




APPLIED PREDICTIVE ANALYTICS: FINAL PROJECT

BANK TELEMARKETING

BY GROUP 9:

ALEXANDER AGUILAR
VINNAKOTA SHASHANK
VENKATA SRI ATHULYA GOPISHETTY
NAGA CHARITHA SADINENI
DHURUVEETH PABBA



Introduction

Data can be seen everywhere, and businesses want to have data in order to find information on their customers. One way that a lot of businesses gain further data and try to persuade a customer to buy their product is through telemarketing. Tele-marketing is where a business employee calls the customer and informs them of a product available to them to purchase. They can also find out other related information to that customer and store data on this call. For this project, the focus is a dataset provided by the Portuguese bank. The Portuguese bank performed their telemarketing campaign through the time period of May 2008 to November 2010. This dataset was stored in the UCI Machine Learning Repository. There is a data folder that contained two datasets. These files were named “bank-additional-full.csv” and “bank-additional.csv”. The first file contains 41,118 records and has 21 attributes. This dataset can be the primary dataset used for cross-validation. The other file is for testing and that contains 4,119 with the 21 attributes.

To delve into the analysis, an individual must fully understand all attributes that represent the records in the dataset.

The attributes descriptions provided by the UCI Machine Learning Repository.

Input variable	Description
age	age of client (numeric)
job	type of job (categorical: 'admin.', 'blue-collar', 'enterprenuer', '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: 'basic.4y', 'basic.6y', 'basic.9y', 'high.school', 'illiterate', 'professional.course', 'university.degree', 'unknown')
default	has credit in default? (categorical: 'no','yes','unknown')
housing	has housing loan? (categorical: 'no','yes','unknown')
loan	has personal loan? (categorical: 'no','yes','unknown')
contact	contact communication type (categorical: 'cellular','telephone')
month	last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')
day_of_week	last contact day of the week (categorical: 'mon','tue','wed','thu','fri')
duration	last contact duration, in seconds (numeric). 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 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.
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; 999 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: 'failure','nonexistent','success')
emp.var.rate	employment variation rate - quarterly indicator (numeric)
cons.price.idx	consumer price index - monthly indicator (numeric)
cons.conf.idx	consumer confidence index - monthly indicator (numeric)
euribor3m	euribor 3-month rate - daily indicator (numeric)
nr.employed	number of employees - quarterly indicator (numeric)

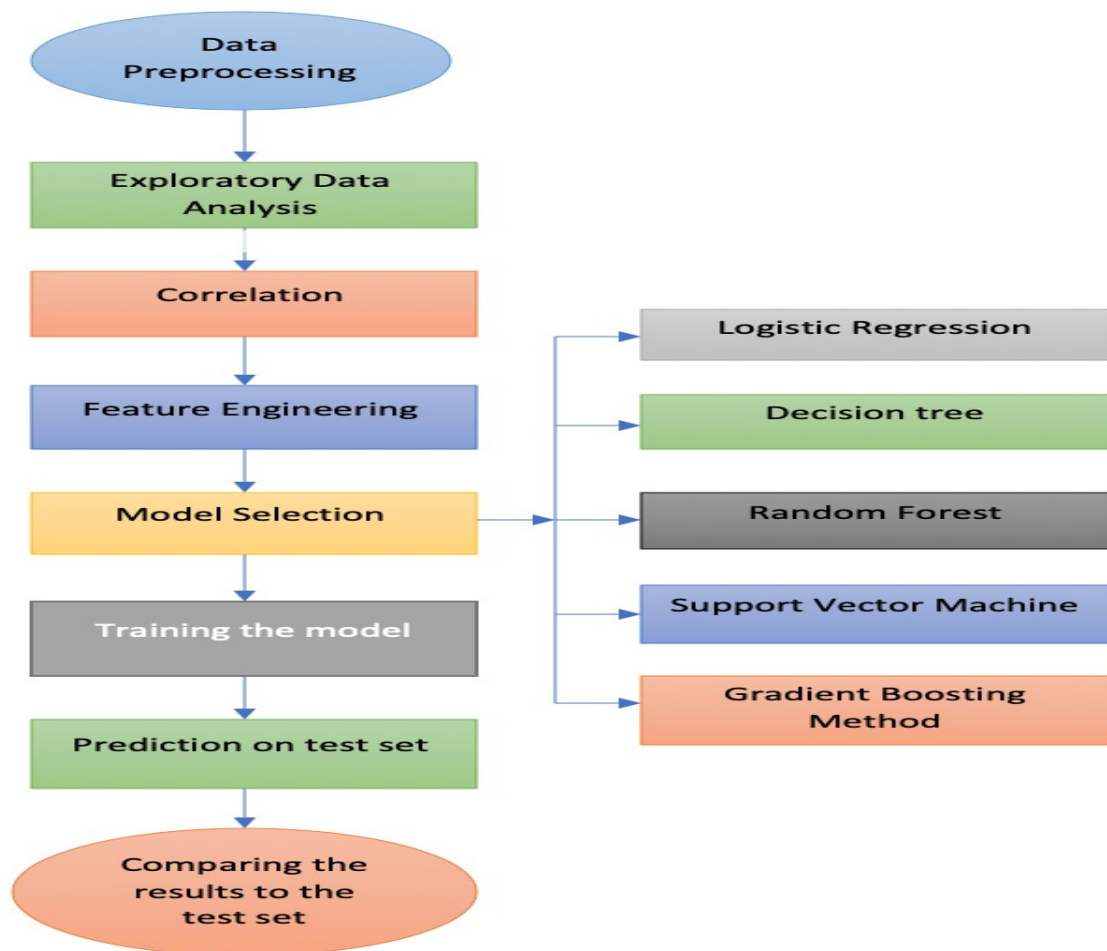
Table 1. Input variables

Output variable (desired target):

Output variable	Description
y	has the client subscribed a term deposit? (binary: 'yes','no')

Table 2. Output variables

As one may notice, 20 of the attributes are considered to be input/ explanatory variables. Some of these are categorical and will need to be formatted as a factor for model development and usage. The output/ response variable is the outcome. Based on the combinations of the explanatory variables, the response variable can be determined through machine learning approaches. For this response variable, the answer is binary and these binary values 1 and 0 represent yes and no, respectively. This being if the customer will subscribe to the term deposit, which is where the client will place his money into the bank for a set term/period to accumulate interest over that time frame.

Flow chart:*Figure 1. Flow chart for the analysis***Objective:**

From the dataset, the identified response variable can give limited machine learning approaches. The goal is to be able to predict if the customer will subscribe to a term deposit. This gives values of 1 or 0 being yes or no, respectively. Thus, machine learning models that use classification methods are used to predict these results.

Data Pre-processing

In the data pre-processing part, it is a technique where we convert the raw data into a clean data which is suitable for the data analysis. In short, we need to organize the data and the data must be clean after this step. Raw data may contain outliers, missing values, noise data or inconsistent data. It consists of three techniques i.e. data cleaning, data transformation and data reduction. In data cleaning, you can deal with the missing data with the mean/median and noisy data with binning, regression etc. methods. In the data transformation technique, we apply normalization, discretization etc. methods. Lastly in the data reduction, dimensionality reduction, attribute selection etc. methods are used. After applying these techniques, consistent / clean data is obtained.

In respect with our project, firstly checked the duplicate data in our dataset. They were 12 records removed from our data set using the function `distinct`. In the second step, we found the missing data in our dataset. The missing data was in housing (990), loan (990) and default (8597). In the default feature, the yes are 3 in number and no's are around 33000, therefore we impute / replace with its mode 'no'. In the housing and loan, we cannot impute / replace the values with median or mode since these variables are binary categorical variables. These 990 rows were removed from the dataset. (as mean cannot be used for binary categorical data) and mode occurrence is low.

In the final steps , the feature engineering, we have converted the Job, Marital ,Education, Default, Housing, loan, contact, month and day_of_week (categorical features) into numeric and then to factor variable and combined all the features into a single data frame `Bank_newdf`. The `sapply` function is used with `Bank_newdf` as an input and returns a vector/matrix as an output. Finally, the dataframe `Bank_newdf` is transformed using the default function `transform` which transforms the data in an easy and quick manner.

The transformation methods were not applied in these particular data variables as the data is being converted into inconsistent and resulting in outliers. The data-pre-processing is concluded as the clean data is obtained with no missing values, stability of data , the values are in the required range for performing the analysis.

```
> summary(Bank_newdf)
```

age		job		marital		education		default	
Min.	:17.00	admin.	:10419	divorced:	4611	university.degree	:12164	no	:32577
1st Qu.	:32.00	blue-collar:	9253	married	:24921	high.school	: 9512	unknown:	8596
Median	:38.00	technician	: 6739	single	:11564	basic.9y	: 6045	yes	: 3
Mean	:40.02	services	: 3967	unknown	: 80	professional.course	:5240		
3rd Qu.	:47.00	management	: 2924			basic.4y	: 4176		
Max.	:98.00	retired	: 1718			basic.6y	: 2291		
		(Other)	: 6156			(Other)	: 1748		

housing		loan		contact		month		day_of_week		duration	
no	:18615	no	:33938	cellular	:26135	may	:13767	fri	:7826	Min.	: 0.0
unknown:	990	unknown:	990	telephone:	15041	jul	: 7169	mon	:8512	1st Qu.	:102.0
yes	:21571	yes	: 6248			aug	: 6176	thu	:8618	Median	: 180.0
						jun	: 5318	tue	:8086	Mean	: 258.3
						nov	: 4100	wed	:8134	3rd Qu.	: 319.0
						apr	: 2631			Max.	:4918.0
						(Other):	2015				

campaign		pdays		previous		poutcome		emp.var.rate	
Min.	: 1.000	Min.	: 0.0	Min.	:0.000	failure	: 4252	Min.	: -3.40000
1st Qu.	: 1.000	1st Qu.	:999.0	1st Qu.	:0.000	nonexistent:	35551	1st Qu.	: -1.80000
Median	: 2.000	Median	:999.0	Median	:0.000	success	: 1373	Median	: 1.10000
Mean	: 2.568	Mean	:962.5	Mean	:0.173			Mean	: 0.08192
3rd Qu.	: 3.000	3rd Qu.	:999.0	3rd Qu.	:0.000			3rd Qu.	: 1.40000
Max.	:56.000	Max.	:999.0	Max.	:7.000			Max.	: 1.40000

cons.price.idx		cons.conf.idx		euribor3m		nr.employed		y	
Min.	:92.20	Min.	: -50.8	Min.	:0.634	Min.	:4964	no	:36537
1st Qu.	:93.08	1st Qu.	: -42.7	1st Qu.	:1.344	1st Qu.	:5099	yes:	4639
Median	:93.75	Median	: -41.8	Median	:4.857	Median	:5191		
Mean	:93.58	Mean	: -40.5	Mean	:3.621	Mean	:5167		
3rd Qu.	:93.99	3rd Qu.	: -36.4	3rd Qu.	:4.961	3rd Qu.	:5228		
Max.	:94.77	Max.	: -26.9	Max.	:5.045	Max.	:5228		

Figure 2: Summary of the new data frame after performing preprocessing

Exploratory Data Analysis:

The exploratory data analysis (EDA) is an important technique used in the data science field which helps us to get an initial insight of the dataset, quality of the data, discover structures in the data, detecting the outliers, finding out the important variables in the dataset. The EDA in short is that we question the data and find the answers by applying the techniques. The EDA technique doesn't have a certain procedure to follow as it depends on the dataset chosen. The EDA helps us to identify initial assumptions and carry out the further analysis after performing the technique. The EDA techniques are combination of graphical and quantitative techniques. The graphical techniques include histograms, box plots, probability plots etc.

Age distribution:

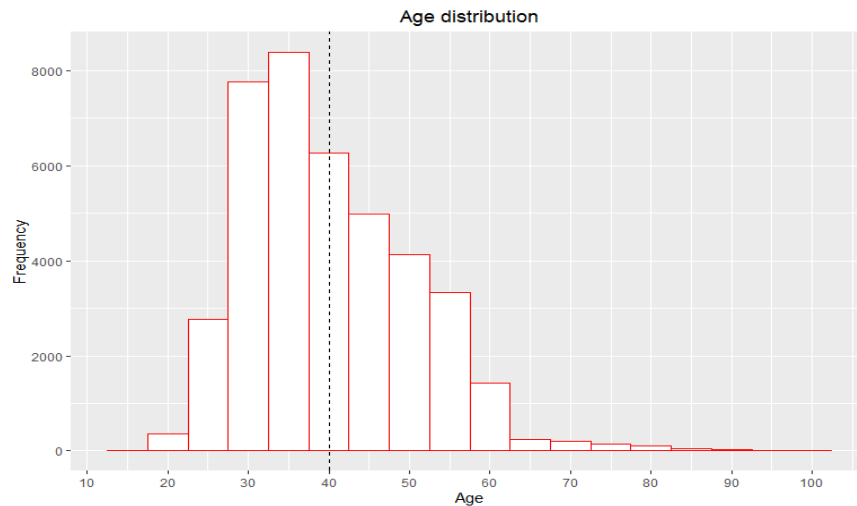


Figure 3. Stacked plot with job against the count having subscription proportion filled

In the age distribution, it is little right skewed with majority of the client people's age are in the age 30 – 50. It is clear that the target age group of the bank is 30-50. The median age is represented by the dotted line which is 40. The distribution covers all the age groups which is beneficial for the training model. We can clearly observe that clients are targeted from the legal age only above 20 years.

Term Deposit Subscription:

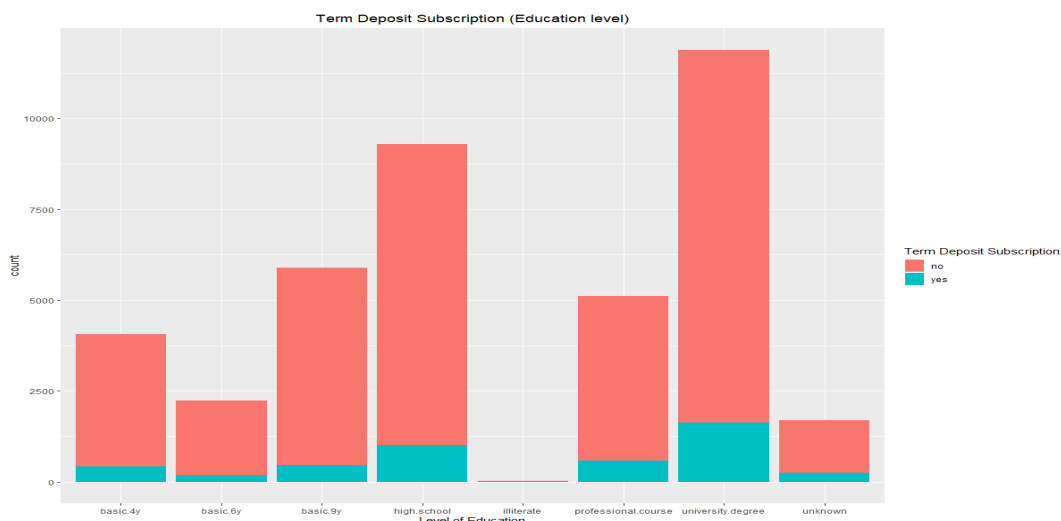


Figure 4. Stacked plot with job against the count having subscription proportion filled

In the level of education, the term deposit subscription for the university degree is high compared to the other education levels and at the same time the rejection rate high which is unexpected scenario. The students who are pursuing education in the 4th, 6th and 9th standards the rejection rate is high, and the acceptance rate is less among the three. In the high school, the rejection is higher and the acceptance rate from the high school students is good. Lastly, the professional course shows a good ratio of acceptance to the reject of the term deposit.

Job vs Subscription:

The stacked plot in the figure 5 depicts job and subscription relationship. It looks like the admin job is targeted the most and it has the highest acceptance and rejection rate for the subscription. Technicians have the second highest approvals followed by blue collar job people, that has the highest rejection rate. We can also add that people under retired job status can likely create positive responses comparatively than others because for limited target, it generated more positive responses than others.

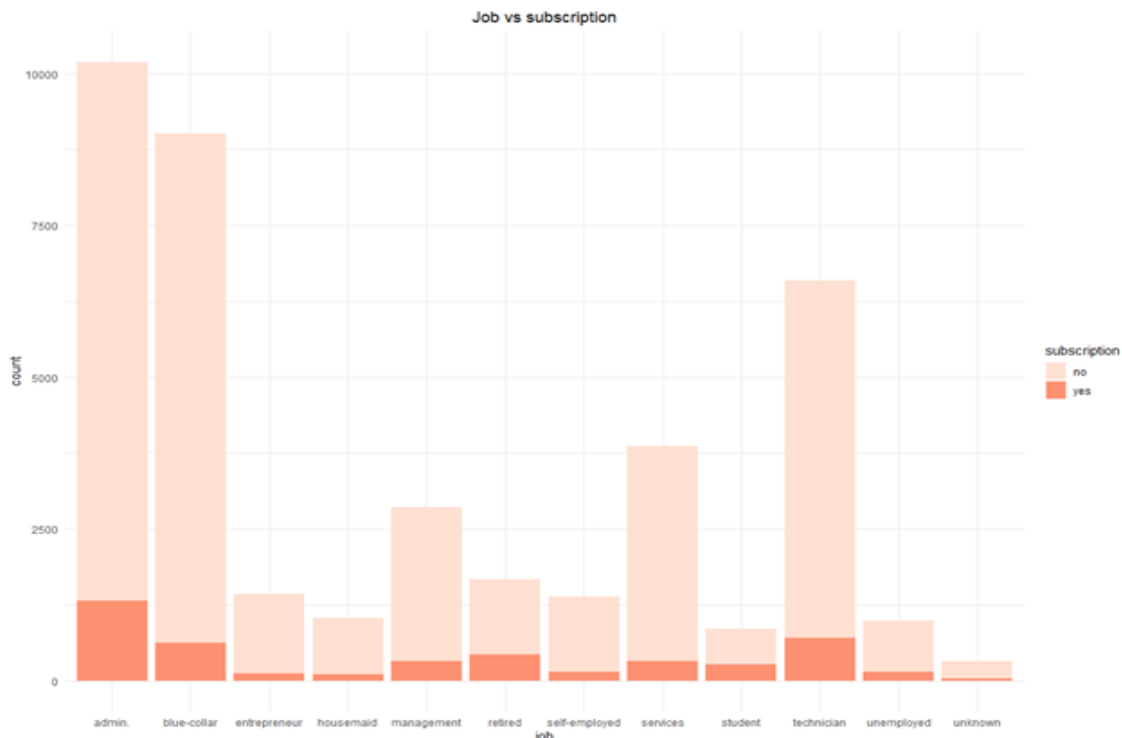


Figure 5. Stacked plot with job against the count having subscription proportion filled

Subscription based on number of contact during period:

Figure 6 visualization is about subscriptions based on number of people contacted during the campaign. Positive responses are recorded the most when 1 or 2 people contacted the patron. Further increase in association with the client can reduce the subscription rate drastically. Subscription rate almost becomes zero when more than 7 people contacted the consumer for the same during the campaign.

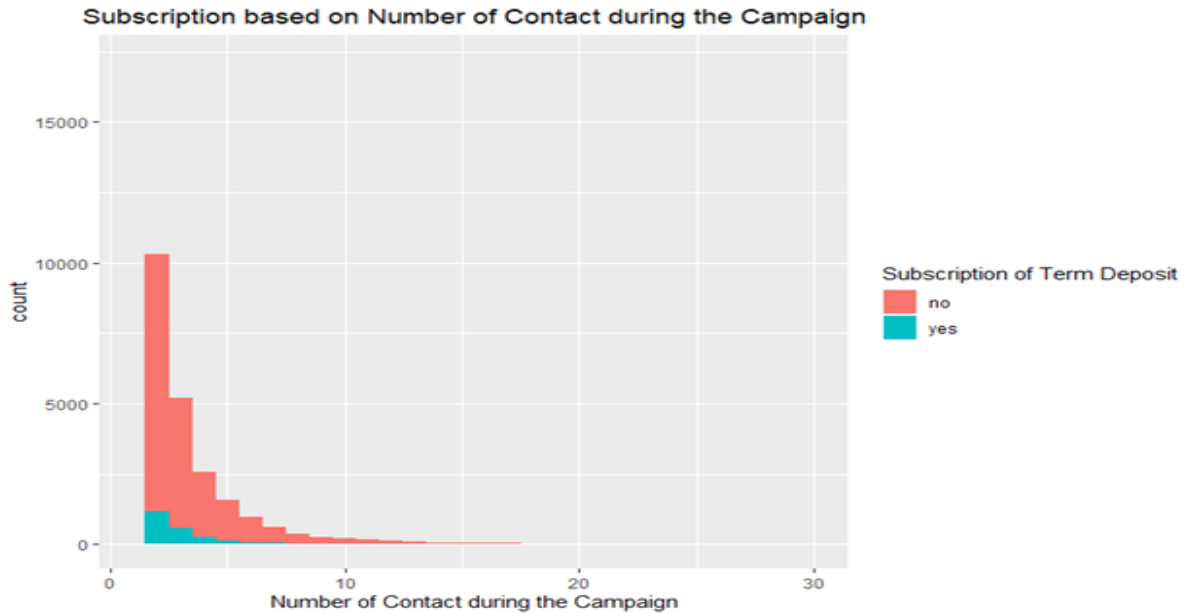


Figure 6. Stacked plot with No. of people contacted during campaign against the count having subscription proportion filled

Age vs Duration plot:

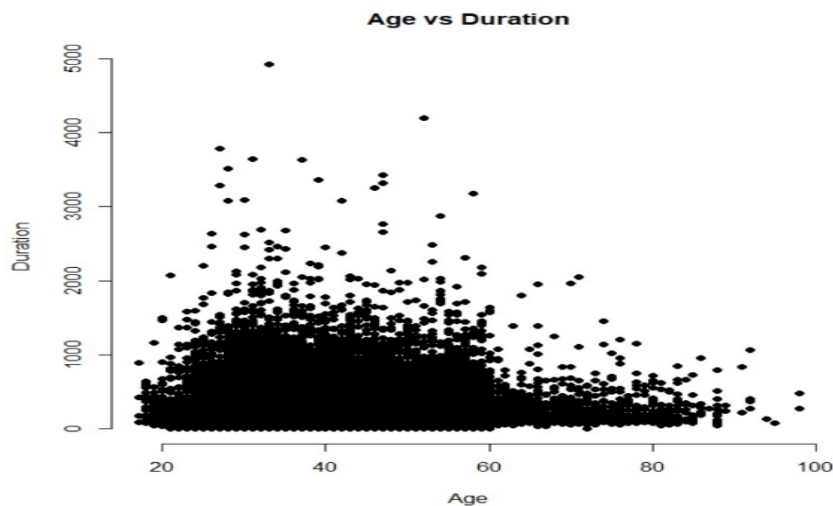


Figure 7. Age vs Duration

The scatter plot clearly describes that most of the call duration who were last contracted are more in number in between ages 25-50. The frequency observed is also high in the same age range as mentioned above. This makes it clearer that the outliers that are found using this are very useful in getting insights of the distribution.

Correlation:

From the correlation plot in figure 8, we can say that there are 3 features, emp.var.rate, euribor3m, nr.employed are strongly correlated. 2 out of 3 variables are eliminated during analysis because presence of them may affect the model. Most of the variables are either weakly correlated or negatively correlated.

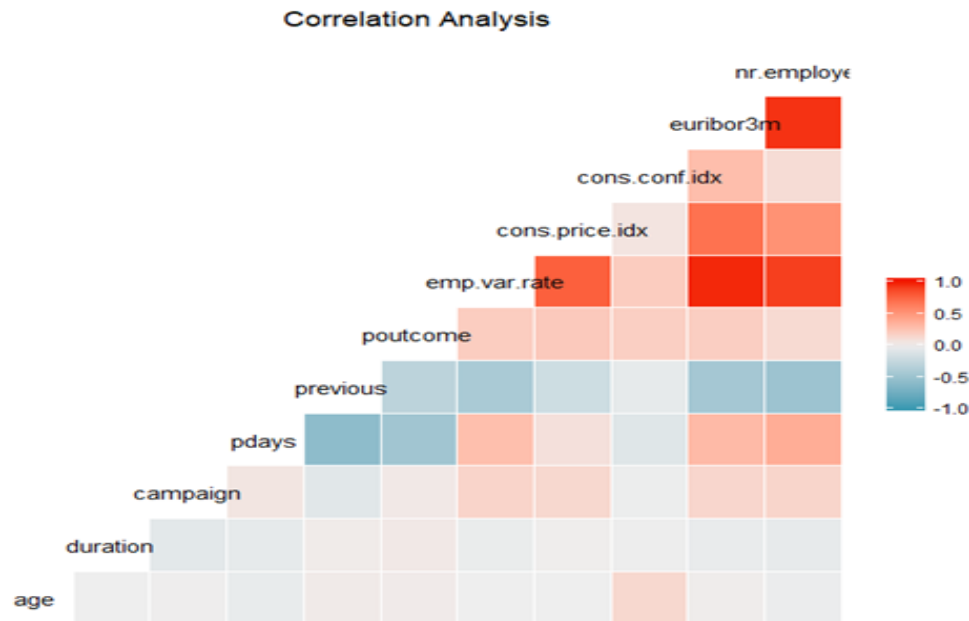


Figure 8. The correlation plot

Predictive Models

Data Split train/test:

Following the data preprocessing, one needs to split the dataset into train and validation sets. The split that will be used for all machine learning classification approaches is 60:40 percent of the preprocessed dataset. The 60%, which is the majority, will belong to the train set that will help to develop the model. The 40% will be the validation that will provide the results/ accuracy of the model that was developed previously.

Logistic Regression:

Since the data preprocessing involved the categorical variables to be converted to factors, the logistic regression approach can be applied to the dataset. The input/explanatory variables that are used for the logistic regression are as follows: age, job, marital, education, housing, month, day_of_week, duration, campaign, pdays, previous, poutcome, cons.price.idx, cons.conf.idx, and nr.employed. And, of course, the output/response variable being y, which is the binary term deposit subscription being yes or no.

The logistic regression model can determine the coefficients determined and the probabilities of the records. Next, the results will need to be analyzed to determine the accuracy of the model.

After applying the logistic regression model, a confusion matrix was developed. The confusion matrix will approximately show the results of the logistic regression model that were predicted against the actual results (if one wants the same results when re-running the model, then the seed needs to be set prior to the split).

Confusion matrix and Results:

	FALSE	TRUE
no	13921	361
yes	1011	782

Fig 9. Confusion matrix for Logistic regression

Logistic Regression	Result
Accuracy	91.17%
Sensitivity	96.15%
Specificity	52.40%

Table 3. Logistic Regression results

Receiving Operator Curve (ROC) is a plot that displays two measures over all possible thresholds: sensitivity or True Positive Rate and 1-specificity or False Positive Rate. Based on the dataset with the applied logistic regression model to determine predictions, a ROC curve is able to be developed.

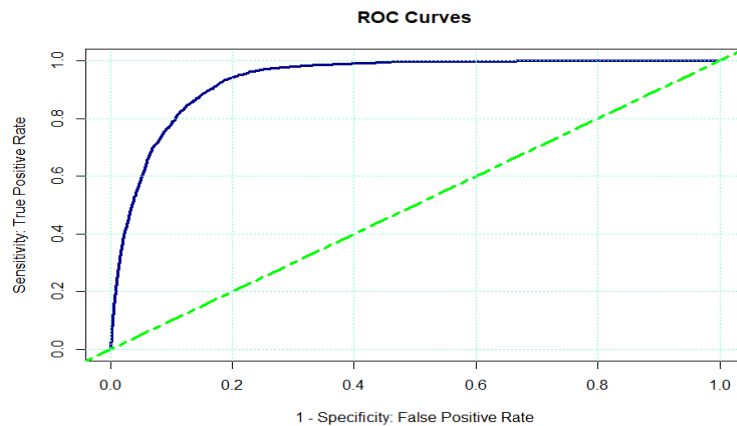


Figure 10. This is a plot that displays the ROC curve for the logistic regression model

For any ROC, the better performance models have the curve closer to top left corner of this plot. This would give the maximum possible area under curve value to be 1. For this specific model on this dataset, the area under the curve gave a value of 0.93. This value is a relatively high value proving that the performance of this logistic regression model is good.

Another important plot to use is a lift chart. A lift chart is used to measure how much better one can expect to do with the classification model such as logistic regression compared to without a model.

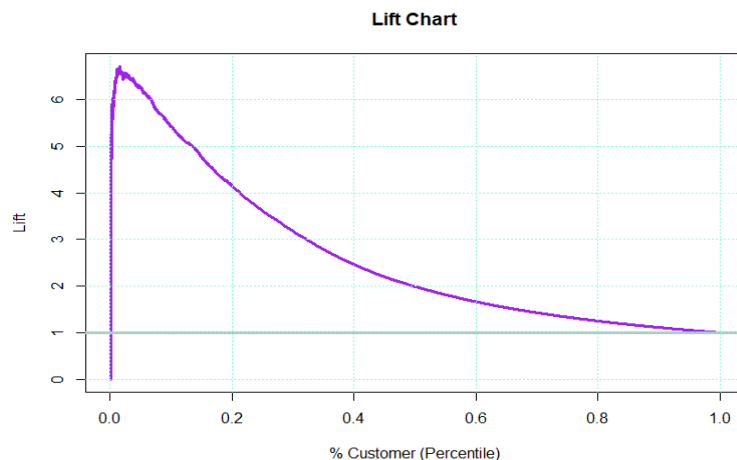


Figure 11. This is the developed lift chart based on the logistic regression model (purple) and without the model (grey) (random)

Lift is the proportion of true positives divided by the proportion of positive hits (target response divided by the average response). For example, set the target at 40%. The lift for the 40% target is approximately 2.5. This means that the first 40% of the records that contain the most likely responders have 2.5 times as many responders as a similarly sized random sample of records.

Decision tree:

Decision tree build model in the form of tree structure. A decision tree is a set of rules represented in tree structure. The aim of decision tree is to create a tree that allows you to define various target groups based on values from a collection of input variables. Decision tree breaks down the dataset into smaller

subsets and grows them as decision tree. Decision tree is based on a set of if-then rules that form a set of partitions. The final result is a tree with decision nodes and leaf nodes.

The decision node has two branches and the leaf node represents classification decision. The topmost decision node is called a root node. Root node corresponds to the best predictor. The root node can be calculated in two methods. One among them is Entropy and the other is Gini index. Decision trees can deal with both categorical and numerical data.

The below is the decision tree that was developed for our model:

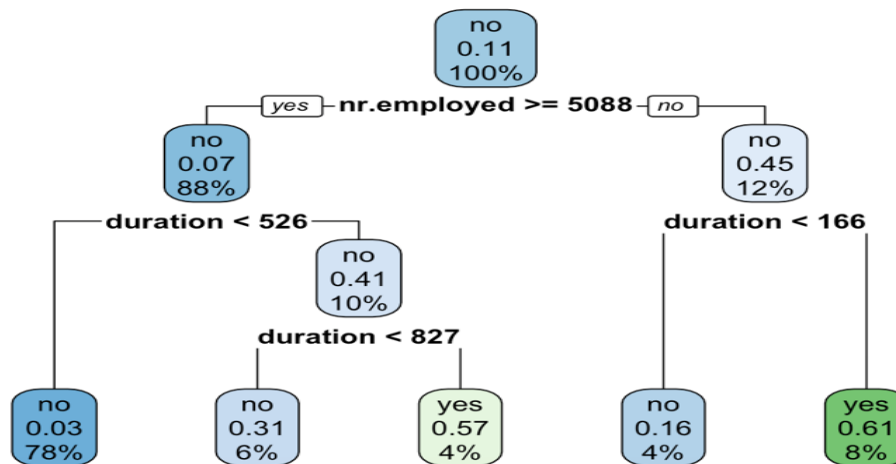


Figure 12. This is the Decision tree obtained

For our model the root node is nr. employed. The decision that was classified at the root node is if nr. employed >= 5088, based on this decision the tree was built.

Confusion matrix and Results:

```
> table(tree.pred, bank_test$y)
```

```

tree.pred   no   yes
no  13547  755
yes   735 1038

```

Figure 13. Confusion matrix for Decision tree

Decision tree	Result
Accuracy	91.17%
Sensitivity	96.15%
Specificity	52.40%

Table 4. Decision tree results

Random Forest:

Random forest is not good with categorical variables, as it favors with the category variables having many levels. So, here starting we have transformed our whole categorical data into numerical data. Then the class of numbers are further changed to factor level representation.

Parameters used	Purpose
Mtry = 2	Total number of base Trees I have considered as 2 as it gave me the best results.
Do.trace=10	For every 10 tress I could trace the procedure that is running in the background.
ntree = 1000	Total number of trees for each base tree is 1000.

We trained our model by using random forest function and having default values of mtry and ntrees. Data passed to our model will be our cleaned dataset and selected features after the correlation. After first

training of our model, the fitted model has OOB estimate error of 9.05% Then we need to tune the model to decrease the error rate. Before that we have selected the features by using varimplot function and plotting the feature importance.

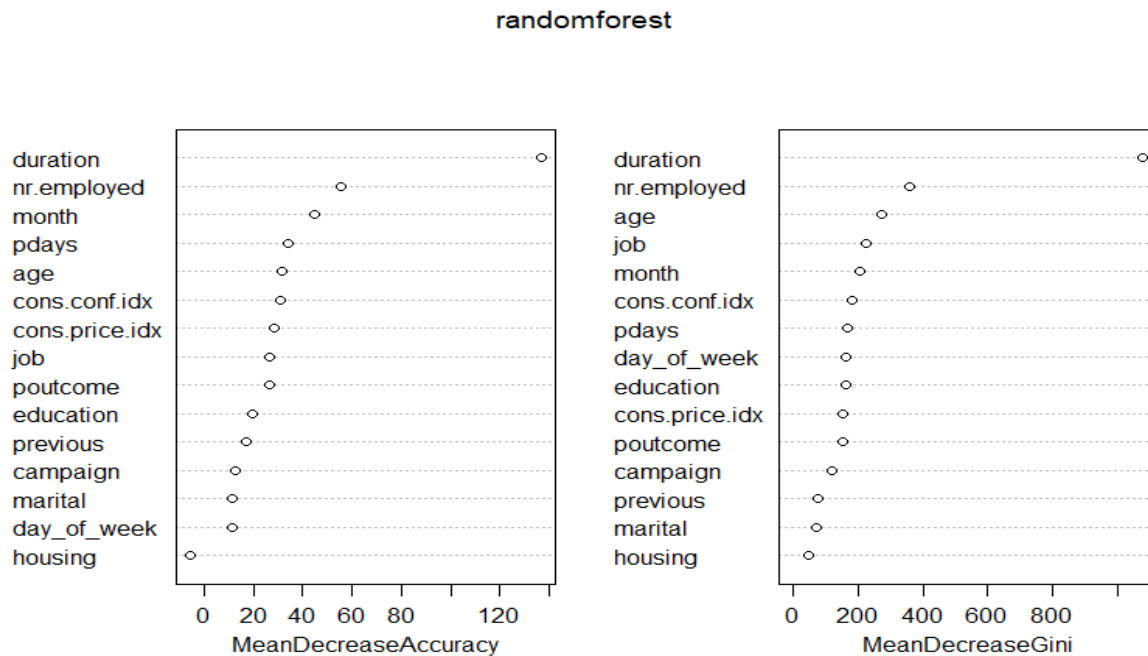


Figure 14. Variable Importance Plot

From the figure 14 we can clearly notice the feature importance of each variable to the contribution of the predictor variable (subscription yes/no).

Duration being the highest of all with a great margin, when compared to other variables, nr.employed, month, pdays & age make to the top 5 variables in the feature importance table, where housing shows the least effect when compared to all other variables.

Confusion matrix and Results:

```
> pred_cm <- table(exp.pred,actual = bank_test$y)
> pred_cm
```

```
      actual
exp.pred no  yes
no  13980 1118
yes   302  675
```

Figure 15. Confusion matrix for Decision tree

Random Forest	Results
Accuracy	91.11%
Sensitivity	97.84%
Specificity	37.64%

Table 5. Random forest results

SVM (Support Vector Machine):

A support vector machine takes these data points and outputs the hyperplane (which in two dimensions it's simply a line) that best separates the tags. This line is the decision boundary. The SVM starts with low degree and then it automatically increases the degree to user specific or automatically as per the default in the algorithm, due to this the maximum margin classifier is lower classifier than SVM (Support Vector Machine).

Variable Importance of SVM:

```

$sresponses[[1]]$n
[1] "age"

$sresponses[[2]]$n
[1] "job"

$sresponses[[3]]$n
[1] "marital"

$sresponses[[4]]$n
[1] "education"

$sresponses[[5]]$n
[1] "default"

```

Figure 16. Output of variable importance

Parameters used	Purpose
Kernel = radial	We have used kernel as radial in order to have a non-linear boundary classification to improve the accuracy of the model.
Cost = 5	After trying many cost numbers at last we got the best result for cost = 5
Gamma = 0.0625	When cost is equal to 0.0625, model gave the best output
Cross validation = 10	We have used k=10 in the k fold cross validation in order to have stability in the training data instead of using the entire training data to build the training model.

Confusion matrix and Results:

```

> table(predict=svm1_sol, truth=bank_test$y)
      truth
predict no  yes
no      13876 998
yes      406  795

```

Figure 17. Confusion matrix for SVM

SVM (Support Vector Machine)	Results
Accuracy	91.26%
Sensitivity	97.30%
Specificity	44.33%

*Table 6. SVM results***Gradient Boosting Method (GBM):**

We wanted to explore more options of exercising the model to improve the accuracy and we chose GBM algorithm. Basic understanding of GBM is it combines and converts all the weak learners into strong learners. In boosting, each new tree is a fit on a modified version of the original data set. The AdaBoost Algorithm begins by training a decision tree in which each observation is assigned an equal weight. Each tree is grown using information from previously grown trees. In each iteration, the algorithm applies greater weight to the records that are misclassified.

The model is trained using train sample with selected variables by setting the cross-validation limit as 10. The trained model has acquired accuracy around 91%. And the test sample has generated accuracy

of 91.57% on the GBM model that is calculated from the confusion matrix in the figure 19.

Parameters used	Purpose
trControl=trainControl(method="cv", number=10)	This function generates parameters that further control how models are created, for example, here, cross validation method with K = 10 is used.
verbose=FALSE	A verbose connection provides much more information about the flow of information between the client and server. Here it is given false, that means there's no connection required.

Confusion matrix and Results:

```
> table(Predic, bank_test$y)
```

```

Predic   no   yes
no  13800  882
yes   482  911

```

Gradient Boosting Method (GBM)	Results
Accuracy	91.57%
Sensitivity	96.62%
Specificity	50.86%

Figure 18. Confusion matrix for GBM

Table 7. GBM results

The influence plot has categorized Pdays (Number of days last contacted from a previous campaign) as the most important variable in classifying the subscription followed by Month7 (July), Job9 (Student).

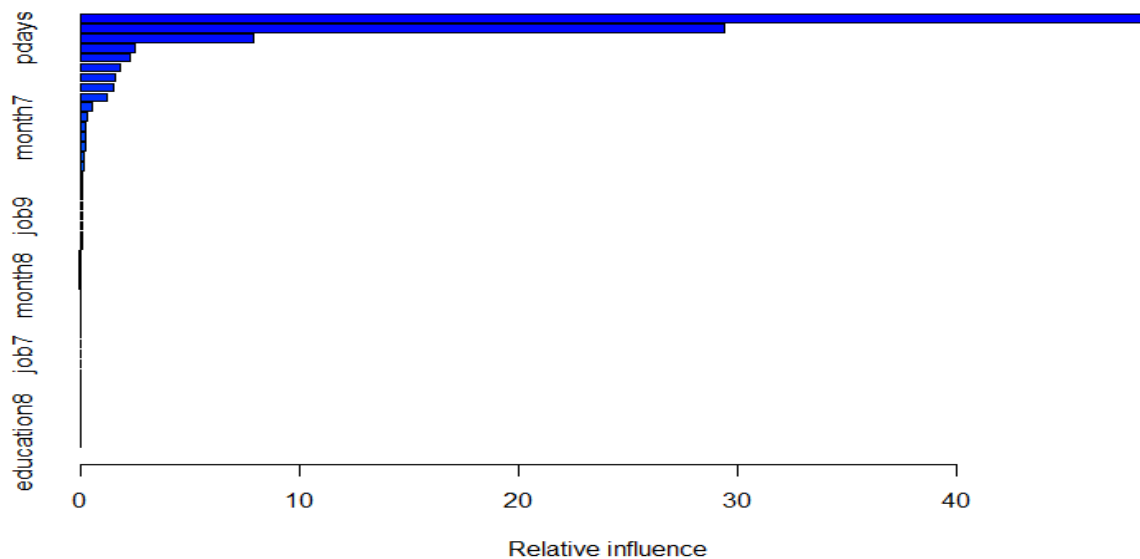


Figure 19. Influence plot for GBM model

Conclusion

GBM has the highest accuracy relatively compared to other models because it learns from its past misclassifications and tries to correct it. Based on these models we could predict if a customer is willing to have term deposit subscription or not. Strong preprocessing made the data stable and have improved the performance of all models with above 90% accuracy. We can predict whether a customer is willing to accept subscription accurately 90% of the times overall in all our models.

Future work:

In our project, we have applied machine learning techniques such as Logistic regression, Random forest, Support vector machines and Decision tree. In the further development of the project, various clustering techniques can be implemented. Apart from this, the study of the bank sector can enable us to provide more relations among the variables in the dataset.

Lessons Learnt:

- Data pre-processing is one of the major steps to build a good working model, as without data cleaning results appear to be very harsh at the accuracy. It took almost 60% of the time in the project to complete the data pre-processing procedure.
- We cannot always rely on accuracy all the time because if the classification of the data is not balanced then we get good accuracy but that is not what we are looking at. Hence, we also have taken into consideration about sensitivity and specificity of the model.
- Selecting the model that we are going to use for the analysis by examining the data set is one of the important factors, as we found that for our data set, Linear regression would not perform well as our output variable is a binary data type (Yes/No).
- Data splitting is one more important feature. We have used both single data split and used K-fold validation in order to get better stability of the output.

Step by Step instructions on how to run the code

Step1: Open the: "OR568proj.R" file

Step 2: Replace the file directory of the dataset according to the place where the dataset is downloaded.

The datasets are included in the zip file or can be found in the link provided in the references.

Step 3: All libraries and dependencies are listed in the code at the beginning. The code can run until the line 199 in the R file.

Step 4: Each model can be found in the respective section labelled with the comments.

Step 5: From line 39, the pre-processing begins. The pre-processing is done in a continuous manner.

Step 6: Each model has a separate confusion matrix generated. Graphs have been generated according to each model.

Step 7: Outputs have been described in the project report in the pdf.

References:

UCI Machine Learning Repository: Bank Marketing Data Set,
<https://archive.ics.uci.edu/ml/datasets/Bank+Marketing>.