Kranthi case 1

2022-07-05

Introduction

Executive Summary

- I started to understand the dataset by importing data and this data set involves mmany visualizations and includes modeling. * # Introduction

TAYKO SOFTWARE

#The data file Tayko.csv consist of 25 columns, with id as sequence number, and we consider 24 variables to predict the output.

Business Problem:

Predicting Software Reselling Profits

Background: Tayko is a software catalog firm that sells games and educational software. It started out as a software manufacturer and later added third-party titles to its offerings. It has recently put together a revised collection of items in a new catalog, which it is preparing to roll out in a mailing.

In addition to its own software titles, Tayko's customer list is a key asset. In an attempt to expand its customer base, it has recently joined a consortium of catalog firms that specialize in computer and software products. The consortium affords members the opportunity to mail catalogs to names drawn from a pooled list of customers. Members supply their own customer lists to the pool, and can "withdraw" an equivalent number of names each quarter. Members are allowed to do predictive modeling on the records in the pool so they can do a better job of selecting names from the pool.

Further, Tayko has supplied its customer list of 200,000 names to the pool, which totals over 5,000,000 names, so it is now entitled to draw 200,000 names for a mailing. Tayko would like to select the names that have the best chance of performing well, so it conducts a test—it draws 20,000 names from the pool and does a test mailing of the new catalog.

OBJECTIVE: From the dataset Tayko.csv, Purchase output variable is considered for the analysis and prediction. The objective of the model is to classify records into 'PURCHASE' or "NO PURCHASE'.

STAGE 1:

Improting the required packages

```
#LOADING AND EXPLORING DATA
#Loading required libraries.
library(knitr)
```

```
## Warning: package 'knitr' was built under R version 4.1.3
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.3
```

```
library(plyr)
```

```
## Warning: package 'plyr' was built under R version 4.1.3
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.1.3
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.1.3

## corrplot 0.92 loaded
```

```
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 4.1.3

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(scales)
```

```
## Warning: package 'scales' was built under R version 4.1.3
```

```
library(ggrepel)
```

```
## Warning: package 'ggrepel' was built under R version 4.1.3
```

```
#Below, I am reading the Tayko.csv's as dataframes into R.
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.1.3

##
## Attaching package: 'readr'

## The following object is masked from 'package:scales':
##
##     col_factor
```

```r
tayko <- read_csv("Tayko.csv")
```

```
## Rows: 2000 Columns: 25

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (25): sequence_number, US, source_a, source_c, source_b, source_d, sourc...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Data size and structure

```r
dim(tayko)
```

```
## [1] 2000   25
```

```r
str(tayko[,c(1:10, 25)]) #display first 10 variables and the response variable
```

```
## tibble [2,000 x 11] (S3: tbl_df/tbl/data.frame)
##  $ sequence_number: num [1:2000] 1 2 3 4 5 6 7 8 9 10 ...
##  $ US             : num [1:2000] 1 1 1 1 1 1 1 1 1 1 ...
##  $ source_a       : num [1:2000] 0 0 0 0 0 0 0 0 1 1 ...
##  $ source_c       : num [1:2000] 0 0 0 1 1 0 0 0 0 0 ...
##  $ source_b       : num [1:2000] 1 0 0 0 0 0 0 1 0 0 ...
##  $ source_d       : num [1:2000] 0 0 0 0 0 0 0 0 0 0 ...
##  $ source_e       : num [1:2000] 0 1 0 0 0 0 0 0 0 0 ...
##  $ source_m       : num [1:2000] 0 0 0 0 0 0 0 0 0 0 ...
##  $ source_o       : num [1:2000] 0 0 0 0 0 0 0 0 0 0 ...
##  $ source_h       : num [1:2000] 0 0 0 0 0 0 0 0 0 0 ...
##  $ Spending       : num [1:2000] 128 0 127 0 0 0 0 0 489 174 ...
```

Data cleaning

```r
# get column names
colnames(tayko)
```

```
##  [1] "sequence_number"      "US"                   "source_a"
##  [4] "source_c"             "source_b"             "source_d"
##  [7] "source_e"             "source_m"             "source_o"
## [10] "source_h"             "source_r"             "source_s"
## [13] "source_t"             "source_u"             "source_p"
## [16] "source_x"             "source_w"             "Freq"
## [19] "last_update_days_ago" "1st_update_days_ago"  "Web order"
## [22] "Gender=male"          "Address_is_res"       "Purchase"
## [25] "Spending"
```

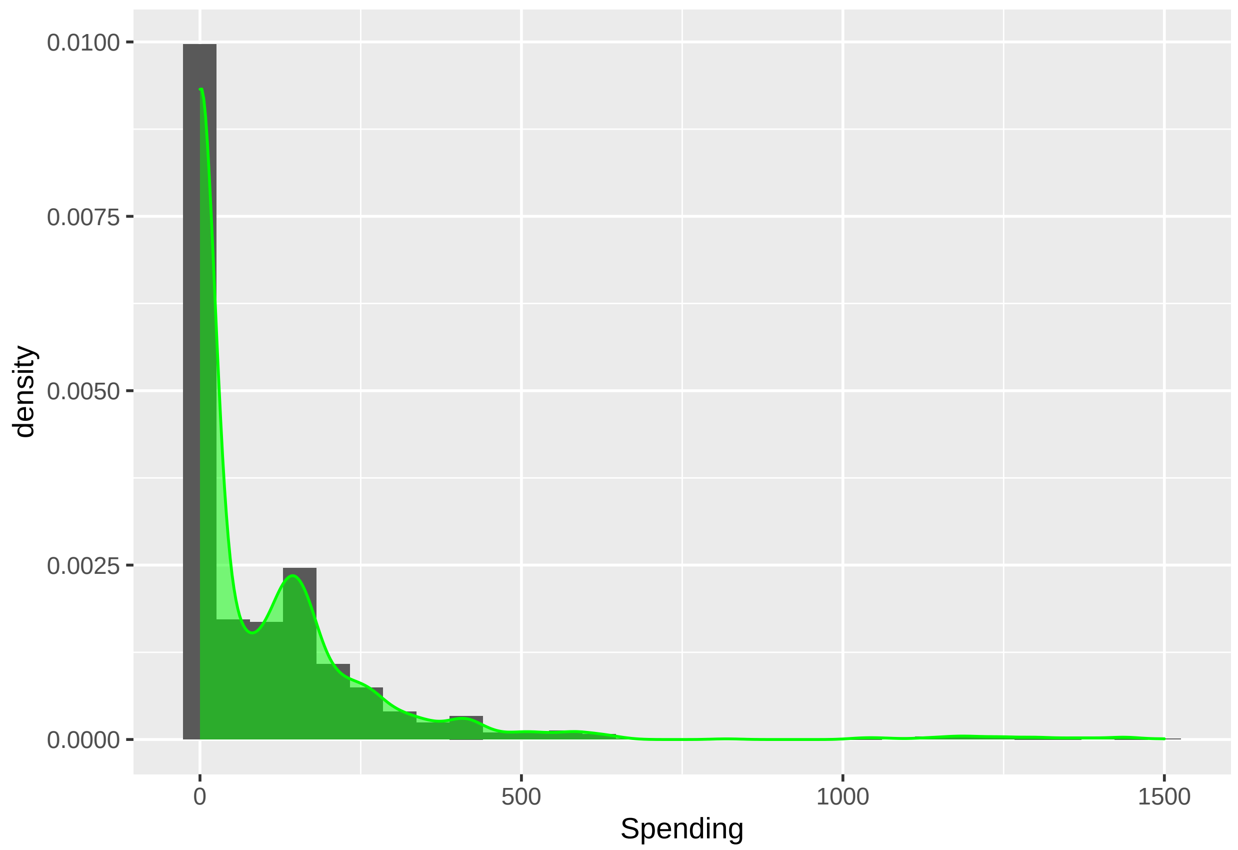
```r
names(tayko)[21] <- "Web.order"
names(tayko)[22] <- "Gender"


# get column names
colnames(tayko)
```

```
##  [1] "sequence_number"      "US"                   "source_a"
##  [4] "source_c"             "source_b"             "source_d"
##  [7] "source_e"             "source_m"             "source_o"
## [10] "source_h"             "source_r"             "source_s"
## [13] "source_t"             "source_u"             "source_p"
## [16] "source_x"             "source_w"             "Freq"
## [19] "last_update_days_ago" "1st_update_days_ago"  "Web.order"
## [22] "Gender"               "Address_is_res"       "Purchase"
## [25] "Spending"
```

```
ggplot(tayko,aes(x = Spending))+
      geom_histogram(aes(y=..density..))+
      geom_density(color="Green", fill="Green", alpha=0.5)
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```



from the above histogram we can say that lower the spending higher the density, It means peolpe with lower spending are very high compared to people with higher spending

```
summary(tayko)
```

```
##  sequence_number        US            source_a         source_c
##  Min.   :   1.0   Min.   :0.0000   Min.   :0.0000   Min.   :0.000
##  1st Qu.: 500.8   1st Qu.:1.0000   1st Qu.:0.0000   1st Qu.:0.000
##  Median :1000.5   Median :1.0000   Median :0.0000   Median :0.000
##  Mean   :1000.5   Mean   :0.8245   Mean   :0.1265   Mean   :0.056
##  3rd Qu.:1500.2   3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:0.000
##  Max.   :2000.0   Max.   :1.0000   Max.   :1.0000   Max.   :1.000
##     source_b       source_d         source_e        source_m
##  Min.   :0.00   Min.   :0.0000   Min.   :0.000   Min.   :0.0000
##  1st Qu.:0.00   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:0.0000
##  Median :0.00   Median :0.0000   Median :0.000   Median :0.0000
##  Mean   :0.06   Mean   :0.0415   Mean   :0.151   Mean   :0.0165
##  3rd Qu.:0.00   3rd Qu.:0.0000   3rd Qu.:0.000   3rd Qu.:0.0000
##  Max.   :1.00   Max.   :1.0000   Max.   :1.000   Max.   :1.0000
##     source_o         source_h         source_r         source_s
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000
##  Median :0.0000   Median :0.0000   Median :0.0000   Median :0.000
##  Mean   :0.0335   Mean   :0.0525   Mean   :0.0685   Mean   :0.047
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.000
##     source_t         source_u        source_p        source_x
##  Min.   :0.0000   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:0.000   1st Qu.:0.000
##  Median :0.0000   Median :0.000   Median :0.000   Median :0.000
##  Mean   :0.0215   Mean   :0.119   Mean   :0.006   Mean   :0.018
##  3rd Qu.:0.0000   3rd Qu.:0.000   3rd Qu.:0.000   3rd Qu.:0.000
##  Max.   :1.0000   Max.   :1.000   Max.   :1.000   Max.   :1.000
##     source_w           Freq        last_update_days_ago 1st_update_days_ago
##  Min.   :0.0000   Min.   : 0.000   Min.   :   1         Min.   :   1
##  1st Qu.:0.0000   1st Qu.: 1.000   1st Qu.:1133         1st Qu.:1671
##  Median :0.0000   Median : 1.000   Median :2280         Median :2721
##  Mean   :0.1375   Mean   : 1.417   Mean   :2155         Mean   :2436
##  3rd Qu.:0.0000   3rd Qu.: 2.000   3rd Qu.:3139         3rd Qu.:3353
##  Max.   :1.0000   Max.   :15.000   Max.   :4188         Max.   :4188
##    Web.order         Gender       Address_is_res     Purchase
##  Min.   :0.000   Min.   :0.0000   Min.   :0.000   Min.   :0.0
##  1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:0.0
##  Median :0.000   Median :1.0000   Median :0.000   Median :0.5
##  Mean   :0.426   Mean   :0.5245   Mean   :0.221   Mean   :0.5
##  3rd Qu.:1.000   3rd Qu.:1.0000   3rd Qu.:0.000   3rd Qu.:1.0
##  Max.   :1.000   Max.   :1.0000   Max.   :1.000   Max.   :1.0
##     Spending
##  Min.   :   0.0
##  1st Qu.:   0.0
##  Median :   2.0
##  Mean   : 102.6
##  3rd Qu.: 153.0
##  Max.   :1500.0
```

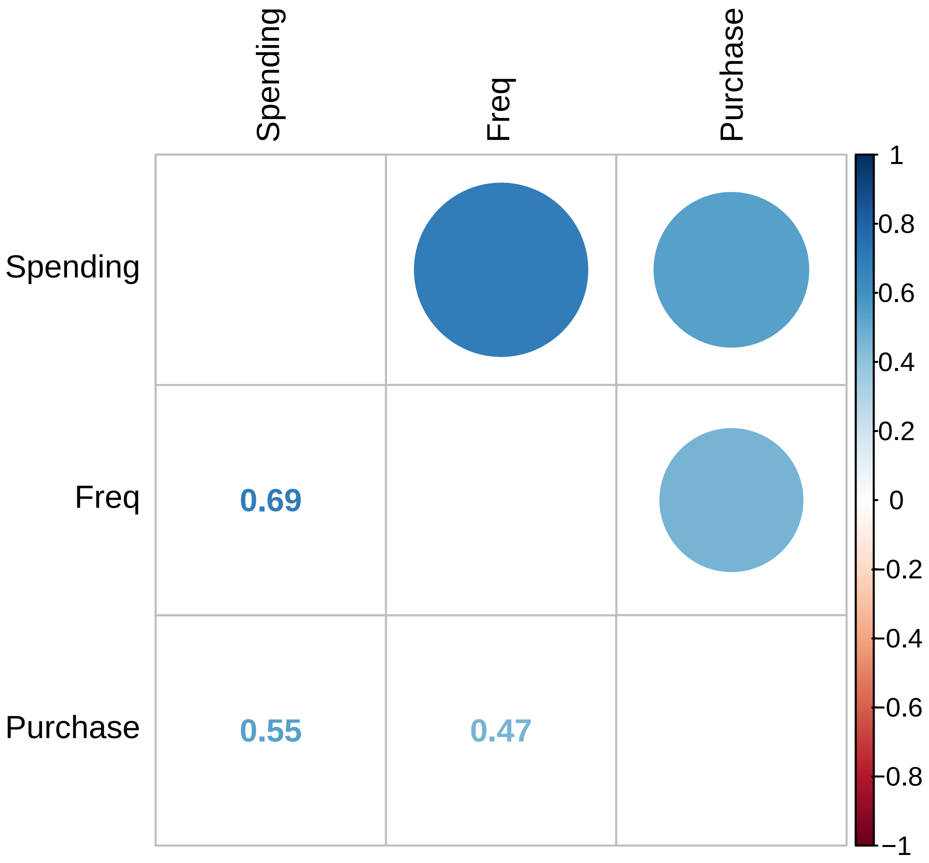
```
numericVars <- which(sapply(tayko, is.numeric)) #index vector numeric variables
numericVarNames <- names(numericVars) #saving names vector for use later on
cat('There are', length(numericVars), 'numeric variables')
```

```
## There are 25 numeric variables
```

```r
tayko_numVar <- tayko[, numericVars]
cor_numVar <- cor(tayko_numVar, use="pairwise.complete.obs") #correlations of all numeric variables
```

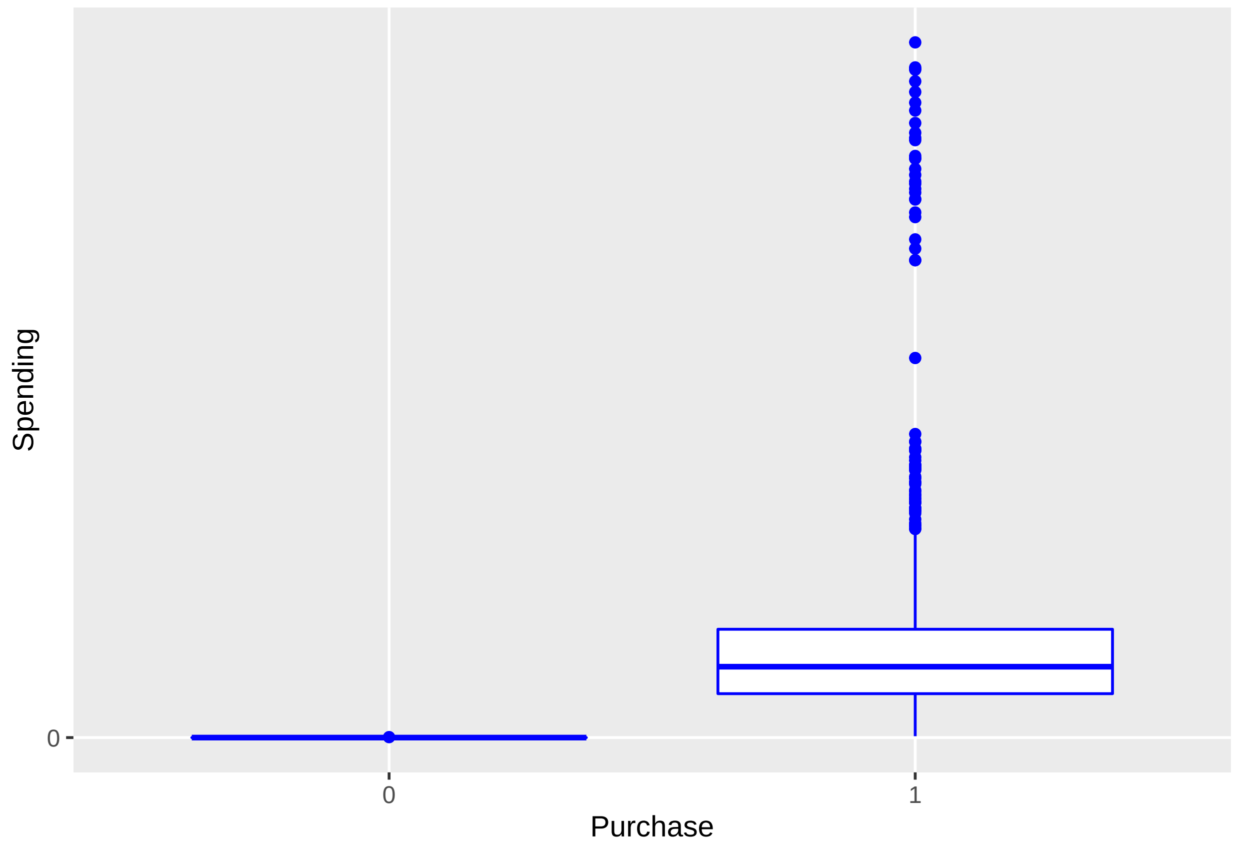
```r
#sort on decreasing correlations with SalePrice
cor_sorted <- as.matrix(sort(cor_numVar[,'Spending'], decreasing = TRUE))
 #select only high corelations
CorHigh <- names(which(apply(cor_sorted, 1, function(x) abs(x)>0.5)))
cor_numVar <- cor_numVar[CorHigh, CorHigh]

corrplot.mixed(cor_numVar, tl.col="black", tl.pos = "lt")
```



the highest correlation is for freq-spending pair when compared to other pairs

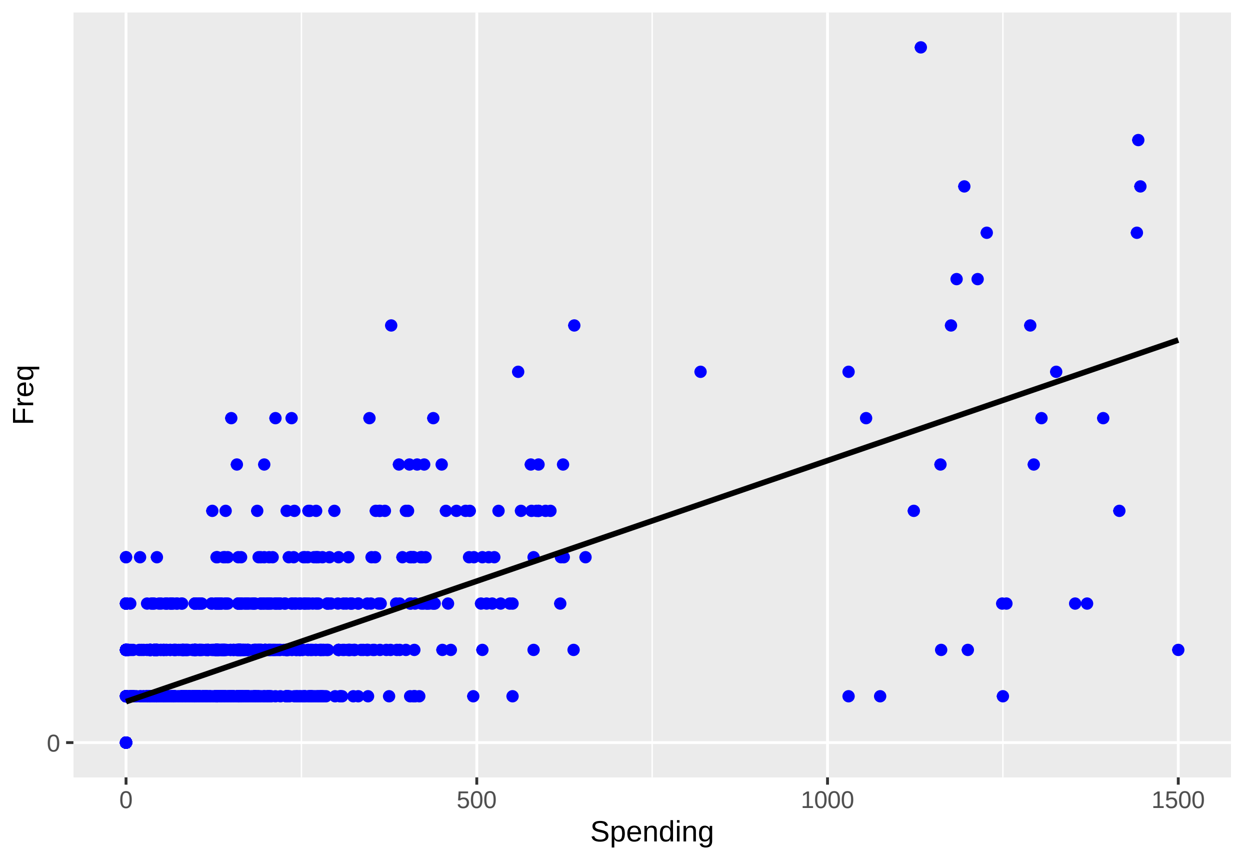
```r
ggplot(data=tayko[!is.na(tayko$Spending),], aes(x=factor(Purchase), y=Spending))+
        geom_boxplot(col='blue') + labs(x='Purchase') +
        scale_y_continuous(breaks= seq(0, 80, by=1000), labels = comma)
```



Based on the the above boxplot we can say that if there is no purchase there is no spending and there is slight increase in spending when purchase is at 1

```
ggplot(data=tayko[!is.na(tayko$Spending),], aes(x=Spending, y=Freq))+
        geom_point(col='blue') + geom_smooth(method = "lm", se=FALSE, color="black", aes(group=1)) +
        scale_y_continuous(breaks= seq(0, 800, by=10000), labels = comma) +
        geom_text_repel(aes(label = ifelse(tayko$Freq[!is.na(tayko$Spending)]>4500, rownames(all), ''))
```

```
## `geom_smooth()` using formula 'y ~ x'
```



when spending is below 500 the frequency of people is more compared to people spending more than 500. it seems that there are many people in the category of spending below 500.

STAGE 2:Data Mining Techniques(Methodology)

We have been instructed to use three data mining techniques to implement our predictive models. The 3 selected techniques were: Multiple regression analysis, Logistic regression and Regression tree.

Logistic Regression - we have implemeneted this technique to help in estimating the probability of an individulas to purchase or not to purchase based on our given Tayko dataset of independent variables. The dependent variable in our case is Purchase variable and is bounded between 0 and 1.

Regression tree - Is a technique that identifies what combination of our dataset factors best differentiates between individuals(who purchases/not purchases) based on our categorical variable of interest which is (Purchase variable)

Multiple regression analysis - Is a technique that have been used to analyze the relationship between a single dependent variable (which is Purchase variable) and several independent variables(the predictor variables). The objective of multiple regression analysis is to use the independent variables whose values are known to predict the value of the single dependent value.

STAGE 3: Implemetation of the data mining techniques

Multiple regression analysis

Below is the summary of Multiple regression analysis technique showing its coefficients after the training has taken place using training dataset and ready for testing using the testing dataset

```
# we partition tayko dataset
set.seed(1234)
## partitioning into training (60%) and validation (40%)
train.rows <- sample(rownames(tayko), dim(tayko)[1]*0.6)
valid.rows <- sample(setdiff(rownames(tayko), train.rows),dim(tayko)[1]*0.2)
# assign the remaining 20% row IDs serve as test
test.rows <- setdiff(rownames(tayko), union(train.rows, valid.rows))
# create the 3 data frames by collecting all columns from the appropriate rows
train.data <- tayko[train.rows, ]
valid.data <- tayko[valid.rows, ]
test.data <- tayko[test.rows, ]

# use lm() to run a linear regression of Price on all 11 predictors in the
# training set.
# use . after ~ to include all the remaining columns in train.df as predictors.
tayko.lm <- lm(Purchase ~ ., data = train.data)

summary(tayko.lm)
```

```
##
```

```
## Call:
## lm(formula = Purchase ~ ., data = train.data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.64286 -0.30559 -0.03657  0.34035  0.87145
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            4.090e-02  7.392e-02   0.553 0.580134
## sequence_number       -2.432e-05  1.955e-05  -1.244 0.213734
## US                     9.289e-03  3.059e-02   0.304 0.761445
## source_a               3.901e-01  6.439e-02   6.059 1.84e-09 ***
## source_c               1.154e-01  7.367e-02   1.567 0.117481
## source_b               1.332e-01  7.207e-02   1.848 0.064900 .
## source_d               2.735e-01  7.999e-02   3.419 0.000651 ***
## source_e               2.230e-01  6.078e-02   3.670 0.000254 ***
## source_m               2.558e-01  1.087e-01   2.354 0.018719 *
## source_o               1.180e-01  8.163e-02   1.446 0.148490
## source_h              -1.660e-01  7.969e-02  -2.083 0.037442 *
## source_r               2.310e-01  6.994e-02   3.303 0.000987 ***
## source_s               1.807e-01  7.537e-02   2.397 0.016675 *
## source_t               3.715e-01  9.002e-02   4.126 3.95e-05 ***
## source_u               3.902e-01  6.366e-02   6.129 1.20e-09 ***
## source_p               6.142e-01  1.565e-01   3.924 9.23e-05 ***
## source_x               1.853e-01  9.453e-02   1.960 0.050231 .
## source_w               2.618e-01  6.721e-02   3.895 0.000104 ***
## Freq                   7.640e-02  1.387e-02   5.508 4.46e-08 ***
## last_update_days_ago   2.137e-05  2.310e-05   0.925 0.354951
## `1st_update_days_ago` -3.223e-05  2.543e-05  -1.267 0.205374
## Web.order              1.675e-01  2.282e-02   7.341 3.95e-13 ***
## Gender                 8.568e-04  2.261e-02   0.038 0.969780
## Address_is_res         3.566e-02  3.093e-02   1.153 0.249193
## Spending               8.227e-04  8.326e-05   9.881  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3852 on 1175 degrees of freedom
## Multiple R-squared:  0.4189, Adjusted R-squared:  0.407
## F-statistic: 35.29 on 24 and 1175 DF,  p-value: < 2.2e-16
```

Below are the predicted values of our Purchase variable based on the predictor variables using our Multiple regression analysis technique

```
# use predict() with type = "response" to compute predicted probabilities.
tayko.lm.pred <- predict(tayko.lm, valid.data, type = "response")
# first 5 actual and predicted records
data.frame(actual = valid.data$Purchase[1:5], predicted = tayko.lm.pred[1:5])
```

```
##   actual predicted
## 1      1 0.7433109
## 2      0 0.2658812
## 3      1 0.6263909
## 4      1 0.6799541
## 5      0 0.3686488
```

```
table_mat <- table(valid.data$Purchase, tayko.lm.pred)

#get the predicted values of Purchase variable
table_mat
```

```
##    tayko.lm.pred
##     -0.035246346094479 -0.0294059956805337 -0.0286543627399991
##   0                  1                   1                   1
##   1                  0                   0                   0
##    tayko.lm.pred
##     -0.018124677573785 -0.0113624458983123 -0.00798186550756044
##   0                  1                   1                    1
##   1                  0                   0                    0
##    tayko.lm.pred
##     -0.00091984371957197 0.0013090477384515 0.00657339451169452
##   0                    1                  1                   1
##   1                    0                  0                   0
##    tayko.lm.pred
##     0.0117225022689344 0.0162688068003893 0.0234215280276279 0.0245168931436022
##   0                  1                  1                  1                  1
##   1                  0                  0                  0                  0
##    tayko.lm.pred
##     0.0266674953207375 0.0290062654323677 0.0383108571824448 0.0411877863379021
##   0                  1                  1                  1                  1
##   1                  0                  0                  0                  0
##    tayko.lm.pred
##     0.0494287693857105 0.0497068580575941 0.057293393303452 0.0748453328757193
##   0                  1                  1                 1                  1
##   1                  0                  0                 0                  0
##    tayko.lm.pred
##     0.0750675085404287 0.0896654584822533 0.0961441427033871 0.097311121196335
##   0                  1                  1                  1                 0
##   1                  0                  0                  0                 1
##    tayko.lm.pred
##     0.0990874446292981 0.108341725259213 0.110576654931697 0.114884892320795
##   0                  1                 1                 1                 1
##   1                  0                 0                 0                 0
##    tayko.lm.pred
##     0.115095608704947 0.116547264299529 0.117076561306171 0.117536873310626
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.120179378723007 0.120901550827185 0.122695592255638 0.12634208414495
##   0                 1                 1                 1                1
##   1                 0                 0                 0                0
##    tayko.lm.pred
##     0.126398812068074 0.131454379866444 0.147899088069494 0.157269154453983
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.158877927739738 0.160544591478956 0.167233612980717 0.174508327936485
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.17519859913437 0.18147843788037 0.187254501650306 0.187894701330005
##   0                1                1                 1                 1
##   1                0                0                 0                 0
##    tayko.lm.pred
##     0.188318679606075 0.189371162559986 0.189596177626871 0.193119646831808
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.193790991880045 0.194518181135393 0.195211546155029 0.196588721626414
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.19691915259278 0.197387463445345 0.205400086727985 0.206361467827616
##   0                1                 1                 1                 1
##   1                0                 0                 0                 0
##    tayko.lm.pred
##     0.206395169310508 0.208644754966379 0.210845552802834 0.211079805747237
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.211989156096219 0.215478540203926 0.216234973833187 0.218811542864317
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.221486105804562 0.223558608251401 0.22462900499741 0.226305069279645
##   0                 1                 1                1                 0
##   1                 0                 0                0                 1
##    tayko.lm.pred
##     0.228749622720003 0.232107434389435 0.233295595597825 0.234779075060207
##   0                 1                 1                 1                 0
##   1                 0                 0                 0                 1
##    tayko.lm.pred
##     0.235346298616418 0.237421252080103 0.247191980202637 0.249377065134815
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.252538172469065 0.256609475806738 0.257818273758601 0.263292356417283
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.265057960995326 0.265436442034652 0.265779901255197 0.265881213529484
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.267338444312242 0.268053839394659 0.273622907284325 0.274124168141228
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.279157311661299 0.284985953647006 0.286134805117315 0.287137094724882
##   0                 1                 1                 0                 1
##   1                 0                 0                 1                 0
##    tayko.lm.pred
##     0.290256141122116 0.290558768026956 0.292084426561816 0.29256544259511
##   0                 0                 1                 1                1
##   1                 1                 0                 0                0
##    tayko.lm.pred
##     0.293731842683685 0.293806737114937 0.294908391613147 0.299805347946752
##   0                 1                 1                 1                 0
##   1                 0                 0                 0                 1
##    tayko.lm.pred
##     0.302741999834637 0.30287618307805 0.303234369245028 0.306810893026607
##   0                 1                1                 1                 0
##   1                 0                0                 0                 1
##    tayko.lm.pred
##     0.307538957642002 0.30783336223458 0.310712161434974 0.312057175603777
##   0                 1                1                 1                 1
##   1                 0                0                 0                 0
##    tayko.lm.pred
##     0.317492739727914 0.317623956912823 0.322400514096722 0.326201299360491
##   0                 1                 1                 1                 1
##   1                 0                 0                 0                 0
##    tayko.lm.pred
##     0.33012200055442 0.330537806090843 0.331081469096124 0.331097870036329
##   0                1                 1                 1                 0
##   1                0                 0                 0                 1
##    tayko.lm.pred
##     0.340390836654454 0.342729928963467 0.34398837909639 0.345095231625948
##   0                 1                 1                1                 1
##   1                 0                 0                0                 0
##    tayko.lm.pred
##     0.347035458862435 0.34949620985148 0.353572839847288 0.355813333864315
##   0                 1                0                 1                 1
##   1                 0                1                 0                 0
##    tayko.lm.pred
##     0.357652844230573 0.358778020230409 0.360387730739832 0.361337999337183
##   0                 1                 0                 1                 0
##   1                 0                 1                 0                 1
##    tayko.lm.pred
##     0.365221124300368 0.368648811768407 0.369554576764573 0.370440726468354
##   0                 1                 1                 1                 0
##   1                 0                 0                 0                 1
##    tayko.lm.pred
##     0.371000565815683 0.372296726254828 0.372858014409988 0.373522430684766
##   0                 1                 1                 0                 1
##   1                 0                 0                 1                 0
##    tayko.lm.pred
##     0.373704431823454 0.374870163552152 0.37545380511286 0.377842770258757
##   0                 1                 0                1                 1
##   1                 0                 1                0                 0
##    tayko.lm.pred
##     0.378670047918457 0.378693973123413 0.37883601952975 0.38204306063235
##   0                 1                 1                1                1
##   1                 0                 0                0                0
##    tayko.lm.pred
##     0.38308208074165 0.386311232842409 0.390224344317735 0.392063366242989
##   0                1                 0                 0                 1
##   1                0                 1                 1                 0
##    tayko.lm.pred
##     0.394823107214143 0.403141546982659 0.403424815847974 0.410067644003439
##   0                 0                 1                 1                 1
##   1                 1                 0                 0                 0
##    tayko.lm.pred
##     0.411038723592986 0.411893869277738 0.413449752080629 0.413624077234248
##   0                 0                 0                 0                 1
##   1                 1                 1                 1                 0
##    tayko.lm.pred
##     0.416605282643348 0.417595908002508 0.424236163833753 0.424278713698511
##   0                 0                 1                 0                 0
##   1                 1                 0                 1                 1
##    tayko.lm.pred
##     0.426009797775625 0.428055270829394 0.431121045210167 0.431177270016523
##   0                 1                 1                 0                 1
##   1                 0                 0                 1                 0
##    tayko.lm.pred
##     0.431180770644854 0.431313690373779 0.432943547902929 0.436797045588794
##   0                 1                 1                 1                 0
##   1                 0                 0                 0                 1
##    tayko.lm.pred
##     0.436834098115691 0.437436110778408 0.437757835051446 0.438303213902131
##   0                 1                 1                 0                 0
##   1                 0                 0                 1                 1
##    tayko.lm.pred
##     0.438485400981641 0.438999274911146 0.439200634074453 0.440430988389098
##   0                 1                 0                 0                 1
##   1                 0                 1                 1                 0
##    tayko.lm.pred
##     0.442151918554648 0.442983360658308 0.443484178665527 0.444880370788235
##   0                 0                 1                 1                 1
##   1                 1                 0                 0                 0
##    tayko.lm.pred
##     0.450371445824405 0.452250695119353 0.455376027306922 0.455942886073836
##   0                 1                 1                 1                 0
##   1                 0                 0                 0                 1
##    tayko.lm.pred
##     0.456100914535601 0.456381562967362 0.45656391470949 0.457644833851458
##   0                 1                 0                1                 1
##   1                 0                 1                0                 0
##    tayko.lm.pred
##     0.458995134104424 0.460553305551633 0.461859632127053 0.463381525708237
##   0                 0                 0                 1                 1
##   1                 1                 1                 0                 0
##    tayko.lm.pred
##     0.464835261160892 0.464862428199205 0.466217590902699 0.467089121190292
##   0                 1                 0                 1                 1
##   1                 0                 1                 0                 0
##    tayko.lm.pred
##     0.470553298538339 0.474019569683486 0.478125229980529 0.478321328552432
##   0                 0                 1                 0                 1
##   1                 1                 0                 1                 0
##    tayko.lm.pred
##     0.478609613550696 0.480504540578836 0.481038114020509 0.48203278168866
##   0                 1                 0                 1                0
##   1                 0                 1                 0                1
##    tayko.lm.pred
##     0.482052038732085 0.483013693498951 0.483427301950924 0.48430443525823
##   0                 0                 1                 1                0
##   1                 1                 0                 0                1
##    tayko.lm.pred
##     0.486193807660764 0.487495042020165 0.487982891523873 0.48825453088161
##   0                 1                 1                 0                0
##   1                 0                 0                 1                1
##    tayko.lm.pred
##     0.491186220542004 0.492166231424419 0.493095717992564 0.49337379954707
##   0                 0                 1                 1                0
##   1                 1                 0                 0                1
##    tayko.lm.pred
##     0.495791795393715 0.497993093209368 0.499306393933502 0.501811109913409
##   0                 0                 1                 1                 1
##   1                 1                 0                 0                 0
##    tayko.lm.pred
##     0.503623689108394 0.503921542348855 0.505886817361641 0.508081094550364
##   0                 0                 0                 0                 1
##   1                 1                 1                 1                 0
##    tayko.lm.pred
##     0.511158942799329 0.512010747585197 0.512987641052771 0.51360213965937
##   0                 0                 0                 0                0
##   1                 1                 1                 1                1
##    tayko.lm.pred
##     0.516170354208703 0.52018522830671 0.521361430845474 0.523093142597419
##   0                 1                0                 0                 0
##   1                 0                0                 1                 1
##    tayko.lm.pred
##     0.523913880974828 0.524968147745397 0.525617031248849 0.525985247587306
##   0                 0                 1                 1                 1
##   1                 1                 0                 0                 0
##    tayko.lm.pred
##     0.527405821227299 0.527865139563013 0.528762471014541 0.530103329296723
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.530759786438573 0.534432373629863 0.538405427155733 0.542799310209495
##   0                 0                 0                 0                 1
##   1                 1                 1                 1                 0
##    tayko.lm.pred
##     0.54384890580407 0.544670409344078 0.544709565972241 0.544916205877548
##   0                0                 0                 0                 0
##   1                1                 1                 1                 1
##    tayko.lm.pred
##     0.553629250578741 0.554284505612843 0.555665620803486 0.557136761816552
##   0                 1                 0                 0                 0
##   1                 0                 1                 1                 1
##    tayko.lm.pred
##     0.557655042474711 0.558704013249597 0.558941965340414 0.560917322096238
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.562512250371346 0.562727445097939 0.562823254088636 0.564599830091232
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.564749372516561 0.565677936060874 0.565744408796228 0.56689587423044
##   0                 0                 1                 0                0
##   1                 1                 0                 1                1
##    tayko.lm.pred
##     0.567218353136853 0.568604824226344 0.575811094529975 0.576059984989667
##   0                 0                 0                 1                 0
##   1                 1                 1                 0                 1
##    tayko.lm.pred
##     0.578817641218183 0.581042155319565 0.589293284607055 0.589438231578657
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.589554087948007 0.594185027782754 0.594238932717224 0.594947116333153
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.597931573805724 0.601527623524638 0.602852256156272 0.60669199719772
##   0                 1                 0                 0                0
##   1                 0                 1                 1                1
##    tayko.lm.pred
##     0.60905613841054 0.609350854248893 0.61153066164153 0.614864912795476
##   0                1                 0                0                 0
##   1                0                 1                1                 1
##    tayko.lm.pred
##     0.615477141052316 0.617471222175525 0.617816359162285 0.620621976643008
##   0                 0                 1                 1                 0
##   1                 1                 0                 0                 1
##    tayko.lm.pred
##     0.622162695971048 0.624269290269644 0.626390877088466 0.626516579408011
##   0                 0                 0                 0                 1
##   1                 1                 1                 1                 0
##    tayko.lm.pred
##     0.62745213996181 0.630372703080998 0.631063704096165 0.637934844076508
##   0                0                 0                 0                 0
##   1                1                 1                 1                 1
##    tayko.lm.pred
##     0.640585305494997 0.641947678167294 0.642189244669554 0.643499418760865
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.648078347449448 0.656513101075065 0.657938253295606 0.658937115905397
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.659324061604466 0.659467976218886 0.660647392788636 0.663126220385377
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##     tayko.lm.pred
##      0.664630335187476 0.668956260167934 0.672800385893837 0.673244486102285
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##     tayko.lm.pred
##      0.67438030535526 0.675192058819001 0.679954062233295 0.681795507032798
##   0                0                 0                 0                 0
##   1                1                 1                 1                 1
##     tayko.lm.pred
##      0.686125023148883 0.697509639412365 0.698304016486775 0.700089049989726
##   0                 1                 0                 0                 0
##   1                 0                 1                 1                 1
##     tayko.lm.pred
##      0.700440521299287 0.701412514533205 0.710558095034333 0.717194900244949
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##     tayko.lm.pred
##      0.720251227862188 0.72516555743705 0.727692684114069 0.729785952927712
##   0                 0                0                 0                 0
##   1                 1                1                 1                 1
##     tayko.lm.pred
##      0.730883370458255 0.733326071611225 0.742345616329778 0.743310917250198
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##     tayko.lm.pred
##      0.745502585928979 0.756746558978856 0.757104647924803 0.758074441306385
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##     tayko.lm.pred
##      0.758156784005234 0.759023264979248 0.760994547622807 0.770230325087591
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##     tayko.lm.pred
##      0.772250580373983 0.775127219916568 0.777795319301726 0.784722527570462
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##     tayko.lm.pred
##      0.790120944582787 0.790209951652418 0.794435092716768 0.801800610925721
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##     tayko.lm.pred
##      0.803524615648872 0.80355394047562 0.80404826512231 0.804181785220893
##   0                 0                0                0                 0
##   1                 1                1                1                 1
##     tayko.lm.pred
##      0.8043753803078 0.8045563738219 0.810915383953021 0.81431372101964
##   0               0               0                 0                0
##   1               1               1                 1                1
##     tayko.lm.pred
##      0.821326811649212 0.835955969796298 0.837491424963505 0.845293131747083
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##     tayko.lm.pred
##     0.848872847678533 0.866952116031234 0.876723495760001 0.883989878576581
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.885268337653394 0.893668070934653 0.897480140840878 0.909650867893149
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.931156265041902 0.966303898129594 0.971694778979677 0.975148419914391
##   0                 0                 0                 0                 0
##   1                 1                 1                 1                 1
##    tayko.lm.pred
##     0.978884489392895 0.987939583633419 0.992340623403102 1.00803658901107
##   0                 0                 0                 0                0
##   1                 1                 1                 1                1
##    tayko.lm.pred
##     1.07012244007684 1.07844832538002 1.0796140079945 1.0804433506072
##   0                0                0               0               0
##   1                1                1               1               1
##    tayko.lm.pred
##     1.09910882149381 1.09918314413734 1.10231077882516 1.17656134749413
##   0                0                0                0                0
##   1                1                1                1                1
##    tayko.lm.pred
##     1.18941793943123 1.19652762691376 1.19900716573443 1.2428232482926
##   0                0                0                0               0
##   1                1                1                1               1
##    tayko.lm.pred
##     1.3346778377636 1.34052625146455 1.4030201510491 2.0572830678476
##   0               0                0               0               0
##   1               1                1               1               1
##    tayko.lm.pred
##     2.08024248454416 2.09059153682287 2.37680981246256
##   0                0                0                0
##   1                1                1                1
```

Below displays the accuracy of our Multiple regression analysis technique in which we can tell whether to be used or not in performing predictions using the dataset given when compared with other techniques performance

```
accuracy <- sum(diag(table_mat)) / sum(table_mat)

print(paste('Accuracy of Multiple regression analysis  is', accuracy))
```

```
## [1] "Accuracy of Multiple regression analysis  is 0.0025"
```

Regression trees

Below is the summary of Regression trees technique showing the prunned regression tree after the training has taken place using training dataset and ready for testing using the testing dataset

```r
library(rpart) #for fitting decision trees
library(rpart.plot) #for plotting decision trees
```

```
## Warning: package 'rpart.plot' was built under R version 4.1.3
```

```r
# we partition tayko dataset
set.seed(1234)
## partitioning into training (60%) and validation (40%)
train.rows <- sample(rownames(tayko), dim(tayko)[1]*0.6)
valid.rows <- sample(setdiff(rownames(tayko), train.rows),dim(tayko)[1]*0.2)
# assign the remaining 20% row IDs serve as test
test.rows <- setdiff(rownames(tayko), union(train.rows, valid.rows))
# create the 3 data frames by collecting all columns from the appropriate rows
train.data <- tayko[train.rows, ]
valid.data <- tayko[valid.rows, ]
test.data <- tayko[test.rows, ]

#build the initial tree
tree <- rpart(Purchase ~ ., data=train.data, control=rpart.control(cp=.0001))

#view results
printcp(tree)
```

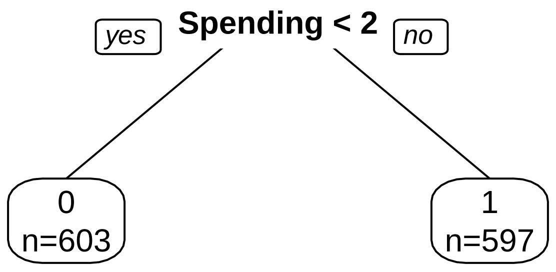
```
##
## Regression tree:
## rpart(formula = Purchase ~ ., data = train.data, control = rpart.control(cp = 1e-04))
##
## Variables actually used in tree construction:
## [1] Spending
##
## Root node error: 299.99/1200 = 0.24999
##
## n= 1200
##
##       CP nsplit rel error    xerror       xstd
## 1 1e+00      0         1 1.0025288 0.00072325
## 2 1e-04      1         0 0.0033334 0.00333203
```

```r
#identify best cp value to use
best <- tree$cptable[which.min(tree$cptable[,"xerror"]),"CP"]

#produce a pruned tree based on the best cp value
pruned_tree <- prune(tree, cp=best)

#plot the pruned tree
```

```r
prp(pruned_tree,
    faclen=0, #use full names for factor labels
    extra=1, #display number of obs. for each terminal node
    roundint=F, #don't round to integers in output
    digits=5) #display 5 decimal places in output
```



Below are the predicted values of our Purchase variable based on the predictor variables using our Regression Tree technique

```r
# use predict() with type = "response" to compute predicted probabilities.
pruned_tree.pred <- predict(pruned_tree, valid.data, type = "matrix")
# first 5 actual and predicted records
data.frame(actual = valid.data$Purchase, predicted = pruned_tree.pred)
```

```
##     actual predicted
## 1        1         1
## 2        0         0
## 3        1         1
## 4        1         1
## 5        0         0
## 6        1         1
## 7        1         1
## 8        1         1
## 9          0           0
## 10         1           1
## 11         0           0
## 12         1           1
## 13         0           0
## 14         1           1
## 15         1           1
## 16         0           0
## 17         0           0
## 18         1           1
## 19         1           1
## 20         1           1
## 21         1           1
## 22         0           0
## 23         1           1
## 24         0           0
## 25         1           1
## 26         1           1
## 27         1           1
## 28         0           0
## 29         0           0
## 30         1           1
## 31         0           0
## 32         0           0
## 33         1           1
## 34         0           0
## 35         1           1
## 36         1           1
## 37         0           0
## 38         0           0
## 39         1           1
## 40         1           1
## 41         1           1
## 42         0           0
## 43         1           1
## 44         1           1
## 45         0           0
## 46         0           0
## 47         1           1
## 48         1           1
## 49         1           1
## 50         0           0
## 51         0           0
## 52         1           1
## 53         1           1
## 54         1           1
## 55         0           0
## 56         1           1
## 57         1           1
## 58         1           1
## 59         0           0
## 60         0           0
## 61         0           0
## 62         1           1
## 63         1          1
## 64         1          1
## 65         0          0
## 66         0          0
## 67         1          1
## 68         0          0
## 69         0          0
## 70         1          1
## 71         1          1
## 72         1          1
## 73         1          1
## 74         1          1
## 75         0          0
## 76         1          1
## 77         0          0
## 78         0          0
## 79         0          0
## 80         1          1
## 81         1          1
## 82         1          1
## 83         1          1
## 84         0          0
## 85         0          0
## 86         0          0
## 87         0          0
## 88         0          0
## 89         1          1
## 90         0          0
## 91         1          1
## 92         1          1
## 93         0          0
## 94         1          1
## 95         1          1
## 96         0          0
## 97         1          1
## 98         1          1
## 99         0          0
## 100        0          0
## 101        1          1
## 102        1          1
## 103        1          1
## 104        0          0
## 105        1          1
## 106        0          0
## 107        0          0
## 108        0          0
## 109        0          0
## 110        1          1
## 111        0          0
## 112        1          1
## 113        0          0
## 114        0          0
## 115        0          0
## 116        1          1
## 117        1           1
## 118        1           1
## 119        1           1
## 120        0           0
## 121        1           1
## 122        0           0
## 123        1           1
## 124        0           0
## 125        0           0
## 126        0           0
## 127        1           1
## 128        0           0
## 129        0           0
## 130        0           0
## 131        0           0
## 132        0           0
## 133        0           0
## 134        1           1
## 135        0           0
## 136        0           0
## 137        1           1
## 138        0           0
## 139        0           0
## 140        1           1
## 141        1           1
## 142        0           0
## 143        1           1
## 144        0           0
## 145        0           0
## 146        1           1
## 147        1           1
## 148        0           0
## 149        0           0
## 150        0           0
## 151        1           1
## 152        0           0
## 153        0           0
## 154        1           1
## 155        0           0
## 156        1           1
## 157        0           0
## 158        0           0
## 159        1           1
## 160        1           1
## 161        0           0
## 162        1           1
## 163        0           0
## 164        1           1
## 165        1           1
## 166        0           0
## 167        0           0
## 168        0           0
## 169        1           1
## 170        1           1
## 171       0         0
## 172       0         0
## 173       1         1
## 174       0         0
## 175       1         1
## 176       1         1
## 177       0         0
## 178       1         1
## 179       1         1
## 180       0         0
## 181       0         0
## 182       1         1
## 183       0         0
## 184       1         1
## 185       0         0
## 186       0         0
## 187       1         1
## 188       0         0
## 189       1         1
## 190       0         0
## 191       0         0
## 192       0         0
## 193       0         0
## 194       1         1
## 195       1         1
## 196       1         1
## 197       0         0
## 198       1         1
## 199       1         1
## 200       1         1
## 201       0         0
## 202       1         1
## 203       0         0
## 204       1         1
## 205       0         0
## 206       0         0
## 207       0         0
## 208       0         0
## 209       1         1
## 210       0         0
## 211       1         1
## 212       0         0
## 213       1         1
## 214       0         0
## 215       1         1
## 216       0         0
## 217       1         1
## 218       0         0
## 219       1         1
## 220       1         1
## 221       1         1
## 222       0         0
## 223       1         1
## 224       1         1
## 225         1            1
## 226         0            0
## 227         1            1
## 228         0            0
## 229         0            0
## 230         0            0
## 231         1            1
## 232         0            0
## 233         0            0
## 234         0            0
## 235         0            0
## 236         1            1
## 237         0            0
## 238         1            1
## 239         0            0
## 240         0            0
## 241         0            0
## 242         1            1
## 243         1            1
## 244         0            0
## 245         1            1
## 246         1            1
## 247         0            0
## 248         0            0
## 249         1            1
## 250         1            1
## 251         0            0
## 252         1            1
## 253         1            1
## 254         1            1
## 255         1            1
## 256         0            0
## 257         1            1
## 258         0            0
## 259         0            0
## 260         0            0
## 261         1            1
## 262         0            0
## 263         0            0
## 264         1            1
## 265         1            1
## 266         1            1
## 267         1            1
## 268         1            1
## 269         0            0
## 270         0            0
## 271         0            0
## 272         0            0
## 273         1            1
## 274         1            1
## 275         0            0
## 276         1            1
## 277         1            1
## 278         0            0
## 279          0           0
## 280          1           1
## 281          0           0
## 282          0           0
## 283          1           1
## 284          0           0
## 285          1           1
## 286          0           0
## 287          0           0
## 288          1           1
## 289          1           1
## 290          1           1
## 291          1           1
## 292          1           1
## 293          0           0
## 294          1           1
## 295          0           0
## 296          0           0
## 297          0           0
## 298          0           0
## 299          1           1
## 300          0           0
## 301          0           0
## 302          1           1
## 303          1           1
## 304          1           1
## 305          0           0
## 306          0           0
## 307          0           0
## 308          0           0
## 309          0           0
## 310          1           1
## 311          1           1
## 312          0           0
## 313          1           1
## 314          1           1
## 315          1           1
## 316          0           0
## 317          0           0
## 318          0           0
## 319          1           1
## 320          1           1
## 321          1           1
## 322          0           0
## 323          1           1
## 324          1           1
## 325          0           0
## 326          0           0
## 327          0           0
## 328          1           1
## 329          0           0
## 330          1           1
## 331          1           1
## 332          1           1
## 333        1           1
## 334        1           1
## 335        1           1
## 336        0           0
## 337        0           0
## 338        0           0
## 339        1           1
## 340        0           0
## 341        1           1
## 342        0           0
## 343        1           1
## 344        0           0
## 345        1           1
## 346        0           0
## 347        0           0
## 348        1           1
## 349        1           1
## 350        1           1
## 351        1           1
## 352        1           1
## 353        0           0
## 354        1           1
## 355        1           1
## 356        1           1
## 357        0           0
## 358        0           0
## 359        1           1
## 360        1           1
## 361        0           0
## 362        0           0
## 363        0           0
## 364        0           0
## 365        1           1
## 366        0           0
## 367        0           0
## 368        1           1
## 369        1           1
## 370        0           0
## 371        0           0
## 372        0           0
## 373        1           1
## 374        1           1
## 375        0           0
## 376        0           0
## 377        1           1
## 378        1           1
## 379        1           1
## 380        0           0
## 381        1           1
## 382        0           0
## 383        1           1
## 384        1           1
## 385        0           0
## 386        1           1
## 387      0         0
## 388      1         1
## 389      1         1
## 390      1         1
## 391      1         1
## 392      1         1
## 393      0         0
## 394      1         1
## 395      0         0
## 396      1         1
## 397      0         0
## 398      1         1
## 399      1         1
## 400      0         0
```

```
length(pruned_tree.pred)
```

```
## [1] 400
```

```
table_mat <- table(valid.data$Purchase, pruned_tree.pred)

table_mat
```

```
##    pruned_tree.pred
##       0   1
##   0 195   0
##   1   0 205
```

Below displays the accuracy of our Regression Tree technique in which we can tell whether to be used or not in performing predictions using the dataset given when compared with other techniques performance

```
accuracy <- sum(diag(table_mat)) / sum(table_mat)

print(paste('Accuracy of Regression Tree  is', accuracy))
```

```
## [1] "Accuracy of Regression Tree  is 1"
```

Logistic regression

Below is the summary of Logistic regression technique showing its coefficients and residuals after the training has taken place using training dataset and ready for testing using the testing dataset

```
# we partition tayko dataset
set.seed(1234)
## partitioning into training (60%) and validation (40%)
train.rows <- sample(rownames(tayko), dim(tayko)[1]*0.6)
```

```
valid.rows <- sample(setdiff(rownames(tayko), train.rows),dim(tayko)[1]*0.2)
# assign the remaining 20% row IDs serve as test
test.rows <- setdiff(rownames(tayko), union(train.rows, valid.rows))
# create the 3 data frames by collecting all columns from the appropriate rows
train.data <- tayko[train.rows, ]
valid.data <- tayko[valid.rows, ]
test.data <- tayko[test.rows, ]


# run logistic regression
# use glm() (general linear model) with family = "binomial" to fit a logistic
# regression.
logit.reg <- glm(Purchase ~ ., data = train.data, family = "binomial")
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(logit.reg)
```

```
##
## Call:
## glm(formula = Purchase ~ ., family = "binomial", data = train.data)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -7.656e-04  -2.000e-08  -2.000e-08   2.000e-08   1.204e-03
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -2.737e+02  4.906e+05  -0.001    1.000
## sequence_number        -4.006e-03  1.036e+00  -0.004    0.997
## US                     -1.065e+01  1.274e+03  -0.008    0.993
## source_a                1.501e+02  4.906e+05   0.000    1.000
## source_c                2.283e+02  4.906e+05   0.000    1.000
## source_b                2.195e+02  5.262e+05   0.000    1.000
## source_d                2.316e+02  4.907e+05   0.000    1.000
## source_e                2.326e+02  4.906e+05   0.000    1.000
## source_m                8.349e+01  5.046e+05   0.000    1.000
## source_o                3.615e+01  4.943e+05   0.000    1.000
## source_h                2.100e+02  5.036e+05   0.000    1.000
## source_r                1.361e+02  4.915e+05   0.000    1.000
## source_s                2.349e+02  4.907e+05   0.000    1.000
## source_t                2.237e+02  5.063e+05   0.000    1.000
## source_u                2.359e+02  4.906e+05   0.000    1.000
## source_p                3.016e+01  4.982e+05   0.000    1.000
## source_x                1.084e+02  5.683e+05   0.000    1.000
## source_w                6.348e+01  4.954e+05   0.000    1.000
## Freq                    1.328e+01  1.588e+03   0.008    0.993
## last_update_days_ago    2.522e-02  5.517e+00   0.005    0.996
## `1st_update_days_ago` -2.095e-02  5.504e+00  -0.004    0.997
## Web.order               4.430e+00  1.308e+03   0.003    0.997
## Gender                -1.483e+00  1.067e+03  -0.001    0.999
## Address_is_res         5.499e+00  1.256e+03   0.004    0.997
## Spending               5.824e+00  1.464e+02   0.040    0.968
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1.6635e+03  on 1199  degrees of freedom
## Residual deviance: 6.2298e-06  on 1175  degrees of freedom
## AIC: 50
##
## Number of Fisher Scoring iterations: 25
```

Below are the predicted values of our Purchase variable based on the predictor variables using our Logistic Regression technique

```
# use predict() with type = "response" to compute predicted probabilities.
logit.reg.pred <- predict(logit.reg, valid.data, type = "response")
# first 5 actual and predicted records
data.frame(actual = valid.data$Purchase, predicted = logit.reg.pred)
```

```
##     actual    predicted
## 1        1 1.000000e+00
## 2        0 2.220446e-16
## 3        1 1.000000e+00
## 4        1 1.000000e+00
## 5        0 2.220446e-16
## 6        1 1.000000e+00
## 7        1 1.000000e+00
## 8        1 1.000000e+00
## 9        0 2.220446e-16
## 10       1 1.000000e+00
## 11       0 2.220446e-16
## 12       1 1.000000e+00
## 13       0 2.220446e-16
## 14       1 1.000000e+00
## 15       1 1.000000e+00
## 16       0 4.215339e-06
## 17       0 2.220446e-16
## 18       1 1.000000e+00
## 19       1 1.000000e+00
## 20       1 1.000000e+00
## 21       1 1.000000e+00
## 22       0 2.220446e-16
## 23       1 1.000000e+00
## 24       0 4.772959e-09
## 25       1 1.000000e+00
## 26       1 1.000000e+00
## 27       1 1.000000e+00
## 28       0 2.220446e-16
## 29       0 2.220446e-16
## 30       1 1.000000e+00
## 31       0 1.897599e-05
## 32        0 2.220446e-16
## 33        1 1.000000e+00
## 34        0 2.220446e-16
## 35        1 1.000000e+00
## 36        1 1.000000e+00
## 37        0 8.630096e-12
## 38        0 2.220446e-16
## 39        1 1.000000e+00
## 40        1 1.000000e+00
## 41        1 1.000000e+00
## 42        0 2.220446e-16
## 43        1 1.000000e+00
## 44        1 1.000000e+00
## 45        0 2.220446e-16
## 46        0 1.050949e-08
## 47        1 1.000000e+00
## 48        1 1.000000e+00
## 49        1 1.000000e+00
## 50        0 2.220446e-16
## 51        0 2.220446e-16
## 52        1 1.000000e+00
## 53        1 1.000000e+00
## 54        1 1.000000e+00
## 55        0 2.220446e-16
## 56        1 1.000000e+00
## 57        1 1.000000e+00
## 58        1 1.000000e+00
## 59        0 2.220446e-16
## 60        0 2.220446e-16
## 61        0 2.220446e-16
## 62        1 1.000000e+00
## 63        1 1.000000e+00
## 64        1 1.000000e+00
## 65        0 2.220446e-16
## 66        0 2.220446e-16
## 67        1 1.000000e+00
## 68        0 2.220446e-16
## 69        0 2.220446e-16
## 70        1 1.000000e+00
## 71        1 1.000000e+00
## 72        1 1.000000e+00
## 73        1 1.000000e+00
## 74        1 1.000000e+00
## 75        0 1.100291e-13
## 76        1 1.000000e+00
## 77        0 2.220446e-16
## 78        0 2.220446e-16
## 79        0 2.220446e-16
## 80        1 1.000000e+00
## 81        1 1.000000e+00
## 82        1 1.000000e+00
## 83        1 1.000000e+00
## 84        0 2.220446e-16
## 85        0 2.220446e-16
## 86        0 2.220446e-16
## 87        0 2.220446e-16
## 88        0 2.671294e-11
## 89        1 1.000000e+00
## 90        0 2.220446e-16
## 91        1 1.000000e+00
## 92        1 1.000000e+00
## 93        0 1.699130e-11
## 94        1 1.000000e+00
## 95        1 1.000000e+00
## 96        0 2.220446e-16
## 97        1 1.000000e+00
## 98        1 1.000000e+00
## 99        0 2.220446e-16
## 100       0 1.714382e-12
## 101       1 1.000000e+00
## 102       1 1.000000e+00
## 103       1 1.000000e+00
## 104       0 2.220446e-16
## 105       1 1.000000e+00
## 106       0 2.220446e-16
## 107       0 2.220446e-16
## 108       0 4.722666e-13
## 109       0 4.137033e-08
## 110       1 1.000000e+00
## 111       0 2.220446e-16
## 112       1 1.000000e+00
## 113       0 2.220446e-16
## 114       0 2.220446e-16
## 115       0 2.220446e-16
## 116       1 1.000000e+00
## 117       1 1.000000e+00
## 118       1 1.000000e+00
## 119       1 1.000000e+00
## 120       0 2.220446e-16
## 121       1 1.000000e+00
## 122       0 2.220446e-16
## 123       1 1.000000e+00
## 124       0 2.220446e-16
## 125       0 2.220446e-16
## 126       0 2.220446e-16
## 127       1 1.000000e+00
## 128       0 2.220446e-16
## 129       0 2.220446e-16
## 130       0 2.220446e-16
## 131       0 2.220446e-16
## 132       0 2.267338e-06
## 133       0 2.220446e-16
## 134       1 1.000000e+00
## 135       0 2.220446e-16
## 136       0 9.037236e-11
## 137       1 1.000000e+00
## 138       0 2.220446e-16
## 139       0 2.220446e-16
## 140      1 1.000000e+00
## 141      1 1.000000e+00
## 142      0 2.220446e-16
## 143      1 1.000000e+00
## 144      0 2.220446e-16
## 145      0 2.220446e-16
## 146      1 1.000000e+00
## 147      1 1.000000e+00
## 148      0 2.220446e-16
## 149      0 2.220446e-16
## 150      0 1.974614e-13
## 151      1 1.000000e+00
## 152      0 2.220446e-16
## 153      0 2.220446e-16
## 154      1 1.000000e+00
## 155      0 2.220446e-16
## 156      1 1.000000e+00
## 157      0 3.646291e-09
## 158      0 2.220446e-16
## 159      1 1.000000e+00
## 160      1 1.000000e+00
## 161      0 2.220446e-16
## 162      1 1.000000e+00
## 163      0 2.220446e-16
## 164      1 1.000000e+00
## 165      1 1.000000e+00
## 166      0 2.220446e-16
## 167      0 9.507445e-14
## 168      0 2.220446e-16
## 169      1 1.000000e+00
## 170      1 1.000000e+00
## 171      0 2.220446e-16
## 172      0 2.220446e-16
## 173      1 1.000000e+00
## 174      0 2.220446e-16
## 175      1 1.000000e+00
## 176      1 1.000000e+00
## 177      0 2.220446e-16
## 178      1 1.000000e+00
## 179      1 1.000000e+00
## 180      0 2.220446e-16
## 181      0 1.737500e-13
## 182      1 1.000000e+00
## 183      0 2.220446e-16
## 184      1 1.000000e+00
## 185      0 2.220446e-16
## 186      0 2.220446e-16
## 187      1 1.000000e+00
## 188      0 2.220446e-16
## 189      1 1.000000e+00
## 190      0 2.220446e-16
## 191      0 2.220446e-16
## 192      0 2.220446e-16
## 193      0 2.220446e-16
## 194       1 1.000000e+00
## 195       1 1.000000e+00
## 196       1 1.000000e+00
## 197       0 2.220446e-16
## 198       1 2.220446e-16
## 199       1 1.000000e+00
## 200       1 1.000000e+00
## 201       0 2.220446e-16
## 202       1 1.000000e+00
## 203       0 2.220446e-16
## 204       1 1.000000e+00
## 205       0 2.220446e-16
## 206       0 2.220446e-16
## 207       0 2.220446e-16
## 208       0 2.220446e-16
## 209       1 1.000000e+00
## 210       0 7.274193e-13
## 211       1 1.000000e+00
## 212       0 2.220446e-16
## 213       1 1.000000e+00
## 214       0 2.220446e-16
## 215       1 1.000000e+00
## 216       0 2.220446e-16
## 217       1 1.000000e+00
## 218       0 2.220446e-16
## 219       1 1.000000e+00
## 220       1 1.000000e+00
## 221       1 1.000000e+00
## 222       0 2.220446e-16
## 223       1 1.000000e+00
## 224       1 1.000000e+00
## 225       1 2.220446e-16
## 226       0 2.220446e-16
## 227       1 1.000000e+00
## 228       0 2.220446e-16
## 229       0 2.220446e-16
## 230       0 2.220446e-16
## 231       1 1.000000e+00
## 232       0 2.220446e-16
## 233       0 2.220446e-16
## 234       0 2.220446e-16
## 235       0 2.220446e-16
## 236       1 1.000000e+00
## 237       0 2.220446e-16
## 238       1 2.220446e-16
## 239       0 2.220446e-16
## 240       0 2.220446e-16
## 241       0 2.220446e-16
## 242       1 1.000000e+00
## 243       1 1.000000e+00
## 244       0 2.220446e-16
## 245       1 1.000000e+00
## 246       1 1.000000e+00
## 247       0 2.220446e-16
## 248      0 2.220446e-16
## 249      1 1.000000e+00
## 250      1 1.000000e+00
## 251      0 2.220446e-16
## 252      1 1.000000e+00
## 253      1 1.000000e+00
## 254      1 1.000000e+00
## 255      1 1.000000e+00
## 256      0 2.220446e-16
## 257      1 1.000000e+00
## 258      0 1.668907e-08
## 259      0 2.220446e-16
## 260      0 2.220446e-16
## 261      1 1.000000e+00
## 262      0 2.220446e-16
## 263      0 2.220446e-16
## 264      1 1.000000e+00
## 265      1 1.000000e+00
## 266      1 1.000000e+00
## 267      1 1.000000e+00
## 268      1 1.000000e+00
## 269      0 2.220446e-16
## 270      0 2.220446e-16
## 271      0 2.926361e-09
## 272      0 2.220446e-16
## 273      1 9.999982e-01
## 274      1 1.000000e+00
## 275      0 2.220446e-16
## 276      1 1.000000e+00
## 277      1 1.000000e+00
## 278      0 2.220446e-16
## 279      0 2.220446e-16
## 280      1 1.000000e+00
## 281      0 2.220446e-16
## 282      0 2.220446e-16
## 283      1 1.000000e+00
## 284      0 2.220446e-16
## 285      1 1.000000e+00
## 286      0 2.220446e-16
## 287      0 2.220446e-16
## 288      1 1.000000e+00
## 289      1 1.000000e+00
## 290      1 1.000000e+00
## 291      1 1.000000e+00
## 292      1 1.000000e+00
## 293      0 2.220446e-16
## 294      1 1.000000e+00
## 295      0 2.414091e-11
## 296      0 2.220446e-16
## 297      0 2.220446e-16
## 298      0 2.220446e-16
## 299      1 1.000000e+00
## 300      0 2.220446e-16
## 301      0 2.220446e-16
## 302      1 1.000000e+00
## 303      1 1.000000e+00
## 304      1 2.220446e-16
## 305      0 2.220446e-16
## 306      0 2.220446e-16
## 307      0 2.220446e-16
## 308      0 2.220446e-16
## 309      0 2.220446e-16
## 310      1 1.000000e+00
## 311      1 1.000000e+00
## 312      0 2.220446e-16
## 313      1 1.000000e+00
## 314      1 1.000000e+00
## 315      1 1.000000e+00
## 316      0 2.220446e-16
## 317      0 2.220446e-16
## 318      0 2.220446e-16
## 319      1 1.000000e+00
## 320      1 1.000000e+00
## 321      1 1.000000e+00
## 322      0 2.220446e-16
## 323      1 1.000000e+00
## 324      1 1.000000e+00
## 325      0 2.220446e-16
## 326      0 2.220446e-16
## 327      0 2.220446e-16
## 328      1 1.000000e+00
## 329      0 2.220446e-16
## 330      1 1.000000e+00
## 331      1 1.000000e+00
## 332      1 1.000000e+00
## 333      1 1.000000e+00
## 334      1 1.000000e+00
## 335      1 1.000000e+00
## 336      0 2.220446e-16
## 337      0 2.220446e-16
## 338      0 2.220446e-16
## 339      1 1.000000e+00
## 340      0 2.220446e-16
## 341      1 1.000000e+00
## 342      0 2.220446e-16
## 343      1 1.000000e+00
## 344      0 2.220446e-16
## 345      1 1.000000e+00
## 346      0 2.220446e-16
## 347      0 2.220446e-16
## 348      1 1.000000e+00
## 349      1 1.000000e+00
## 350      1 1.000000e+00
## 351      1 1.000000e+00
## 352      1 1.000000e+00
## 353      0 4.472597e-10
## 354      1 1.000000e+00
## 355      1 1.000000e+00
## 356      1 1.000000e+00
## 357      0 2.220446e-16
## 358      0 1.543554e-12
## 359      1 1.000000e+00
## 360      1 1.000000e+00
## 361      0 2.220446e-16
## 362      0 2.220446e-16
## 363      0 2.220446e-16
## 364      0 2.220446e-16
## 365      1 1.000000e+00
## 366      0 2.220446e-16
## 367      0 2.220446e-16
## 368      1 1.000000e+00
## 369      1 1.000000e+00
## 370      0 2.220446e-16
## 371      0 2.220446e-16
## 372      0 3.866907e-13
## 373      1 1.000000e+00
## 374      1 1.000000e+00
## 375      0 2.871882e-13
## 376      0 2.220446e-16
## 377      1 1.000000e+00
## 378      1 1.000000e+00
## 379      1 1.000000e+00
## 380      0 2.220446e-16
## 381      1 1.000000e+00
## 382      0 1.897004e-09
## 383      1 1.000000e+00
## 384      1 1.000000e+00
## 385      0 2.220446e-16
## 386      1 1.000000e+00
## 387      0 2.220446e-16
## 388      1 1.000000e+00
## 389      1 1.000000e+00
## 390      1 1.000000e+00
## 391      1 1.000000e+00
## 392      1 1.000000e+00
## 393      0 1.967193e-13
## 394      1 2.220446e-16
## 395      0 2.220446e-16
## 396      1 1.000000e+00
## 397      0 2.220446e-16
## 398      1 1.000000e+00
## 399      1 1.000000e+00
## 400      0 2.220446e-16
```

```
table_mat <- table(valid.data$Purchase, logit.reg.pred)
```

Below displays the accuracy of our Logistic Regression technique in which we can tell whether to be used or not in performing predictions using the dataset given when compared with other techniques performance

```
accuracy <- sum(diag(table_mat)) / sum(table_mat)

print(paste('Accuracy of Logistic Regression is', accuracy))
```

```
## [1] "Accuracy of Logistic Regression is 0.42"
```

In nutshell, from the 3 data mining results, based on each technique accuracy, its clear that, Regression tree best fits to the dataset given as its performance on the accuracy is perfect when compared to Logistic regression and Multiple Regression Analysis.

Therefore, i select Regression tree as the best fit technique to be used to classify a "purchase" or "no purchase"