HarvardX: Capstone

Predicting Subscription to Term Deposit Using the Bank Marketing Data Set

Dwaipayan Chatterjee

20/03/2022

Abstract

The core business of a financial institution can be broadly classified as lending and borrowing. Lending generates revenue to the bank in the form of interest from customers with some level of default risk involved. Borrowing, or rather attracting public's savings into the bank is another source of revenue generation, which can be less risky than the former. A bank usually invests the customer's long-term deposits into riskier financial assets which can earn the better return than what they pay to their customer. The customer, on the other hand, is assured a risk-free return on his/her deposit. However, the return on the fixed-term deposit is better than the savings account as the customer is deprived off the rights to use the fund prior to the maturity unless one is ready to compensate the bank as per the pre-specified agreements on the particular term deposit scheme.

There is a stiff competition among the financial institutions/banks in increasing the customer base in their retail banking segment. Along with offering innovative products to the public, a huge amount of money is spent on marketing their products. The term deposit is very important among the diverse range of products and services offered by banks in retail banking segment. With advancement in data science and machine learning and availability of data, most banks are adapting to a data-driven decision. The dataset here consists of direct marketing by contacting the clients and assessing the success rate of sales made.

In this project, we apply machine learning algorithms to build a predictive model of the data set to provide a necessary suggestion for marketing campaign team. The goal is to predict whether a client will subscribe a term deposit (variable y) with the help of a given set of dependent variables. This is a real dataset collected from a Portuguese bank that used its own contact-center to do direct marketing campaigns to motivate and attract the clients for their term deposit scheme to enhance the business.

Contents

1. Introduction

The objective of this project is to predict whether the client will subscribe (yes/no) to a term deposit at a Portuguese bank using the data from May 2008 to November 2010.The data set used is sourced from the UCI Machine Learning Repository and is based on the phone calls made (often more than once) for the marketing campaign to access if the term deposit (product/target feature) would be subscribed (Yes/No). The dataset can be accessed at https://archive.ics.uci.edu/ml/datasets/bank+marketing [1]. This project has two phases. While the Phase I focuses on data preprocessing and exploration, as covered in this report, the Phase II covers the model building and its performance analysis.

Dataset Description

The data set for the project was downloaded from the website "UCI Machine Learning Repository" into an excel spreadsheet so that we could convert into CSV file and read in R studio. The data set is related to a Portuguese banking institution's marketing campaign. The marketing campaigns were based on telemarketing. The contact information includes date, time and number of contacts made to a customer in order to get the response of "yes" or "no" to their term deposit. The whole data set is the bank's client database consisting of 17 different variables/attribute which is elaborated below.

- Number of Observations: 4521 Number of Attributes: 17

Target Feature

The desired target feature is "y" - "Has the client subscribed to a term deposit?"

According to the dataset, "y" has two classes so it is identified as a binary classification problem.

- Yes: The client has subscribed to the term deposit.
- No: The client has not subscribed to the term deposit.

➢ Descriptive Features

The 17 inputs contained in the dataset are:

- ❖ Bank client:
  - age: the client's age (numeric)
  - job: type of job (categorical: 'admin.', 'blue-collar', 'entrepreneur', 'housemaid', 'management', 'retired', 'self-employed', 'services', 'student', 'technician', 'unemployed', 'unknown')
  - marital: marital status (categorical: 'divorced', 'married', 'single', 'unknown'; note: 'divorced' means divorced or widowed)
  - education (categorical: 'unknown', 'primary', 'secondary', 'tertiary')
  - default: has credit in default? (Categorical: 'no', 'yes', 'unknown')
  - balance: average yearly balance, in euros (numeric)
  - housing: has housing loan? (Categorical: 'no', 'yes', 'unknown')
  - loan: has personal loan? (Categorical: 'no', 'yes', 'unknown')
- ❖ Related with the last contact of the current campaign:
  - contact: contact communication type (categorical: "unknown", "telephone", "cellular")
  - day: last contact day of the month (numeric)
  - month: last contact month of year (categorical: "jan", "feb", "mar", …, "nov", "dec")
  - duration: last contact duration, in seconds (numeric)
  - Other attributes:
  - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)
  - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric, -1 means client was not previously contacted)
  - previous: number of contacts performed before this campaign and for this client (numeric)
  - poutcome: outcome of the previous marketing campaign (categorical: "unknown", "other", "failure", "success")
- ❖ Output variable (desired target):
  - y - has the client subscribed to the term deposit? (Binary: "yes", "no")

2. Purpose

The final goal of this project is to fit the possible set of the models to predict whether or not the marketing campaign is successful in the acquisition of customers into the bank's term deposit. We analyzed the performances of three different machine learning algorithms by training and testing data sets and selected the best according to the degree of accuracy. This would suggest if the marketing campaign team of the Portuguse bank should continue investing in telemarketing their term deposit scheme.

The objective of the project will be to include:

- Methodology for building algorithms
- Details of the algorithms and fine-tuning over the data set
- Performance comparison and choosing the best model
- Limitations of the algorithms – Summary and Conclusion

In this document, we create a Customer Segmentation Using thi dataset [1] and applying the courses/lessons learned during the HarvardX's Data Science Professional Certificate program.

3. DATA PREPROCESSING

➢ Preliminaries

The necessary libraries are loaded and the downloaded dataset is imported in R using base R read.csv () function and assigned to an object "bank" and redundant variables were dropped before proceeding further with data preprocessing.

Lets start with Loading the Following Library:

```
library(readr)
library(knitr)
library(mlr)
library(ggplot2)
library(magrittr)
library(cowplot)
library(dplyr)
library(gridExtra)
library(GGally)
library(stringr)
library(glmnet)
library(doParallel)
library(class)
library(gmodels)
library(TSA)
library(FitAR)
```

```
library(car)
library(FNN)
library(reshape2)
library(e1071)
library(caret)
library(psych)
library(dplyr)
```

First, we are reading te data

```
##   age         job marital education default balance housing loan  contact
day
## 1  30  unemployed married   primary      no    1787      no   no cellular
19
## 2  33    services married secondary      no    4789     yes  yes cellular
11
## 3  35  management  single  tertiary      no    1350     yes   no cellular
16
## 4  30  management married  tertiary      no    1476     yes  yes  unknown
3
## 5  59 blue-collar married secondary      no       0     yes   no  unknown
5
## 6  35  management  single  tertiary      no     747      no   no cellular
23
##   month duration campaign pdays previous poutcome  y
## 1   oct       79        1    -1        0  unknown no
## 2   may      220        1   339        4  failure no
## 3   apr      185        1   330        1  failure no
## 4   jun      199        4    -1        0  unknown no
## 5   may      226        1    -1        0  unknown no
## 6   feb      141        2   176        3  failure no
```

- **Data Cleaning and Transformation**

To confirm that the feature type matches the description as outlined in the documentation, str () function is used.

```
## 'data.frame':    4521 obs. of  17 variables:
##  $ age      : int  30 33 35 30 59 35 36 39 41 43 ...
##  $ job      : chr  "unemployed" "services" "management" "management" ...
##  $ marital  : chr  "married" "married" "single" "married" ...
##  $ education: chr  "primary" "secondary" "tertiary" "tertiary" ...
##  $ default  : chr  "no" "no" "no" "no" ...
##  $ balance  : int  1787 4789 1350 1476 0 747 307 147 221 -88 ...
##  $ housing  : chr  "no" "yes" "yes" "yes" ...
##  $ loan     : chr  "no" "yes" "no" "yes" ...
##  $ contact  : chr  "cellular" "cellular" "cellular" "unknown" ...
##  $ day      : int  19 11 16 3 5 23 14 6 14 17 ...
##  $ month    : chr  "oct" "may" "apr" "jun" ...
##  $ duration : int  79 220 185 199 226 141 341 151 57 313 ...
##  $ campaign : int  1 1 1 4 1 2 1 2 2 1 ...
```

```
##  $ pdays    : int  -1 339 330 -1 -1 176 330 -1 -1 147 ...
##  $ previous : int  0 4 1 0 0 3 2 0 0 2 ...
##  $ poutcome : chr  "unknown" "failure" "failure" "unknown" ...
##  $ y        : chr  "no" "no" "no" "no" ...
```

Now The dataset is summarised feature wise to get a summary about the numeric features.

Feature Summary of Bank Data

name	type	na	mean	disp	median	mad	min	max	nlevs
age	integer	0	41.1700951	10.5762110	39	10.3782	19	87	0
job	character	0	NA	0.7856669	NA	NA	38	969	12
marital	character	0	NA	0.3813316	NA	NA	528	2797	3
education	character	0	NA	0.4899359	NA	NA	187	2306	4
default	character	0	NA	0.0168104	NA	NA	76	4445	2
balance	integer	0	1422.657819	3009.638142	444	658.2744	-3313	71188	0
housing	character	0	NA	0.4339748	NA	NA	1962	2559	2
loan	character	0	NA	0.1528423	NA	NA	691	3830	2
contact	character	0	NA	0.3594338	NA	NA	301	2896	3
day	integer	0	15.9152842	8.2476673	16	10.3782	1	31	0
month	character	0	NA	0.6907764	NA	NA	20	1398	12
duration	integer	0	263.9612917	259.8566326	185	143.8122	4	3025	0
campaign	integer	0	2.7936297	3.1098067	2	1.4826	1	50	0
pdays	integer	0	39.7666445	100.1211244	-1	0.0000	-1	871	0
previous	integer	0	0.5425791	1.6935624	0	0.0000	0	25	0

poutcome	character	0	NA	0.1804910	NA	NA	129	3705	4
y	character	0	NA	0.1152400	NA	NA	521	4000	2

➢ Scanning for NAs

The dataset is scanned for missing values using is.na () function and no missing values are found. It can be assumed now that the dataset doesn't contain any missing values.

```
##       age       job   marital education   default   balance   housing
loan
##         0         0         0         0         0         0         0
0
##   contact       day     month  duration  campaign     pdays  previous  pou
tcome
##         0         0         0         0         0         0         0
0
##         y
##         0
```

➢ Scanning for White Space

In order to avoid any discrepancy later, extra white space, if any, is removed for all the character features.

➢ Scanning for Case Errors

Scanning case error means scanning inconsistency in categorical attributes caused by random upper/lower case mistakes. First we applied unique() function to show all unique available values in each specific categorical variable, on that basis, we can detect any odd cases.

```
##  [1] "unemployed"    "services"      "management"    "blue-collar"
##  [5] "self-employed" "technician"    "entrepreneur"  "admin."
##  [9] "student"       "housemaid"     "retired"       "unknown"

## [1] "married"  "single"   "divorced"

## [1] "primary"   "secondary" "tertiary"  "unknown"

## [1] "no"  "yes"

## [1] "no"  "yes"

## [1] "no"  "yes"

## [1] "unknown" "failure" "other"   "success"
```

After performing the function, it is confirmed that there are no upper/lowercase errors in the outputs and they matched with the descriptive features.

➢ Renaming some variable's values

There are 4 descriptive features ("default", "housing", "loan" and "target y") that have the same binary responses ("yes" or "no"). In order to avoid confusion with target y during the visualisation/exploration, the values of these descriptive features are labelled differently as follows: (a) defaulter, no defaulter (b) housing loan, no housing loan (c) personal loan , no personal loan.

4. Exploratory Data Aalysis

➢ Univariate Visualization

For Univariate visualization, Bar chart and BoxHistogram Plot are used for categorical and numerical features respectively. For categorical features, Bar plot illustrates a bar chart with the categories on X axis and frequency/count on the Y axis and is useful in presenting the count by categories. For the numerical features, BoxHistogramPlot depicts a histogram and a box plot. While a histogram is useful in visualizing the shape of the underlying distribution, a box plot tells the range of the attribute and helps detect any outliers.

➢ Categorical Features

Figure 1 indicates that out of the total number of people contacted for marketing campaign, collectively close to 50 % are blue-collar, management professionals and technicians. Around 28,000 of those contacted in total are married (figure 2) while according to figure 3, 25,000 out of total are possessing secondary education. Almost all of the people have paid their dues on time with less than 1000 having default credit(figure 4). While more than half of the people have running housing loan(figure 5), comparatively fewer people, around 7000, avail personal loan(figure 6). The figure 7 indicates that the outcome of the previous calls for a substantial amount of individuals is unknown hence it can be deduced intuitively that most likely it has no bearing on the predicted outcome and can be left out during predictive modelling. The bar chart of the target feature, figure 8, illustrates that a large proportion of individuals do not subscribe to term deposit.

Figure 1 - Job

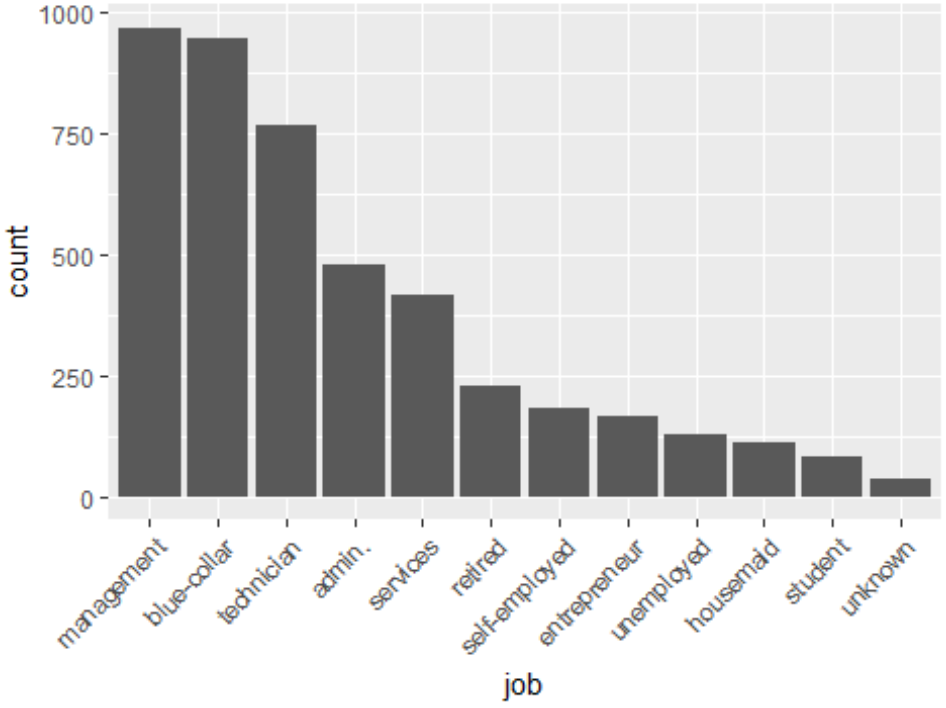


Figure 2 - Marital Status

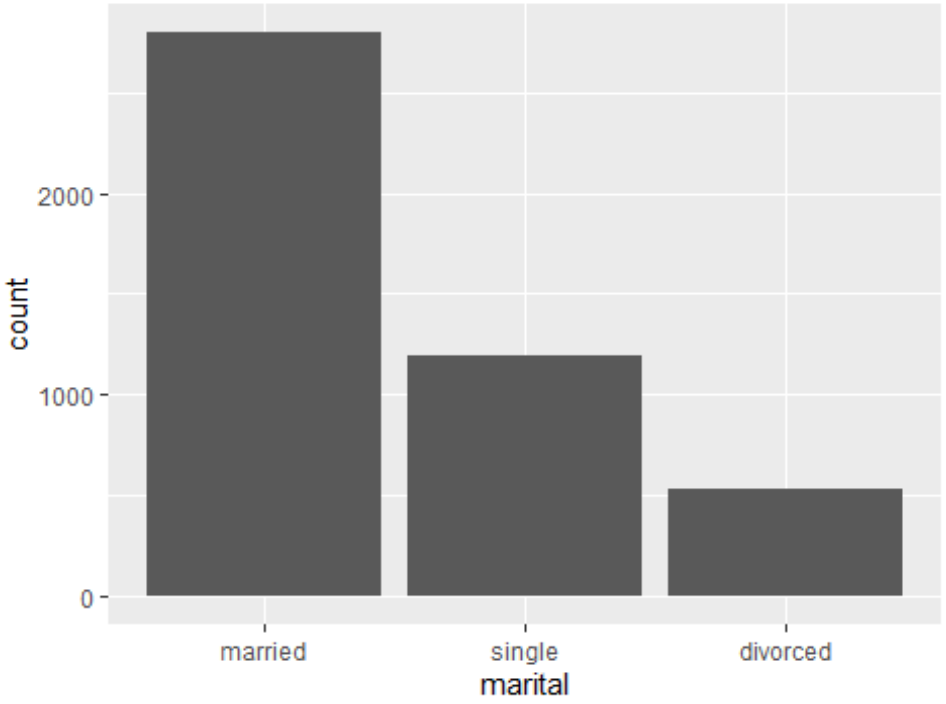


Figure 3 - Education

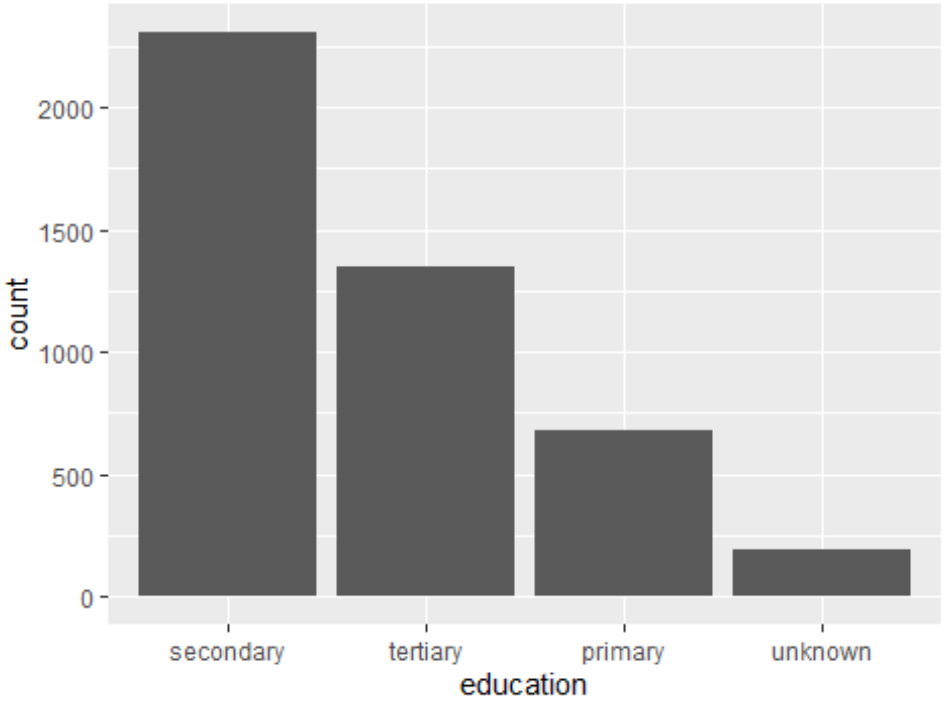


Figure 4 - Default Credit

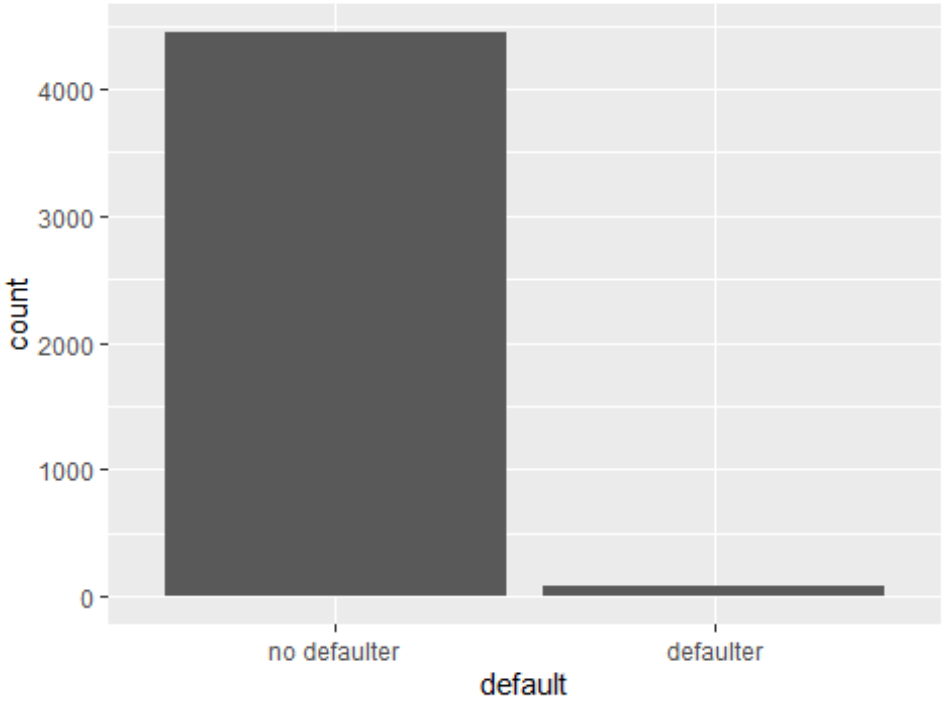


Figure 5 - Housing Loan

Figure 6 - Personal loan

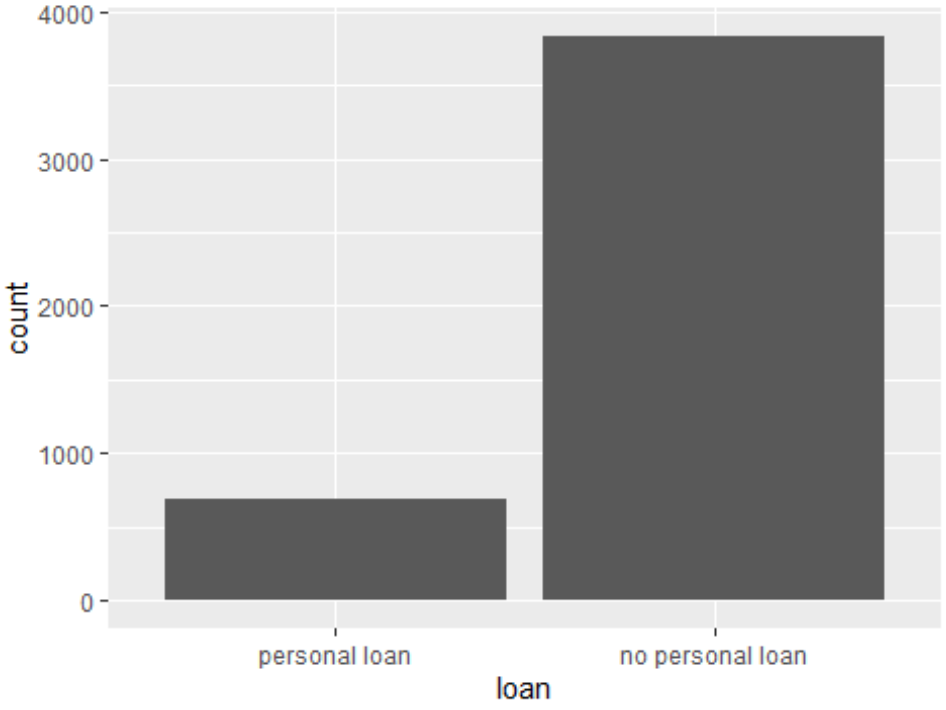


Figure 7 - Previous Outcome

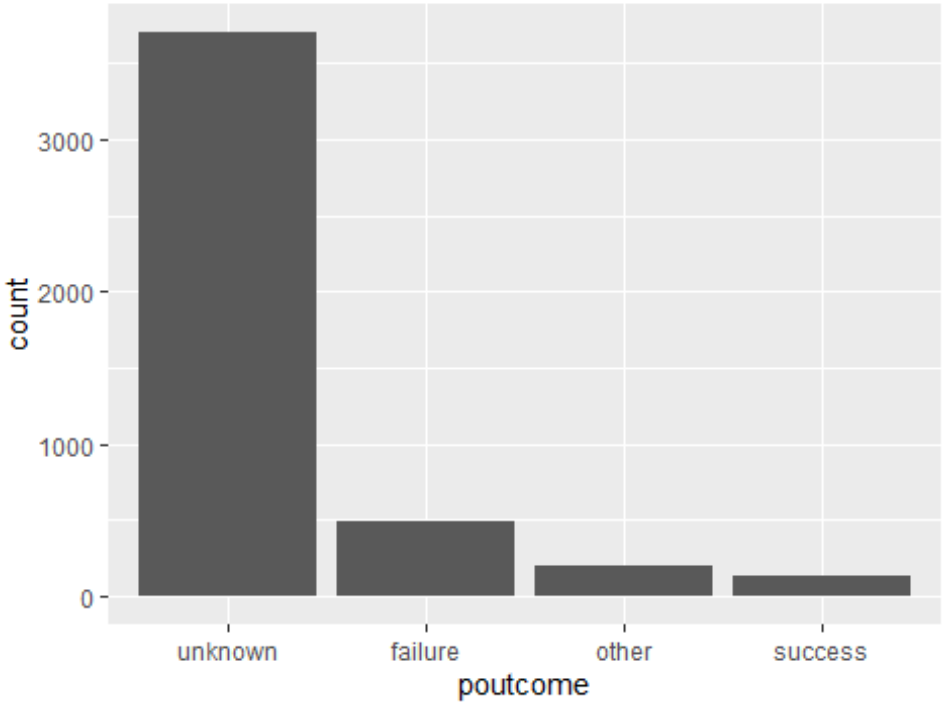
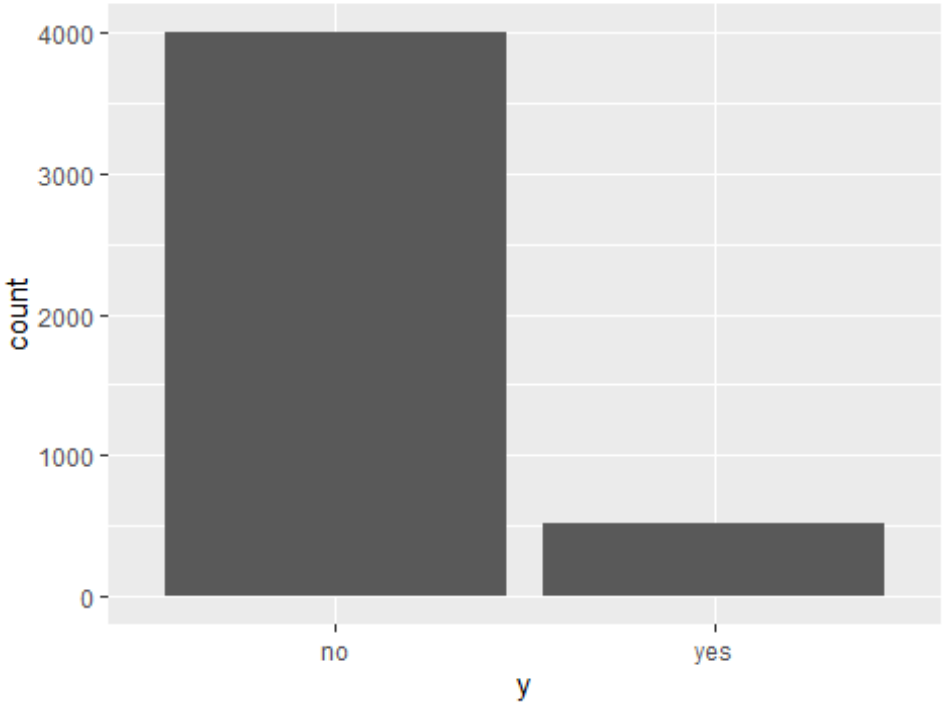


Figure 8 - Target Y



➢ Numerical features

Figure 9 depicts that most of the individuals aged between 30-60 years. The boxplot for the average yearly balance, figure 10, shows a median of zero signifying that most of the people contacted for this campaign have negative or nearly zero average yearly balance.Figure 11 instantiates that vast majority of people decide about the subscription in the first 500 seconds(8 minutes), with a median of around 300 seconds(5 minutes). Most of the clients were contacted only once or twice during this campaign as illustated in figure 12. The Figure 13 displays the number of days passed after the client/individual was last contacted and the median 0 without any Inter-quartile range reflects that almost all of the individuals are contacted for the first time during this campaign. The fact assumed from the interpretation of the figure 13 is confirmed by the figure 14 where number of contacts performed before this campaign to the same individual is plotted and the median is zero with no interquartile range means that no contacts were made previously to the clients contacted during this marketing campaign.

Figure 10 - Average Yearly Balance

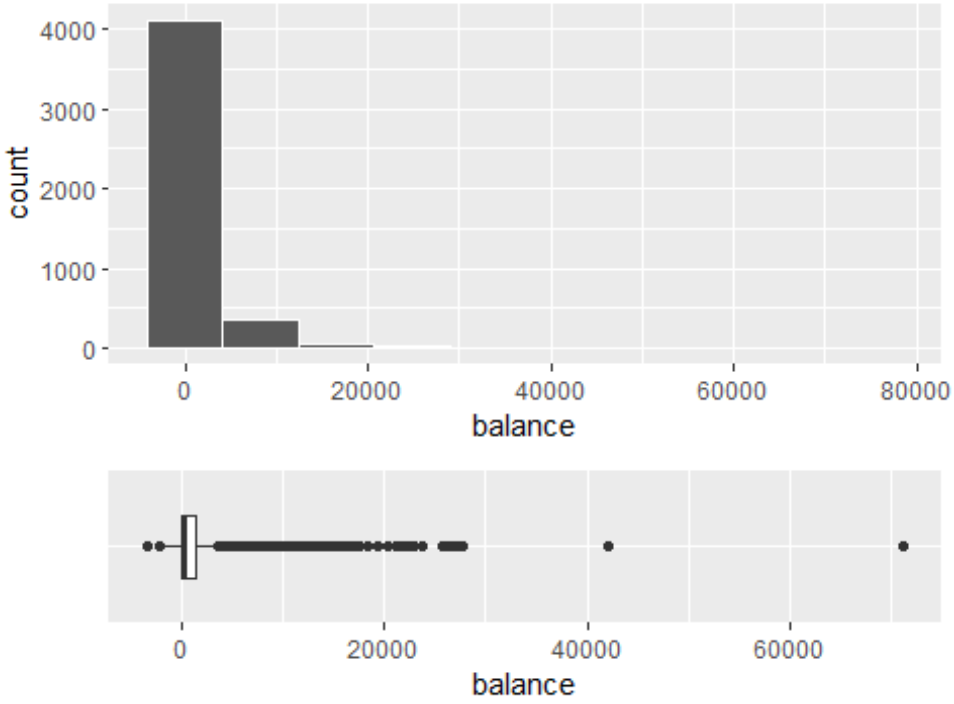


Figure 11 - Duration of last call

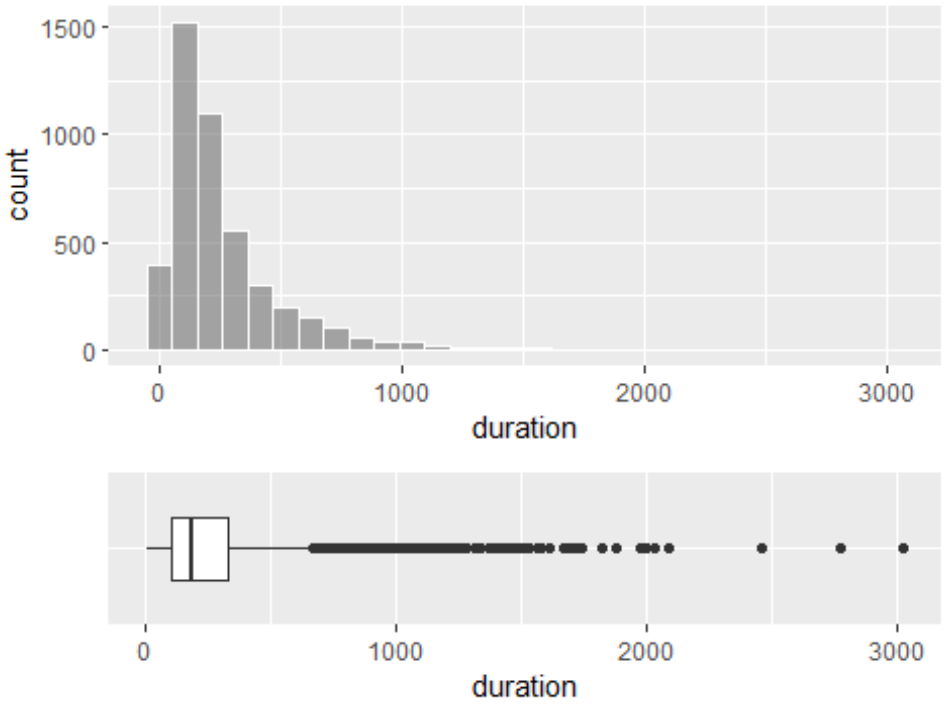


Figure 12 - Campaign

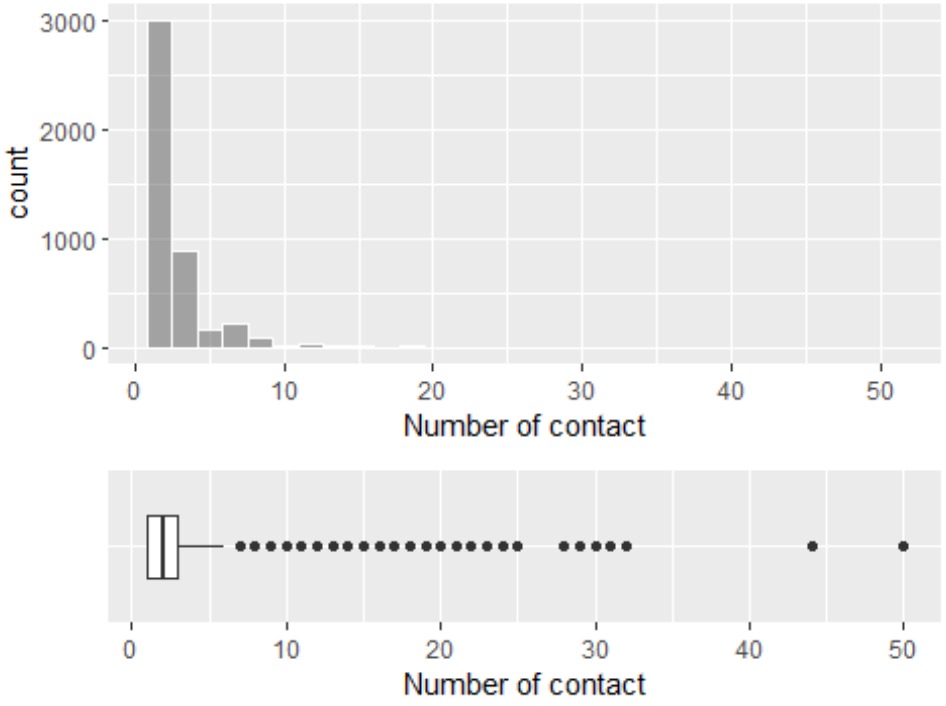


Figure 13 - Number of days passed after the client wa

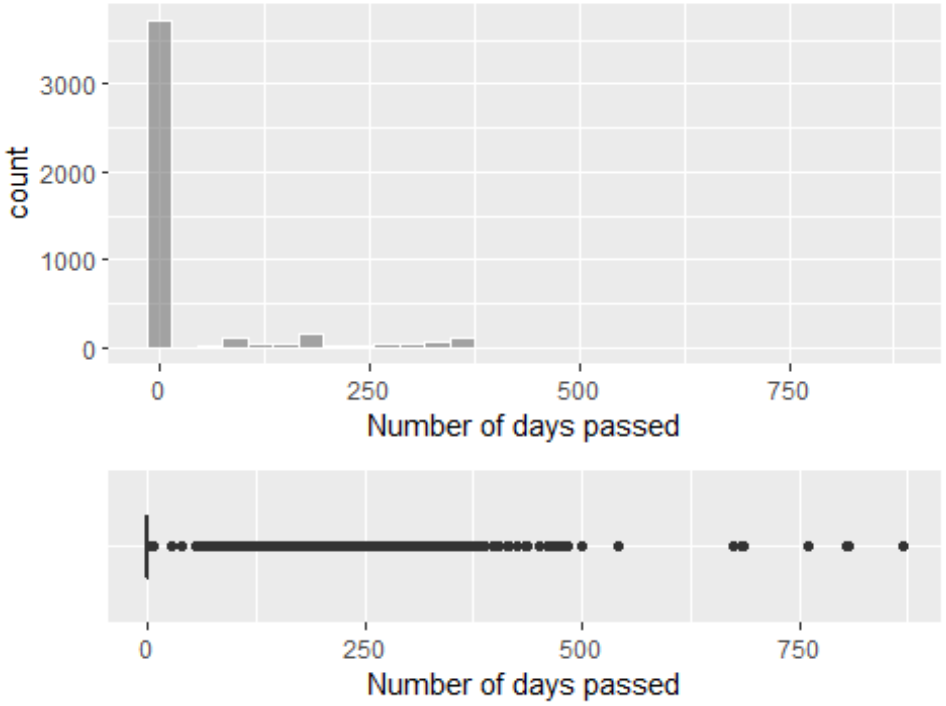
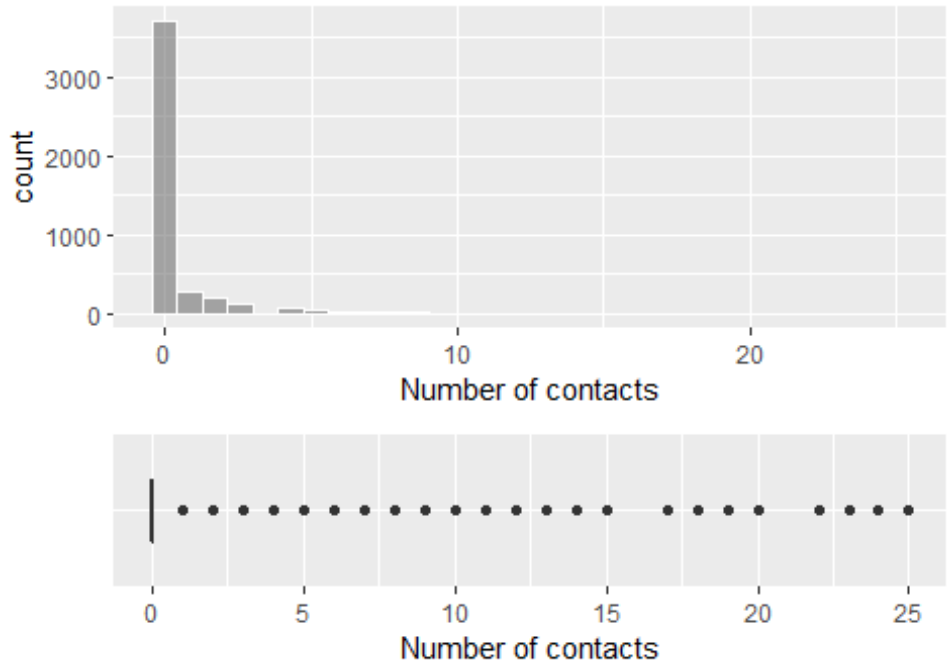


Figure 14 - Number of contacts performed before this



➢ Multivariate Visualization

Each feature of the dataset is already explored individually and now these can be explored in relation to the target feature "y" with its respective levels (Yes/No). After this,likely relationship between two probable descriptive features in relation to the target feature is explored, followed by exploring the correlation amongst all the numerical features with the help of a scatter matrix.

In the below chunk,the target feature is factorised and ordered and then divided into two separate subsets with "Yes" and "No" in order to make the further multivariate visualizations easy to plot and interpret.

```
#Factorise the target feature y
bank$y <- factor(c(bank$y), levels = c("yes","no"), ordered = TRUE)

#Divide the target feature into two seperate subsets with Yes and No
bank_yes <- bank %>% filter(bank$y =="yes")
bank_no <- bank %>% filter(bank$y =="no")
```

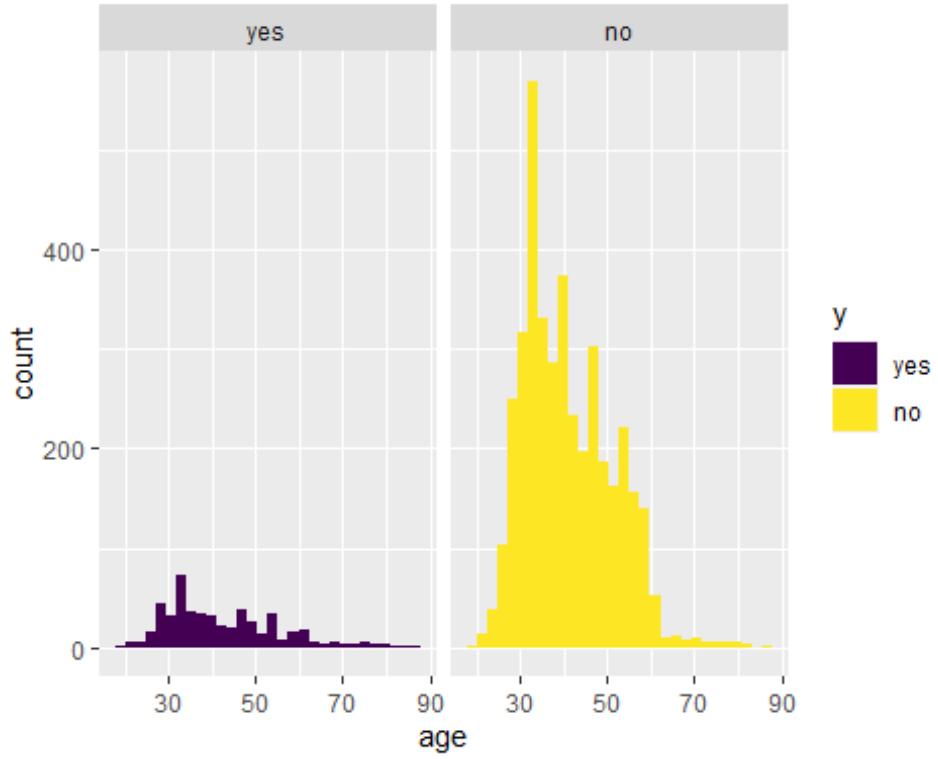
➢ Numeric Features Segregated by Target Y

First all the numeric features are explored in relation to target feature except "Pdays" and "Previous" because almost all of the clients contacted during this campaign were contacted for the first time and drawing comparison and exploring on such basis will not yield any meaningful insights. Also, majority of people were contacted only once or twice hence

exploring “campaign” might also not yield anything meaningful. Hence campaign is also not explored with respect to target feature. Later, The categorical features are explored.

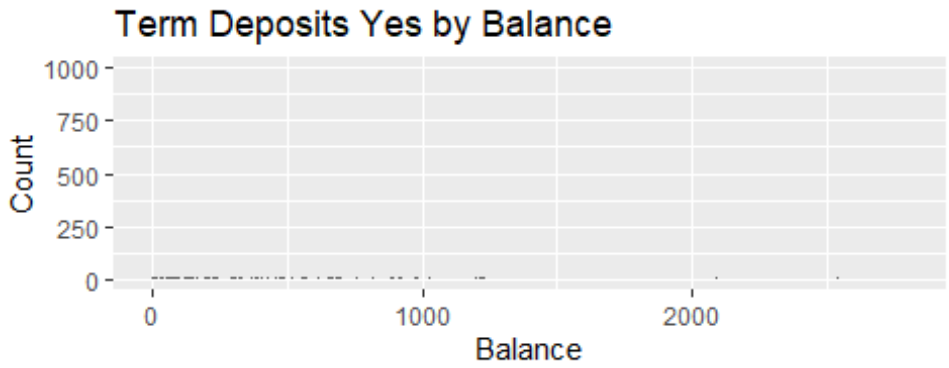
- Age

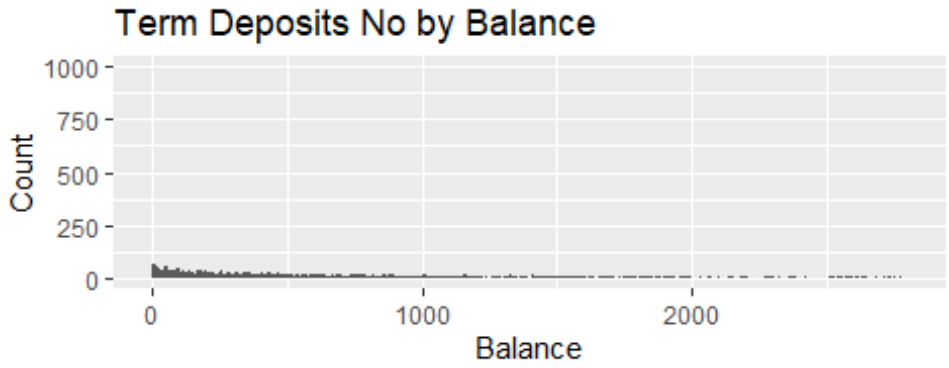
Although people from all age group are subscribing to the term deposit however people somewhere between 30 and 40 avail it the most. Interestingly, the same age group has the highest count who did not subscribe to the term deposit. This may lead to an interesting fact that this is the most sought after group due to its highest proportion in total.



- Balance

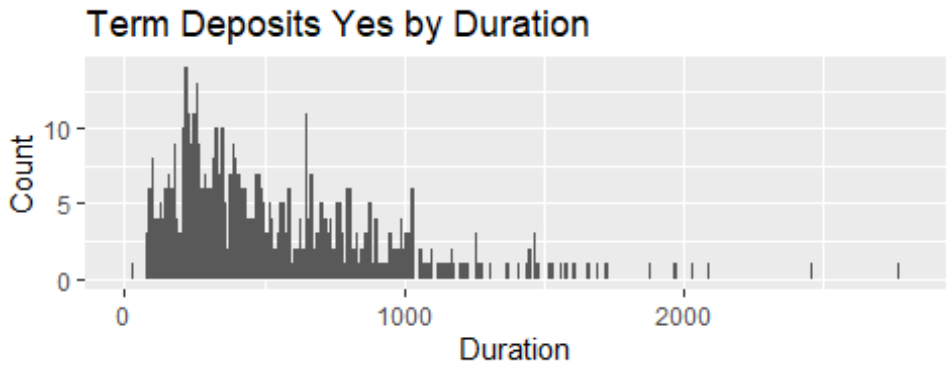
A careful comparison between both the visualizations demonstrates that people with near zero or low balance are least likely to subscribe to term deposits and very few people with low average yearly balance opt for term deposit.

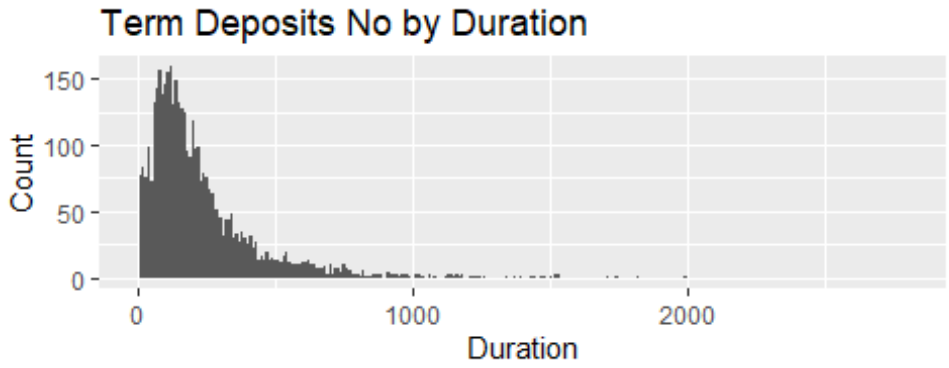




- Duration

The duration has an important bearing on the target outcome in a way that when duration is “0”, the term deposit outcome is always “No”. The other significant find is- almost all of the people who do not wish to subscribe term deposit decide in the first 8 minutes and people who wish to subscribe sometimes take little longer in getting convinced and deciding.

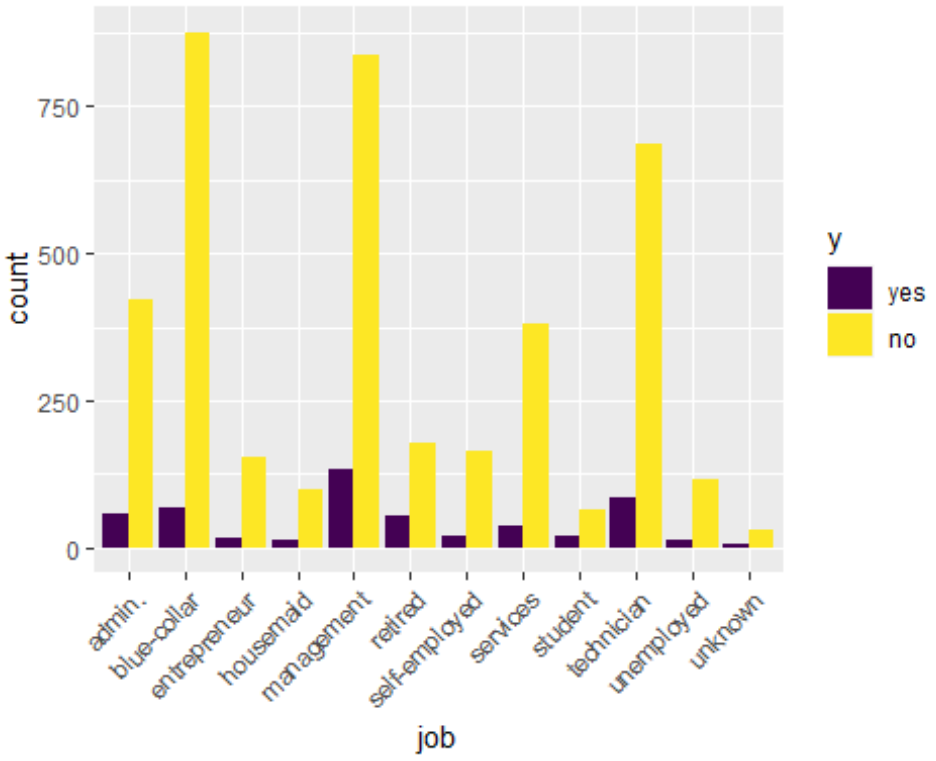




- **Categorical Features Segregated by Target Y**

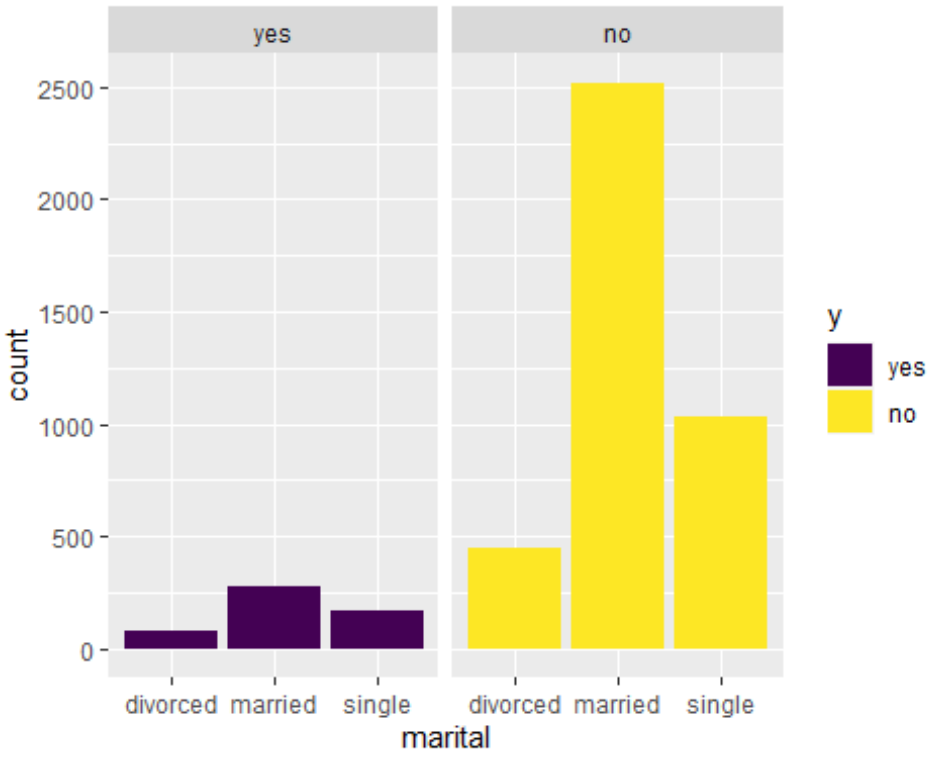
  - Job

Out of the top three jobs (on the basis of total count), people in management jobs avail the highest number of term deposits followed by technicians. However, this figure/value seems quite obvious due to their high proportion in total count.



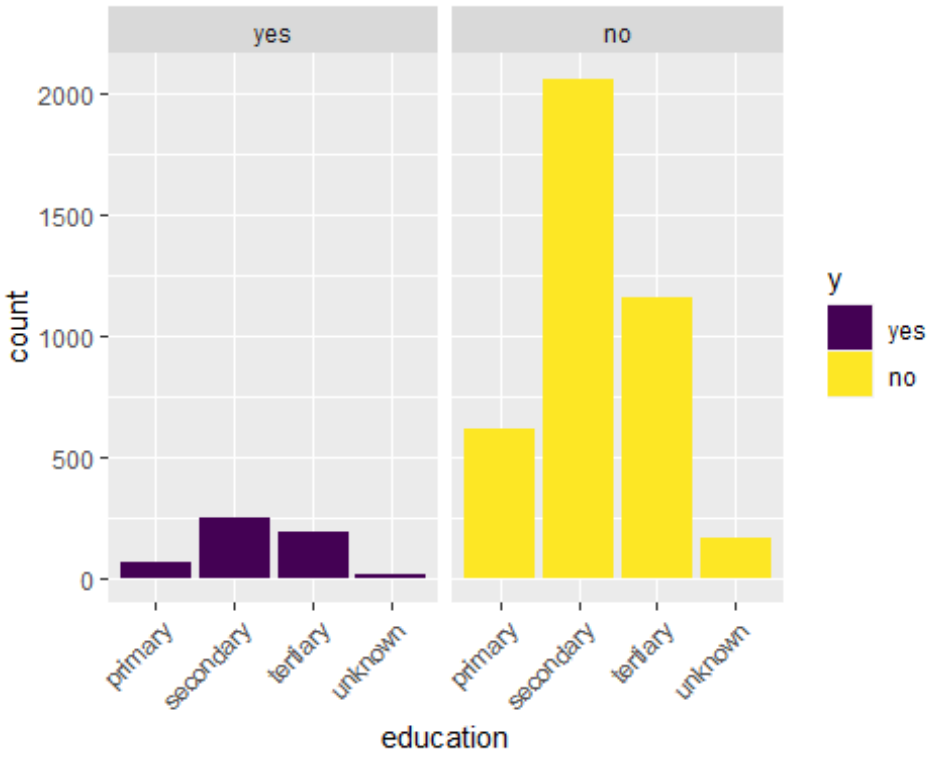
- Marital status

Due to high proportion of married people in total count, it is not surprising to see that this group tops the list in subscribing the term deposit in comparison to the other two groups.



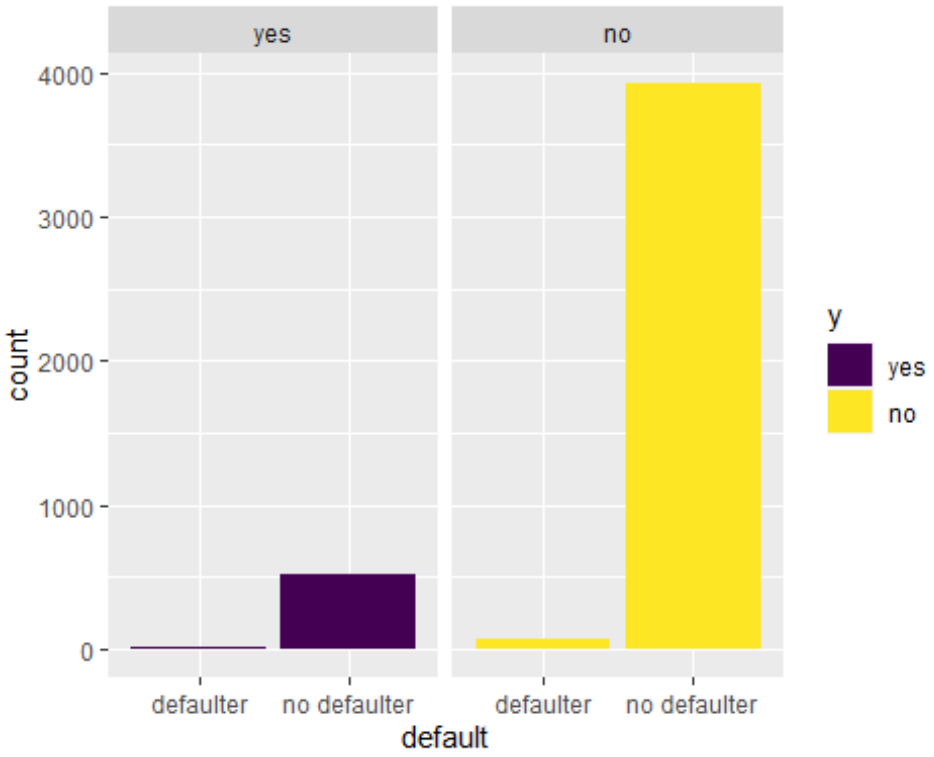
- Education

Due to large proportion of secondary education in the total count, it's not uncommon they top the chart in terms of subscribing the term deposit as well.



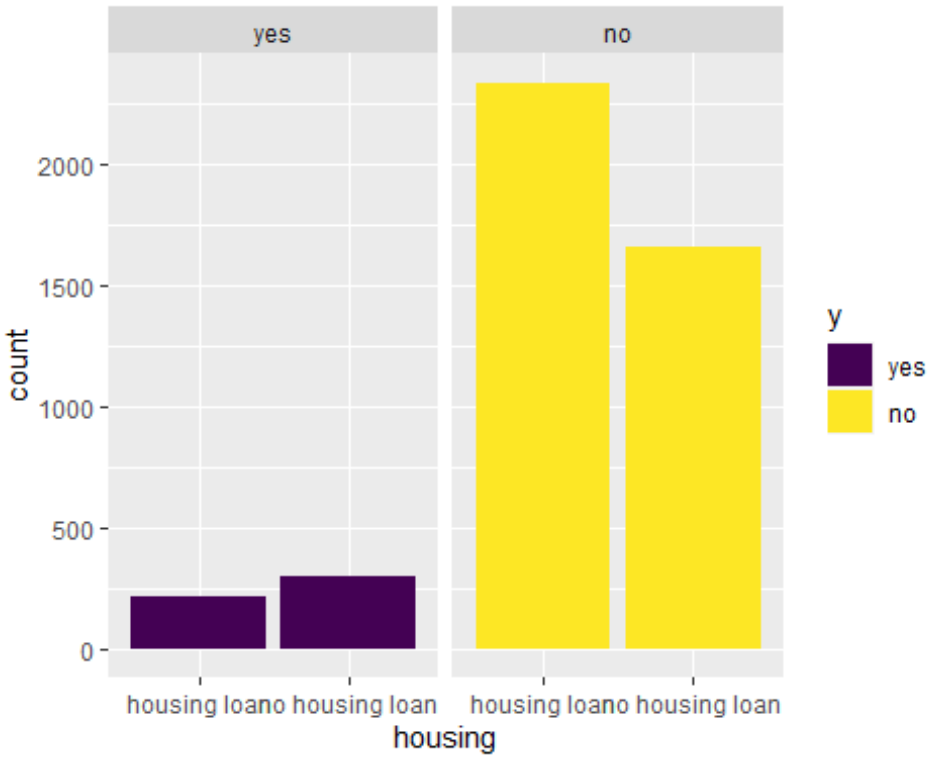
- Default

It is quite normal to see that “no defaulters” are opting for term deposit because of their high ratio in the total count as well. What is intriguing is that defaulters also subscribe to term deposit.



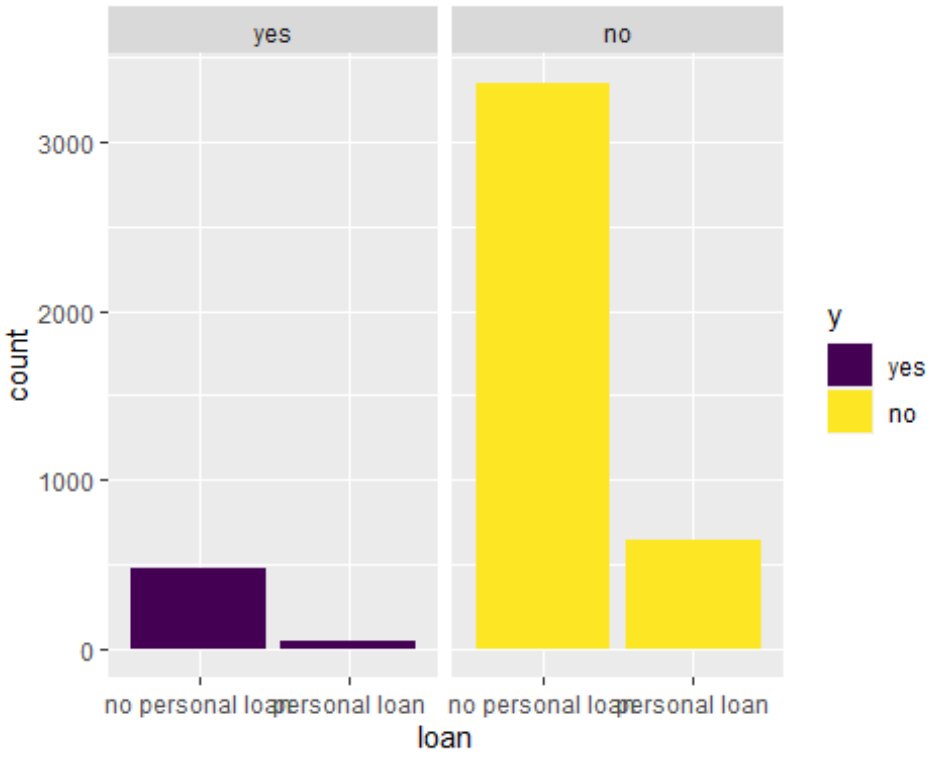
- Housing

It is fascinating to find that although housing loan has high proportion than no housing loan in the total count however their count in subscribing to term deposit is lesser than those who have not taken housing loan. It is probable that housing loan affects the propensity/inclination of a person towards investing in a term deposit.



- Loan

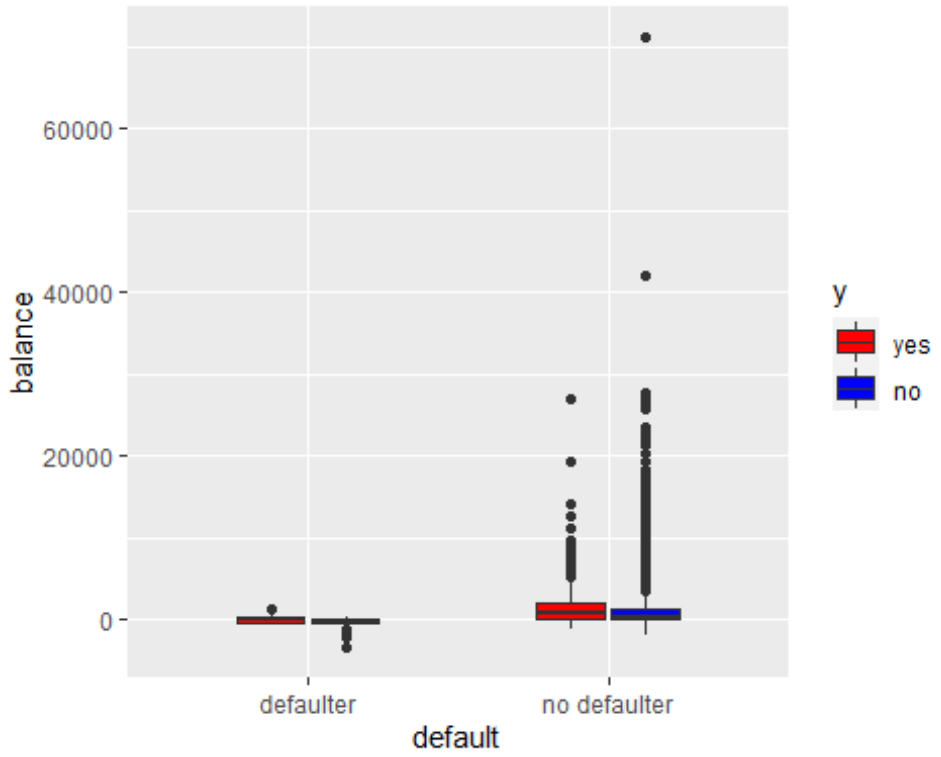
Due to large proportion of people having personal loan in the total count, It is not surprising if they are again topping the chart in terms of subscribing to term deposits. Therefore, it would be interesting to find whether a loan be it housing or personal affects a person's likelihood of subscribing to term deposit. Consequently, this aspect is further explored later in this section with other features.



➢ Interaction between Categorical and Numeric Features

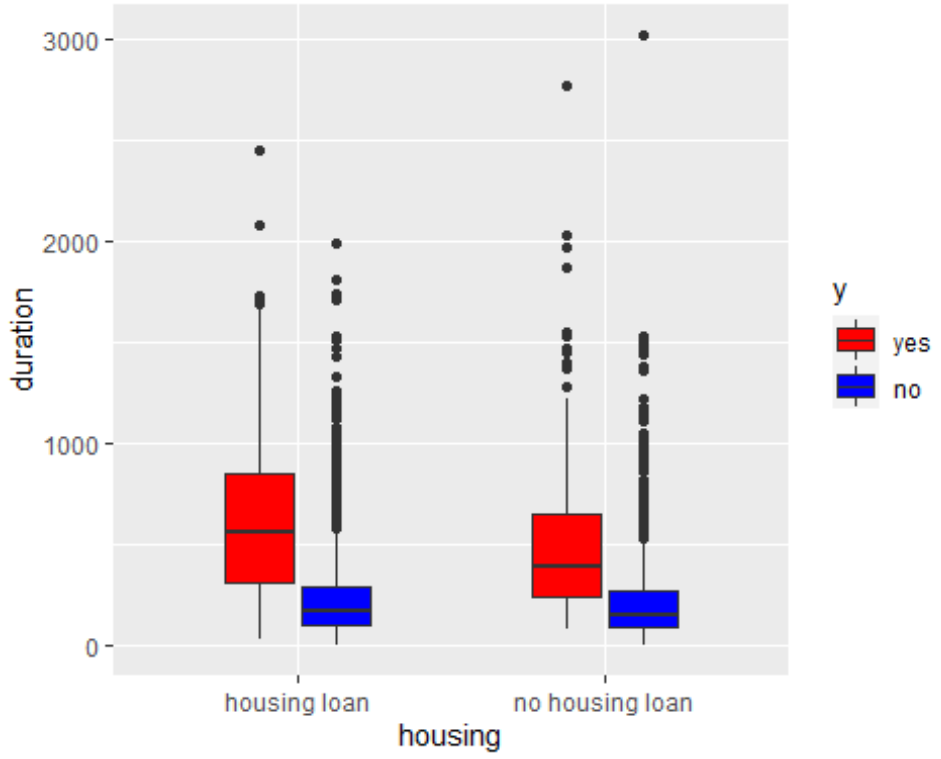
❖ Balance VS Default

It is very difficult to draw any interpretation from the “defaulter” and balance due to the distorted boxplot but it is clear that people who pay their dues on time (i.e. no defaulters) have slightly more average yearly balance and people who have more balance are more likely to subscribe to term deposits as the figure shows that the median balance for such a group is higher than the others. So,in short, we can infer from the figure that no defaulters having close to median yearly average balance are more likely to avail term deposits at the bank.



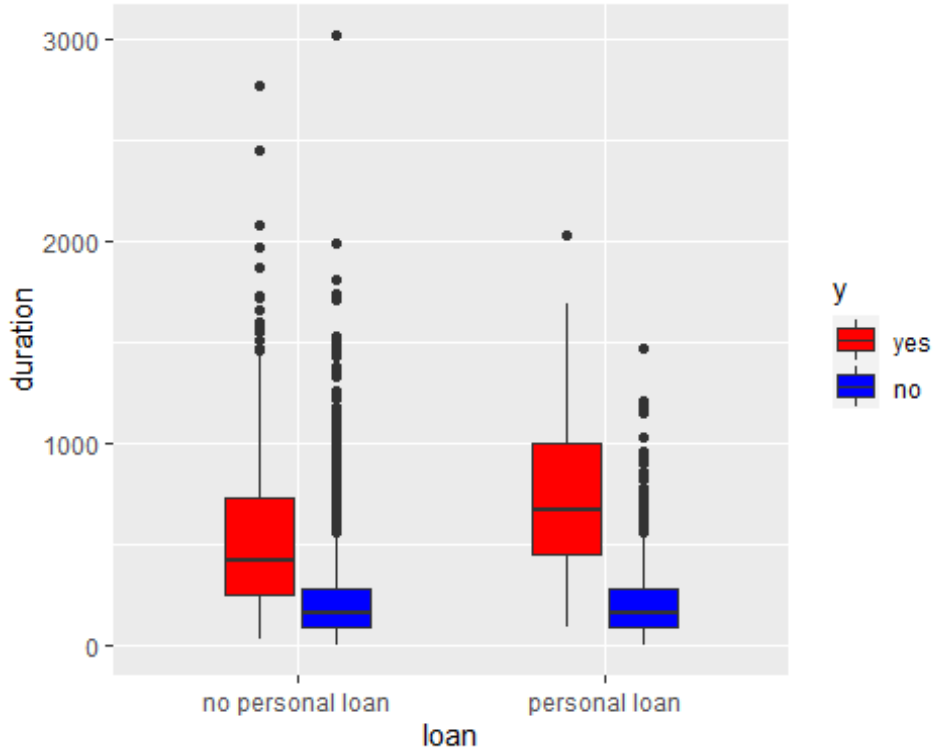
- Duration VS Housing

It is interesting to note from this figure that whether people have housing loans or not, those subscribing to term loans spend more time on the marketing phone calls than those who do not subscribe to term deposits. Also, those who already have running housing loans spend more time on call duration in deciding about the subscription than those who have no house loans.



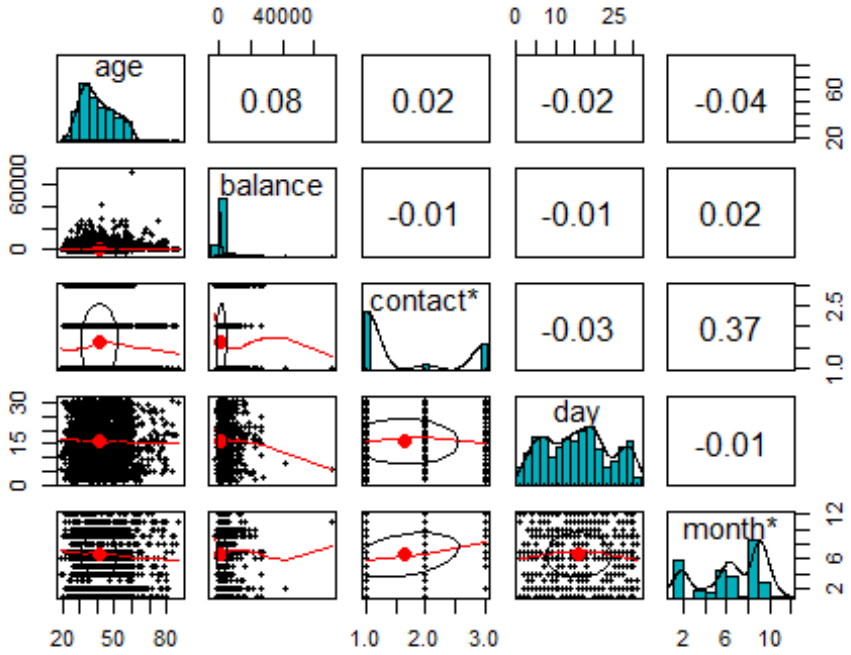
- Duration VS Loan

People already having personal loans spend more time over phone during the marketing campaign than those who have no personal loans. Also, those who have personal loans are more likely to subscribe to term deposits. It can be said, that people availing personal loans give comparatively more time on phone and generally more likely to subscribe to the term deposit. However, all these are assumptions only and can be confirmed with further investigation in this matter.



- **Scatter plot matrix of Numerical Features by Target Y**

The following scatter plot for the significant numerical features is generated using "GGally" package in R. A scatter plot matrix is a collection of scatterplots organised into a grid or matrix where each scatterplot shows a relationship between pair of variables.This scatterplot matrix shows that none of the pair of numerical features have any significant relationship between them. This can be confirmed with the correlation proportion. None of the pairs have proportion any higher than 0.09 (positive or negative), some of them are as low as 0.02, hence can be considered uncorrelated.



5. Methodology

- Building Machine Learning Algorithms

As the dataset is cleaned and preprocessed in the first phase, three machine learning framework shortlisted for this problem are discussed below. Reading and preprocessing the data for each of the three algorithms are performed separately. There are further modification and manipulation of data tailored to need of specific algorithms. Prediction check, misclassification errors and cross table performed for each model.

- Standard Logistic Regression Using Lasso Regularisation

One of the suitable approaches for this classification problem is the logistic regression algorithm as outcome variables in the data set contains binary responses. Selecting the significant variables for the model is the primary aspect of regression approach. Exploring the possibility of improving the model is another important aspect of the model building process. So, Lasso (least absolute shrinkage and selection operator) is applied to perform the variable selection and regularization to improve the prediction accuracy and interpretability of the model. It is mandatory to normalize the data set because of different range of data values. After appropriately normalization the variables using the min/max normalization method, the data set is split into the training group and testing group in the ratio of 80 to 20 percent. A standard logistic regression is built on the training set and test data set is used to get the confusion matrix. Detailed information is derived from the cross tabulation.

- K-Nearest Neighbor using Bias-Variance Trade-off

The k-nearest neighbors' algorithms(k-NN) is another sensible machine learning approach for this classification problem because K-NN algorithm is the simplest non-parametric method that we can effectively implement for classification problem. The random sample of the whole data set has been chosen to perform the algorithm. In this case, the data is split into three groups, training, validation and testing sets. Training and validation data for different values of k is used to run the nearest neighbours algorithm. The value of k for final K-NN classifier has been picked out from the bias-variance trade-off plot between training and validation sets. The algorithm is then applied to the test set to get the confusion matrix and misclassification error rate.

- Naïve Bayes Classifier

Naïve Bayes classifier is a probability-based classifier for a classification machine learning problem. It is based on Bayes theorem with an assumption of independence between varibles. The response variable to be predicted here is classed as "yes" and "No" and fundamental nature of predictors appear to be relatively independent of each other. Moreover, because our training set is relatively small, the possibility of having noisy and unknown data is there, so this approach stands to be suitable. Another advantage of Naïve Bayes Classifier is that the probability of a prediction can be easily calculated. Also, a diagnostic analysis of the model is performed before any conclusion on the model is made.

- Performance Analysis of Algorithms

Primary means of analysing the algorithms are using the misclassification error rate of prediction on the test data set. The model with least misclassification error represents a model with better accuracy. Definitely, the model with small misclassification error is chosen as the final model from the three algorithms. Overall adequacy of the algorithms was also performed.

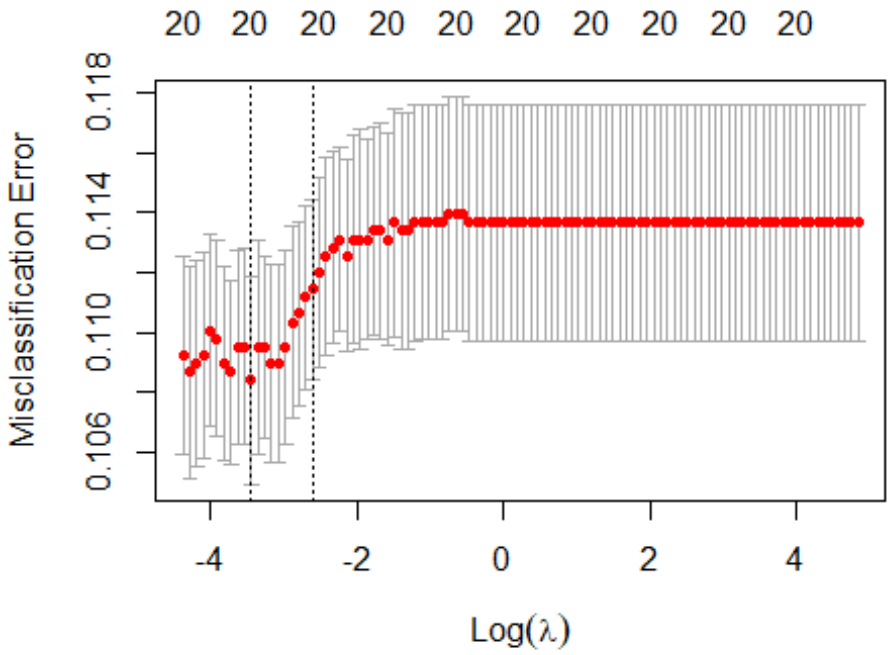
6. Model Building and Predictions

- Logistic Regression Model

It needs to be taken care to ensure that the response/dependent variable is dichotomous (or binary) in order to apply logistic regression algorithm. As already discussed the response variable in this problem is to predict either "yes" or "no" for a term deposit. The character variables are converted into numeric and also to include the unknown responses of the attributes in the model, four new variables are created. The following chunks of code load the required library to perform the logistic regression.

First of all, we build the Ridge Regrssion. This helps to interpret the data and suggest for further necessity of regularization.

```
## [1] 0.03201287
```



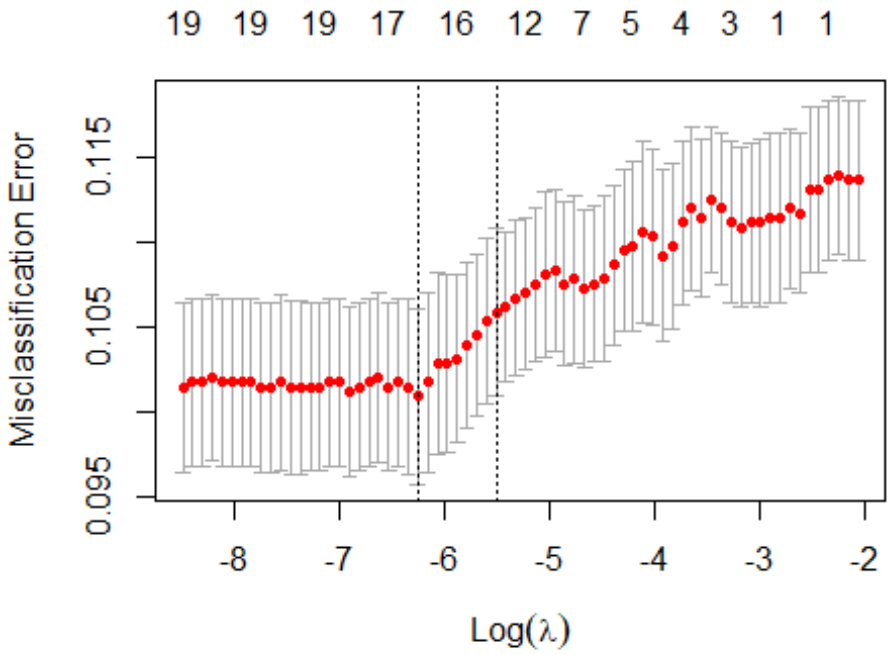
When lambda = 0, the penalty term has no effect, and ridge regression will produce the least squares estimates. Since, our minimumm value of the lambda is tends to zero. Thus, from the above plot it is conformed that we do not need to do the regularization.

The following code manipulates the coeffiecient of the ridge regression

Now we perform regression using LASSO setting alpha to be 1.

```
## Warning: executing %dopar% sequentially: no parallel backend registered
```

```
## [1] 0.001919121
```



Clearly the lasso leads to qualitatively similar behavior to ridge regression. Since best vallue of value of lambda is still close to zero. thus from the above plot of misclassification error versus lambda, it is conformed that we do not need to do the regularization.

The following code manipulates the coeffiecient of the lasso regression

```
##             as.vector(coef(bank.lasso, s = bank.lasso$lambda.min))
## (Intercept)                                                -3.23675991
## age                                                         0.48035031
## job                                                         0.18016337
## marital                                                     0.08710205
## education                                                   0.76444055
## default                                                     0.35433688
## balance                                                     0.45458744
## housing                                                    -0.55985098
## loan                                                       -0.66222434
## contact                                                     0.00000000
## day                                                        -0.01334521
## month                                                      -0.04136417
## duration                                                   11.96483748
## campaign                                                   -3.42227170
## pdays                                                       0.00000000
## previous                                                    0.00000000
## poutcome                                                    2.53136783
## job_unk                                                     0.34333149
## edu_unk                                                    -0.93180452
```

```
## cont_unk                                                    -0.86350944
## pout_unk                                                    -2.84544751
```

Get the features with the non zero lasso coefficient. From the above output it is observed that all the above predictors would be important to perform the logistic regression. The following code manipulates the data into the new matrix with the nonzero features and re split the data into training and test sets.

```
##  [1] "(Intercept)" "age"         "job"         "marital"     "education"
##  [6] "default"     "balance"     "housing"     "loan"        "day"
## [11] "month"       "duration"    "campaign"    "poutcome"    "job_unk"
## [16] "edu_unk"     "cont_unk"    "pout_unk"
```

From the above output it is observed that all the above predictors would be important to perform the logistic regression. The following code manipulates the data into the new matrix with the nonzero features and re split the data into training and test sets.

Now standard logistic regression is run using non zero features identified by a LASSO.

```
##
## Call:
## glm(formula = Y ~ ., family = binomial(link = "logit"), data = bank_1$trai
ning)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.3004  -0.4102  -0.2836  -0.1817   3.0490
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.70317    0.40831  -9.070  < 2e-16 ***
## age          0.79812    0.43130   1.850 0.064243 .
## job          0.22922    0.22431   1.022 0.306842
## marital      0.19991    0.22701   0.881 0.378518
## education    1.00002    0.29707   3.366 0.000762 ***
## default      0.56851    0.44309   1.283 0.199466
## balance      0.78616    1.37516   0.572 0.567534
## housing     -0.55092    0.13525  -4.073 4.64e-05 ***
## loan        -0.75548    0.21144  -3.573 0.000353 ***
## day         -0.08998    0.23227  -0.387 0.698456
## month       -0.09447    0.22576  -0.418 0.675608
## duration    12.48137    0.67119  18.596  < 2e-16 ***
## campaign    -4.29522    1.50092  -2.862 0.004213 **
## poutcome     3.30396    0.43096   7.666 1.77e-14 ***
## job_unk      0.51796    0.53144   0.975 0.329746
## edu_unk     -1.30845    0.40055  -3.267 0.001088 **
## cont_unk    -0.93365    0.19545  -4.777 1.78e-06 ***
## pout_unk    -3.43934    0.34246 -10.043  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2560.9  on 3615  degrees of freedom
## Residual deviance: 1850.3  on 3598  degrees of freedom
## AIC: 1886.3
##
## Number of Fisher Scoring iterations: 6

##  [1] "(Intercept)" "age"         "job"         "marital"     "education"
##  [6] "default"     "balance"     "housing"     "loan"        "day"
## [11] "month"       "duration"    "campaign"    "poutcome"    "job_unk"
## [16] "edu_unk"     "cont_unk"    "pout_unk"
```

Prediction and misclassification of the model

```
## [1] 0.1049724
```

Confusion matrix from the test data

```
##
## predictions   0   1
##           0 771  71
##           1  24  39
```

The confusion matrix with a more infomative outputs offered by CrossTable() in gmodels package helps analyse the prediction accuracy of the model. The output table includes proportion in each cell that tells the percentage of table's row, column or overall total counts on the class of the response variable.

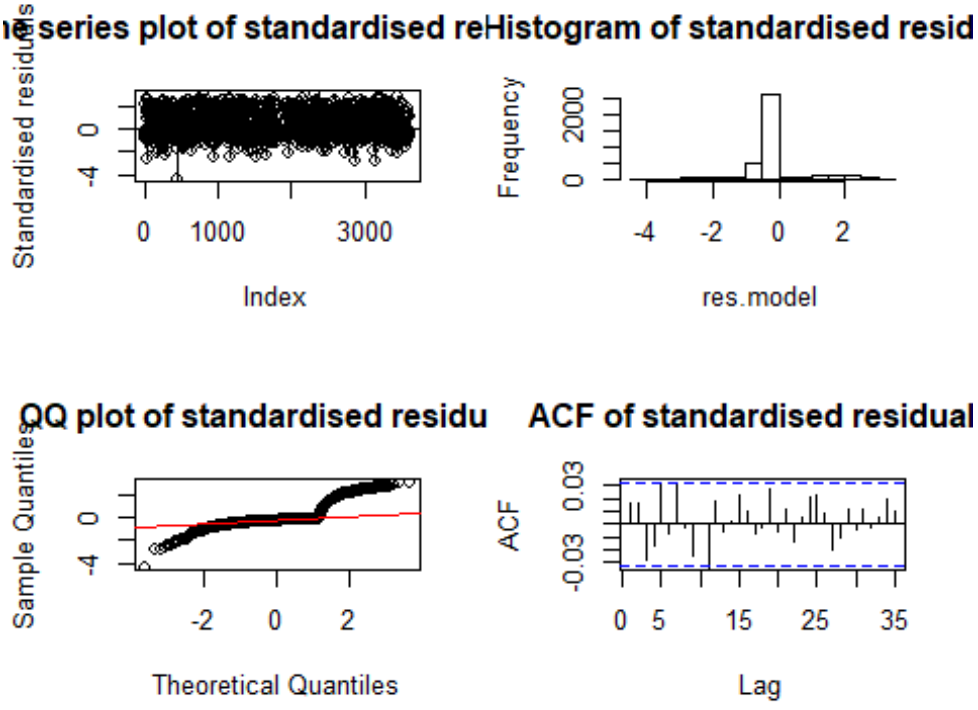
From the cross table below it is observed that using seed as 45, 92% of people not subscribing the term deposit in the data set is predicted correctly while 65% of people subscribing term deposit is predicted correctly. Thus, from the confusion matrix or Cross table we can safely say that the model perfoms well to predict the customer subscribe term deposit with misclassification error of 9%.

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  905
##
##
##              | bank_1$test$Y
```

```
##  predictions |         0 |         1 | Row Total |
## -------------|-----------|-----------|-----------|
##            0 |       771 |        71 |       842 |
##              |     0.916 |     0.084 |     0.930 |
##              |     0.970 |     0.645 |           |
##              |     0.852 |     0.078 |           |
## -------------|-----------|-----------|-----------|
##            1 |        24 |        39 |        63 |
##              |     0.381 |     0.619 |     0.070 |
##              |     0.030 |     0.355 |           |
##              |     0.027 |     0.043 |           |
## -------------|-----------|-----------|-----------|
## Column Total |       795 |       110 |       905 |
##              |     0.878 |     0.122 |           |
## -------------|-----------|-----------|-----------|
##
##
```

- ❖ Residual Analysis

The follwing code builds the function to perform residual analysis which can be used to do residual checks for all the three models.




```
##
##  Shapiro-Wilk normality test
##
```

```
## data:  res.model
## W = 0.68155, p-value < 2.2e-16
```

#Durbin-Watsson test for autocorrelation of residuals

```
##  lag Autocorrelation D-W Statistic p-value
##    1      0.06040692      1.878892       0
##  Alternative hypothesis: rho != 0

##       age       job   marital education   default   balance   housing
loan
##  1.376018  1.114455  1.262253  1.346433  1.012701  1.036235  1.190281  1.0
34793
##       day     month  duration  campaign  poutcome   job_unk   edu_unk  con
t_unk
##  1.037085  1.160230  1.085897  1.063671  6.166137  1.131444  1.346855  1.2
30301
##  pout_unk
##  6.219546
```

➢ K-Nearest Neighbor Model

The following chunk of codes select the random sample of size 500 from the population and split the data in to training, validation and test sets.

- ❖ K-NN Prediction and misclassification Error

On the validation set, initially a random value of k=3 is chosen to make the prediction of the algorithm. Also, to see the accuracy of the model the misclassification error is compouted. The computations are demonstrated by the following codes.

```
## [1] 0.37
```

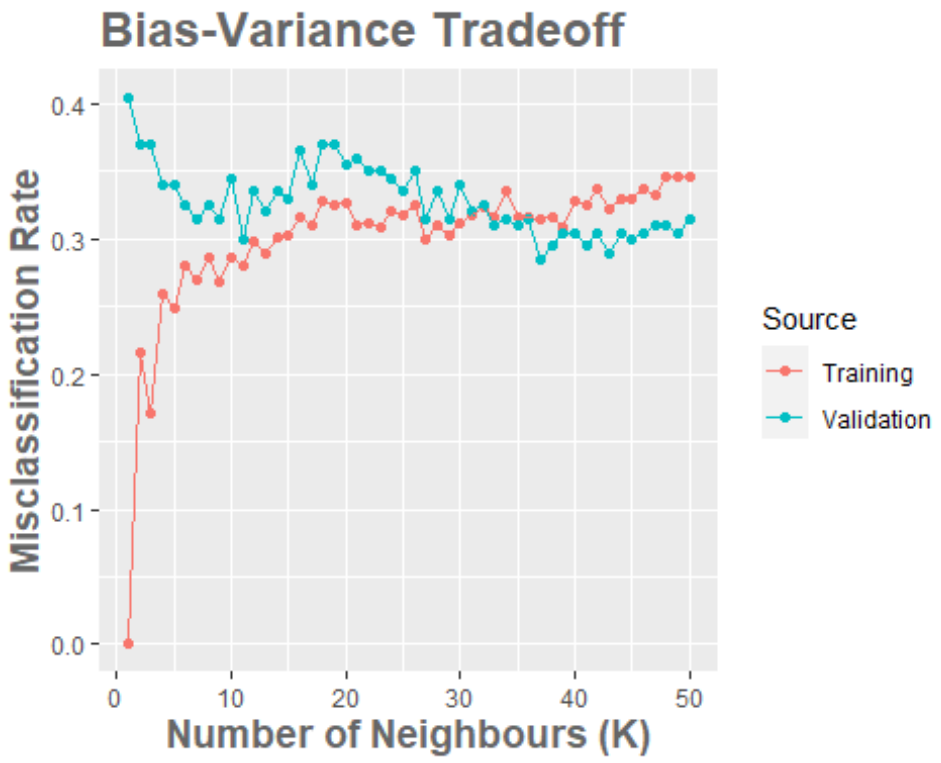
The code in the for loop will create a data frame in order to plot the bias-variance tradeoff.

```
## iteration:  FALSE 1
## iteration:  FALSE 2
## iteration:  FALSE 3
## iteration:  FALSE 4
## iteration:  FALSE 5
## iteration:  FALSE 6
## iteration:  FALSE 7
## iteration:  FALSE 8
## iteration:  FALSE 9
## iteration:  FALSE 10
## iteration:  FALSE 11
## iteration:  FALSE 12
## iteration:  FALSE 13
## iteration:  FALSE 14
## iteration:  FALSE 15
## iteration:  FALSE 16
```

```
## iteration:  FALSE 17
## iteration:  FALSE 18
## iteration:  FALSE 19
## iteration:  FALSE 20
## iteration:  FALSE 21
## iteration:  FALSE 22
## iteration:  FALSE 23
## iteration:  FALSE 24
## iteration:  FALSE 25
## iteration:  FALSE 26
## iteration:  FALSE 27
## iteration:  FALSE 28
## iteration:  FALSE 29
## iteration:  FALSE 30
## iteration:  FALSE 31
## iteration:  FALSE 32
## iteration:  FALSE 33
## iteration:  FALSE 34
## iteration:  FALSE 35
## iteration:  FALSE 36
## iteration:  FALSE 37
## iteration:  FALSE 38
## iteration:  FALSE 39
## iteration:  FALSE 40
## iteration:  FALSE 41
## iteration:  FALSE 42
## iteration:  FALSE 43
## iteration:  FALSE 44
## iteration:  FALSE 45
## iteration:  FALSE 46
## iteration:  FALSE 47
## iteration:  FALSE 48
## iteration:  FALSE 49
## iteration:  FALSE 50
```

- ❖ Bias- Variance Trade-off

Following the execution of the above code the bias variance can be ploted as follows.It is clear that as the number of neighburs (k) increases, the misclassification error increases and stabilizes between 30 to 40 percent. The validation set also shows that the error rate is steady within similar range as the training set. Also, from the plot it appears the “best” value of K is between 4 or 8. We choose 4 as it is a simpler model and this will be used it to score our test set



- Misclassification error in the test set

```
## [1] 0.37
```

Confusion matrix from test set

```
##
## knn.pred3  0  1
##         0 78 52
##         1 22 48
```

Cross Table will return the confusion matrix with more infomation where we can get the proportion of each cell value making easy coparision. From the following cross table it is observed that the model predicted 62% of non-subscription of term deposit correctly and 66% of the subscription of term deposit correctly.

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |         N / Table Total |
## |-------------------------|
##
##
```

```
## Total Observations in Table:  200
##
##
##              | final_sample.Y$test
##    knn.pred3 |         0 |         1 | Row Total |
## -------------|-----------|-----------|-----------|
##            0 |        78 |        52 |       130 |
##              |     0.600 |     0.400 |     0.650 |
##              |     0.780 |     0.520 |           |
##              |     0.390 |     0.260 |           |
## -------------|-----------|-----------|-----------|
##            1 |        22 |        48 |        70 |
##              |     0.314 |     0.686 |     0.350 |
##              |     0.220 |     0.480 |           |
##              |     0.110 |     0.240 |           |
## -------------|-----------|-----------|-----------|
## Column Total |       100 |       100 |       200 |
##              |     0.500 |     0.500 |           |
## -------------|-----------|-----------|-----------|
##
##
```

- **Naive Bayes Model**

Loading the data and Preprocessing The following chunks of code load the required library to perfrom the Naive Bayes regression

```
## [1] 3140   21
```

```
## [1] 1381   21
```

The following code returns the percentage of customer subscribing term deposit from whole data set

```
## [1] 0.1115134
```

Only eleven percentage of people in our test set suscribe the term deposite. Let's see how much will predict by our model.

Lets Employ the naive Bayes function to build the classifier

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = trainset[, !names(trainset) %in% c("y")],
##     y = trainset$y, na.action = na.pass)
##
## A-priori probabilities:
## trainset$y
##       no      yes
```

```
## 0.883121 0.116879
##
## Conditional probabilities:
##           age
## trainset$y     [,1]     [,2]
##        no  40.94266 10.17058
##        yes 42.31608 12.89802
##
##           job
## trainset$y     [,1]     [,2]
##        no  5.409304 3.272852
##        yes 5.509537 3.128276
##
##           marital
## trainset$y     [,1]      [,2]
##        no  2.148936 0.5889462
##        yes 2.177112 0.6642464
##
##           education
## trainset$y     [,1]      [,2]
##        no  2.223585 0.7524317
##        yes 2.269755 0.7248408
##
##           default
## trainset$y       [,1]      [,2]
##        no  0.01767039 0.1317741
##        yes 0.01907357 0.1369704
##
##           balance
## trainset$y     [,1]     [,2]
##        no  1349.691 2813.552
##        yes 1610.853 2601.014
##
##           housing
## trainset$y      [,1]      [,2]
##        no  0.5791561 0.4937836
##        yes 0.4277929 0.4954341
##
##           loan
## trainset$y       [,1]      [,2]
##        no  0.15831230 0.3650994
##        yes 0.09264305 0.2903274
##
##           contact
## trainset$y     [,1]      [,2]
##        no  1.694194 0.9165538
##        yes 1.326975 0.6870443
##
##           day
## trainset$y     [,1]     [,2]
```

```
##        no  15.86441 8.181408
##        yes 15.47411 8.151487
##
##           month
## trainset$y     [,1]     [,2]
##        no  6.610530 2.949636
##        yes 6.427793 3.430750
##
##           duration
## trainset$y     [,1]     [,2]
##        no  227.5662 218.0825
##        yes 537.3542 377.1710
##
##           campaign
## trainset$y     [,1]     [,2]
##        no  2.843851 3.170446
##        yes 2.141689 1.972926
##
##           pdays
## trainset$y     [,1]      [,2]
##        no  37.09809  96.73482
##        yes 69.76839 122.61713
##
##           previous
## trainset$y      [,1]     [,2]
##        no  0.4835918 1.626186
##        yes 1.1389646 2.166507
##
##           poutcome
## trainset$y     [,1]      [,2]
##        no  3.571944 0.9990792
##        yes 3.305177 1.0633232
##
##           job_unk
## trainset$y        [,1]       [,2]
##        no  0.009376127 0.09639277
##        yes 0.008174387 0.09016495
##
##           edu_unk
## trainset$y       [,1]      [,2]
##        no  0.04543815 0.2083007
##        yes 0.03269755 0.1780866
##
##           cont_unk
## trainset$y      [,1]      [,2]
##        no  0.3137396 0.4640956
##        yes 0.1253406 0.3315567
##
##           pout_unk
## trainset$y      [,1]      [,2]
```

```
##        no  0.8362784 0.3700895
##        yes 0.6348774 0.4821218
```

The priori and conditional probabilities has been observed from the above outputs.

Rename the data (predictors) to performe the prediction

- ❖ Prediction and misclassification Error

```
##      y
##         no  yes
##   no  1055   76
##   yes  172   78

## [1] 0.17958
```

We got about 17% misclassification error while doing the prediction of customer suscribe to term deposit.

— Confusion matrix

```
## Confusion Matrix and Statistics
##
##      y
##         no  yes
##   no  1055   76
##   yes  172   78
##
##                Accuracy : 0.8204
##                  95% CI : (0.7991, 0.8403)
##     No Information Rate : 0.8885
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.2879
##
##  Mcnemar's Test P-Value : 1.614e-09
##
##             Sensitivity : 0.8598
##             Specificity : 0.5065
##          Pos Pred Value : 0.9328
##          Neg Pred Value : 0.3120
##              Prevalence : 0.8885
##          Detection Rate : 0.7639
##    Detection Prevalence : 0.8190
##       Balanced Accuracy : 0.6832
##
##        'Positive' Class : no
##
```

The confusion matrix above shows the 82% of acuracy and 95% confidence interval of the perdicted acuracy. The cross table from the confusion matrix shows that model predicted

more closely for the customer who did not suscribe term deposit but for thoese custome who had suscribed term deposit has been predicted badly.

7. Results

As already seen the three models we have built have their own accuracy of predicting whether a client will say “yes” or “no” to a term deposit of the bank. As expected there is some variation in the misclassification error rate among the three classification algorithm.

➢ Algorithms Error Rate

- Logistic Regression Model 9.40%
- K-NN classifier 36.5%
- Naive Bayes Classifier 17.95%

Based on the misclassification error rate, the most reliable model for the data set appears to be the logistic regression model with just 9.4%.

➢ Adequacy of Logistic Regression Model

❖ Residual Analysis

The residual analysis includes a check for normality, autocorrelation and time series plot to inspect if there is any trend present in the residuals. As per the previous output in the regression algorithm section, there are no autocorrelation parts in this model confirming a white noise process. However, the residuals are not purely normally distributed which is seen from Shapiro-Wilk test and histogram of the standardized residuals. Time series plot of residuals indicates residuals have almost equal change in variances and non- existence of trends.

❖ Variable Inflation Factor (VIF) Test for Multicollinearity of Variables

Presence of collinearity among the variables negatively affects logistic regression model. The measure of VIF for variables greater than 5 is usually considered to create collinearity. From the output, it is seen almost all the variables have VIF less than 5 which proves that logistic regression is not influenced by collinearity. There are few attributes with VIF greater than 5, but they are not significant to the model.

❖ Durbin-Watson test of the model

Durbin-Watson is another test to find the effect of autocorrelation in a data set. The appropriate hypothesis for this test is H0: Errors (residuals) are uncorrelated H1: Errors (residuals) are correlated

Since the p-value is less than 0.05 we have enough evidence to reject H0. This implies that the residuals are correlated.

8. Advantages & Limitations

The advantage of logistic regression is that it is easy to interpret, it directs model logistic probability, and provides a confidence interval for the result. However, the main drawback of the logistic algorithm is that it suffers from multicollinearity and, therefore, the explanatory variables must be linearly independent. But, it is certain that the variables of the model do not exhibit multicollinearity as shown by VIF test.

Some limitations of logistic regression approach in context to the above model are

— The model has some class of unknown predictors which are significant in the model. These variables actually do not carry any useful information fundamentally but their significance might affect the predictability of the model.

— Residuals of the model are not normally distributed when doing the residual analysis are not reliable though the model demonstrates a good accuracy.

— Residuals are correlated in the Durbin-Watson test. The test for autocorrelation using the Durbin-Watson test proved that the residuals are correlated. This shows that the residuals have an autocorrelation effect which might accept the model's accuracy.

9. Conclusion

In this project, the dataset chosen has 16 descriptive features and 1 target feature. All features are taken into account except "Contact", "day" and "month" as they are not contributing anything significant to the outcome. Later, after exploring the features individually it is found that other attributes -"campaign", "poutcome", "pday" and"previous" are related to the previous campaigns and have no significant bearing on the outcome/target feature of the current campaign as almost all of the people contacted during this campaign are new. Consequently, these were also omitted from multivariate exploration. The data Chosen is found to have no missing values, typo errors, case errors or extra white spaces after checking thoroughly during data preprocessing. The dataset has outliers in almost all the numerical attributes which can be seen during the individual visualizations. However, these were not removed or imputed because of two reasons: First, these outliers were found to be a part of the dataset and not just random figures and removing them would have completely modified the data. Secondly, for removing or handling any outliers(if they are in significant proportion or otherwise), subject matter expertise is needed and due to lack of desired domain expertise/knowledge it was decided to proceed without handling them. The dataset is explored to dive deep and gain meaningful insights from data that can be considered and attended to during model building .

From the study conducted, the results are impressive and convincing in terms of using a machine learning algorithm to decide on the marketing campaign of the bank. Almost all of the attributes contribute significantly to the building of a predictive model. Among the three classification approach used to model the data, the logistic regression model yielded

the best accuracy with just 9.4% misclassification error rate. This model is simple and easy to implement.

The bank marketing manager can identify the potential client by using the model if the client's information like education, housing loan, Personal loan, duration of the call, number of contacts performed during this campaign, previous outcomes, etc is available. This will help in minimizing the cost to the bank by avoiding to call customers who are unlikely to subscribe the term deposit. They can run a more successful telemarketing campaign using this model.

10. References

[1] UCI Machine Learning Repository, Bank Marketing Data Set viewed online on (https://archive.ics.uci.edu/ml/datasets/Bank+Marketing)

[2] Rafael A. Irizarry (2020), Introduction to Data Science: Data Analysis and Prediction Algorithms with R

[3] https://www.edx.org/professional-certificate/harvardx-data-science↩

[4] An Introduction to Statistical Learning: With Applications in R by Gareth M. James, Daniela Witten, Trevor Hastie, Robert Tibshirani