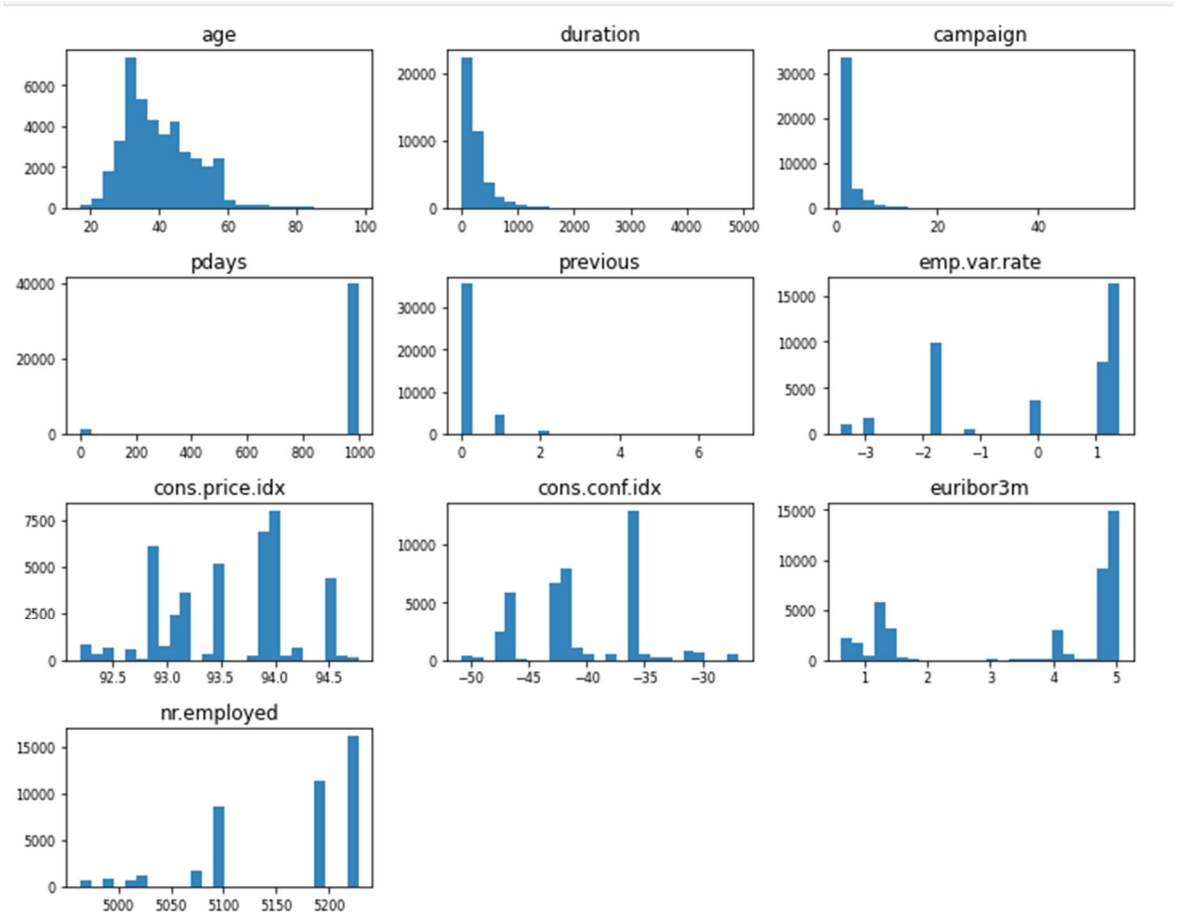
5. cons_price_idx: the index varied from [92.2, 94.8]

6. cons_conf_idx: the consumer confidence level during that period kept always negative with a range of variation of [-51, -27]. These negative values might be explained by the recession that severely affected Portugal due the financial global crisis during that same period the data was recorded

7. euribor3m: there were a huge variation of the euribor rate during the period of analysis [5% to 0.6%]. This abrupt change in euribor together with the negative confidence verified reinforces the hypothesis that this data provides information from a crisis period

8. nr_employed: the number of employed people varied around 200 during the campaign

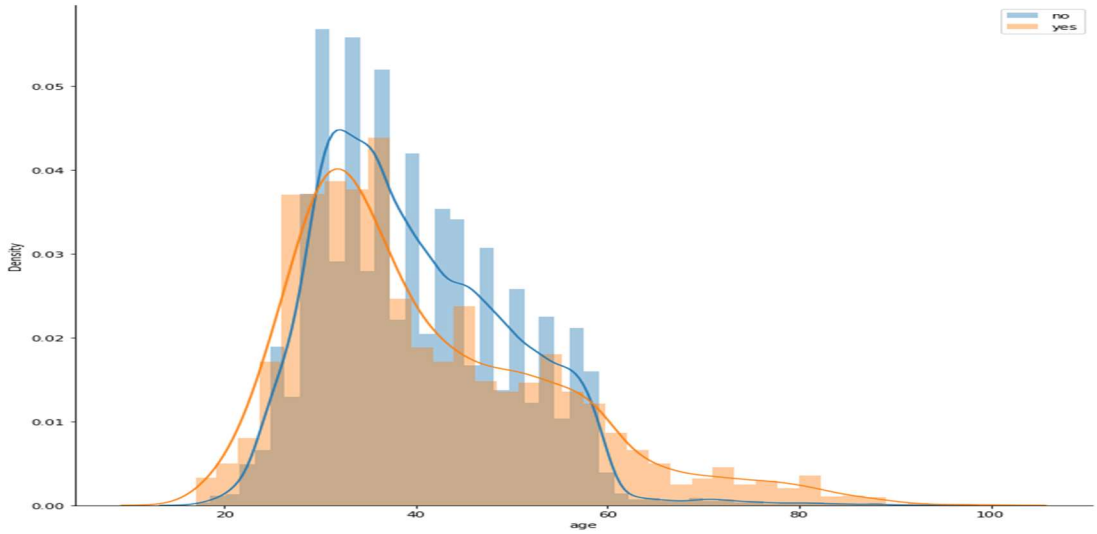
3.1.4. Visualization of Numeric Dataset



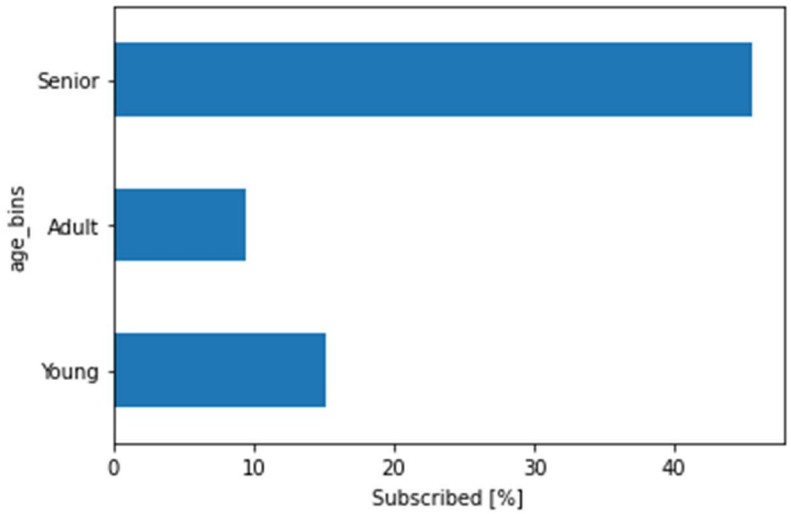
3.1.5. Detail Analysis of Numeric Data

3.1.5.1. Age

Customers aged 30-40, 20-30, and 40-50 had a higher percentage of subscription to a deposit account. It is more clear that age might not be very helpful in prediction of class labels because there is so much of overlapping. After age of 60 which might be our outliers, there is not that much of overlapping



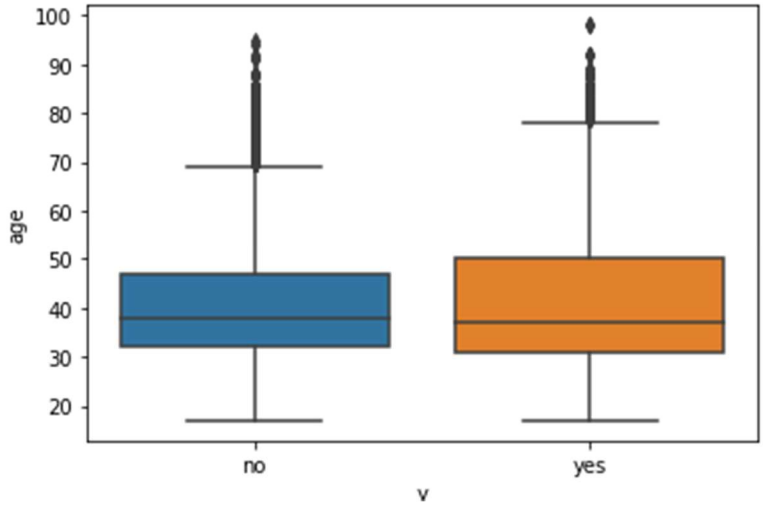
It is very clear the relation between the subscription rate and age of customers:



45.5% of Seniors (+60 years old) subscribed to the term deposit

less than 10% Adults (>30 and <=60 years old) subscribed

Young people were the 2nd group that subscribed the deposit corresponding to 1/6 of all young people

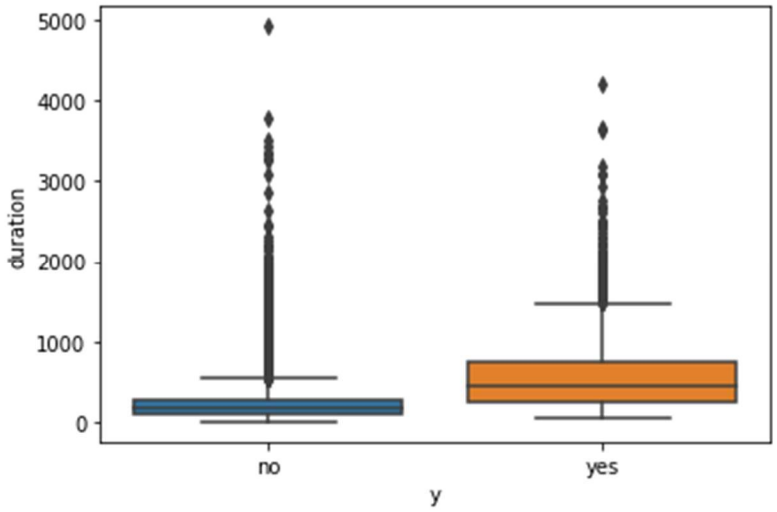


From the above it is clearly visible that there are outliers present for both the class. In No class, outliers are present above age 70 and for Yes class, outliers are present above age 75. Median for No class is around 38-40 which is same for Yes class. Also, it is visible that IQR range is almost overlapping so age might not be very helpful in predicting class label.

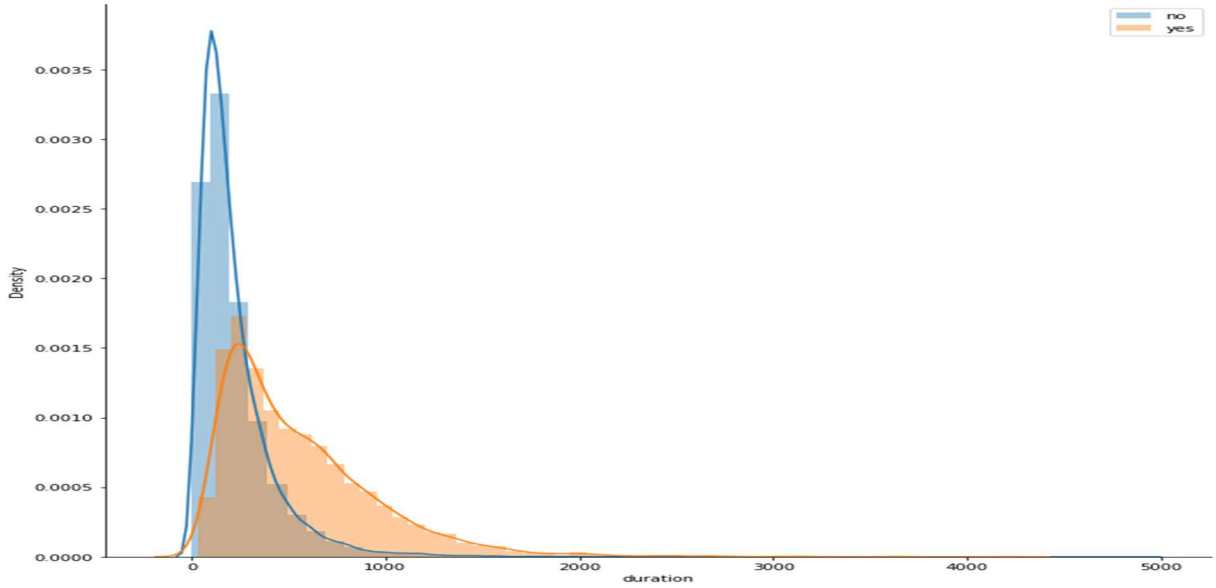
3.1.5.2. Duration

This feature is very interesting in our case study. It denotes the duration of the last contact, in seconds. It is mentioned in the source of the dataset:

> Important note: This attribute highly affects the output target (e.g., if duration=0 then y=’no’). Yet, the duration is not known before a call is performed. Also, after the end of the call, the target variable y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.



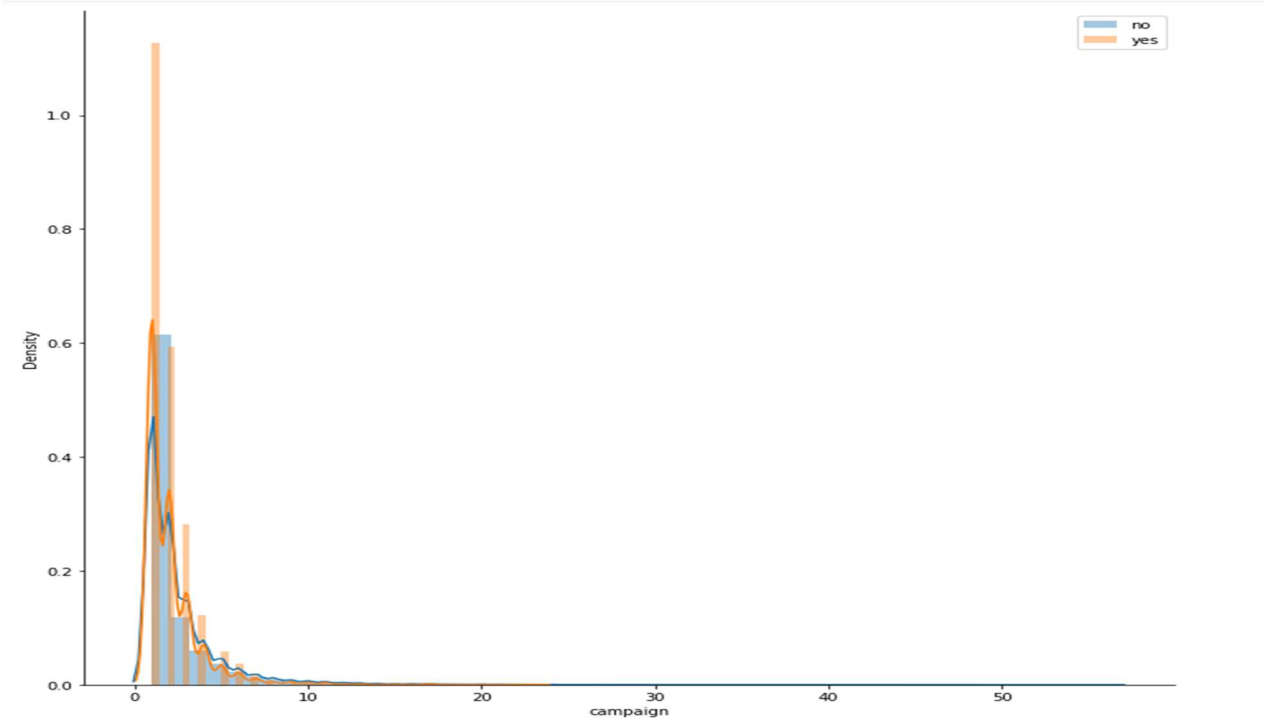
Any duration of call with class labels as no, more than 1000 are considered as outliers while with class labels yes, more than 1500 would be considered as outliers.

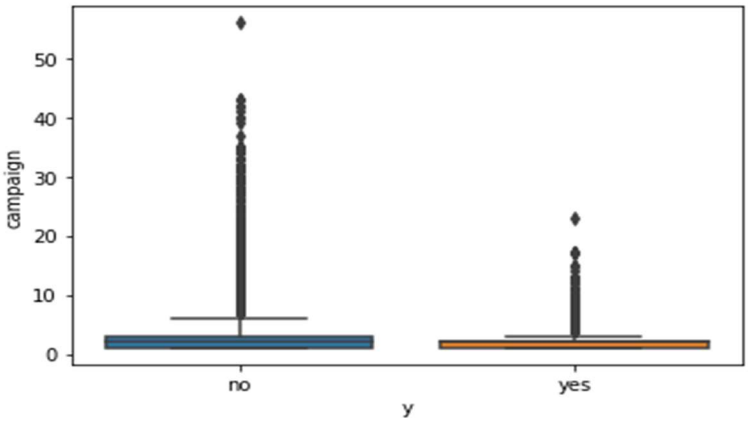


From the above plot it is clear that, the duration (last contact duration) of a customer can be useful for predicting the target variable. It is expected because it is already mentioned in the data overview that this field highly affects the target variable and should only be used for benchmark purposes.

3.1.5.3. Campaign

List number of contacts performed during this campaign and for this client. From the below plot, though outliers are present but there is so much overlapping

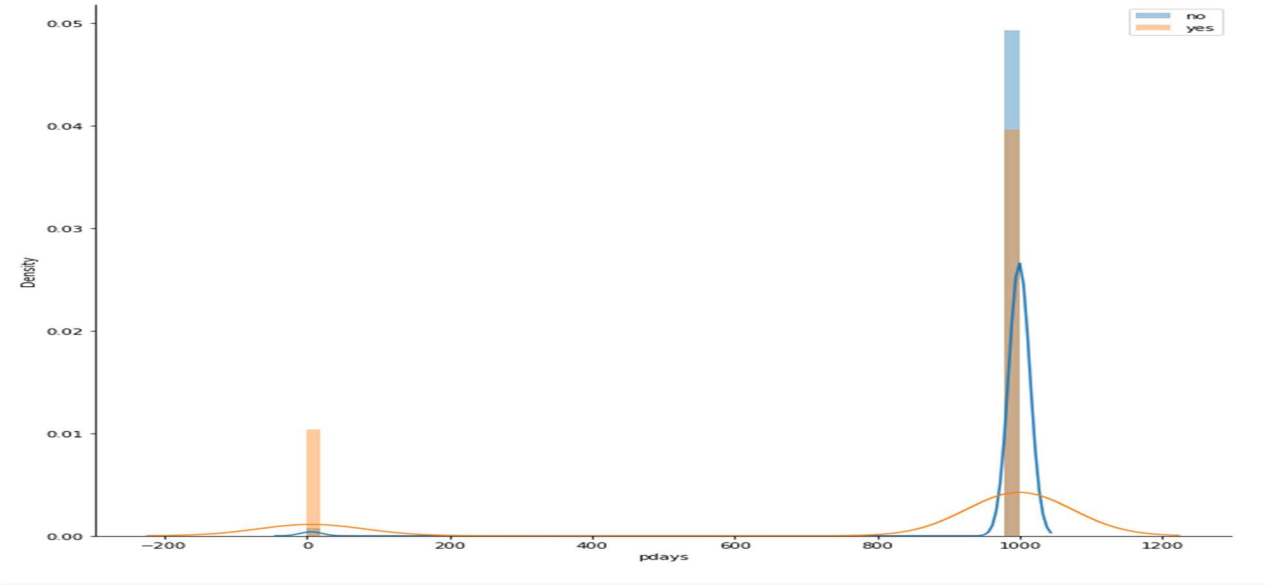


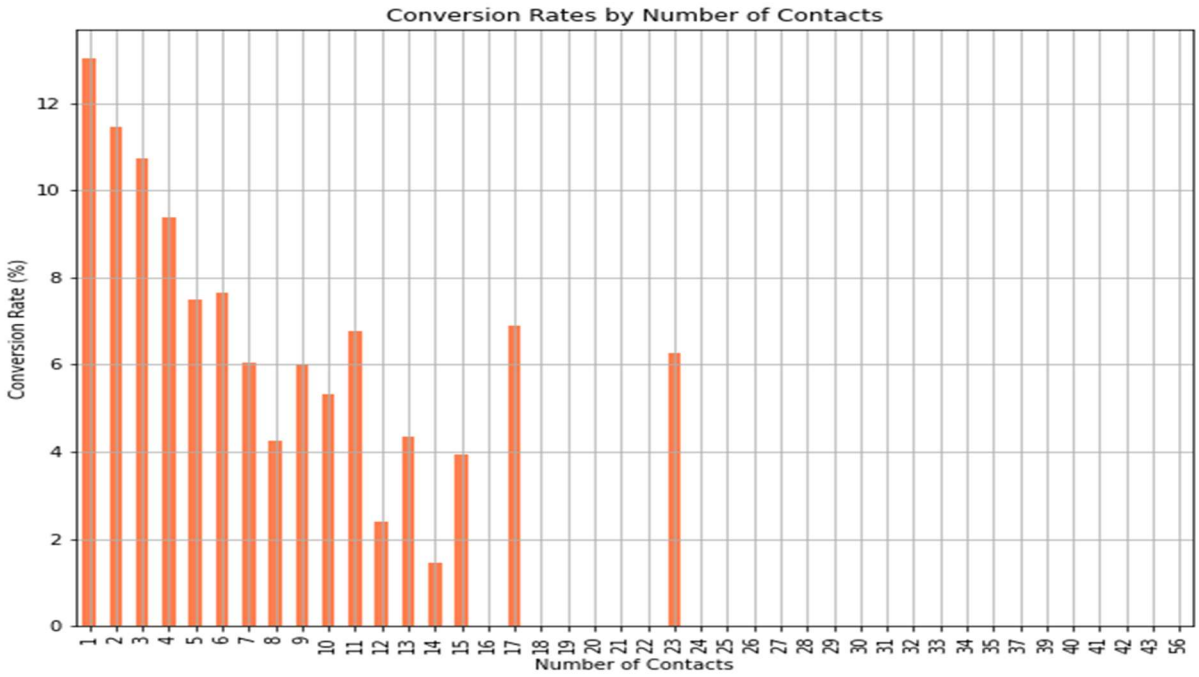


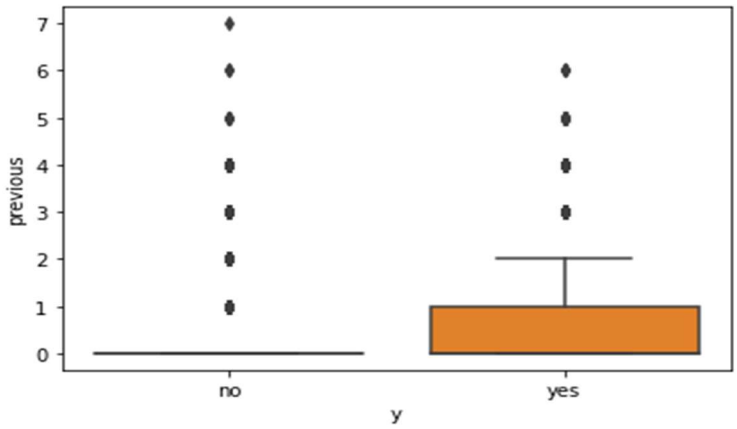
Outliers are present when number of campaign are more than 10 irrespective of any class labels.

3.1.5.4. pdays

List number of days that passed by after the client was last contacted from a previous campaign. Most of the values are 999, which means that the most of the customers have never been contacted before.

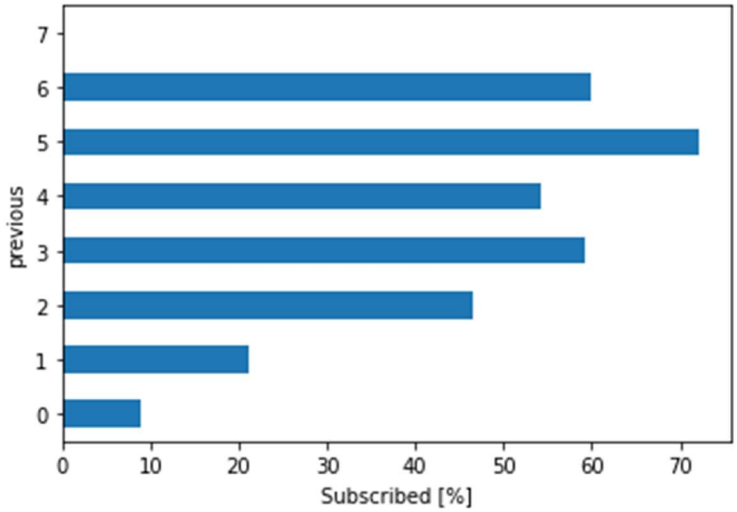

no
yes
Density
0.05
0.04
0.03
0.02
0.01
0.00
−200
0
200
400
600
800
1000
1200
pdays


Conversion Rates by Number of Contacts
Conversion Rate (%)
0
2
4
6
8
10
12
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 37 39 40 41 42 43 56
Number of Contacts



From the above plot it is visible that irrespective of class labels, mostly people has not been contacted by the bank. Very people has been contacted by the bank and number of days passed for previous campaign is between 0–100. It means we either have to compute pdays or drop the pdays depends on the percentage of values.

```
How many people were previously contacted? ................. 5625
How many people were never re-contacted 1 times? ..................... 35563
How many people were contacted atleast 1 times? ..................... 4561
How many people were previously contacted with sucess? ..... 4252
How many people were previously contacted with failure? .... 1373
```



People that were previously contacted subscribed in a much higher rate to the term deposit. While in people never contacted only 10% subscribed to the deposit, for people that was previously contacted more than twice the campaign success increases to >45%.

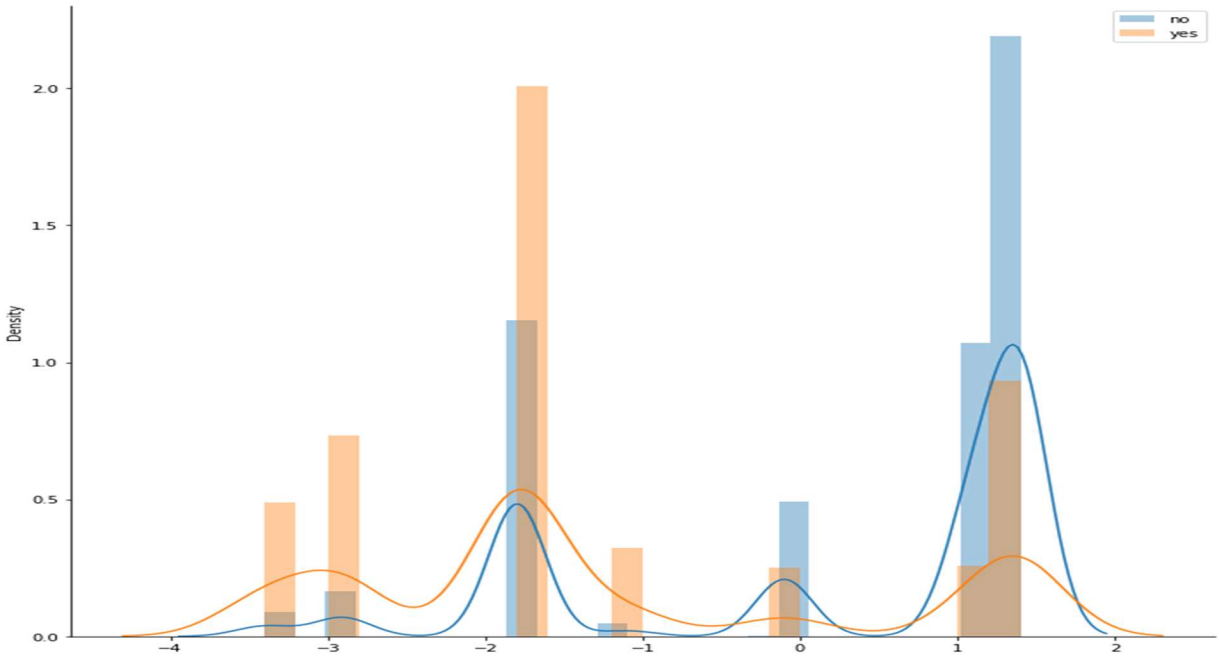
Indexes variables

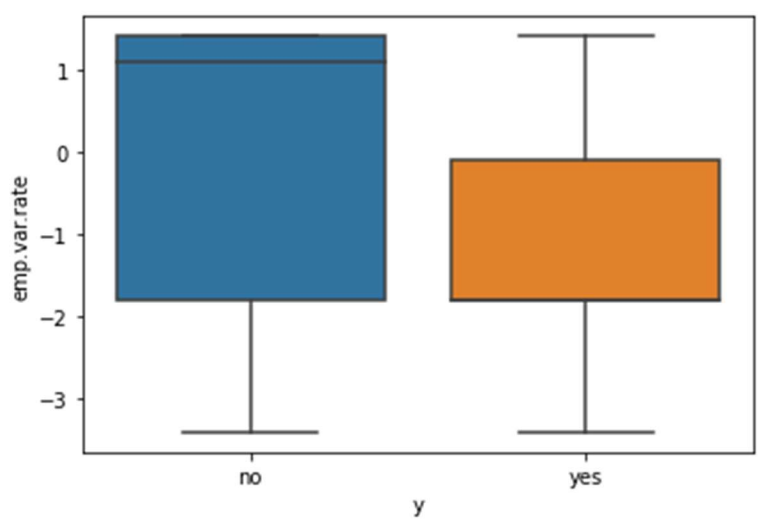
There are 5 macro rating variables, or economic indexes, present in the dataset.

a) emp.var.rate
b) cons.price.idx
c) cons.conf.idx
d) euribor3m
e) nr.employed

	cons.price.idx	cons.conf.idx	euribor3m	emp.var.rate	nr.employed	target
cons.price.idx	1.000000	0.058986	0.688230	0.775334	0.522034	-0.136211
cons.conf.idx	0.058986	1.000000	0.277686	0.196041	0.100513	0.054878
euribor3m	0.688230	0.277686	1.000000	0.972245	0.945154	-0.307771
emp.var.rate	0.775334	0.196041	0.972245	1.000000	0.906970	-0.298334
nr.employed	0.522034	0.100513	0.945154	0.906970	1.000000	-0.354678
target	-0.136211	0.054878	-0.307771	-0.298334	-0.354678	1.000000

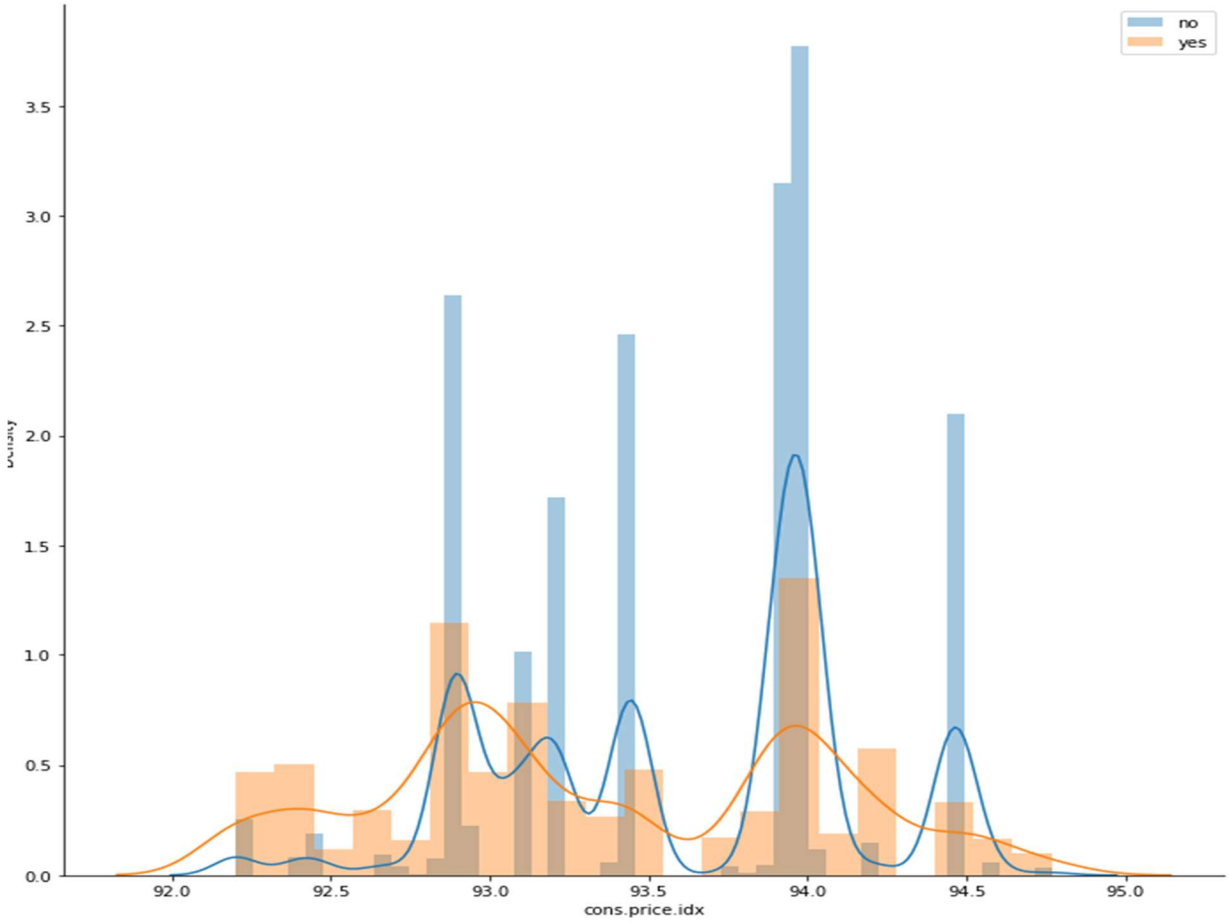
3.1.5.5. emp.var.rate

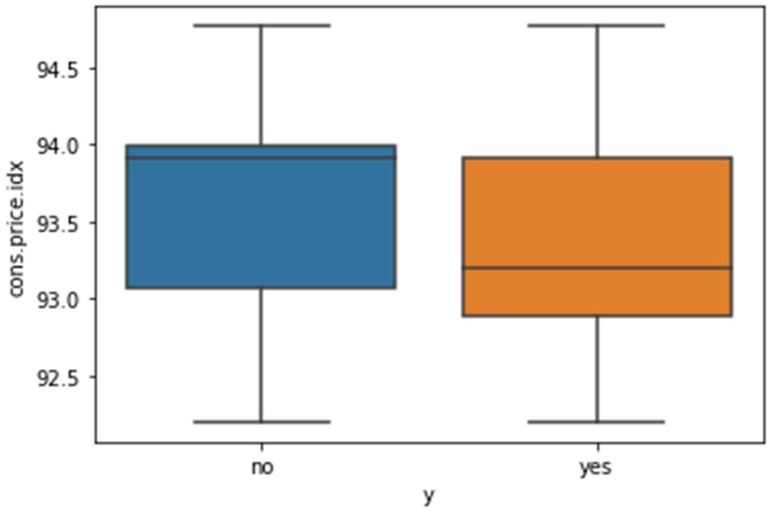




There are no outliers present

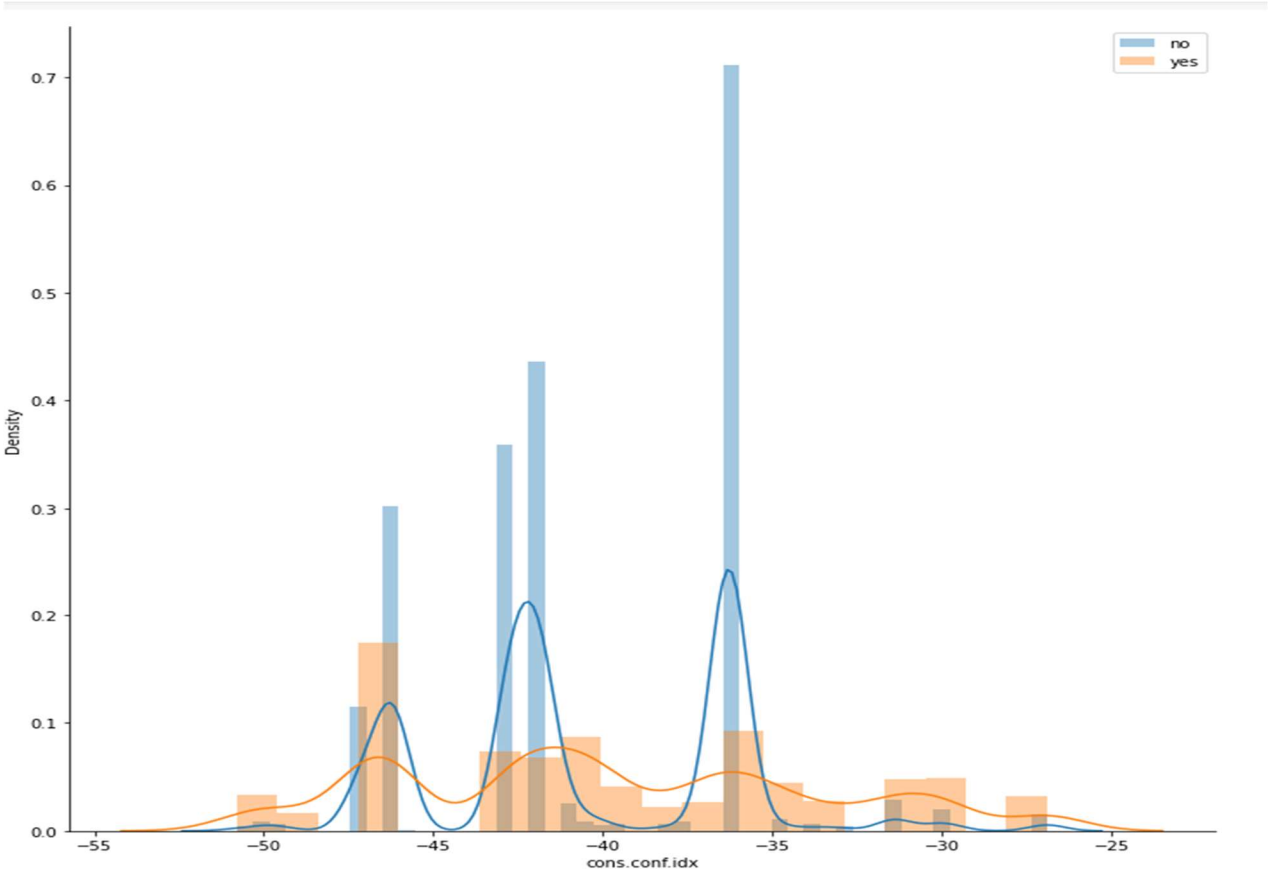
3.1.5.6. cons.price.idx

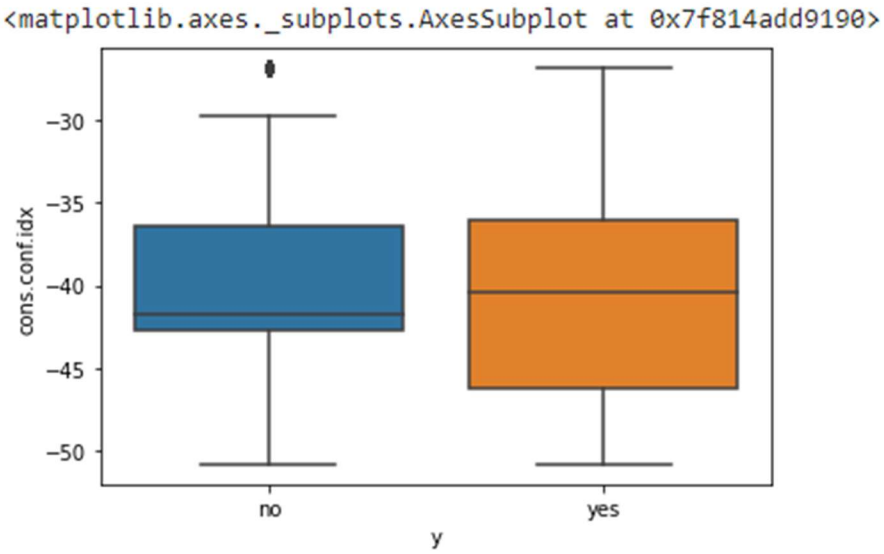




3.1.5.7. cons.conf.idx

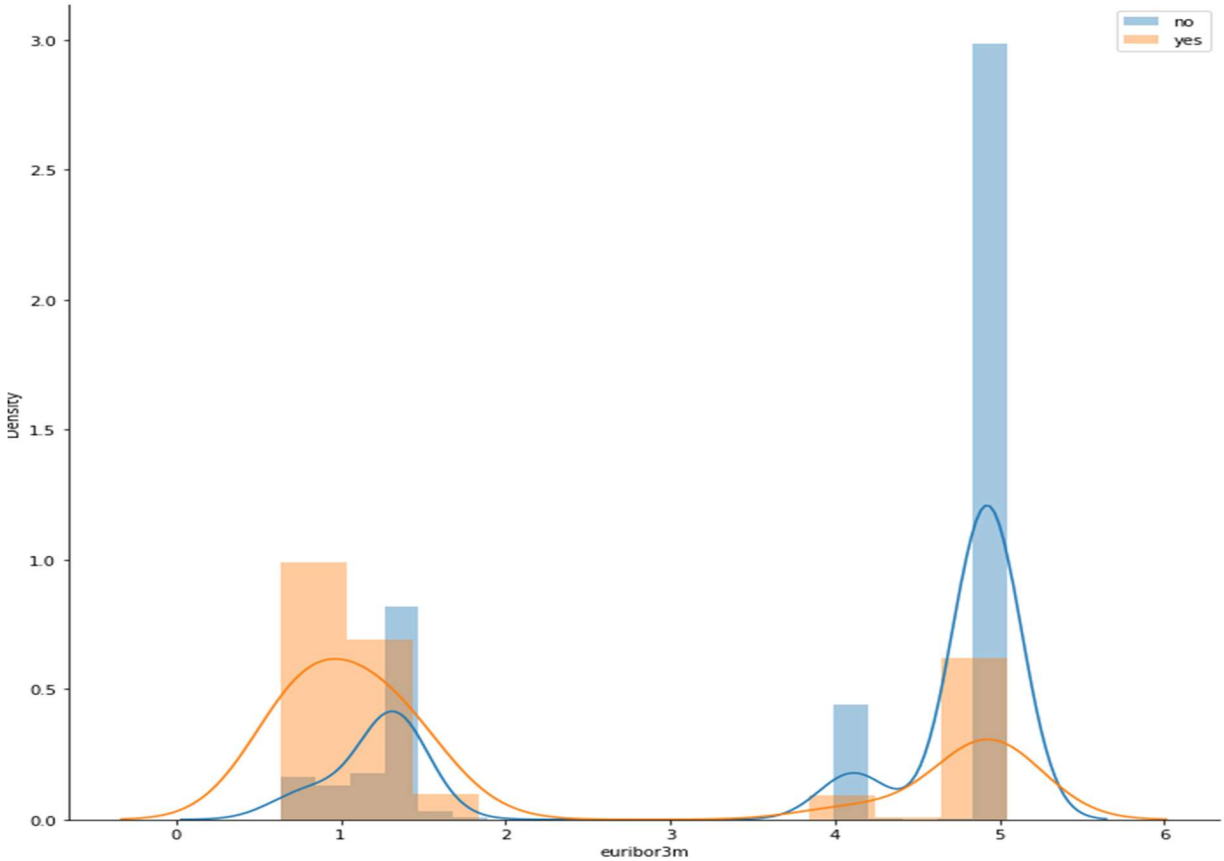
Cons.price.idx would be helpful in predicting class labels and there are no outliers present.



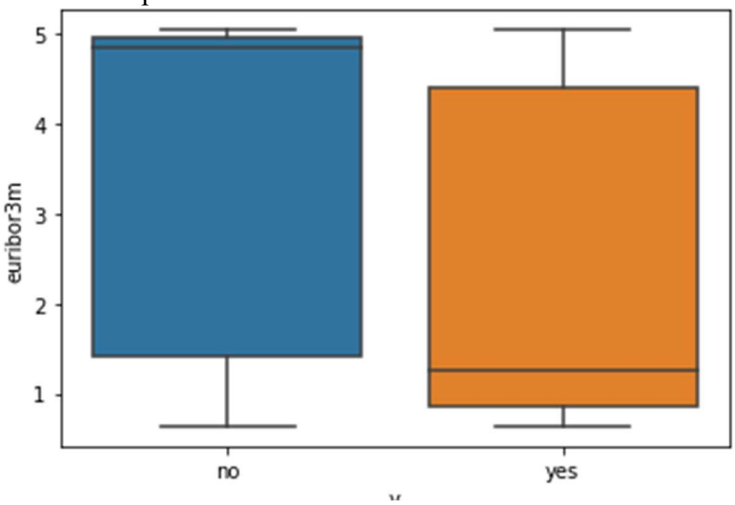


3.1.5.8. euribor3m

euribor 3 month rate, daily indicator. The euribor denotes the basic rate of interest used in lending between banks on the European Union interbank market and also used as a reference for setting the interest rate on other loans.

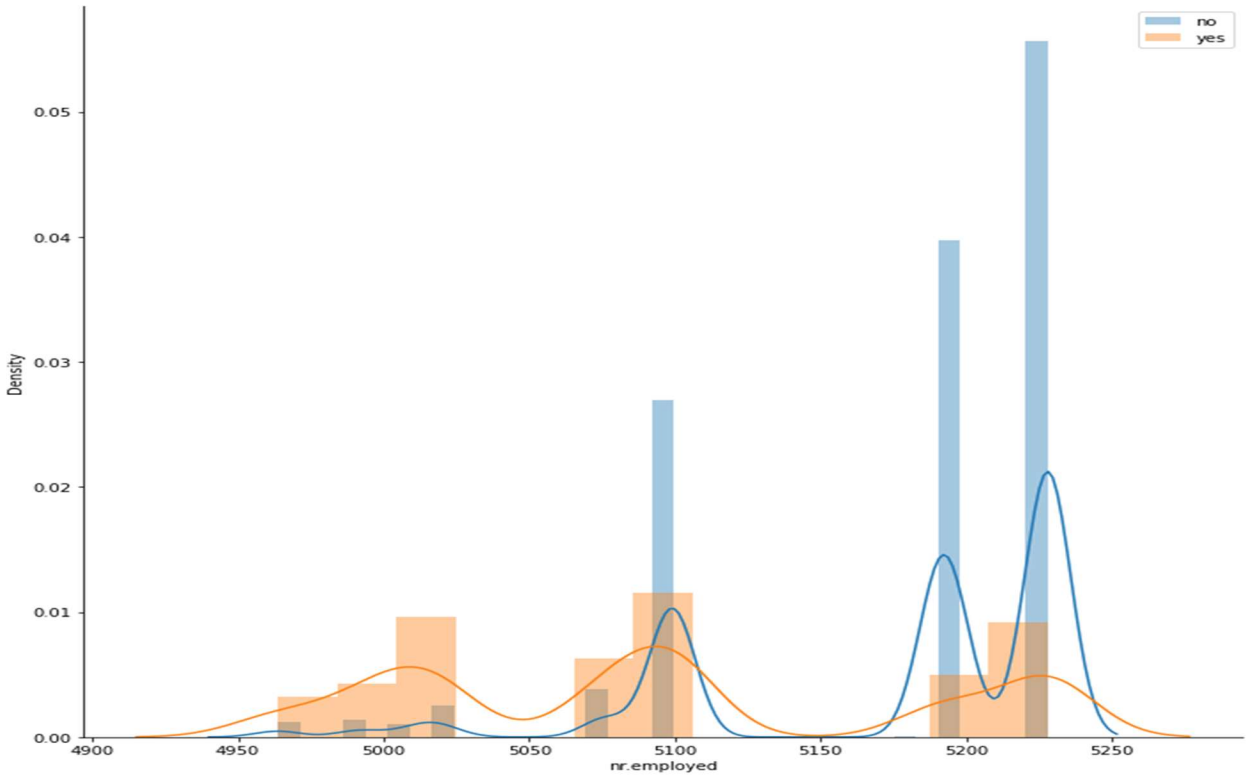


From the above plot it is visible that, Euribor3m would be helpful in predicting class labels and there are no outliers present.

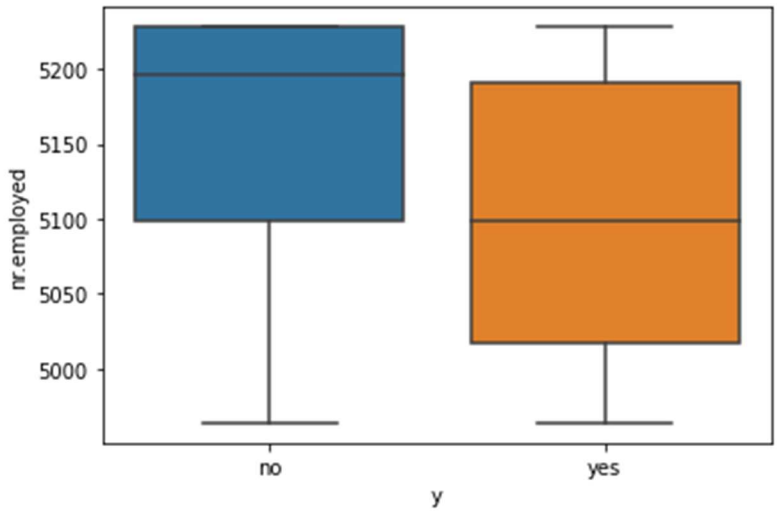


From the above plot, we can clearly see the difference in median for both the classes. This indicates that the feature can be very useful for our case study. But we can validate the assumption only by applying models and extracting feature importance.

3.1.5.9. nr.employed



From the above plot, there are no outliers present also this feature would be helpful in predicting class labels.



3.1.6. Categorical variables Analysis

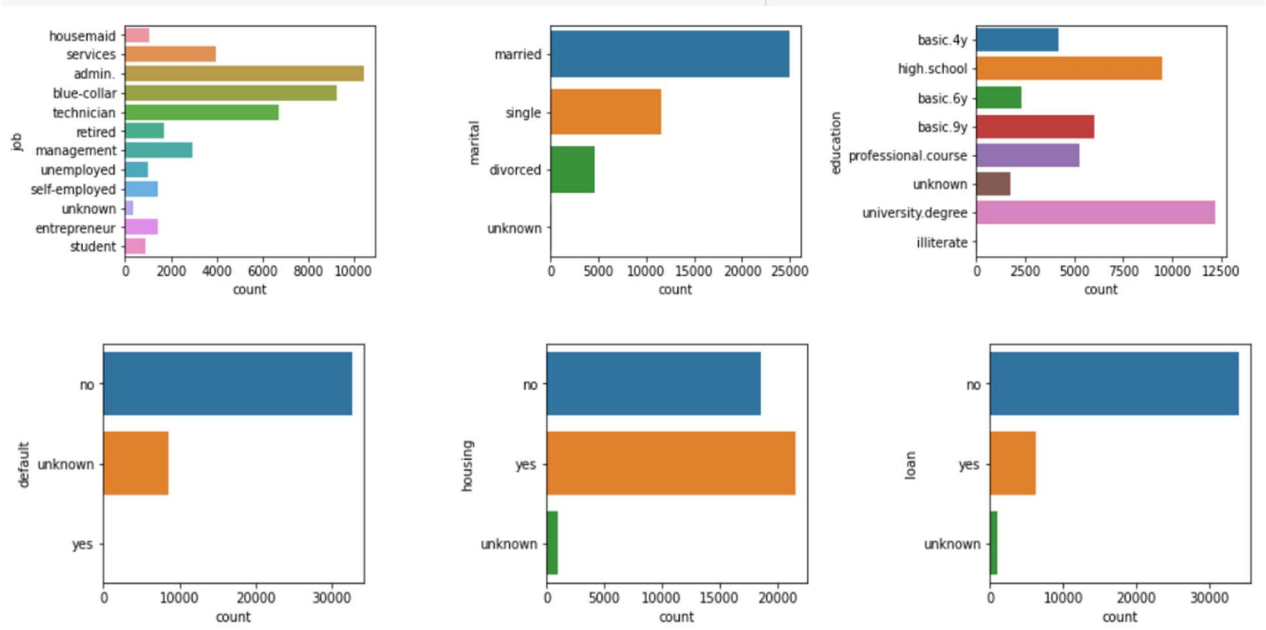
S. No.	Categorical Attributes
1	job
2	marital
3	education
4	default
5	housing
6	loan
7	contact
8	month
9	day_of_week
10	poutcome
11	y

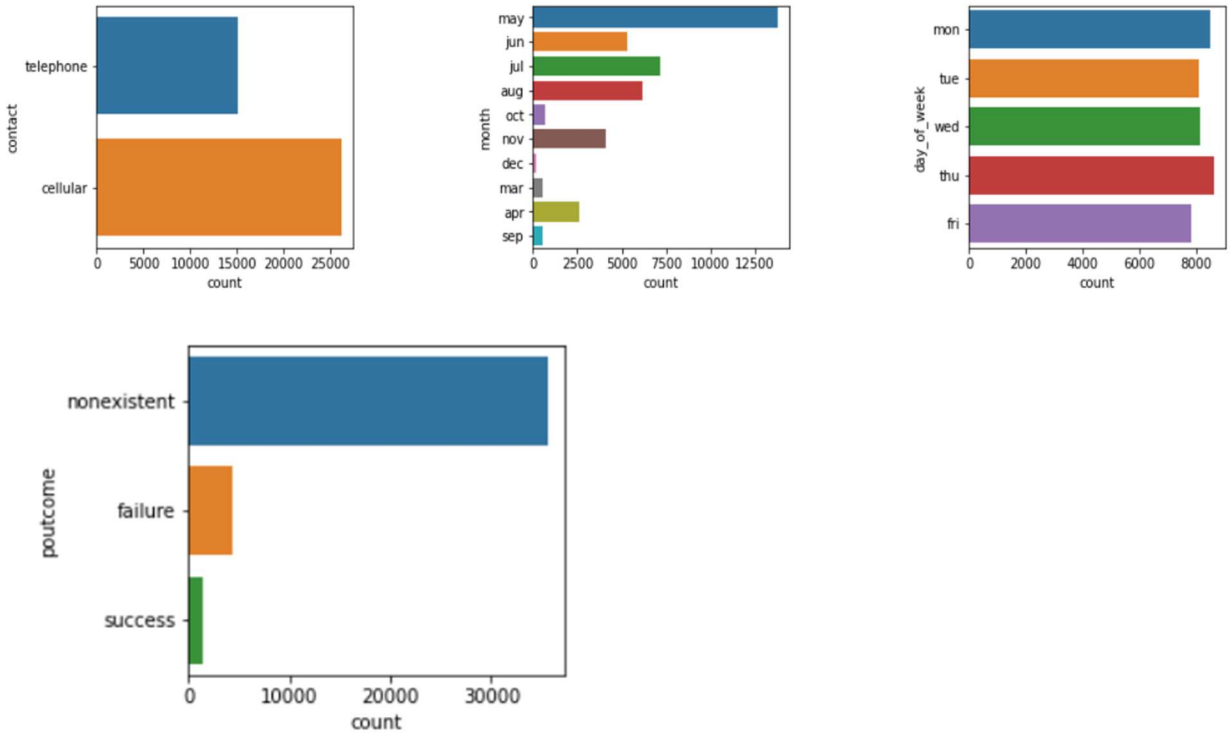
```
bank.describe(include=['object'])
```

	job	marital	education	default	housing	loan	contact	month	day_of_week	poutcome	y
count	41188	41188	41188	41188	41188	41188	41188	41188	41188	41188	41188
unique	12	4	8	3	3	3	2	10	5	3	2
top	admin.	married	university.degree	no	yes	no	cellular	may	thu	nonexistent	no
freq	10422	24928	12168	32588	21576	33950	26144	13769	8623	35563	36548

1. job: there are 12 types of jobs recordings in which the 'administrative' role is the most common with almost 10.5k of the clients
2. marital: the majority of clients are married with almost 25k records
3. education: more than 12k people have university degree
4. default: from all the 41.188 clients, 32.588 don't have any credit in default
5. housing: almost half of the customers have a housing loan
6. loan: almost 34k clients don't have any personal loans
7. contact: cellular is the most preferred choice of contact
8. month: month of may is the
9. day_of_week: All days are more or less equal so not much relevant
10. poutcome: there is no information about the outcome of any previous marketing campaign

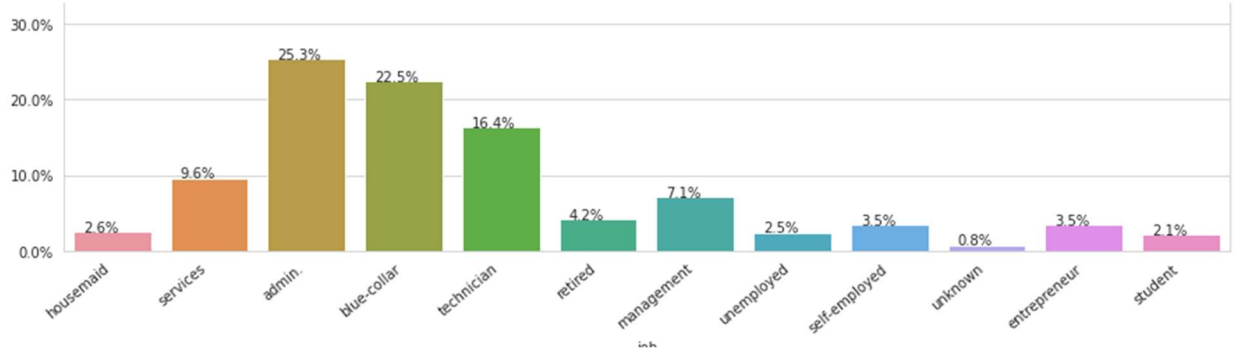
3.1.7. Visualization of Categorical dataset



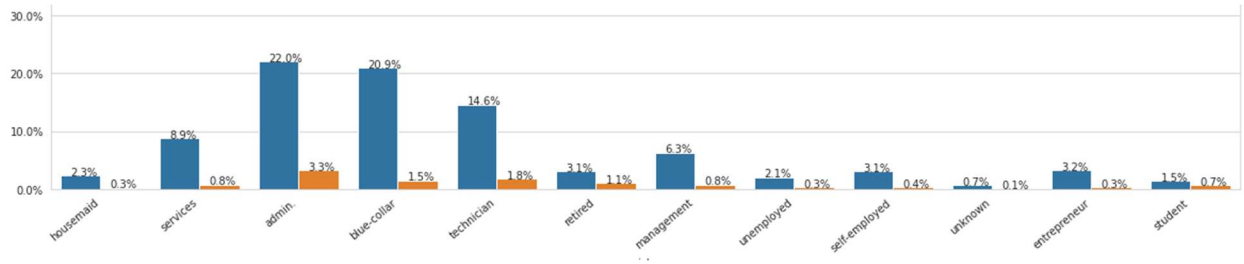


3.1.8. Visualization and analysis of the categorical Attributes

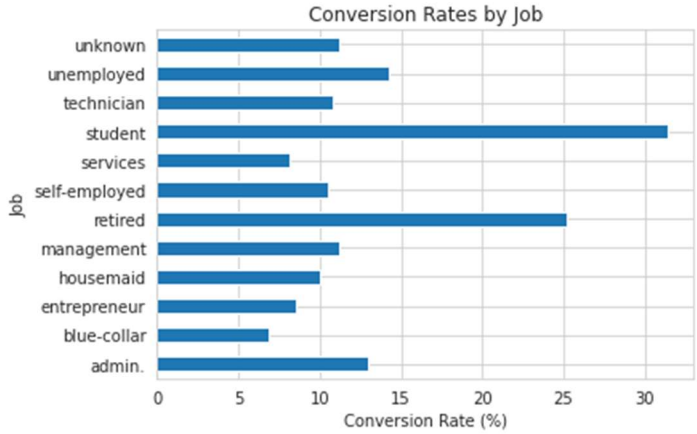
3.1.8.1. <u>Job</u>



From the above plot we can observe that people with admin jobs have been contacted more by the bank. People with unknown jobs are very few. Let’s check people with which jobs have subscribed for the deposits.



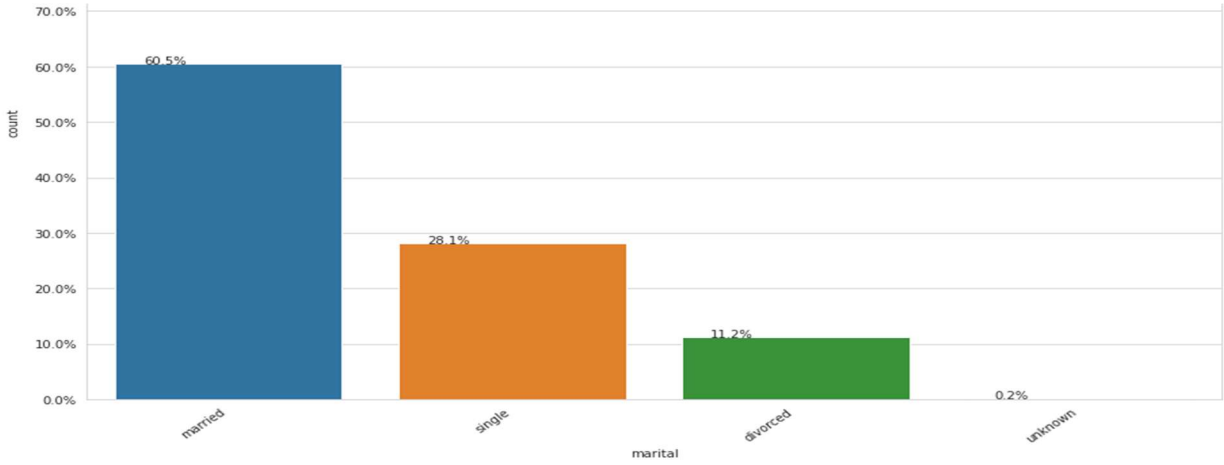
From the above plot we can observe people with admin jobs have subscribed more for the deposits than people with any other profession followed by technicians and blue collar had a higher percentage of subscription to a deposit account..

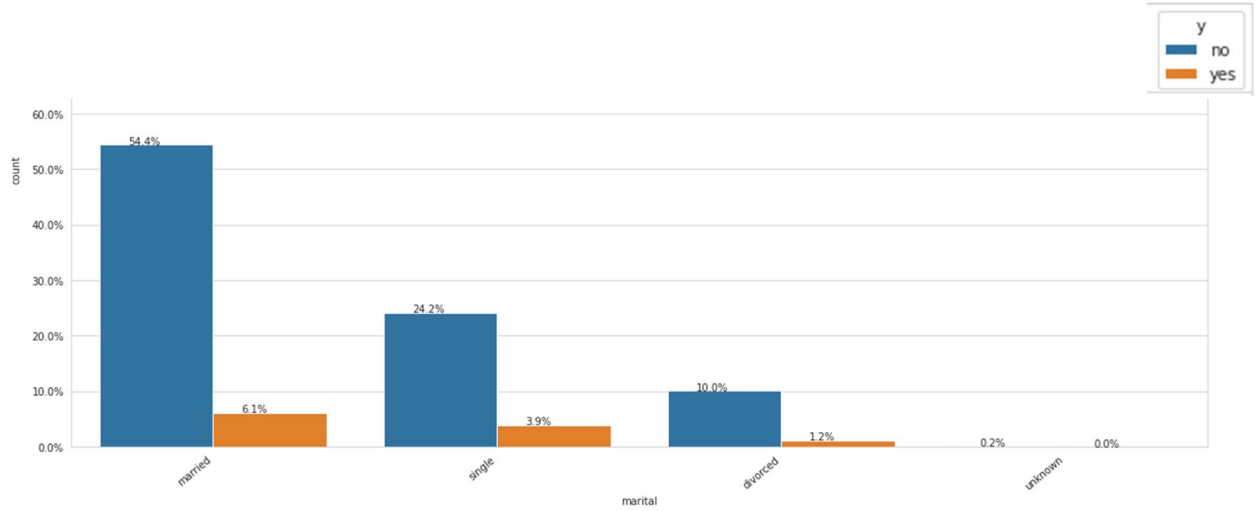


From the above plot we can observe that student have highest conversion rate followed by retired, unemployed and admin.

3.1.8.2. Marital

This simply denotes the customer’s marital status. Customer who has been contacted most are married. About 0.2% of marital status of customer is unknown

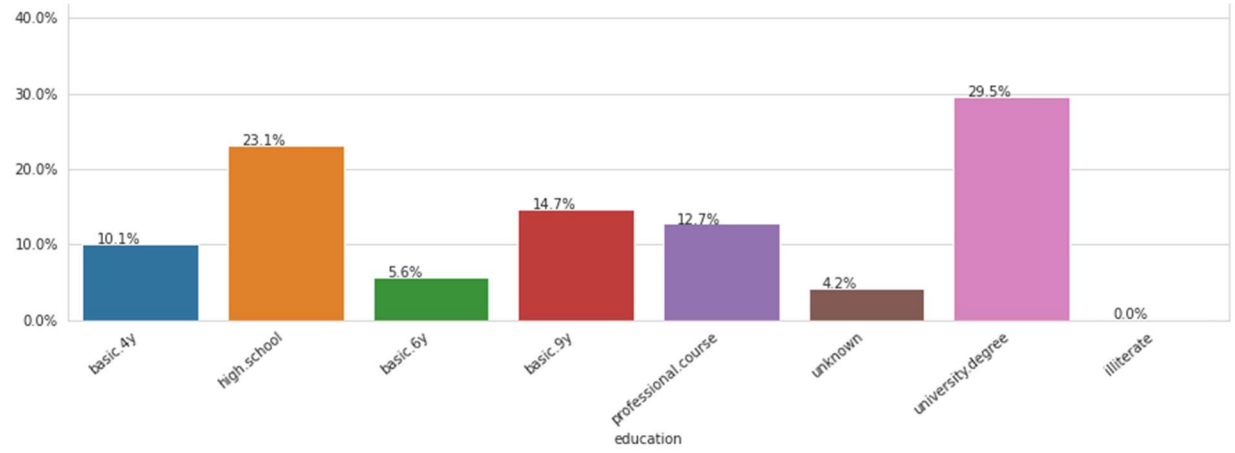


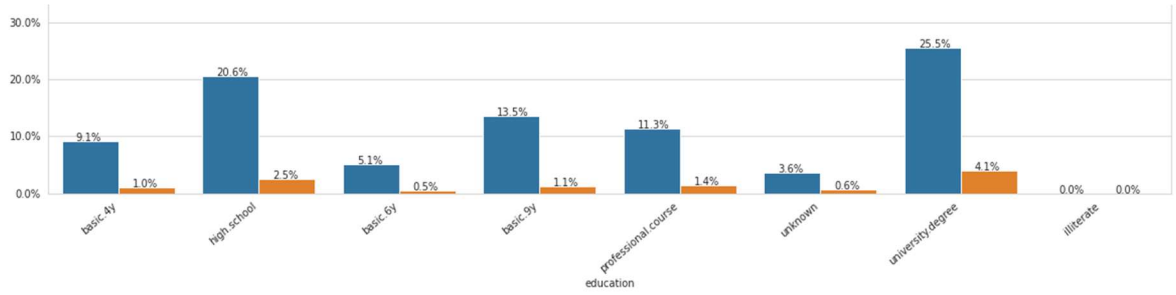


People who are married have subscribed for deposits more than people with any other marital status. They are also the most one’s who have turned down the deposits offered by the bank.

3.1.8.3. Education

People contacted by the bank with university degree as their educational qualification are more than the people with any other educational qualification followed by high school, basic 9Y and professional course. Bank has not contacted illiterate people.

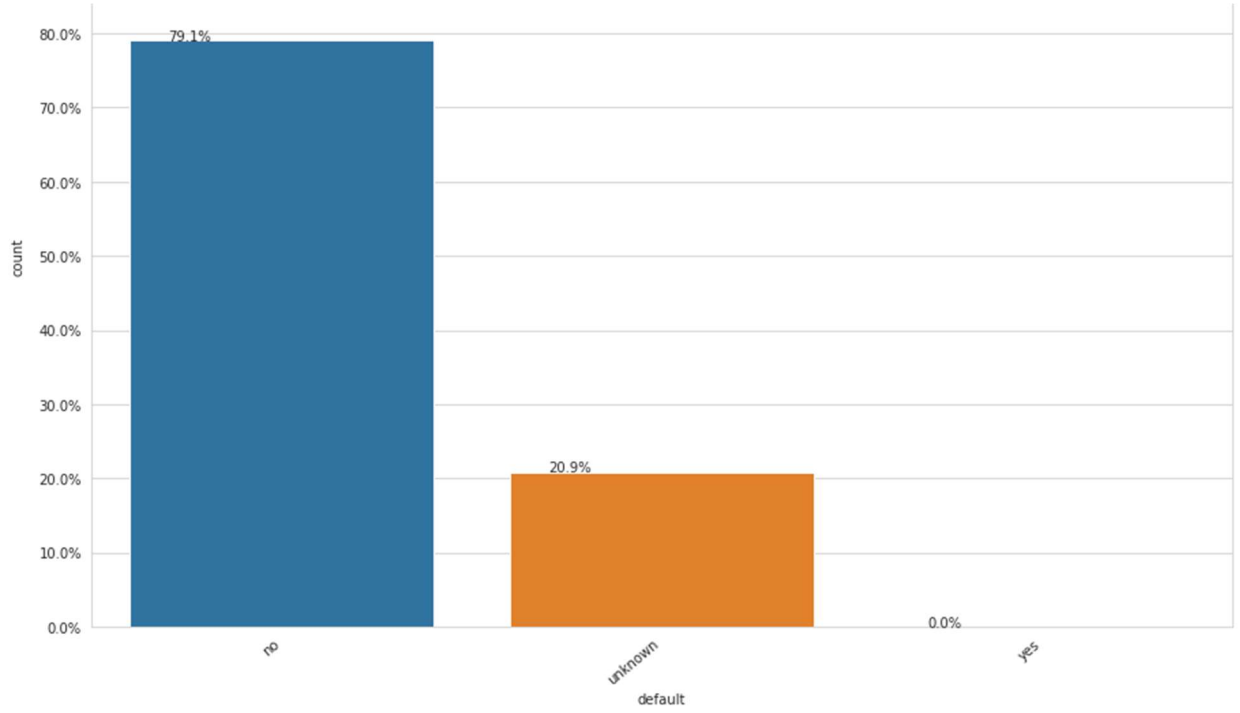




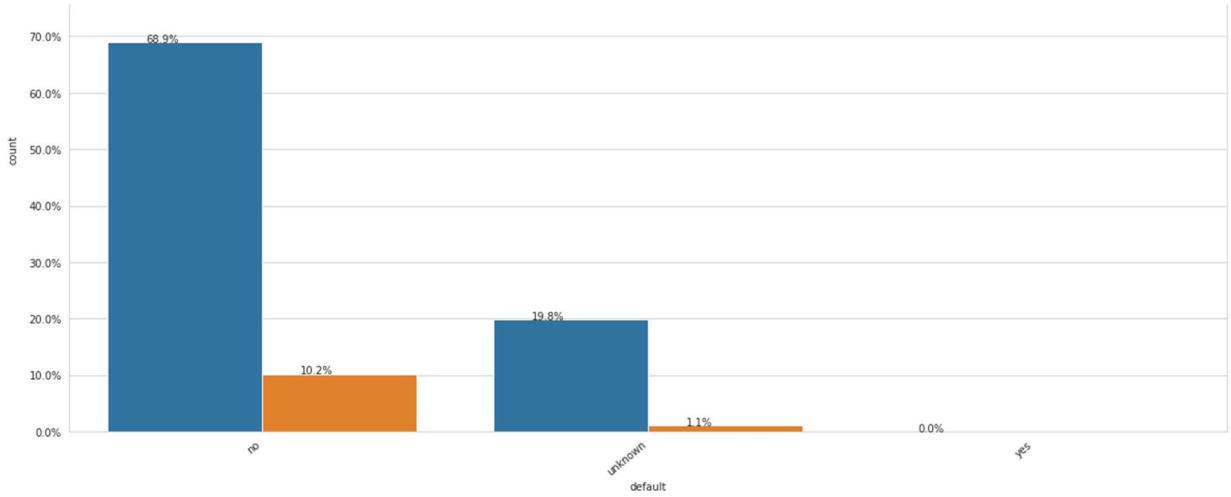
People with university degree as education qualification are the most who have subscribed for the deposits. They are also the most who have not subscribed for deposits.

3.1.8.4. Default

The category denotes if the customer has credit in default or not. The categories are yes, no and unknown. We can clearly see that the people with default status as ‘no’ are the most who have been contacted by the bank for the deposits. People with default status ‘yes’ have not been contacted by the bank at all. While very few people with unknown default status have been contacted by the bank.

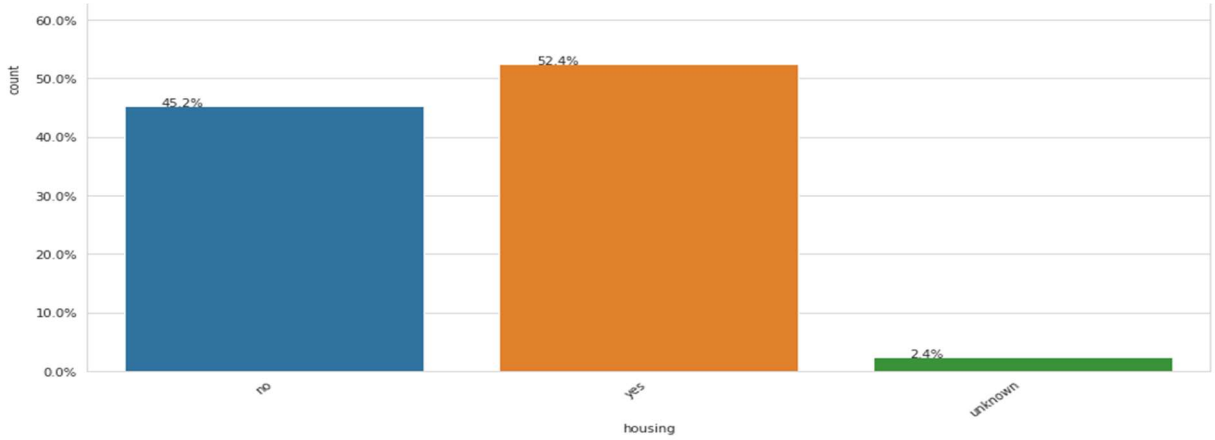


People with default status as no are the most one’s who have and have not subscribed for bank deposits.

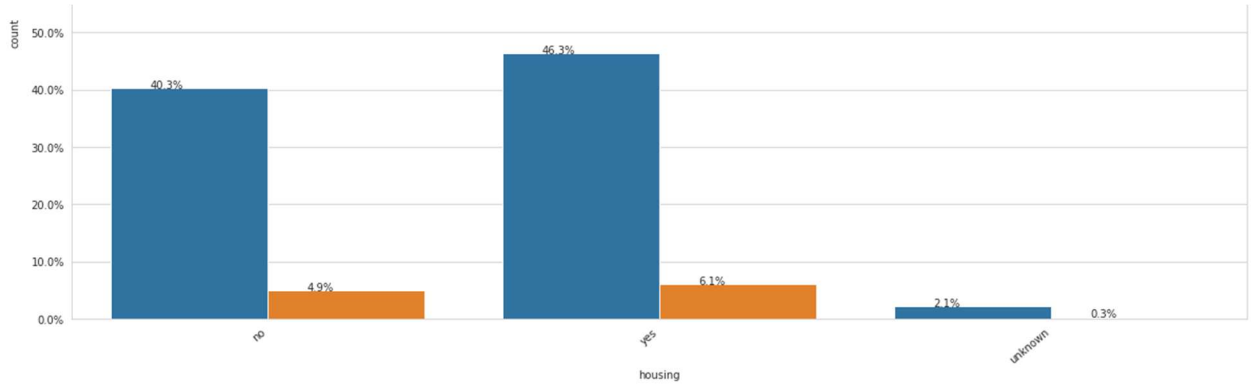


3.1.8.5. Housing

Denotes if the customer has a housing loan. Three categories are ‘no’, ’yes’, ’unknown’. housing loan has been contacted more by the bank. People who has no housing has also been contacted pretty much. People who has status unknown has been least contacted.

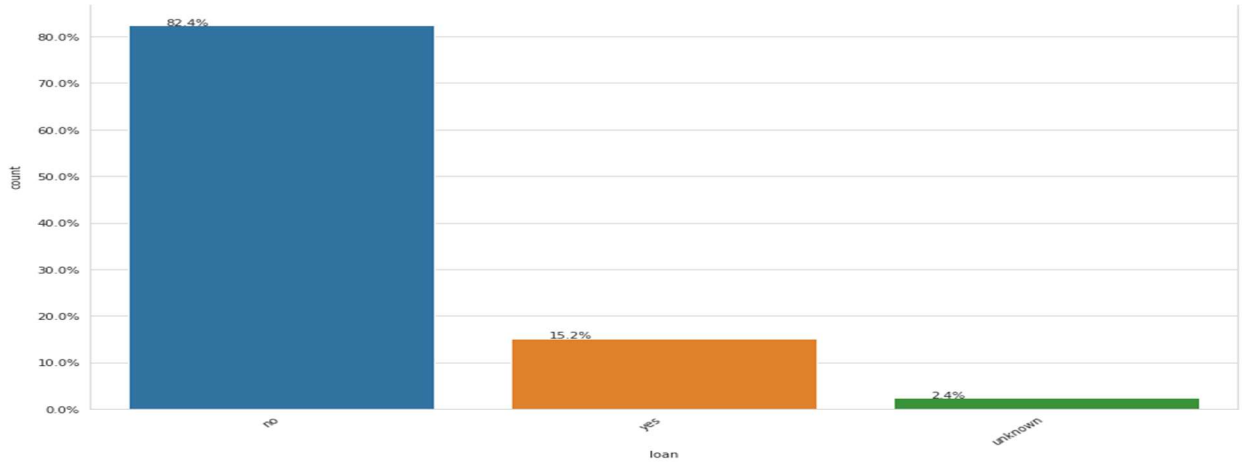


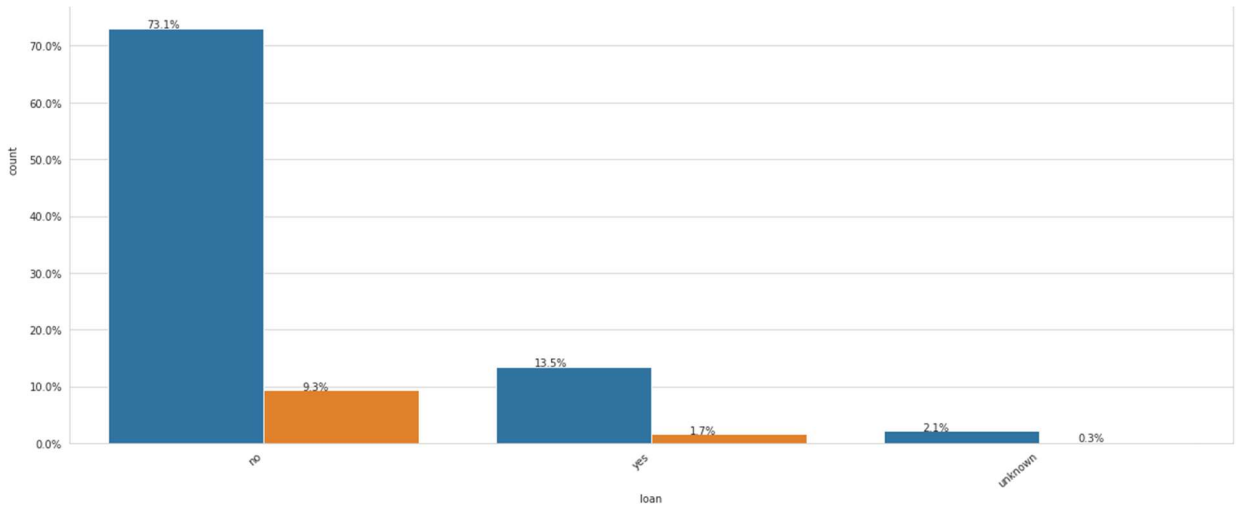
People with housing loan are the most ones who have subscribed for deposits. They are also the most ones who have not subscribed for the deposits. Very few people with unknown housing loan status have subscribed for the deposits offered by the bank.



3.1.8.6. Loan

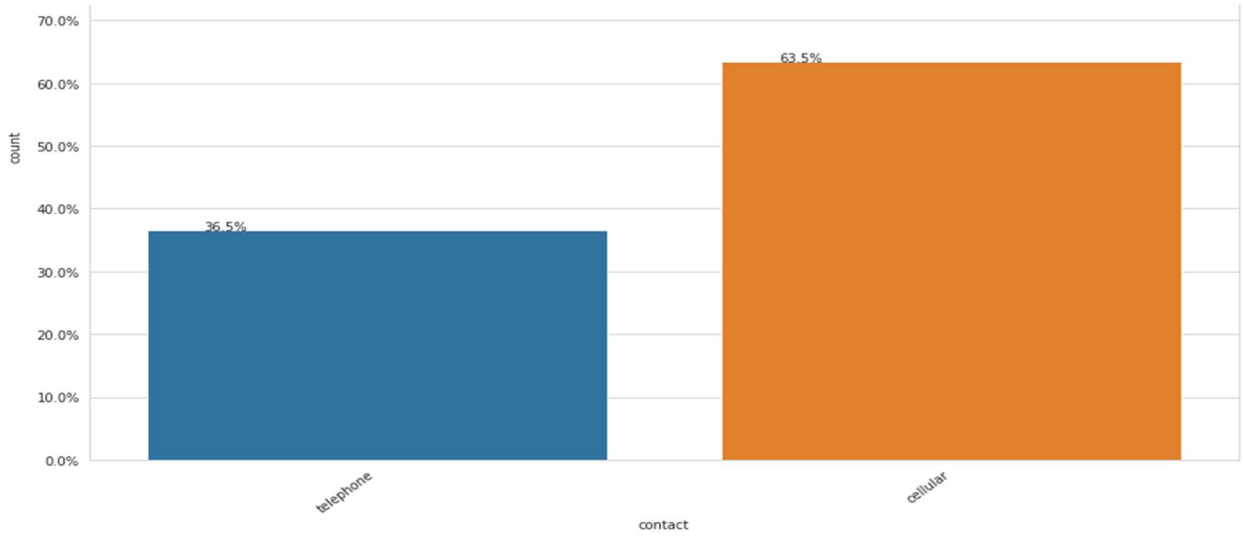
People with no personal loan are the most ones who have been contacted by the bank for the deposits. Very few people with personal loan are contacted by the bank for the deposits.

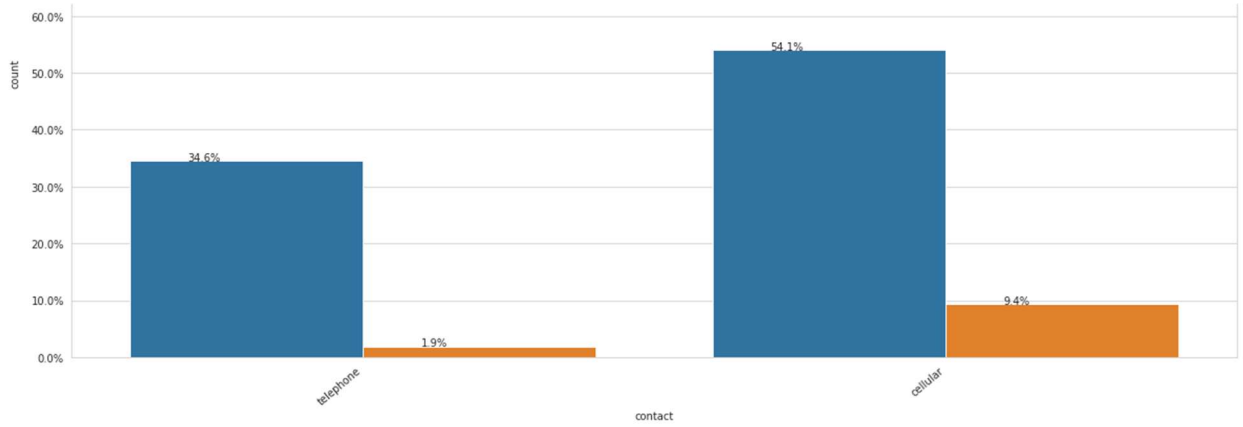




3.1.8.7. Contact

Most people are contacted more in cellular than telephone.



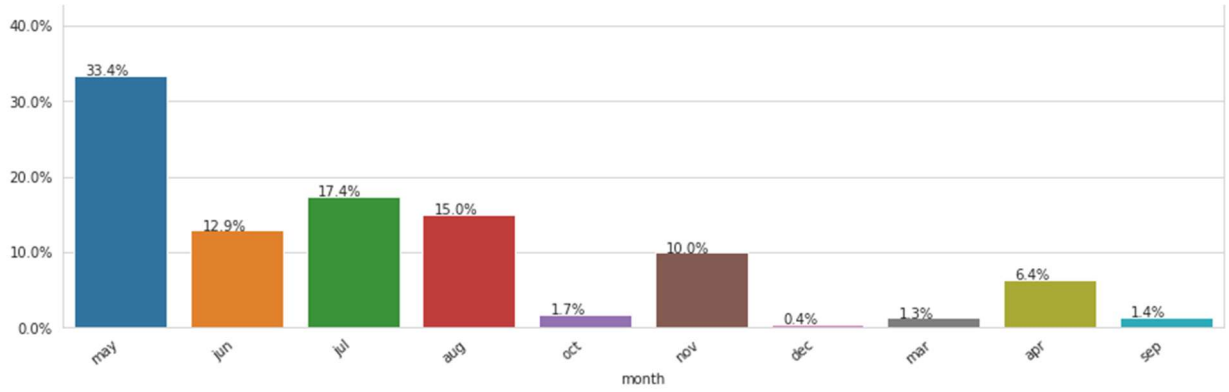


More people contacted on cellular by bank have subscribed the deposits offered by the bank than the ones contacted on telephone.

3.1.8.8. Month

People have been contacted more in the month of May, followed by July, August, June. Very few people have been contacted in the month of December. People have not been contacted

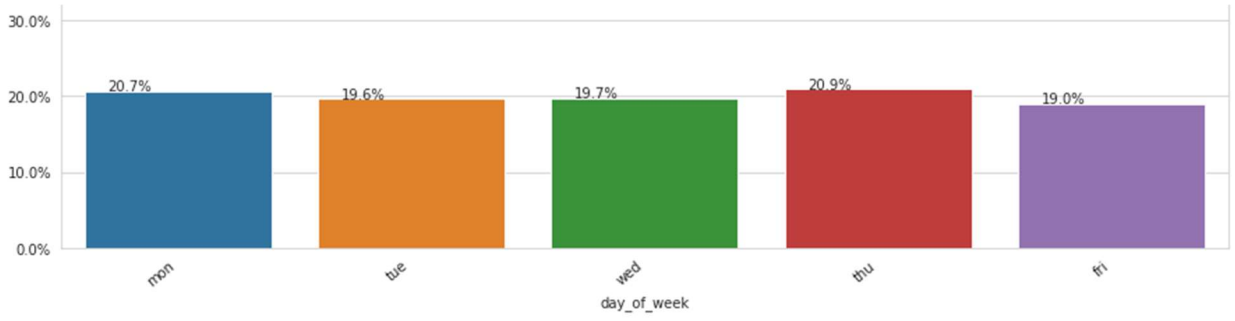
in the month of January and February.

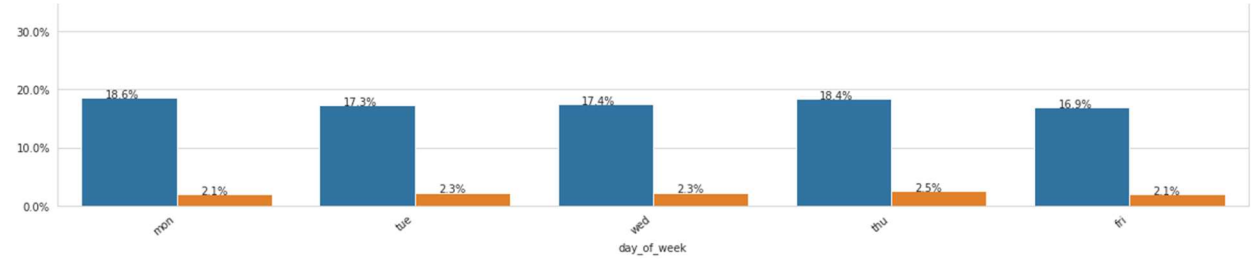


People contacted in May have higher chances to subscribe for longer term deposits but have also higher chances for not subscribing the long-term deposits. Very few people are contacted in the month of December, March, September, October and have almost equal chances for subscribing the deposits or not.

3.1.8.9. Days of the Week

People have not been contacted on Saturday and Sunday. Rest all the days, count of people contacted by the bank is almost same.

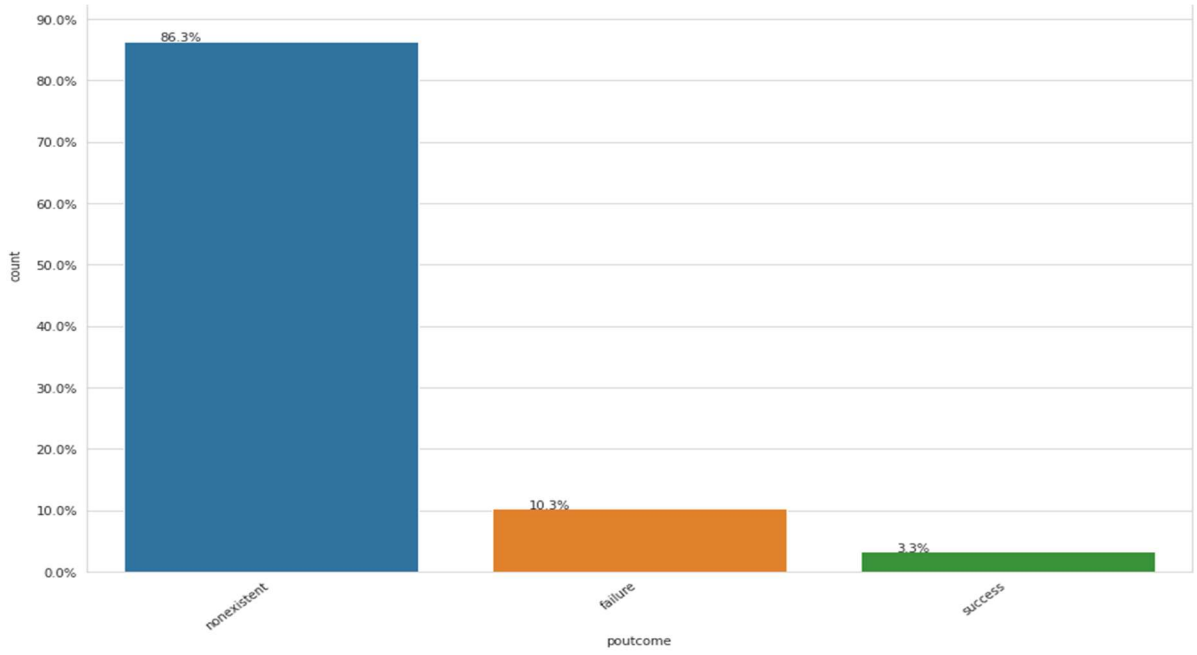


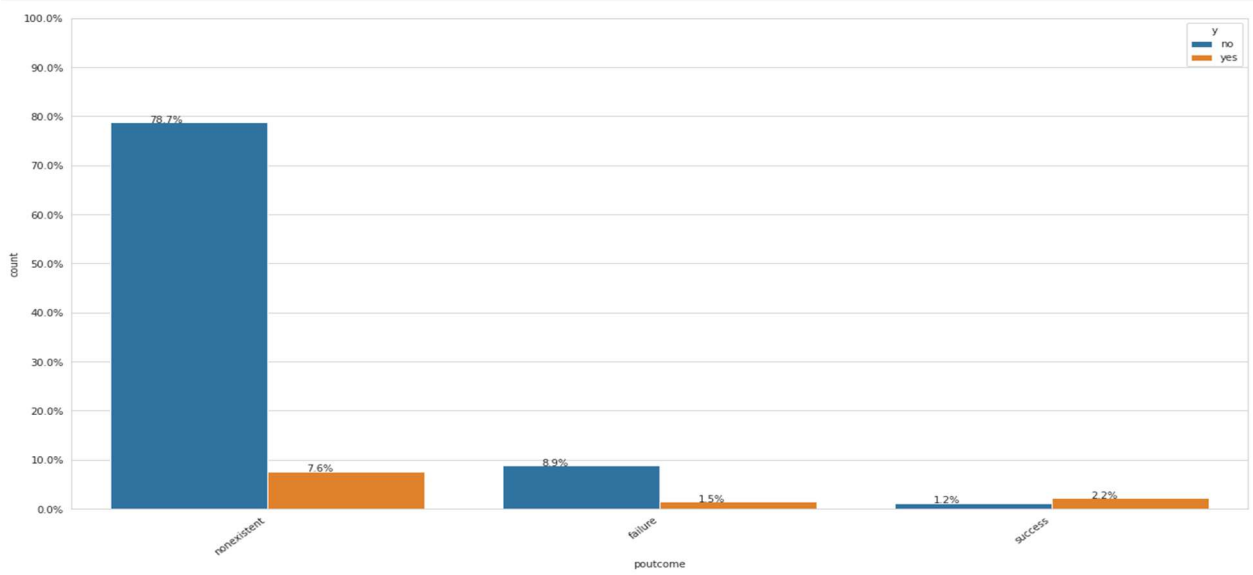


In all the days they have equal chances for subscribing and not subscribing the term deposits. Day_of_week may not be very helpful in predicting whether the customer will subscribe for long term deposits or not.

3.1.8.10. Poutcone:

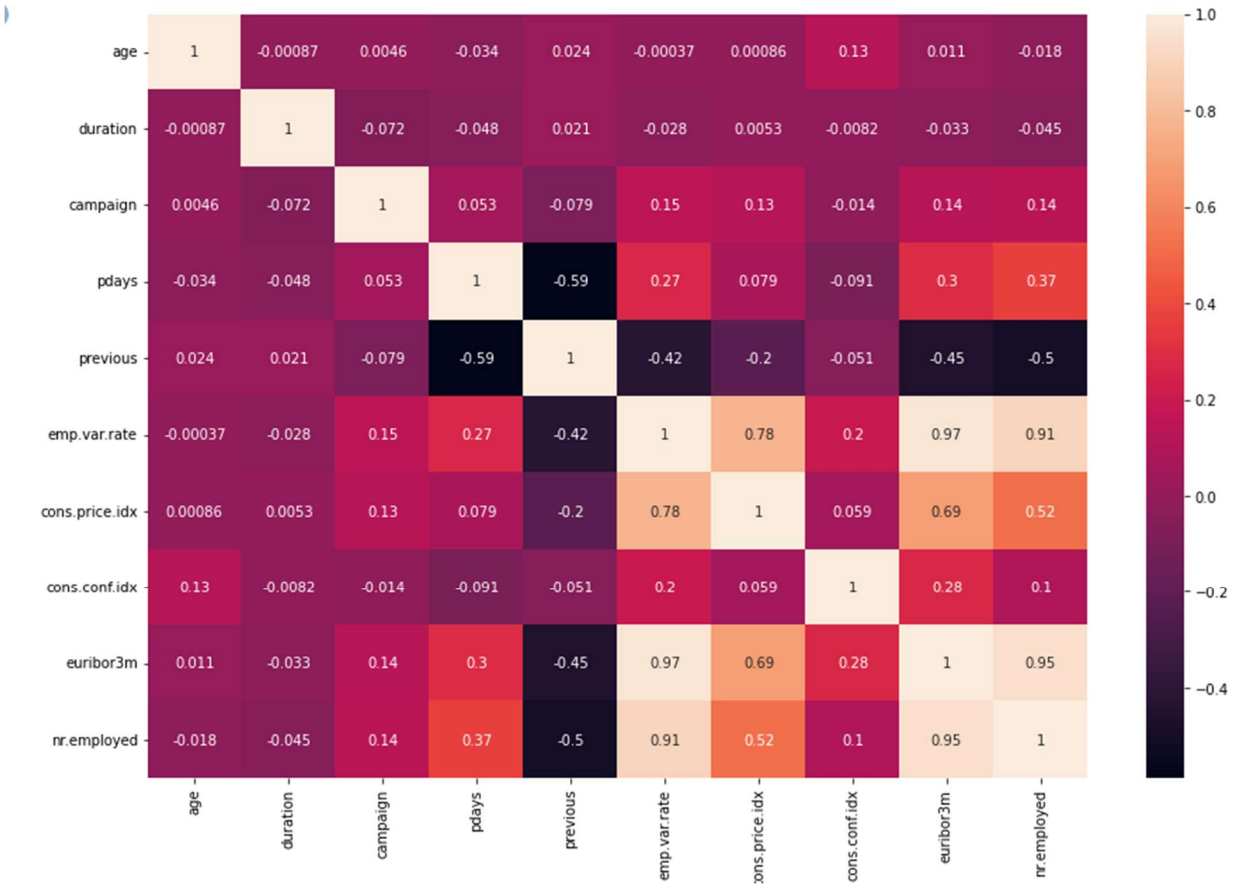
This feature denotes the outcome of the previous marketing campaign. Majority of the outcome of the previous campaign is Non-Existent. People have been contacted more in the month of May, followed by July, August, June





For most of the customers, the previous marketing campaign outcome does not exists. It means that most of the customers are new customers who have not been contacted earlier. Also one thing to note here that, for the customers who had a successful outcome from the previous campaign, majority of those customers did subscribe for a term deposit. As it has the class distribution of 2.2% for positive class, and 1.2% for negative class. From this, we can make an assumption, that this feature may hold some value in predicting the target variable. specially the poutcome_success category.

3.1.9. Correlation Matrix of Numerical Data



The emp.var.rate, euribor3m, nr.employed and cons.price.index have very high correlation. Euribor3m with nr.employed and emp.var.rate with nr.employed with the highest correlation with more than 0.9 value.

4. Preprocessing techniques to clean and prepare the data

There are no missing values in the data set, but there was 0 values found in attributes, appearing 4 times in “duration”, 15 times in “pdays”, and 35,563 times in “previous”.

Dealing with Missing and Duplicate Values

	Missing Value	Duplicate Values
No.	0	12
Action	No Action	Removed

4.1. Dropping Columns

4.1.1. Duration

According to the dataset documentation, we need to remove the 'duration' column because in real case the duration is only known after the label column is known. This problem can be considered to be 'data leakage' where predictors include data that will not be available at the time you make predictions.

4.1.2. Pdays

The attribute of “pdays”, as there is not enough information for further analysis. It is observed that 999 makes 96% of the values of the column, from attribute information 999 means the client was not previously contacted

4.1.3. Previous

I decided to drop duration . The “previous” attribute has null 35,563 from the total observation of 41,188, so I dropped this attribute off.

4.2. Outliers Treatment

Summary of Outlier

- age = there are no significant outliers, and that there are many datapoints that are outside the boxplot. Therefore, i will not be removing the datapoints that are identified here as outliers
- campaign = column has anomaly data point when compare to the other points (Try to remove it)
- pdays = Cannot remove any points, because it having lot of data points same place

- previous = we cannot see any anomaly
- df_cont = we cannot see any anomaly
- cons.conf.idx = has one outlier point and that can be visible that are outside the boxplot. Therefore, i will not be removing the datapoints that are identified here as outliers
- euribor3m = we cannot see any anomaly
- nr.employed = we cannot see any anomaly

4.3. Dealing with categorical values

There are many ways to convert categorical values into numerical values. Each approach has its own trade-offs and impact on the feature set. Hereby, I would focus on 2 main methods: One-Hot-Encoding and Label-Encoder. Both of these encoders are part of SciKit-learn library (one of the most widely used Python library) and are used to convert text or categorical data into numerical data which the model expects and perform better with.

Label Encoding: This approach is very simple and it involves converting each value in a column to a number in a sequence. But depending upon the data values and type of data, label encoding induces a new problem since it uses number sequencing. The problem using the number is that they introduce relation/comparison between them.

One-Hot Encoder: Though label encoding is straight but it has the disadvantage that the numeric values can be misinterpreted by algorithms as having some sort of hierarchy/order in them. This ordering issue is addressed in another common alternative approach called ‘One-Hot Encoding’. In this strategy, each category value is converted into a new column and assigned a 1 or 0 (notation for true/false) value to the column. Though this approach eliminates the hierarchy/order issues but does have the downside of adding more columns to the data set. It can cause the number of columns to expand greatly if you have many unique values in a category column.

We have 11 Category attributes, using one-hot encoder will increase number of columns to more than 50 and considering the size of data I will be using Label encoding. However to eliminate hierarchy/order issue I will perform label encoding to only selected columns where relationship issue do no occur

4.4. Train Test Split

For the experiments, an 80:20 ratio of the data set was chosen, i.e., the training data uses 80% of full data set, and 20% is used for testing. Below is the overall statistics for the division among the train vs test sets

```
Shape of training feature: (32940, 14)
Shape of testing feature: (8235, 14)
Shape of training label: (32940,)
Shape of testing label: (8235,)
```

4.5. Standardization of Data

Data standardization is the way of the rescaling one or multiple features so that they can have a mean value of 0 and a standard deviation of 1. Standardization assumes that your data has a Gaussian (bell curve) distribution. but it is not strictly have to be true, but it is the technique which is considered as more effective if your feature values distribution is belongs Gaussian. Why to standardize before fitting a ML model? Well, the idea is simple. Variables that are measured at different scales do not contribute equally to the model fitting & model learned function and might end up creating a bias. Thus, to deal with this potential problem feature-wise standardized ($\mu=0$, $\sigma=1$) is usually used prior to model fitting (Loukas, S. 2020, June 10)..

To do that using scikit-learn, we first need to construct an input array X containing the features and samples with X.shape being[number_of_samples, number_of_features] .

Keep in mind that all scikit-learn machine learning (ML) functions expect as input an NumPy array X with that shape i.e. the rows are the samples and the columns are the features/variables. Having said that, let’s assume that we have a matrix X where each row/line is a sample/observation and each column is a variable/feature.

Standardization:

$$z = \frac{x - \mu}{\sigma}$$

with mean:

$$\mu = \frac{1}{N}\sum_{i=1}^{N}(x_i)$$

and standard deviation

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

StandardScaler is the industry’s go-to algorithm. StandardScaler standardizes a feature by subtracting the mean and then scaling to unit variance. Unit variance means dividing all the values by the standard deviation. StandardScaler does not meet the strict definition of scale I introduced earlier (Loukas, S., 2020, June 10).

StandardScaler results in a distribution with a standard deviation equal to 1. The variance is equal to 1 also, because variance = standard deviation squared. And 1 squared = 1.

StandardScaler makes the mean of the distribution approximately 0.

I used Standard Scaler however there was no effect on the algorithms.

Normalization which is Min-Max Scalar technic, will not work as this data doesn't need to suppress outliers, as outliers are already dropped.

4.6. SMOTE for Class Imbalance

Class imbalance, where the number of positive samples is significantly less than the number of negative samples, is a common problem in data science. A typical machine learning algorithm works best when the number of instances of each class is roughly equal. Problems can appear when the number of instances of one class greatly exceeds the other.

Since only 11.6% of our data set has the positive response, the imbalance has to be handled. Thus, resampling techniques are used to improve the classification accuracy

The class imbalance influences the models for its disproportionate number of different class instances in practice. Thus, to deal with this there are several ways such as cost functions and

sampling. For this experiment Synthetic Minority Over-sampling Technique (SMOTE) resampling techniques is used to get a higher classification accuracy. SMOTE is an over sampling technique. For binary classification, SMOTE sampling has proven in the past to be a good choice. SMOTE should only be used to augment training data. Test dataset should remain untouched. Applying SMOTE to the entire dataset will result in data leakage.

Before SMOTE the shape of y_train was for 0 class “no” 16,270 and 1 class “yes” 1,595

After SMOTE the shape of y_train was for 0 class “no” 16,270 and 1 class “yes” 16,270

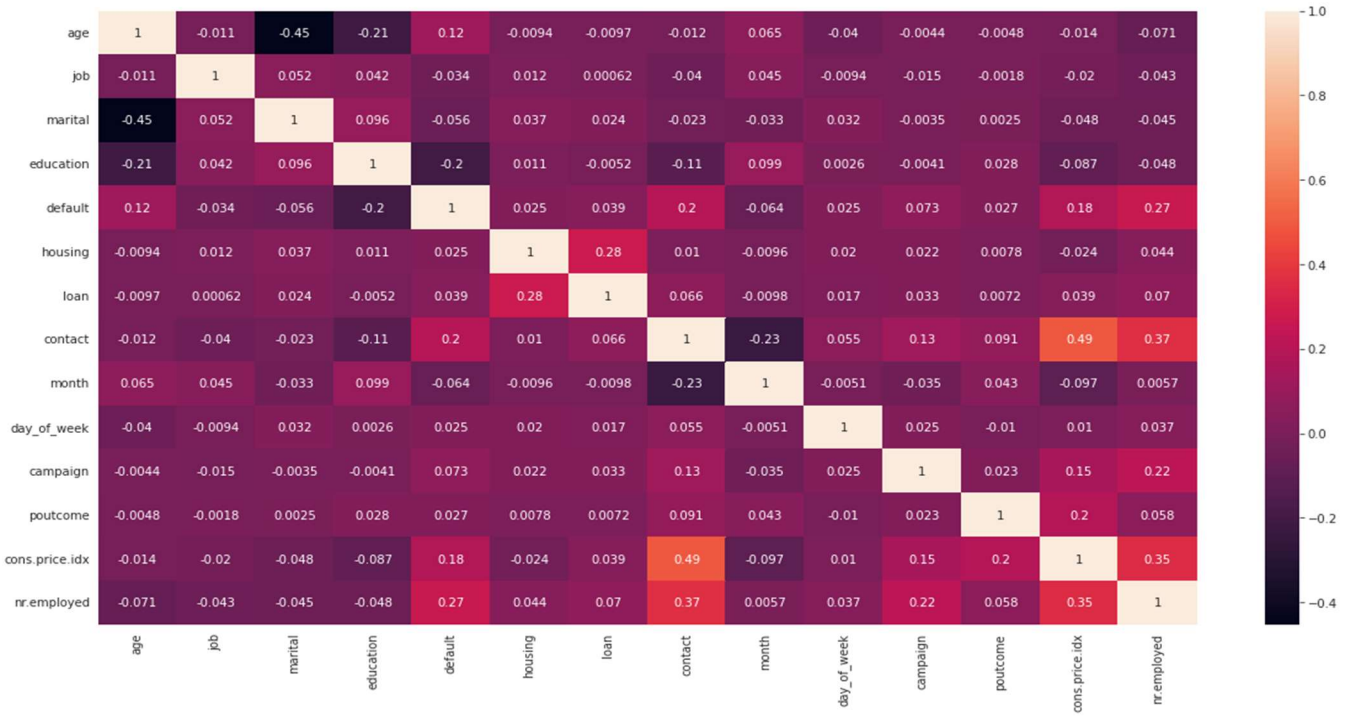
4.7. Correlation Matrix after Standardization & Class Imbalance

The Correlation matrix is an important data analysis metric. It is computed to summarize data to understand the relationship between various variables and make decisions accordingly. Pearson correlation coefficient, is a measure of the linear association between two variables. It has a value between -1 and 1 where:

- -1 indicates a perfectly negative linear correlation between two variables
- 0 indicates no linear correlation between two variables
- 1 indicates a perfectly positive linear correlation between two variables

The further away the correlation coefficient is from zero, the stronger the relationship between the two variables

Correlation Matrix of Numerical Value



No significant correlation exist after Standardization & Class Imbalance (SMOTE) after data processing, and removing columns with a high correlation to prevent Multicollinearity.

5. Modeling

When a set of elements is divided in the two groups then the process is called a Binary Classification. The data set has two labels for the output variable, Yes and No. Thus, Binary Classification algorithms have to be used.

There are may popular Binary Classification algorithms, for this project five classification algorithms have been selected to predict future subscription:

- Logistic Regression (LR),
- Random Forest (RF),
- Multi-Layer Perceptron (MPL),
- Support Vector Machine (SVM),
- XG Boost (XGB)

Logistic Regression is a classification algorithm used to find the probability of event success and event failure. It is used when the dependent variable is binary (0/1) in nature. It supports categorizing data into discrete classes by studying the relationship from a given set of labelled data. It learns a linear relationship from the given dataset and then introduces a nonlinearity in the form of the Sigmoid function or also known as the ‘logistic function’ instead of a linear function.

The **Random Forest** algorithm is an ensemble method used mainly for classification and regression. Random Forests grow a multitude of decision trees. Each tree gives a classification, and “votes” for that class, after which the classification with the most votes is selected from all the trees within the “forest”. Random Forests do not over fit as decision trees and are able to balance error in classification caused by imbalanced data sets. RF works well with both categorical and continuous variables since no feature scaling is required. Likewise, it handles non-linear parameters efficiently, algorithm is very stable. Random Forest is comparatively less impacted by noise on the other hand it complex and requires much more computational power and resources.

A **multilayer perceptron (MLP)** is a neural network connecting multiple layers in a directed graph, which means that the paths connecting nodes in layers only go one way. Each node, apart from the input nodes, has a nonlinear activation function. An MLP consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function. An MLP uses backpropagation as a supervised learning technique so that the error value can be updated in a much successful manner when manner, considering what the model has already learned (*What is a Multilayer Perceptron (MLP)? - Definition from Techopedia*. (n.d.)).

A **support vector machine (SVM)** is machine learning algorithm that analyzes data for classification and regression analysis. SVM is a supervised learning method that looks at data and sorts it into one of two categories. It is trained with a series of data already classified into two categories, building the model as it is initially trained. The task of an SVM algorithm is to determine which category a new data point belongs in. This makes SVM a kind of non-binary linear classifier. An SVM outputs a map of the sorted data with the margins between the two as far apart as possible. SVMs are used in text categorization, image classification, handwriting recognition and in the sciences (*What is a Support Vector Machine (SVM)? - Definition from Techopedia. (2019)*.

XGBoost (Extreme Gradient Boosting Decision Tree) is a common tool for creating machine learning models for classification and regression**.** XGBoost is an open source software library implementation of the Gradient Boosting Machine Learning algorithm. Gradient Boosting is an optimization algorithm whereby the optimization is based on a differentiable and/or loss function. It is actually an ensemble form of the weak prediction model. Generally for Machine Learning purposes, Decision Tree is used as a weak model and thus is used for the Gradient Boosting Algorithm. XGBoost has a feature of Regularization which prevents overfitting **XGBoost** (Extreme Gradient Boosting Decision Tree) is a common tool for creating machine learning models for classification and regression and also utilizes parallel processing. This algorithm implementation use tree pruning, so in many cases it gives better performance by reducing unwanted processing steps. In addition, it also has the capability to deal with missing values (Kumar, N. 2019, March 9).

5.1. Models Performance Measurement:

For the measurements, Precision, Recall, F1-Score, Accuracy and AUC were used. Precision and Recall is important if the target is to measure the positive class. For example, in this experiment, the number of yes among client was used, which is the positive class. Before explaining these parameters, some relevant terms TP, FP, TN, FN are introduced. True Positive (TP) is number of positive samples correctly classified. False Positive (FP) is number of positive samples incorrectly classified. True Negative (TN) is number of negative examples correctly classified, and False Negative (FN) is the number of negative samples wrongly classified.

True positive (TP): Predicting positive class as positive (ok) - indicates that the outcome of the model or predicted value matches that of the actual value.

True negative (TN): Predicting negative class as negative (ok) - also indicates how well the model is performing.

False positive (FP): Predicting negative class as positive (not ok) - measures a mismatch between the actual value and the predicted value by the model. Also known as type I error.

False negative (FN): Predicting positive class as negative (not ok) - measures a mismatch between the actual value and the predicted value by the model. Also known as type II error.

Receiver Operating Characteristic Curve (ROC Curve) is a curve in a graph with the value of TPR versus FPR at different classification thresholds. The area under the ROC curve is

called AUC. It is used to measure how well the predictions are ranked by means of threshold. It also measures the quality of the model's predictions irrespective of what classification threshold is chosen. The value of the threshold is generally between 0 and 1. The larger the threshold the better the prediction. But with respect to the increase of threshold, not only the True Positives should be measured but also the False Positives. Thus, it is a trade-off.

Accuracy is the number of correctly predicted data points out of all the data points. More formally, it is defined as the number of true positives and true negatives divided by the number of true positives, true negatives, false positives, and false negatives.

The number of correctly predicted data out of all is called the Accuracy measure of the model:

$$\textbf{Accuracy} = (\ TP + TN\ )/(\ TP + FP + TN + FN\ )$$

This equation tells how well the model would predict the outcome.

All the measures here uses a value between 0 to 1, where 1 is best score.

A confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm. Each row of the matrix represents the instances in an actual class while each column represents the instances in a predicted class. To measure how well the algorithms are performing and predicting binary class, I applied evaluation matrix 2x2, which shows the correct and incorrect (i.e. true or false) predictions on each class.

Confusion matrix is used to calculate precision and recall. It is not possible to maximize both precision and recall because there is a trade-off between them. Increasing precision decreases recall. Below I will discuss which one is more important for this dataset.

Precision measures how good our model is when the prediction is positive. The focus of precision is positive predictions. It indicates how many positive predictions are true. Precision can be seen as a measure of quality, and recall as a measure of quantity. Higher precision means that an algorithm returns more relevant results than irrelevant ones, and high recall means that an algorithm returns most of the relevant results (whether or not irrelevant ones are also returned). Precision is a good evaluation metric to use when the cost of a false positive is very high and the cost of a false negative is low. FP means, in bank marketing complain, that bank employees will contact clients

that are predicted as 1 (“yes”) class but actually, they are class 0 (“no”), and ask them to subscribe to term deposit, so it is inconvenient. But in this case, it is not as high cost as, for example, telling people they were sick when they were not.

Precision is the positive predictive rate:

$$\textbf{Precision} = TP/(TP+FP) \quad (1)$$

Recall measures how good our model is at correctly predicting positive classes. The focus of recall is actual positive classes. It indicates how many of the positive classes the model is able to predict correctly. Recall calculates the percentage of actual positives a model correctly identified (True Positive). When the cost of a false negative is high, you should use recall. FN means, in bank marketing complain, that bank employees will not contact clients that are predicted as 0 (‘no”) class but actually they are class 1 (“yes”). They could subscribe to term deposit but no one contacted them so the bank will lose sales. We can assume that, it is going to be a high cost for the bank to lose potential subscription but on the other hand to bother customers with inconvenienced calls are costly as well. So there is another measure that combines precision and recall into a single number and that is the F1 score. F1 score represents the harmonic mean of the precision and recall. It is a more useful measure than accuracy for problems with uneven class distribution because it takes into account both false positive and false negatives.

Recall is the true positive rate:

$$\textbf{Recall} = TP/(TP+FN)$$

F-Measure or F1 score is a measure of test accuracy. It is accomplished by the weighted harmonic mean:

$$\textbf{F1 Score} = 2 * (\text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

F1 is more effective for accuracy when the data set has imbalance classes, and there is a need to measure the accuracy of minority class.

For a high F1-score, both precision and recall must be high. Prediction of prospect customer response “YES” or “NO” to open a term deposit account.

AUC - ROC Curve is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. The higher the AUC, the better the model is at predicting 0 classes as 0 and 1 classes as 1. The ROC curve is plotted with TPR against the FPR where TPR is on the y-axis and FPR is on the x-axis. When AUC is approximately 0.5, the model has no discrimination capacity to distinguish between positive class and negative class. When AUC is approximately 0, the model is predicting a negative class as a positive class.

Sensitivity is the metric that evaluates a model’s ability to predict true positives of each available category.

Specificity is the metric that evaluates a model’s ability to predict true negatives of each available category. These metrics apply to any categorical model.

$$\textbf{Sensitivity} = \frac{\textbf{True Positives}}{\textbf{True Positives} + \textbf{False Negatives}}$$

$$\textbf{Specificity} = \frac{\textbf{True Negatives}}{\textbf{True Negatives} + \textbf{False Positives}}$$

The equations for calculating sensitivity and specificity

We may have noticed that the equation for recall looks exactly the same as the equation for sensitivity. When to use either term depends on the task at hand.

6. Results of Baseline Models without Feature Selection

Baseline Model	Logistic Regression (LR)	Multi-Layer Perceptron (MPL)	Support Vector Machine (SVM)	Random Forest (RF)	XG Boost (XGB).
CONFUSION MATRIX					
True Positive (TP)	619	302	560	589	475
True Negative (TN)	5567	6829	6277	6299	6599
False Positive (FP)	1695	433	985	963	663
False Negative (FN)	354	671	413	384	498
Model Evaluation					
Time (Seconds)	0.5023	387.51	252.73	1.09	10.03
ROC_AUC	0.70138	0.62538	0.73085	0.73637	0.69844
Accuracy	0.75118	0.86594	0.82356	0.83643	0.85902
Precision	0.26750	0.41088	0.35594	0.37951	0.41740
Recall	0.63618	0.31038	0.60946	0.60534	0.48818
f1-Score	0.37664	0.35363	0.44941	0.46653	0.45002
Training Set Score	0.75130	0.68721	0.76725	0.75717	0.90668
Accuracy	75%	87%	83%	84%	86%
Specificity	77%	94%	86%	87%	91%
Sensitivity	64%	31%	58%	61%	49%
Risk	25%	13%	17%	16%	14%

Table 6.1

Scale	Highest	Lowest	Average

By looking at Figure 6.1, we can see confusion matrixes that represent the correct and incorrect prediction of 5 classifiers. The result shows that Logistic Regression highest number of TP, however its also has highest number of FP.

To better understand the confusion matrixes we can check Precision, Recall, and F1 score in table 6.1. XGBoost has the highest Precision 42% but lower Recall 48%. Multi-Layer Perceptron (MPL) has also higher Precision 41% but the lowest Recall of 31%. There is a trade-off between those two, one increase and the other one decrease. As explained above in the bank marketing campaign the Recall and Precision are both important because the FP and FN are both costly to the company so the best score to compare tests is the F1 score. Random Forest (RF) has the highest F1

score of 47% and highest ROC_AUC of 74%, however both Precision and Recall are lower. XGBoost has the second best F1 score of 45%.

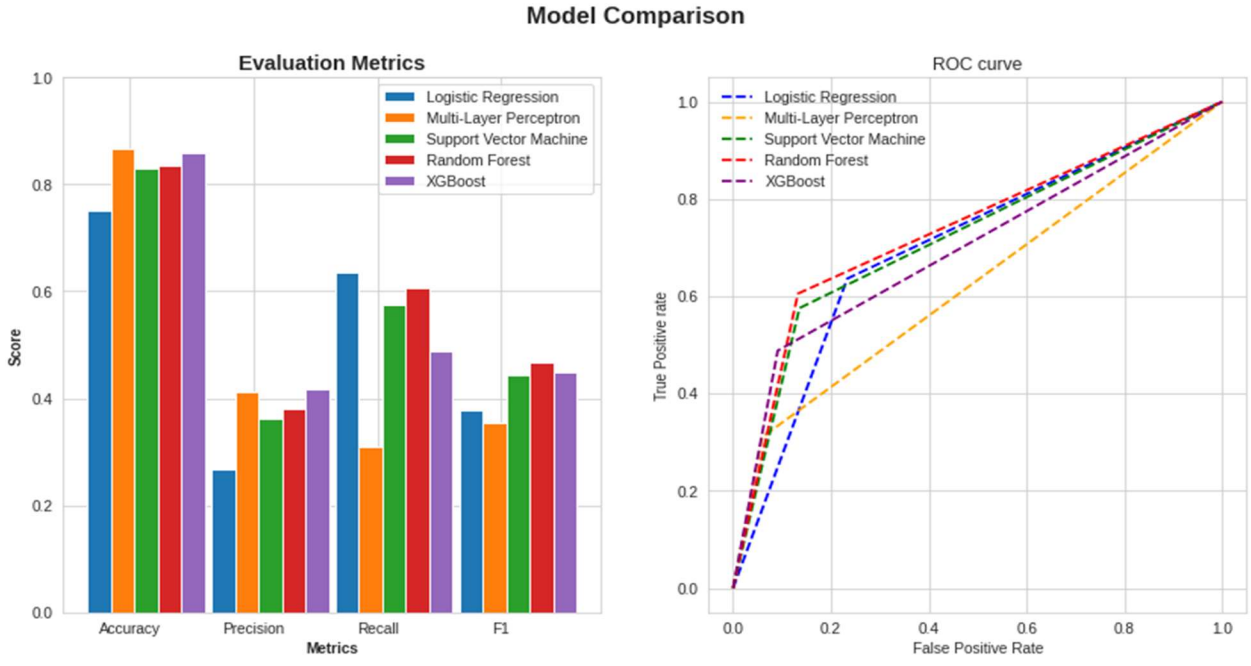
In terms of Accuracy Random Forest has the highest score of 74% followed by Support Vector Machine 73% and XGBoost of 70%

Considering the Others measures as the specificity, sensitivity and the risk, are presented in Table 6.1 . The risk here is defined as the proportion of observations in the test dataset that are wrongly classified by the model.

MLP also has the highest specificity of 94% and lowest risk of 13% followed by XGBoost also of 91% and 14% respectively

One last evaluation matrix which I want to highlight is the speed of model, MLP has the worst speed with average time of 388 seconds. Logistic Regression is highest followed by Random Forest and XGBoost.

Considering all matrix it seems that XGBoost has the best result followed by MLP and RF.



7. Feature Selection Technics

Feature Selection is the process where you automatically or manually select those features which contribute most to your prediction variable or output in which you are interested in.

Having irrelevant features in your data can decrease the accuracy of the models and make your model learn based on irrelevant features.

How to select features and what are Benefits of performing feature selection before modeling your data?

- Reduces Overfitting: Less redundant data means less opportunity to make decisions based on noise.
- Improves Accuracy: Less misleading data means modeling accuracy improves.
- Reduces Training Time: fewer data points reduce algorithm complexity and algorithms train faster.

The feature selection algorithms can be divided in to three categories: filter method, wrapper method, and embedded method.

Filter method pick up the intrinsic properties of the features that is the “relevance” of the features, measured via univariate statistics instead of cross-validation performance. Individual features are ranked according to specific criteria. The top N features are then selected.

Different types of ranking criteria are used for univariate filter methods, for example, mutual information gain (IG) which was used in my research. One of the major disadvantages of univariate filter methods is that they may select redundant features because the relationship between individual features is not taken into account.

Wrapper method requires some method to search the space of all possible subsets of features, assessing their quality by learning and evaluating a classifier with that feature subset.For example, I used forward feature selection with the algorithm Random Forest Classifier. Forward feature selection is used to select the best important features from the bank marketing dataset concerning the target output. It is an iterative method in which we start having no feature in the model. In each iteration, we keep adding the feature which best improves our model till a addition of a new variable does not improve the performance of the model. The evaluation criteria are AUC.

Embedded method contains the benefits of both the wrapper and filter methods by including interactions of features but also maintaining reasonable computational cost. Embedded methods are iterative in the sense that takes care of each iteration of the model training process and carefully extract those features which contribute the most to the training for a particular iteration. In my research, I used LASSO Regularization (L1) as it works better and achieves best results for my dataset compared to RIDGE Regularization (L2). Regularization consists of adding a penalty to the different parameters of the machine learning model to reduce the freedom of the model that is to avoid over-fitting. The penalty is applied over the coefficients that multiply each of the predictors. L1 has the property that is able to shrink some of the coefficients to zero. So, that feature can be removed from the model

8. Comparison of Models with Feature Selection

Feature Selection	Original Data	Baseline Model with 14 Features	Classification with Filter Methods for Feature Selection - Mutual Information	Classification with Wrapper Feature Selection - Forward feature selection	Classification Embedded Methods - LASSO Regularization (L1):
age	✓	✓	✓	✓	✓
job	✓	✓	✓	✓	
marital	✓	✓		✓	✓
education	✓	✓		✓	
default	✓	✓	✓	✓	✓
housing	✓	✓		✓	✓
loan	✓	✓		✓	✓
contact	✓	✓	✓	✓	✓
month	✓	✓	✓	✓	
day_of_week	✓	✓		✓	✓
duration	✓				
campaign	✓	✓	✓	✓	✓
pdays	✓				
previous	✓				
poutcome	✓	✓	✓	✓	
emp.var.rate	✓				
cons.price.idx	✓	✓	✓	✓	✓
cons.conf.idx	✓				
euribor3m	✓				
nr.employed	✓	✓	✓	✓	✓
y (outcome)					
Total	**20**	**14**	**9**	**14**	**10**

Table 8.1 Feature selected by Featuring Technics

CONFUSION MATRIX	Logistic Regression (LR)	Multi-Layer Perceptron (MPL)	Support Vector Machine (SVM)	Random Forest (RF)	XG Boost (XGB).
Baseline Model with 15 Features					
True Positive (TP)	619	302	560	589	475
True Negative (TN)	5567	6829	6277	6299	6599
False Positive (FP)	1695	433	985	963	663
False Negative (FN)	354	671	413	384	498
Classification with Filter Methods for Feature Selection - Mutual Information					
True Positive (TP)	648	565	560	558	498
True Negative (TN)	5651	5917	6277	6423	6577
False Positive (FP)	1611	1345	985	839	685
False Negative (FN)	325	408	413	415	475
Classification with Wrapper Feature Selection - Forward feature selection					
True Positive (TP)	619	302	560	588	475
True Negative (TN)	5567	6829	6277	6249	6599
False Positive (FP)	1695	433	985	1013	663
False Negative (FN)	354	671	413	385	498
Classification Embedded Methods - LASSO Regularization (L1):					
True Positive (TP)	608	511	604	594	506
True Negative (TN)	5532	5867	6040	6013	6466
False Positive (FP)	1730	1395	1222	1249	796
False Negative (FN)	365	462	369	379	467

Table 8.2 Confusion Matrix

Table 8.3 Model Evaluation

Model Evaluation	Logistic Regression (LR)	Multi-Layer Perceptron (MPL)	Support Vector Machine (SVM)	Random Forest (RF)	XG Boost (XGB).
Baseline Model					
Time (Seconds)	0.6128	400.00	292.24	1.15	10.19
ROC_AUC	0.70138	0.62538	0.71995	0.73637	0.69844
Accuracy	0.75118	0.86594	0.83024	0.83643	0.85902
Precision	0.26750	0.41088	0.36246	0.37951	0.41740
Recall	0.63618	0.31038	0.57554	0.60534	0.48818
f1-Score	0.37664	0.35363	0.44480	0.46653	0.45002
Training Set Score	0.75130	0.68721	0.81011	0.75717	0.90668
Classification with Filter Methods for Feature Selection - Mutual Information					
Time (Seconds)	0.44	319.48	232.39	1.18	7.86
ROC_AUC	0.72207	0.69773	0.73085	0.72898	0.70875
Accuracy	0.76491	0.78713	0.82356	0.84772	0.85914
Precision	0.28685	0.29581	0.35594	0.39943	0.42096
Recall	0.66598	0.58068	0.60946	0.57348	0.51182
f1-Score	0.40099	0.39195	0.44941	0.47089	0.46197
Training Set Score	0.73837	0.73178	0.76725	0.75403	0.85458
Classification with Wrapper Feature Selection - Forward feature selection					
Time	0.52	390.93	290.46	1.23	10.35
ROC_AUC	0.70138	0.62538	0.71995	0.73241	0.69844
Accuracy	0.75118	0.86594	0.83024	0.83024	0.85902
Precision	0.26750	0.41088	0.36246	0.36727	0.41740
Recall	0.63618	0.31038	0.57554	0.60432	0.48818
f1-Score	0.37664	0.35363	0.44480	0.45688	0.45002
Training Set Score	0.75130	0.68721	0.81011	0.75717	0.90668
Classification Embedded Methods -LASSO Regularization (L1):					
Time	108.59	306.87	205.15	1.16	8.04
ROC_AUC	0.69332	0.66654	0.72624	0.71925	0.70521
Accuracy	0.74560	0.77450	0.80680	0.80231	0.84663
Precision	0.26005	0.26810	0.33078	0.32230	0.38863
Recall	0.62487	0.52518	0.62076	0.61048	0.52004
f1-Score	0.36726	0.35498	0.43158	0.42188	0.44484
Training Set Score	0.75130	0.74628	0.77038	0.75130	0.87378

Table 8.4 Other Evaluation Matrix

	Logistic Regression (LR)	Multi-Layer Perceptron (MPL)	Support Vector Machine (SVM)	Random Forest (RF)	XG Boost (XGB).
Baseline Model					
Accuracy	75%	87%	83%	84%	86%
Specificity	77%	94%	86%	87%	91%
Sensitivity	64%	31%	58%	61%	49%
Risk	25%	13%	17%	16%	14%
Classification with Filter Methods for Feature Selection - Mutual Information					
Accuracy	76%	79%	82%	85%	86%
Specificity	78%	81%	85%	88%	91%
Sensitivity	67%	58%	61%	57%	51%
Risk	24%	21%	18%	15%	14%
Classification with Wrapper Feature Selection - Forward feature selection					
Accuracy	75%	87%	83%	83%	86%
Specificity	77%	94%	86%	86%	91%
Sensitivity	64%	31%	58%	60%	49%
Risk	25%	13%	17%	17%	14%
Classification Embedded Methods - LASSO Regularization (L1):					
Accuracy	75%	77%	81%	80%	85%
Specificity	76%	81%	83%	83%	89%
Sensitivity	62%	53%	62%	61%	52%
Risk	25%	23%	19%	20%	15%

Table 8.5 ROC_AUC Training and Test Result

	ROC_AUC on Training Set	ROC_AUC on Testing set
Baseline Model		
Logistic Regression (LR)	0.8245	0.7530
Multi-Layer Perceptron (MPL)	0.8199	0.7349
Support Vector Machine (SVM)	0.9039	0.7548
Random Forest (RF)	0.8327	0.7806
XG Boost (XGB).	0.9710	0.7554
Classification with Filter Methods for Feature Selection - Mutual Information		
Logistic Regression (LR)	0.8013	0.7754
Multi-Layer Perceptron (MPL)	0.8017	0.7535
Support Vector Machine (SVM)	0.9039	0.7548
Random Forest (RF)	0.8233	0.7820
XG Boost (XGB).	0.9398	0.7606
Classification with Wrapper Feature Selection - Forward feature selection		
Logistic Regression (LR)	0.8245	0.7530
Multi-Layer Perceptron (MPL)	0.8199	0.7349
Support Vector Machine (SVM)	0.9039	0.7548
Random Forest (RF)	0.8344	0.7778
XG Boost (XGB).	0.9710	0.7554
Classification Embedded Methods - LASSO Regularization (L1):		
Logistic Regression (LR)	0.8235	0.7495
Multi-Layer Perceptron (MPL)	0.8179	0.7535
Support Vector Machine (SVM)	0.8571	0.7536
Random Forest (RF)	0.8317	0.7734
XG Boost (XGB).	0.9520	0.7574

9. Summary of Model Comparison

- Classification with Wrapper Feature Selection - Forward feature selection did not reduce any more features and retains all 14 columns as the baseline models as shown in table 8.1
- XGBoost Performs best in both Classification with Filter Methods for Feature Selection - Mutual Information and Classification Embedded Methods -LASSO Regularization (L1) (Table 8.6)

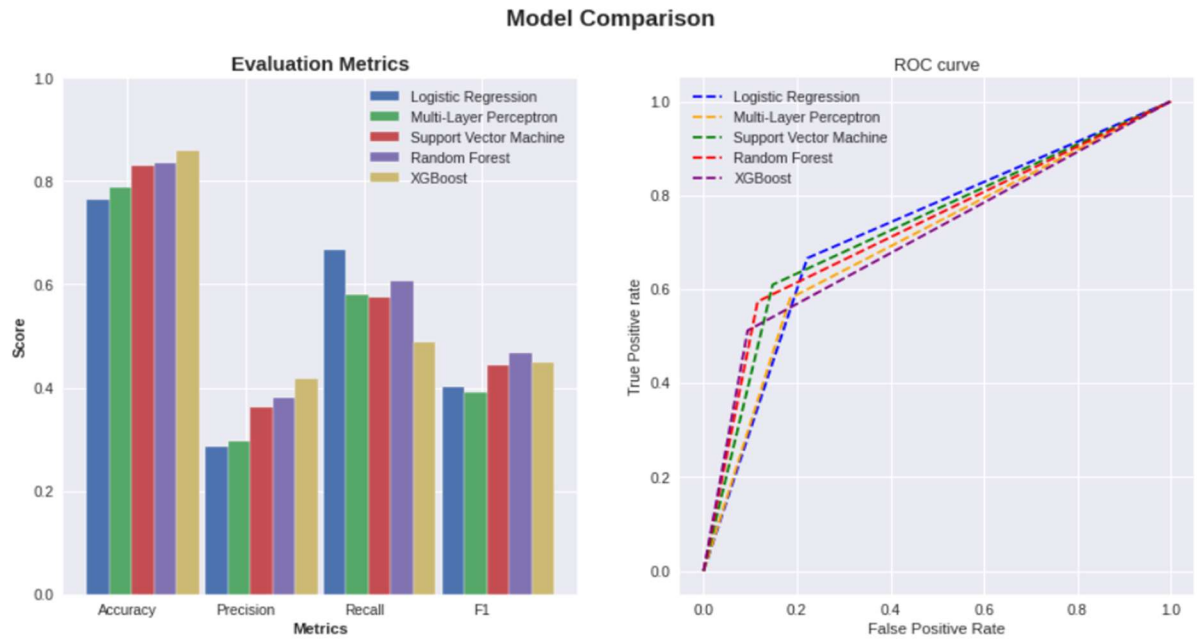
XG Boost (XGB).	Baseline Model	Classification with Filter Methods for Feature Selection - Mutual Information	Classification with Wrapper Feature Selection - Forward feature selection	Classification Embedded Methods - LASSO Regularization (L1):
CONFUSION MATRIX				
True Positive (TP)	475	498	475	506
True Negative (TN)	6599	6577	6599	6466
False Positive (FP)	663	685	663	796
False Negative (FN)	498	475	498	467
Model Evaluation				
Time (Seconds)	10.18767	7.86248	10.34631	8.04074
ROC_AUC	0.69844	0.70875	0.69844	0.70521
Accuracy	0.85902	0.85914	0.85902	0.84663
Precision	0.41740	0.42096	0.41740	0.38863
Recall	0.48818	0.51182	0.48818	0.52004
f1-Score	0.45002	0.46197	0.45002	0.44484
Training Set Score	0.90668	0.85458	0.90668	0.87378
Accuracy	0.85902	0.85914	0.85902	0.84663
Specificity	0.90870	0.90567	0.90870	0.89039
Sensitivity	0.48818	0.51182	0.48818	0.52004
Risk	0.14098	0.14086	0.14098	0.15337
ROC_AUC on Training Set	0.97098	0.93980	0.97098	0.95196
ROC_AUC on Testing set	0.75537	0.76063	0.75537	0.75745

Table 8.6

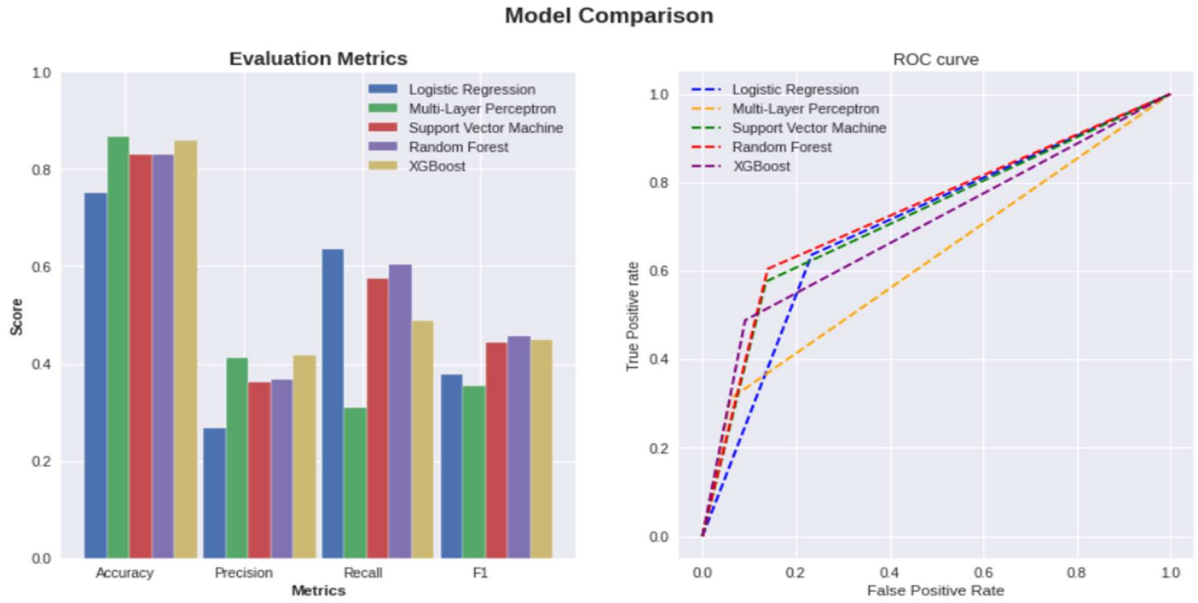
- Logistic Regression perform the worst of all 5 classification model with highest recall in both baseline and both Classification with Filter Methods for Feature Selection - Mutual Information and Embedded Methods -LASSO Regularization (L1)

- Both Multi-Layer Perceptron (MPL) and Support Vector Machine (SVM) takes longer time to run
- Multi-Layer Perceptron (MPL) perform better without feature selection
- Random Forest performed better under Classification with Filter Methods for Feature Selection - Mutual Information
- Support Vector Machine (SVM) performed better Classification with Filter Methods for Feature Selection - Mutual Information
- Random Forest is the most efficient model in terms of time to run

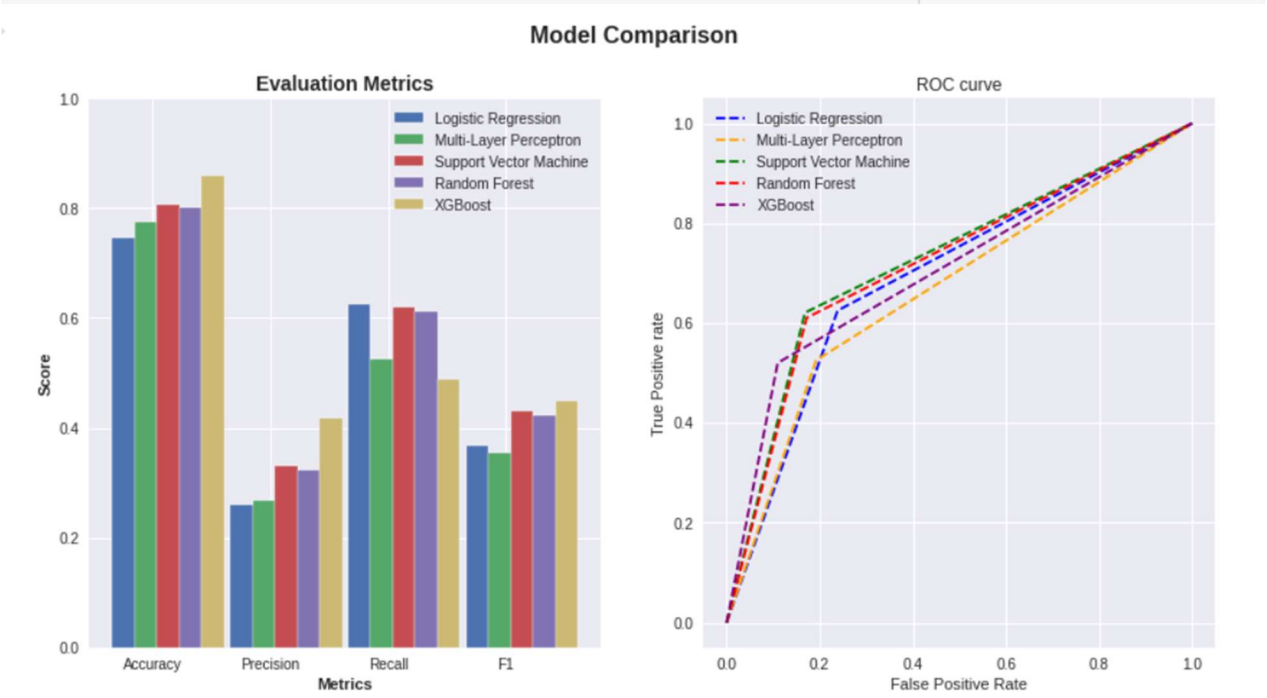
8.6 **Classification with Filter Methods for Feature Selection-Mutual Information Gain**



8.7 **Classification with Wrapper Feature Selection - Forward feature selection**



8.8 Classification Embedded Methods -LASSO Regularization (L1):



10. Conclusion:

Before conclusion I would like to revisit nature of Term Deposit and link it with the analysis of result of exploratory data analysis

A term deposit is a type of deposit account held at a financial institution where money is locked up for some set period of time. Term deposits are usually short-term deposits with maturities ranging from one month to a few years. Typically, term deposits offer higher interest rates than traditional liquid savings accounts, whereby customers can withdraw their money at any time.

Major findings of EDA can help understand the type of customer and features will most likely to subscribe the term deposit ad.

Personal Attributes

- Customers aged 30-40, 20-30, and 40-50 had a higher percentage of subscription to a deposit account.
- 45.5% of Seniors (+60 years old) subscribed to the term deposit
- Student have highest conversion rate followed by retired, unemployed and admin..
- people with admin jobs have subscribed more for the deposits than people with any other profession followed by technicians and blue collar had a higher percentage of subscription to a deposit account
- People who are married have subscribed for deposits more than people with any other marital status.
- People with university degree as education qualification are the most who have subscribed for the deposits. They are also the most who have not subscribed for deposits.
- People with default status as no are the most one’s who have and have not subscribed for bank deposits.
- People with housing loan are the most ones who have subscribed for deposits.
- People with no personal loan are the most ones who have been contacted by the bank for the deposits.

last contact of the current campaign:

- Most people are contacted more in cellular than telephone
- People have been contacted more in the month of May, followed by July, August, June
- Days_of_week are irrelevant
- the duration (last contact duration) of a customer can be useful for predicting

campaign attributes

- the most of the customers have never been contacted before. However People that were previously contacted subscribed in a much higher rate to the term deposit. While in people never contacted only 10% subscribed to the deposit, for people that was previously contacted more than twice the campaign success increases to >45%.
- The longer the conversation was with clients, the more likely they were to make a

social and economic context attributes

- As highlighted earlier during EDA social and economic context attributes are highly correlated
- the lower the euribor3mis, the higher the amount of subscriptions
- when the cons_price_idx(consumer price index) increases there is a strong negative response from the clients' subscriptions
- when the emp_var_rate (employment rate) is negative, there is a higher positive response to the campaign.

Feature Selected

The most useful feature selected by Classification with Filter Methods for Feature Selection - Mutual Information & Classification Embedded Methods -LASSO Regularization (L1) are given in table 10.1 and following features are common in both of the methods

“Age, default, contact, campaign, cons.price.idx, nr.employed”

Most useful features

Feature Selection	Classification with Filter Methods for Feature Selection - Mutual Information	Classification Embedded Methods -LASSO Regularization (L1):
age	✓	✓
job	✓	
marital		✓
default	✓	✓
housing		✓
loan		✓
contact	✓	✓
month	✓	
day_of_week		✓
campaign	✓	✓
poutcome	✓	
cons.price.idx	✓	✓
nr.employed	✓	✓
Total	**9**	**10**

Table 10.1

- Overall XGBoost Performs best with and without feature selection and also efficient to run

The limitation of this research was working with Scikit-learn (Sklearn) library in Python Other libraries that are available in Python are not used. This study is limited to the Classification Model selected and other classification models behavior are not tested. Dimension reduction through PCA is not been performed

References

A.Elsalamony, H. (2014). Bank Direct Marketing Analysis of Data Mining Techniques. *International Journal of Computer Applications*, *85*(7), 12–22. https://doi.org/10.5120/14852-3218

Binary classification. (2021, May 9). Wikipedia. https://en.wikipedia.org/wiki/Binary_classification

Chen, J. (2019). *Term Deposit Definition*. Investopedia. https://www.investopedia.com/terms/t/termdeposit.asp

Feature Scaling | Standardization Vs Normalization. (2020, April 3). Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/

Ghatasheh, N., Faris, H., AlTaharwa, I., Harb, Y., & Harb, A. (2020). Business Analytics in Telemarketing: Cost-Sensitive Analysis of Bank Campaigns Using Artificial Neural Networks. *Applied Sciences*, *10*(7), 2581. https://doi.org/10.3390/app10072581

Kumar, N. (2019, March 9). *The Professionals Point: Advantages of XGBoost Algorithm in Machine Learning*. The Professionals Point. http://theprofessionalspoint.blogspot.com/2019/03/advantages-of-xgboost-algorithm-in.html

Loukas, S. (2020, June 10). *How Scikit-Learn’s StandardScaler works*. Medium. https://towardsdatascience.com/how-and-why-to-standardize-your-data-996926c2c832

Moro, S., Cortez, P., & Rita, P. (2014a). A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, *62*, 22–31. https://doi.org/10.1016/j.dss.2014.03.001

Moro, S., Cortez, P., & Rita, P. (2014b). Using customer lifetime value and neural networks to improve the prediction of bank deposit subscription in telemarketing campaigns. *Neural Computing and Applications*, *26*(1), 131–139. https://doi.org/10.1007/s00521-014-1703-0

Nair, A. (2022, January 24). *Create Artificial Data With SMOTE*. Medium. https://towardsdatascience.com/create-artificial-data-with-smote-2a31ee855904#:~:text=SMOTE%20should%20only%20be%20used%20to%20augment%20training

Neural Networks From Scratch in Python & R | With Mathematics in Python. (2020, July 23). Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/07/neural-networks-from-scratch-in-python-and-r/

Raheel Shaikh. (2018, October 28). *Feature Selection Techniques in Machine Learning with Python*. Medium; Towards Data Science. https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e

Safarkhani, F., & Moro, S. (2021). Improving the Accuracy of Predicting Bank Depositor’s Behavior Using a Decision Tree. *Applied Sciences*, *11*(19), 9016. https://doi.org/10.3390/app11199016

Sołtys, M., Jaroszewicz, S., & Rzepakowski, P. (2014). Ensemble methods for uplift modeling. *Data Mining and Knowledge Discovery*, *29*(6), 1531–1559. https://doi.org/10.1007/s10618-014-0383-9

What is a Multilayer Perceptron (MLP)? - Definition from Techopedia. (n.d.). Techopedia.com. https://www.techopedia.com/definition/20879/multilayer-perceptron-mlp

What is a Support Vector Machine (SVM)? - Definition from Techopedia. (2019). Techopedia.com. https://www.techopedia.com/definition/30364/support-vector-machine-

svm

Wikipedia Contributors. (2019a, April 7). *Multilayer perceptron*. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Multilayer_perceptron

Wikipedia Contributors. (2019b, May 30). *Gradient boosting*. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Gradient_boosting