Kranthi case 1

2022-07-05

Introduction

Executive Summary

- I started to understand the dataset by importing data and this data set involves many 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
```

```
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

```
dim(tayko)
```

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

```
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

```
# 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"
```

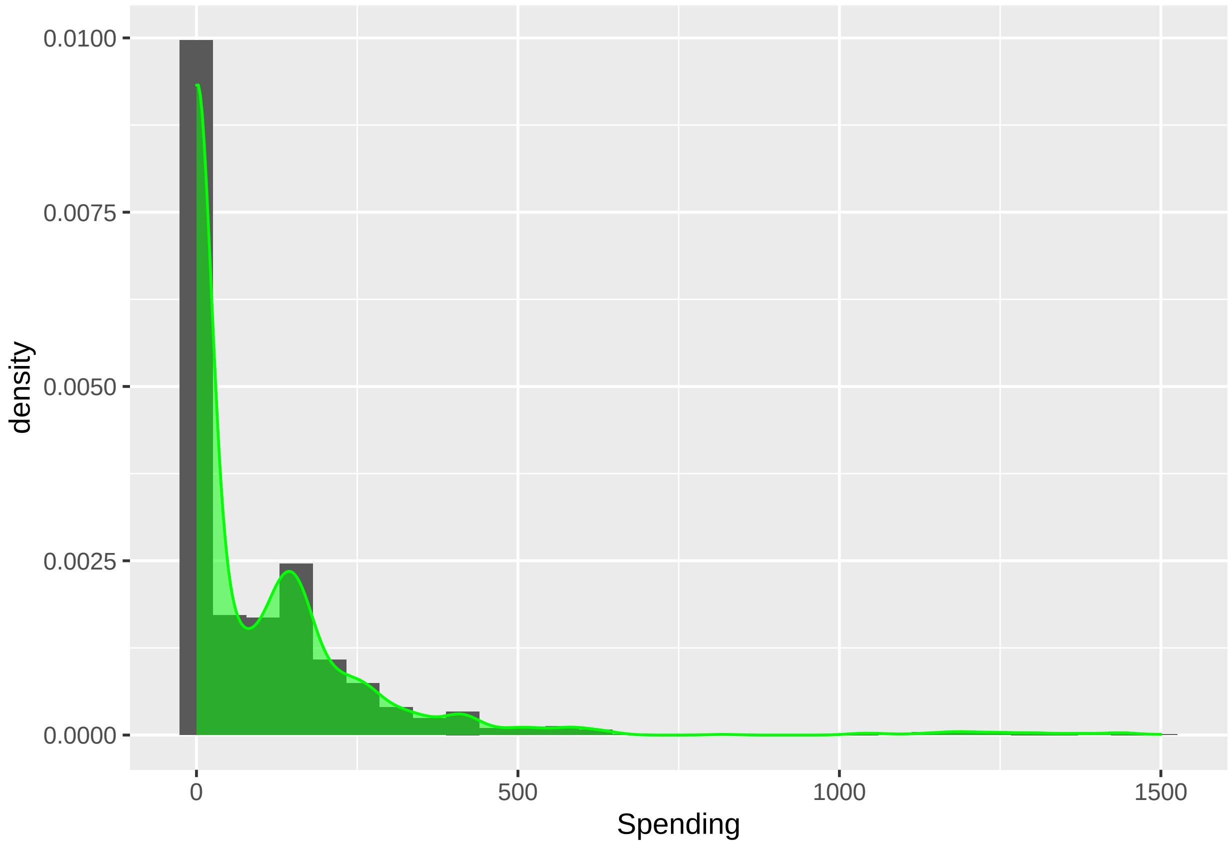
```
names(tayko)[21] <- "Web.order"
names(tayko)[22] <- "Gender"


# 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
```

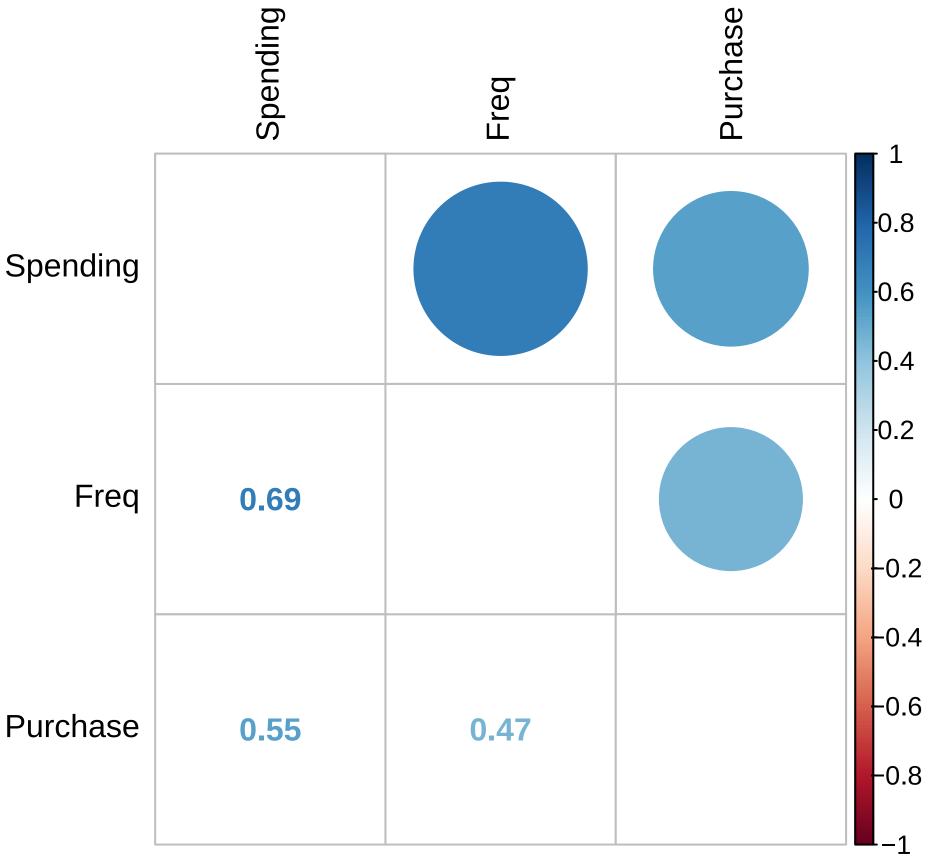
```r
numericVars <- which(sapply(tayko, is.numeric)) #index vector numeric variables
numericVarNames <- names(numericVars) #saving names vector for use later on
cat('There are', length(numericVars), 'numeric variables')
```

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

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

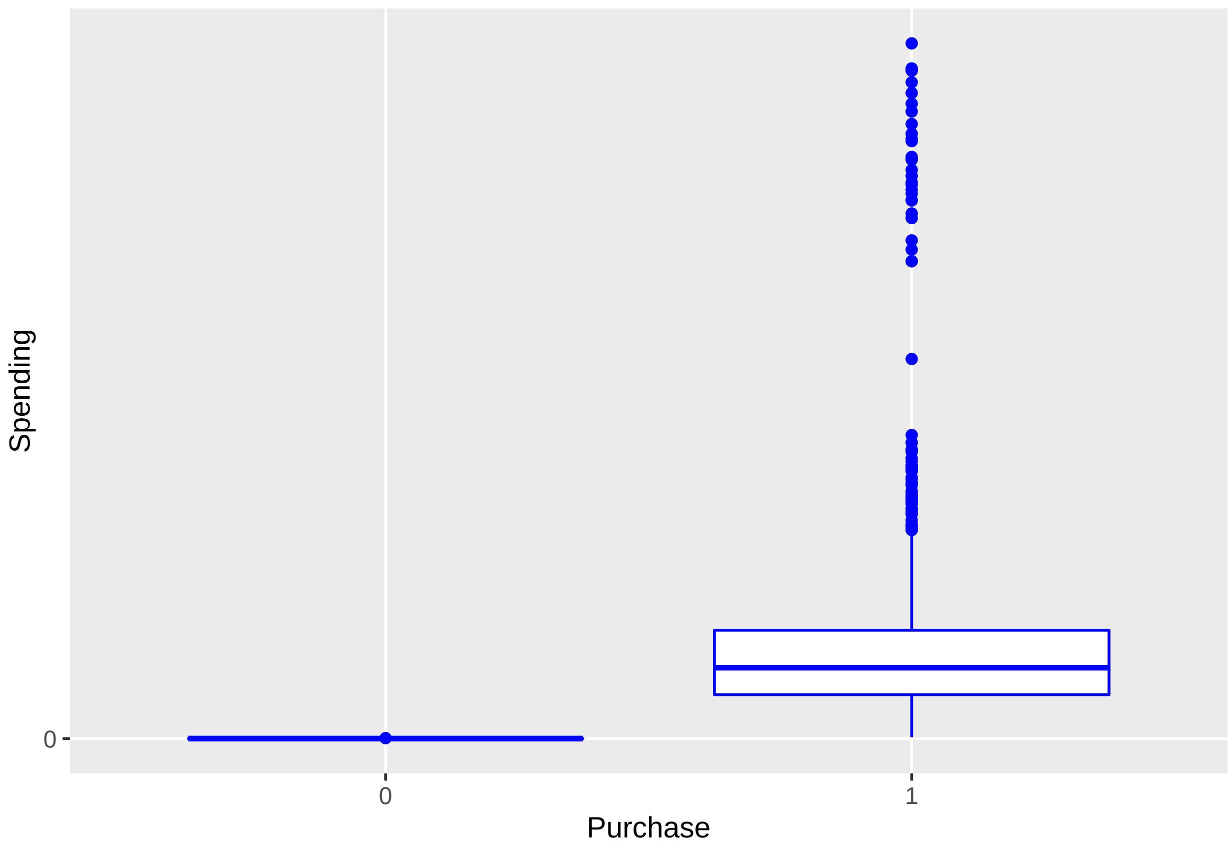
```
#sort on decreasing correlations with SalePrice
cor_sorted <- as.matrix(sort(cor_numVar[,'Spending'], decreasing = TRUE))
 #select only high corelations
CorHigh <- names(which(apply(cor_sorted, 1, function(x) abs(x)>0.5)))
cor_numVar <- cor_numVar[CorHigh, CorHigh]

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



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

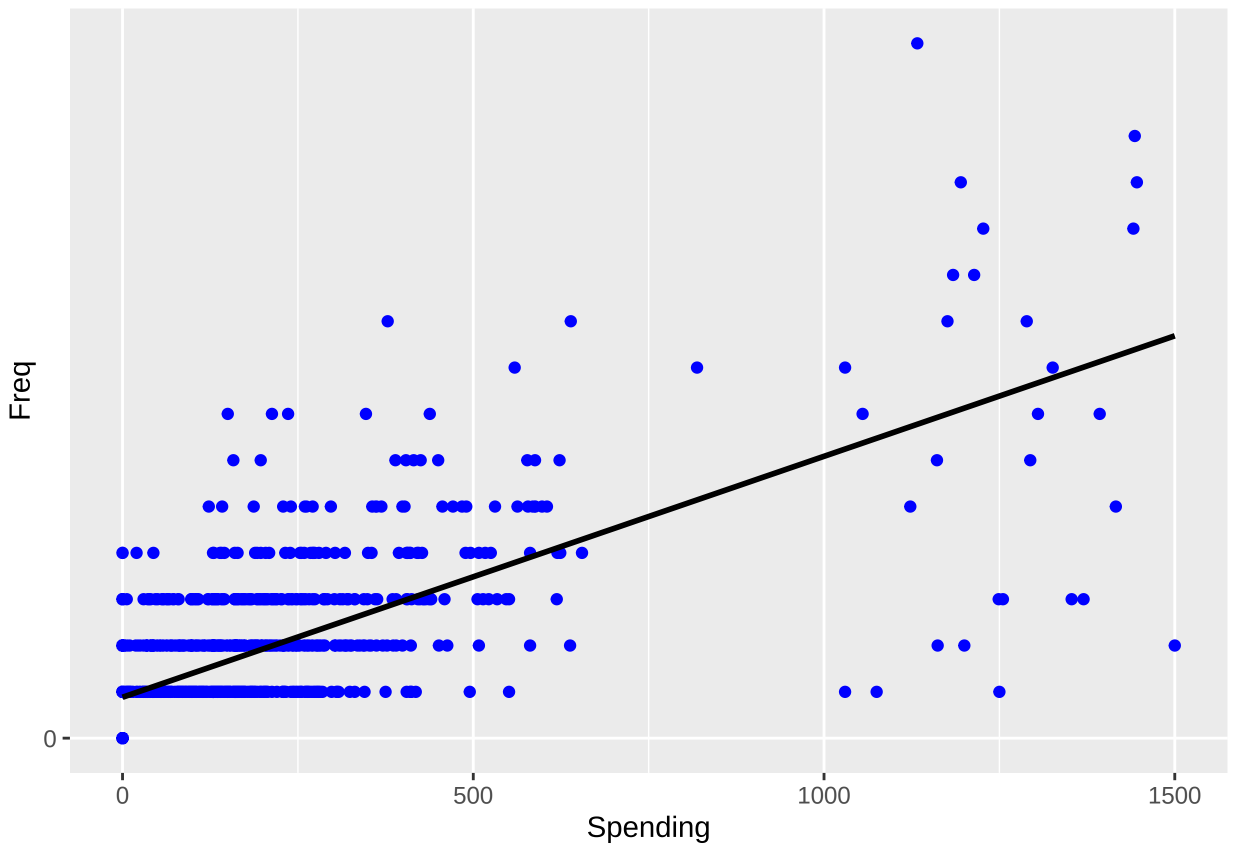
```
ggplot(data=tayko[!is.na(tayko$Spending),], aes(x=factor(Purchase), y=Spending))+
        geom_boxplot(col='blue') + labs(x='Purchase') +
        scale_y_continuous(breaks= seq(0, 80, by=1000), labels = comma)
```



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.

Assignment

1. Each catalog costs approximately $2 to mail (including printing, postage,

and mailing costs). Estimate the gross profit that the firm could expect

from the remaining 180,000 names if it selects them randomly from the

pool.

```
## 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.
```

2. Develop a model for classifying a customer as a purchaser or nonpurchaser.

a. Partition the data randomly into a training set (800 records), validation

set (700 records), and test set (500 records).

```
## 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.

## # A tibble: 5 x 25
##   sequence_number    US source_a source_c source_b source_d source_e source_m
##             <dbl> <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1            1308     1        0        0        0        0        0        0
## 2            1872     1        0        0        0        0        1        0
## 3            1018     1        0        0        0        0        0        0
## 4            1942     1        0        0        1        0        0        0
## 5            1125     1        0        0        0        0        0        0
## # ... with 17 more variables: source_o <dbl>, source_h <dbl>, source_r <dbl>,
## #   source_s <dbl>, source_t <dbl>, source_u <dbl>, source_p <dbl>,
## #   source_x <dbl>, source_w <dbl>, Freq <dbl>, last_update_days_ago <dbl>,
## #   ‘1st_update_days_ago‘ <dbl>, ‘Web order‘ <dbl>, ‘Gender=male‘ <dbl>,
## #   Address_is_res <dbl>, Purchase <dbl>, Spending <dbl>

## # A tibble: 5 x 25
##   sequence_number    US source_a source_c source_b source_d source_e source_m
##             <dbl> <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1            1223     1        0        0        0        0        0        1
## 2             610     1        0        0        0        0        0        0
## 3            1435     1        0        0        0        0        1        0
## 4              48     1        0        0        0        0        0        0
## 5             785     0        0        0        1        0        0        0
## # ... with 17 more variables: source_o <dbl>, source_h <dbl>, source_r <dbl>,
## #   source_s <dbl>, source_t <dbl>, source_u <dbl>, source_p <dbl>,
## #   source_x <dbl>, source_w <dbl>, Freq <dbl>, last_update_days_ago <dbl>,
## #   ‘1st_update_days_ago‘ <dbl>, ‘Web order‘ <dbl>, ‘Gender=male‘ <dbl>,
## #   Address_is_res <dbl>, Purchase <dbl>, Spending <dbl>

## # A tibble: 5 x 25
##   sequence_number    US source_a source_c source_b source_d source_e source_m
##             <dbl> <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1               1     1        0        0        1        0        0        0
## 2               3     1        0        0        0        0        0        0
## 3               6     1        0        0        0        0        0        0
## 4               8     1        0        0        1        0        0        0
## 5               9     1        1        0        0        0        0        0
## # ... with 17 more variables: source_o <dbl>, source_h <dbl>, source_r <dbl>,
## #   source_s <dbl>, source_t <dbl>, source_u <dbl>, source_p <dbl>,
## #   source_x <dbl>, source_w <dbl>, Freq <dbl>, last_update_days_ago <dbl>,
## #   ‘1st_update_days_ago‘ <dbl>, ‘Web order‘ <dbl>, ‘Gender=male‘ <dbl>,
## #   Address_is_res <dbl>, Purchase <dbl>, Spending <dbl>
```

b. Run stepwise logistic regression using backward elimination to select the best subset of variables, then use this model to classify the data into purchasers and nonpurchasers. Use only the training set for running the model. (Logistic regression is used because it yields an estimated "probability of purchase," which is required later in the analysis.)

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.7     v stringr 1.4.0
## v tidyr   1.2.0     v forcats 0.5.1
## v purrr   0.3.4

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::arrange()     masks plyr::arrange()
## x readr::col_factor()  masks scales::col_factor()
## x gridExtra::combine() masks dplyr::combine()
## x purrr::compact()     masks plyr::compact()
## x dplyr::count()       masks plyr::count()
## x purrr::discard()     masks scales::discard()
## x dplyr::failwith()    masks plyr::failwith()
## x dplyr::filter()      masks stats::filter()
## x dplyr::id()          masks plyr::id()
## x dplyr::lag()         masks stats::lag()
## x dplyr::mutate()      masks plyr::mutate()
## x dplyr::rename()      masks plyr::rename()
## x dplyr::summarise()   masks plyr::summarise()
## x dplyr::summarize()   masks plyr::summarize()

## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

##
## Call:
## glm(formula = Purchase ~ Spending, family = "binomial", data = train.data)
```

```
## 
## Deviance Residuals: 
##        Min          1Q      Median          3Q         Max  
## -9.151e-04  -5.259e-05  -5.259e-05   2.000e-08   1.371e-03  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -20.399    659.346  -0.031    0.975
## Spending       5.713    155.600   0.037    0.971
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 1.1090e+03  on 799  degrees of freedom
## Residual deviance: 3.8307e-06  on 798  degrees of freedom
## AIC: 4
## 
## Number of Fisher Scoring iterations: 25
```

3. Develop a model for predicting spending among the purchasers.

a. Create a vector of ID’s of only purchasers’ records (Purchase = 1).

```
## # A tibble: 5 x 25
##   sequence_number    US source_a source_c source_b source_d source_e source_m
##             <dbl> <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1               1     1        0        0        1        0        0        0
## 2               3     1        0        0        0        0        0        0
## 3               9     1        1        0        0        0        0        0
## 4              10     1        1        0        0        0        0        0
## 5              14     1        1        0        0        0        0        0
## # ... with 17 more variables: source_o <dbl>, source_h <dbl>, source_r <dbl>,
## #   source_s <dbl>, source_t <dbl>, source_u <dbl>, source_p <dbl>,
## #   source_x <dbl>, source_w <dbl>, Freq <dbl>, last_update_days_ago <dbl>,
## #   ‘1st_update_days_ago‘ <dbl>, ‘Web order‘ <dbl>, ‘Gender=male‘ <dbl>,
## #   Address_is_res <dbl>, Purchase <dbl>, Spending <dbl>
```

b. Partition this dataset into the training and validation records. (Use the same training/validation labels from the earlier partitioning; one way is to use function intersect() to find IDs of purchasers in the original partitions).

```
## # A tibble: 5 x 25
##   sequence_number    US source_a source_c source_b source_d source_e source_m
##             <dbl> <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1             496     1        1        0        0        0        0        0
## 2            1672     1        1        0        0        0        0        0
## 3            1829     1        0        0        0        0        0        0
## 4             190     1        0        0        0        0        0        0
## 5            1209     1        0        0        0        0        0        0
## # ... with 17 more variables: source_o <dbl>, source_h <dbl>, source_r <dbl>,
## #   source_s <dbl>, source_t <dbl>, source_u <dbl>, source_p <dbl>,
## #   source_x <dbl>, source_w <dbl>, Freq <dbl>, last_update_days_ago <dbl>,
## #   ‘1st_update_days_ago‘ <dbl>, ‘Web order‘ <dbl>, ‘Gender=male‘ <dbl>,
## #   Address_is_res <dbl>, Purchase <dbl>, Spending <dbl>
```

```
## # A tibble: 5 x 25
##   sequence_number    US source_a source_c source_b source_d source_e source_m
##             <dbl> <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1              15     0        0        0        0        0        0        0
## 2              22     1        0        0        0        0        0        0
## 3              25     1        0        0        1        0        0        0
## 4              30     1        0        0        0        0        0        0
## 5              46     1        1        0        0        0        0        0
## # ... with 17 more variables: source_o <dbl>, source_h <dbl>, source_r <dbl>,
## #   source_s <dbl>, source_t <dbl>, source_u <dbl>, source_p <dbl>,
## #   source_x <dbl>, source_w <dbl>, Freq <dbl>, last_update_days_ago <dbl>,
## #   ‘1st_update_days_ago‘ <dbl>, ‘Web order‘ <dbl>, ‘Gender=male‘ <dbl>,
## #   Address_is_res <dbl>, Purchase <dbl>, Spending <dbl>
```

```
## # A tibble: 5 x 25
##   sequence_number    US source_a source_c source_b source_d source_e source_m
##             <dbl> <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
## 1             289     1        0        0        0        0        0        0
## 2            1720     1        0        0        0        0        0        1
## 3            1106     1        1        0        0        0        0        0
## 4             365     1        1        0        0        0        0        0
## 5              89     0        0        0        0        0        0        0
## # ... with 17 more variables: source_o <dbl>, source_h <dbl>, source_r <dbl>,
## #   source_s <dbl>, source_t <dbl>, source_u <dbl>, source_p <dbl>,
## #   source_x <dbl>, source_w <dbl>, Freq <dbl>, last_update_days_ago <dbl>,
## #   ‘1st_update_days_ago‘ <dbl>, ‘Web order‘ <dbl>, ‘Gender=male‘ <dbl>,
## #   Address_is_res <dbl>, Purchase <dbl>, Spending <dbl>
```

c. Develop models for predicting spending, using:

i. Multiple linear regression (use stepwise regression)

```
##                      Step Df    Deviance Resid. Df Resid. Dev      AIC
## 1                         NA          NA       999   48677227 10794.97
## 2                  + Freq -1 20234715.55       998   28442511 10259.64
## 3        + Address_is_res -1  1131609.96       997   27310901 10221.04
## 4  + last_update_days_ago -1   462419.64       996   26848482 10205.96
## 5              + source_r -1   253458.95       995   26595023 10198.48
## 6              + source_a -1   194825.90       994   26400197 10193.13
## 7              + source_u -1   178600.13       993   26221597 10188.34
## 8 + ‘1st_update_days_ago‘ -1    88550.58       992   26133046 10186.96
```

ii. Regression trees

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

```
# use predict()  to compute predicted probabilities.
reg.tree.pred <- predict(reg.tree, new.valid.data, type = "vector")


table_mat_data <- data.frame(actual = new.valid.data$Spending, predicted = reg.tree.pred)

head(table_mat_data, n=5)
```

```
##   actual predicted
## 1    129        16
## 2    141        16
## 3    285        16
## 4     35        65
## 5    248        16
```

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

d. Choose one model on the basis of its performance on the validation

data

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 Multiple linear regression (use stepwise regression)

Therefore, i select Regression tree as the best fit technique to be used to predict "Spending" as the target variable

4. Return to the original test data partition. Note that this test data partition

includes both purchasers and nonpurchasers. Create a new data frame

called Score Analysis that contains the test data portion of this dataset.

a. Add a column to the data frame with the predicted scores from the

logistic regression.

```
##   sequence_number US source_a source_c source_b source_d source_e source_m
## 1               1  1        0        0        1        0        0        0
## 2               9  1        1        0        0        0        0        0
## 3              12  1        0        0        0        0        0        0
## 4              14  1        1        0        0        0        0        0
## 5              15  0        0        0        0        0        0        0
##   source_o source_h source_r source_s source_t source_u source_p source_x
## 1        0        0        0        0        0        0        0        0
## 2        0        0        0        0        0        0        0        0
## 3        0        1        0        0        0        0        0        0
## 4        0        0        0        0        0        0        0        0
## 5        0        0        0        0        0        0        0        0
##   source_w Freq last_update_days_ago 1st_update_days_ago Web order Gender=male
## 1        0    2                 3662                3662         1           0
## 2        0    4                  525                2914         1           1
## 3        0    2                 1275                1313         0           0
## 4        0    5                 2081                2438         0           1
## 5        1    1                 1465                1465         0           0
##   Address_is_res Purchase Spending predicted$predicted
## 1              1        1      128           152.05335
## 2              0        1      489            57.73835
## 3              1        0        0           162.13419
## 4              0        1     1416           206.63725
## 5              1        1      192           269.68019
```

b. Add another column with the predicted spending amount from the prediction model chosen.

```
##   sequence_number US source_a source_c source_b source_d source_e source_m
## 1               1  1        0        0        1        0        0        0
## 2               9  1        1        0        0        0        0        0
## 3              12  1        0        0        0        0        0        0
## 4              14  1        1        0        0        0        0        0
## 5              15  0        0        0        0        0        0        0
##   source_o source_h source_r source_s source_t source_u source_p source_x
## 1        0        0        0        0        0        0        0        0
## 2        0        0        0        0        0        0        0        0
## 3        0        1        0        0        0        0        0        0
## 4        0        0        0        0        0        0        0        0
## 5        0        0        0        0        0        0        0        0
##   source_w Freq last_update_days_ago 1st_update_days_ago Web order Gender=male
## 1        0    2                 3662                3662         1           0
## 2        0    4                  525                2914         1           1
## 3        0    2                 1275                1313         0           0
## 4        0    5                 2081                2438         0           1
## 5        1    1                 1465                1465         0           0
##   Address_is_res Purchase Spending logistic.pred best.selected.pred
## 1              1        1      128     152.05335                 16
## 2              0        1      489      57.73835                 16
## 3              1        0        0     162.13419                 16
## 4              0        1     1416     206.63725                 65
## 5              1        1      192     269.68019                 16
```

c. Add a column for "adjusted probability of purchase" by multiplying

"predicted probability of purchase" by 0.107. This is to adjust for oversampling the purchasers (see earlier description).

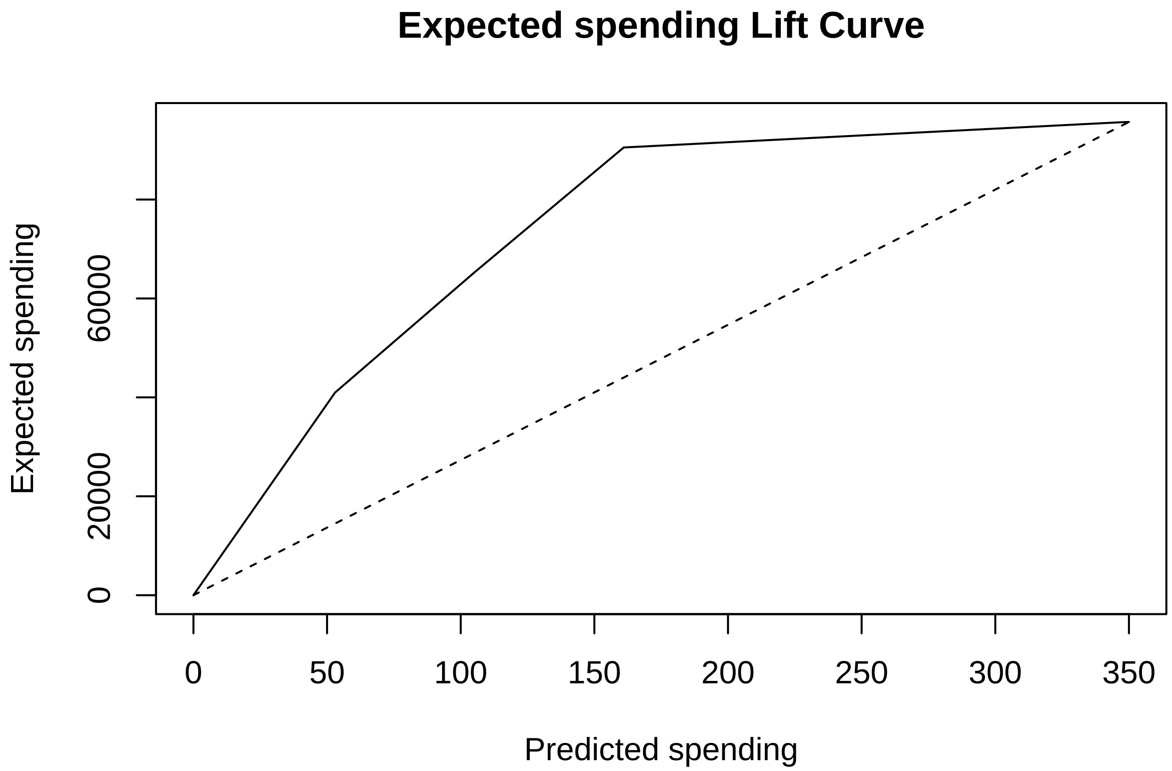
```
##   sequence_number US source_a source_c source_b source_d source_e source_m
## 1               1  1        0        0        1        0        0        0
## 2               9  1        1        0        0        0        0        0
## 3              12  1        0        0        0        0        0        0
## 4              14  1        1        0        0        0        0        0
## 5              15  0        0        0        0        0        0        0
##   source_o source_h source_r source_s source_t source_u source_p source_x
## 1        0        0        0        0        0        0        0        0
## 2        0        0        0        0        0        0        0        0
## 3        0        1        0        0        0        0        0        0
## 4        0        0        0        0        0        0        0        0
## 5        0        0        0        0        0        0        0        0
##   source_w Freq last_update_days_ago 1st_update_days_ago Web order Gender=male
## 1        0    2                 3662                3662         1           0
## 2        0    4                  525                2914         1           1
## 3        0    2                 1275                1313         0           0
## 4        0    5                 2081                2438         0           1
## 5        1    1                 1465                1465         0           0
##   Address_is_res Purchase Spending logistic.pred best.selected.pred
## 1              1        1      128     152.05335                 16
## 2              0        1      489      57.73835                 16
## 3              1        0        0     162.13419                 16
## 4              0        1     1416     206.63725                 65
## 5              1        1      192     269.68019                 16
##   adjusted.column
## 1           1.712
## 2           1.712
## 3           1.712
## 4           6.955
## 5           1.712
```

d. Add a column for expected spending: adjusted probability of purchase

× predicted spending.

```
##   sequence_number US source_a source_c source_b source_d source_e source_m
## 1               1  1        0        0        1        0        0        0
## 2               9  1        1        0        0        0        0        0
## 3              12  1        0        0        0        0        0        0
## 4              14  1        1        0        0        0        0        0
## 5              15  0        0        0        0        0        0        0
##   source_o source_h source_r source_s source_t source_u source_p source_x
## 1        0        0        0        0        0        0        0        0
## 2        0        0        0        0        0        0        0        0
## 3        0        1        0        0        0        0        0        0
## 4        0        0        0        0        0        0        0        0
## 5        0        0        0        0        0        0        0        0
##   source_w Freq last_update_days_ago 1st_update_days_ago Web order Gender=male
## 1        0    2                 3662                3662         1           0
## 2        0    4                  525                2914         1           1
## 3        0    2                 1275                1313         0           0
## 4        0    5                 2081                2438         0           1
## 5        1    1                 1465                1465         0           0
##   Address_is_res Purchase Spending logistic.pred best.selected.pred
## 1              1        1      128     152.05335                 16
## 2              0        1      489      57.73835                 16
## 3              1        0        0     162.13419                 16
## 4              0        1     1416     206.63725                 65
## 5              1        1      192     269.68019                 16
##   adjusted.column Expected.spending
## 1           1.712            27.392
## 2           1.712            27.392
## 3           1.712            27.392
## 4           6.955           452.075
## 5           1.712            27.392
```

e. Plot the lift chart of the expected spending.



f. Using this lift curve, estimate the gross profit that would result from

mailing to the 180,000 names on the basis of your data mining models.

The Lift curve tells us that by picking the 180,000 of names as ranked by the model, we are going to hit four times more positive instances than by selecting a random sample with 180,000 of the names.