IMARTICUS
LEARNING

CAPSTONE PROJECT - PGA

Title: Term Deposit Subscription Prediction

ABSTRACT

Goal of this project is to successfully predict if a given customer will subscribe to a term deposit in a telemarketing campaign held by the bank.

AUTHORS

Nayan Waghaye
Kalyani Bangale
Navin Pandiyan
Neha Bhirud

Problem Statement

The data is about telemarketing campaigns of a European banking institution. The European bank wants to predict which clients will secure a term deposit based on a set of information on client and purchase of term deposit. The marketing is usually based on phone calls. Often, a client needs to be persuaded multiple times in order to assess if the product (bank term deposit) would be or not subscribed. Predictive modelling approach will help the bank to manage their telemarketing campaign efficiently.

The Data

Sourcing

The data was sourced from UC Irvine Machine Learning Repository. The link to the dataset is given below.

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

Dataset Reference: [Moro et al., 2014] S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014.

Variables Description

Information was collected on 41,188 clients against 20 variables for the prediction term deposit (yes/no).

#Numerical Data

1. **Age:** Age of the person
2. **Duration:** Last call duration in seconds.
3. **Campaign:** Number of contacts performed during this campaign and for this client.
4. **Pdays:** Number of days that passed by after the client was last contacted from a previous campaign. (999 means that the client was not previously contacted)
5. **Previous:** Number of contacts performed before this campaign and for this client.
6. **Emp.var.rate:** Employment Variation Rate - Quarterly indicator.
7. **Cons.price.idx:** Consumer Price Index - Monthly indicator.
8. **Cons.conf.idx:** Consumer Confidence Index – Monthly indicator.
9. **Euribor3m:** Euribor 3 Month Rate – Daily Indicator.
10. **Nr.employed:** Number of Employees – Quarterly Indicator.

#Categorical Data

1. **Job:** Type of Job. (admin., blue-collar, entrepreneur, housemaid, management, retired, self-employed, services, student, technician, unemployed)

2. **Marital:** Marital Status. (married, single, divorced) note: divorced means divorced or widowed.

3. **Education:** Education Level. (basic.4y, basic.6y, basic.9y, high.school, illiterate, professional.course, university.degree)

4. **Default:** Tells if the person has credit in default. (no, yes)

5. **Housing:** Tells if the person has a housing loan. (no, yes)

6. **Loan:** Tells if the person has a personal loan. (no, yes)

7. **Contact:** Customer communication type. (cellular, telephone)

8. **Month:** Last contact month. (jan, feb, mar, …, nov, dec)

9. **Day_of_week:** Last contact day of the week. (mon, tue, wed, …, sat, sun)

10. **Poutcome:** Outcome of the previous marketing campaign. (failure, non-existent, success)

11. **y:** Tells us if the person subscribes to term deposit. (Dependent variable). (yes, no)

Summary of the Data

```
> summary(bank)
      age                 job            marital                    education      default    
 Min.   :17.00   admin.      :10422          :    0   university.degree  :12168     :    0  
 1st Qu.:32.00   blue-collar: 9254   divorced: 4612   high.school        : 9515   no  :32588  
 Median :38.00   technician : 6743   married :24928   basic.9y           : 6045   yes :    3  
 Mean   :40.02   services   : 3969   single  :11568   professional.course: 5243   NA's: 8597  
 3rd Qu.:47.00   management : 2924   NA's    :   80   basic.4y           : 4176               
 Max.   :98.00   (Other)    : 7546                    (Other)            : 2310               
                 NA's       :  330                    NA's               : 1731               
 housing        loan           contact          month        day_of_week    duration     
     :    0       :    0   cellular :26144   may    :13769   fri:7827    Min.   :   0.0  
 no  :18622   no  :33950   telephone:15044   jul    : 7174   mon:8514    1st Qu.: 102.0  
 yes :21576   yes : 6248                     aug    : 6178   thu:8623    Median : 180.0  
 NA's:  990   NA's:  990                     jun    : 5318   tue:8090    Mean   : 258.3  
                                             nov    : 4101   wed:8134    3rd Qu.: 319.0  
                                             apr    : 2632               Max.   :4918.0  
                                             (Other): 2016                               
    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:35563   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   : 0.08189  
 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 :36548  
 1st Qu.:93.08   1st Qu.:-42.7   1st Qu.:1.344   1st Qu.:5099   yes: 4640  
 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              
```

- The data consists of 41,188 clients' information with 36,548 subscribing to term deposit and 4640 not subscribing.

- The age of clients varies from the youngest being only 17 years old to the oldest being 98 years old.

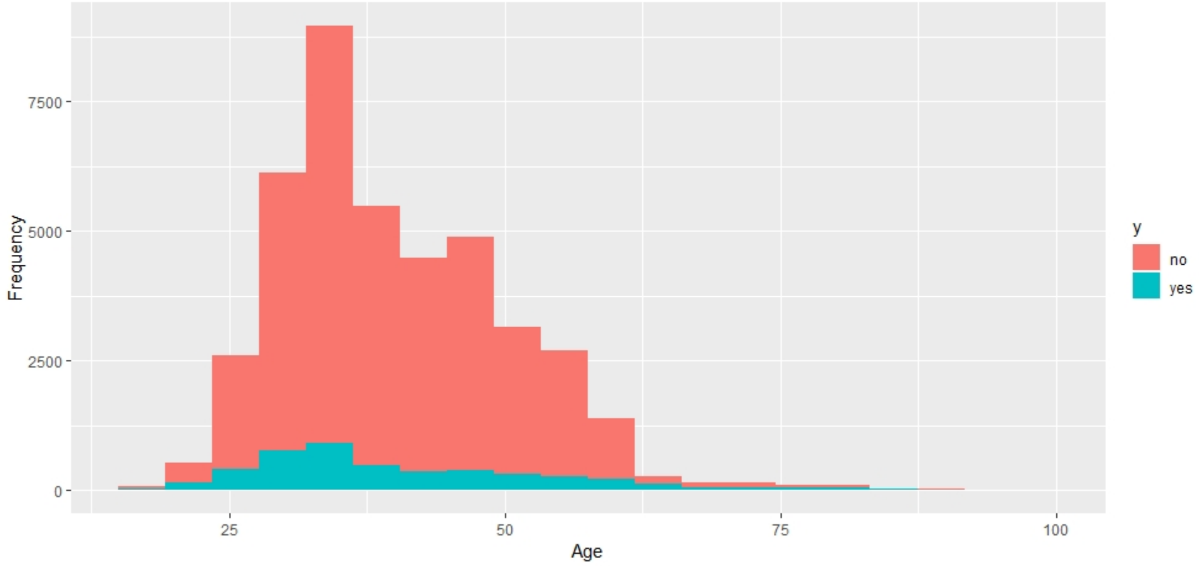
- Some of the clients have failed to provide certain information. These can be seen from the amount of missing values (NA's) in job, marital, education, default, housing, loan columns. We will be seeing how to deal with these values shortly.

- The default column is predominantly 'no'. Only 3 clients have credit in default and around 20% of the clients have failed to provide the necessary information.

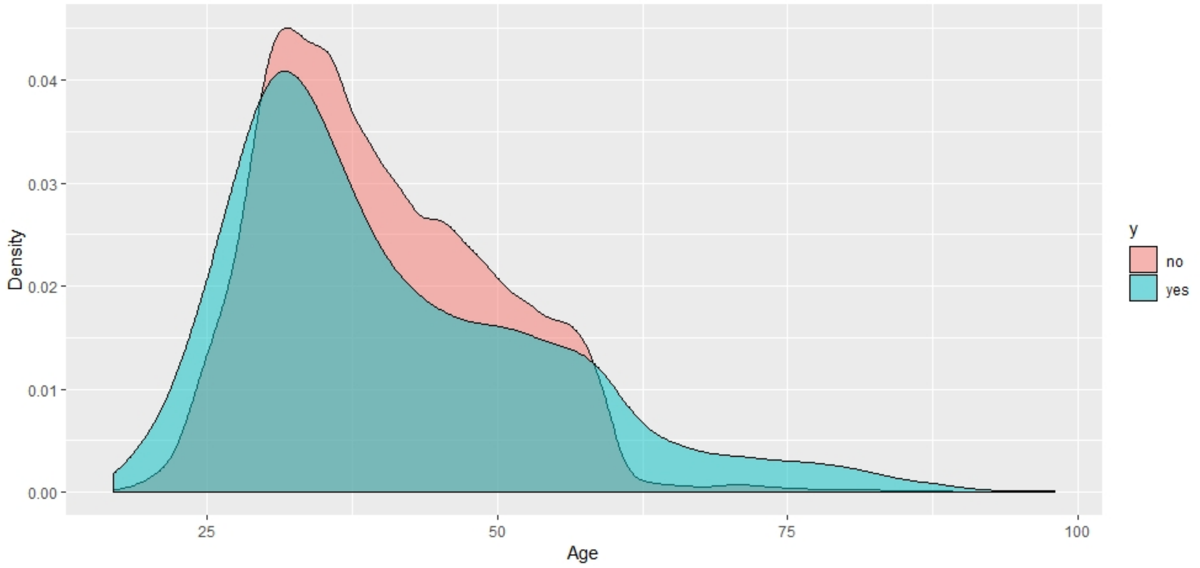
- Most of the clients don't have any previously existing personal loans.

Data Visualization

Histograms

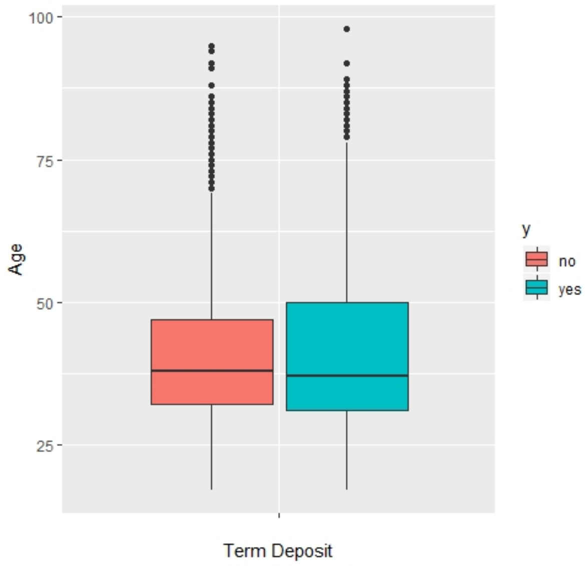


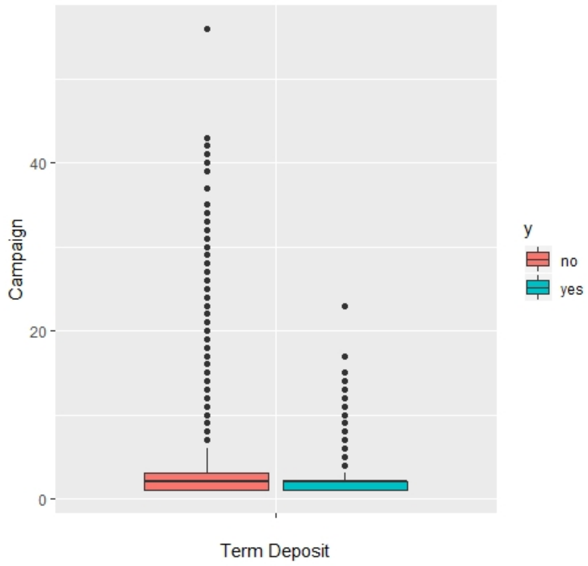
- The above histogram shows us that **Age** obeys a fairly pleasing **Normal Distribution.**
- It is also seen that majority of the people haven't subscribed to Term Deposit.



- The above density plot also tells us that both the classes follow a similar distribution.

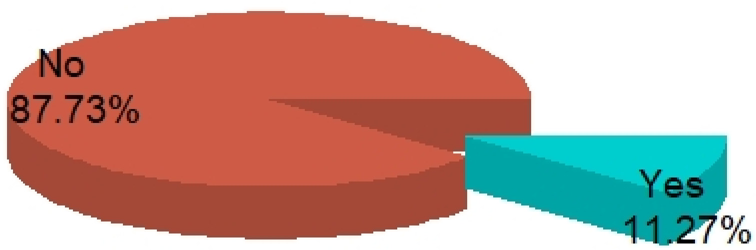
Box-whiskers Plots





- We can see that **Age** and **Campaign** have a significant number of **outliers**.

Pie Chart

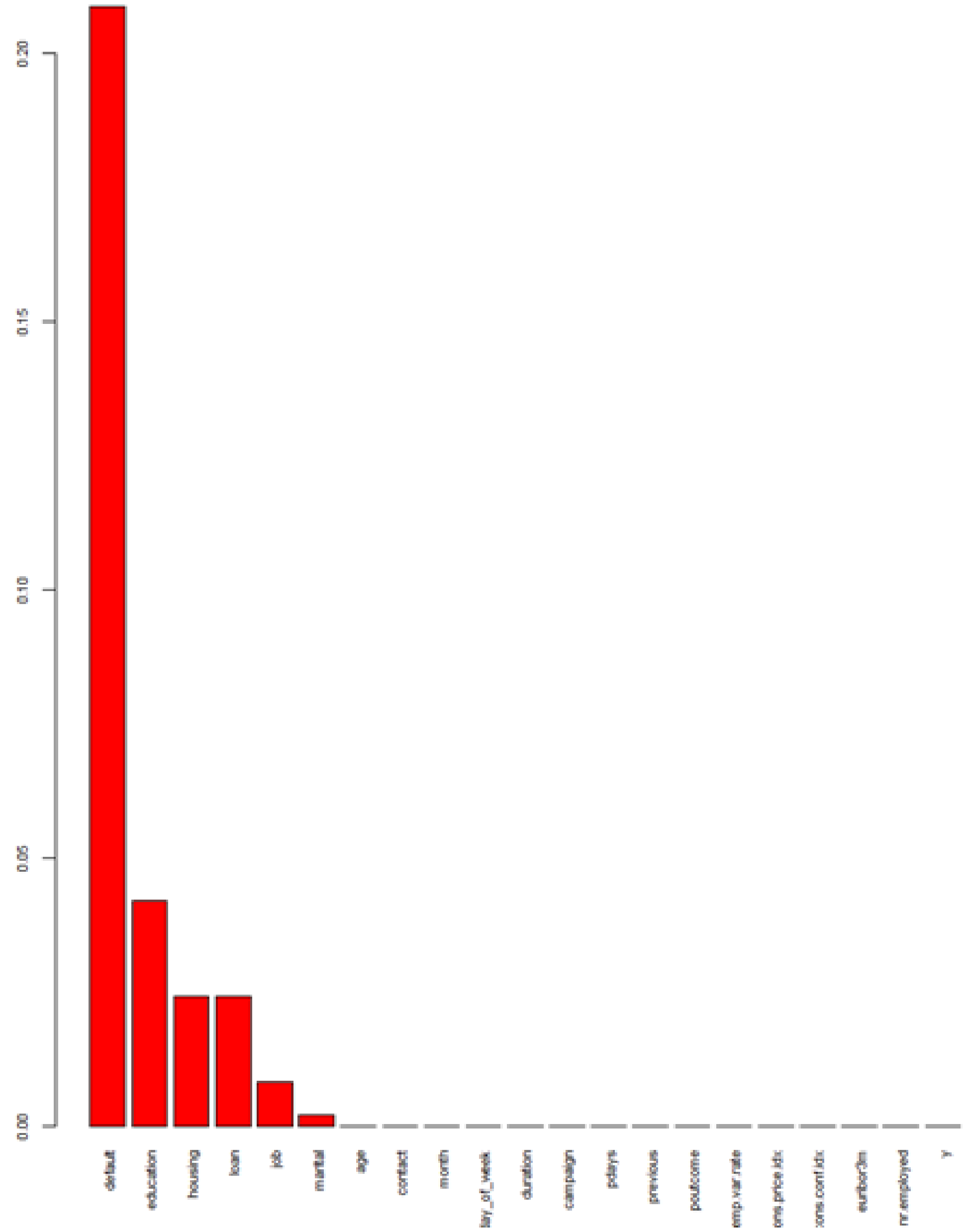


- It can be inferred from the pie chart that only 11.27% of the clients ended up subscribing to the Term deposit while the remaining 87.73% of them didn't.

- This poses as a huge **Class imbalance** issue which should be resolved before building any model.

Data Transformation

Missing values



- Education, Housing, Loan, Job, Marital constitute together for 9% for the missing values, whereas Default itself accounts for 20%.

- The summary of Default tells us that there are only 3 ‘Yes’ and 32588 ‘No’.

Solution

- For our convenience, we assume the missing data in **Default** column to be **'No'**.

  bank[is.na(bank$default),] ← "no"

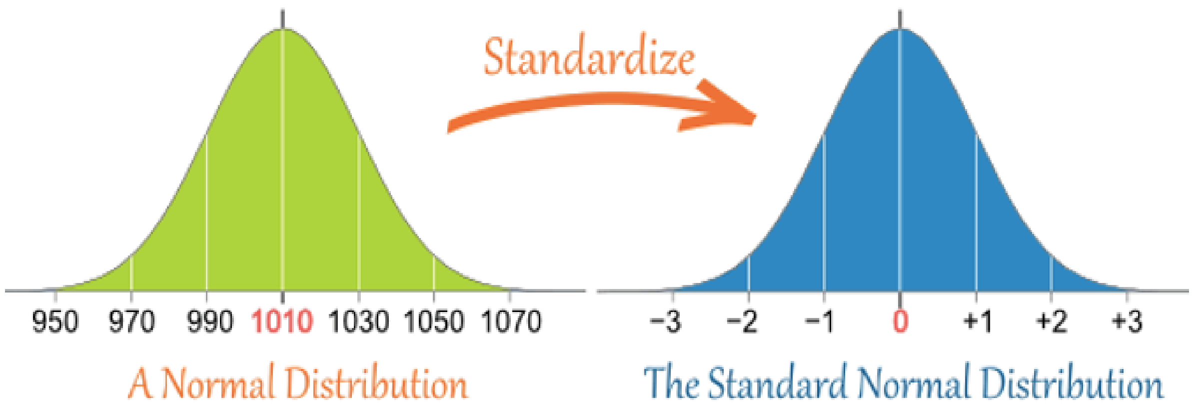
- We use **kNN imputation** to impute values for Education, Housing, Loan, Job, Marital columns.

  This can be achieved by using **kNN** function from **VIM package.**

  bank ← kNN(bank, variable = c("job", "marital", "education", "housing", "loan"), k = 3)

Data Standardization

Data Standardization is a process in which data attributes within a data model are organized to increase the cohesion of entity types. In other words, the goal of data standardization is to reduce and even eliminate data redundancy, an important consideration for application developers because it is incredibly difficult to store objects in a database that maintains the same information in several places.



This is example of how Data Standardization works.

Class Imbalance

Class imbalance is a supervised learning problem where one class outnumbers other class by a large proportion. From the pie chart, we found that our dataset exhibits this problem. Building a model on imbalanced datasets gives a **reduced accuracy**.

Below are the reasons which leads to reduction in accuracy of ML algorithms on imbalanced data sets:

1. ML algorithms struggle with accuracy because of the unequal distribution in dependent variable.
2. This causes the performance of existing classifiers to get biased towards majority class.
3. The algorithms are accuracy driven i.e. they aim to minimize the overall error to which the minority class contributes very little.
4. ML algorithms assume that the data set has balanced class distributions.
5. They also assume that errors obtained from different classes have same cost.

Solution

There are four methods to overcome class imbalance.

1. Undersampling
2. Oversampling
3. Synthetic Data Generation (SMOTE)
4. Cost Sensitive Learning (CSL)

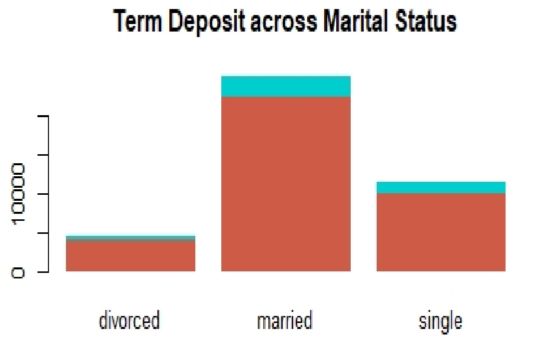
In our case study, we have used **SMOTE**. It is a powerful and widely used method. SMOTE algorithm creates artificial data based on feature space (rather than data space) similarities from minority samples. We can also say, it generates a random set of minority class observations to shift the classifier learning bias towards minority class.

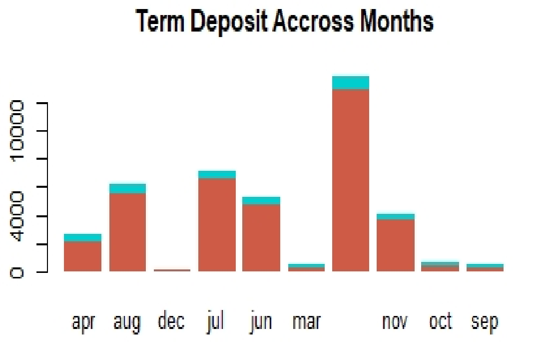
This can be achieved by using **SMOTE** function from **DMwR** package.

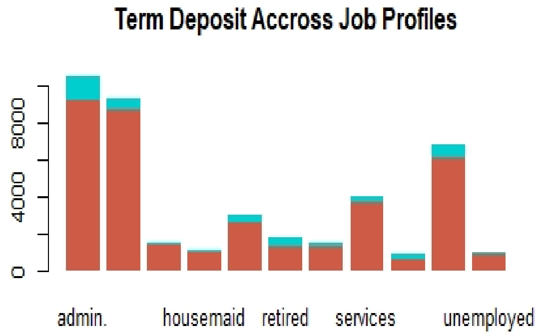
Note: SMOTE is done one the training set only.

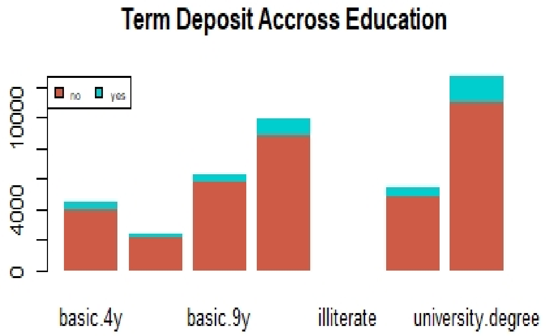
balancedTRAIN ← SMOTE(y ~ ., data = trainSET, perc.over = 500, perc.under = 100, k = 3)

BEFORE APPLYING SMOTE

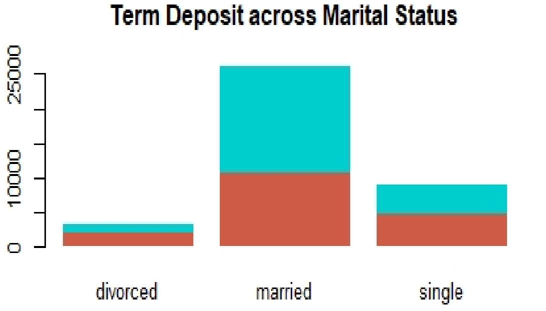


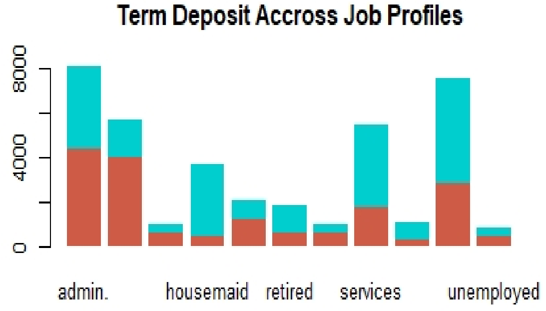


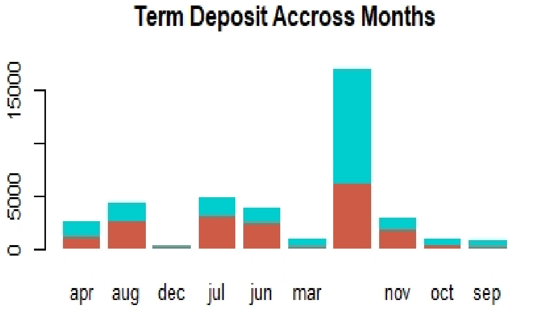


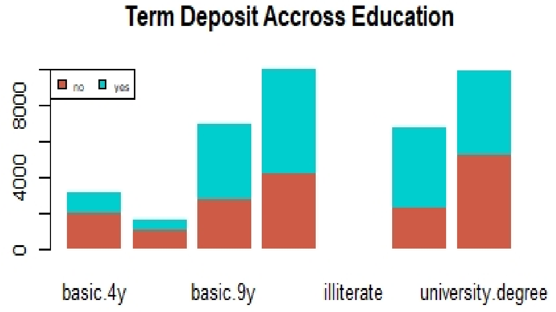


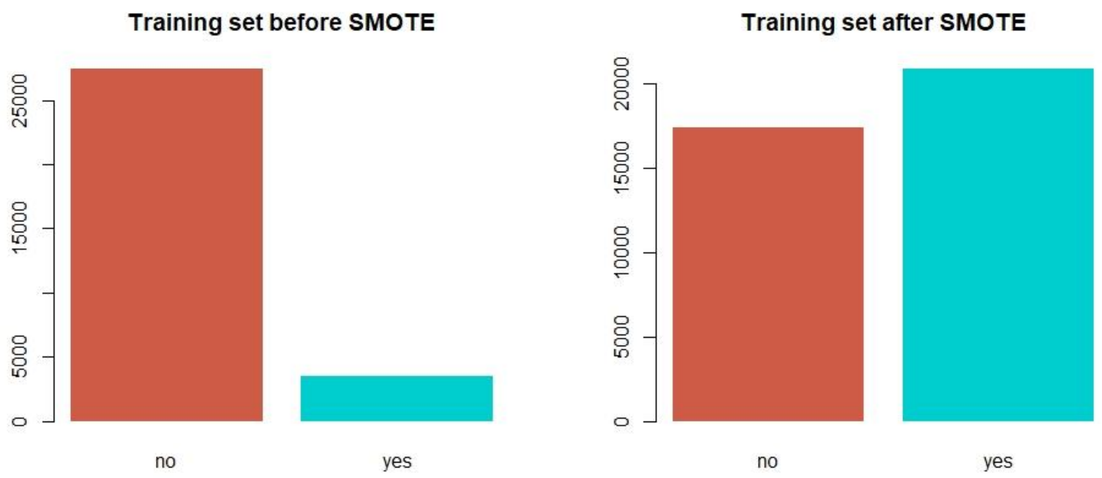
AFTER APPLYING SMOTE











Now that our dataset is cleaned and balanced we can proceed to build our model.

Model Building

Libraries Imported

library(ROSE)
library(dplyr)
library(ggplot2)
library(caret)
library(pROC)
library(VIM)
library(DMwR)
library(Amelia)
library(rattle)
library(rapportools)
library(randomForest)
library(ROCR)
library(e1071)

Algorithms used

This is bi-variate classification problem. There are many machine learning algorithms that can be used. However, in this scenario, we have built 9 different models using 9 ML algorithms.

1. Generalized Linear Model (GLM Logistic Regression)
2. Decision Tree
3. Random Forest
4. Naïve Bayes
5. KNN
6. Bagged CART (Classification and Regression Trees)
7. SVM Linear kernel
8. SVM Radial kernel
9. SVM Polynomial kernel

Significant Variables

After the base models were built, **feature selection** was done based on the **variable importance charts** mainly to –

1. Simplify models to make it easier to interpret.
2. Shorter training time.
3. Avoid curse of dimensionality.
4. Enhance generalization by reducing overfitting.

Variable Importance Charts

The charts below show us how important each variable is in its respective model.

Note: Variable Important Charts are not generated for SVM models.

Generalized Linear Model (GLM)

	Overall
duration	80.311
educationbasic.9y	21.836
day_of_weekmon	20.916
euribor3m	19.205
emp.var.rate	18.899
housingyes	18.625
monthnov	17.433
educationuniversity.degree	15.332
campaign	15.022
\`jobblue-collar\`	12.369
nr.employed	12.101
monthjun	10.713
monthoct	10.533
monthmar	10.484
pdays	9.952
jobentrepreneur	9.044
educationhigh.school	8.218
educationprofessional.course	7.631
monthjul	7.606
monthsep	7.493

Decision Tree

	Overall
duration	10623.534
nr.employed	8832.936
emp.var.rate	8816.997
euribor3m	7687.263
cons.conf.idx	6081.825
cons.price.idx	1509.568
pdays	1149.837
monthmay	143.642
monthoct	73.152
poutcomesuccess	11.452
contacttelephone	8.144
age	7.383
monthaug	6.284
\`jobblue-collar\`	0.000
maritalmarried	0.000
monthsep	0.000
educationuniversity.degree	0.000
jobretired	0.000
educationhigh.school	0.000
day_of_weekthu	0.000

Random Forest

	Overall
duration	6.392677e+03
emp.var.rate	2.146126e+03
nr.employed	2.145666e+03
euribor3m	1.829386e+03
cons.conf.idx	1.454842e+03
cons.price.idx	9.880876e+02
job	6.761971e+02
month	4.738364e+02
age	4.396639e+02
pdays	3.293054e+02
day_of_week	3.134631e+02
campaign	2.839392e+02
education	2.565634e+02
previous	1.733011e+02
housing	1.249976e+02
marital	1.170853e+02
poutcome	1.060367e+02
contact	9.564997e+01
loan	6.002859e+01
default	3.249418e-04

Naive Bayes

	Importance
duration	0.9024
nr.employed	0.7078
euribor3m	0.6592
emp.var.rate	0.6573
cons.conf.idx	0.6378
housing	0.6011
previous	0.5981
pdays	0.5969
age	0.5957
month	0.5807
campaign	0.5791
day_of_week	0.5732
contact	0.5614
poutcome	0.5561
loan	0.5371
job	0.5305
education	0.5139
marital	0.5106
cons.price.idx	0.5088
default	0.5001

KNN

	Importance
duration	0.9024
nr.employed	0.7078
euribor3m	0.6592
emp.var.rate	0.6573
cons.conf.idx	0.6378
housing	0.6011
previous	0.5981
pdays	0.5969
age	0.5957
month	0.5807
campaign	0.5791
day_of_week	0.5732
contact	0.5614
poutcome	0.5561
loan	0.5371
job	0.5305
education	0.5139
marital	0.5106
cons.price.idx	0.5088
default	0.5001

Bagged CART

	Overall
duration	13264.9
nr.employed	8984.4
emp.var.rate	8887.4
euribor3m	8755.3
cons.conf.idx	6292.1
cons.price.idx	2038.0
age	1542.3
pdays	1472.8
campaign	717.3
monthmay	352.3
housingyes	346.3
educationuniversity.degree	293.3
maritalmarried	289.0
maritalsingle	261.5
previous	252.5
day_of_weekmon	252.3
loanyes	221.2
day_of_weekwed	221.2
jobblue-collar	220.5
day_of_weektue	213.7

- It is seen that **Default** variable is the least significant in every model. This is ideal as the column itself is almost constant (only 3 Yes).
- **Duration** seems to be the most significant variable out of all, followed by **nr.employed**, **euribor3m** and **emp.var.rate**.

Comparison between Models

Models were assessed by predicting on the testing set, looking at the **confusion matrix** and compared on the basis of four metrics – **Accuracy**, **Sensitivity**, **Specificity**, **AUC** (Area under the Curve).

Below, we will see a sample **confusion matrix** and how the metrics differ between each model.

```
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  8054  302
       yes 1083  858

               Accuracy : 0.8655
                 95% CI : (0.8588, 0.872)
    No Information Rate : 0.8873
    P-Value [Acc > NIR] : 1

                  Kappa : 0.48
 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.73966
            Specificity : 0.88147
         Pos Pred Value : 0.44204
         Neg Pred Value : 0.96386
             Prevalence : 0.11265
         Detection Rate : 0.08333
   Detection Prevalence : 0.18850
      Balanced Accuracy : 0.81056

       'Positive' Class : yes
```

- Accuracy, Sensitivity and Specificity can be seen from the confusion matrix.
- It would be **profitable** situation for the company to find **how many people subscribe to the term deposi**t than to see how many people don't subscribe.
- Hence, our focus is on **reducing** the number of **False Negatives (FN)** and **increasing** the number of **True Positives (TP)**.
- This can be assessed using **Sensitivity**. It is also called as **Recall**.

  Sensitivity = TP/(TP+FN)
- **Sensitivity** tells us the **True Positive Rate**, whereas **Specificity** tells us the **True Negative Rate**.

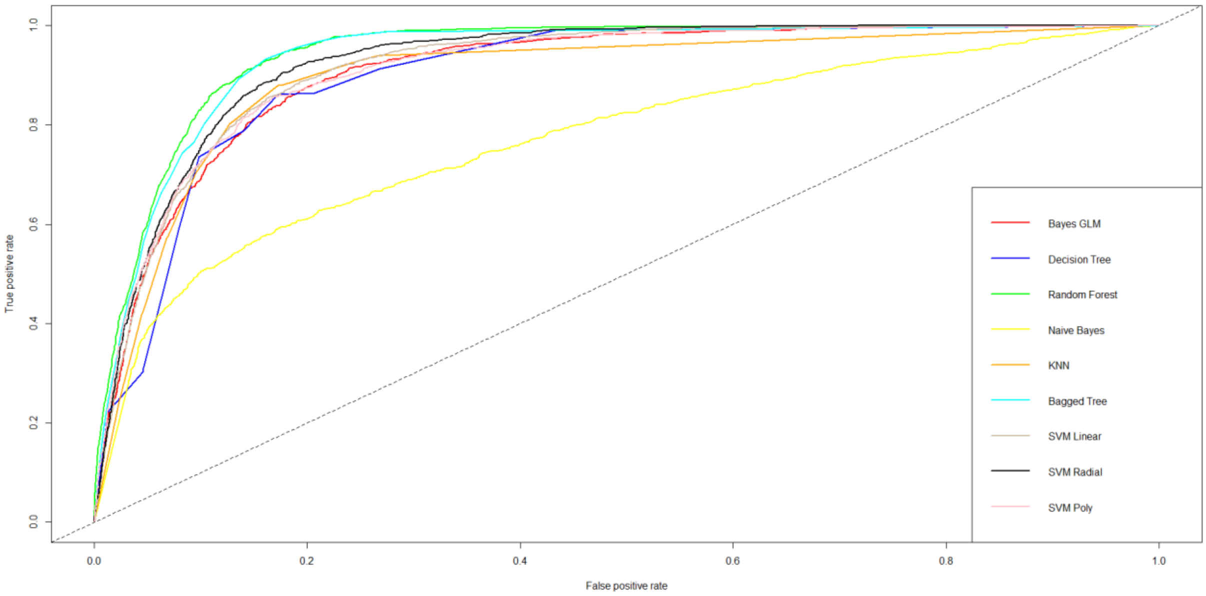
  Specificity = TN/(TN+FP)

Tabular Comparison

Model	Accuracy	Sensitivity	Specificity
GLM	0.8691852	0.7293103	0.8869432
Decision Tree	0.8828785	0.7362069	0.9014994
Random Forest	0.9050209	0.7542759	0.9280946
Naïve Bayes	0.8011071	0.5905172	0.8278428
KNN	0.8815189	0.7034483	0.9041261
Bagged CART	0.8975430	0.7431034	0.9171500
SVM Linear	0.8734583	0.7551724	0.8884754
SVM Radial	0.8912305	0.6939655	0.9162745
SVM Poly	0.8860833	0.7025862	0.9093794

- **Random Forest** has the best Accuracy and Specificity.
  However, **SVM Linear** gives us a slightly better Sensitivity, but a lower Specificity and Accuracy.

- We will further plot the **'Area under the Receiver Operating Characteristic Curve' (AUROC)**, to choose the best model.



- Random Forest has the highest AUC, followed by Bagged CART. SVM Linear has a relatively lower AUC.

Conclusion

Random Forest outperformed other models with the highest metrics and an AUC of ~0.9435. The model is 90.5% accurate in predicting if a person will subscribe to a term deposit or not. The model is also very sensitive and specific to the same.

This model can now be used by the bank to make the telemarketing campaign much more efficient and help them in targeting and securing key clients.

Data-driven decision making (DDDM) is very powerful and accurate than decisions that are intuitive or based or observation alone. Analytical techniques allow us to understand and stimulate demand, develop an efficient production plan, effectively source and allocate production resources, and lower distribution costs. Across all industries, many companies are excelling at applying these techniques, recognizing them as necessary to maintain a competitive advantage.

I would like to conclude by saying that analytics is one of the most powerful resources of our generation and tools like R, Python, SAS have made it even more easier and flexible.

References

1. www.cran.r-project.org (R packages)

2. www.en.wikipedia.org (Theoretical Information)

3. An Introduction to Statistical Learning in R (ISLR)
   by University of Southern California (Statistical Information)

4. www.archive.ics.uci.edu (Dataset sourcing)

5. StatQuest – YouTube Channel (Statistics)