BUAN 573: Week 7 Assignment

Wenkkatessh

July 25, 2022

Step 1: Exploratory Data Analysis (EDA)

In step 1, we are going to perform the initial investigations on our datasets so as to discover patterns,to spot anomalies our dataset might have and to check assumptions with the help of summary statistics and graphical representations(visualizations).It is alwasy a good practice to understand the data first and try to gather as many insights from it. EDA is all about making sense of data in hand,before getting them dirty with it.

1. Importing the Fundraising.csv dataset and displaying its structure

```
## 'data.frame':    3120 obs. of  23 variables:
##  $ ï..Row.Id      : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Row.Id.        : int  17 25 29 38 40 53 58 61 71 87 ...
##  $ zipconvert_2   : int  0 1 0 0 0 0 0 1 0 1 ...
##  $ zipconvert_3   : int  1 0 0 0 1 1 0 0 0 0 ...
##  $ zipconvert_4   : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ zipconvert_5   : int  0 0 1 1 0 0 1 0 0 0 ...
##  $ homeowner.dummy: int  1 1 0 1 1 1 1 1 1 1 ...
##  $ NUMCHLD        : int  1 1 2 1 1 1 1 1 1 1 ...
##  $ INCOME         : int  5 1 5 3 4 4 4 1 4 4 ...
##  $ gender.dummy   : int  1 0 1 0 0 1 1 0 0 1 ...
##  $ WEALTH         : int  9 7 8 4 8 8 8 7 5 8 ...
##  $ HV             : int  1399 698 828 1471 547 482 857 1355 505 1438 ...
##  $ Icmed          : int  637 422 358 484 386 242 450 411 333 458 ...
##  $ Icavg          : int  703 463 376 546 432 275 498 497 388 533 ...
##  $ IC15           : int  1 4 13 4 7 28 5 9 16 8 ...
##  $ NUMPROM        : int  74 46 32 94 20 38 47 77 51 21 ...
##  $ RAMNTALL       : num  102 94 30 177 23 73 139 249 63 26 ...
##  $ MAXRAMNT       : num  6 12 10 10 11 10 20 15 15 16 ...
##  $ LASTGIFT       : num  5 12 5 8 11 10 20 7 10 16 ...
##  $ totalmonths    : int  29 34 29 30 30 31 37 35 37 30 ...
##  $ TIMELAG        : int  3 6 7 3 6 3 3 3 8 6 ...
##  $ AVGGIFT        : num  4.86 9.4 4.29 7.08 7.67 ...
##  $ TARGET_B       : int  1 1 1 0 0 1 1 1 1 0 ...
```

1. Importing the FutureFundraising.csv dataset and displaying its structure

```
## 'data.frame':    3120 obs. of  24 variables:
```

```
##  $ ï..Row.Id      : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Row.Id.        : int  17 25 29 38 40 53 58 61 71 87 ...
##  $ zipconvert_2   : int  0 1 0 0 0 0 0 1 0 1 ...
##  $ zipconvert_3   : int  1 0 0 0 1 1 0 0 0 0 ...
##  $ zipconvert_4   : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ zipconvert_5   : int  0 0 1 1 0 0 1 0 0 0 ...
##  $ homeowner.dummy: int  1 1 0 1 1 1 1 1 1 1 ...
##  $ NUMCHLD        : int  1 1 2 1 1 1 1 1 1 1 ...
##  $ INCOME         : int  5 1 5 3 4 4 4 1 4 4 ...
##  $ gender.dummy   : int  1 0 1 0 0 1 1 0 0 1 ...
##  $ WEALTH         : int  9 7 8 4 8 8 8 7 5 8 ...
##  $ HV             : int  1399 698 828 1471 547 482 857 1355 505 1438 ...
##  $ Icmed          : int  637 422 358 484 386 242 450 411 333 458 ...
##  $ Icavg          : int  703 463 376 546 432 275 498 497 388 533 ...
##  $ IC15           : int  1 4 13 4 7 28 5 9 16 8 ...
##  $ NUMPROM        : int  74 46 32 94 20 38 47 77 51 21 ...
##  $ RAMNTALL       : num  102 94 30 177 23 73 139 249 63 26 ...
##  $ MAXRAMNT       : num  6 12 10 10 11 10 20 15 15 16 ...
##  $ LASTGIFT       : num  5 12 5 8 11 10 20 7 10 16 ...
##  $ totalmonths    : int  29 34 29 30 30 31 37 35 37 30 ...
##  $ TIMELAG        : int  3 6 7 3 6 3 3 3 8 6 ...
##  $ AVGGIFT        : num  4.86 9.4 4.29 7.08 7.67 ...
##  $ TARGET_B       : int  1 1 1 0 0 1 1 1 1 0 ...
##  $ TARGET_D       : num  5 10 5 0 0 8 10 20 5 0 ...
```

2.Below is the dimension of the Fundraising.csv dataset

```
## [1] 3120   23
```

3.Displaying the Descriptive Statistics of our dataset

```
## new.fund.df
##
##  23  Variables      3120  Observations
## ---------------------------------------------------------------------------
## ï..Row.Id
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0     3120        1     1560     1040    157.0    312.9
##      .25      .50      .75      .90      .95
##    780.8   1560.5   2340.2   2808.1   2964.0
##
## lowest :    1    2    3    4    5, highest: 3116 3117 3118 3119 3120
## ---------------------------------------------------------------------------
## Row.Id.
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0     3120        1    11616     7736     1181     2254
##      .25      .50      .75      .90      .95
##     5821    11736    17436    20727    22089
##
## lowest :    17    25    29    38    40, highest: 23256 23258 23261 23265 23293
## ---------------------------------------------------------------------------
## zipconvert_2
```

```
##        n  missing distinct     Info      Sum     Mean      Gmd
##     3120        0        2    0.505      669   0.2144    0.337
##
## ---------------------------------------------------------------------------
## zipconvert_3
##        n  missing distinct     Info      Sum     Mean      Gmd
##     3120        0        2    0.453      578   0.1853    0.302
##
## ---------------------------------------------------------------------------
## zipconvert_4
##        n  missing distinct     Info      Sum     Mean      Gmd
##     3120        0        2    0.505      669   0.2144    0.337
##
## ---------------------------------------------------------------------------
## zipconvert_5
##        n  missing distinct     Info      Sum     Mean      Gmd
##     3120        0        2     0.71     1200   0.3846   0.4735
##
## ---------------------------------------------------------------------------
## homeowner.dummy
##        n  missing distinct     Info      Sum     Mean      Gmd
##     3120        0        2    0.531     2403   0.7702   0.3541
##
## ---------------------------------------------------------------------------
## NUMCHLD
##        n  missing distinct     Info     Mean      Gmd
##     3120        0        5    0.136    1.069   0.1334
##
## lowest : 1 2 3 4 5, highest: 1 2 3 4 5
##
## Value          1     2     3     4     5
## Frequency   2972    99    31    17     1
## Proportion 0.953 0.032 0.010 0.005 0.000
## ---------------------------------------------------------------------------
## INCOME
##        n  missing distinct     Info     Mean      Gmd
##     3120        0        7    0.951    3.894    1.821
##
## lowest : 1 2 3 4 5, highest: 3 4 5 6 7
##
## Value          1     2     3     4     5     6     7
## Frequency    282   468   296  1053   535   246   240
## Proportion 0.090 0.150 0.095 0.338 0.171 0.079 0.077
## ---------------------------------------------------------------------------
## gender.dummy
##        n  missing distinct     Info      Sum     Mean      Gmd
##     3120        0        2    0.714     1901   0.6093   0.4763
##
## ---------------------------------------------------------------------------
## WEALTH
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0       10    0.837    6.402    2.521        1        2
##      .25      .50      .75      .90      .95
##        5        8        8        8        9
```

```
##
## lowest : 0 1 2 3 4, highest: 5 6 7 8 9
##
## Value          0     1     2     3     4     5     6     7     8     9
## Frequency    112   138   138   162   153   186   162   180  1700   189
## Proportion 0.036 0.044 0.044 0.052 0.049 0.060 0.052 0.058 0.545 0.061
## ---------------------------------------------------------------------------------
## HV
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0     1552        1     1141    893.9    343.0    413.9
##      .25      .50      .75      .90      .95
##    556.0    822.0   1338.8   2357.4   3138.1
##
## lowest :    0  163  171  200  205, highest: 5888 5908 5926 5932 5945
## ---------------------------------------------------------------------------------
## Icmed
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0      654        1    388.2    176.7    188.0    220.0
##      .25      .50      .75      .90      .95
##    278.0    356.0    465.0    591.1    684.0
##
## lowest :    0   68   71   72   77, highest: 1340 1434 1469 1496 1500
## ---------------------------------------------------------------------------------
## Icavg
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0      674        1    432.1    178.8      232      264
##      .25      .50      .75      .90      .95
##      318      396      516      646      761
##
## lowest :    0   89   90   94  121, highest: 1217 1228 1236 1273 1331
## ---------------------------------------------------------------------------------
## IC15
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0       69    0.999     14.7    12.93        0        2
##      .25      .50      .75      .90      .95
##        5       12       21       30       39
##
## lowest :  0  1  2  3  4, highest: 68 69 75 85 90
## ---------------------------------------------------------------------------------
## NUMPROM
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0      125        1    49.09    25.42       19       22
##      .25      .50      .75      .90      .95
##       29       48       65       78       86
##
## lowest :  11  12  13  14  15, highest: 140 141 144 147 157
## ---------------------------------------------------------------------------------
## RAMNTALL
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0      423        1    110.4    95.15     25.0     30.0
##      .25      .50      .75      .90      .95
##     45.0     81.0    134.6    214.0    282.0
##
## lowest :   15.0   16.0   18.0   19.0   20.0, highest: 1111.0 1174.0 1622.0 2200.0 5674.9
```

```
## ---------------------------------------------------------------------------
## MAXRAMNT
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0       57    0.988    16.65    10.64        6        7
##      .25      .50      .75      .90      .95
##       10       15       20       25       30
##
## lowest :    5    6    7    8    9, highest:  140  175  250  375 1000
## ---------------------------------------------------------------------------
## LASTGIFT
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0       53    0.988    13.52     9.16        4        5
##      .25      .50      .75      .90      .95
##        7       10       16       25       25
##
## lowest :   0.0   1.0   2.0   2.5   3.0, highest:  80.0  90.0 100.0 125.0 219.0
## ---------------------------------------------------------------------------
## totalmonths
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0       21    0.986    31.14    4.317       23       28
##      .25      .50      .75      .90      .95
##       29       31       34       37       37
##
## lowest : 17 18 19 20 21, highest: 33 34 35 36 37
## ---------------------------------------------------------------------------
## TIMELAG
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0       43    0.991    6.862     5.32        1        2
##      .25      .50      .75      .90      .95
##        3        5        9       13       17
##
## lowest :  0  1  2  3  4, highest: 38 44 48 62 77
## ---------------------------------------------------------------------------
## AVGGIFT
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     3120        0     1298        1    10.69    6.559    3.938    4.699
##      .25      .50      .75      .90      .95
##    6.356    9.000   12.812   18.333   22.500
##
## lowest :   2.138889   2.260870   2.354839   2.439815   2.445946
## highest:  77.571429  80.000000  85.000000 100.000000 122.166667
## ---------------------------------------------------------------------------
## TARGET_B
##        n  missing distinct     Info      Sum     Mean      Gmd
##     3120        0        2     0.75     1560      0.5   0.5002
##
## ---------------------------------------------------------------------------
```

2.Clean the dataset

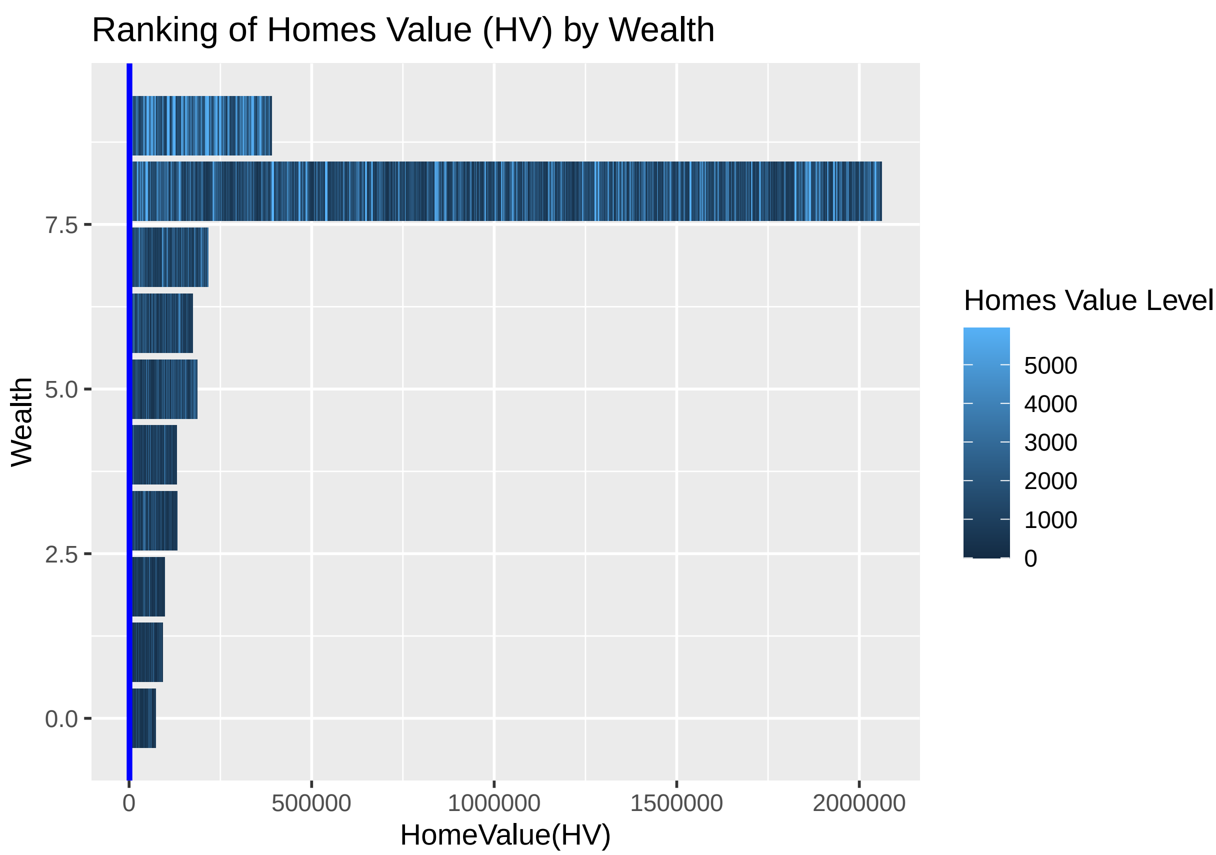
We clean the dataset by removing the TARGET_D which will not be used in our case

Covert categorical variables into factor data type

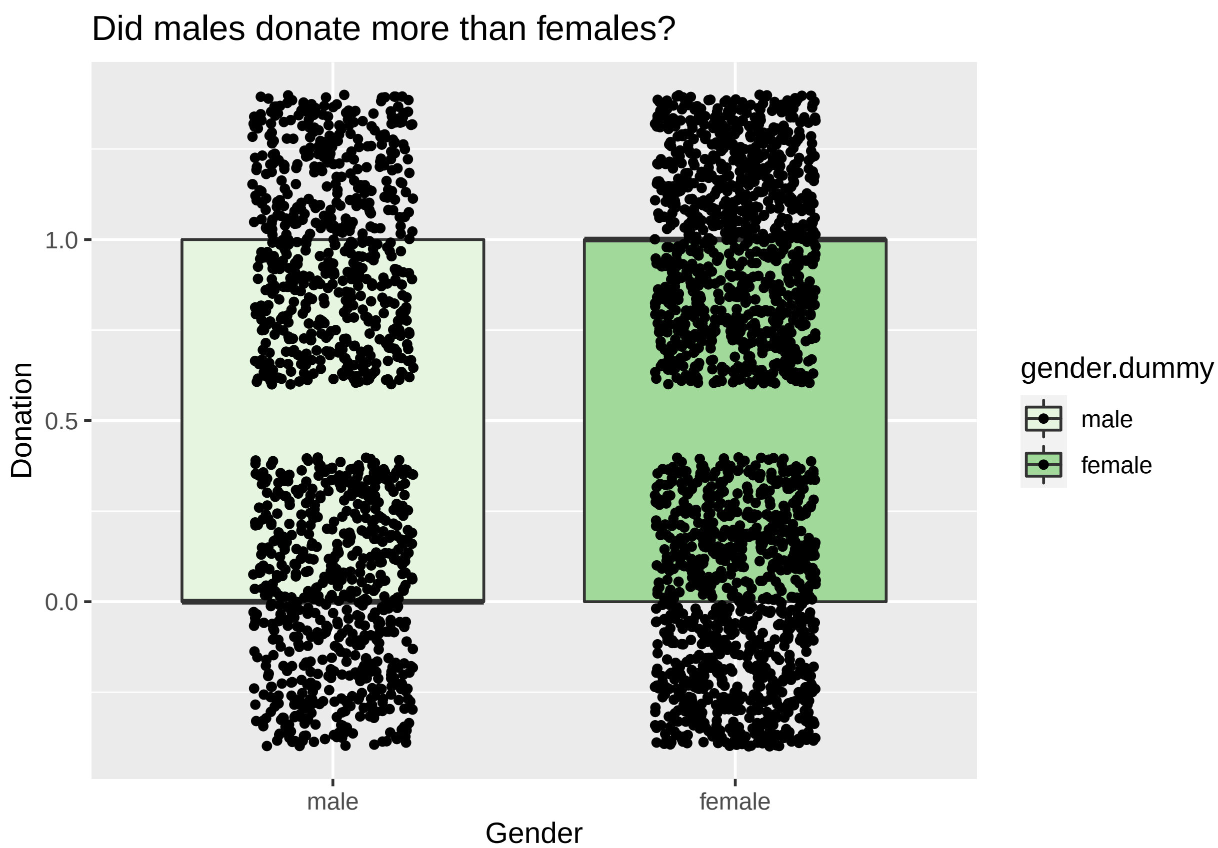
Visualizing the dataset

This is achieved by exploring some of the important variables

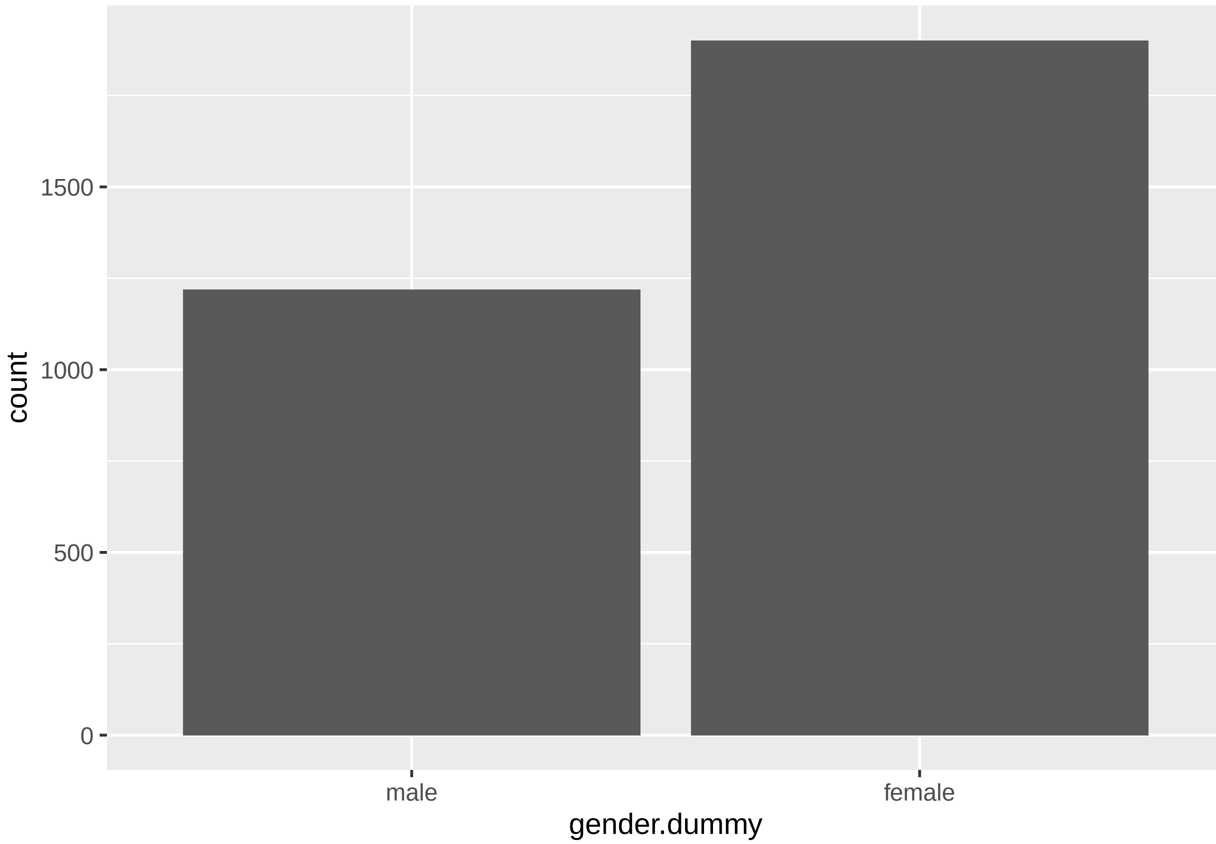
Barplot Ranking of Homes Value (HV) by Wealth



Boxplot

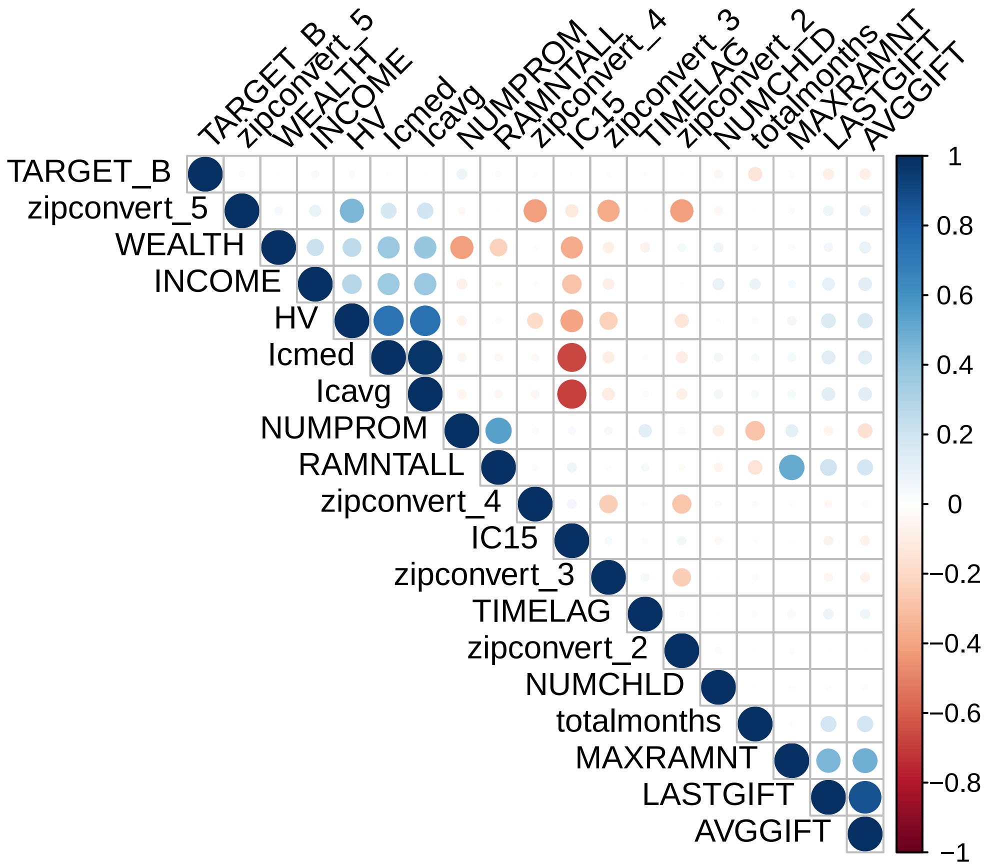


Barplot Ranking Gender by frequency


1500
1000
500
0
count
male
female
gender.dummy

Correlation

Correlation Plot



Step 2: Methodology(Data Mining Techniques Used)

In our case, we have been instructed to use three models in implementation of our predictive models. The 3 selected models were: Logistic regression, Classification tree and Neural Networks. Among these 3 models, we are to find one as the best model and perform testing using it.

Logistic Regression - we have used this model since it will help in estimating the probability of an individulas to donate or not to donate based on our given fundraising dataset of independent variables. The dependent variable in our case is TARGET_B and is bounded between 0 and 1.

Classification tree - A classification tree identifies what combination of our dataset factors best differentiates between individuals(donors/not donors) based on our categorical variable of interest which is (TARGET_B)

Neural Networks - is a technique applied in our dataset in order to find hidden patterns in our fundraising dataset.

Step 3: Use different models

STEP 1:Partitioning

```
# use set.seed() to get the same partitions when re-running the R code.
set.seed(12345)
## partitioning into training (60%) and validation (40%)
# randomly sample 60% of the row IDs for training; the remaining 40% serve as
# validation
train.rows <- sample(rownames(new.fund.df), dim(new.fund.df)[1]*0.6)
valid.rows <- sample(setdiff(rownames(new.fund.df), train.rows),dim(new.fund.df)[1]*0.2)
# assign the remaining 20% row IDs serve as test
test.rows <- setdiff(rownames(new.fund.df), union(train.rows, valid.rows))
# create the 3 data frames by collecting all columns from the appropriate rows
train.data <- new.fund.df[train.rows, ]
valid.data <- new.fund.df[valid.rows, ]
test.data <- new.fund.df[test.rows, ]
#
#print the train data
head(train.data, n=5)
```

```
##      ï..Row.Id Row.Id. zipconvert_2 zipconvert_3 zipconvert_4 zipconvert_5
## 2250      2250   16725            0            0            1            0
## 2732      2732   20288            0            0            0            1
## 2373      2373   17667            0            0            0            1
## 2763      2763   20451            0            1            0            0
## 1423      1423   10709            0            0            1            0
##      homeowner.dummy NUMCHLD INCOME gender.dummy WEALTH   HV Icmed Icavg IC15
## 2250               0       1      3            1      8 4345   285   378   26
## 2732               1       1      5            1      3  795   357   391   11
```

```
## 2373                0       1      5            0      8  749    366   415   15
## 2763                1       1      5            1      8  770    511   542    4
## 1423                0       1      1            0      1  324    258   299   24
##      NUMPROM RAMNTALL MAXRAMNT LASTGIFT totalmonths TIMELAG   AVGGIFT TARGET_B
## 2250      25       40       10       10          28       4 10.000000        1
## 2732     117      521       20        7          18       6 10.215686        0
## 2373      31       77       25       17          37       2 19.250000        0
## 2763      58       85       10       10          30       8  7.727273        0
## 1423      68       71       10        2          32       6  4.437500        1
```

```
#print the validation data
head(valid.data, n=5)
```

```
##      ï..Row.Id Row.Id. zipconvert_2 zipconvert_3 zipconvert_4 zipconvert_5
## 119        119     839            0            0            1            0
## 1548      1548   11609            0            0            0            1
## 429        429    3141            0            0            0            1
## 1929      1929   14370            0            0            1            0
## 1157      1157    8642            0            1            0            0
##      homeowner.dummy NUMCHLD INCOME gender.dummy WEALTH   HV Icmed Icavg IC15
## 119                0       1      3            0      5  622   339   377   13
## 1548               1       1      4            1      8  241   163   149   44
## 429                1       1      5            1      8 1611   318   351    7
## 1929               1       1      1            1      1  336   256   285   23
## 1157               1       1      5            1      8  599   551   540    2
##      NUMPROM RAMNTALL MAXRAMNT LASTGIFT totalmonths TIMELAG   AVGGIFT TARGET_B
## 119       87    241.0       17       15          23       5 13.388889        1
## 1548      39     92.5       10        3          32       0  3.425926        1
## 429       31     29.0       10        5          28      12  5.800000        1
## 1929      83    244.0        9        8          32       0  6.256410        0
## 1157      56    172.0       20       20          30       5 13.230769        1
```

```
#print the test data
head(test.data, n=5)
```

```
##    ï..Row.Id Row.Id. zipconvert_2 zipconvert_3 zipconvert_4 zipconvert_5
## 3          3      29            0            0            0            1
## 6          6      53            0            1            0            0
## 8          8      61            1            0            0            0
## 9          9      71            0            0            1            0
## 10        10      87            1            0            0            0
##    homeowner.dummy NUMCHLD INCOME gender.dummy WEALTH   HV Icmed Icavg IC15
## 3                0       2      5            1      8  828   358   376   13
## 6                1       1      4            1      8  482   242   275   28
## 8                1       1      1            0      7 1355   411   497    9
## 9                1       1      4            0      5  505   333   388   16
## 10               1       1      4            1      8 1438   458   533    8
##    NUMPROM RAMNTALL MAXRAMNT LASTGIFT totalmonths TIMELAG   AVGGIFT TARGET_B
## 3       32       30       10        5          29       7  4.285714        1
## 6       38       73       10       10          31       3  7.300000        1
## 8       77      249       15        7          35       3  9.576923        1
## 9       51       63       15       10          37       8  9.000000        1
## 10      21       26       16       16          30       6 13.000000        0
```

STEP 2:Model Building

Classification under asymmetric response and cost

Weighted sampling allow us to reconfigure the sample as if it was a simple random draw of the whole dataset, and hence yield accurate dataset estimates for the main parameters of interest than when compared to the random sampling

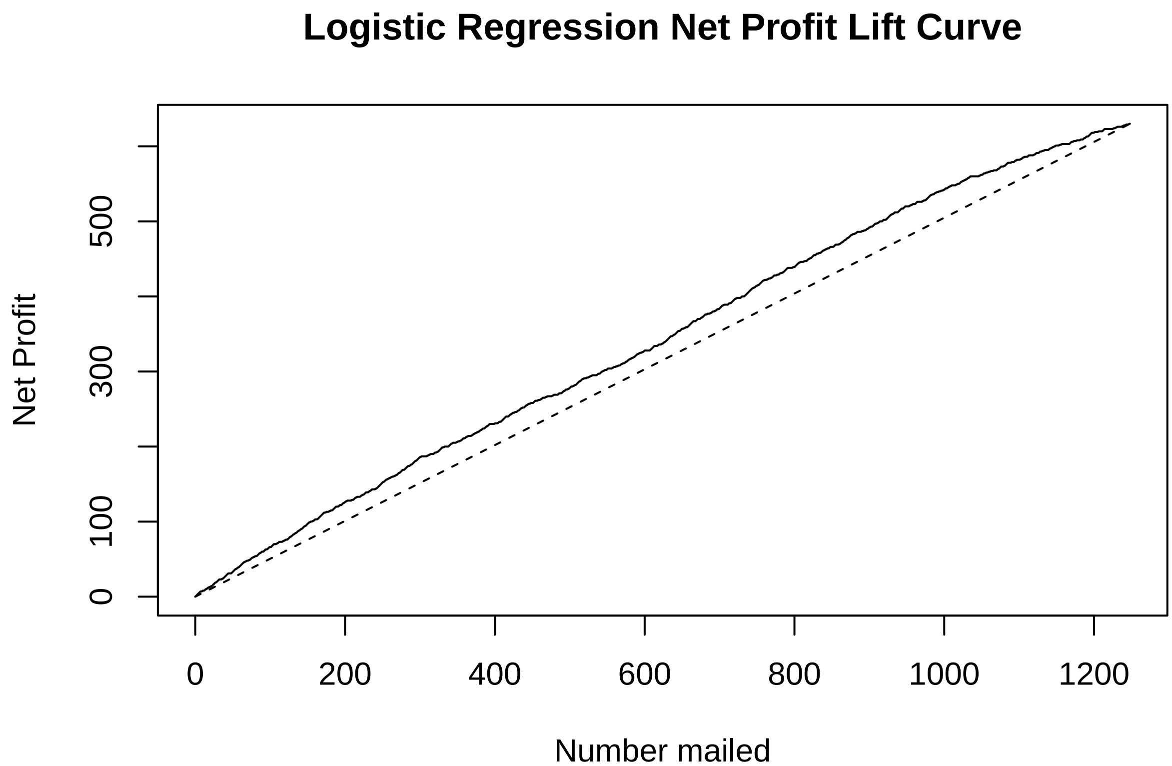
Classification tools, Net Profit and Lift curves for each model

Logistic regression

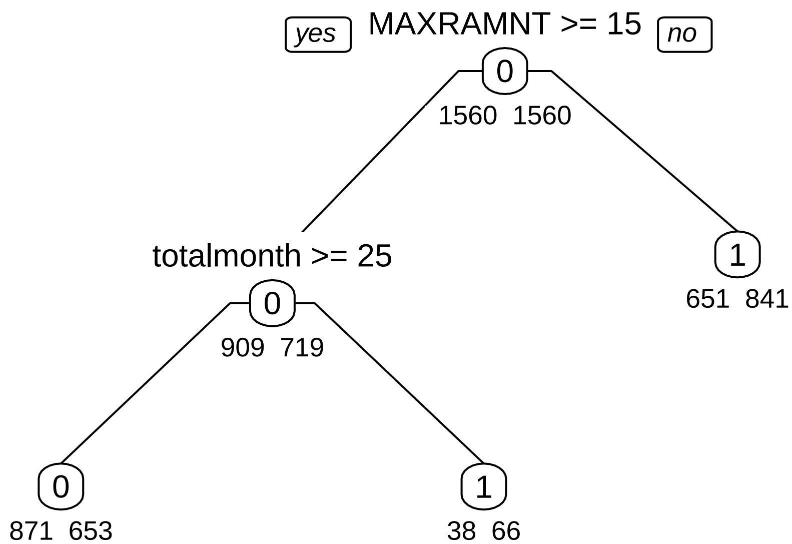
```
##
## Call:
## glm(formula = TARGET_B ~ ., family = "binomial", data = train.data)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.791  -1.155  -0.729   1.154   2.119
##
## Coefficients:
##                    Estimate   Std. Error z value  Pr(>|z|)
## (Intercept)     -12.08644309 307.82425217  -0.039    0.9687
## zipconvert_2     13.62807301 307.82386988   0.044    0.9647
## zipconvert_3     13.58267852 307.82387403   0.044    0.9648
## zipconvert_4     13.44195213 307.82387393   0.044    0.9652
## zipconvert_5     13.57873808 307.82384721   0.044    0.9648
## homeowner.dummy   0.07722964   0.11921007   0.648    0.5171
## NUMCHLD          -0.28565556   0.13761760  -2.076    0.0379 *
## INCOME            0.07942684   0.03275478   2.425    0.0153 *
## gender.dummy      0.07535469   0.09755440   0.772    0.4399
## WEALTH            0.01666131   0.02259012   0.738    0.4608
## HV                0.00012004   0.00008892   1.350    0.1770
## Icmed             0.00064840   0.00117175   0.553    0.5800
## Icavg            -0.00103438   0.00127434  -0.812    0.4170
## IC15              0.00253430   0.00560629   0.452    0.6512
## NUMPROM           0.00504683   0.00334015   1.511    0.1308
## RAMNTALL         -0.00033957   0.00068839  -0.493    0.6218
## MAXRAMNT          0.00470886   0.00813143   0.579    0.5625
## LASTGIFT         -0.02271811   0.01112948  -2.041    0.0412 *
## totalmonths      -0.05668007   0.01288633  -4.398 0.0000109 ***
## TIMELAG           0.00598617   0.00876821   0.683    0.4948
## AVGGIFT           0.00638631   0.01565310   0.408    0.6833
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2595.1  on 1871  degrees of freedom
## Residual deviance: 2526.2  on 1851  degrees of freedom
## AIC: 2568.2
##
## Number of Fisher Scoring iterations: 12
```

```
##   actual predicted
## 1      1 0.6047590
## 3      1 0.5031665
## 6      1 0.5274315
## 8      1 0.4447540
## 9      1 0.3897165

## [1] "Accuracy of Logistic Regression is 0.0016025641025641"
```



Classification Trees



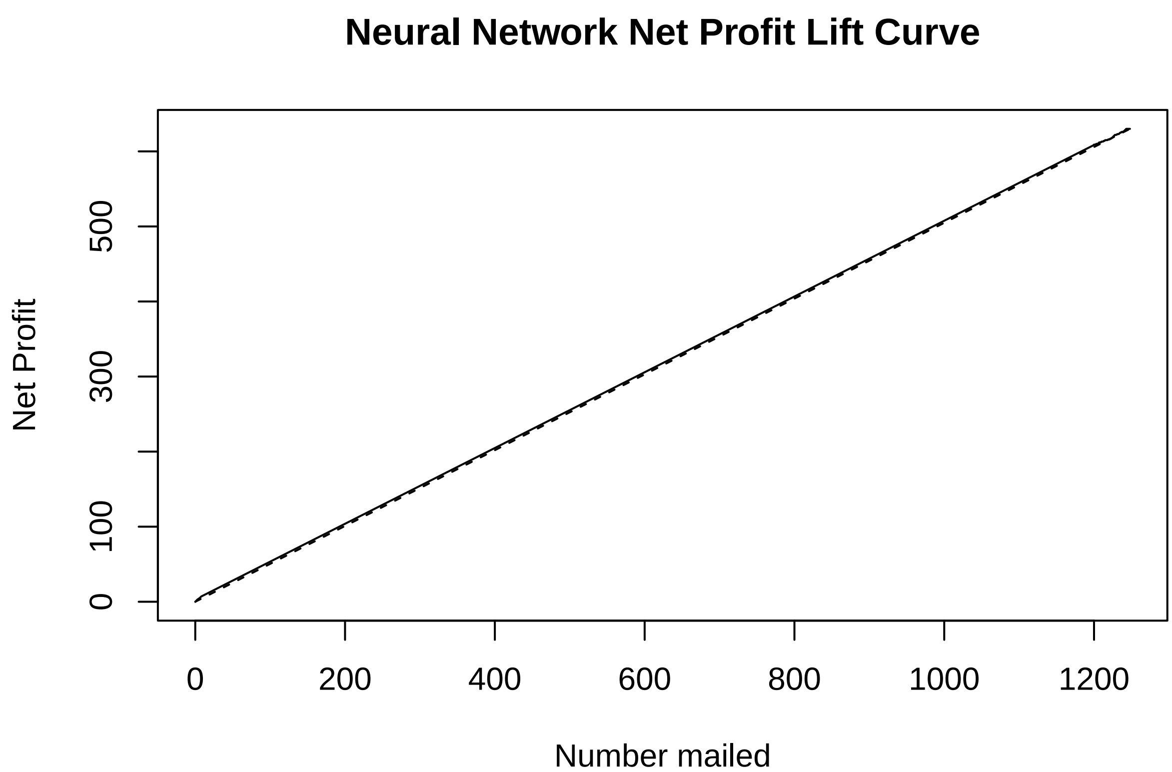

```
##   actual predicted
## 1      1         1
## 3      1         1
## 6      1         1
## 8      1         0
## 9      1         0
```

```
## [1] "Accuracy of Classification tree is 0.580929487179487"
```

Neural Networks model

```
## [1] "Accuracy of Neural Networks is 0.000801282051282051"
```

```
##   actual predicted
## 1      1 0.5011052
## 2      1 0.5011052
## 3      1 0.5011052
## 4      1 0.5011052
## 5      1 0.5011052
```



Selected model - Classification tree

Classification tree is the best model because of its high accuracy on our dataset when compared with logistic and Neural Networks predictive models.

STEP 3: Testing

```
##      testing_mat.Row.Id. testing_mat.TARGET_B
## 3120               23293                    0
## 3119               23265                    0
## 3118               23261                    0
## 3117               23258                    1
## 3116               23256                    0
```

Number (6) - In this step, we are supposed to find the probability of people donating using the Futurefundraising dataset and thereafter arrange the results in decsinding order and make conlcusions frtom what we are seeing. From the testing results, when you keenly analyse the results from the display, its clear that most people were not willing to donate. Therefore I would'nt go on with the mailing campaign since most people were not willing to donate