Capstone Project - R

Project Report

Title

Data-Driven Approach to Predict Success of Telemarketing Campaign by Bank

Report submitted in part fulfilment of

Post Graduate Program in Data Analytics 2017

by

Mr. Milind Jeurkar

Mr. Grillo Jiu

Mr. Sujan Joe Jacob

Mr. Sandeep Singh

Project Guide: Mr. Arun Upadhyay

Imarticus Learning Pvt. Ltd.

Declaration

This report has been prepared on the basis of our group work. Where other published and unpublished source materials have been used, these have been acknowledged.

Students Name:

Mr. Milind Jeurkar

Mr. Grillo Jiu

Mr. Sujan Joe Jacob

Mr. Sandeep Singh

Date of Submission: 21.12.2017

Table of Contents

Figures

Abstract

The European bank wants to know potential customers who can secure term deposit with the bank. A set of information has been collected through tele-marketing campaign for the purpose.

We have proposed a predictive modelling approach to know the success rate of telemarketing campaign. A large data set named “bank.csv” of 41,188 observations with 20 features was analysed related to bank client, product and social-economic attributes.

Data preparation was done for making predictive data models & further analysis which can determine the main characteristics that affect success and selection of potential buying customers.

Four machine learning algorithms Logistic Regression, Decision Trees, Naïve Bayes, Support Vector Machine (SVM) were applied on the cleaned dataset. Decision Trees was chosen as the best model based on confusion matrix and accuracy scores. This model was used to predict outcome for the additional dataset provided. The most popular tool R was used to complete the project

It has been observed that increasingly vast number of marketing campaigns over a time has reduced its effect on the general public. Economic pressures and competition have led marketing managers to invest on directed campaigns with a strict and rigorous selection of contacts. Although telemarketing is a direct mode of communication with the prospective customer, this may make customers grumpy, contacts should be planned optimistically which will give better success rate.

Chapter 1: Introduction

1.1 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 need 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.

1.2 Data Description

Data Description:

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

Data source: 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.

1.3 Objective

Main objective of the project is to,

- Design most appropriate machine learning model to predict with high accuracy about the potential customers who can secure the term deposit.
- Identify the key differentiators between the ones who have subscribed (Yes) and who did not (No).
- Conclude if the telemarketing campaign was a success

Chapter 2: Plan of Action

Following steps were planned to complete the project.

2.1 Load data into R and install required packages

Loading the dataset into R environment and importing required packages/ libraries which will include caret, e1071, h2o, dplyr, readr, reshape2, ggplot, ggplot2, magrittr, pander.

2.2 Data Cleaning

Dataset was checked for discrepancies if any. Data cleaning to be done to make it ready for analysis which will involve

- **Missing values** - Missing values will be replaced with mean values for numerical variables and with mode values for categorical variables.
- **Duplicate observations** - Duplicate observations will be dropped if any.
- **Outliers** - Boxplot will be used to check outliers if any in the numerical variables. Outliers will be replaced with the values eq. to Q3 + 1.5 X (IQR)

2.3 Making data models and validating data models

Models - The dataset will be split to training set and test set in 80:20 proportion. Various machine learning algorithms will be run

Selection of variables - The data has 20 independent variables. Initially all the variables will be checked initially model will be made considering all variables. Subsequently Backward selection method will be used for finding the relevant variables, i.e. variables will be dropped one by one based on p values.

Selection of best Model – Best model will be selected based on Confusion Matrix and Accuracy scores. Final model will be run on additional dataset to predict the term deposit subscription, and conclude if the telemarketing campaign was a success or not. Variables in the final model will be the key differentiators between the ones who have subscribed term deposit (Yes) and who did not (No)

2.4 Data Visualisation/ Exploratory Analysis

Data will be visualised through various graphical features in R so as to get feel of the data and association of all variables

Chapter 3: Data Preparation

3.1 Data Description

"Bank.csv" is a raw file in csv format, and contain 41,188 observations with 20 variables as below.

Variables

Bank Client Data

1. age (numeric)
2. job : type of job (categorical: 'admin.', 'blue-collar', 'entrepreneur', 'housemaid', 'management', 'retired', 'self-employed', 'services', 'student', 'technician', 'unemployed', 'unknown')
3. marital : marital status (categorical: 'divorced', 'married', 'single', 'unknown'; note: 'divorced' means divorced or widowed)
4. education (categorical: 'basic.4y', 'basic.6y', 'basic.9y', 'high.school', 'illiterate', 'professional.course', 'university.degree', 'unknown')
5. default: has credit in default? (categorical: 'no', 'yes', 'unknown')
6. housing: has housing loan? (categorical: 'no', 'yes', 'unknown')
7. loan: has personal loan? (categorical: 'no', 'yes', 'unknown')

Data related with the last contact of the current campaign:

8. contact: contact communication type (categorical: 'cellular', 'telephone')
9. month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')
10. day_of_week: last contact day of the week (categorical: 'mon','tue','wed','thu','fri')
11. duration: last contact duration, in seconds (numeric). 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.

Other attributes:

12. campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)
13. pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)
14. previous: number of contacts performed before this campaign and for this client (numeric)
15. poutcome: outcome of the previous marketing campaign (categorical: 'failure','nonexistent','success')

Social and economic context attributes

16. emp.var.rate: employment variation rate - quarterly indicator (numeric)
17. cons.price.idx: consumer price index - monthly indicator (numeric)
18. cons.conf.idx: consumer confidence index - monthly indicator (numeric)
19. euribor3m: euribor 3 month rate - daily indicator (numeric)
20. nr.employed: number of employees - quarterly indicator (numeric)

3.2 Install packages and call libraries in R

The data file “Bank.csv” was imported in working directory. Required packages were installed and libraries were called for processing/ analysis.

3.2.1 Code to load the dataset and view structure, dimensions, column names

```
setwd("C:/Users/Admin/Desktop/Assignment 1 - Plan of Action")
bank <- read.csv("bank.csv")
colnames(bank)
dim(bank)
str(bank)
```

3.2.2 Code for calling libraries

```
library(caret)
library(caTools)
library(e1071)
library(MLmetrics)
library(ggplot2)
library(rpart)
library(randomForest)
```

3.3 Data Summary

Summary was checked to get a feel of all variables and possible values. This was also useful for identifying missing values.

Code for dataset summary and column wise tables

```
Summary(bank)
table(bank$job)
table(bank$marital)
table(bank$education)
table(bank$default)
table(bank$housing)
table(bank$loan)
```

The dataset has missing values as below.

Variable name	**-**	**Total no. of missing values**
Job	**-**	330 no.
marital	-	80 no.
education	-	1731 no.
default	-	8597 no.
housing	-	990 no.
loan	-	990 no.

Figure 3-1: Summary of dataset

```
> summary(bank)
      age                job            marital                    education
 Min.   :17.00   admin.     :10422          :   80   university.degree  :12168
 1st Qu.:32.00   blue-collar: 9254   divorced: 4612   high.school       : 9515
 Median :38.00   technician : 6743   married :24928   basic.9y          : 6045
 Mean   :40.02   services   : 3969   single  :11568   professional.course: 5243
 3rd Qu.:47.00   management : 2924                    basic.4y          : 4176
 Max.   :98.00   retired    : 1720                    basic.6y          : 2292
                 (Other)    : 6156                    (Other)           : 1749
 default     housing      loan           contact          month       day_of_week
    : 8597       :  990       :  990   cellular :26144   may    :13769   fri:7827
 no :32588   no :18622   no :33950   telephone:15044   jul    : 7174   mon:8514
 yes:    3   yes:21576   yes: 6248                     aug    : 6178   thu:8623
                                                       jun    : 5318   tue:8090
                                                       nov    : 4101   wed:8134
                                                       apr    : 2632
                                                       (Other): 2016
    duration         campaign          pdays          previous            poutcome
 Min.   :   0.0   Min.   : 1.000   Min.   :  0.0   Min.   :0.000   failure    : 4252
 1st Qu.: 102.0   1st Qu.: 1.000   1st Qu.:999.0   1st Qu.:0.000   nonexistent:35563
 Median : 180.0   Median : 2.000   Median :999.0   Median :0.000   success    : 1373
 Mean   : 258.3   Mean   : 2.568   Mean   :962.5   Mean   :0.173
 3rd Qu.: 319.0   3rd Qu.: 3.000   3rd Qu.:999.0   3rd Qu.:0.000
 Max.   :4918.0   Max.   :56.000   Max.   :999.0   Max.   :7.000

  emp.var.rate      cons.price.idx  cons.conf.idx     euribor3m      nr.employed
 Min.   :-3.40000   Min.   :92.20   Min.   :-50.8   Min.   :0.634   Min.   :4964
 1st Qu.:-1.80000   1st Qu.:93.08   1st Qu.:-42.7   1st Qu.:1.344   1st Qu.:5099
 Median : 1.10000   Median :93.75   Median :-41.8   Median :4.857   Median :5191
 Mean   : 0.08189   Mean   :93.58   Mean   :-40.5   Mean   :3.621   Mean   :5167
 3rd Qu.: 1.40000   3rd Qu.:93.99   3rd Qu.:-36.4   3rd Qu.:4.961   3rd Qu.:5228
 Max.   : 1.40000   Max.   :94.77   Max.   :-26.9   Max.   :5.045   Max.   :5228

   y
 no :36548
 yes: 4640
```

3.4 Data Cleaning

3.4.1 Handling Missing Values

Missing values are appearing only in categorical variables “Job”, “marital”, “education”, “default”, “housing”, and “loan”. These are replaced with the mode values as per the standard practise.

Code for replacing missing values with user defined value

```
levels(bank$job)[levels(bank$job)==""]="admin."
levels(bank$marital)[levels(bank$marital)==""]="married"
levels(bank$education)[levels(bank$education)==""]="university.degree"
levels(bank$default)[levels(bank$default)==""]="no"
levels(bank$housing)[levels(bank$housing)==""]="yes"
levels(bank$loan)[levels(bank$loan)==""]="no"
```

3.4.2 Dropping of Duplicate observations

12 no. of duplicate observations in the dataset were dropped with the following code.

Code for dropping duplicate observations

```
bank=unique(bank)
```

3.4.3 Levelling

Applied level function to outcome variable "y" making defining two levels “0” and “1”

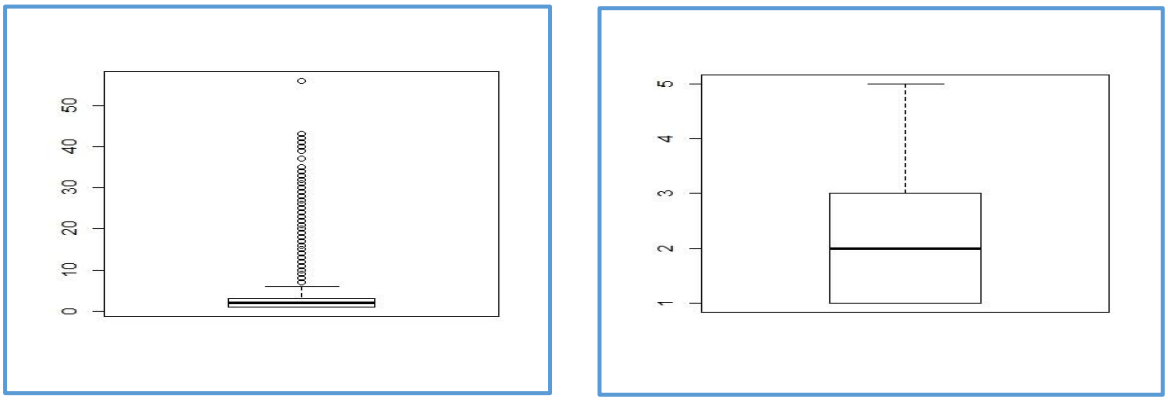
Code for Levelling

```
levels(bank$y)[levels(bank$y)=="yes"]="1"
levels(bank$y)[levels(bank$y)=="no"]="0"
```

3.4.4 Handling Outliers

Boxplots were check to check the outliers in all numerical variables. Outliers in the age variable value was capped to 0 and 80. Outliers (1.5 times of IQR beyond Q1 & Q3) in all other variables were replaced with the mean value.

Figure 3-2: Sample Boxplots of Campaign variable before and after handlingoutliers



Sample Code for handling outliers

```
str(bank)
summary(bank$age)
bank$age[bank$age<18]=18
bank$age[bank$age>80]=80

summary(bank$campaign)
#> summary(bank$campaign)
#Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
#1.000   1.000   2.000   2.568   3.000  56.000

boxplot(bank$campaign)
x=3+1.5*(3-1) #Q1 = 1, Q3=3
x

bank$campaign[bank$campaign>=x]=round(mean(bank$campaign))
boxplot(bank$campaign)
```

Chapter 4: Data Analysis

4.1 Split data into training set and testing set

The data was split randomly into training set and testing set with a ratio of 75%:25% respectively.

Code for splitting the data

```
set.seed(123)
split = sample.split(bank$y, SplitRatio = 0.75)
training_set = subset(bank, split == TRUE)
test_set = subset(bank, split == FALSE)
```

4.2 Building Models

Following machine learning algorithms were used to build various data models on the training set. All the data models were tested on testing set. Confusion matrix was prepared in each case tabulating the predicted values and actual values of target variables.

4.2.1 Model 1 – Logistic Regression

Target variable has binary output values. Using "Logistic Regression" package Logistic Regression model was applied on the dataset.

Code for Logistic Regression and Confusion Matrix

```
log_reg_model=glm(y~.,data = training_set,family = "binomial")
y_pred=predict(log_reg_model,test_set,type = "response")
pred0 = ifelse(y_pred > 0.5, 1, 0)
ConfusionMatrix(pred0,test_set$y)
table(pred0,test_set$y)
Accuracy(pred0,test_set$y)
```

Confusion Matrix - Logistic Regression

Y Predicted		Y True		Model Accuracy
		0	1	90.84%
	0	8984	150	
	1	793	367	

Figure 4-1: Confusion Matrix – Logistic Regression

4.2.2 Model 2 – Decision Tree Algorithm

Using "RPart" package, Decision Tree algorithm is applied and classification tree is fitted.

Code for Decision Tree Algorithm

```
dt_model=rpart(y~.,data=training_set,method = 'class')
pred_dt=predict(dt_model,test_set,type = "class")
ConfusionMatrix(pred_dt,test_set$y)
Accuracy(pred1,test_set$y)
```

Confusion Matrix – Decision Tree

	Y True		Model Accuracy
Y Predicted	0	1	
0	8968	166	91.1%
1	746	414	

Figure 4-2: Confusion Matrix – Decision Tree

4.2.3 Model 3 - Naive Bayes Algorithm

By using the "NaïveBayes" package, conditional posterior probabilities of a categorical class variable are computed given independent predictor variables.

Code for Naïve Bayes Algorithm

```
nb<-naiveBayes(y~contact+month+day_of_week+poutcome+cons.price.idx,data = training_set)
pred2=predict(nb,test_set,type = "class")
ConfusionMatrix(pred2,test_set$y)
Accuracy(pred2,test_set$y)
```

Confusion Matrix – Naïve Bayes

	Y True		Model Accuracy
Y Predicted	0	1	
0	8858	276	89.2%
1	834	326	

Figure 4-3: Confusion Matrix – Naïve Bayes

4.2.4 Model 4 – Support Vector Machine Algorithm

A package e1071 has been used for SVM models. Models with kernel type linear, radial, polynomial, sigmoid are checked for accuracy, the best SVM model is given below.

Code for SVM Algorithm

```
svm_model = svm(formula = y ~., data = training_set, kernel = 'polynomial')
pred3=predict(svm_model,test_set,type = "class")
ConfusionMatrix(pred3,test_set$y)
Accuracy(pred3,test_set$y)
```

Confusion Matrix – SVM

Y Predicted	Y True		Model Accuracy
	0	1	90.7%
0	9051	83	
1	873	287	

Figure 4-4: Confusion Matrix – SVM

4.2.5 Best Model

Best model was selected based on the accuracy score of all the models.

Sr. No.	Data Model	Accuracy Score
1	Logistic Regression	90.8%
2	Decision Tree	91.1%
3	Naïve Bayes	89.2%
4	SVM	90.7%

Figure 4-5: Comparison of Models

4.2.6 Predictions on additional dataset using the best model

Model…… has been selected as the best model based on confusion matrix and accuracy score of all the models. Validation was done for this model using the additional dataset provided which has 3090 no. of observations.

Code for predictions on additional dataset

```
add_bank<-read.csv('bank-additional.csv')
pred_dt=predict(dt_model,add_bank,type = "class")
summary(pred_dt)
```

Prediction Output on the additional dataset

0 (No)	1 (Yes)	Total
2875	215	3090

Figure 4-6: Prediction on additional dataset

4.3 Data Visualisation - Exploratory Analysis

Various plots were developed for exploratory data analysis.

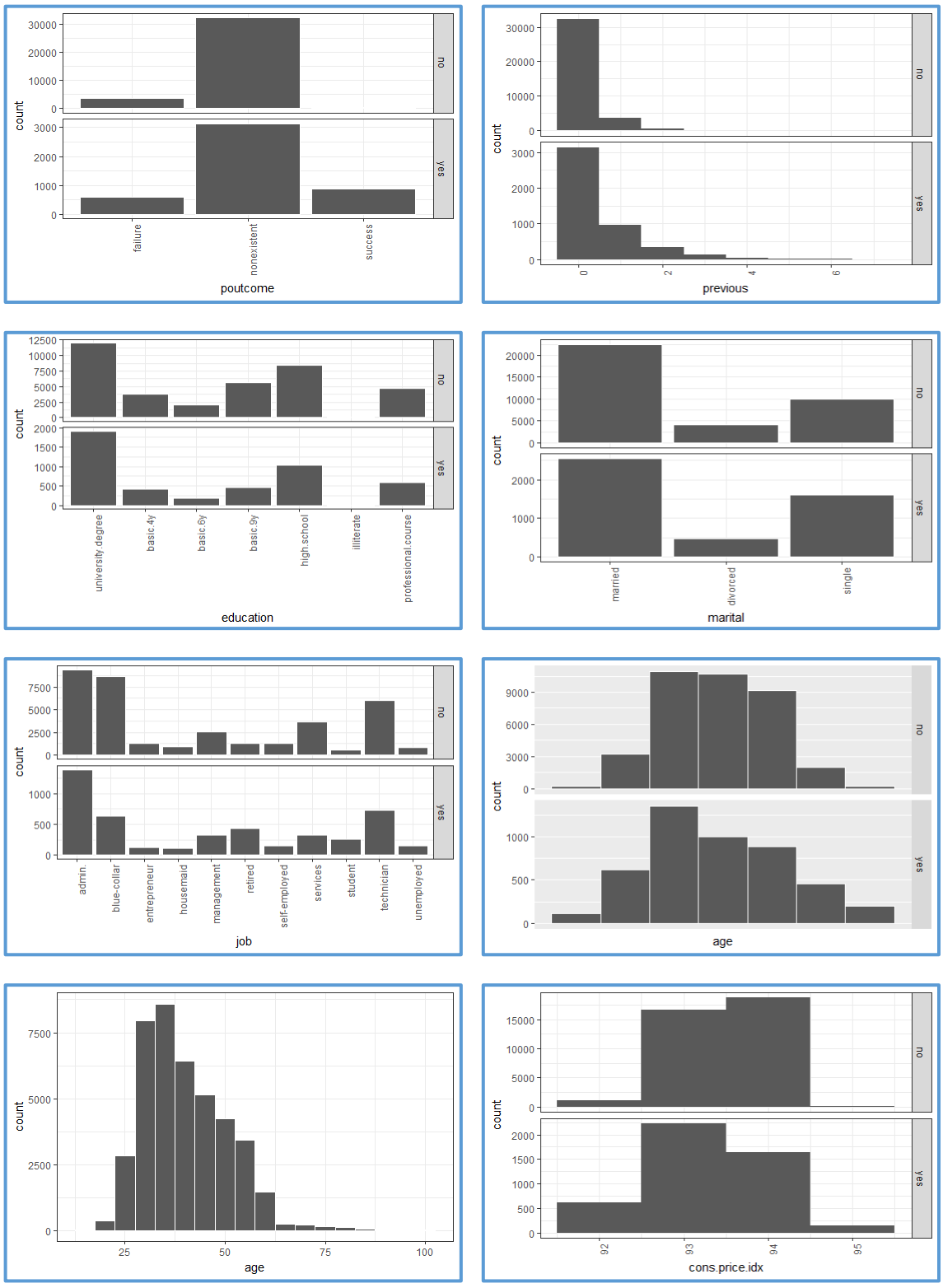


Figure 4-7: Plots developed during exploratory analysis

Code for developing the plots

```
#age vs y

ggplot(bank) + geom_histogram(aes(x = age), binwidth = 0.1, col = "white") +  facet_grid(y~., scales = "free") + scale_x_log10() + theme()

#job vs y

ggplot(bank) + geom_bar(aes(x = job), col = "white") + facet_grid(y~., scales = "free")  + theme_bw() + theme(axis.text.x = element_text(angle = 90, hjust = 1))

#marital vs y

ggplot(bank) + geom_bar(aes(x = marital), col = "white") + facet_grid(y~., scales = "free")  + theme_bw()+ theme(axis.text.x = element_text(angle = 90, hjust = 1))

#education vs y

ggplot(bank) + geom_bar(aes(x = education), col = "white") + facet_grid(y~., scales = "free")+theme_bw() + theme(axis.text.x = element_text(angle = 90, hjust = 1))

#previous

ggplot(bank) + geom_histogram(aes(x = previous), binwidth = 1) + facet_grid(y~., scales = "free")  + theme_bw() + theme(axis.text.x = element_text(angle = 90, hjust = 1))

#poutcomes

ggplot(bank) + geom_bar(aes(x = poutcome), col = "white")+ facet_grid(y~., scales = "free") + theme_bw() + theme(axis.text.x = element_text(angle = 90, hjust = 1))

#Cons.price.idx

ggplot(bank) + geom_histogram(aes(x = cons.price.idx), binwidth = 1) + facet_grid(y~., scales = "free")  + theme_bw() + theme(axis.text.x = element_text(angle = 90, hjust = 1))


# age count

ggplot(bank, aes(x = age)) + geom_histogram(binwidth = 5, col = "white") + theme_bw()
```

4.4 Findings

4.4.1 Best Model

Decision Tree has been identified as the best model for the given problem which gives highest accuracy as compared to other models.

4.4.2 Inference

It can be analysed that following variables are the most relevant inputs in predicting the success rate of bank direct marketing campaign.

- Duration - call duration
- Month - month of contact
- Age - customer age
- Contact - cellular/ Telephone
- Credit default
- Job

4.4.3 Conclusion

- Marketing Campaign is more likely to be successful during March, September, December (end of every trimester).
- Customers are more likely to subscribe term deposit if the conversation duration is relatively more.
- Customers with default credit would not go for term deposit subscription.
- Campaign reach is good among blue collar, admin, retired and housemaids.

Bibliography

[1] S. Moro, P. Cortez, P. Rita,"A Data-Driven Approach to Predict the Success of Bank Telemarketing", Decision Support Systems, Elsevier, 62, pp. 22-31, June 2014

[2] S. Moro, R. Laureano, P. Cortez "Using Data Mining for Bank Direct Marketing: An Application of the CRISPDM Methodology", In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference