Capstone Project

Group 5

2023-06-11

#Loading Required Packages

```
rm(list = ls()) #cleaning the environment
library(readr)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v dplyr   1.0.10
## v tibble  3.1.8      v stringr 1.5.0
## v tidyr   1.3.0      v forcats 0.5.2
## v purrr   1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
## Warning: package 'lattice' was built under R version 4.2.3
```

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

```
library(knitr)
library(class)
library(ggplot2)
library(ggcorrplot)
library(dplyr)
library(e1071)
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
##
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
library(caret)
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```
library(cluster)
library(cowplot)
library(pander)
library(kernlab)
```

```
##
## Attaching package: 'kernlab'
##
## The following object is masked from 'package:purrr':
##
##     cross
##
## The following object is masked from 'package:ggplot2':
##
##     alpha
```

```
library(tidyr)
library(fastDummies)
```

```
## Warning: package 'fastDummies' was built under R version 4.2.3
```

```
library(FactoMineR)
library(ROCR)
```

```
## Warning: package 'ROCR' was built under R version 4.2.3
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
##
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
library(rpart)
library(cutpointr)
```

```
## Warning: package 'cutpointr' was built under R version 4.2.3
```

```
##
## Attaching package: 'cutpointr'
##
```

```
## The following objects are masked from 'package:pROC':
##
##     auc, roc
##
## The following objects are masked from 'package:caret':
##
##     precision, recall, sensitivity, specificity
```

```
library(ROSE)
```

```
## Warning: package 'ROSE' was built under R version 4.2.3
```

```
## Loaded ROSE 0.0-4
```

```
library(writexl)
```

```
## Warning: package 'writexl' was built under R version 4.2.3
```

```
library(mice)
```

```
## Warning: package 'mice' was built under R version 4.2.3
```

```
##
## Attaching package: 'mice'
##
## The following object is masked from 'package:kernlab':
##
##     convergence
##
## The following object is masked from 'package:stats':
##
##     filter
##
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
library(VIM)
```

```
## Warning: package 'VIM' was built under R version 4.2.3
```

```
## Loading required package: colorspace
##
## Attaching package: 'colorspace'
##
## The following object is masked from 'package:pROC':
##
##     coords
##
## Loading required package: grid
```

```
## VIM is ready to use.
##
## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues
##
## Attaching package: 'VIM'
##
## The following object is masked from 'package:datasets':
##
##     sleep
```

```
library(lattice)
library(gmodels)
```

```
##
## Attaching package: 'gmodels'
##
## The following object is masked from 'package:pROC':
##
##     ci
```

```
library(rpart)
library(rpart.plot)
library(e1071)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.2.3
```

```
##
## Attaching package: 'psych'
##
## The following object is masked from 'package:kernlab':
##
##     alpha
##
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

#Importing the Car's bad buy , good buy dataset.

```
setwd("C:/Users/Chaur/OneDrive/Desktop/Capstone_Project")  #set working directory
Car_data <- read.csv("car_kick.csv")  #load the data
head(Car_data)
```

```
##    PurchDate VehYear VehicleAge VehOdo MMRAcquisitionAuctionAveragePrice
## 1 1289952000    2006          4  51954                              6197
```

```
## 2 1242691200    2005          4  89127                                3688
## 3 1248220800    2006          3  71271                                6897
## 4 1285718400    2008          2  83338                                7878
## 5 1237334400    2007          2  58698                                8800
## 6 1292198400    2006          4  77096                                5246
##   MMRAcquisitionAuctionCleanPrice MMRAcquisitionRetailAveragePrice
## 1                            7062                              9605
## 2                            4783                              4483
## 3                            8449                              7949
## 4                            8925                             11723
## 5                           10091                             10004
## 6                            6141                              8760
##   MMRAcquisitonRetailCleanPrice MMRCurrentAuctionAveragePrice
## 1                         10426                          5341
## 2                          5666                          3688
## 3                          9625                          6868
## 4                         13026                          7801
## 5                         11398                          7355
## 6                          9817                          4260
##   MMRCurrentAuctionCleanPrice MMRCurrentRetailAveragePrice
## 1                        6351                         8513
## 2                        4783                         4483
## 3                        8549                         7917
## 4                        8704                        11995
## 5                        8543                         8443
## 6                        5006                         6906
##   MMRCurrentRetailCleanPrice VehBCost WarrantyCost Auction      Make
## 1                       9822     6500         1086 MANHEIM     DODGE
## 2                       5666     3680          983   ADESA      FORD
## 3                       9733     7170         1974 MANHEIM   PONTIAC
## 4                      12901     7670         2152 MANHEIM CHEVROLET
## 5                       9726     7165         1500   ADESA CHEVROLET
## 6                       8352     5500          569 MANHEIM       KIA
##                    Model Trim              SubModel  Color Transmission
## 1          'STRATUS V6'  SXT     '4D SEDAN SXT FFV' SILVER         AUTO
## 2   'TAURUS 3.0L V6 EFI'   SE          '4D SEDAN SE' SILVER         AUTO
## 3 'GRAND PRIX 3.8L V6 S'  Bas             '4D SEDAN'    RED         AUTO
## 4           'IMPALA V6'   LS '4D SEDAN LS 3.5L FFV'  BLACK         AUTO
## 5                IMPALA   LT     '4D SEDAN LT 3.5L'  WHITE         AUTO
## 6               SPECTRA   EX             '4D SEDAN' SILVER         AUTO
##   WheelTypeID WheelType   Nationality   Size TopThreeAmericanName BYRNO VNZIP1
## 1           2    Covers      AMERICAN MEDIUM             CHRYSLER 99750  32124
## 2           2    Covers      AMERICAN MEDIUM                 FORD 20833  78754
## 3           1     Alloy      AMERICAN  LARGE                   GM 22916  80011
## 4           2    Covers      AMERICAN  LARGE                   GM 23657  94544
## 5           1     Alloy      AMERICAN  LARGE                   GM 20833  77086
## 6           1     Alloy 'OTHER ASIAN' MEDIUM                OTHER 52117  27542
##   VNST IsOnlineSale Class
## 1   FL            0     0
## 2   TX            0     0
## 3   CO            0     0
## 4   CA            0     0
## 5   TX            0     0
## 6   NC            0     0
```

```
dim(Car_data)
```

```
## [1] 67211    31
```

#Understanging the structure and summary of the Training dataset, Also Figuring out the Missing Values

```
str(Car_data)#To see the structure of data set
```

```
## 'data.frame':    67211 obs. of  31 variables:
##  $ PurchDate                         : num  1.29e+09 1.24e+09 1.25e+09 1.29e+09 1.24e+09 ...
##  $ VehYear                           : num  2006 2005 2006 2008 2007 ...
##  $ VehicleAge                        : int  4 4 3 2 2 4 4 4 5 4 ...
##  $ VehOdo                            : num  51954 89127 71271 83338 58698 ...
##  $ MMRAcquisitionAuctionAveragePrice: num  6197 3688 6897 7878 8800 ...
##  $ MMRAcquisitionAuctionCleanPrice  : num  7062 4783 8449 8925 10091 ...
##  $ MMRAcquisitionRetailAveragePrice : num  9605 4483 7949 11723 10004 ...
##  $ MMRAcquisitonRetailCleanPrice    : num  10426 5666 9625 13026 11398 ...
##  $ MMRCurrentAuctionAveragePrice    : num  5341 3688 6868 7801 7355 ...
##  $ MMRCurrentAuctionCleanPrice      : num  6351 4783 8549 8704 8543 ...
##  $ MMRCurrentRetailAveragePrice     : num  8513 4483 7917 11995 8443 ...
##  $ MMRCurrentRetailCleanPrice       : num  9822 5666 9733 12901 9726 ...
##  $ VehBCost                          : num  6500 3680 7170 7670 7165 ...
##  $ WarrantyCost                      : num  1086 983 1974 2152 1500 ...
##  $ Auction                           : chr  "MANHEIM" "ADESA" "MANHEIM" "MANHEIM" ...
##  $ Make                              : chr  "DODGE" "FORD" "PONTIAC" "CHEVROLET" ...
##  $ Model                             : chr  "'STRATUS V6'" "'TAURUS 3.0L V6 EFI'" "'GRAND PRIX 3.8L V
##  $ Trim                              : chr  "SXT" "SE" "Bas" "LS" ...
##  $ SubModel                          : chr  "'4D SEDAN SXT FFV'" "'4D SEDAN SE'" "'4D SEDAN'" "'4D SE
##  $ Color                             : chr  "SILVER" "SILVER" "RED" "BLACK" ...
##  $ Transmission                      : chr  "AUTO" "AUTO" "AUTO" "AUTO" ...
##  $ WheelTypeID                       : num  2 2 1 2 1 1 1 2 1 2 ...
##  $ WheelType                         : chr  "Covers" "Covers" "Alloy" "Covers" ...
##  $ Nationality                       : chr  "AMERICAN" "AMERICAN" "AMERICAN" "AMERICAN" ...
##  $ Size                              : chr  "MEDIUM" "MEDIUM" "LARGE" "LARGE" ...
##  $ TopThreeAmericanName              : chr  "CHRYSLER" "FORD" "GM" "GM" ...
##  $ BYRNO                             : int  99750 20833 22916 23657 20833 52117 20234 22808 21973 817
##  $ VNZIP1                            : int  32124 78754 80011 94544 77086 27542 92337 78754 33314 372
##  $ VNST                              : chr  "FL" "TX" "CO" "CA" ...
##  $ IsOnlineSale                      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Class                             : int  0 0 0 0 0 0 0 0 0 0 ...
```

```
summary(Car_data)#To see the summary (min, max and mean, median of the variable)
```

```
##    PurchDate            VehYear       VehicleAge        VehOdo
##  Min.   :1.231e+09   Min.   :2001   Min.   :0.000   Min.   :  5368
##  1st Qu.:1.248e+09   1st Qu.:2004   1st Qu.:3.000   1st Qu.: 62163
##  Median :1.264e+09   Median :2005   Median :4.000   Median : 73530
##  Mean   :1.263e+09   Mean   :2005   Mean   :4.169   Mean   : 71735
##  3rd Qu.:1.279e+09   3rd Qu.:2007   3rd Qu.:5.000   3rd Qu.: 82530
##  Max.   :1.294e+09   Max.   :2010   Max.   :9.000   Max.   :115717
##  MMRAcquisitionAuctionAveragePrice MMRAcquisitionAuctionCleanPrice
```

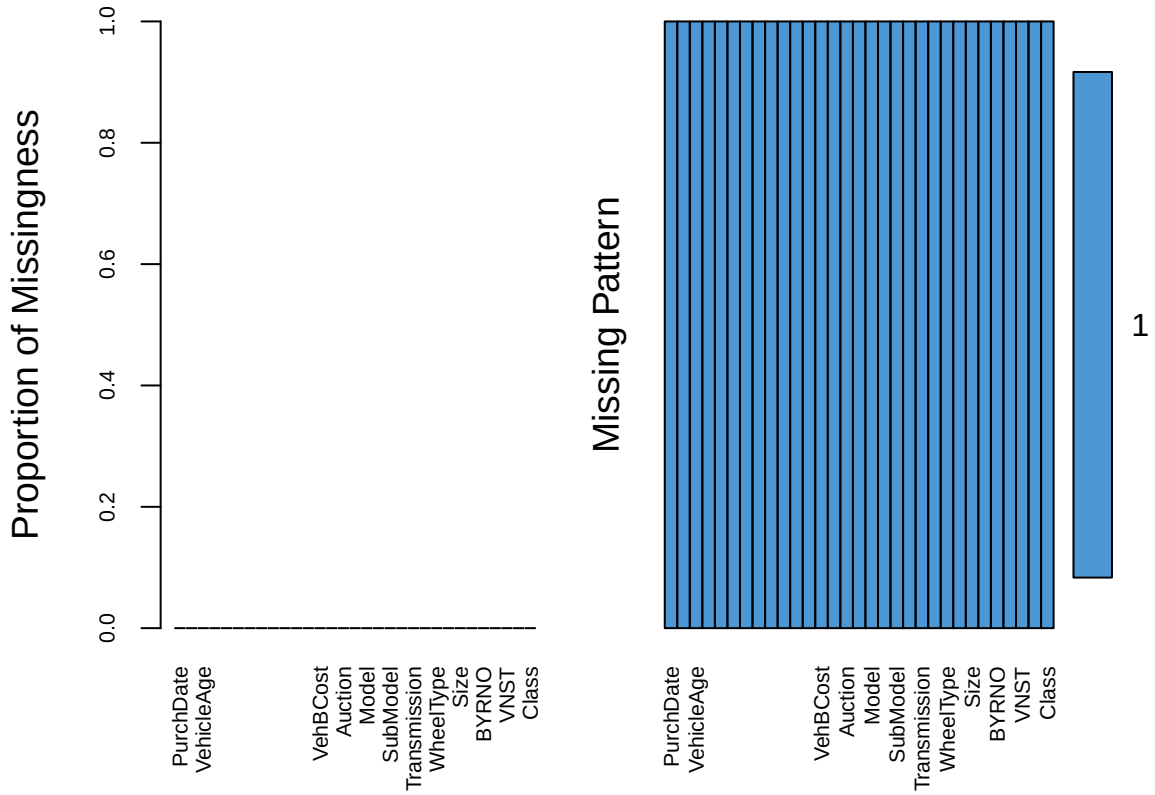
```
##  Min.   :    0                    Min.   :    0
##  1st Qu.: 4311                    1st Qu.: 5456
##  Median : 6163                    Median : 7380
##  Mean   : 6162                    Mean   : 7412
##  3rd Qu.: 7806                    3rd Qu.: 9049
##  Max.   :35722                    Max.   :36859
##  MMRAcquisitionRetailAveragePrice MMRAcquisitonRetailCleanPrice
##  Min.   :    0                   Min.   :    0
##  1st Qu.: 6319                   1st Qu.: 7526
##  Median : 8498                   Median : 9868
##  Mean   : 8538                   Mean   : 9896
##  3rd Qu.:10710                   3rd Qu.:12154
##  Max.   :39080                   Max.   :40308
##  MMRCurrentAuctionAveragePrice MMRCurrentAuctionCleanPrice
##  Min.   :    0                 Min.   :    0
##  1st Qu.: 4311                 1st Qu.: 5468
##  Median : 6130                 Median : 7390
##  Mean   : 6167                 Mean   : 7430
##  3rd Qu.: 7776                 3rd Qu.: 9045
##  Max.   :35722                 Max.   :36859
##  MMRCurrentRetailAveragePrice MMRCurrentRetailCleanPrice    VehBCost
##  Min.   :    0                Min.   :    0              Min.   : 1400
##  1st Qu.: 6565                1st Qu.: 7822              1st Qu.: 5470
##  Median : 8811                Median :10175              Median : 6750
##  Mean   : 8818                Mean   :10190              Mean   : 6754
##  3rd Qu.:10972                3rd Qu.:12370              3rd Qu.: 7910
##  Max.   :39080                Max.   :40308              Max.   :35900
##   WarrantyCost    Auction              Make              Model
##  Min.   : 462   Length:67211       Length:67211       Length:67211
##  1st Qu.: 853   Class :character   Class :character   Class :character
##  Median :1169   Mode  :character   Mode  :character   Mode  :character
##  Mean   :1279
##  3rd Qu.:1623
##  Max.   :7498
##      Trim              SubModel            Color           Transmission
##  Length:67211       Length:67211       Length:67211       Length:67211
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##   WheelTypeID     WheelType          Nationality            Size
##  Min.   :1.000   Length:67211       Length:67211       Length:67211
##  1st Qu.:1.000   Class :character   Class :character   Class :character
##  Median :1.000   Mode  :character   Mode  :character   Mode  :character
##  Mean   :1.493
##  3rd Qu.:2.000
##  Max.   :3.000
##  TopThreeAmericanName     BYRNO           VNZIP1          VNST
##  Length:67211         Min.   :  835   Min.   : 2764   Length:67211
##  Class :character     1st Qu.:17212   1st Qu.:32124   Class :character
##  Mode  :character     Median :19662   Median :74135   Mode  :character
##                       Mean   :26443   Mean   :58272
##                       3rd Qu.:22808   3rd Qu.:80022
```

```
##                       Max.   :99761   Max.   :99224
##   IsOnlineSale         Class
##  Min.   :0.00000   Min.   :0.00000
##  1st Qu.:0.00000   1st Qu.:0.00000
##  Median :0.00000   Median :0.00000
##  Mean   :0.02506   Mean   :0.09546
##  3rd Qu.:0.00000   3rd Qu.:0.00000
##  Max.   :1.00000   Max.   :1.00000
```

```
colMeans(is.na(Car_data))#Return percentage of each dimension that is missing values
```

```
##                          PurchDate                            VehYear
##                                  0                                  0
##                         VehicleAge                             VehOdo
##                                  0                                  0
## MMRAcquisitionAuctionAveragePrice   MMRAcquisitionAuctionCleanPrice
##                                  0                                  0
##  MMRAcquisitionRetailAveragePrice     MMRAcquisitonRetailCleanPrice
##                                  0                                  0
##     MMRCurrentAuctionAveragePrice       MMRCurrentAuctionCleanPrice
##                                  0                                  0
##      MMRCurrentRetailAveragePrice        MMRCurrentRetailCleanPrice
##                                  0                                  0
##                           VehBCost                       WarrantyCost
##                                  0                                  0
##                            Auction                               Make
##                                  0                                  0
##                              Model                               Trim
##                                  0                                  0
##                           SubModel                              Color
##                                  0                                  0
##                       Transmission                        WheelTypeID
##                                  0                                  0
##                          WheelType                        Nationality
##                                  0                                  0
##                               Size                TopThreeAmericanName
##                                  0                                  0
##                              BYRNO                              VNZIP1
##                                  0                                  0
##                               VNST                       IsOnlineSale
##                                  0                                  0
##                              Class
##                                  0
```

```
aggr(Car_data, col = mdc(1:2), numbers = TRUE, sortVars = TRUE, labels = names(Car_data), cex.axis = .7
```




```
##
##  Variables sorted by number of missings:
##                              Variable Count
##                             PurchDate     0
##                               VehYear     0
##                            VehicleAge     0
##                                VehOdo     0
##  MMRAcquisitionAuctionAveragePrice     0
##    MMRAcquisitionAuctionCleanPrice     0
##   MMRAcquisitionRetailAveragePrice     0
##      MMRAcquisitonRetailCleanPrice     0
##      MMRCurrentAuctionAveragePrice     0
##        MMRCurrentAuctionCleanPrice     0
##       MMRCurrentRetailAveragePrice     0
##         MMRCurrentRetailCleanPrice     0
##                             VehBCost     0
##                         WarrantyCost     0
##                              Auction     0
##                                 Make     0
##                                Model     0
##                                 Trim     0
##                             SubModel     0
##                                Color     0
##                         Transmission     0
##                          WheelTypeID     0
##                            WheelType     0
```

```
##                        Nationality     0
##                               Size     0
##               TopThreeAmericanName     0
##                              BYRNO     0
##                             VNZIP1     0
##                               VNST     0
##                       IsOnlineSale     0
##                              Class     0
```

#“Class” is our target variable : “0” means “Good Buy” and “1” means “Bad Buy” or Kicked Cars - Check for share of good buy or bad buy in the entire data

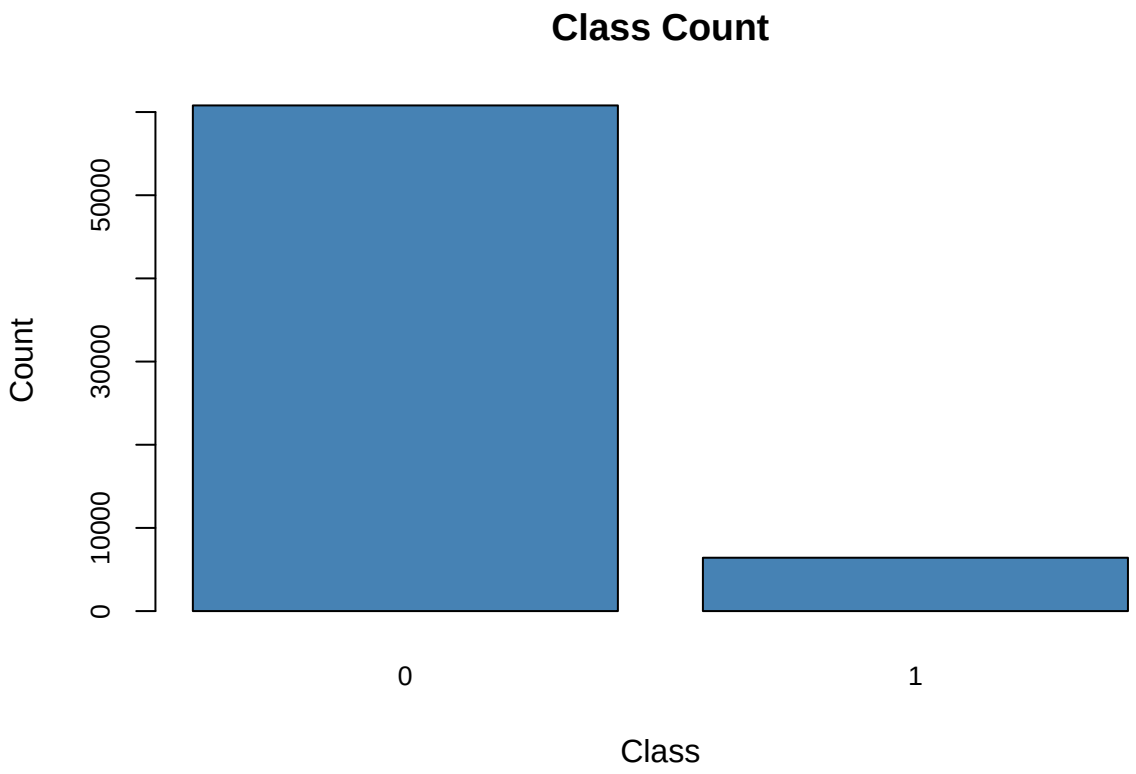
```
#Number of class count of "0" as good cars/"1" as bad cars
Count_Class <-table(Car_data$Class)
Count_Class_prop <- prop.table(Count_Class)
Count_Class #483 Customers churn
```

```
##
##     0     1
## 60795  6416
```

```
Count_Class_prop
```

```
##
##          0          1
## 0.90453944 0.09546056
```

```
barplot(Count_Class,
        main = "Class Count",
        xlab = "Class",
        ylab = "Count",
        col = "steelblue",
        cex.names = 0.8,
        cex.axis = 0.8,
        width = 0.3) #"0" means "Good Buy" and "1" means "Bad Buy"
```



#Creating two different dataset (1) Car_num : All numerical Variabls (2)Car_char : All Categorical Variables

```
t(t(names(Car_data)))#column names
```

```
##       [,1]
##  [1,] "PurchDate"
##  [2,] "VehYear"
##  [3,] "VehicleAge"
##  [4,] "VehOdo"
##  [5,] "MMRAcquisitionAuctionAveragePrice"
##  [6,] "MMRAcquisitionAuctionCleanPrice"
##  [7,] "MMRAcquisitionRetailAveragePrice"
##  [8,] "MMRAcquisitonRetailCleanPrice"
##  [9,] "MMRCurrentAuctionAveragePrice"
## [10,] "MMRCurrentAuctionCleanPrice"
## [11,] "MMRCurrentRetailAveragePrice"
## [12,] "MMRCurrentRetailCleanPrice"
## [13,] "VehBCost"
## [14,] "WarrantyCost"
## [15,] "Auction"
## [16,] "Make"
## [17,] "Model"
## [18,] "Trim"
## [19,] "SubModel"
## [20,] "Color"
```

```
## [21,] "Transmission"
## [22,] "WheelTypeID"
## [23,] "WheelType"
## [24,] "Nationality"
## [25,] "Size"
## [26,] "TopThreeAmericanName"
## [27,] "BYRNO"
## [28,] "VNZIP1"
## [29,] "VNST"
## [30,] "IsOnlineSale"
## [31,] "Class"
```

```
Car_num <- Car_data[,c(2:14,22,27,28,30,31)]#Numerical variables

Car_char <- Car_data[,c(15:21,23:26,27,29,31)]#Categorical variables

#Checking the categories for variables seems useful for the analysis
unique(Car_char$Auction)
```

```
## [1] "MANHEIM" "ADESA"   "OTHER"
```

```
unique(Car_char$Make)
```

```
##  [1] "DODGE"          "FORD"           "PONTIAC"        "CHEVROLET"
##  [5] "KIA"            "ISUZU"          "JEEP"           "HYUNDAI"
##  [9] "CHRYSLER"       "MITSUBISHI"     "TOYOTA"         "NISSAN"
## [13] "MERCURY"        "SATURN"         "BUICK"          "MAZDA"
## [17] "GMC"            "HONDA"          "VOLKSWAGEN"     "VOLVO"
## [21] "OLDSMOBILE"     "SCION"          "SUZUKI"         "MINI"
## [25] "SUBARU"         "ACURA"          "LINCOLN"        "CADILLAC"
## [29] "PLYMOUTH"       "INFINITI"       "'TOYOTA SCION'" "LEXUS"
```

```
unique(Car_char$Transmission)
```

```
## [1] "AUTO"   "MANUAL" "Manual"
```

```
unique(Car_char$WheelType)
```

```
## [1] "Covers"  "Alloy"   "Special"
```

```
unique(Car_char$Nationality)
```

```
## [1] "AMERICAN"         "'OTHER ASIAN'"    "'TOP LINE ASIAN'" "OTHER"
```

```
unique(Car_char$Size)
```

```
##  [1] "MEDIUM"        "LARGE"         "'MEDIUM SUV'"  "COMPACT"
##  [5] "'LARGE TRUCK'" "VAN"           "'SMALL SUV'"   "'LARGE SUV'"
##  [9] "SPECIALTY"     "CROSSOVER"     "SPORTS"        "'SMALL TRUCK'"
```

```r
unique(Car_char$TopThreeAmericanName)
```

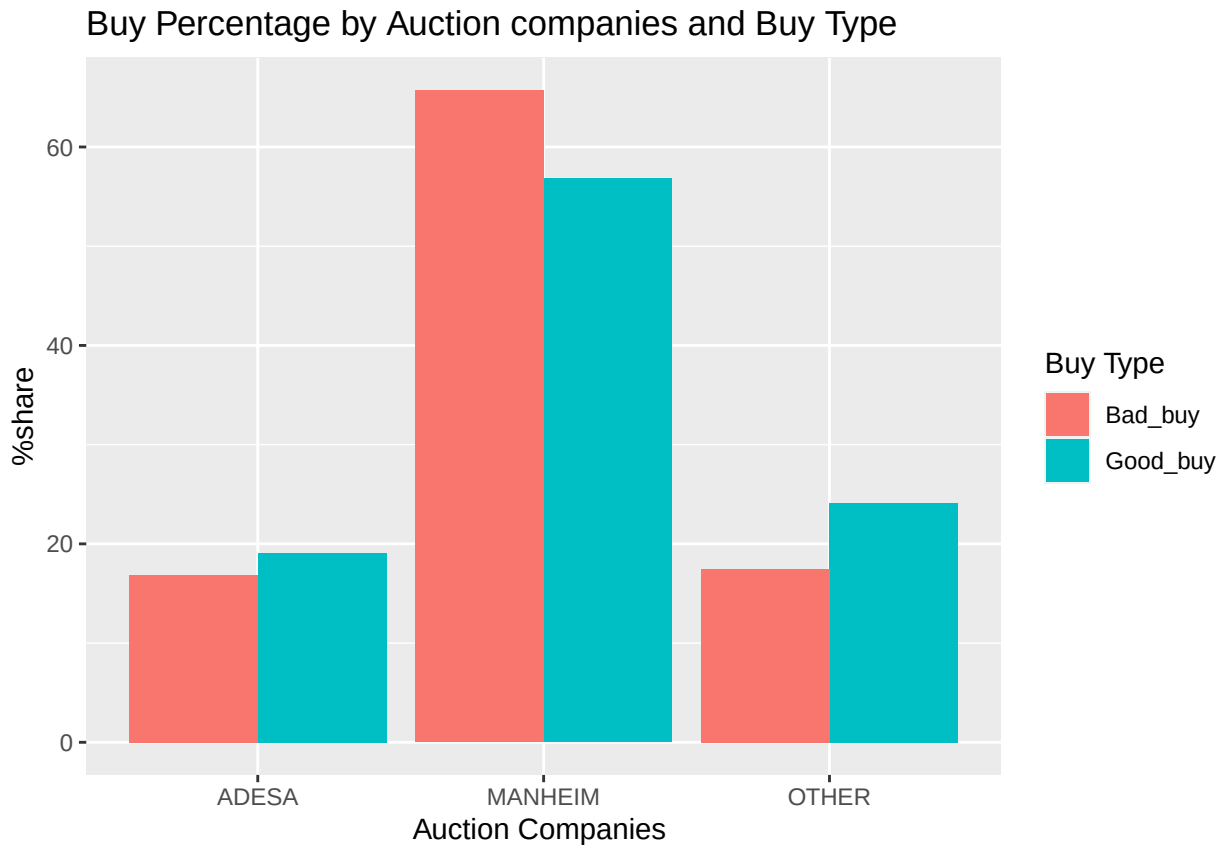
```
## [1] "CHRYSLER" "FORD"     "GM"       "OTHER"
```

```r
unique(Car_char$VNST)
```

```
##  [1] "FL" "TX" "CO" "CA" "NC" "TN" "SC" "AZ" "IA" "VA" "MD" "AL" "OK" "IN" "LA"
## [16] "UT" "KY" "NM" "OH" "NV" "GA" "MO" "WA" "NJ" "WV" "MS" "ID" "PA" "IL" "NH"
## [31] "OR" "NE" "AR" "MN" "MA" "NY" "MI"
```

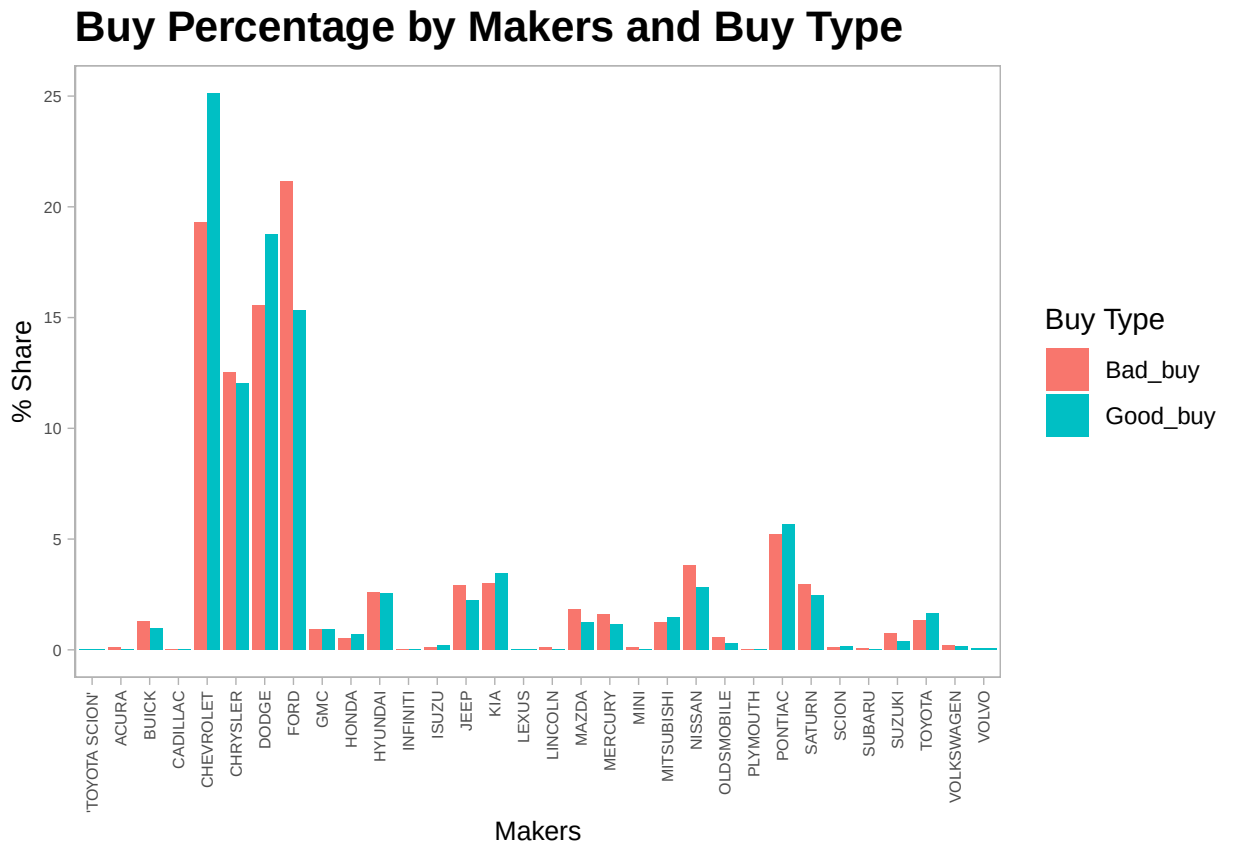
#Data Visualization : Categorical variables

```r
#buy rate by Auction Companies
Car_char$Class <- as.factor(Car_char$Class) #Converting "Class into Factor"
Good_buy <- subset(Car_char, Class == 0)
Bad_buy <- subset(Car_char, Class == 1)
Bad_buy_Auction_comp <- Bad_buy %>% group_by(Auction) %>% summarise(buy_count = n())
Good_buy_Auction_comp <- Good_buy %>% group_by(Auction) %>% summarise(buy_count = n())

total_bad_buy <- sum(Bad_buy_Auction_comp$buy_count)
total_good_buy <- sum(Good_buy_Auction_comp$buy_count)

Bad_buy_Auction_comp$buy_percentage <- (Bad_buy_Auction_comp$buy_count / total_bad_buy) * 100
Good_buy_Auction_comp$buy_percentage <- (Good_buy_Auction_comp$buy_count / total_good_buy) * 100

merged_Auction <- rbind(transform(Good_buy_Auction_comp, Buy_Type = "Good_buy"),
                        transform(Bad_buy_Auction_comp, Buy_Type = "Bad_buy"))

ggplot(merged_Auction, aes(x = Auction, y = buy_percentage, fill = Buy_Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Auction Companies ", y = "%share", fill = "Buy Type") +
  ggtitle("Buy Percentage by Auction companies and Buy Type")# Create bar plot using ggplot
```

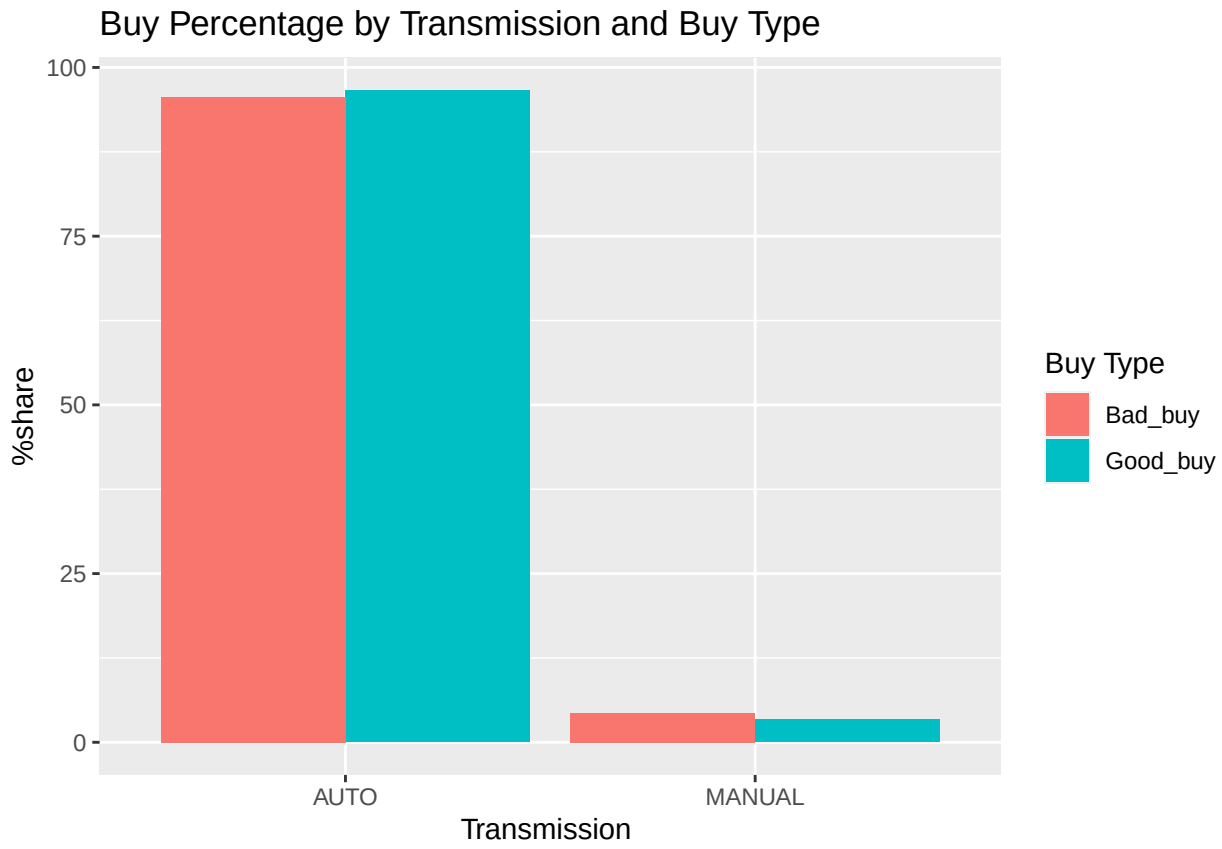



```
#buy rate by Car Makers
Bad_buy_Make <- Bad_buy %>% group_by(Make) %>% summarise(buy_count = n())
Good_buy_Make <- Good_buy %>% group_by(Make) %>% summarise(buy_count = n())

total_bad_buy <- sum(Bad_buy_Make$buy_count)
total_good_buy <- sum(Good_buy_Make$buy_count)

Bad_buy_Make$buy_percentage <- (Bad_buy_Make$buy_count / total_bad_buy) * 100
Good_buy_Make$buy_percentage <- (Good_buy_Make$buy_count / total_good_buy) * 100

merged_Make <- rbind(transform(Good_buy_Make, Buy_Type = "Good_buy"),
                     transform(Bad_buy_Make, Buy_Type = "Bad_buy"))

ggplot(merged_Make, aes(x = Make, y = buy_percentage, fill = Buy_Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Makers", y = "% Share", fill = "Buy Type") +
  ggtitle("Buy Percentage by Makers and Buy Type") +
  theme_light() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.text.x = element_text(size = 6, angle = 90, vjust = 0.5, hjust = 1),
    axis.text.y = element_text(size = 6),
    axis.title.x = element_text(size = 10),
    axis.title.y = element_text(size = 10),
    plot.title = element_text(size = 16, face = "bold")
```

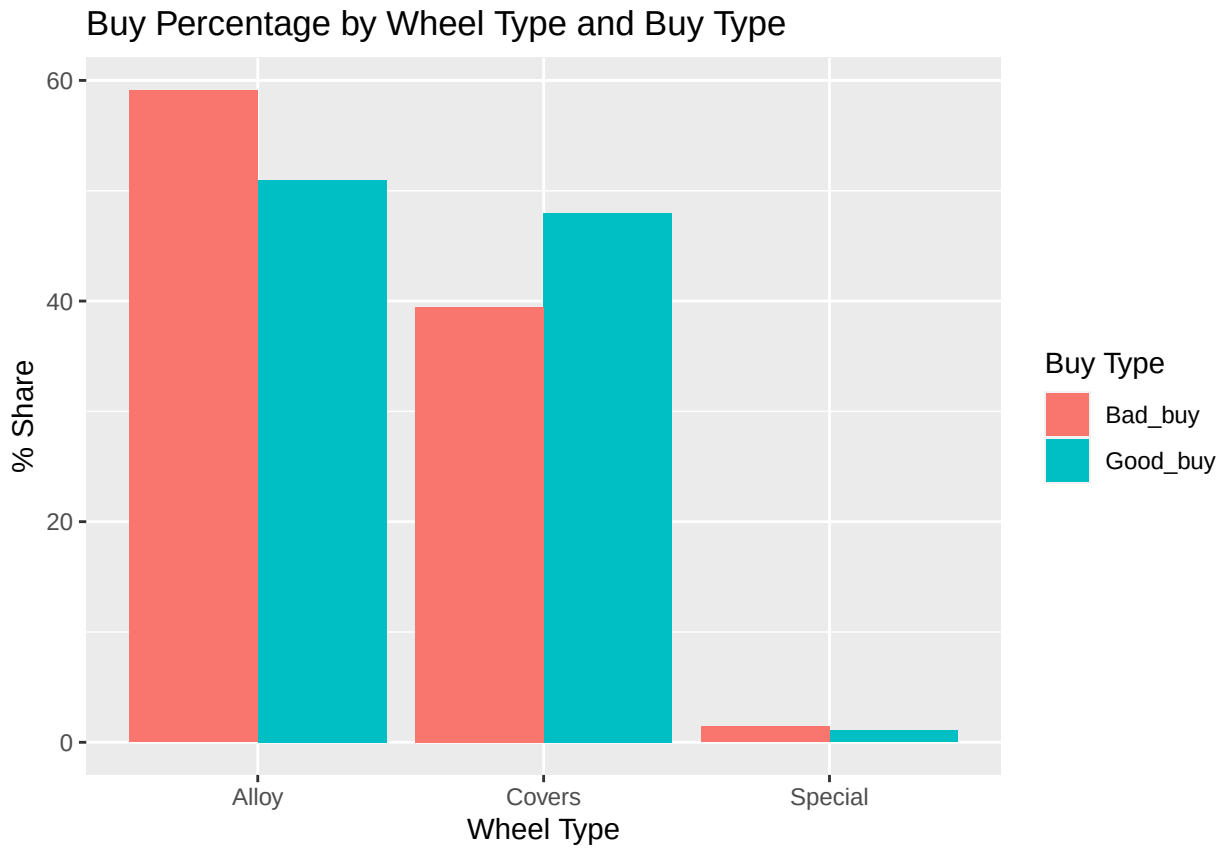
```
)
```



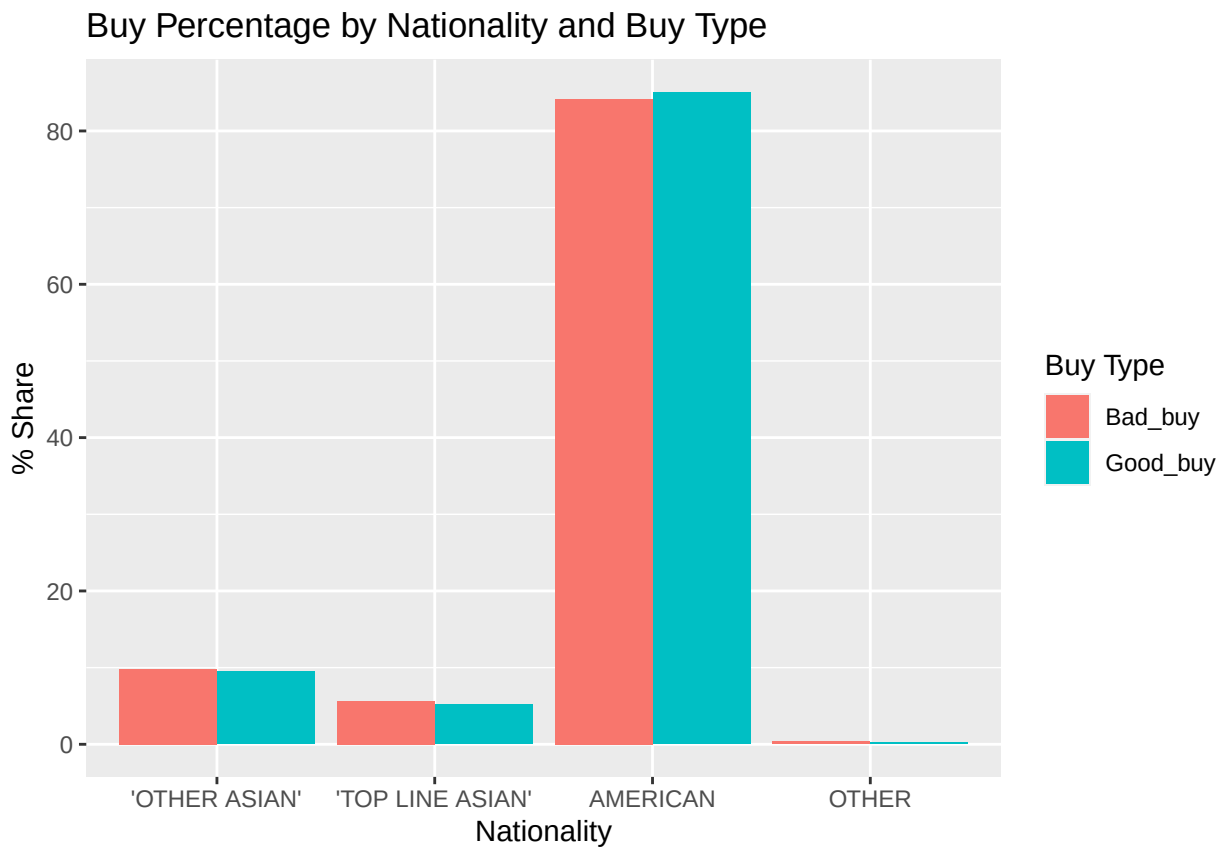

```
#buy rate by Transmission
Bad_buy_Transmission <- Bad_buy %>% group_by(Transmission) %>% summarise(buy_count = n())
Good_buy_Transmission <- Good_buy %>% group_by(Transmission) %>% summarise(buy_count = n())

total_bad_buy <- sum(Bad_buy_Transmission$buy_count)
total_good_buy <- sum(Good_buy_Transmission$buy_count)

Bad_buy_Transmission$buy_percentage <- (Bad_buy_Transmission$buy_count / total_bad_buy) * 100
Good_buy_Transmission$buy_percentage <- (Good_buy_Transmission$buy_count / total_good_buy) * 100

merged_Transmission <- rbind(transform(Good_buy_Transmission, Buy_Type = "Good_buy"),
                             transform(Bad_buy_Transmission, Buy_Type = "Bad_buy"))

merged_Transmission1 <- subset(merged_Transmission, Transmission != "Manual")

ggplot(merged_Transmission1, aes(x = Transmission, y = buy_percentage, fill = Buy_Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Transmission ", y = "%share", fill = "Buy Type") +
  ggtitle("Buy Percentage by Transmission and Buy Type")
```




```r
#buy rate by Wheeltype
Bad_buy_WheelType <- Bad_buy %>% group_by(WheelType) %>% summarise(buy_count = n())
Good_buy_WheelType <- Good_buy %>% group_by(WheelType) %>% summarise(buy_count = n())

total_bad_buy <- sum(Bad_buy_WheelType$buy_count)
total_good_buy <- sum(Good_buy_WheelType$buy_count)

Bad_buy_WheelType$buy_percentage <- (Bad_buy_WheelType$buy_count / total_bad_buy) * 100
Good_buy_WheelType$buy_percentage <- (Good_buy_WheelType$buy_count / total_good_buy) * 100

merged_WheelType <- rbind(transform(Good_buy_WheelType, Buy_Type = "Good_buy"),
                          transform(Bad_buy_WheelType, Buy_Type = "Bad_buy"))
ggplot(merged_WheelType, aes(x = WheelType, y = buy_percentage, fill = Buy_Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Wheel Type", y = "% Share", fill = "Buy Type") +
  ggtitle("Buy Percentage by Wheel Type and Buy Type")# Create bar plot using ggplot
```

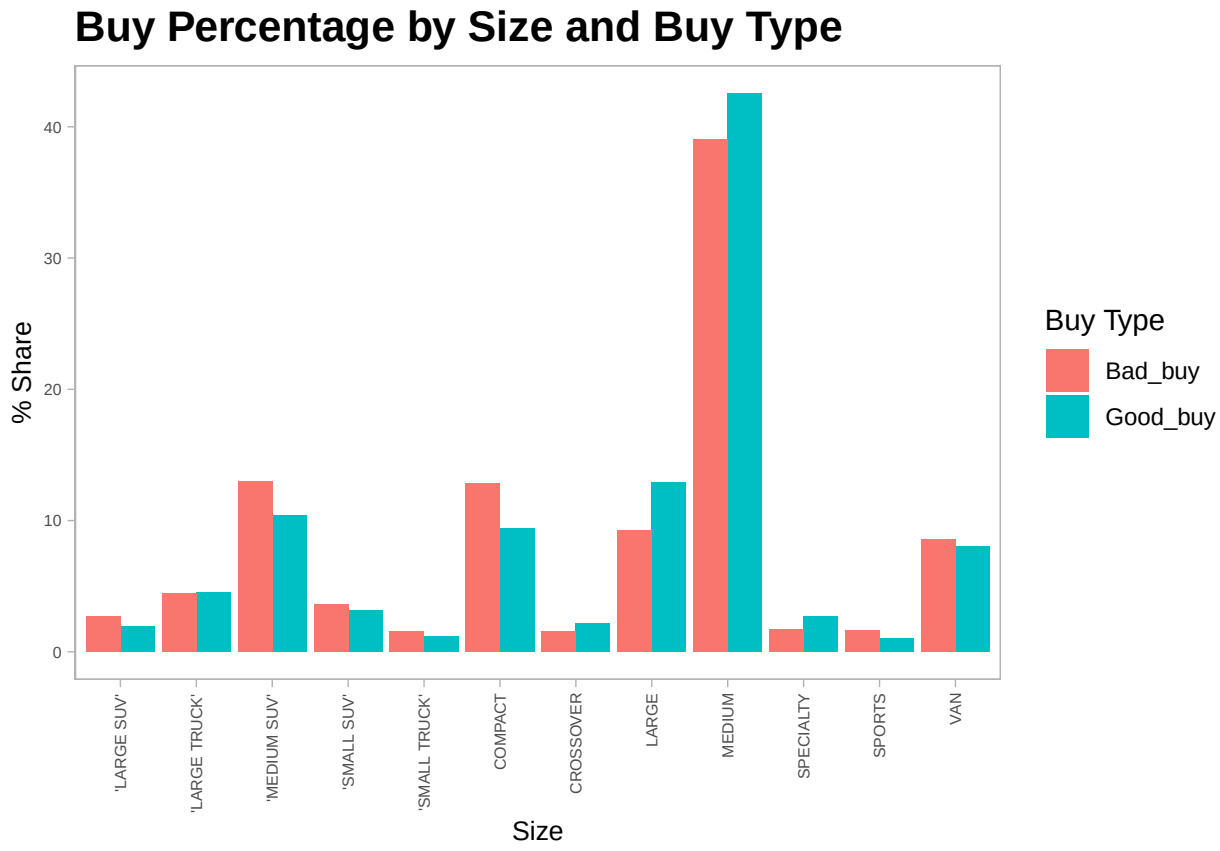



```
#buy rate by Nationality
Bad_buy_Nationality <- Bad_buy %>% group_by(Nationality) %>% summarise(buy_count = n())
Good_buy_Nationality <- Good_buy %>% group_by(Nationality) %>% summarise(buy_count = n())

total_bad_buy <- sum(Bad_buy_Nationality$buy_count)
total_good_buy <- sum(Good_buy_Nationality$buy_count)

Bad_buy_Nationality$buy_percentage <- (Bad_buy_Nationality$buy_count / total_bad_buy) * 100
Good_buy_Nationality$buy_percentage <- (Good_buy_Nationality$buy_count / total_good_buy) * 100

merged_Nationality <- rbind(transform(Good_buy_Nationality, Buy_Type = "Good_buy"),
                            transform(Bad_buy_Nationality, Buy_Type = "Bad_buy"))


ggplot(merged_Nationality, aes(x = Nationality, y = buy_percentage, fill = Buy_Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Nationality", y = "% Share", fill = "Buy Type") +
  ggtitle("Buy Percentage by Nationality and Buy Type")# Create bar plot using ggplot
```

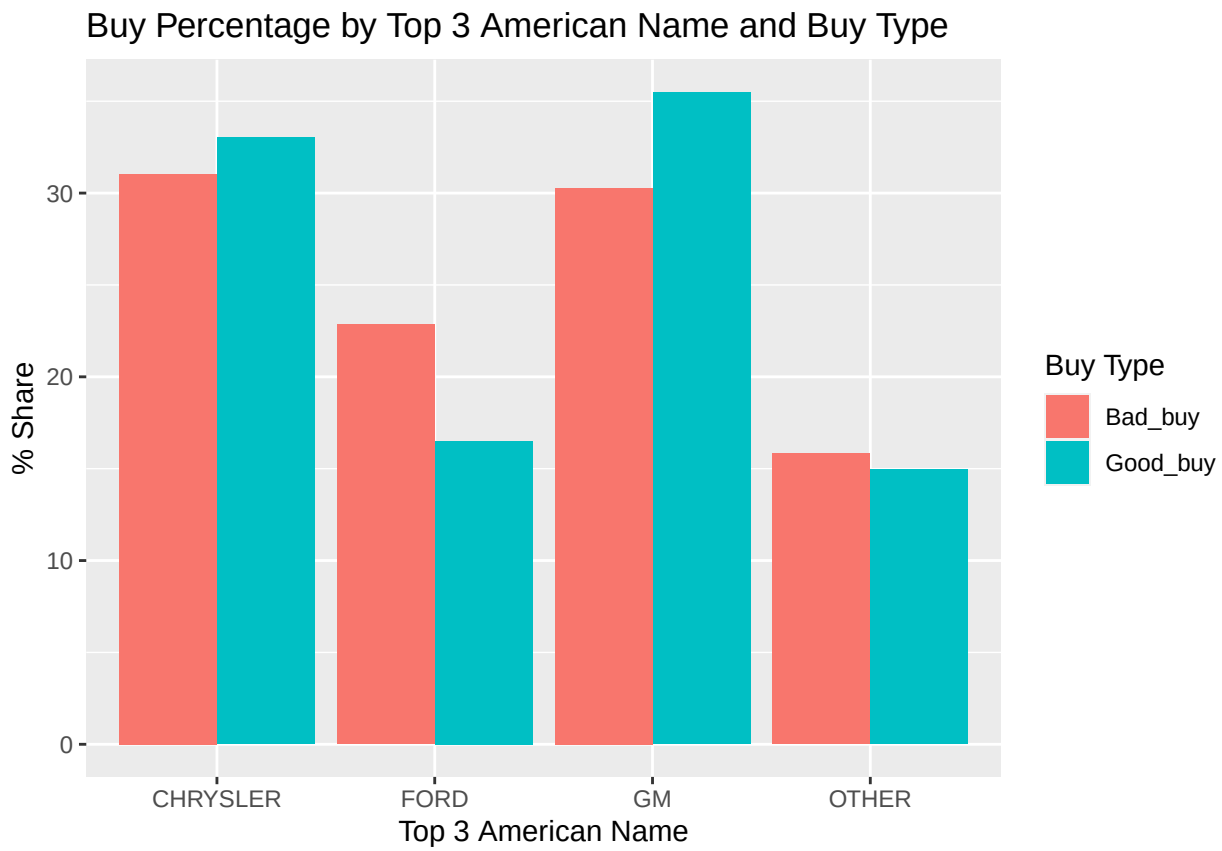



```r
#buy rate by Size
Bad_buy_Size <- Bad_buy %>% group_by(Size) %>% summarise(buy_count = n())
Good_buy_Size <- Good_buy %>% group_by(Size) %>% summarise(buy_count = n())

total_bad_buy <- sum(Bad_buy_Size$buy_count)
total_good_buy <- sum(Good_buy_Size$buy_count)

Bad_buy_Size$buy_percentage <- (Bad_buy_Size$buy_count / total_bad_buy) * 100
Good_buy_Size$buy_percentage <- (Good_buy_Size$buy_count / total_good_buy) * 100

merged_Size <- rbind(transform(Good_buy_Size, Buy_Type = "Good_buy"),
                     transform(Bad_buy_Size, Buy_Type = "Bad_buy"))

ggplot(merged_Size, aes(x = Size, y = buy_percentage, fill = Buy_Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Size", y = "% Share", fill = "Buy Type") +
  ggtitle("Buy Percentage by Size and Buy Type") +
  theme_light() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.text.x = element_text(size = 6, angle = 90, vjust = 0.5, hjust = 1),
    axis.text.y = element_text(size = 6),
    axis.title.x = element_text(size = 10),
    axis.title.y = element_text(size = 10),
    plot.title = element_text(size = 16, face = "bold")
```

```
)
```

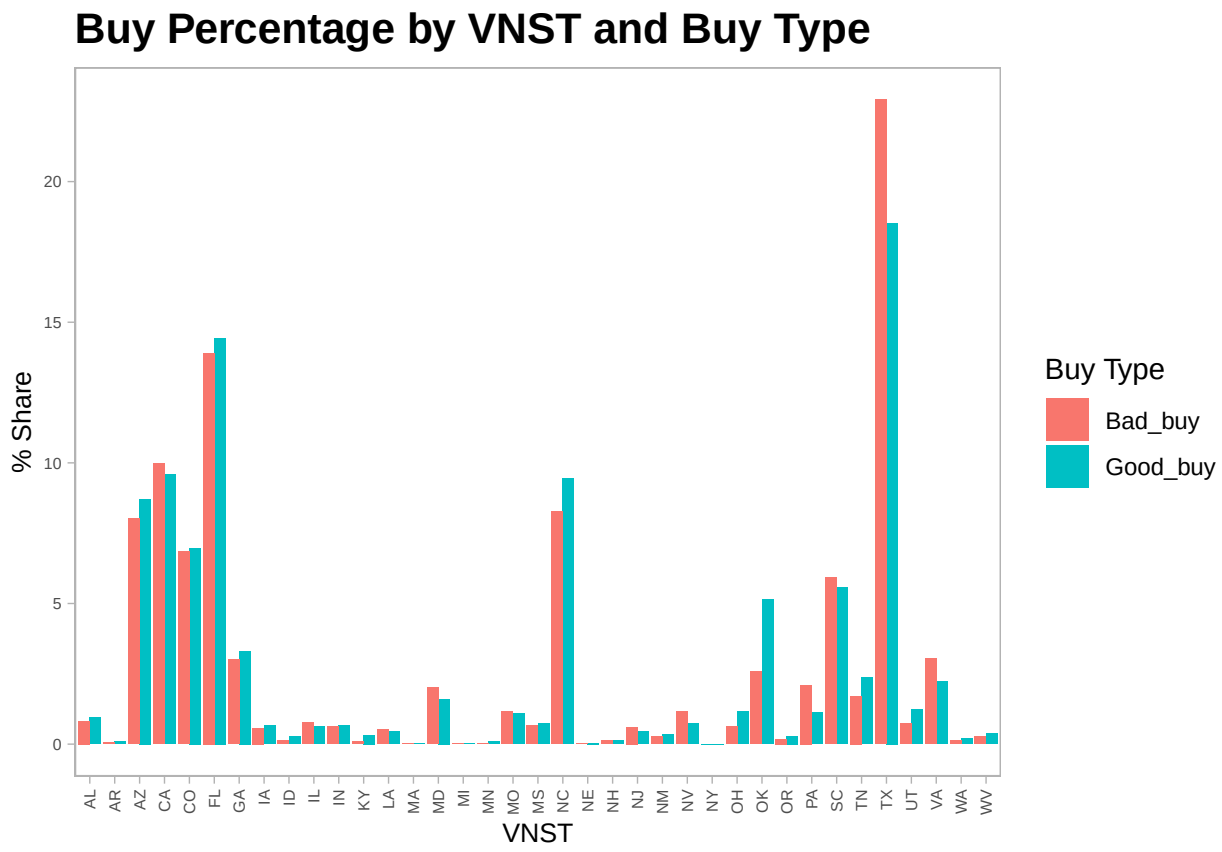



```
#buy rate by top 3 American Name
Bad_buy_topamericanname <- Bad_buy %>% group_by(TopThreeAmericanName) %>% summarise(buy_count = n())
Good_buy_topamericanname <- Good_buy %>% group_by(TopThreeAmericanName) %>% summarise(buy_count = n())

total_bad_buy <- sum(Bad_buy_topamericanname$buy_count)
total_good_buy <- sum(Good_buy_topamericanname$buy_count)

Bad_buy_topamericanname$buy_percentage <- (Bad_buy_topamericanname$buy_count / total_bad_buy) * 100
Good_buy_topamericanname$buy_percentage <- (Good_buy_topamericanname$buy_count / total_good_buy) * 100

merged_topamericanname <- rbind(transform(Good_buy_topamericanname, Buy_Type = "Good_buy"),
                                transform(Bad_buy_topamericanname, Buy_Type = "Bad_buy"))

ggplot(merged_topamericanname, aes(x = TopThreeAmericanName, y = buy_percentage, fill = Buy_Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Top 3 American Name", y = "% Share", fill = "Buy Type") +
  ggtitle("Buy Percentage by Top 3 American Name and Buy Type")
```




```r
#buy rate by States
Bad_buy_VNST <- Bad_buy %>% group_by(VNST) %>% summarise(buy_count = n())
Good_buy_VNST <- Good_buy %>% group_by(VNST) %>% summarise(buy_count = n())

total_bad_buy <- sum(Bad_buy_VNST$buy_count)
total_good_buy <- sum(Good_buy_VNST$buy_count)

Bad_buy_VNST$buy_percentage <- (Bad_buy_VNST$buy_count / total_bad_buy) * 100
Good_buy_VNST$buy_percentage <- (Good_buy_VNST$buy_count / total_good_buy) * 100

merged_VNST <- rbind(transform(Good_buy_VNST, Buy_Type = "Good_buy"),
                     transform(Bad_buy_VNST, Buy_Type = "Bad_buy"))

ggplot(merged_VNST, aes(x = VNST, y = buy_percentage, fill = Buy_Type)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "VNST", y = "% Share", fill = "Buy Type") +
  ggtitle("Buy Percentage by VNST and Buy Type") +
  theme_light() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.text.x = element_text(size = 6, angle = 90, vjust = 0.5, hjust = 1),
    axis.text.y = element_text(size = 6),
    axis.title.x = element_text(size = 10),
    axis.title.y = element_text(size = 10),
    plot.title = element_text(size = 16, face = "bold")
```

```
)
```

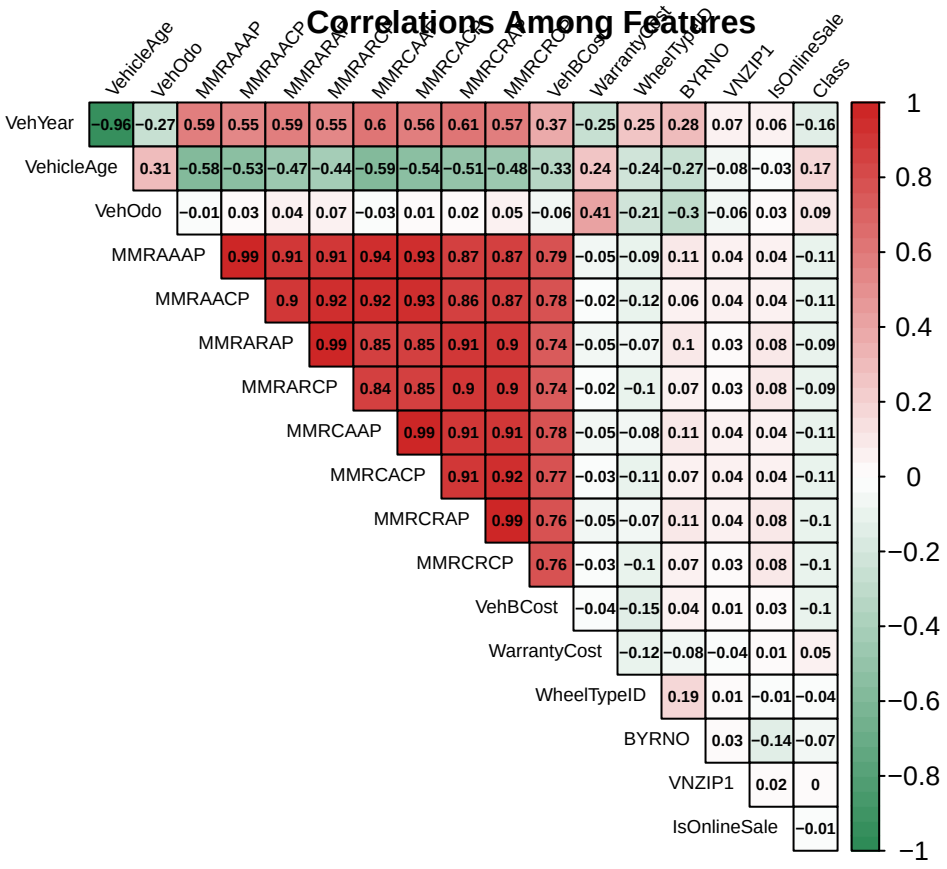


#Data Visualization : numerical variables

```
#correlations amoung feetures
t(t(names(Car_num)))#column names
```

```
##       [,1]
##  [1,] "VehYear"
##  [2,] "VehicleAge"
##  [3,] "VehOdo"
##  [4,] "MMRAcquisitionAuctionAveragePrice"
##  [5,] "MMRAcquisitionAuctionCleanPrice"
##  [6,] "MMRAcquisitionRetailAveragePrice"
##  [7,] "MMRAcquisitonRetailCleanPrice"
##  [8,] "MMRCurrentAuctionAveragePrice"
##  [9,] "MMRCurrentAuctionCleanPrice"
## [10,] "MMRCurrentRetailAveragePrice"
## [11,] "MMRCurrentRetailCleanPrice"
## [12,] "VehBCost"
## [13,] "WarrantyCost"
## [14,] "WheelTypeID"
## [15,] "BYRNO"
## [16,] "VNZIP1"
## [17,] "IsOnlineSale"
## [18,] "Class"
```

```
names(Car_num)[4] <- "MMRAAAP"
names(Car_num)[5] <- "MMRAACP"
names(Car_num)[6] <- "MMRARAP"
names(Car_num)[7] <- "MMRARCP"
names(Car_num)[8] <- "MMRCAAP"
names(Car_num)[9] <- "MMRCACP"
names(Car_num)[10] <- "MMRCRAP"
names(Car_num)[11] <- "MMRCRCP"


par(mar = c(6, 4, 3, 2) + 0.5)
cor_matrix <- round(cor(Car_num), 2)
corrplot(cor_matrix, method = "color", type = "upper", tl.col = "black",
         tl.srt = 50, tl.cex = 0.6, diag = FALSE, outline = TRUE,
         col = colorRampPalette(c("seagreen", "white", "firebrick3"))(50),
         addCoef.col = "black", number.cex = 0.5)
title("Correlations Among Features", line = 2.7, cex.main = 1)
```




```
#Summary table showing the median of each variable
Car_num$Class <- as.factor(Car_num$Class)
Car_num$IsOnlineSale <- as.factor(Car_num$IsOnlineSale)

#Also checking the proportion for Online sale for Class(good buy or bad buy)
Online_Sale <- table(IsOnlineSale = Car_num$IsOnlineSale, Class= Car_num$Class )
Online_Sale_prop <- prop.table(Online_Sale)
```

```
Online_Sale
```

```
##             Class
## IsOnlineSale     0     1
##            0 59245  6282
##            1  1550   134
```

```
Online_Sale_prop
```

```
##             Class
## IsOnlineSale           0           1
##            0 0.881477734 0.093466843
##            1 0.023061701 0.001993721
```

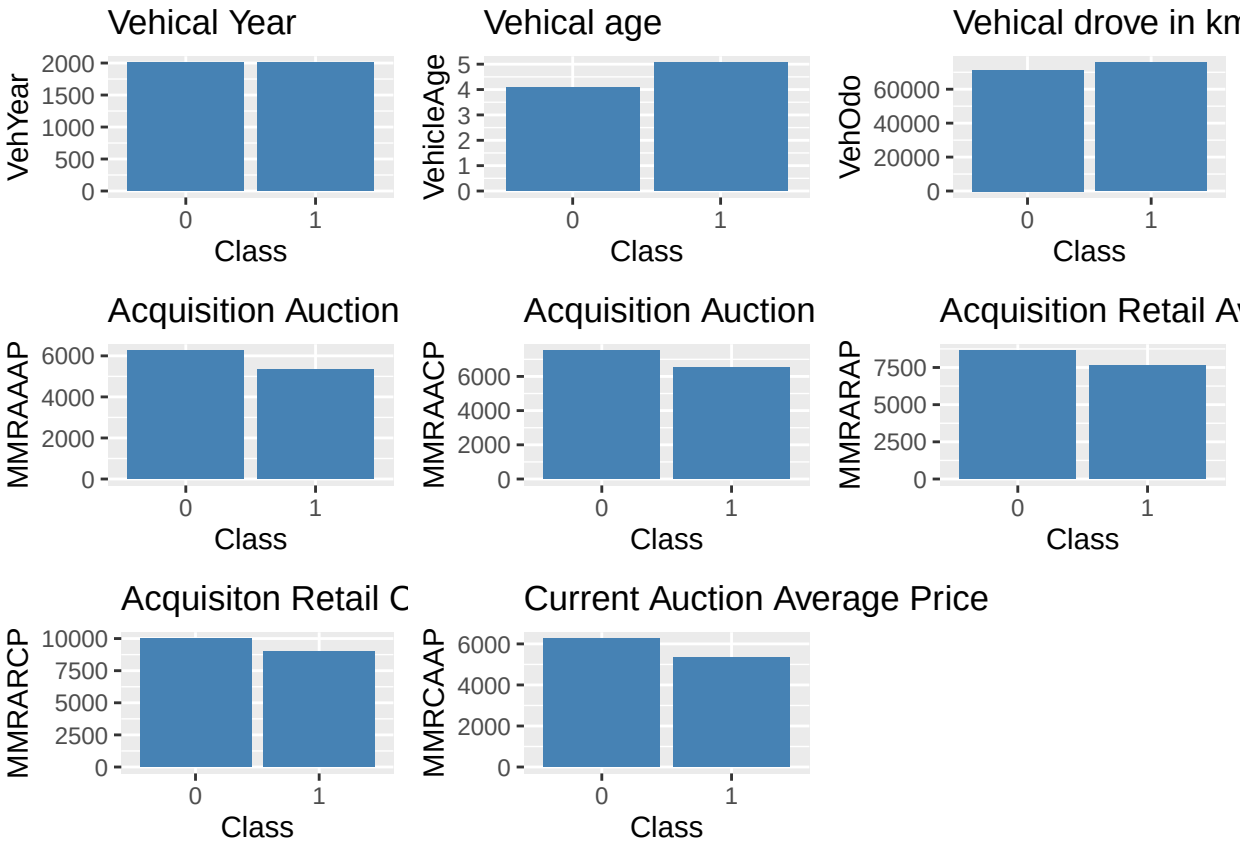
```
Var_table <- Car_num[,c(1:16,18)] %>% group_by(Class) %>% summarize(across(.cols = everything(), .fns =
print(Var_table)
```

```
## # A tibble: 2 x 17
##   Class VehYear Vehicle~1 VehOdo MMRAAAP MMRAACP MMRARAP MMRARCP MMRCAAP MMRCACP
##   <fct>   <dbl>     <dbl>  <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 0       2005.      4.07 71323.   6252.   7505.   8631.   9991.   6255.   7521.
## 2 1       2004.      5.06 75639.   5313.   6533.   7663.   8999.   5330.   6560.
## # ... with 7 more variables: MMRCRAP <dbl>, MMRCRCP <dbl>, VehBCost <dbl>,
## #   WarrantyCost <dbl>, WheelTypeID <dbl>, BYRNO <dbl>, VNZIP1 <dbl>, and
## #   abbreviated variable name 1: VehicleAge
```

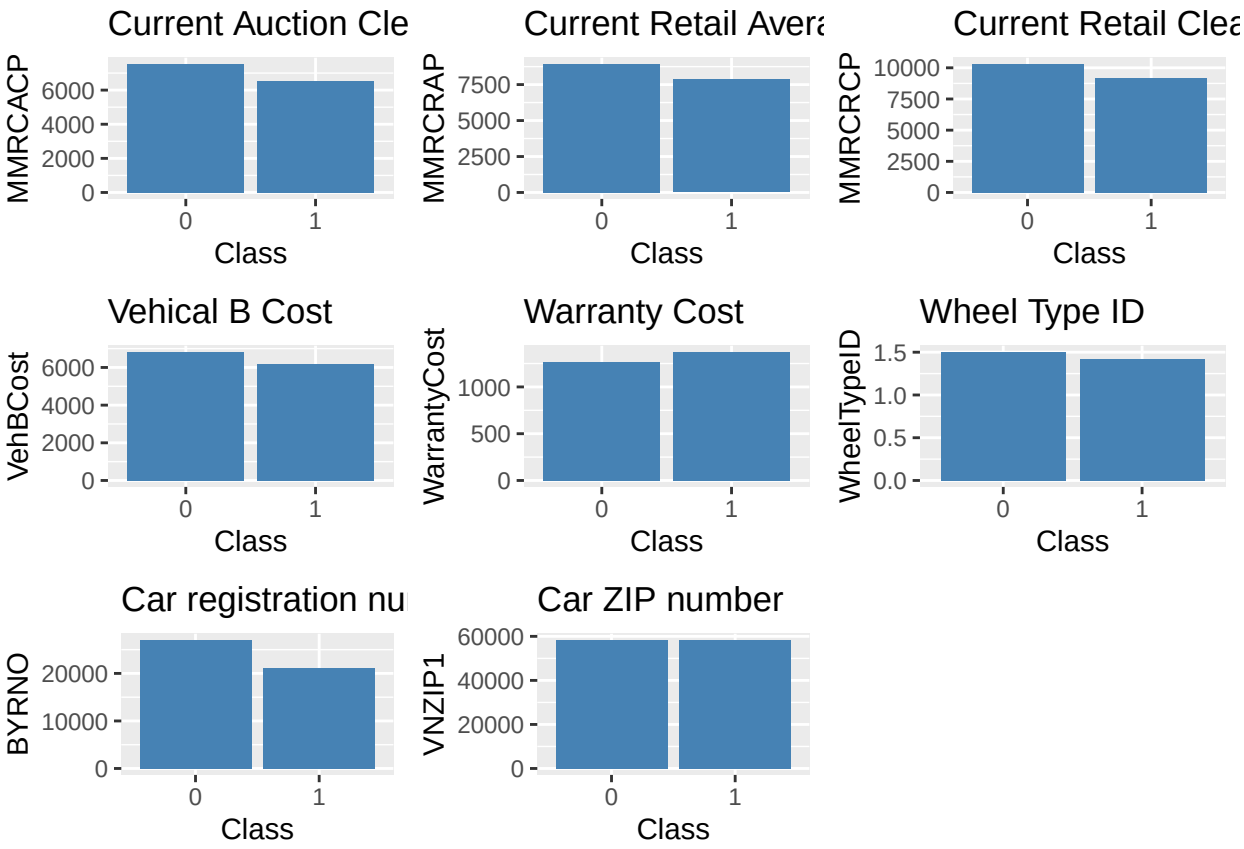
```
# Create bar graph
VehYear <- ggplot(Var_table, aes(x = Class, y = VehYear)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "VehYear") +
  ggtitle("Vehical Year")

VehicleAge <- ggplot(Var_table, aes(x = Class, y = VehicleAge)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "VehicleAge") +
  ggtitle("Vehical age") #Slight high for bad buy data

VehOdo <- ggplot(Var_table, aes(x = Class, y = VehOdo)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "VehOdo") +
  ggtitle("Vehical drove in km") #slight high for bad buy

MMRAcquisitionAuctionAveragePrice <- ggplot(Var_table, aes(x = Class, y = MMRAAAP)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "MMRAAAP") +
  ggtitle("Acquisition Auction Average Price")#low for bad Buy

MMRAcquisitionAuctionCleanPrice <- ggplot(Var_table, aes(x = Class, y = MMRAACP)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "MMRAACP") +
  ggtitle("Acquisition Auction Clean Price")
```

```
MMRAcquisitionRetailAveragePrice <- ggplot(Var_table, aes(x = Class, y = MMRARAP)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "MMRARAP") +
  ggtitle("Acquisition Retail Average Price")

MMRAcquisitonRetailCleanPrice <- ggplot(Var_table, aes(x = Class, y = MMRARCP)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "MMRARCP") +
  ggtitle("Acquisiton Retail Clean Price")

MMRCurrentAuctionAveragePrice <- ggplot(Var_table, aes(x = Class, y = MMRCAAP)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "MMRCAAP") +
  ggtitle("Current Auction Average Price")

MMRCurrentAuctionCleanPrice <- ggplot(Var_table, aes(x = Class, y = MMRCACP)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "MMRCACP") +
  ggtitle("Current Auction Clean Price")

MMRCurrentRetailAveragePrice <- ggplot(Var_table, aes(x = Class, y = MMRCRAP)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "MMRCRAP") +
  ggtitle("Current Retail Average Price")

MMRCurrentRetailCleanPrice <- ggplot(Var_table, aes(x = Class, y = MMRCRCP)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "MMRCRCP") +
  ggtitle("Current Retail Clean Price")

VehBCost <- ggplot(Var_table, aes(x = Class, y = VehBCost)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "VehBCost") +
  ggtitle("Vehical B Cost")

WarrantyCost <- ggplot(Var_table, aes(x = Class, y = WarrantyCost)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "WarrantyCost") +
  ggtitle("Warranty Cost")#high for bad buy

WheelTypeID <- ggplot(Var_table, aes(x = Class, y = WheelTypeID)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "WheelTypeID") +
  ggtitle("Wheel Type ID")

BYRNO <- ggplot(Var_table, aes(x = Class, y = BYRNO)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "BYRNO") +
  ggtitle("Car registration number")

VNZIP1 <- ggplot(Var_table, aes(x = Class, y = VNZIP1)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(x = "Class", y = "VNZIP1") +
```

```
  ggtitle("Car ZIP number")

plot_grid(VehYear, VehicleAge, VehOdo, MMRAcquisitionAuctionAveragePrice,MMRAcquisitionAuctionCleanPric
```



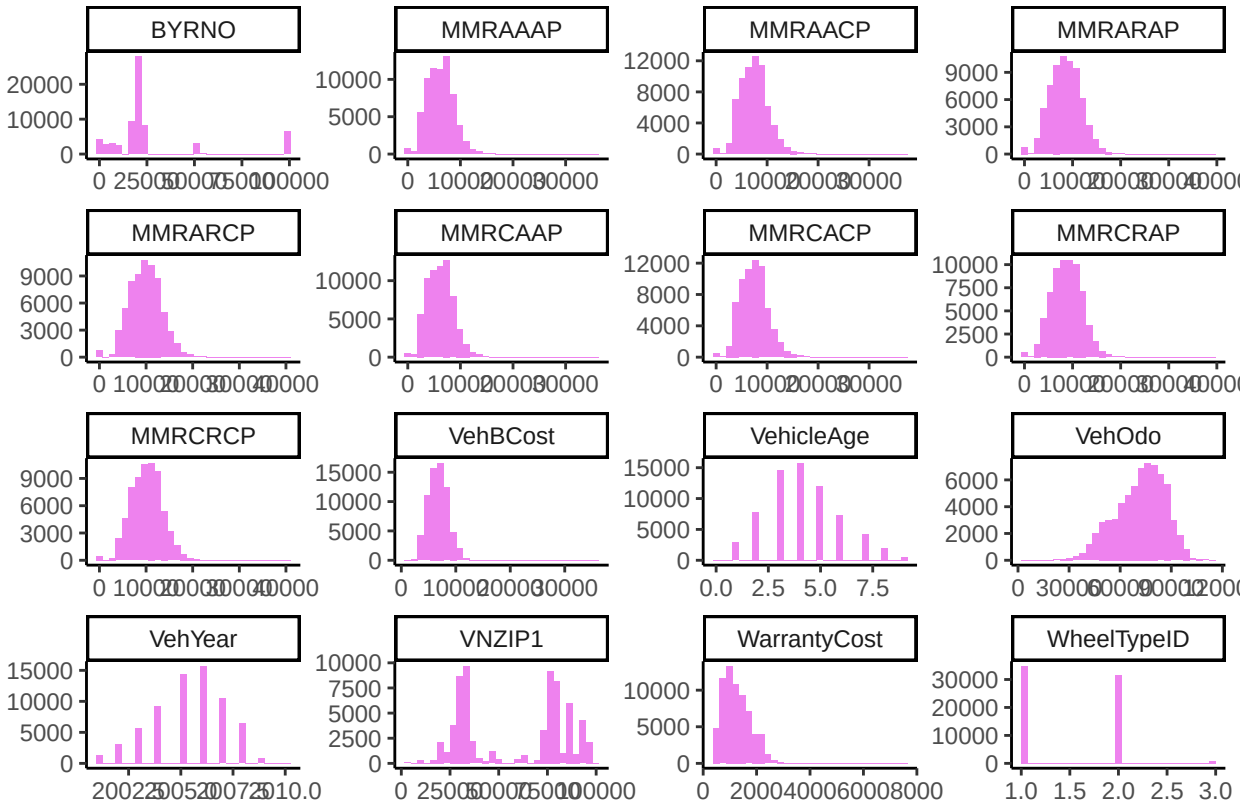

```
plot_grid (MMRCurrentAuctionCleanPrice,MMRCurrentRetailAveragePrice,MMRCurrentRetailCleanPrice,VehBCost
```



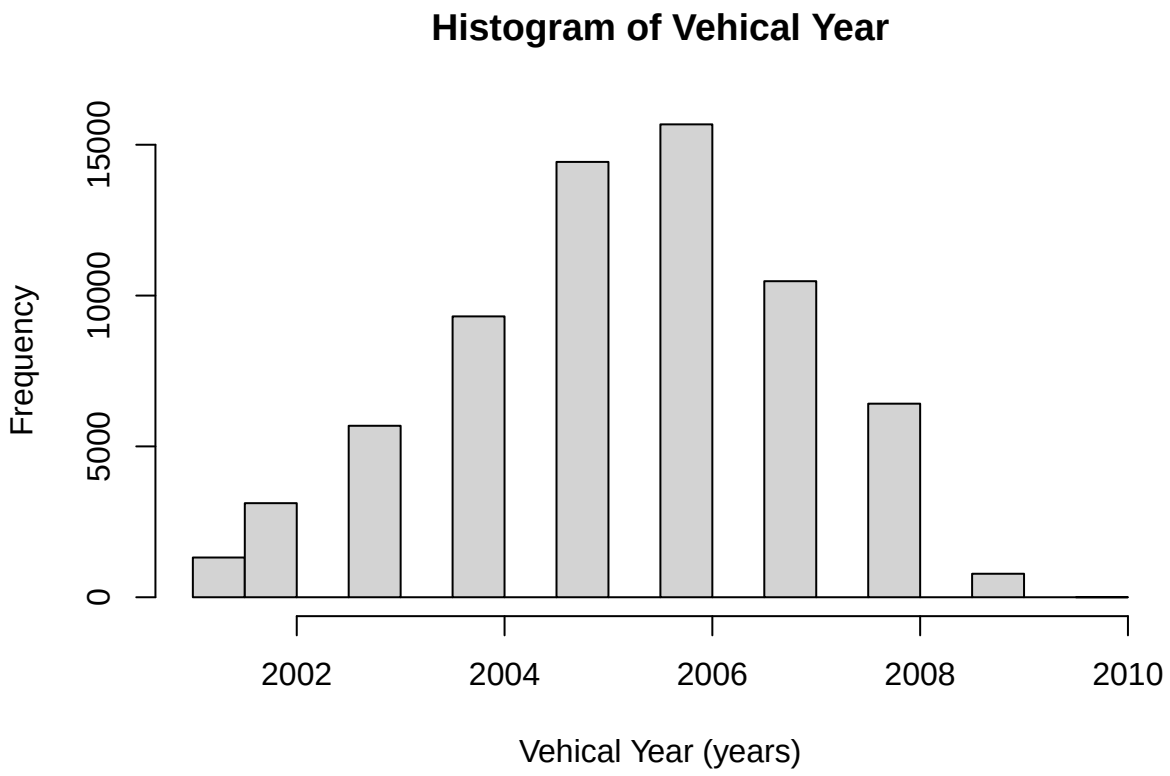
#Data Prepration: Checking the distribution and skweness of the numerical Variables

```
#Histogram of all the numerical variables
Car_num[,c(1:16)] %>%
  gather(key = Variable, value = Value) %>%
  ggplot() +
  geom_histogram(aes(x = Value), fill = "violet") +
  facet_wrap(~Variable, scales='free') +
  theme_classic() +
  theme(aspect.ratio = 0.5, axis.title = element_blank(), panel.grid = element_blank())
```

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

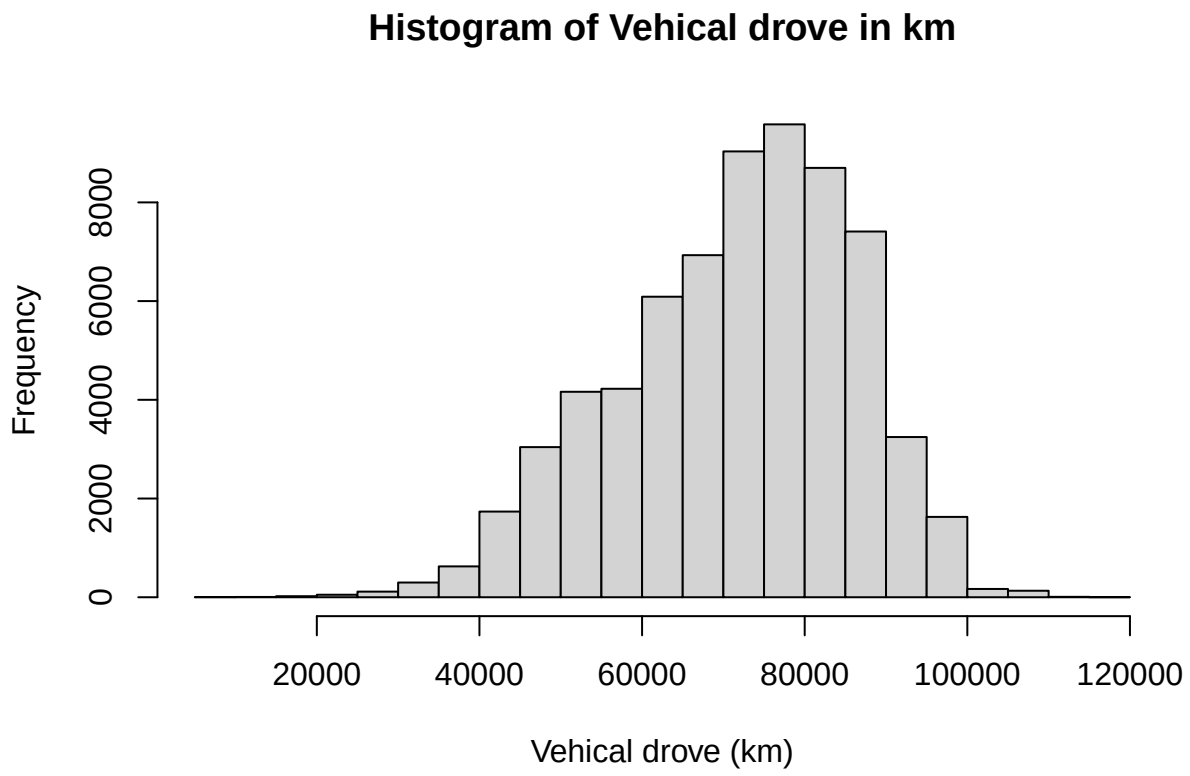



```
#From the above output, we can identify a bell curve distribution of data for maximum amount of the dat

#Create the histograms for all the variables to see the normal distribution.

#Variable "VehYear"
hist(Car_num$VehYear, breaks = 30, main = "Histogram of Vehical Year", xlab = "Vehical Year (years)", y
```

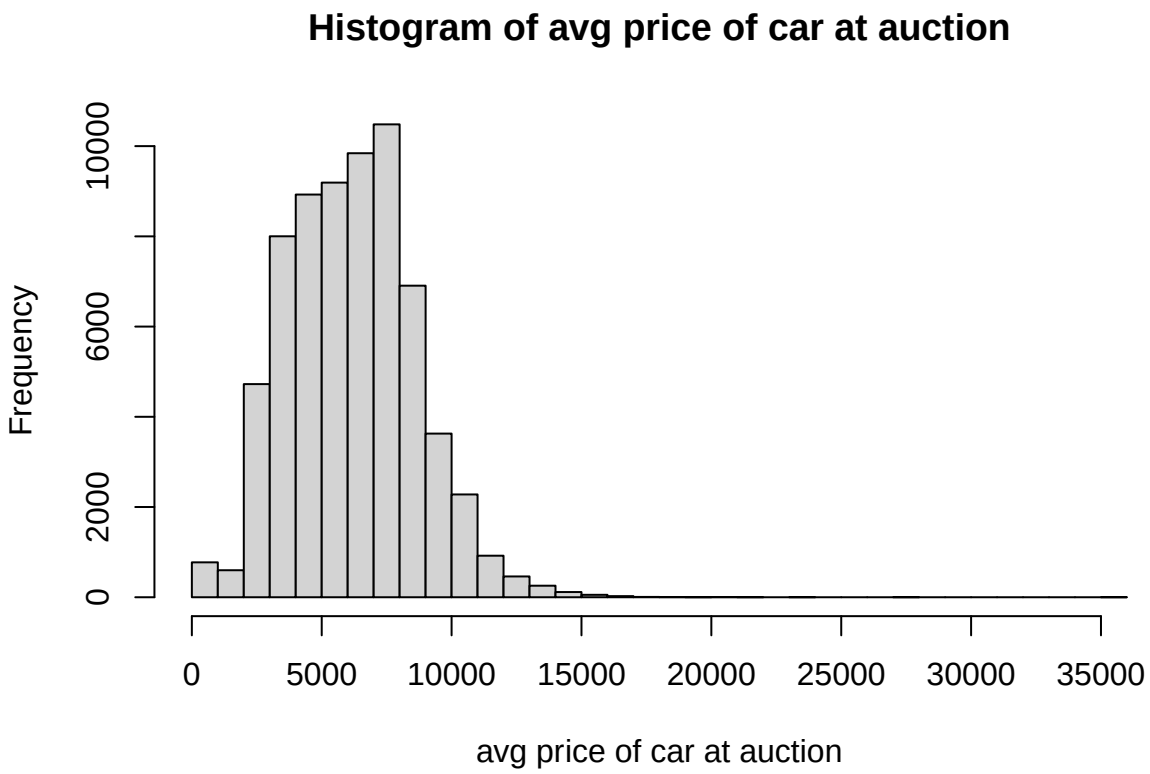



```
#Variable "VehicleAge"
hist(Car_num$VehicleAge, breaks = 30, main = "Histogram of Vehicle Age", xlab = "Vehical age (years)", y
```

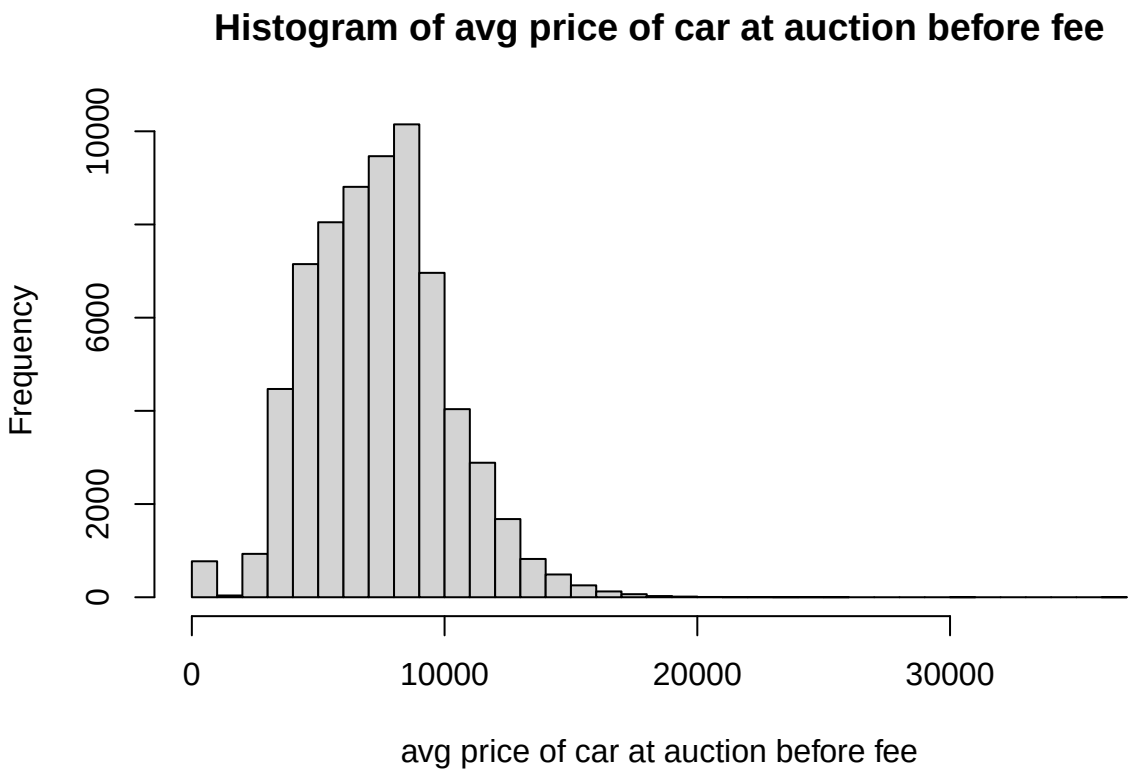

```
#Variable "VehOdo"
hist(Car_num$VehOdo, breaks = 30, main = "Histogram of Vehical drove in km", xlab = "Vehical drove (km)
```



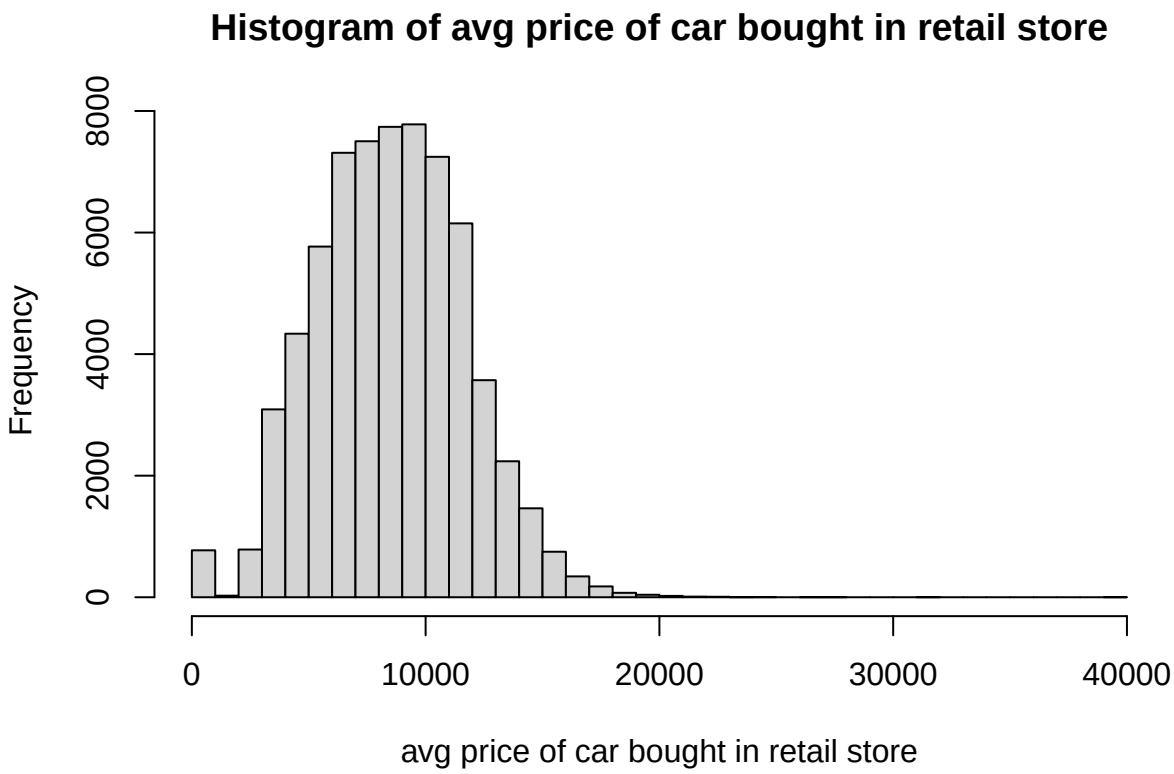

```
#Variable "MMRAcquisitionAuctionAveragePrice"
hist(Car_num$MMRAAAP, breaks = 30, main = "Histogram of avg price of car at auction", xlab = "avg price
```



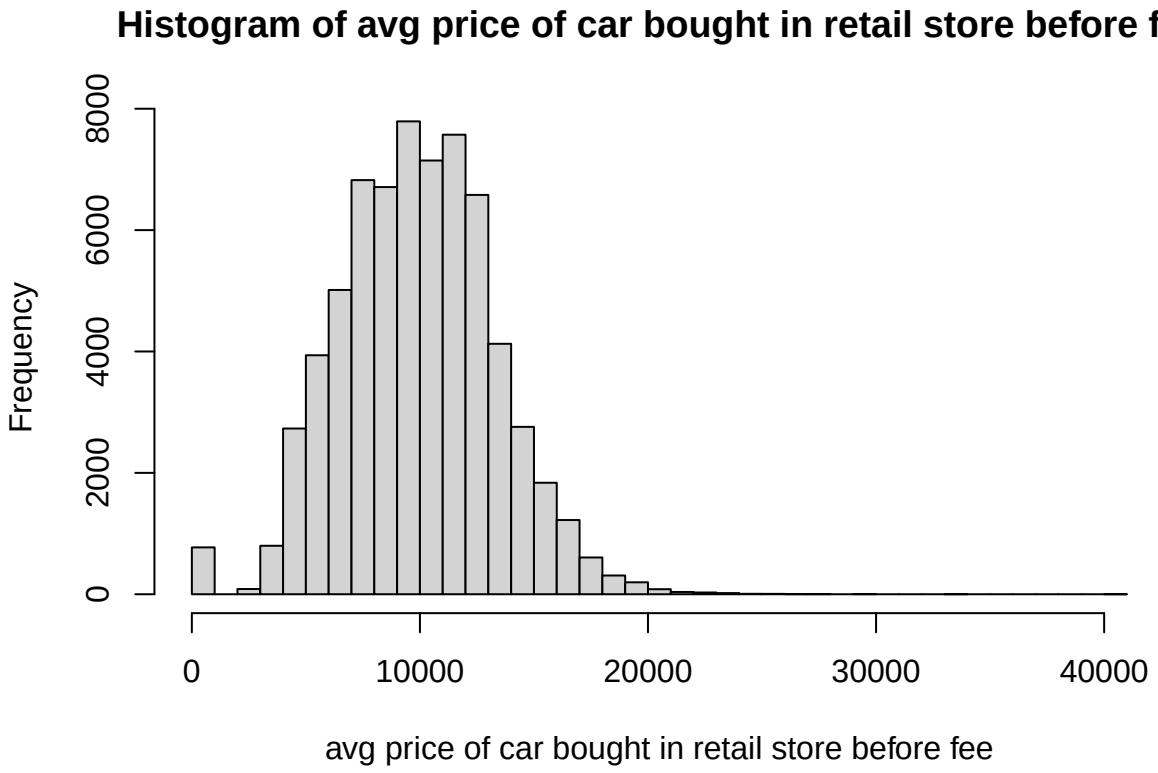

```
#Variable "MMRAcquisitionAuctionCleanPrice"
hist(Car_num$MMRAACP, breaks = 30, main = "Histogram of avg price of car at auction before fee", xlab =
```



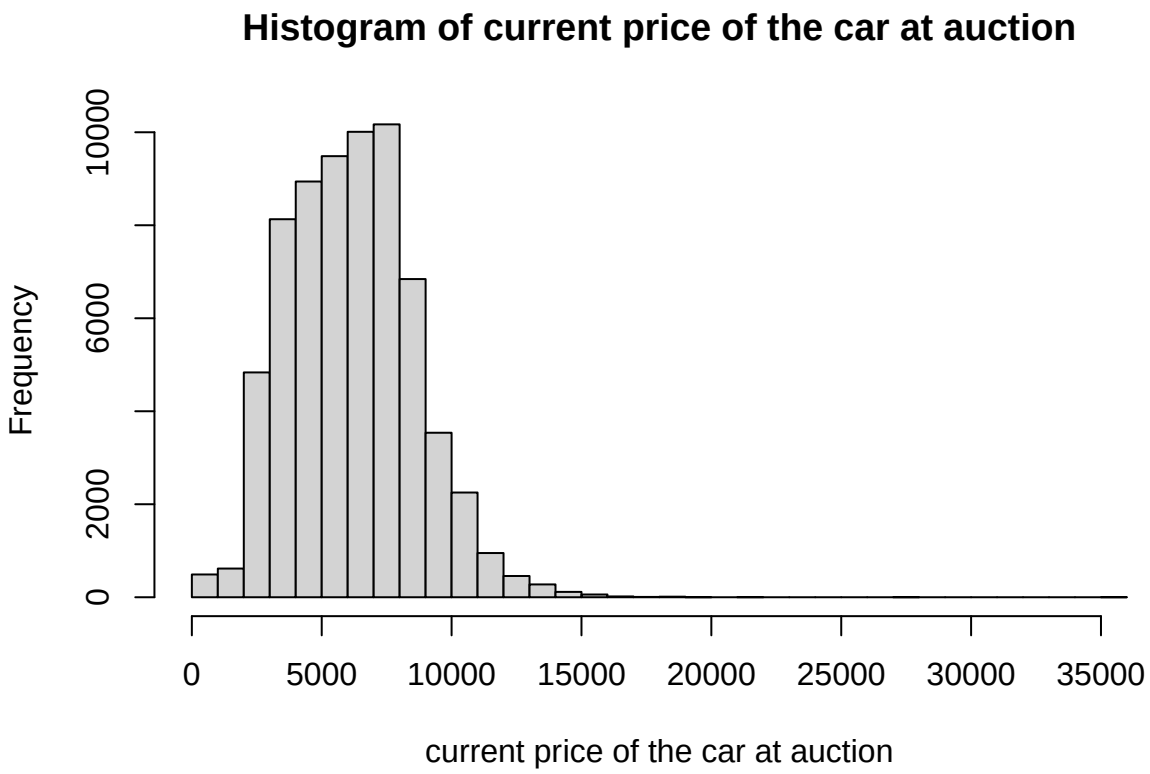

```
#Variable "MMRAcquisitionRetailAveragePrice"
hist(Car_num$MMRARAP, breaks = 30, main = "Histogram of avg price of car bought in retail store", xlab =
```



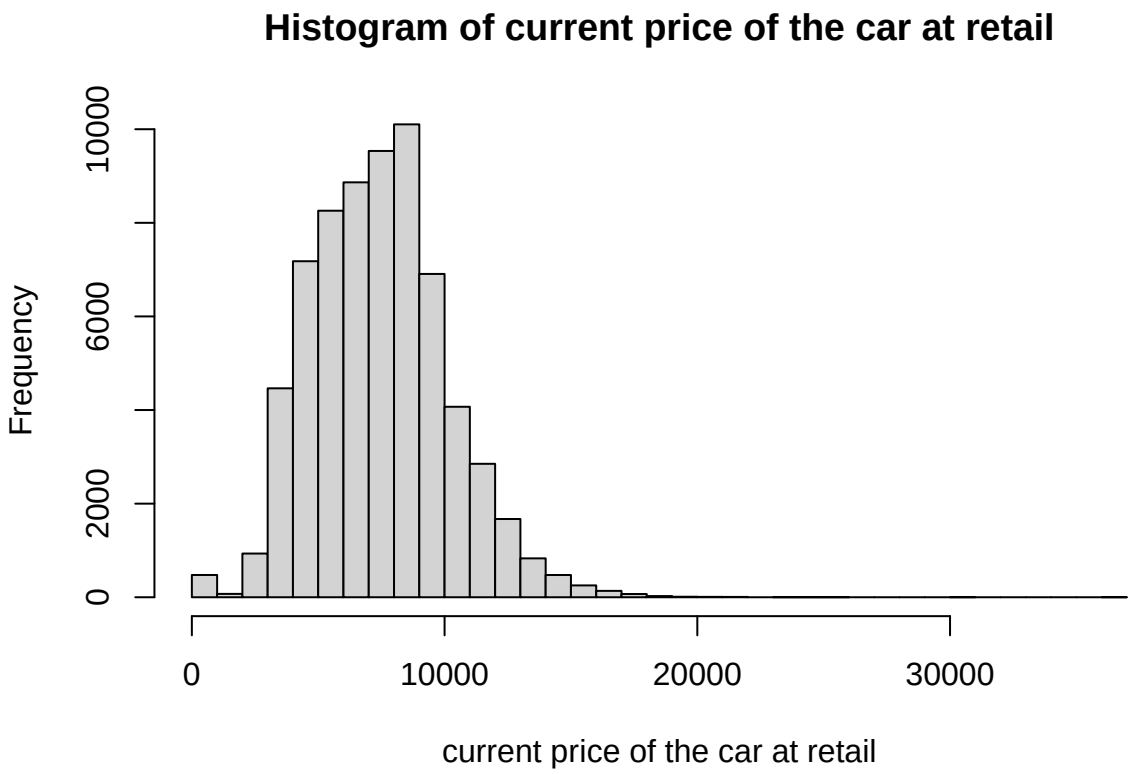

```
#Variable "MMRAcquisitonRetailCleanPrice"
hist(Car_num$MMRARCP, breaks = 30, main = "Histogram of avg price of car bought in retail store before
```



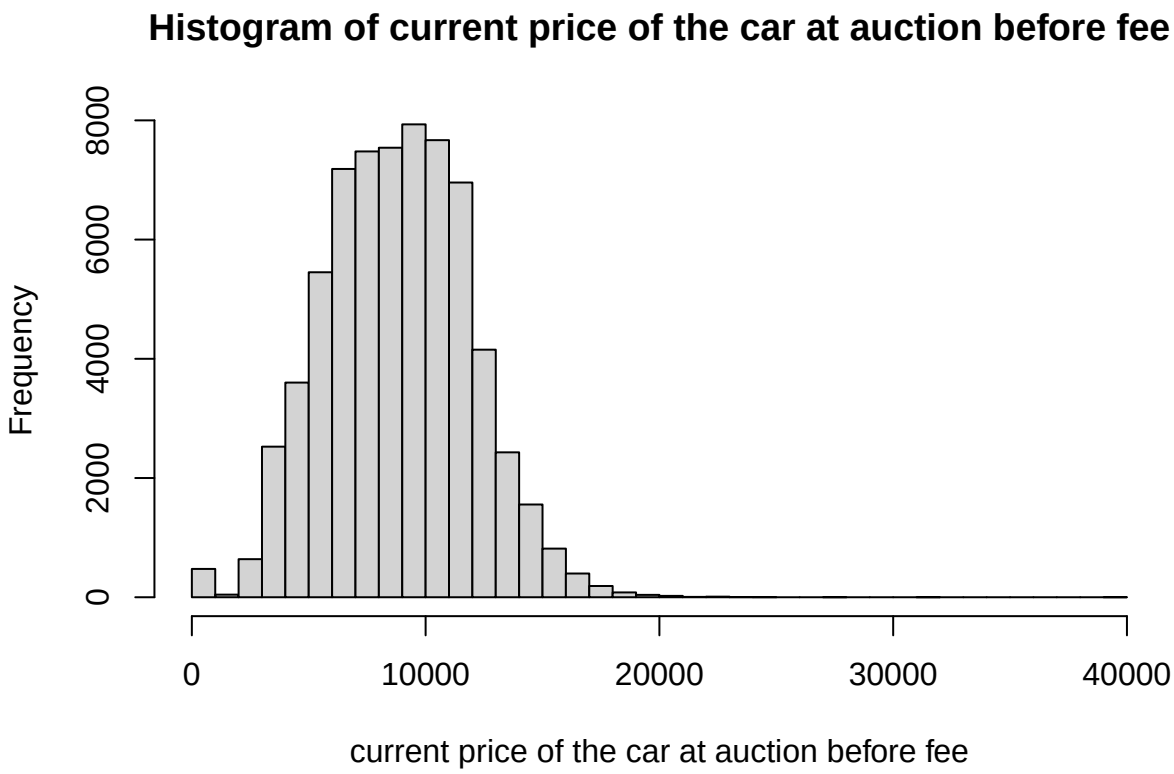

```
#Variable "MMRCurrentAuctionAveragePrice"
hist(Car_num$MMRCAAP, breaks = 30, main = "Histogram of current price of the car at auction", xlab = "cu
```



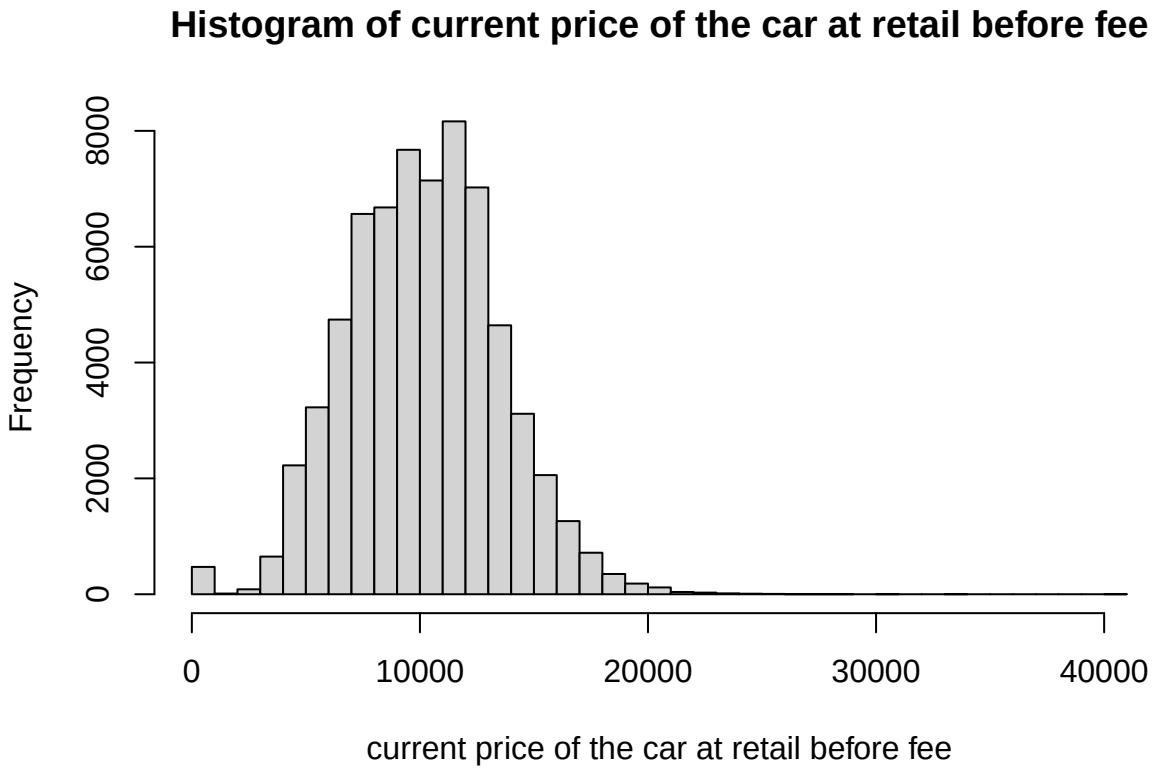

```
#Variable "MMRCurrentRetailAveragePrice"
hist(Car_num$MMRCACP, breaks = 30, main = "Histogram of current price of the car at retail", xlab = "cu
```



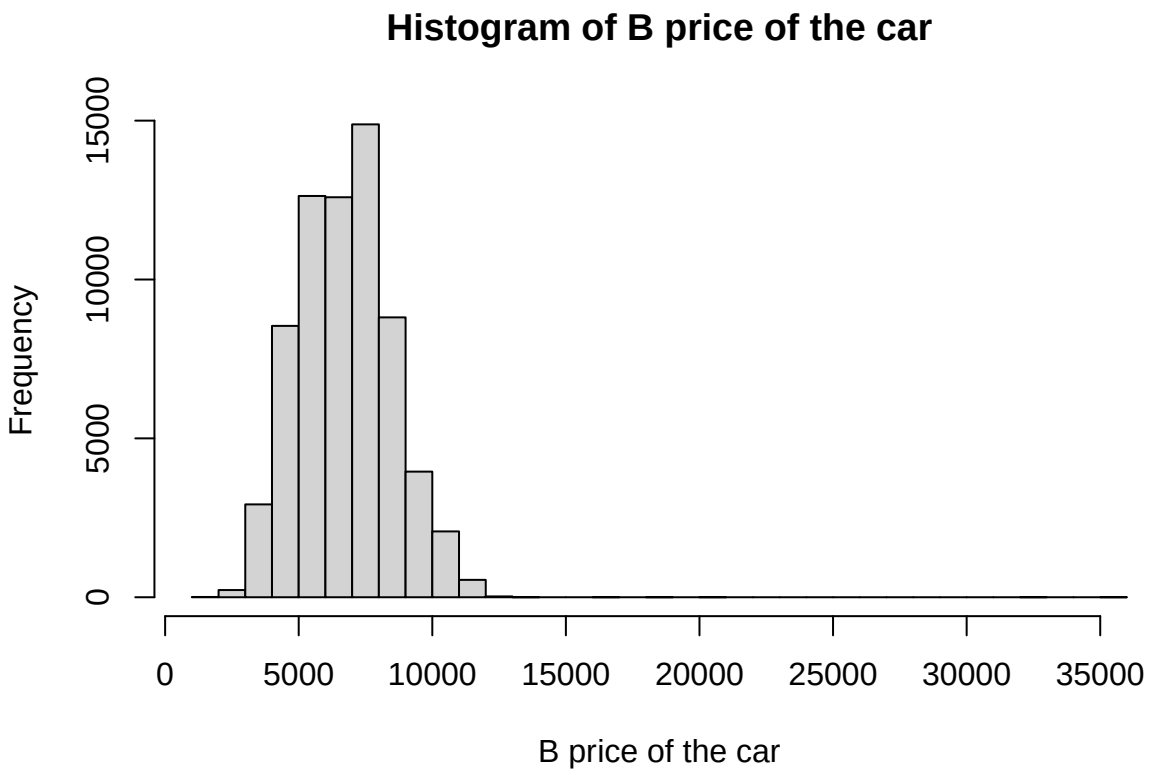

```
#Variable "MMRCurrentAuctionCleanPrice"
hist(Car_num$MMRCRAP, breaks = 30, main = "Histogram of current price of the car at auction before fee"
```



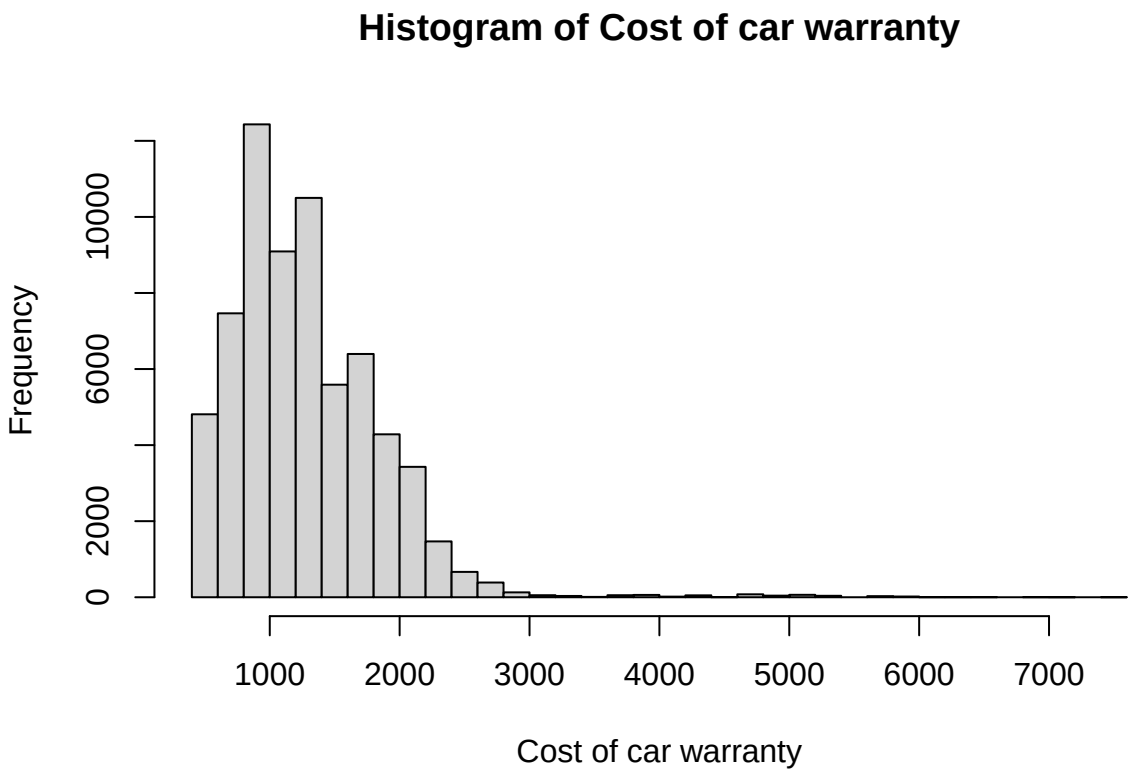

```
#Variable "MMRCurrentRetailCleanPrice"
hist(Car_num$MMRCRCP, breaks = 30, main = "Histogram of current price of the car at retail before fee",
```




```
#Variable "VehBCost"
hist(Car_num$VehBCost, breaks = 30, main = "Histogram of B price of the car", xlab = "B price of the ca
```




```
#Variable "WarrantyCost"
hist(Car_num$WarrantyCost, breaks = 30, main = "Histogram of Cost of car warranty", xlab = "Cost of car
```

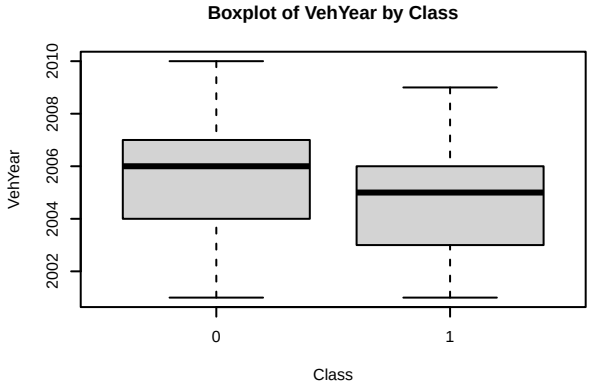



```
#check for the Skewness of the Numerical Variables
Car_num1 <- Car_num[,c(2:16)]
skewness_values <- sapply(Car_num1, skewness)
skewness_values #most of the variables have skewness values close to zero, indicating relatively symmet
```

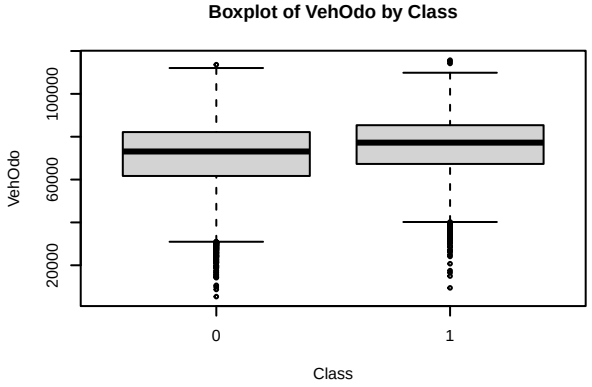
```
##   VehicleAge       VehOdo      MMRAAAP      MMRAACP      MMRARAP      MMRARCP
##   0.39260214  -0.43825949   0.33758581   0.35424369   0.12913203   0.09665368
##      MMRCAAP      MMRCACP      MMRCRAP      MMRCRCP     VehBCost WarrantyCost
##   0.40657244   0.43168631   0.11999095   0.11453034   0.32940399   1.91451261
##  WheelTypeID        BYRNO       VNZIP1
##   0.25648713   2.10593865  -0.12502354
```

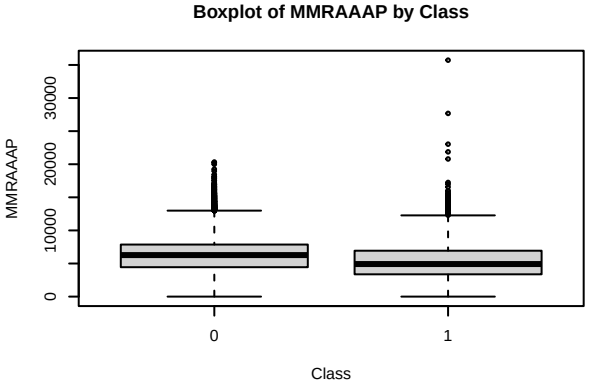
#Data Exploration 1: Looking for the outliers

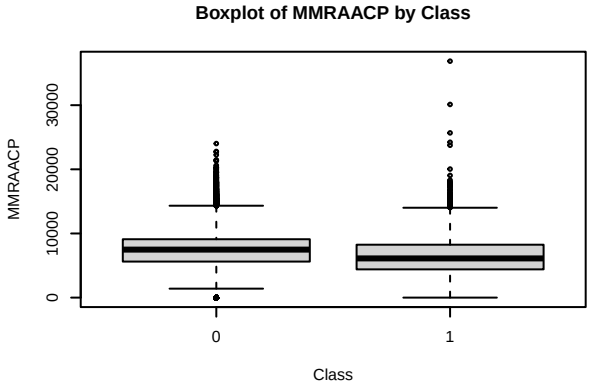
```
#Creating Box Plot for all the numerical variables to see the outliers.
my_data <- Car_num[,c(1:16,18)]
par(mfrow = c(2,2),cex = 0.5)
 for (i in 1:(ncol(my_data) - 1)) {
 # Create a boxplot for the current variable by "churn"
  boxplot(my_data[, i] ~ my_data[, ncol(my_data)],
          main = paste("Boxplot of", names(my_data)[i], "by", names(my_data)[ncol(my_data)]),
          xlab = names(my_data)[ncol(my_data)], ylab = names(my_data)[i])
} #warranty cost and Auction price fields have some outliers
```

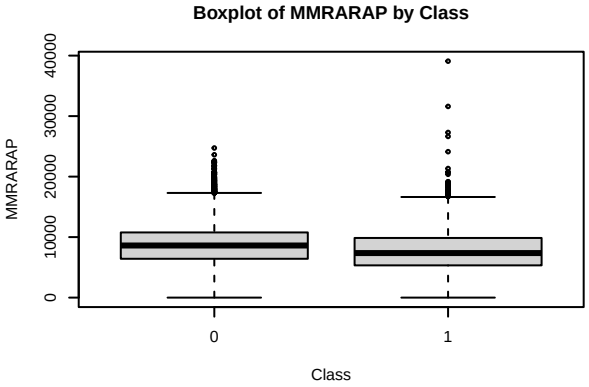

Boxplot of VehYear by Class
VehYear
2002
2004
2006
2008
2010
0
1
Class

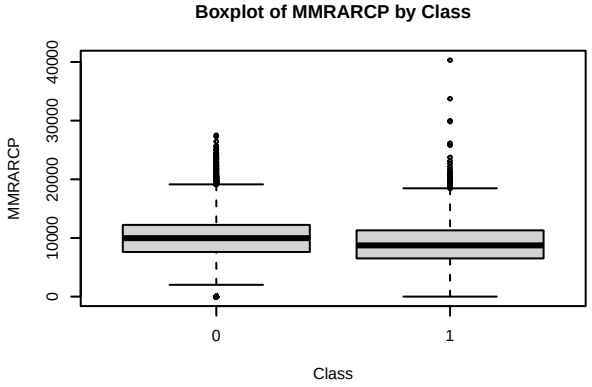

Boxplot of VehicleAge by Class
VehicleAge
0
2
4
6
8
0
1
Class

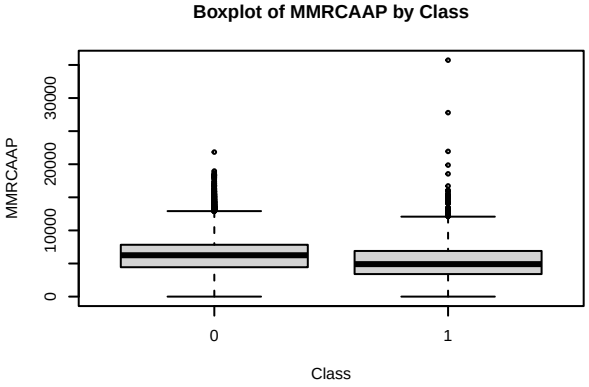

Boxplot of VehOdo by Class
VehOdo
20000
60000
100000
0
1
Class

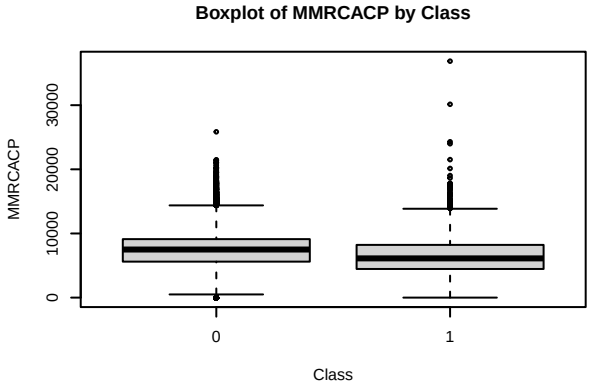

Boxplot of MMRAAAP by Class
MMRAAAP
0
10000
20000
30000
0
1
Class

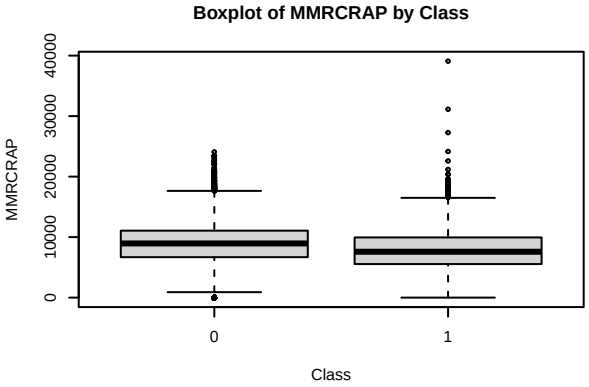

Boxplot of MMRAACP by Class
MMRAACP
0
10000
20000
30000
0
1
Class

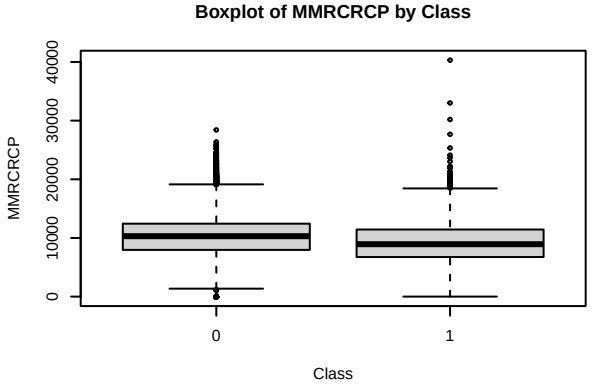

Boxplot of MMRARAP by Class
MMRARAP
0
10000
20000
30000
40000
0
1
Class

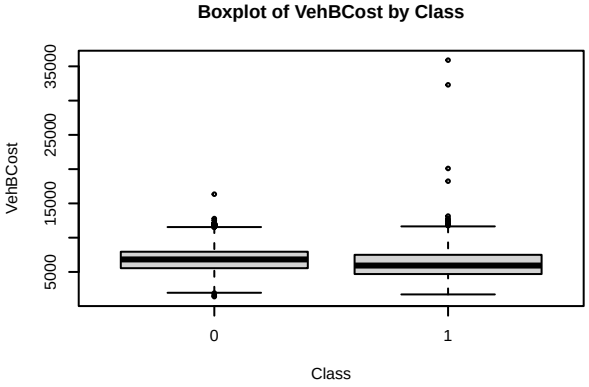

Boxplot of MMRARCP by Class
MMRARCP
0
10000
20000
30000
40000
0
1
Class

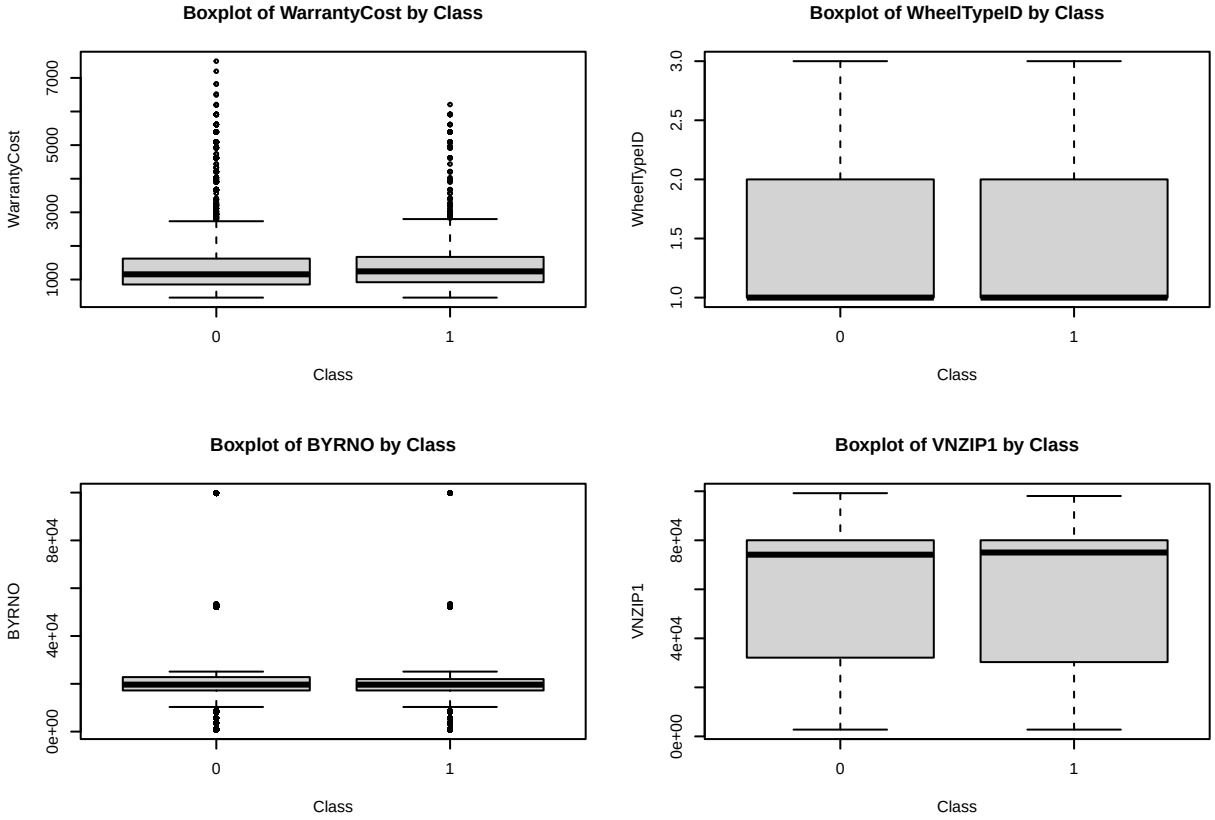

Boxplot of MMRCAAP by Class
MMRCAAP
0
10000
20000
30000
0
1
Class


Boxplot of MMRCACP by Class
MMRCACP
0
10000
20000
30000
0
1
Class


Boxplot of MMRCRAP by Class
MMRCRAP
0
10000
20000
30000
40000
0
1
Class

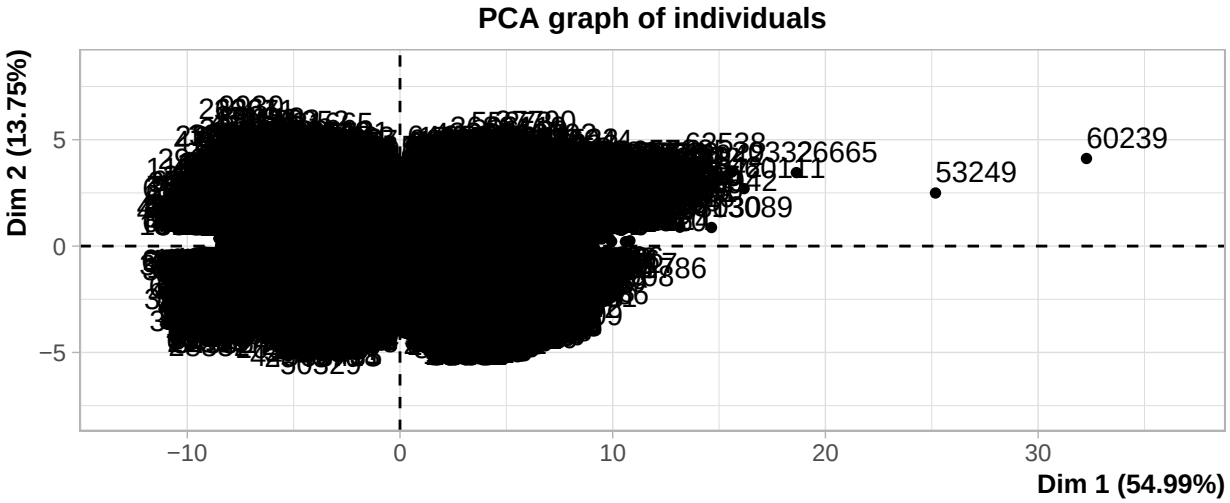

Boxplot of MMRCRCP by Class
MMRCRCP
0
10000
20000
30000
40000
0
1
Class


Boxplot of VehBCost by Class
VehBCost
5000
15000
25000
35000
0
1
Class

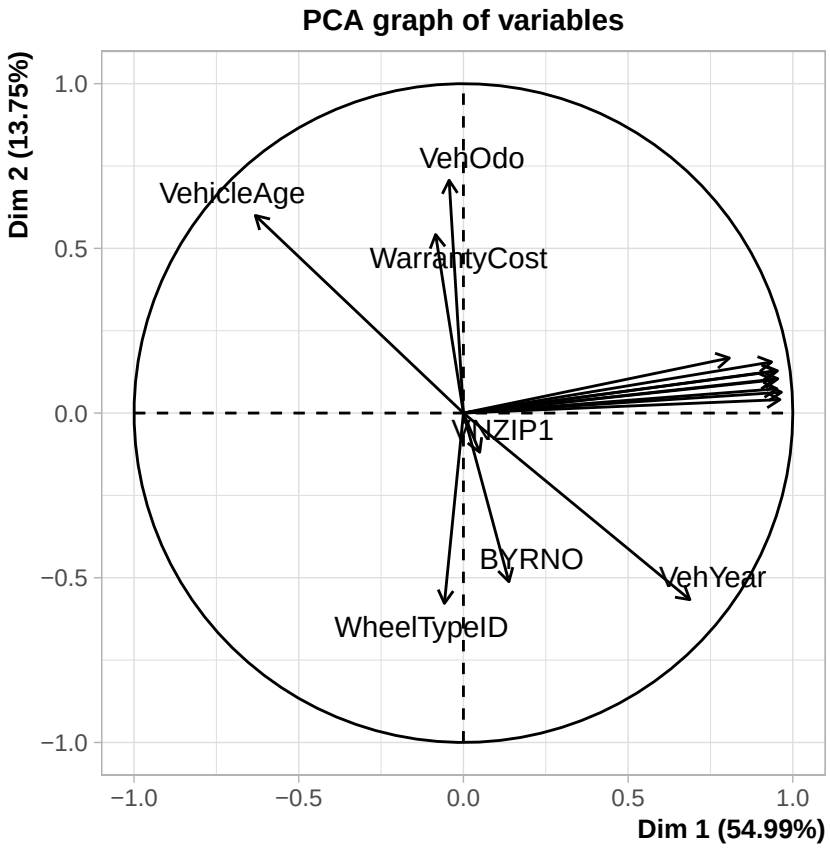


#Data Visualization (2):Determining the relative importance of the primary variables in the data set using principal component analysis.

```
library(FactoMineR)
pca <- PCA(Car_num[,c(1:16)])
```



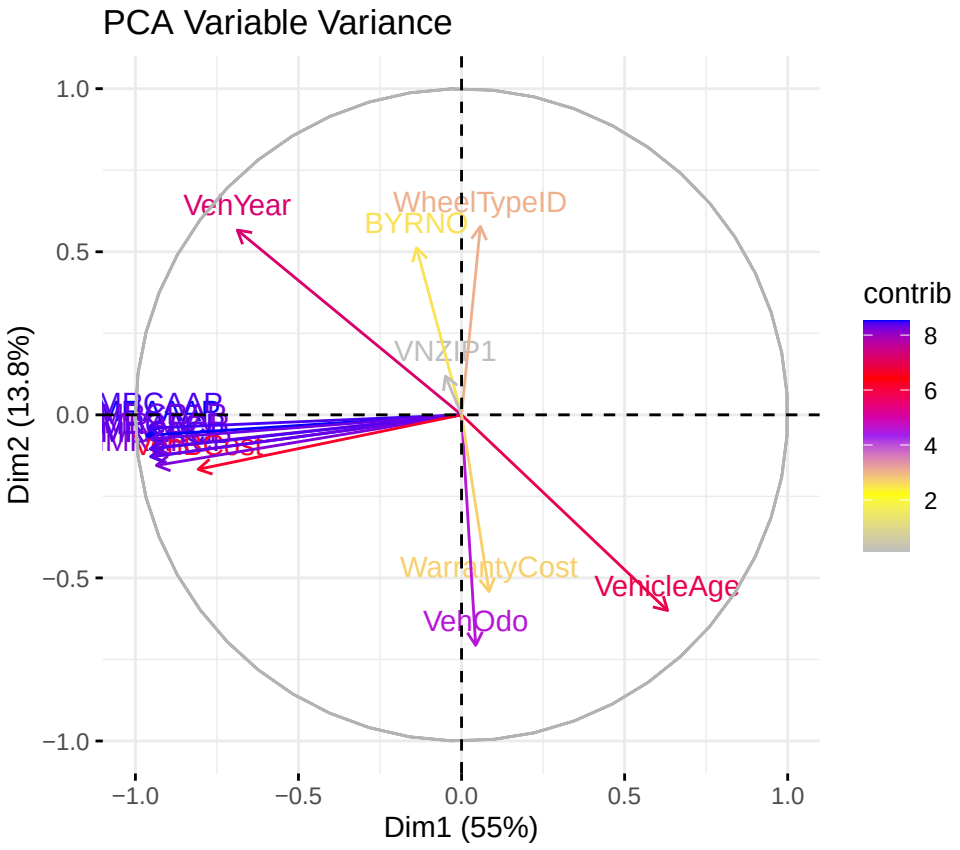

```
## Warning: ggrepel: 9 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```




```
pca <- prcomp(Car_num[,c(1:16)], scale = TRUE)
# extract loadings
loadings <- pca$rotation
# print loadings for the first two PCs
print(loadings[, 1:2])
```

```
##                     PC1         PC2
## VehYear      -0.23177506  0.38186047
## VehicleAge    0.21295700 -0.40434243
## VehOdo        0.01462315 -0.47634201
## MMRAAAP      -0.32545084 -0.04256459
## MMRAACP      -0.32144118 -0.08635540
## MMRARAP      -0.31736734 -0.06772236
## MMRARCP      -0.31545061 -0.10457147
## MMRCAAP      -0.32396970 -0.02751861
## MMRCACP      -0.32180762 -0.07019367
## MMRCRAP      -0.32092783 -0.04965003
## MMRCRCP      -0.31937621 -0.08642090
## VehBCost     -0.27220877 -0.11265913
## WarrantyCost  0.02842326 -0.36529776
## WheelTypeID   0.01944592  0.38944972
## BYRNO        -0.04655560  0.34517973
## VNZIP1       -0.01669116  0.08015470
```

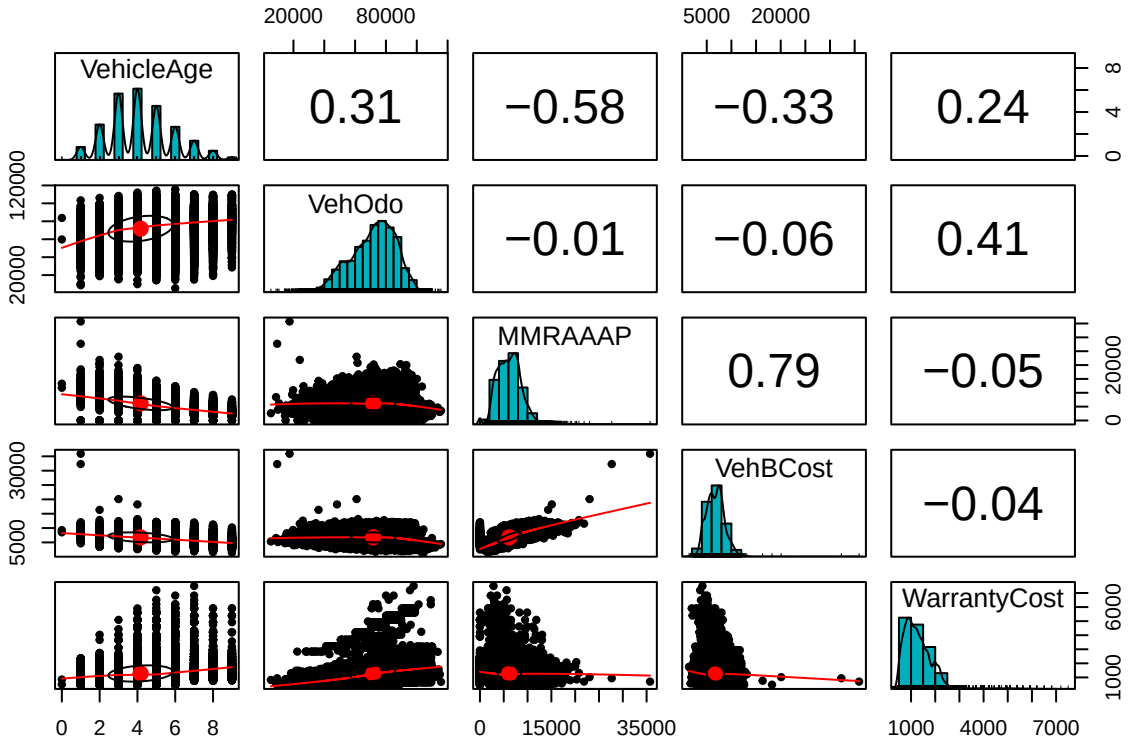
```
var <- get_pca_var(pca)
fviz_pca_var(pca, col.var="contrib",
gradient.cols = c("grey","yellow","purple","red","blue"),ggrepel = TRUE ) + labs( title = "PCA Variable
```




```
#PC1: The first principal component (PC1) explains the largest amount of variance in the data. The vari

#PC2: The second principal component (PC2) explains the second-largest amount of variance in the data,
```

#Data Exploration 1: Creating the scatterplot for important variables "vehicalage, Vehodo, Warrentycost, MMRAcquisitionAuctionAveragePrice, VehBCost"

```
pairs.panels(Car_num[,c(2,3,4,12,13)],
             method = "pearson", # correlation method
             hist.col = "#00AFBB",
             density = TRUE,  # show density plots
             ellipses = TRUE # show correlation ellipses
             )
```



#Important Categorical variables :Converting categorical values to Dummy variables for better usage

```
Auction_MANHEIM <- ifelse(Car_data$Auction == "MANHEIM", 1,0)
Auction_MANHEIM <- as.factor(Auction_MANHEIM)

WheelType_Covers  <- ifelse(Car_data$WheelType == "Covers", 1,0)
WheelType_Covers <- as.factor(WheelType_Covers)

WheelType_Alloy  <- ifelse(Car_data$WheelType == "Alloy", 1,0)
WheelType_Alloy <- as.factor(WheelType_Alloy)

Nationality_AMERICAN  <- ifelse(Car_data$Nationality == "AMERICAN", 1,0)
Nationality_AMERICAN <- as.factor(Nationality_AMERICAN)

Nationality_topline_Asian <- ifelse(Car_data$Nationality == "'TOP LINE ASIAN'", 1,0)
Nationality_topline_Asian <- as.factor(Nationality_topline_Asian)

Size_MEDIUM <- ifelse(Car_data$Size %in% c("MEDIUM", "MEDIUM SUV"), 1, 0)
Size_MEDIUM <- as.factor(Size_MEDIUM)

Size_Large <- ifelse(Car_data$Size %in% c("LARGE", "LARGE TRUCK", "LARGE SUV"), 1, 0)
Size_Large <- as.factor(Size_Large)

Size_small <- ifelse(Car_data$Size %in% c("COMPACT", "SMALL SUV", "SMALL TRUCK"), 1, 0)
Size_small <- as.factor(Size_small)

TopThreeAmerican_GM <- ifelse(Car_data$TopThreeAmericanName %in% c("GM"), 1, 0)
```

```
TopThreeAmerican_GM <- as.factor(TopThreeAmerican_GM)

TopThreeAmerican_CHRYSLER <- ifelse(Car_data$TopThreeAmericanName %in% c("CHRYSLER"), 1, 0)
TopThreeAmerican_CHRYSLER <- as.factor(TopThreeAmerican_CHRYSLER)

TopThreeAmerican_FORD <- ifelse(Car_data$TopThreeAmericanName %in% c("FORD"), 1, 0)
TopThreeAmerican_FORD <- as.factor(TopThreeAmerican_FORD)

VNST_West <- ifelse(Car_data$VNST %in% c("AK","CA","CO","HI","ID","MT","NV","OR","UT","WA","WY"), 1, 0)
VNST_West <- as.factor(VNST_West)

VNST_SouthWest <- ifelse(Car_data$VNST %in% c("AZ","NM","OK","TX","AL","AR","FL","GA","KY","LA","MS","N
VNST_SouthWest <- as.factor(VNST_SouthWest)

VNST_Northeast <- ifelse(Car_data$VNST %in% c("CT","DE","ME","MD","MA","NH","NJ","NY","PA","RI","VT"),
VNST_Northeast <- as.factor(VNST_Northeast)

VNST_Midwest <- ifelse(Car_data$VNST %in% c("IL","IN","IA","KS","MI","MN","MO","NE","ND","OH","SD","WI"
VNST_Midwest <- as.factor(VNST_Midwest)

Car_data2 <- data.frame(Car_data ,
Auction_MANHEIM = Auction_MANHEIM,
WheelType_Covers = WheelType_Covers,
WheelType_Alloy = WheelType_Alloy,
Nationality_AMERICAN = Nationality_AMERICAN,
Nationality_topline_Asian = Nationality_topline_Asian,
Size_MEDIUM = Size_MEDIUM,
Size_Large = Size_Large,
Size_small = Size_small,
TopThreeAmerican_GM =TopThreeAmerican_GM,
TopThreeAmerican_CHRYSLER = TopThreeAmerican_CHRYSLER,
TopThreeAmerican_FORD = TopThreeAmerican_FORD,
VNST_West = VNST_West,
VNST_SouthWest = VNST_SouthWest,
VNST_Northeast = VNST_Northeast,
VNST_Midwest = VNST_Midwest)
```

#Selecting the important variables for lm() and ANOVA

```
Car_data3 <- Car_data2[,c(3,4,5,13,14,31:46)]
t(t(names(Car_data3)))
```

```
##       [,1]
##  [1,] "VehicleAge"
##  [2,] "VehOdo"
##  [3,] "MMRAcquisitionAuctionAveragePrice"
##  [4,] "VehBCost"
##  [5,] "WarrantyCost"
##  [6,] "Class"
##  [7,] "Auction_MANHEIM"
##  [8,] "WheelType_Covers"
##  [9,] "WheelType_Alloy"
## [10,] "Nationality_AMERICAN"
```

```
## [11,] "Nationality_topline_Asian"
## [12,] "Size_MEDIUM"
## [13,] "Size_Large"
## [14,] "Size_small"
## [15,] "TopThreeAmerican_GM"
## [16,] "TopThreeAmerican_CHRYSLER"
## [17,] "TopThreeAmerican_FORD"
## [18,] "VNST_West"
## [19,] "VNST_SouthWest"
## [20,] "VNST_Northeast"
## [21,] "VNST_Midwest"
```

```
set.seed(2023)
Norm_model <- preProcess(Car_data3, method = c("center", "scale"))
Car_data3_norm <-predict(Norm_model,Car_data3)
head(Car_data3_norm)
```

```
##    VehicleAge      VehOdo MMRAcquisitionAuctionAveragePrice   VehBCost
## 1 -0.09873405 -1.37486769                        0.01424666 -0.1457925
## 2 -0.09873405  1.20887915                       -1.00863049 -1.7630331
## 3 -0.68438531 -0.03222013                        0.29962490  0.2384455
## 4 -1.27003656  0.80650887                        0.69956211  0.5251903
## 5 -1.27003656 -0.90611917                        1.07544602  0.2355781
## 6 -0.09873405  0.37265236                       -0.37346007 -0.7192821
##   WarrantyCost      Class Auction_MANHEIM WheelType_Covers WheelType_Alloy
## 1   -0.3331889 -0.3248591               1                1               0
## 2   -0.5110579 -0.3248591               0                1               0
## 3    1.2002838 -0.3248591               1                0               1
## 4    1.5076691 -0.3248591               1                1               0
## 5    0.3817409 -0.3248591               0                0               1
## 6   -1.2259878 -0.3248591               1                0               1
##   Nationality_AMERICAN Nationality_topline_Asian Size_MEDIUM Size_Large
## 1                    1                         0           1          0
## 2                    1                         0           1          0
## 3                    1                         0           0          1
## 4                    1                         0           0          1
## 5                    1                         0           0          1
## 6                    0                         0           1          0
##   Size_small TopThreeAmerican_GM TopThreeAmerican_CHRYSLER
## 1          0                   0                         1
## 2          0                   0                         0
## 3          0                   1                         0
## 4          0                   1                         0
## 5          0                   1                         0
## 6          0                   0                         0
##   TopThreeAmerican_FORD VNST_West VNST_SouthWest VNST_Northeast VNST_Midwest
## 1                     0         0              1              0            0
## 2                     1         0              1              0            0
## 3                     0         1              0              0            0
## 4                     0         1              0              0            0
## 5                     0         0              1              0            0
## 6                     0         0              1              0            0
```

```
str(Car_data3_norm)
```

```
## 'data.frame':    67211 obs. of  21 variables:
##  $ VehicleAge                       : num  -0.0987 -0.0987 -0.6844 -1.27 -1.27 ...
##  $ VehOdo                           : num  -1.3749 1.2089 -0.0322 0.8065 -0.9061 ...
##  $ MMRAcquisitionAuctionAveragePrice: num  0.0142 -1.0086 0.2996 0.6996 1.0754 ...
##  $ VehBCost                         : num  -0.146 -1.763 0.238 0.525 0.236 ...
##  $ WarrantyCost                     : num  -0.333 -0.511 1.2 1.508 0.382 ...
##  $ Class                            : num  -0.325 -0.325 -0.325 -0.325 -0.325 ...
##  $ Auction_MANHEIM                  : Factor w/ 2 levels "0","1": 2 1 2 2 1 2 2 1 2 2 ...
##  $ WheelType_Covers                 : Factor w/ 2 levels "0","1": 2 2 1 2 1 1 1 2 1 2 ...
##  $ WheelType_Alloy                  : Factor w/ 2 levels "0","1": 1 1 2 1 2 2 2 1 2 1 ...
##  $ Nationality_AMERICAN             : Factor w/ 2 levels "0","1": 2 2 2 2 2 1 1 2 2 2 ...
##  $ Nationality_topline_Asian        : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Size_MEDIUM                      : Factor w/ 2 levels "0","1": 2 2 1 1 1 2 1 1 1 1 ...
##  $ Size_Large                       : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 1 1 1 1 ...
##  $ Size_small                       : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 2 1 1 ...
##  $ TopThreeAmerican_GM              : Factor w/ 2 levels "0","1": 1 1 2 2 2 1 1 2 1 2 ...
##  $ TopThreeAmerican_CHRYSLER        : Factor w/ 2 levels "0","1": 2 1 1 1 1 1 1 1 1 1 ...
##  $ TopThreeAmerican_FORD            : Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 1 2 1 ...
##  $ VNST_West                        : Factor w/ 2 levels "0","1": 1 1 2 2 1 1 2 1 1 1 ...
##  $ VNST_SouthWest                   : Factor w/ 2 levels "0","1": 2 2 1 1 2 2 1 2 2 2 ...
##  $ VNST_Northeast                   : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ VNST_Midwest                     : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```
# Fit a linear regression model
lm_model <- lm(Class ~ ., data = Car_data3_norm)
summary(lm_model)
```

```
##
## Call:
## lm(formula = Class ~ ., data = Car_data3_norm)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1019 -0.4106 -0.2779 -0.1366  4.2492
##
## Coefficients: (2 not defined because of singularities)
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                        0.028062   0.043312   0.648   0.5170
## VehicleAge                         0.145643   0.006096  23.892  < 2e-16 ***
## VehOdo                             0.031647   0.004575   6.917 4.65e-12 ***
## MMRAcquisitionAuctionAveragePrice  0.041158   0.007832   5.255 1.48e-07 ***
## VehBCost                          -0.075791   0.007162 -10.583  < 2e-16 ***
## WarrantyCost                       0.025179   0.005495   4.582 4.60e-06 ***
## Auction_MANHEIM1                   0.062026   0.007778   7.974 1.56e-15 ***
## WheelType_Covers1                 -0.074757   0.036996  -2.021   0.0433 *
## WheelType_Alloy1                  -0.050024   0.036770  -1.360   0.1737
## Nationality_AMERICAN1              0.013633   0.015615   0.873   0.3826
## Nationality_topline_Asian1        -0.012060   0.021926  -0.550   0.5823
## Size_MEDIUM1                       0.022558   0.011513   1.959   0.0501 .
## Size_Large1                       -0.012699   0.014540  -0.873   0.3825
## Size_small1                        0.087511   0.017774   4.924 8.52e-07 ***
```

```
## TopThreeAmerican_GM1              -0.089238   0.012292  -7.260 3.92e-13 ***
## TopThreeAmerican_CHRYSLER1         0.001911   0.012226   0.156   0.8758
## TopThreeAmerican_FORD1                   NA         NA      NA       NA
## VNST_West1                        -0.013518   0.020283  -0.666   0.5051
## VNST_SouthWest1                   -0.001202   0.018852  -0.064   0.9492
## VNST_Northeast1                    0.114403   0.027493   4.161 3.17e-05 ***
## VNST_Midwest1                            NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9815 on 67192 degrees of freedom
## Multiple R-squared:  0.03691,    Adjusted R-squared:  0.03666
## F-statistic: 143.1 on 18 and 67192 DF,  p-value: < 2.2e-16
```

```
#Perform ANOVA
anova_results <- aov(Class ~ ., data = Car_data3_norm)
summary(anova_results)
```

```
##                                    Df Sum Sq Mean Sq  F value   Pr(>F)
## VehicleAge                          1   1930  1929.8 2003.257  < 2e-16 ***
## VehOdo                              1     97    96.8  100.531  < 2e-16 ***
## MMRAcquisitionAuctionAveragePrice   1     47    47.0   48.803 2.86e-12 ***
## VehBCost                            1    176   176.2  182.914  < 2e-16 ***
## WarrantyCost                        1      0     0.0    0.032 0.857683
## Auction_MANHEIM                     1     69    68.6   71.234  < 2e-16 ***
## WheelType_Covers                    1      8     8.2    8.543 0.003469 **
## WheelType_Alloy                     1      2     1.6    1.617 0.203498
## Nationality_AMERICAN                1      1     0.6    0.592 0.441577
## Nationality_topline_Asian           1      0     0.0    0.000 0.991082
## Size_MEDIUM                         1      0     0.0    0.028 0.867886
## Size_Large                          1     23    22.8   23.649 1.16e-06 ***
## Size_small                          1      8     7.6    7.848 0.005088 **
## TopThreeAmerican_GM                 1     89    89.1   92.465  < 2e-16 ***
## TopThreeAmerican_CHRYSLER           1      0     0.0    0.023 0.878753
## VNST_West                           1      3     3.1    3.246 0.071622 .
## VNST_SouthWest                      1     13    12.8   13.311 0.000264 ***
## VNST_Northeast                      1     17    16.7   17.315 3.17e-05 ***
## Residuals                       67192  64729     1.0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

#———————————— Second half stepwise selection , lm, ANOVA and PCA ————————————#

#Features Backward selection.

```
t(t(names(Car_data2)))
```

```
##       [,1]
##  [1,] "PurchDate"
##  [2,] "VehYear"
##  [3,] "VehicleAge"
##  [4,] "VehOdo"
##  [5,] "MMRAcquisitionAuctionAveragePrice"
```

```
##  [6,] "MMRAcquisitionAuctionCleanPrice"
##  [7,] "MMRAcquisitionRetailAveragePrice"
##  [8,] "MMRAcquisitonRetailCleanPrice"
##  [9,] "MMRCurrentAuctionAveragePrice"
## [10,] "MMRCurrentAuctionCleanPrice"
## [11,] "MMRCurrentRetailAveragePrice"
## [12,] "MMRCurrentRetailCleanPrice"
## [13,] "VehBCost"
## [14,] "WarrantyCost"
## [15,] "Auction"
## [16,] "Make"
## [17,] "Model"
## [18,] "Trim"
## [19,] "SubModel"
## [20,] "Color"
## [21,] "Transmission"
## [22,] "WheelTypeID"
## [23,] "WheelType"
## [24,] "Nationality"
## [25,] "Size"
## [26,] "TopThreeAmericanName"
## [27,] "BYRNO"
## [28,] "VNZIP1"
## [29,] "VNST"
## [30,] "IsOnlineSale"
## [31,] "Class"
## [32,] "Auction_MANHEIM"
## [33,] "WheelType_Covers"
## [34,] "WheelType_Alloy"
## [35,] "Nationality_AMERICAN"
## [36,] "Nationality_topline_Asian"
## [37,] "Size_MEDIUM"
## [38,] "Size_Large"
## [39,] "Size_small"
## [40,] "TopThreeAmerican_GM"
## [41,] "TopThreeAmerican_CHRYSLER"
## [42,] "TopThreeAmerican_FORD"
## [43,] "VNST_West"
## [44,] "VNST_SouthWest"
## [45,] "VNST_Northeast"
## [46,] "VNST_Midwest"
```

```
Car_data4 <- Car_data2[,c(1:14,22,27,28,30:46)]
Car_data4$IsOnlineSale <- as.factor(Car_data4$IsOnlineSale)

set.seed(2023)
Norm_model <- preProcess(Car_data4, method = c("center", "scale"))
Car_data4_norm <-predict(Norm_model,Car_data4)
head(Car_data4_norm)
```

```
##    PurchDate    VehYear  VehicleAge      VehOdo
## 1  1.4711469  0.3746768 -0.09873405 -1.37486769
## 2 -1.1244899 -0.2040693 -0.09873405  1.20887915
## 3 -0.8207957  0.3746768 -0.68438531 -0.03222013
```

```
## 4  1.2386310  1.5321689 -1.27003656  0.80650887
## 5 -1.4186937  0.9534228 -1.27003656 -0.90611917
## 6  1.5945227  0.3746768 -0.09873405  0.37265236
##   MMRAcquisitionAuctionAveragePrice MMRAcquisitionAuctionCleanPrice
## 1                        0.01424666                      -0.1289098
## 2                       -1.00863049                      -0.9683562
## 3                        0.29962490                       0.3819776
## 4                        0.69956211                       0.5573073
## 5                        1.07544602                       0.9867916
## 6                       -0.37346007                      -0.4681508
##   MMRAcquisitionRetailAveragePrice MMRAcquisitonRetailCleanPrice
## 1                        0.33813114                    0.15664076
## 2                       -1.28579183                   -1.24996469
## 3                       -0.18690136                   -0.08005902
## 4                        1.00964009                    0.92495466
## 5                        0.46463354                    0.44387195
## 6                        0.07022507                   -0.02332200
##   MMRCurrentAuctionAveragePrice MMRCurrentAuctionCleanPrice
## 1                    -0.3402254                  -0.4027653
## 2                    -1.0213281                  -0.9882135
## 3                     0.2889603                   0.4179076
## 4                     0.6733939                   0.4757803
## 5                     0.4896239                   0.4156674
## 6                    -0.7856410                  -0.9049514
##   MMRCurrentRetailAveragePrice MMRCurrentRetailCleanPrice   VehBCost
## 1                  -0.09879576                 -0.1112094 -0.1457925
## 2                  -1.40444318                 -1.3684959 -1.7630331
## 3                  -0.29188902                 -0.1381340  0.2384455
## 4                   1.02930954                  0.8202595  0.5251903
## 5                  -0.12147450                 -0.1402517  0.2355781
## 6                  -0.61943482                 -0.5559186 -0.7192821
##   WarrantyCost WheelTypeID      BYRNO     VNZIP1 IsOnlineSale      Class
## 1   -0.3331889   0.9719537  2.8262824 -1.0011894            0 -0.3248591
## 2   -0.5110579   0.9719537 -0.2162877  0.7842112            0 -0.3248591
## 3    1.2002838  -0.9465773 -0.1359796  0.8323400            0 -0.3248591
## 4    1.5076691   0.9719537 -0.1074110  1.3887892            0 -0.3248591
## 5    0.3817409  -0.9465773 -0.2162877  0.7203457            0 -0.3248591
## 6   -1.2259878  -0.9465773  0.9898373 -1.1766281            0 -0.3248591
##   Auction_MANHEIM WheelType_Covers WheelType_Alloy Nationality_AMERICAN
## 1               1                1               0                    1
## 2               0                1               0                    1
## 3               1                0               1                    1
## 4               1                1               0                    1
## 5               0                0               1                    1
## 6               1                0               1                    0
##   Nationality_topline_Asian Size_MEDIUM Size_Large Size_small
## 1                         0           1          0          0
## 2                         0           1          0          0
## 3                         0           0          1          0
## 4                         0           0          1          0
## 5                         0           0          1          0
## 6                         0           1          0          0
##   TopThreeAmerican_GM TopThreeAmerican_CHRYSLER TopThreeAmerican_FORD VNST_West
## 1                   0                         1                     0         0
```

```
## 2                  0                         0                    1         0
## 3                  1                         0                    0         1
## 4                  1                         0                    0         1
## 5                  1                         0                    0         0
## 6                  0                         0                    0         0
##   VNST_SouthWest VNST_Northeast VNST_Midwest
## 1              1              0            0
## 2              1              0            0
## 3              0              0            0
## 4              0              0            0
## 5              1              0            0
## 6              1              0            0
```

```
set.seed(2023)
# Set up repeated k-fold cross-validation
train.control <- trainControl(method = "cv", number = 15)
# Train the model
step.model <- train(Class ~ ., data = Car_data4_norm,
                    method = "leapBackward",
                    tuneGrid = data.frame(nvmax = 1:5),
                    trControl = train.control
                    )
```

```
## Warning in train.default(x, y, weights = w, ...): You are trying to do
## regression and your outcome only has two possible values Are you trying to do
## classification? If so, use a 2 level factor as your outcome column.

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:
```

```
## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:
```

```
step.model$results
```

```
##   nvmax      RMSE   Rsquared       MAE     RMSESD  RsquaredSD      MAESD
## 1     1 0.9856980 0.02810626 0.5712031 0.02156548 0.004449268 0.01335340
## 2     2 0.9844094 0.03068330 0.5702616 0.02130328 0.005146272 0.01331122
## 3     3 0.9833835 0.03270743 0.5707027 0.02129592 0.004969672 0.01309015
## 4     4 0.9830785 0.03330780 0.5700636 0.02132791 0.005033805 0.01307996
## 5     5 0.9824739 0.03452076 0.5705124 0.02167744 0.005189205 0.01340507
```

```
summary(step.model$finalModel)
```

```
## Subset selection object
## 33 Variables  (and intercept)
##                                   Forced in Forced out
## PurchDate                             FALSE      FALSE
## VehYear                               FALSE      FALSE
## VehicleAge                            FALSE      FALSE
## VehOdo                                FALSE      FALSE
## MMRAcquisitionAuctionAveragePrice     FALSE      FALSE
## MMRAcquisitionAuctionCleanPrice       FALSE      FALSE
## MMRAcquisitionRetailAveragePrice      FALSE      FALSE
## MMRAcquisitonRetailCleanPrice         FALSE      FALSE
## MMRCurrentAuctionAveragePrice         FALSE      FALSE
## MMRCurrentAuctionCleanPrice           FALSE      FALSE
## MMRCurrentRetailAveragePrice          FALSE      FALSE
## MMRCurrentRetailCleanPrice            FALSE      FALSE
## VehBCost                              FALSE      FALSE
## WarrantyCost                          FALSE      FALSE
## WheelTypeID                           FALSE      FALSE
## BYRNO                                 FALSE      FALSE
## VNZIP1                                FALSE      FALSE
## IsOnlineSale1                         FALSE      FALSE
## Auction_MANHEIM1                      FALSE      FALSE
## WheelType_Covers1                     FALSE      FALSE
## Nationality_AMERICAN1                 FALSE      FALSE
## Nationality_topline_Asian1            FALSE      FALSE
## Size_MEDIUM1                          FALSE      FALSE
## Size_Large1                           FALSE      FALSE
## Size_small1                           FALSE      FALSE
## TopThreeAmerican_GM1                  FALSE      FALSE
## TopThreeAmerican_CHRYSLER1            FALSE      FALSE
## VNST_West1                            FALSE      FALSE
## VNST_SouthWest1                       FALSE      FALSE
## VNST_Northeast1                       FALSE      FALSE
## WheelType_Alloy1                      FALSE      FALSE
## TopThreeAmerican_FORD1                FALSE      FALSE
## VNST_Midwest1                         FALSE      FALSE
## 1 subsets of each size up to 6
## Selection Algorithm: backward
##          PurchDate VehYear VehicleAge VehOdo MMRAcquisitionAuctionAveragePrice
## 1  ( 1 ) " "       " "     "*"        " "    " "
## 2  ( 1 ) " "       " "     "*"        " "    " "
```

```
## 3  ( 1 ) " "       " "     "*"        "*"    " "
## 4  ( 1 ) " "       " "     "*"        "*"    " "
## 5  ( 1 ) " "       " "     "*"        "*"    " "
## 6  ( 1 ) " "       " "     "*"        "*"    " "
##          MMRAcquisitionAuctionCleanPrice MMRAcquisitionRetailAveragePrice
## 1  ( 1 ) " "                               " "
## 2  ( 1 ) " "                               " "
## 3  ( 1 ) " "                               " "
## 4  ( 1 ) " "                               " "
## 5  ( 1 ) " "                               " "
## 6  ( 1 ) " "                               " "
##          MMRAcquisitonRetailCleanPrice MMRCurrentAuctionAveragePrice
## 1  ( 1 ) " "                           " "
## 2  ( 1 ) " "                           " "
## 3  ( 1 ) " "                           " "
## 4  ( 1 ) " "                           " "
## 5  ( 1 ) " "                           " "
## 6  ( 1 ) " "                           " "
##          MMRCurrentAuctionCleanPrice MMRCurrentRetailAveragePrice
## 1  ( 1 ) " "                         " "
## 2  ( 1 ) " "                         " "
## 3  ( 1 ) " "                         " "
## 4  ( 1 ) " "                         " "
## 5  ( 1 ) " "                         " "
## 6  ( 1 ) " "                         " "
##          MMRCurrentRetailCleanPrice VehBCost WarrantyCost WheelTypeID BYRNO
## 1  ( 1 ) " "                        " "      " "          " "         " "
## 2  ( 1 ) " "                        "*"      " "          " "         " "
## 3  ( 1 ) " "                        "*"      " "          " "         " "
## 4  ( 1 ) " "                        "*"      " "          " "         " "
## 5  ( 1 ) " "                        "*"      " "          " "         " "
## 6  ( 1 ) " "                        "*"      " "          " "         " "
##          VNZIP1 IsOnlineSale1 Auction_MANHEIM1 WheelType_Covers1
## 1  ( 1 ) " "    " "           " "              " "
## 2  ( 1 ) " "    " "           " "              " "
## 3  ( 1 ) " "    " "           " "              " "
## 4  ( 1 ) " "    " "           " "              " "
## 5  ( 1 ) " "    " "           "*"              " "
## 6  ( 1 ) " "    " "           "*"              " "
##          WheelType_Alloy1 Nationality_AMERICAN1 Nationality_topline_Asian1
## 1  ( 1 ) " "              " "                   " "
## 2  ( 1 ) " "              " "                   " "
## 3  ( 1 ) " "              " "                   " "
## 4  ( 1 ) " "              " "                   " "
## 5  ( 1 ) " "              " "                   " "
## 6  ( 1 ) " "              " "                   " "
##          Size_MEDIUM1 Size_Large1 Size_small1 TopThreeAmerican_GM1
## 1  ( 1 ) " "          " "         " "         " "
## 2  ( 1 ) " "          " "         " "         " "
## 3  ( 1 ) " "          " "         " "         " "
## 4  ( 1 ) " "          " "         " "         "*"
## 5  ( 1 ) " "          " "         " "         "*"
## 6  ( 1 ) " "          " "         " "         "*"
##          TopThreeAmerican_CHRYSLER1 TopThreeAmerican_FORD1 VNST_West1
```

```
## 1  ( 1 ) " "                        " "                     " "
## 2  ( 1 ) " "                        " "                     " "
## 3  ( 1 ) " "                        " "                     " "
## 4  ( 1 ) " "                        " "                     " "
## 5  ( 1 ) " "                        " "                     " "
## 6  ( 1 ) " "                        " "                     " "
##          VNST_SouthWest1 VNST_Northeast1 VNST_Midwest1
## 1  ( 1 ) " "             " "             " "
## 2  ( 1 ) " "             " "             " "
## 3  ( 1 ) " "             " "             " "
## 4  ( 1 ) " "             " "             " "
## 5  ( 1 ) " "             " "             " "
## 6  ( 1 ) " "             "*"             " "
```

```
step.model$bestTune
```

```
##   nvmax
## 5     5
```

```
coef(step.model$finalModel, 4)
```

```
##                (Intercept)                 VehicleAge
##                -0.01557973                 0.14104975
##                     VehOdo                   VehBCost
##                 0.04514033                -0.05582599
## TopThreeAmerican_CHRYSLER1
##                 0.04743723
```

#Features Forward selection.

```
t(t(names(Car_data2)))
```

```
##       [,1]
##  [1,] "PurchDate"
##  [2,] "VehYear"
##  [3,] "VehicleAge"
##  [4,] "VehOdo"
##  [5,] "MMRAcquisitionAuctionAveragePrice"
##  [6,] "MMRAcquisitionAuctionCleanPrice"
##  [7,] "MMRAcquisitionRetailAveragePrice"
##  [8,] "MMRAcquisitonRetailCleanPrice"
##  [9,] "MMRCurrentAuctionAveragePrice"
## [10,] "MMRCurrentAuctionCleanPrice"
## [11,] "MMRCurrentRetailAveragePrice"
## [12,] "MMRCurrentRetailCleanPrice"
## [13,] "VehBCost"
## [14,] "WarrantyCost"
## [15,] "Auction"
## [16,] "Make"
## [17,] "Model"
## [18,] "Trim"
## [19,] "SubModel"
```

```
## [20,] "Color"
## [21,] "Transmission"
## [22,] "WheelTypeID"
## [23,] "WheelType"
## [24,] "Nationality"
## [25,] "Size"
## [26,] "TopThreeAmericanName"
## [27,] "BYRNO"
## [28,] "VNZIP1"
## [29,] "VNST"
## [30,] "IsOnlineSale"
## [31,] "Class"
## [32,] "Auction_MANHEIM"
## [33,] "WheelType_Covers"
## [34,] "WheelType_Alloy"
## [35,] "Nationality_AMERICAN"
## [36,] "Nationality_topline_Asian"
## [37,] "Size_MEDIUM"
## [38,] "Size_Large"
## [39,] "Size_small"
## [40,] "TopThreeAmerican_GM"
## [41,] "TopThreeAmerican_CHRYSLER"
## [42,] "TopThreeAmerican_FORD"
## [43,] "VNST_West"
## [44,] "VNST_SouthWest"
## [45,] "VNST_Northeast"
## [46,] "VNST_Midwest"
```

```
Car_data4 <- Car_data2[,c(1:14,22,27,28,30:46)]
Car_data4$IsOnlineSale <- as.factor(Car_data4$IsOnlineSale)

set.seed(2023)
Norm_model <- preProcess(Car_data4, method = c("center", "scale"))
Car_data4_norm <-predict(Norm_model,Car_data4)
head(Car_data4_norm)
```

```
##    PurchDate    VehYear  VehicleAge      VehOdo
## 1  1.4711469  0.3746768 -0.09873405 -1.37486769
## 2 -1.1244899 -0.2040693 -0.09873405  1.20887915
## 3 -0.8207957  0.3746768 -0.68438531 -0.03222013
## 4  1.2386310  1.5321689 -1.27003656  0.80650887
## 5 -1.4186937  0.9534228 -1.27003656 -0.90611917
## 6  1.5945227  0.3746768 -0.09873405  0.37265236
##   MMRAcquisitionAuctionAveragePrice MMRAcquisitionAuctionCleanPrice
## 1                        0.01424666                      -0.1289098
## 2                       -1.00863049                      -0.9683562
## 3                        0.29962490                       0.3819776
## 4                        0.69956211                       0.5573073
## 5                        1.07544602                       0.9867916
## 6                       -0.37346007                      -0.4681508
##   MMRAcquisitionRetailAveragePrice MMRAcquisitonRetailCleanPrice
## 1                       0.33813114                    0.15664076
## 2                      -1.28579183                   -1.24996469
## 3                      -0.18690136                   -0.08005902
```

```
## 4                      1.00964009                   0.92495466
## 5                      0.46463354                   0.44387195
## 6                      0.07022507                  -0.02332200
##   MMRCurrentAuctionAveragePrice MMRCurrentAuctionCleanPrice
## 1                    -0.3402254                 -0.4027653
## 2                    -1.0213281                 -0.9882135
## 3                     0.2889603                  0.4179076
## 4                     0.6733939                  0.4757803
## 5                     0.4896239                  0.4156674
## 6                    -0.7856410                 -0.9049514
##   MMRCurrentRetailAveragePrice MMRCurrentRetailCleanPrice   VehBCost
## 1                  -0.09879576                -0.1112094 -0.1457925
## 2                  -1.40444318                -1.3684959 -1.7630331
## 3                  -0.29188902                -0.1381340  0.2384455
## 4                   1.02930954                 0.8202595  0.5251903
## 5                  -0.12147450                -0.1402517  0.2355781
## 6                  -0.61943482                -0.5559186 -0.7192821
##   WarrantyCost WheelTypeID      BYRNO     VNZIP1 IsOnlineSale      Class
## 1   -0.3331889   0.9719537  2.8262824 -1.0011894            0 -0.3248591
## 2   -0.5110579   0.9719537 -0.2162877  0.7842112            0 -0.3248591
## 3    1.2002838  -0.9465773 -0.1359796  0.8323400            0 -0.3248591
## 4    1.5076691   0.9719537 -0.1074110  1.3887892            0 -0.3248591
## 5    0.3817409  -0.9465773 -0.2162877  0.7203457            0 -0.3248591
## 6   -1.2259878  -0.9465773  0.9898373 -1.1766281            0 -0.3248591
##   Auction_MANHEIM WheelType_Covers WheelType_Alloy Nationality_AMERICAN
## 1               1                1               0                    1
## 2               0                1               0                    1
## 3               1                0               1                    1
## 4               1                1               0                    1
## 5               0                0               1                    1
## 6               1                0               1                    0
##   Nationality_topline_Asian Size_MEDIUM Size_Large Size_small
## 1                         0           1          0          0
## 2                         0           1          0          0
## 3                         0           0          1          0
## 4                         0           0          1          0
## 5                         0           0          1          0
## 6                         0           1          0          0
##   TopThreeAmerican_GM TopThreeAmerican_CHRYSLER TopThreeAmerican_FORD VNST_West
## 1                   0                         1                     0         0
## 2                   0                         0                     1         0
## 3                   1                         0                     0         1
## 4                   1                         0                     0         1
## 5                   1                         0                     0         0
## 6                   0                         0                     0         0
##   VNST_SouthWest VNST_Northeast VNST_Midwest
## 1              1              0            0
## 2              1              0            0
## 3              0              0            0
## 4              0              0            0
## 5              1              0            0
## 6              1              0            0
```

```r
set.seed(2023)
# Set up repeated k-fold cross-validation
train.control <- trainControl(method = "cv", number = 15)
# Train the model
step.model <- train(Class ~ ., data = Car_data4_norm,
                    method = "leapForward",
                    tuneGrid = data.frame(nvmax = 1:5),
                    trControl = train.control
                    )
```

```
## Warning in train.default(x, y, weights = w, ...): You are trying to do
## regression and your outcome only has two possible values Are you trying to do
## classification? If so, use a 2 level factor as your outcome column.

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:
```

```
## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:

## Warning in leaps.setup(x, y, wt = weights, nbest = nbest, nvmax = nvmax, : 3
## linear dependencies found

## Reordering variables and trying again:
```

```
step.model$results
```

```
##   nvmax      RMSE   Rsquared       MAE     RMSESD  RsquaredSD      MAESD
## 1     1 0.9853413 0.02882063 0.5708446 0.02163188 0.004336307 0.01333716
## 2     2 0.9840557 0.03138581 0.5699376 0.02136676 0.004958819 0.01333491
## 3     3 0.9832475 0.03297571 0.5707664 0.02130794 0.004932217 0.01307108
## 4     4 0.9830281 0.03341693 0.5700575 0.02145401 0.005135197 0.01314229
## 5     5 0.9821710 0.03512047 0.5705407 0.02147113 0.005376369 0.01337705
```

```
summary(step.model$finalModel)
```

```
## Subset selection object
## 33 Variables  (and intercept)
##                                   Forced in Forced out
## PurchDate                             FALSE      FALSE
## VehYear                               FALSE      FALSE
## VehicleAge                            FALSE      FALSE
## VehOdo                                FALSE      FALSE
## MMRAcquisitionAuctionAveragePrice     FALSE      FALSE
## MMRAcquisitionAuctionCleanPrice       FALSE      FALSE
## MMRAcquisitionRetailAveragePrice      FALSE      FALSE
## MMRAcquisitonRetailCleanPrice         FALSE      FALSE
## MMRCurrentAuctionAveragePrice         FALSE      FALSE
## MMRCurrentAuctionCleanPrice           FALSE      FALSE
## MMRCurrentRetailAveragePrice          FALSE      FALSE
## MMRCurrentRetailCleanPrice            FALSE      FALSE
## VehBCost                              FALSE      FALSE
## WarrantyCost                          FALSE      FALSE
## WheelTypeID                           FALSE      FALSE
## BYRNO                                 FALSE      FALSE
## VNZIP1                                FALSE      FALSE
## IsOnlineSale1                         FALSE      FALSE
## Auction_MANHEIM1                      FALSE      FALSE
## WheelType_Covers1                     FALSE      FALSE
## Nationality_AMERICAN1                 FALSE      FALSE
## Nationality_topline_Asian1            FALSE      FALSE
## Size_MEDIUM1                          FALSE      FALSE
## Size_Large1                           FALSE      FALSE
## Size_small1                           FALSE      FALSE
## TopThreeAmerican_GM1                  FALSE      FALSE
## TopThreeAmerican_CHRYSLER1            FALSE      FALSE
## VNST_West1                            FALSE      FALSE
## VNST_SouthWest1                       FALSE      FALSE
## VNST_Northeast1                       FALSE      FALSE
## WheelType_Alloy1                      FALSE      FALSE
## TopThreeAmerican_FORD1                FALSE      FALSE
## VNST_Midwest1                         FALSE      FALSE
## 1 subsets of each size up to 6
## Selection Algorithm: forward
##          PurchDate VehYear VehicleAge VehOdo MMRAcquisitionAuctionAveragePrice
## 1  ( 1 ) " "       " "     "*"        " "    " "
## 2  ( 1 ) " "       " "     "*"        " "    " "
## 3  ( 1 ) " "       " "     "*"        "*"    " "
## 4  ( 1 ) " "       " "     "*"        "*"    " "
## 5  ( 1 ) " "       " "     "*"        "*"    " "
## 6  ( 1 ) " "       " "     "*"        "*"    " "
##          MMRAcquisitionAuctionCleanPrice MMRAcquisitionRetailAveragePrice
## 1  ( 1 ) " "                             " "
## 2  ( 1 ) " "                             " "
## 3  ( 1 ) " "                             " "
## 4  ( 1 ) " "                             " "
## 5  ( 1 ) " "                             " "
```

```
## 6  ( 1 ) " "                              " "
##          MMRAcquisitonRetailCleanPrice MMRCurrentAuctionAveragePrice
## 1  ( 1 ) " "                           " "
## 2  ( 1 ) " "                           " "
## 3  ( 1 ) " "                           " "
## 4  ( 1 ) " "                           " "
## 5  ( 1 ) " "                           " "
## 6  ( 1 ) " "                           "*"
##          MMRCurrentAuctionCleanPrice MMRCurrentRetailAveragePrice
## 1  ( 1 ) " "                         " "
## 2  ( 1 ) " "                         " "
## 3  ( 1 ) " "                         " "
## 4  ( 1 ) " "                         " "
## 5  ( 1 ) " "                         " "
## 6  ( 1 ) " "                         " "
##          MMRCurrentRetailCleanPrice VehBCost WarrantyCost WheelTypeID BYRNO
## 1  ( 1 ) " "                        " "      " "          " "         " "
## 2  ( 1 ) " "                        "*"      " "          " "         " "
## 3  ( 1 ) " "                        "*"      " "          " "         " "
## 4  ( 1 ) " "                        "*"      " "          " "         " "
## 5  ( 1 ) " "                        "*"      " "          " "         " "
## 6  ( 1 ) " "                        "*"      " "          " "         " "
##          VNZIP1 IsOnlineSale1 Auction_MANHEIM1 WheelType_Covers1
## 1  ( 1 ) " "    " "           " "              " "
## 2  ( 1 ) " "    " "           " "              " "
## 3  ( 1 ) " "    " "           " "              " "
## 4  ( 1 ) " "    " "           " "              " "
## 5  ( 1 ) " "    " "           "*"              " "
## 6  ( 1 ) " "    " "           "*"              " "
##          WheelType_Alloy1 Nationality_AMERICAN1 Nationality_topline_Asian1
## 1  ( 1 ) " "              " "                   " "
## 2  ( 1 ) " "              " "                   " "
## 3  ( 1 ) " "              " "                   " "
## 4  ( 1 ) " "              " "                   " "
## 5  ( 1 ) " "              " "                   " "
## 6  ( 1 ) " "              " "                   " "
##          Size_MEDIUM1 Size_Large1 Size_small1 TopThreeAmerican_GM1
## 1  ( 1 ) " "          " "         " "         " "
## 2  ( 1 ) " "          " "         " "         " "
## 3  ( 1 ) " "          " "         " "         " "
## 4  ( 1 ) " "          " "         " "         "*"
## 5  ( 1 ) " "          " "         " "         "*"
## 6  ( 1 ) " "          " "         " "         "*"
##          TopThreeAmerican_CHRYSLER1 TopThreeAmerican_FORD1 VNST_West1
## 1  ( 1 ) " "                        " "                    " "
## 2  ( 1 ) " "                        " "                    " "
## 3  ( 1 ) " "                        " "                    " "
## 4  ( 1 ) " "                        " "                    " "
## 5  ( 1 ) " "                        " "                    " "
## 6  ( 1 ) " "                        " "                    " "
##          VNST_SouthWest1 VNST_Northeast1 VNST_Midwest1
## 1  ( 1 ) " "             " "             " "
## 2  ( 1 ) " "             " "             " "
## 3  ( 1 ) " "             " "             " "
```

```
## 4  ( 1 ) " "             " "             " "
## 5  ( 1 ) " "             " "             " "
## 6  ( 1 ) " "             " "             " "
```

```
step.model$bestTune
```

```
##   nvmax
## 5     5
```

```
coef(step.model$finalModel, 4)
```

```
##                (Intercept)                 VehicleAge
##                -0.01557973                 0.14104975
##                      VehOdo                   VehBCost
##                 0.04514033                -0.05582599
## TopThreeAmerican_CHRYSLER1
##                 0.04743723
```

#After using Backward and forward stepwise method we found 4 important variables. #lm() and ANOVA using selected independet variables

```
t(t(names(Car_data2)))
```

```
##       [,1]
##  [1,] "PurchDate"
##  [2,] "VehYear"
##  [3,] "VehicleAge"
##  [4,] "VehOdo"
##  [5,] "MMRAcquisitionAuctionAveragePrice"
##  [6,] "MMRAcquisitionAuctionCleanPrice"
##  [7,] "MMRAcquisitionRetailAveragePrice"
##  [8,] "MMRAcquisitonRetailCleanPrice"
##  [9,] "MMRCurrentAuctionAveragePrice"
## [10,] "MMRCurrentAuctionCleanPrice"
## [11,] "MMRCurrentRetailAveragePrice"
## [12,] "MMRCurrentRetailCleanPrice"
## [13,] "VehBCost"
## [14,] "WarrantyCost"
## [15,] "Auction"
## [16,] "Make"
## [17,] "Model"
## [18,] "Trim"
## [19,] "SubModel"
## [20,] "Color"
## [21,] "Transmission"
## [22,] "WheelTypeID"
## [23,] "WheelType"
## [24,] "Nationality"
## [25,] "Size"
## [26,] "TopThreeAmericanName"
## [27,] "BYRNO"
## [28,] "VNZIP1"
```

```
## [29,] "VNST"
## [30,] "IsOnlineSale"
## [31,] "Class"
## [32,] "Auction_MANHEIM"
## [33,] "WheelType_Covers"
## [34,] "WheelType_Alloy"
## [35,] "Nationality_AMERICAN"
## [36,] "Nationality_topline_Asian"
## [37,] "Size_MEDIUM"
## [38,] "Size_Large"
## [39,] "Size_small"
## [40,] "TopThreeAmerican_GM"
## [41,] "TopThreeAmerican_CHRYSLER"
## [42,] "TopThreeAmerican_FORD"
## [43,] "VNST_West"
## [44,] "VNST_SouthWest"
## [45,] "VNST_Northeast"
## [46,] "VNST_Midwest"
```

```
Car_data5 <- Car_data2[,c(3,4,13,31,41)]

set.seed(2023)
Norm_model <- preProcess(Car_data5, method = c("center", "scale"))
Car_data5_norm <-predict(Norm_model,Car_data5)
head(Car_data5_norm)
```

```
##    VehicleAge      VehOdo   VehBCost      Class TopThreeAmerican_CHRYSLER
## 1 -0.09873405 -1.37486769 -0.1457925 -0.3248591                         1
## 2 -0.09873405  1.20887915 -1.7630331 -0.3248591                         0
## 3 -0.68438531 -0.03222013  0.2384455 -0.3248591                         0
## 4 -1.27003656  0.80650887  0.5251903 -0.3248591                         0
## 5 -1.27003656 -0.90611917  0.2355781 -0.3248591                         0
## 6 -0.09873405  0.37265236 -0.7192821 -0.3248591                         0
```

```
str(Car_data5_norm)
```

```
## 'data.frame':    67211 obs. of  5 variables:
##  $ VehicleAge              : num  -0.0987 -0.0987 -0.6844 -1.27 -1.27 ...
##  $ VehOdo                  : num  -1.3749 1.2089 -0.0322 0.8065 -0.9061 ...
##  $ VehBCost                : num  -0.146 -1.763 0.238 0.525 0.236 ...
##  $ Class                   : num  -0.325 -0.325 -0.325 -0.325 -0.325 ...
##  $ TopThreeAmerican_CHRYSLER: Factor w/ 2 levels "0","1": 2 1 1 1 1 1 1 1 1 1 ...
```

```
# Fit a linear regression model
lm_model <- lm(Class ~ ., data = Car_data5_norm)
summary(lm_model)
```

```
##
## Call:
## lm(formula = Class ~ ., data = Car_data5_norm)
##
## Residuals:
```

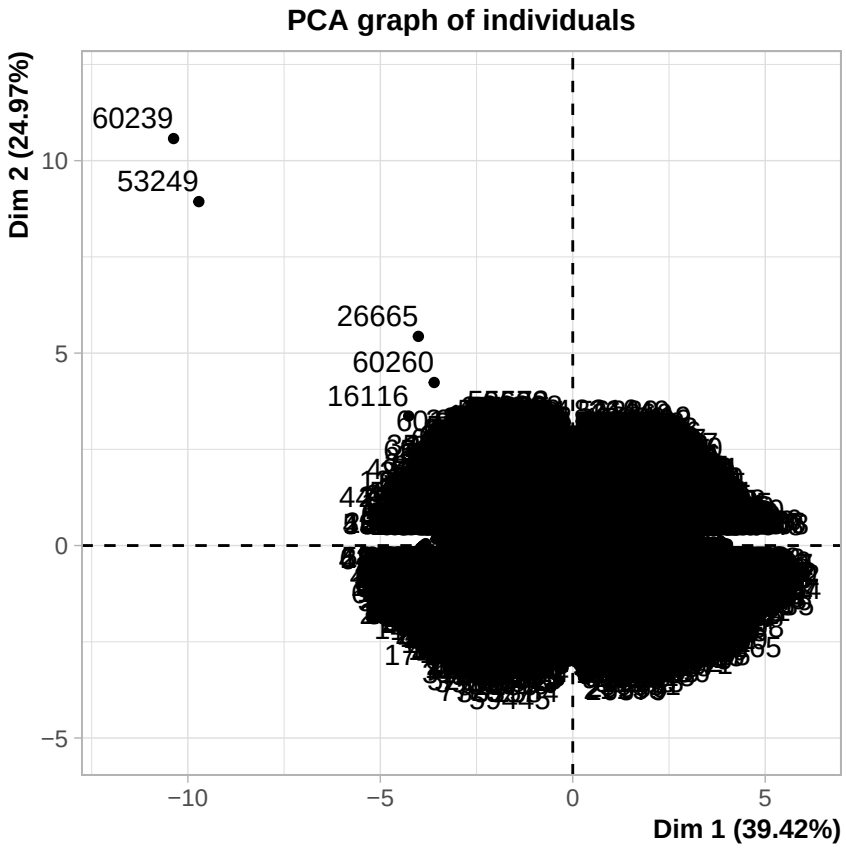
```
##     Min      1Q  Median      3Q     Max
## -0.9825 -0.4113 -0.2772 -0.1438  4.4588
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -0.015580   0.004663  -3.341 0.000834 ***
## VehicleAge                  0.141050   0.004239  33.274  < 2e-16 ***
## VehOdo                      0.045140   0.004022  11.224  < 2e-16 ***
## VehBCost                   -0.055826   0.004017 -13.897  < 2e-16 ***
## TopThreeAmerican_CHRYSLER1  0.047437   0.008259   5.744 9.31e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9832 on 67206 degrees of freedom
## Multiple R-squared:  0.03338,    Adjusted R-squared:  0.03332
## F-statistic: 580.2 on 4 and 67206 DF,  p-value: < 2.2e-16
```

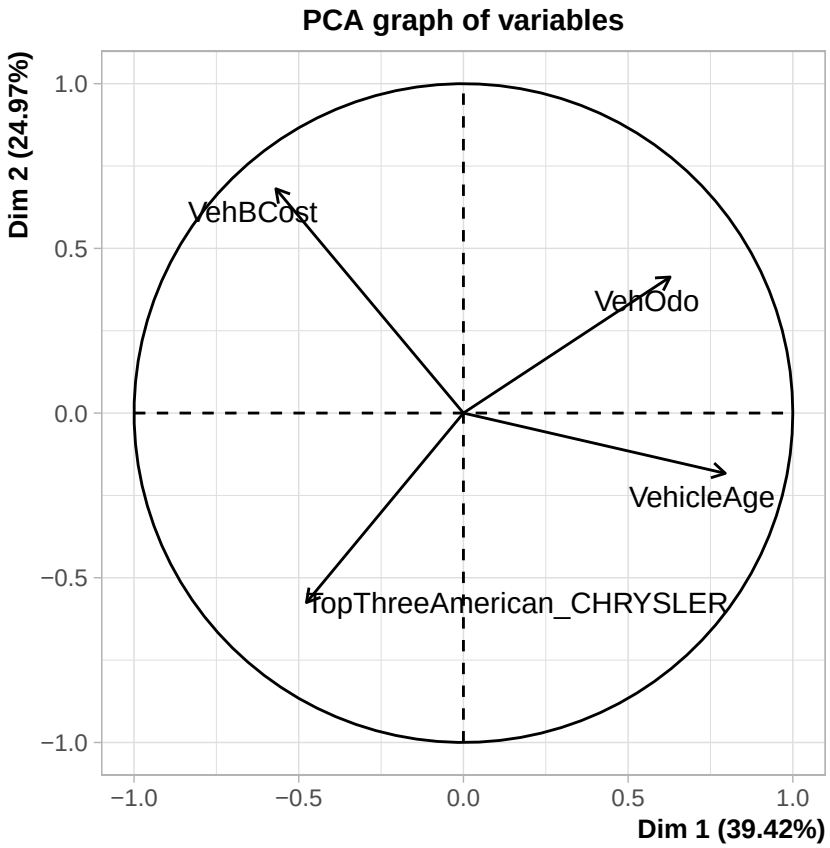
```
#Perform ANOVA
anova_results <- aov(Class ~ ., data = Car_data5_norm)
summary(anova_results)
```

```
##                               Df Sum Sq Mean Sq F value   Pr(>F)
## VehicleAge                     1   1930  1929.8 1996.35  < 2e-16 ***
## VehOdo                         1     97    96.8  100.18  < 2e-16 ***
## VehBCost                       1    185   185.0  191.35  < 2e-16 ***
## TopThreeAmerican_CHRYSLER      1     32    31.9   32.99 9.31e-09 ***
## Residuals                  67206  64966     1.0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

#PCA : Determining the relative importance of the primary variables in the data set using principal component analysis.

```
library(FactoMineR)

Car_data5$TopThreeAmerican_CHRYSLER <- as.numeric(Car_data5$TopThreeAmerican_CHRYSLER)

pca <- PCA(Car_data5[,-4])
```

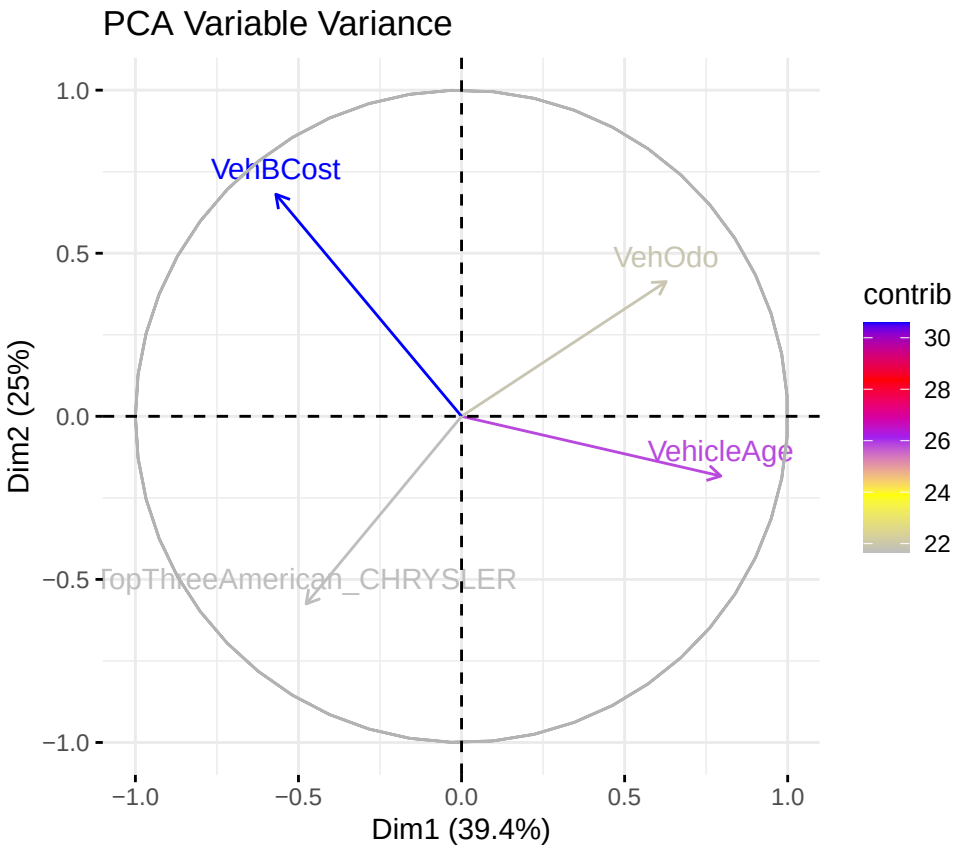

PCA graph of individuals
Dim 2 (24.97%)
Dim 1 (39.42%)
60239
53249
26665
60260
16116
10
5
0
−5
−10
−5
0
5




```
pca <- prcomp(Car_data5[,-4], scale = TRUE)
# extract loadings
loadings <- pca$rotation
# print loadings for the first two PCs
print(loadings[, 1:2])
```

```
##                                  PC1        PC2
## VehicleAge                 0.6330862 -0.1831728
## VehOdo                     0.4995866  0.4132633
## VehBCost                  -0.4534281  0.6813806
## TopThreeAmerican_CHRYSLER -0.3794972 -0.5756576
```

```
var <- get_pca_var(pca)
fviz_pca_var(pca, col.var="contrib",
gradient.cols = c("grey","yellow","purple","red","blue"),ggrepel = TRUE ) + labs( title = "PCA Variable
```


PCA Variable Variance
VehBCost
VehOdo
VehicleAge
TopThreeAmerican_CHRYSLER
contrib
30
28
26
24
22
1.0
0.5
0.0
−0.5
−1.0
Dim2 (25%)
Dim1 (39.4%)