1.0 Introduction

A common technique for growing an organization is to run marketing selling campaigns. Direct marketing is used by businesses to reach certain categories of customers to achieve a specific goal. Customer distant interactions may be integrated into a support department, making campaign administration easier. Customers can communicate with these centers via a variety of means, the most common of which is the cellphone. (Moro, et al., 2014) The data analyzed in this paper is connected to a banking institution's direct marketing initiatives. Customers were encouraged to sign up for a term deposit through a telephone marketing campaign. To determine the prediction old dataset was used if the product (bank term deposit) would be subscribed ('yes') or not ('no'). There were 41188 observations with 22 variables out of which 10 are numerical variables. 4640 customers had subscribed for bank term deposits which are around 8.9% customers of the whole data set. We must first import and pre-process data before we can start building the model.

Before being analyzed for outliers and missing values, the dataset is first to read into a variable. The data is cleaned and separated into two files, test and train, before being used to build a regression model. With the use of a summary table and assumptions, insights are provided. Caret, corrplot, lmtest, psych, ggplot2, dplyr, and car are some of the R programs utilized. Model 1, Model 2, and Model 3 are three logistic regression models using 3, 6, and 9 variables, respectively. The dataset is explored using a logistic regression model with ten variables in total. The dependent variable is 'y,' which represents whether a term deposit is subscribed or not. The variables that are utilized as independent variables include job, default, contact, month, duration, poutcome, emp.var.rate, euribor3m and nr.employed.

2.0 Methodology

The CRISP-DM (Cross-Industry Standard Process for Data Mining) is a widely used approach for improving the success of data mining operations. The technique specifies a non-linear series of six steps that enable the creation and deployment of a DM model in a real-world setting, assisting with business choices. (Chapman, et al., 2000) CRISP-DM would be considered to understand the step by step process which was utilized in this paper to analyse the banking data set.

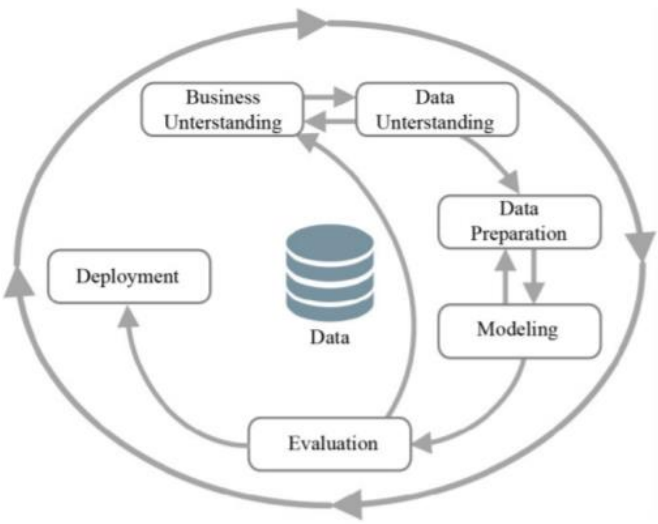

Fig. 1 Shows the CRISP-DM process

2.1. *Business Understanding:*

It is important to note that banks are under enormous pressure to raise their financial assets related to internal competitiveness. To address this problem, one technique used is to promote appealing long-term deposit applications with high-interest rates, particularly through targeted marketing initiatives. The same factors are also pushing for cost and time savings. As a result, efficiency must be improved: fewer contacts should be made, but an approximate number of successes (customers subscribing to the deposit) should be retained. (Moro & Laureano, 2011)

2.2. *Data Understanding:*

Once test and train banking data is downloaded into the system. It is then loaded into R-Studio. Attributes, properties, size, and structure are analysed to get better insights. The 2 excel file – test and train banking data sets were merged
For better insights, data quality issues and observations were revealed by a more thorough investigation. colSums(is.na()) and summary() function was used to analyze whether there are any NA values and basic summary of the data set. Looking for new frequency patterns in data would be a relevant approach in this case of a predictive deposit scenario with the goal of detecting the number of subscribers and analyzing the important attributes.

2.3. *Data Preparation:*

The engineer gathers necessary data and prepares it for the real task during the "Data Preparation" phase. This comprises data reduction and filtering, as well as preparation. *(Hubera, et al., 2019)* Summarizing the age of customers it was observed that the lowest age was 3 and the highest age was 170, it seemed to be uncertain values. So, the age below 17 and above 98 were replaced by the mean values. 'Education' attribute 'basic.4y', 'basic.6y' and 'basic.9y' were replaced with 'basic' to get proper understanding. For month attribute re-leveling of months was done to get proper insights in ggplot visualization. The y variable with 'yes' or 'no' was replaced with 'Subscribed' and 'Not Subscribed'. A new table was created using the filter() function which consisted of the values where term deposit (y) was only 'yes'. It was also made sure that all the character variables were converted into factors. The chisq.test() function (*Pearson's Chi-squared test)* was used to test hypotheses about the relationship between categorical variables. Pearson's correlation was also used to understand the relationship between 2 numerical variables. Once, the data quality issue and sorting were done the data was split into 80-20 partition which was named as test and train data.

2.4. *Modelling*

The "modeling" step is designed to determine the required parameter values for the chosen algorithms and to run the data analytics task on the preprocessed data. Using the test and train dataset 3 models are generated with assumptions and checks MODEL 1 consisting of 'default', 'contact' and 'poutcome'. Model 2 consists of 'default', 'contact', 'poutcome', 'month', 'duration' and 'emp.var.rate' and Model 3 consist of 'job', 'euribor3m' and 'nr.employed' extra variable to model 2. NOTE: 'duration' variable is taken for benchmark purpose, to increase the accuracy. If this model was about to be implemented in the industry then it won't be a fair practice to take duration attribute. When the target (dependent) variable is categorical and has two categories, logistic regression should be applied. To do a logistic regression in R, should use glm() function. The postResample() method is used to determine accuracy and Kappa values. The models that will be used will be examined to assess their correctness, and the confusion matrix will be used to determine this accuracy. The residuals are calculated using the resid() function. The VIF() function is used to determine whether there is a problem with multicollinearity.

2.5. Evaluation

The trained model is tested against real data sets in a production situation during the "Evaluation" step, and the outcomes are evaluated against the underlying business objectives. Test data sets are created for this purpose by following the processes outlined in the "Data Preparation" and "Modeling" stages. *(Hubera, et al., 2019)* The model with high accuracy would be taken into consideration. *The coefficients are interpreted in the same way as linear regression coefficients are. The coefficient indicates the change in the logit of the outcome variable caused by a one-unit change in the predictor variable. A pseudo R square can be used to evaluate a logistic regression model (it has a similar interpretation to the R squared in R).* The residuals are obtained using the resid() function. The residuals are useful for determining how well the model matches the data. Residuals above 1.96 were calculated. As a rule of thumb, only 5% should lie outside of ± 1.96.

2.6. Deployment

Creation of the Logistic regression model is not the endpoint of the project. Typically, the knowledge gathered must be arranged and presented in such a way that the consumer can make use of it. The deployment step might be as easy as creating a report or as sophisticated as establishing a repeatable data mining process, depending on the needs. In many circumstances, the user, not the data analyst, will do the deployment processes. In any instance, it is critical to understand what steps must be taken ahead of time to use the models that have been built.

3. Results (Descriptive statistics, visualisation, and measures of association)

3.1. Descriptive Statistics

Group	Age			Total
	Mean	Median	Max	Count
Admin	38.19	36	72	10422
Blue-Collar	39.56	39	80	9254
Entrepreneur	41.72	41	69	1456
Housemaid	45.5	45	85	1060
Management	42.36	42	80	2924
Retired	62.03	59	98	1720
Self-employed	39.95	39	71	1421
Services	37.93	36	69	3969
Student	25.9	25	47	875
technician	38.51	37	70	6743

Table 1. Shows descriptive statistics of jobs of customers with respect to age.

Loan	
Housing	Personal
21576	6245

Table 2. Shows Customer count with loan

Divorced	Single	Married	unknown
4612	11568	24928	80

Table 3. Shows Marital Status

3.2. Data Visualisations

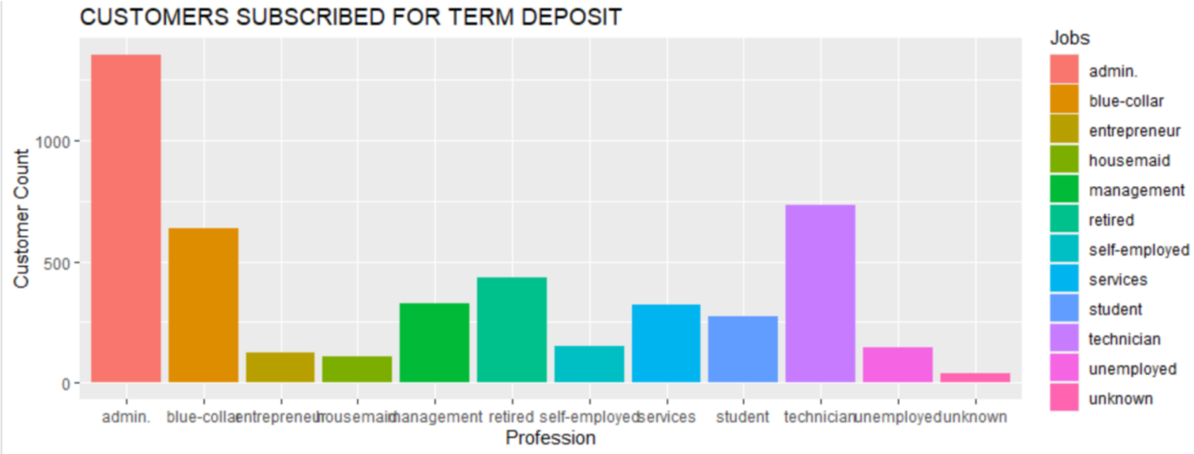

Fig 2. Shows Customers subscribed for term deposit with respect to profession (BAR)

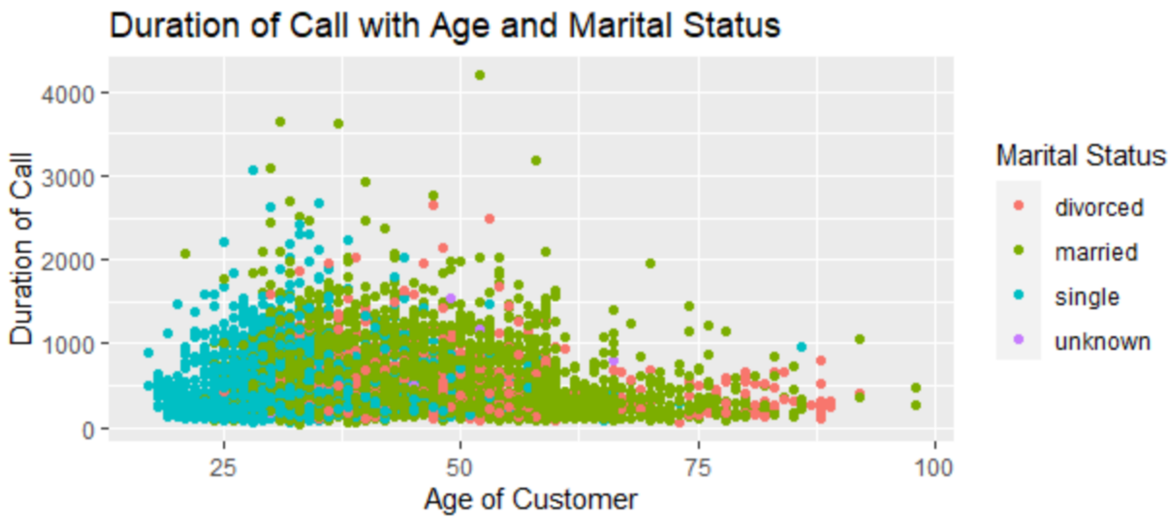

Fig 3. Shows subscribed customer with term deposit with marital status and age (POINT)

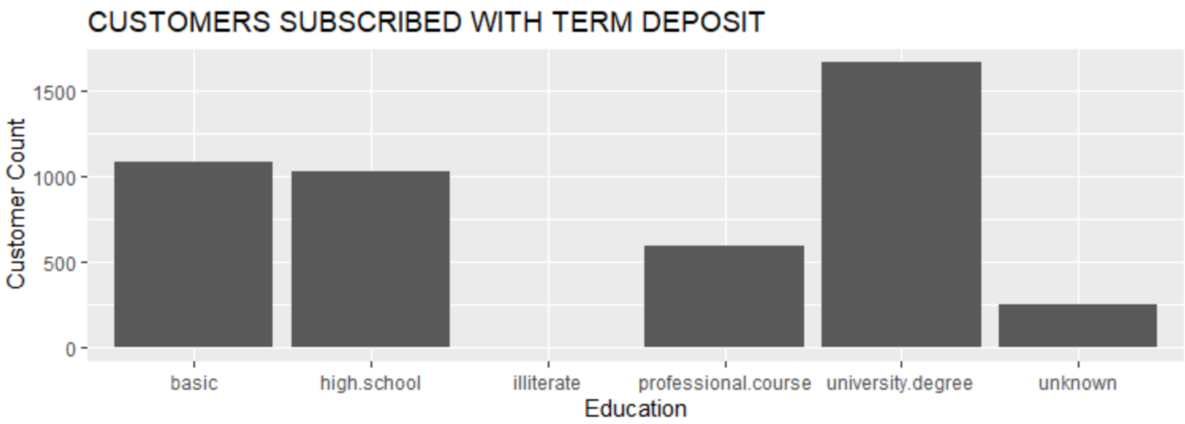

Fig 4. Shows subscribed customer with term deposit and education background (BAR)

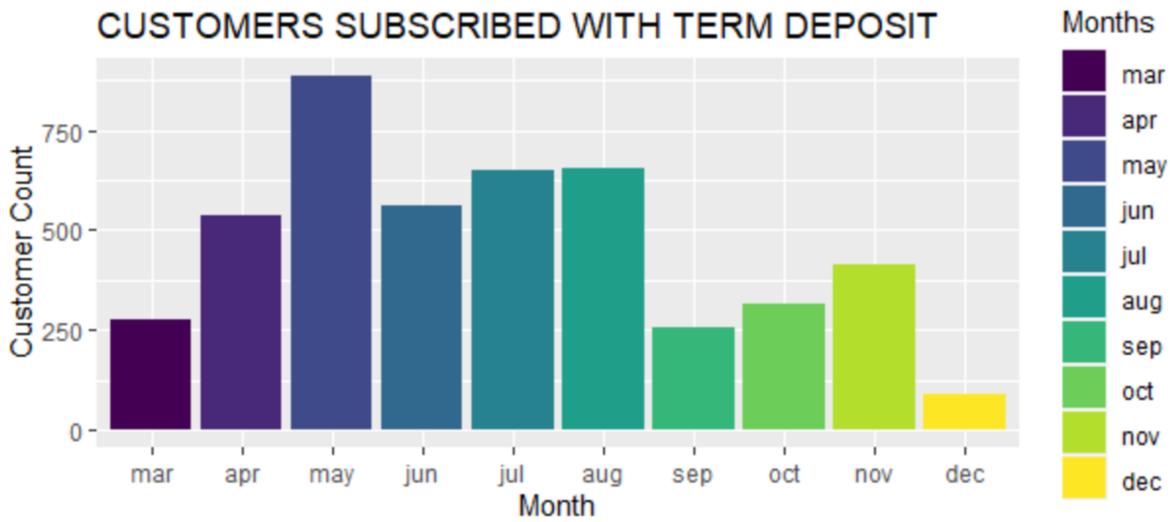

Fig 5. Shows subscribed customer with term deposit and Months (BAR)

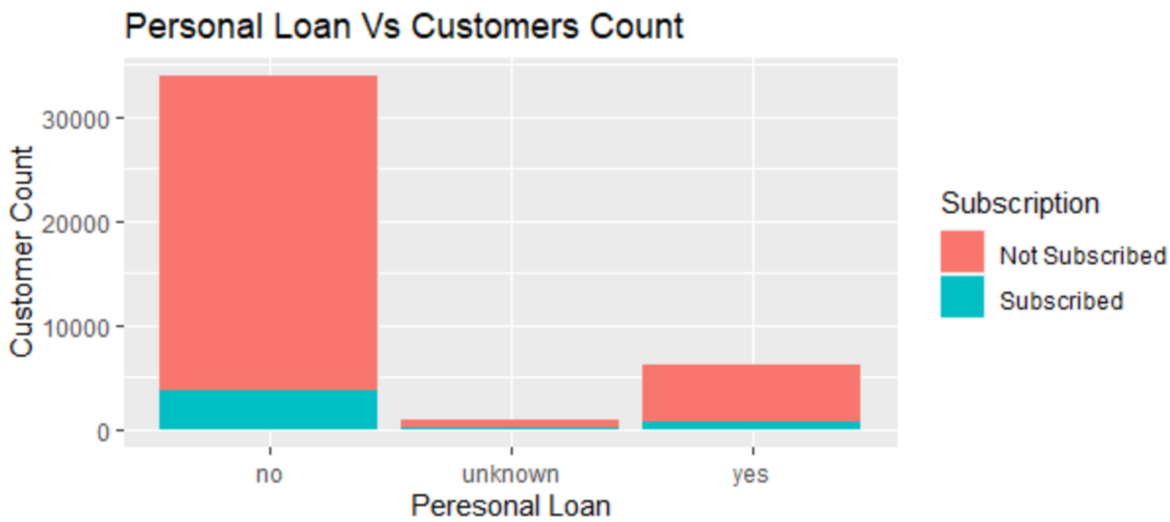

Fig 6. Shows Customers who have taken personal loan with respect to subscription

3.3. Measures of Association

Month	Mar	Apr	May	June	July	Aug	Sep	Oct	Nov	Dec
Not Subscribed	270	2093	12883	4759	6525	5523	314	403	3685	93
Subscribed	276	539	886	559	649	655	256	315	416	89
Pearson's Chi-squared test data: bank$y and bank$month X-squared = 3101.1, df = 9, p-value < 2.2e-16										

Table 4. Shows p-value and tabular comparison of Month and subscription

Previous Outcome	Failure	Non Existent	Success
Not Subscribed	3647	32422	479
Subscribed	605	3141	894
Pearson's Chi-squared test data: bank$y and bank$poutcome X-squared = 4230.5, df = 2, p-value < 2.2e-16			

Table 5. Shows p-value and tabular comparison of Outcome of the previous marketing and subscription

Marital Status	Divorced	Married	Single	Unknown
Not Subscribed	4136	22396	9948	68
Subscribed	476	2532	1620	12
Pearson's Chi-squared test data: bank$y and bank$marital X-squared = 122.66, df = 3, p-value < 2.2e-16				

Table 6. Shows p-value and tabular comparison of Marital Status and subscription

Contact	Cellular	Telephone
Not Subscribed	22291	14257
Subscribed	3853	787
Pearson's Chi-squared test		
data: bank$y and bank$contact		
X-squared = 863.27, df = 1, p-value < 2.2e-16		

Table 7. Shows p-value and tabular comparison of contact and subscription

Consumer Price Index and confidence Index			
Correlation	p-value <	t	df
0.0384699	5.72E-15	7.813	41186
95 percent confidence interval:			
0.02882305 0.04810949			

Table 8. Shows correlation between Consumer price index and Consumer confidence index

4. Regression Analysis Results

A. Logistics Regression Model 1

```
> summary(model1)

Call:
glm(formula = formula1, family = "binomial", data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5072  -0.5201  -0.3633  -0.3429   2.6776

Coefficients:
                    Estimate Std. Error z value              Pr(>|z|)
(Intercept)         -1.65368    0.04939 -33.483 < 0.0000000000000002 ***
defaultunknown      -0.75245    0.05984 -12.574 < 0.0000000000000002 ***
defaultyes          -9.73218  113.53237  -0.086                0.932
contacttelephone    -0.87196    0.04767 -18.293 < 0.0000000000000002 ***
poutcomenonexistent -0.27867    0.05445  -5.118          0.000000309 ***
poutcomesuccess      2.40225    0.08072  29.762 < 0.0000000000000002 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 23146  on 32884  degrees of freedom
Residual deviance: 20478  on 32879  degrees of freedom
AIC: 20490

Number of Fisher Scoring iterations: 10
```

Fig.7 Shows Summary of Model 1

Model 1	
Accuracy	Kappa
0.89635	0.232578

Table 9. Shows Accuracy and Kappa value for Model 1

```
>   confusionMatrix(class_pred1, test$y)
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  7212  769
       yes   83  156

               Accuracy : 0.8964
                 95% CI : (0.8896, 0.9029)
    No Information Rate : 0.8875
    P-Value [Acc > NIR] : 0.005298

                  Kappa : 0.2326

 Mcnemar's Test P-Value : < 0.00000000000000022

            Sensitivity : 0.9886
            Specificity : 0.1686
         Pos Pred Value : 0.9036
         Neg Pred Value : 0.6527
             Prevalence : 0.8875
         Detection Rate : 0.8774
   Detection Prevalence : 0.9709
      Balanced Accuracy : 0.5786

       'Positive' Class : no
```

Fig. 8 Shows Confusion Matrix for Model 1

```
>   exp(model1$coefficients)
    (Intercept)      defaultunknown          defaultyes    contacttelephone poutcomenonexistent
  0.19134433637       0.47121289196       0.00005934284       0.41812962987       0.75679260791
poutcomesuccess
 11.04804656155
```

Fig. 9 Shows Odds Ratios for Model 1

B. Logistics Regression for Model 2

```
> summary(model2)

Call:
glm(formula = formula2, family = "binomial", data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.3675  -0.3280  -0.2031  -0.1405   2.9182

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)         -3.865e+00  8.091e-02 -47.765  < 2e-16 ***
defaultunknown      -4.476e-01  7.241e-02  -6.182 6.33e-10 ***
defaultyes          -7.570e+00  1.135e+02  -0.067 0.946816
contacttelephone    -1.430e-01  6.681e-02  -2.140 0.032360 *
poutcomenonexistent  2.185e-01  6.589e-02   3.315 0.000915 ***
poutcomesuccess      2.170e+00  9.186e-02  23.625  < 2e-16 ***
month.L             -2.402e-01  1.262e-01  -1.904 0.056901 .
month.Q              5.463e-01  1.322e-01   4.132 3.59e-05 ***
month.C             -1.146e+00  1.226e-01  -9.344  < 2e-16 ***
month^4              1.196e+00  9.834e-02  12.158  < 2e-16 ***
month^5              4.418e-03  9.090e-02   0.049 0.961231
month^6              3.810e-01  8.404e-02   4.534 5.78e-06 ***
month^7              6.163e-01  7.572e-02   8.140 3.96e-16 ***
month^8             -2.694e-01  8.487e-02  -3.174 0.001501 **
month^9              1.272e-01  7.219e-02   1.763 0.077978 .
duration             4.519e-03  8.025e-05  56.308  < 2e-16 ***
emp.var.rate        -6.242e-01  1.815e-02 -34.391  < 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: 23146  on 32884  degrees of freedom
Residual deviance: 14272  on 32868  degrees of freedom
AIC: 14306

Number of Fisher Scoring iterations: 10
```

Fig. 10 Shows Summary of Model 2

Model 2	
Accuracy	Kappa
0.911071	0.453363

Table 10. Shows Accuracy and Kappa value for Model 2

```
>   confusionMatrix(class_pred2, test$y)
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  7126  562
       yes  169  363

               Accuracy : 0.9111
                 95% CI : (0.9047, 0.9171)
    No Information Rate : 0.8875
    P-Value [Acc > NIR] : 0.000000000001522

                  Kappa : 0.4534

 Mcnemar's Test P-Value : < 0.0000000000000022

            Sensitivity : 0.9768
            Specificity : 0.3924
         Pos Pred Value : 0.9269
         Neg Pred Value : 0.6823
             Prevalence : 0.8875
         Detection Rate : 0.8669
   Detection Prevalence : 0.9353

      Balanced Accuracy : 0.6846

       'Positive' Class : no
```

Fig. 11 Shows Confusion Matrix for Model 1

```
exp(model2$coefficients)
    (Intercept)      defaultunknown          defaultyes    contacttelephone poutcomenonexistent
   0.0209690580        0.6391475237        0.0005158335        0.8667815819        1.2441594029
poutcomesuccess             month.L             month.Q             month.C             month^4
   8.7596046948        0.7864673202        1.7268010695        0.3179168561        3.3054175842
        month^5             month^6             month^7             month^8             month^9
   1.0044282539        1.4637851220        1.8521220013        0.7638386673        1.1356853576
       duration        emp.var.rate
   1.0045289284        0.5356760976
```

Fig. 12 Shows Odds Ratios for Model 2

C. Logistics Regression for Model 3

```
Call:
glm(formula = Formula3, family = "binomial", data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.5950  -0.3021  -0.1893  -0.1392   3.0631

Coefficients:
                       Estimate   Std. Error z value             Pr(>|z|)
(Intercept)         85.61862930   6.09649408  14.044 < 0.0000000000000002 ***
defaultunknown      -0.33079466   0.07421705  -4.457     0.00000830661035 ***
defaultyes          -7.41455451 113.49419194  -0.065             0.947911
contacttelephone    -0.47438750   0.07560587  -6.274     0.00000000035081 ***
poutcomenonexistent  0.40496422   0.06856604   5.906     0.00000000350106 ***
poutcomesuccess      1.84128181   0.09432691  19.520 < 0.0000000000000002 ***
month.L             -1.05350620   0.15416449  -6.834     0.00000000000828 ***
month.Q              0.53172332   0.13263358   4.009     0.00006098558927 ***
month.C             -0.37146762   0.12985044  -2.861             0.004227 **
month^4              1.06485121   0.09964973  10.686 < 0.0000000000000002 ***
month^5             -0.31946381   0.09316159  -3.429             0.000606 ***
month^6             -0.21784718   0.08988789  -2.424             0.015370 *
month^7              0.41982523   0.07701203   5.451     0.00000004996811 ***
month^8             -0.02132056   0.08587866  -0.248             0.803930
month^9              0.45602609   0.07506464   6.075     0.00000000123901 ***
duration             0.00458725   0.00008175  56.112 < 0.0000000000000002 ***
emp.var.rate        -0.62368063   0.06163952 -10.118 < 0.0000000000000002 ***
jobblue-collar      -0.33008621   0.07385353  -4.469     0.00000784132604 ***
jobentrepreneur     -0.11581483   0.13828398  -0.838             0.402303
jobhousemaid        -0.01392170   0.15370869  -0.091             0.927833
jobmanagement        0.00585378   0.09220252   0.063             0.949378
jobretired           0.29055027   0.09235122   3.146             0.001654 **
jobself-employed    -0.13041958   0.13118204  -0.994             0.320131
jobservices         -0.15792711   0.09053287  -1.744             0.081086 .
jobstudent           0.24293749   0.11085214   2.192             0.028412 *
jobtechnician       -0.04384211   0.07081886  -0.619             0.535868
jobunemployed        0.01865111   0.13722928   0.136             0.891891
jobunknown          -0.27625796   0.28251300  -0.978             0.328144
euribor3m            0.70995265   0.10136169   7.004     0.00000000000248 ***
nr.employed         -0.01787621   0.00124441 -14.365 < 0.0000000000000002 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 23146  on 32884  degrees of freedom
Residual deviance: 13826  on 32855  degrees of freedom
AIC: 13886

Number of Fisher Scoring iterations: 10
```

Fig. 13 Shows Summary of Model 3

Model 3	
Accuracy	**Kappa**
0.91253	0.465434

Table 11. Shows Accuracy and Kappa value for Model 3

```
>   confusionMatrix(class_pred3, test$y)
Confusion Matrix and Statistics

          Reference
Prediction   no  yes
       no  7127  551
       yes  168  374

               Accuracy : 0.9125
                 95% CI : (0.9062, 0.9186)
    No Information Rate : 0.8875
    P-Value [Acc > NIR] : 0.00000000000005628

                  Kappa : 0.4654

 Mcnemar's Test P-Value : < 0.0000000000000022

            Sensitivity : 0.9770
            Specificity : 0.4043
         Pos Pred Value : 0.9282
         Neg Pred Value : 0.6900
             Prevalence : 0.8875
         Detection Rate : 0.8670
   Detection Prevalence : 0.9341
      Balanced Accuracy : 0.6906

       'Positive' Class : no
```

Fig. 14 Shows Confusion Matrix for Model 1

4.1. Prediction and Accuracy

A. Prediction and Accuracy for Model 1

```
>   head(data.frame(train$predictedProbabilities, train$y))
  train.predictedProbabilities train.y
1                   0.05709169      no
2                   0.05709169      no
3                   0.05709169      no
4                   0.02773979      no
5                   0.05709169      no
6                   0.05709169      no
```

Fig. 15 shows probability of y, and the actual outcome

```
>   sum(train$standardisedResiduals > 1.96)
[1] 2543
```

Fig. 15 Shows Residual above 1.96

```
>   #check if any values are above 0.0009
>   sum(train$leverage > 0.0009)
[1] 132
```

Fig. 16 Shows Leverage for Model 1

```
>   vif(model1)
             GVIF Df GVIF^(1/(2*Df))
default  1.010992  2        1.002737
contact  1.045575  1        1.022534
poutcome 1.045416  2        1.011166
```

Fig. 17 Shows VIF for model 1

B. Prediction and Accuracy for Model 2

```
>   head(data.frame(train$predictedProbabilities, train$y))
  train.predictedProbabilities train.y
1                  0.010635908      no
2                  0.007601875      no
3                  0.015264975      no
4                  0.006553930      no
5                  0.021104342      no
6                  0.004829753      no
```

Fig. 18 shows probability of y, and the actual outcome for model 2

```
>   sum(train$standardisedResiduals > 1.96)
[1] 684
```

Fig. 19 Shows Residual above 1.96 for model 2

```
>   sum(train$leverage > 0.0009)
[1] 5438
```

Fig. 20 Shows Leverage for Model 2

```
>   vif(model2)
                 GVIF Df GVIF^(1/(2*Df))
default      1.063399  2        1.015486
contact      1.403600  1        1.184736
poutcome     1.173821  2        1.040879
month        1.774207  9        1.032366
duration     1.194334  1        1.092856
emp.var.rate 1.915135  1        1.383884
```

Fig. 21 Shows VIF for model 2

C. Prediction and Accuracy for Model 3

```
  train.predictedProbabilities train.y
1                  0.014664374      no
2                  0.012204412      no
3                  0.021124029      no
4                  0.008562189      no
5                  0.032702986      no
6                  0.006594401      no
```

Fig. 22 Shows probability of y, and the actual outcome for model 3

```
>   sum(train$standardisedResiduals > 1.96)
[1] 641
```

Fig. 23 Shows Residual above 1.96 for model 3

```
>   sum(train$leverage > 0.0009)
[1] 5438
```

Fig. 24 Shows Leverage for Model 3

```
                  GVIF Df GVIF^(1/(2*Df))
default       1.104628  2        1.025189
contact       1.785967  1        1.336401
poutcome      1.310100  2        1.069858
month         5.445002  9        1.098724
duration      1.227564  1        1.107955
emp.var.rate 21.511727  1        4.638074
job           1.239027 11        1.009790
euribor3m    65.828072  1        8.113450
nr.employed  21.946117  1        4.684668
```

Fig. 25 Shows VIF for model 2

5. Discussion

It can be observed from Fig 1. that customer who has a job as Admin are the most subscribed customers for a term deposit. But only 1352 admins have subscribed out of 10422 which is around 13%. This is because a maximum of the customers have a job profile as admin. Observing Fig 2. it can be noted that maximum customers having a duration of call had marital status as married. This is due to maximum customers are married. The highest subscription rate is 14% with singles whereas for divorcees and married it's around 10%. Fig 4. depicts that the education background of customers having illiterate did not subscribe to the term deposit at all Customer with University degree as their education background were the maximum subscriber for the term deposit.

Observing Fig. 5, it can be analyzed that the maximum customers who subscribed to the term deposits are in May. It is the least subscription rate with 6.43%. The least customers subscribed month is December. We can also observe that there is around a 50% subscription rate in March and December. The bank office operations might not be operative in December due to the Christmas holidays with a 100% workforce, this might be one of the reason customer might not get support from the bank to subscribe to term deposits. Fig. 6, depicts that a customer who has not taken any personal loan has more subscribers than the one who has taken a personal loan. It can also be noted that customers with the personal loan are in minority.

Observing Table 4,5,6 and 7. It can be observed that month, previous outcome, marital status, and contact are statistically significant concerning variable y. As the p-value for all of them is 2.2e-16 which is lesser than 0.05. The lower the p-value, the greater the statistical significance of the observed difference.

While considering all the logistic regression models referring to figures 7, 10, and 13 it can be observed that the null deviance is greater than residual deviance. The null deviance is the deviance of the model with no predictors, while the residual deviance is the deviance of the model with the predictor. As a result, the null deviance should be greater than the residual deviation. It can also be observed from Table 16. 17. and 18. that the value of R-square is increasing as there is an increase in the attributes of a logistics regression model. From fig 12. it can be observed that the confidence interval is above 1 for 'poutcome', 'month', and 'duration'. If the confidence interval exceeds one, we cannot be certain of the relationship's direction (and the b will probably not be statistically significant).

Logistic Regression	no. of Attributes	Accuracy	Kappa	Residuals > 1.96	Leverage > 0.0009
MODEL 1	3	0.89635	0.232578	2543	132
MODEL 2	6	0.91107	0.453363	684	5438
MODEL 3	9	0.91253	0.465434	641	5438

Table 12. Shows comparison of the model

Analysing Table 12. it can be observed for all the 3 models that accuracy and kappa value increases as the number of attributes increases. For model 1 the kappa value was 0.23 from model 2 onwards the kappa value is above 0.3. The larger the kappa value the better is the model. Residual above 1.96 was analyzed and it is noticed that residual count was decreasing as the model accuracy was increasing and the leverage value above 0.0009 increased. GVIF variable was very high for 3 variables.

6. Conclusion

For the next marketing campaign, it would be better for the bank to focus on March and December instead of May. As the subscription rate is seen to be 50% around these 2 months. More Singles and youngsters in their 20s must be targeted as the probability of them subscribing to the term deposit is higher. Customers with no prior history of credit default have a greater probability of subscribing to the bank's term deposit. Customer with no default history must only be targeted for a better probability rate. To know the bank customers it needs to perform analysis timely basis so that the banking products/services offered to meet their demands and the sale is assured. As a result, the bank should be as involved in their clients' operations as possible, providing financial and logistical support, specialized consultation, and help. The bank must aim for a long-term competitive advantage through promoting good interest rates for term deposits, and the growth of customer loyalty. (Catalina, 2010)

7. *Appendix 1: R Code Used*

```
#Install the required packages
#Read the Packages
library(readxl)
library(psych)
library(ggplot2)
library(tidyverse)
library(dplyr)
library(caret) #to split the data
library(Hmisc) #For rcorr() function
library(corrplot)
library(lmtest)
library(car)

#Set Workind Directory
setwd('D:/Business Analytics/Statistics For Business/Assignment 2')

#Read the excel sheet into variable test and train
test <- read_excel('bank_test.xlsx')
train <- read_excel('bank_train.xlsx')

#Combine the train and test data into BANK
bank <- rbind(train,test)

#To remove 10E values
options(scipen = 10000)

# :: Summarize the Data ::
#Analyze the Columns with NA
colSums(is.na(bank))

#Summarise the data
summary(bank)

#::::: Data Quality Issues and Action ::::::

#AGE: Maximum Age is 170 and minimmum is 3. Change the age at appropriate value.
#Re summarize Age above 98 and below 17 as mean value
bank$age[(bank$age > 98)] <- mean(bank$age)
bank$age[(bank$age < 17)] <- mean(bank$age)

#Convert job into factor
bank$job <- as.factor(bank$job)

#Convert marital into factor
bank$marital <- as.factor(bank$marital)

#Convert education into  factor and Combine basic education
bank$education[bank$education == 'basic.4y'] <- 'basic'
bank$education[bank$education == 'basic.6y'] <- 'basic'
bank$education[bank$education == 'basic.9y'] <- 'basic'
```

```
bank$education <- as.factor(bank$education)

#In Contact change the name of 1 Mobile to cellular and contact as factor
bank$contact[bank$contact == 'mobile'] <- 'cellular'
bank$contact <- as.factor(bank$contact)

#Convert housing into factor
bank$housing <- as.factor(bank$housing)

#Convert loan into factor and change the name pf NA's to unknown
bank$loan <- as.factor(bank$loan)
bank$loan[is.na(bank$loan)] <- 'unknown'

#Convert Month in factor and level-up in sequence to show proper interpretation
bank$month <- as.factor(bank$month)
bank$month <- factor(bank$month, levels = c("jan", "feb", "mar", "apr", "may", "jun", "jul", "aug",
"sep", "oct", "nov", "dec"), ordered = TRUE)

#Convert day_of_week in factor and level-up in sequence to show proper interpretation (it consists
83 NA's)
bank$day_of_week <- as.factor(bank$day_of_week)
bank$day_of_week <- factor(bank$day_of_week, levels = c("mon", "tue", "wed", "thu", "fri"),
ordered = TRUE)

#Convert y variable into factor as 'yes' or 'no'
#bank$y[bank$y == 'yes'] <- 'Subscribed'
#bank$y[bank$y == 'no'] <- 'Not Subscribed'
bank$y <- as.factor(bank$y)

#bank with y column as yes
bank %>% filter(y == 'Subscribed') -> yes

#::: Descriptive Statistics :::

#1. descriptive statistics of jobs of customers with respect to age.
describeBy(x = bank$age, group = bank$job)

#2. Credit Default with respect to Marketing Campaign duration
describeBy(x = bank$duration, group = bank$default, na.rm = TRUE)

#3. Subscribed Customers with respect to Marketing Campaign duration
describeBy(x = bank$duration, group = bank$y, na.rm = TRUE)

#4. Customers who have taken Housing and Personal Loan
summary(bank$loan)
summary(bank$housing)

#5. Customer Count of Marital status
summary(bank$marital)

#6. Eduaction status count of customer
```

```
summary(bank$education)

#:: GGPLOT VISUALISATIONS::

#1. Customers subscribed for term deposit with respect to profession (BAR)
yes %>% ggplot(aes(x=job,,fill = job))+
  geom_bar()+
  labs(title = "CUSTOMERS SUBSCRIBED FOR TERM DEPOSIT", x="Profession", ,
       y= "Customer Count", fill = 'Jobs')+
  scale_y_continuous(labels = function(x) format(x, scientific = FALSE))

#2. subscribed customer with term deposit with marital status and age
yes %>% ggplot(aes(x = age, y=duration, color = marital),stat = "Summary", fun.y = "mean")+
  geom_point()+
  labs(title = "Duration of Call with Age and Marital Status", x="Age of Customer", y= "Duration of
Call", color = "Marital Status")+
  scale_y_continuous(labels = function(x) format(x, scientific = FALSE))
scale_fill_brewer(palette = "Dark2")

#3. subscribed customer with term deposit and education background
yes %>% ggplot(aes(x=education))+
  geom_bar()+
  labs(title = "CUSTOMERS SUBSCRIBED WITH TERM DEPOSIT", x="Education",
       y= "Customer Count")+
  scale_y_continuous(labels = function(x) format(x, scientific = FALSE))

#4.subscribed customer with term deposit and Months
  yes %>% ggplot(aes(x=month, fill = month))+
  geom_bar()+
  labs(title = "CUSTOMERS SUBSCRIBED WITH TERM DEPOSIT", x="Month",
       y= "Customer Count", fill = 'Months')+
  scale_y_continuous(labels = function(x) format(x, scientific = FALSE))

#5. Customers who have taken personal loan with respect to subscription
  bank %>% ggplot(aes(x = loan, fill = y))+
    geom_bar()+
    labs(title = "Personal Loan Vs Customers Count", x="Peresonal Loan", y= "Customer Count", fill =
"Subscription")+
    scale_y_continuous(labels = function(x) format(x, scientific = FALSE))
  scale_fil_brewer(palette = "Dark2")

#::: MEASURES OF ASSOSCIATION::::
#Chi Square Tests in R

#1.cross tabs for the variables MONTH
table(bank$y,bank$month)
#chisq.test() function to perform the test
chisq.test(bank$y, bank$month, correct = FALSE)

#2.cross tabs for the variables Outcome of previous Marketing
table(bank$y,bank$poutcome)
```

```
#chisq.test() function to perform the test
chisq.test(bank$y, bank$poutcome, correct = FALSE)

#3.cross tabs for the variables MONTH
table(bank$y,bank$marital)
#chisq.test() function to perform the test
chisq.test(bank$y, bank$marital, correct = FALSE)

#4. cross tabs for the variables MONTH
table(bank$y,bank$contact)
#chisq.test() function to perform the test
chisq.test(bank$y, bank$contact, correct = FALSE)

#5. Relationship between Consumer price index and Consumer confidence index
cor.test(x=bank$cons.conf.idx, y=bank$cons.price.idx)

#::::: SPLIT THE BANK DATA INTO TRAINING AND TEST::::
 #Delete TEST and TRAIN data first, it would set it as empty
 test <- NULL
 train <- NULL
 #to create a partition with 80%
 bank <- bank %>% filter(!is.na(day_of_week))
 bank<- bank %>% mutate_if(is.character, as.factor)
 set.seed(123) #generate a sequence of random numbers
 index <- createDataPartition(bank$y, p = 0.8, list = FALSE,)
 train <- bank[index, ] #first 80% for training
 test <- bank[-index, ] #bottom 20% for testing


 # ::: BUILD THE MODEL :::

#1. :::: Logistic Regression MODEL 1 ::::
formula1 <- y ~ default + contact + poutcome
model1 <- glm(formula1, data = train, family = "binomial")
#Summary of Logistic Regression MODEL 1
summary(model1)
#prediction using the model
predictions1 <- predict(model1,test,type ="response")
#Convert probabilities to yes or no
class_pred1 <-as.factor(ifelse(predictions1 > 0.5,"yes","no"))
#evaluate the accuracy of the predictions
 postResample(class_pred1,test$y)

#Confusion Matrix
 confusionMatrix(class_pred1, test$y)


#2. :::: Logistic Regression MODEL 2 ::::
 formula2 <- y ~ default + contact + poutcome + month + duration + emp.var.rate
 model2 <- glm(formula2, data = train, family = "binomial")
 #Summary of Logistic Regression MODEL 2
```

```
summary(model2)
#prediction using the model
predictions2 <- predict(model2,test,type ="response")
#Convert probabilities to yes or no
class_pred2<-as.factor(ifelse(predictions2 > 0.5,"yes","no"))
#evaluate the accuracy of the predictions
postResample(class_pred2,test$y)

#Confusion Matrix
confusionMatrix(class_pred2, test$y)

#3. :::: Logistic Regression MODEL 3 ::::
Formula3 <- y ~ default + contact + poutcome + month + duration + emp.var.rate + job + euribor3m
+ nr.employed
model3 <- glm(Formula3, data = train, family = "binomial")
#Summary of Logistic Regression MODEL 3
summary(model3)
#prediction using the model
Predictions3 <- predict(model3,test,type ="response")
#Convert probabilities to yes or no
class_pred3<-as.factor(ifelse(Predictions3 > 0.5,"yes","no"))
#evaluate the accuracy of the predictions
postResample(class_pred3,test$y)

#Confusion Matrix
confusionMatrix(class_pred3, test$y)
```

```
#Assessing Model R-Square
 logisticPseudoR2s <- function(LogModel) {
   dev <- LogModel$deviance
   nullDev <- LogModel$null.deviance
   modelN <- length(LogModel$fitted.values)
   R.l <-  1 -  dev / nullDev
   R.cs <- 1- exp ( -(nullDev - dev) / modelN)
   R.n <- R.cs / ( 1 - ( exp (-(nullDev / modelN))))
   cat("Pseudo R^2 for logistic regression\n")
   cat("Hosmer and Lemeshow R^2  ", round(R.l, 3), "\n")
   cat("Cox and Snell R^2        ", round(R.cs, 3), "\n")
   cat("Nagelkerke R^2           ", round(R.n, 3),    "\n")
 }
 logisticPseudoR2s(model1)
#Odds Ratio (Exponential of coefficient)
 exp(model1$coefficients)
 exp(model2$coefficients)
 exp(model3$coefficients)
#confidence interval
```

```
exp(confint(model1))
exp(confint(model2))
exp(confint(model3))


#:::::evaluate the model assumption::::

#::MODEL 1 ASSUMPTIONS::
 #Add the predicted probabilities to the data frame
 train$predictedProbabilities <- fitted(model1)

 #This shows the probability of churn, and the actual outcome.
 head(data.frame(train$predictedProbabilities, train$y))

 #Add the standardised and Studentised residuals can be added to the data frame
 train$standardisedResiduals <- rstandard(model1)
 train$studentisedResiduals <- rstudent(model1)

 #count the residuals above 1.96
 sum(train$standardisedResiduals > 1.96)

 #COOKs Distance
 train$cook <- cooks.distance(model1)
 sum(train$cook > 1)

 train$leverage <- hatvalues(model1)
 #check if any values are above 0.0009
 sum(train$leverage > 0.0009)

 #VIF to identify if there is a potential problem with multicolinearity
 vif(model1)

 #::MODEL 2 ASSUMPTIONS::
 #Add the predicted probabilities to the data frame
 train$predictedProbabilities <- fitted(model2)

 #This shows the probability of churn, and the actual outcome.
 head(data.frame(train$predictedProbabilities, train$y))

 #Add the standardised and Studentised residuals can be added to the data frame
 train$standardisedResiduals <- rstandard(model2)
 train$studentisedResiduals <- rstudent(model2)

 #count the residuals above 1.96
 sum(train$standardisedResiduals > 1.96)

 #COOKs Distance
 train$cook <- cooks.distance(model2)
 sum(train$cook > 1)

 train$leverage <- hatvalues(model2)
```

```
#check if any values are above 0.0009
sum(train$leverage > 0.0009)

#VIF to identify if there is a potential problem with multicolinearity
vif(model2)

#::MODEL 3 ASSUMPTIONS::
#Add the predicted probabilities to the data frame
train$predictedProbabilities <- fitted(model3)

#This shows the probability of churn, and the actual outcome.
head(data.frame(train$predictedProbabilities, train$y))

#Add the standardised and Studentised residuals can be added to the data frame
train$standardisedResiduals <- rstandard(model3)
train$studentisedResiduals <- rstudent(model3)

#count the residuals above 1.96
sum(train$standardisedResiduals > 1.96)

#COOKs Distance
train$cook <- cooks.distance(model3)
sum(train$cook > 1)

train$leverage <- hatvalues(model3)
#check if any values are above 0.0009
sum(train$leverage > 0.0009)

#VIF to identify if there is a potential problem with multicolinearity
vif(model3)
```

8. *Appendix 2: R/tables Screenshot*

Credit Default	Call Duration (MEAN)
YES	103.33
NO	259.84
UNKNOWN	252.44

Table 13. Shows Credit Default with respect to Marketing Campaign duration

Subscription?	Call Duration	
	Mean	Median
YES	553.2	449
NO	220.8	163.5

Table 14. Subscription of Customers with respect to Marketing Campaign duration

Education	Count
Basic	12513
High School	9515
Proffesional course	5243
University Degree	12168
illiterate	18
unknown	1731

Table 15. Education background of customers

```
>   postResample(class_pred1,test$y)
 Accuracy     Kappa 
0.8963504 0.2325779 
```

Fig. 27 Show Accuracy and kappa for Model 1

```
                        2.5 %      97.5 %
(Intercept)         0.1735011   0.2105728
defaultunknown      0.4184296   0.5290846
defaultyes                 NA 161.1995339
contacttelephone    0.3805754   0.4587755
poutcomenonexistent 0.6807519   0.8427476
poutcomesuccess     9.4397930  12.9537417
```

Fig 8. Shows Confidence Interval for Model 1

```
>   postResample(class_pred2,test$y)
 Accuracy     Kappa 
0.9110706 0.4533630 
```

Fig. 28 Show Accuracy and kappa for Model 3

```
                         2.5 %        97.5 %
(Intercept)          0.01787372    0.02454553
defaultunknown       0.55376889    0.73557287
defaultyes                   NA 1390.27940630
contacttelephone     0.75989549    0.98743498
poutcomenonexistent 1.09423844    1.41678308
poutcomesuccess      7.32232427   10.49661639
month.L              0.61384571    1.00684366
month.Q              1.33209290    2.23737033
month.C              0.24991162    0.40428426
month^4              2.72572590    4.00829558
month^5              0.84039133    1.20019355
month^6              1.24162456    1.72610770
month^7              1.59652115    2.14830960
month^8              0.64693149    0.90231835
month^9              0.98594156    1.30847311
duration             1.00437202    1.00468805
emp.var.rate         0.51692262    0.55504766
```

Fig. 12 Shows Confidence Interval for Model 2

```
>   postResample(class_pred3,test$y)
 Accuracy     Kappa
0.9125304 0.4654343
```

Fig 29. Show Accuracy and kappa for Model 3

Pseudo R^2 for MODEL 1	
Hosmer and Lemeshow	0.115
Cox and Snell	0.078
Nagelkerke	0.154

Table 16. Shows R square for model 1

Pseudo R^2 for MODEL 2	
Hosmer and Lemeshow	0.383
Cox and Snell	0.237
Nagelkerke	0.468

Table 17. Shows R square for model 2

Pseudo R^2 for MODEL 3	
Hosmer and Lemeshow	0.403
Cox and Snell	0.247
Nagelkerke	0.488

Table 18. Shows R square for model 3

9. References

Catalina, T. M., 2010. CONCEPT AND EVOLUTION OF BANK MARKETING. *Research Gate.*

Chapman, P. et al., 2000. *CRISP-DM 1.0: Step-by-step data mining guide.* [Online]
Available at: https://www.kde.cs.uni-kassel.de/wp-content/uploads/lehre/ws2012-13/kdd/files/CRISPWP-0800.pdf
[Accessed 5 January 2022].

Hubera, S., Wiemer, H., Schneider, D. & Ihlenfeldt, S., 2019. DMME: Data mining methodology for engineering applications – a holistic extension to the CRISP DM Model. *Elsevier B.V,* p. 403–408.

Moro, S., Cortez, P. & Rita, P., 2014. A data-driven approach to predict the success of bank telemarketing. *ELSEVIER,* pp. 22-31.

Moro, S. & Laureano, R. M. S., 2011. USING DATA MINING FOR BANK DIRECT MARKETING: AN APPLICATION OF THE CRISP-DM METHODOLOGY. *EUROSIS-ETI.*