USED CAR PRICES CASE STUDY

Deliverable I: Data Processing, Description, Validation and Profiling

Miquel Parra i Xavier Alaman

March 26, 2022

Contents

1 R libraries imports, useful functions and data loading

In this first section we will load all required packages and libraries, declare additional functions, and load our data.

1.1 Load required packages

```
options(contrasts=c("contr.treatment","contr.treatment"))

requiredPackages <- c("effects","FactoMineR","car",
                      "factoextra","RColorBrewer","ggplot2","dplyr","ggmap",
                      "ggthemes","knitr","treemap")

#use this function to check if each package is on the local machine
#if a package is installed, it will be loaded
#if any are not, the missing package(s) will be installed and loaded
package.check <- lapply(requiredPackages, FUN = function(x) {
  if (!require(x, character.only = TRUE)) {
    install.packages(x, dependencies = TRUE)
    library(x, character.only = TRUE)
  }
})

#verify they are loaded
search()
```

1.2 Sample load

```
# Clear plots
if(!is.null(dev.list())) dev.off()

# Clean workspace
rm(list=ls())

# Users file path
miquel_fp <- "C:/Users/Miquel/Documents/GitHub/ADEI/"
xavi_fp <- "~/Documents/FIB/ADEI/ADEI/"
filepath <- xavi_fp

# Set working directory
setwd(filepath)

# Load data from file
load(paste0(filepath,"MyOldCars-Raw.RData"))

# Index reset
row.names(df) <- NULL
```

1.3 Useful functions

```
#Mout <- which((df$tax < var_out$mouti)|(df$tax > var_out$mouts))

# Some useful functions
calcQ <- function(x) {
  s.x <- summary(x)
  iqr<-s.x[5]-s.x[2]
  list(souti=s.x[2]-3*iqr, mouti=s.x[2]-1.5*iqr, min=s.x[1], q1=s.x[2], q2=s.x[3],
       q3=s.x[5], max=s.x[6], mouts=s.x[5]+1.5*iqr, souts=s.x[5]+3*iqr ) }

countNA <- function(x) {
  mis_x <- NULL
  for (j in 1:ncol(x)) {mis_x[j] <- sum(is.na(x[,j])) }
  mis_x <- as.data.frame(mis_x)
  rownames(mis_x) <- names(x)
  mis_i <- rep(0,nrow(x))
  for (j in 1:ncol(x)) {mis_i <- mis_i + as.numeric(is.na(x[,j])) }
  list(mis_col=mis_x,mis_ind=mis_i) }

countX <- function(x,X) {
  n_x <- NULL
  for (j in 1:ncol(x)) {n_x[j] <- sum(x[,j]==X) }
  n_x <- as.data.frame(n_x)
  rownames(n_x) <- names(x)
  nx_i <- rep(0,nrow(x))
  for (j in 1:ncol(x)) {nx_i <- nx_i + as.numeric(x[,j]==X) }
  list(nx_col=n_x,nx_ind=nx_i) }
```

2 Data Description

During this project we will be working with a subset of the pre-treated original dataset “Uk used car dataset”. A sample of 5000 cars has been randomly selected from Mercedes, BMW, Volkwagen and Audi manufacturers and stored into a RData file ***MyOldCars-Raw.RData***.

2.1 Original variables description

- **model:** Car model.
- **year:** Car registration year.
- **price:** Car price in £.
- **transmission:** Type of transmission [“Manual”, “Automatic”, “Semi-Auto”].
- **mileage:** Distance used, accumulated miles.
- **fuelType:** Type of engine fuel [“Petrol”, “Diesel”, “Hybrid”, “Other”].
- **tax:** Applied road tax.
- **mpg:** Miles per gallon.
- **engineSize:** Engine size in liters. The cars with engine size 0 are in fact electric cars, nevertheless Mercedes C class, and other given cars are not electric cars,so data imputation is required.
- **manufacturer:** Car manufacturer [“Audi”, “BMW”, “Mercedes”, “VW”].

```
summary(df)
```

```
##     model                year          price        transmission
##  Length:5000        Min.   :1999   Min.   :   650   Length:5000
##  Class :character   1st Qu.:2016   1st Qu.: 13995   Class :character
##  Mode  :character   Median :2017   Median : 19498   Mode  :character
##                     Mean   :2017   Mean   : 21470
##                     3rd Qu.:2019   3rd Qu.: 26039
##                     Max.   :2020   Max.   :109990
##     mileage         fuelType              tax              mpg
##  Min.   :     4   Length:5000        Min.   :  0.0   Min.   :  8.80
##  1st Qu.:  5999   Class :character   1st Qu.:125.0   1st Qu.: 44.80
##  Median : 16619   Mode  :character   Median :145.0   Median : 53.30
##  Mean   : 23312                      Mean   :125.3   Mean   : 53.89
##  3rd Qu.: 33834                      3rd Qu.:145.0   3rd Qu.: 61.40
##  Max.   :153000                      Max.   :580.0   Max.   :470.80
##    engineSize    manufacturer
##  Min.   :0.000   Length:5000
##  1st Qu.:1.500   Class :character
##  Median :2.000   Mode  :character
##  Mean   :1.917
##  3rd Qu.:2.000
##  Max.   :6.600
```

```
head(df, 3)
```

```
##   model year price transmission mileage fuelType tax  mpg engineSize
## 1    A1 2016 11000       Manual   29946   Petrol  30 55.4        1.4
## 2    A3 2015 10200       Manual   46112   Petrol  20 60.1        1.4
## 3    A4 2017 18500    Automatic   17418   Diesel 145 62.8        2.0
##   manufacturer
## 1         Audi
## 2         Audi
## 3         Audi
```

3 Univariate Descriptive Analysis

In this step of the process original numeric variables corresponding to qualitative concepts have to be converted to factors. New factors grouping original levels will be considered very positively.

Additionally original numeric variables corresponding to real quantitative concepts are kept as numeric but additional factors should also be created as a discretization of each numeric variable.

For each variable we will perform the necessary transformations and also visualize its distribution.

3.1 Numeric variables

3.1.1 years_sell

From **year** we can create a new variable called **years_sell**. It will contain the same information as **year** but it will give more valuable information for a human understanding perspective.

years_sell represents the years the car has been sold.

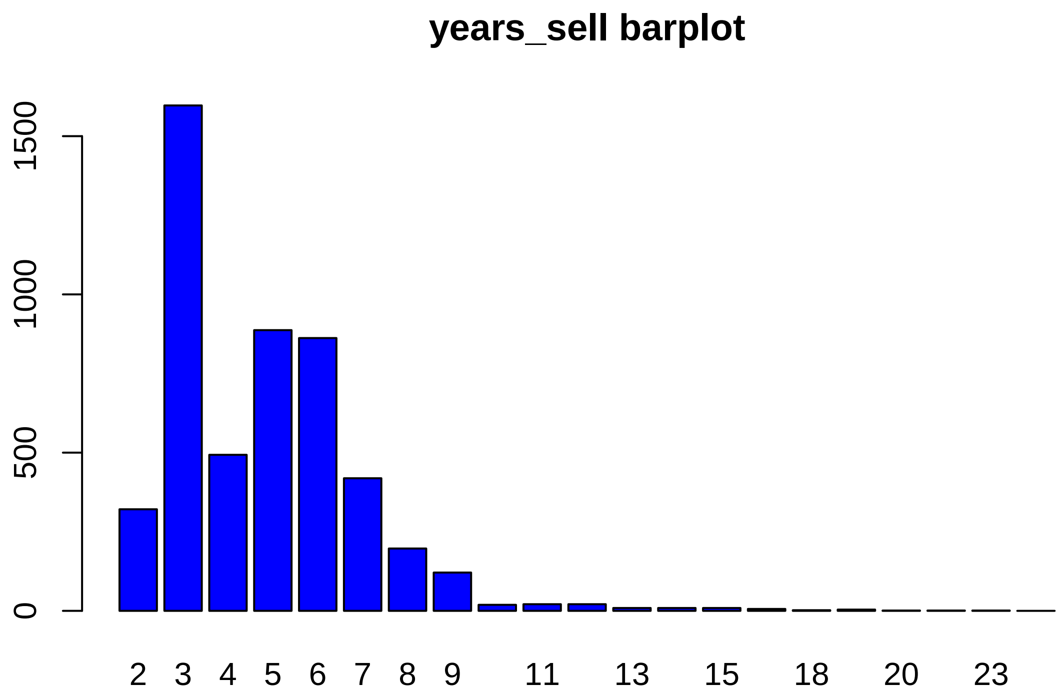
```
df$years_sell <-  2022 - df$year
summary(df$years_sell)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.000   3.000   5.000   4.787   6.000  23.000
```

```
table(df$years_sell,useNA="always")
```

```
##
##    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   18
##  321 1597  493  887  862  419  197  121   19   21   21    9    9    9    6    2
##   19   20   21   23 <NA>
##    4    1    1    1    0
```

```
barplot(table(df$years_sell,useNA="always"), main = "years_sell barplot", col = "blue")
```

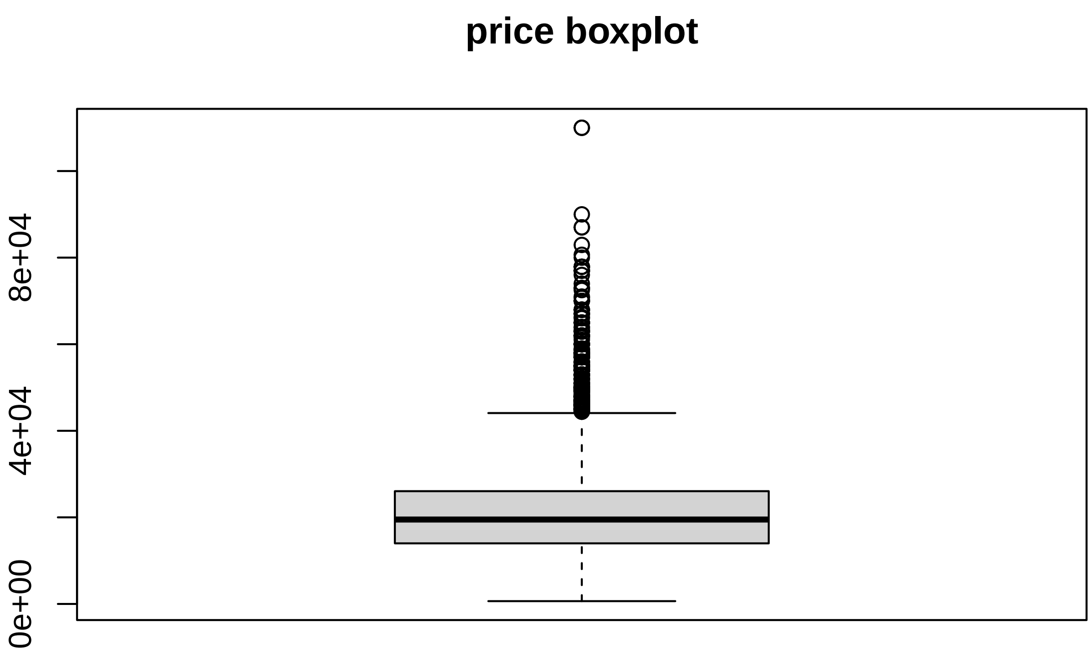


3.1.2 price

```
summary(df$price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     650   13995   19498   21470   26039  109990
```

```
boxplot(df$price, main="price boxplot")
```

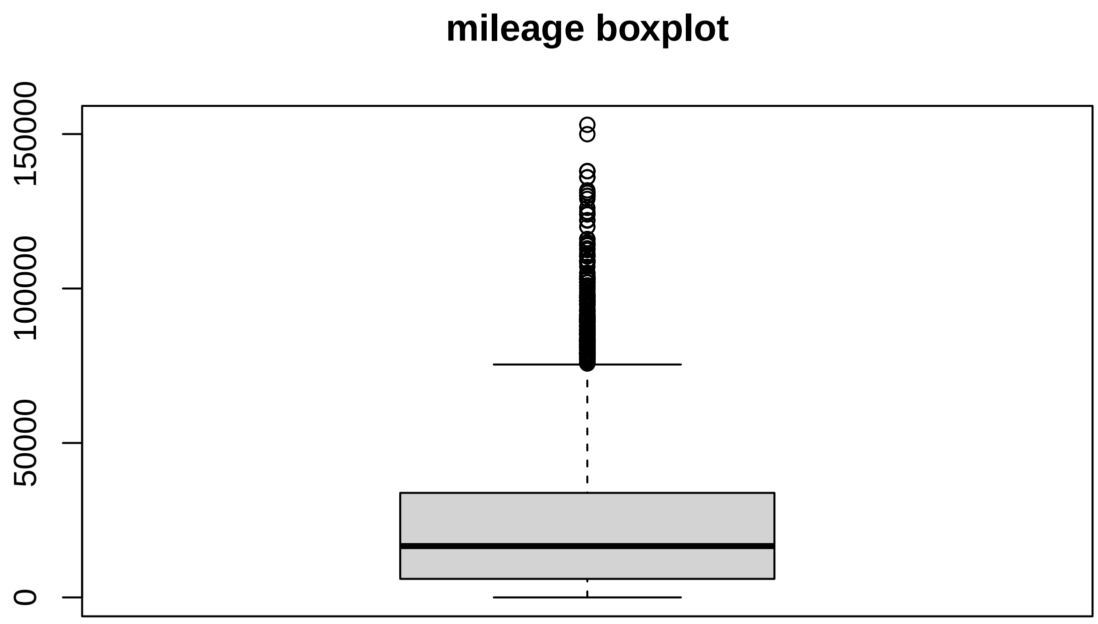


3.1.3 mileage

```
summary(df$mileage)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##       4    5999   16619   23312   33834  153000
```

```
boxplot(df$mileage, main="mileage boxplot")
```

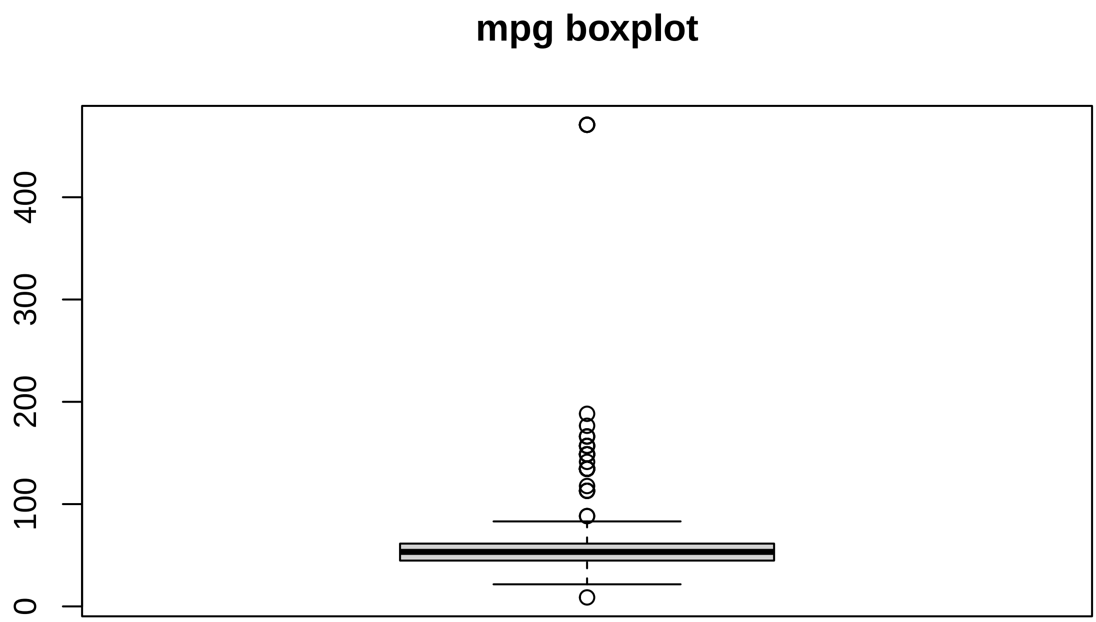


3.1.4 mpg

```
summary(df$mpg)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##     8.80    44.80    53.30    53.89    61.40   470.80
```

```
boxplot(df$mpg, main="mpg boxplot")
```

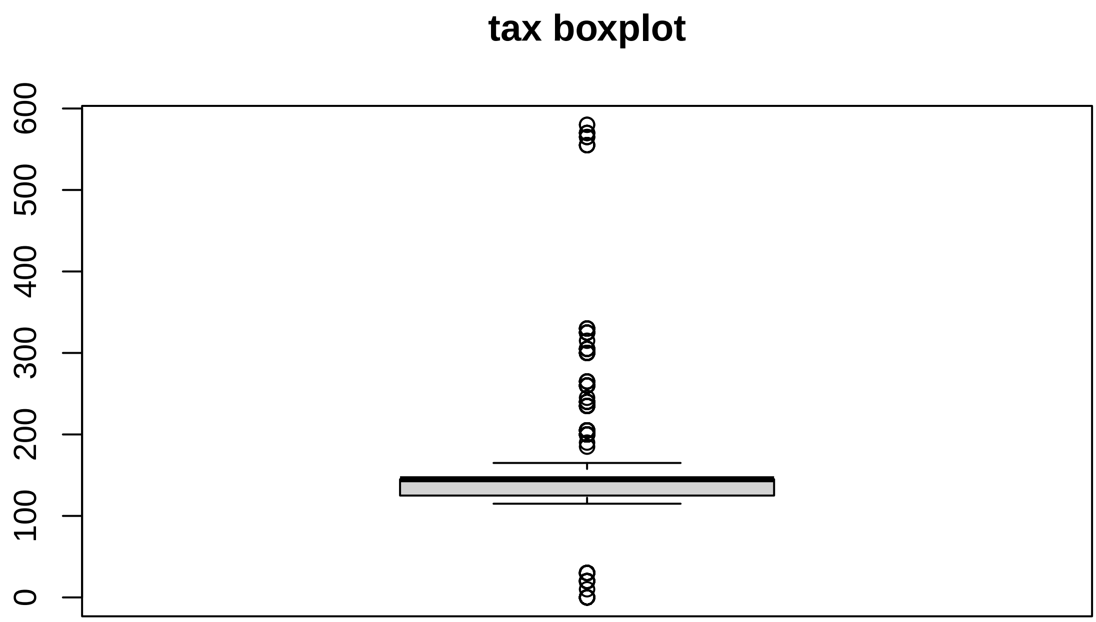


3.1.5 tax

```
summary(df$tax)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   125.0   145.0   125.3   145.0   580.0
```

```
boxplot(df$tax, main="tax boxplot")
```



3.2 Factors

3.2.1 model

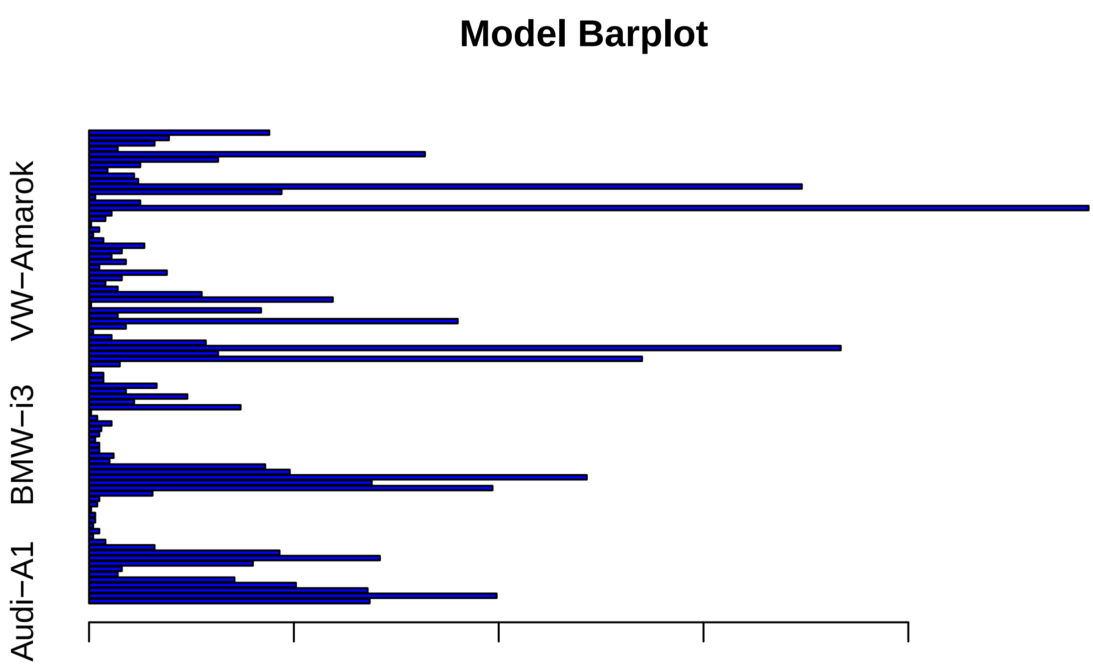
```
df$model<-factor(paste0(trimws(df$manufacturer),"-",trimws(df$model)))

summary(df$model)
```

```
##           Audi-A1           Audi-A3           Audi-A4           Audi-A5 
##               137               199               136               101 
##           Audi-A6           Audi-A7           Audi-A8           Audi-Q2 
##                71                14                16                80 
##           Audi-Q3           Audi-Q5           Audi-Q7           Audi-Q8 
##               142                93                32                 8 
##           Audi-R8          Audi-RS3          Audi-RS4          Audi-RS5 
##                 2                 5                 2                 3 
##          Audi-RS6           Audi-S3           Audi-S4          Audi-SQ5 
##                 3                 1                 4                 5 
##           Audi-TT      BMW-1 Series      BMW-2 Series      BMW-3 Series 
##                31               197               138               243 
##      BMW-4 Series      BMW-5 Series      BMW-6 Series      BMW-7 Series 
##                98                86                10                12 
##      BMW-8 Series            BMW-i3            BMW-i8            BMW-M2 
##                 5                 5                 3                 5 
##            BMW-M3            BMW-M4            BMW-M5            BMW-M6 
##                 6                11                 4                 1 
##            BMW-X1            BMW-X2            BMW-X3            BMW-X4 
##                74                22                48                18 
##            BMW-X5            BMW-X6            BMW-X7            BMW-Z3 
##                33                 7                 7                 1 
##            BMW-Z4  Mercedes-A Class  Mercedes-B Class  Mercedes-C Class 
##                15               270                63               367 
##  Mercedes-CL Class Mercedes-CLA Class Mercedes-CLC Class Mercedes-CLS Class
```

```
##                57                 11                  2                 18
##    Mercedes-E Class   Mercedes-GL Class Mercedes-GLA Class Mercedes-GLB Class
##               180                 14                 84                  1
## Mercedes-GLC Class Mercedes-GLE Class Mercedes-GLS Class   Mercedes-M Class
##               119                 55                 14                  8
##    Mercedes-S Class   Mercedes-SL CLASS       Mercedes-SLK   Mercedes-V Class
##                16                 38                  5                 18
##    Mercedes-X-CLASS          VW-Amarok          VW-Arteon          VW-Beetle
##                11                 16                 27                  7
##       VW-Caddy Life VW-Caddy Maxi Life      VW-California       VW-Caravelle
##                 2                  5                  1                  8
##               VW-CC            VW-Golf         VW-Golf SV           VW-Jetta
##                11                488                 25                  3
##           VW-Passat            VW-Polo        VW-Scirocco          VW-Sharan
##                94                348                 24                 22
##          VW-Shuttle         VW-T-Cross          VW-T-Roc          VW-Tiguan
##                 9                 25                 63                164
## VW-Tiguan Allspace         VW-Touareg          VW-Touran              VW-Up
##                14                 32                 39                 88
```

```
barplot(summary(df$model), main = "Model Barplot", col = "blue", horiz=TRUE)
```



3.2.2 year

As you could imagine the distribution of years and years_sell is the same, but moved from right to left as a result of the subtract operation. We have considered to join cars from year 1999 to 2009 because they are residual values, i.e. the amount of individuals per each one (year) is not representative enough.

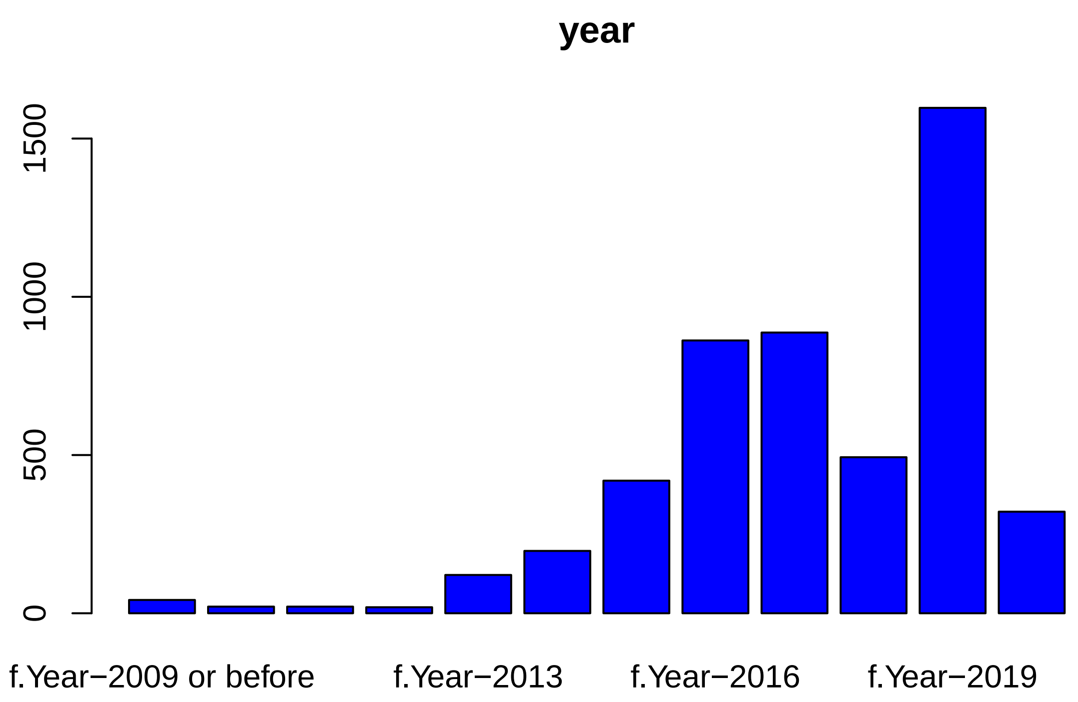
```
df[which(df$year<=2009),"year"] <- "2009 or before"

df$year <- factor(df$year)
df$year <- factor(df$year, labels = paste0("f.Year-",levels(df$year)))

summary(df$year)
```

```
## f.Year-2009 or before           f.Year-2010           f.Year-2011
##                    42                    21                    21
##           f.Year-2012           f.Year-2013           f.Year-2014
##                    19                   121                   197
##           f.Year-2015           f.Year-2016           f.Year-2017
##                   419                   862                   887
##           f.Year-2018           f.Year-2019           f.Year-2020
##                   493                  1597                   321
```

```
barplot(summary(df$year), main = "year", col = "blue")
```



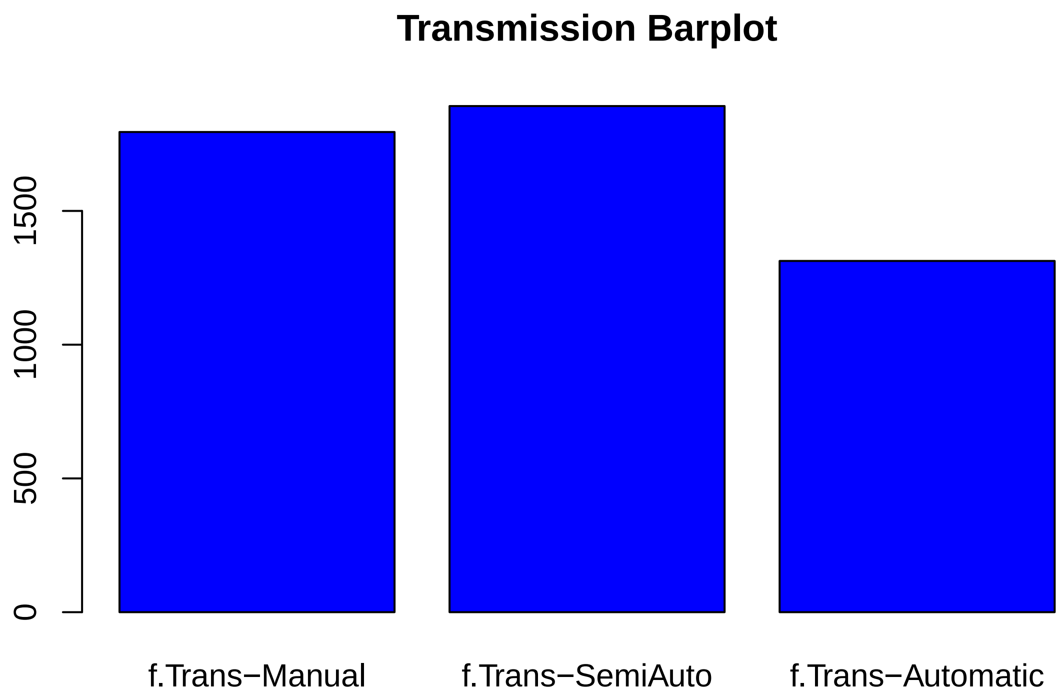
3.2.3 transmission

```
df$transmission<-factor(df$transmission)
df$transmission <- factor(df$transmission, levels = c("Manual","Semi-Auto","Automatic"),labels = paste0(

summary(df$transmission)
```

```
##    f.Trans-Manual  f.Trans-SemiAuto f.Trans-Automatic
##              1795              1892              1313
```

```
barplot(summary(df$transmission), main = "Transmission Barplot", col = "blue")
```



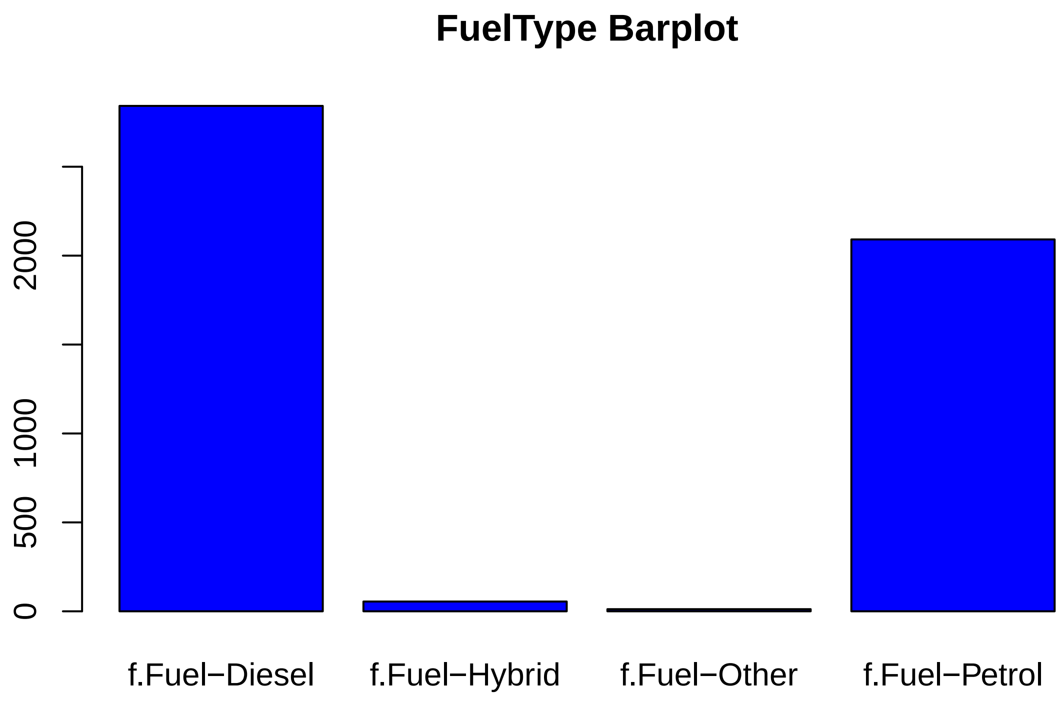
3.2.4 fuelType

```
df$fuelType<-factor(df$fuelType)
df$fuelType <- factor(df$fuelType, labels = paste0("f.Fuel-",levels(df$fuelType)))

summary(df$fuelType)
```

```
## f.Fuel-Diesel f.Fuel-Hybrid  f.Fuel-Other f.Fuel-Petrol
##          2842            55            12          2091
```

```
barplot(summary(df$fuelType), main = "FuelType Barplot", col = "blue")
```



3.2.5 engineSize

We have considered to join cars with engineSize 4.2, 4.4, 4.7, 5, 5.2, 5.5, 6.2 and 6.6 because they are residual values, i.e. the amount of individuals per each one (engineSize) is not representative enough.

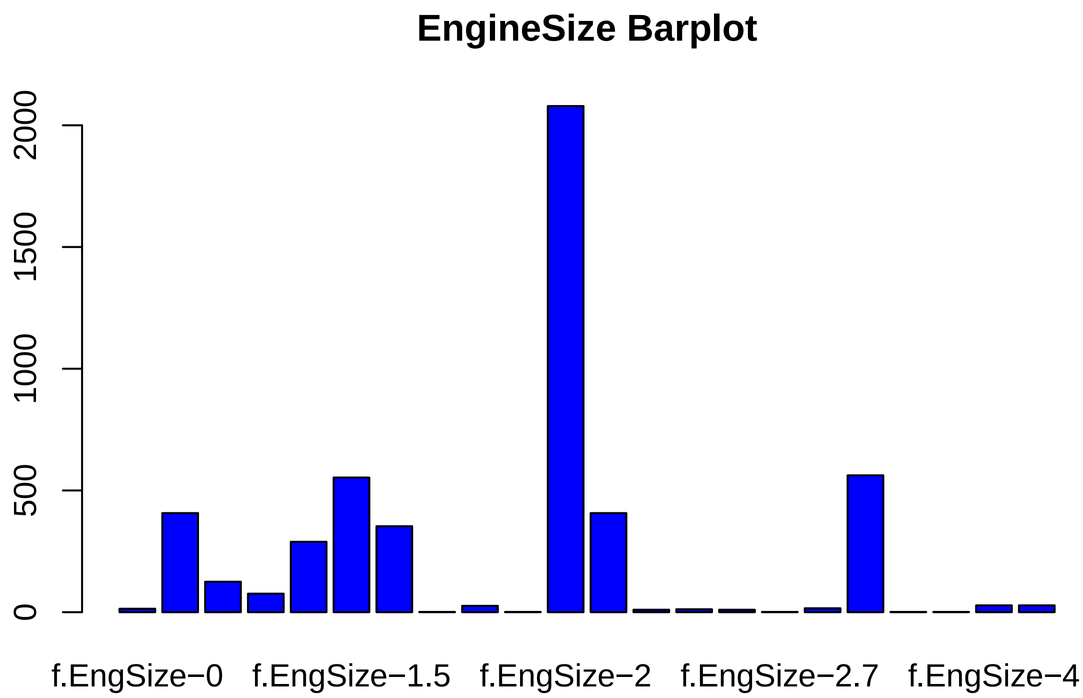
```
df[which(df$engineSize>=4.2),"engineSize"] <- "4.2 or more"

df$engineSize<-factor(df$engineSize)
df$engineSize <- factor(df$engineSize, labels = paste0("f.EngSize-",levels(df$engineSize)))

summary(df$engineSize)
```

```
##          f.EngSize-0          f.EngSize-1        f.EngSize-1.2
##                   14                  407                  125
##        f.EngSize-1.3        f.EngSize-1.4        f.EngSize-1.5
##                   76                  289                  553
##        f.EngSize-1.6        f.EngSize-1.7        f.EngSize-1.8
##                  353                    1                   26
##        f.EngSize-1.9          f.EngSize-2        f.EngSize-2.1
##                    1                 2079                  407
##        f.EngSize-2.2        f.EngSize-2.3        f.EngSize-2.5
##                   10                   12                   10
##        f.EngSize-2.7        f.EngSize-2.9          f.EngSize-3
##                    1                   16                  562
##        f.EngSize-3.2        f.EngSize-3.5          f.EngSize-4
##                    1                    1                   28
## f.EngSize-4.2 or more
##                   28
```

```
barplot(summary(df$engineSize), main = "EngineSize Barplot", col = "blue")
```



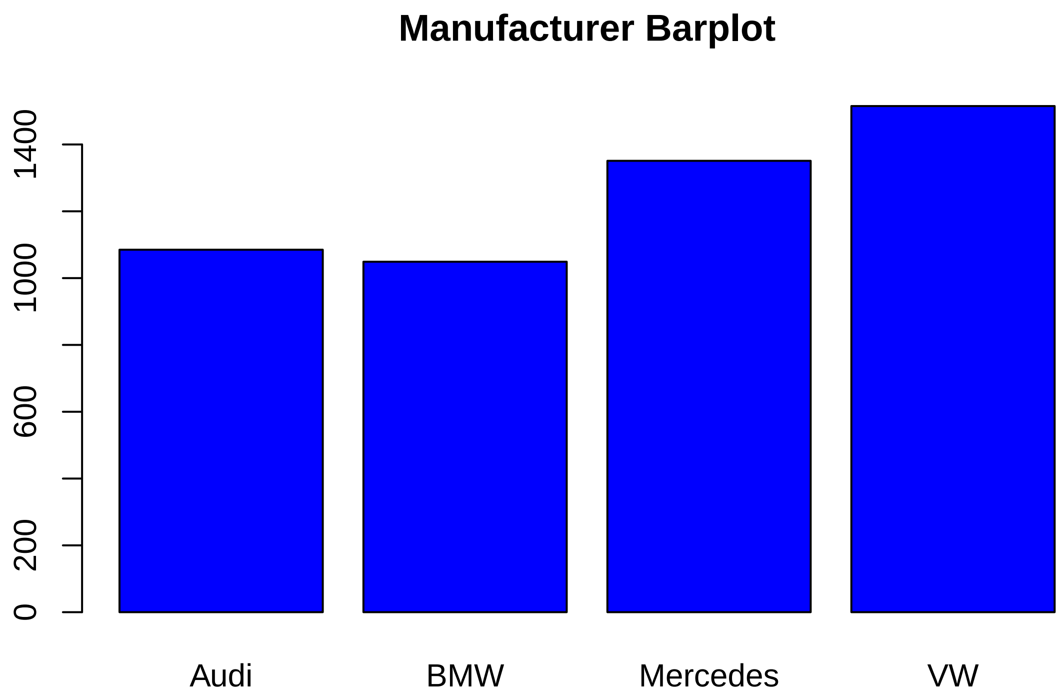
3.2.6 manufacturer

```
df$manufacturer<-factor(df$manufacturer)
df$manufacturer <- factor(df$manufacturer, labels = levels(df$manufacturer))

summary(df$manufacturer)
```

```
##      Audi      BMW Mercedes       VW
##      1085     1049     1351     1515
```

```
barplot(summary(df$manufacturer), main = "Manufacturer Barplot", col = "blue")
```



4 Data Quality Report

4.1 Initialization of counts for missings, outliers and errors.

```
########################################################
imis<-rep(0,nrow(df))  # rows - cars
jmis<-rep(0,ncol(df))  # columns - variables
#######################################################
mis1<-countNA(df) #There are no missings at the beginning

# Number of missings for the current set of cars
sum(mis1$mis_ind)
```

```
## [1] 0
```

```
# Number of missings for the current set of variables
sum(mis1$mis_col)
```

```
## [1] 0
```

```
########################################################
iouts<-rep(0,nrow(df))  # rows - cars
jouts<-rep(0,ncol(df))  # columns - variables
#######################################################

########################################################
ierrs<-rep(0,nrow(df))  # rows - cars
jerrs<-rep(0,ncol(df))  # columns - variables
#######################################################
```

As you can see from the previous stats there are no missings in the variables for the random data subset.

4.2 Errors

After the first analysis of the samples and the provided documentation of the dataset we could say that the only visible errors are in the engineSize variable.

Engine size equal to zero is considered as an electrical vehicle so this error in the data needs to be considered and treated properly.

```
sel<-which(df$engineSize==0 & df$fuelType!="f.Fuel-Electric")
ierrs[sel]<-ierrs[sel]+1
df[sel,"engineSize"]<-NA
selmiss <- sel
jerrs[9] <- length(sel)
```

4.3 Univariate Outliers

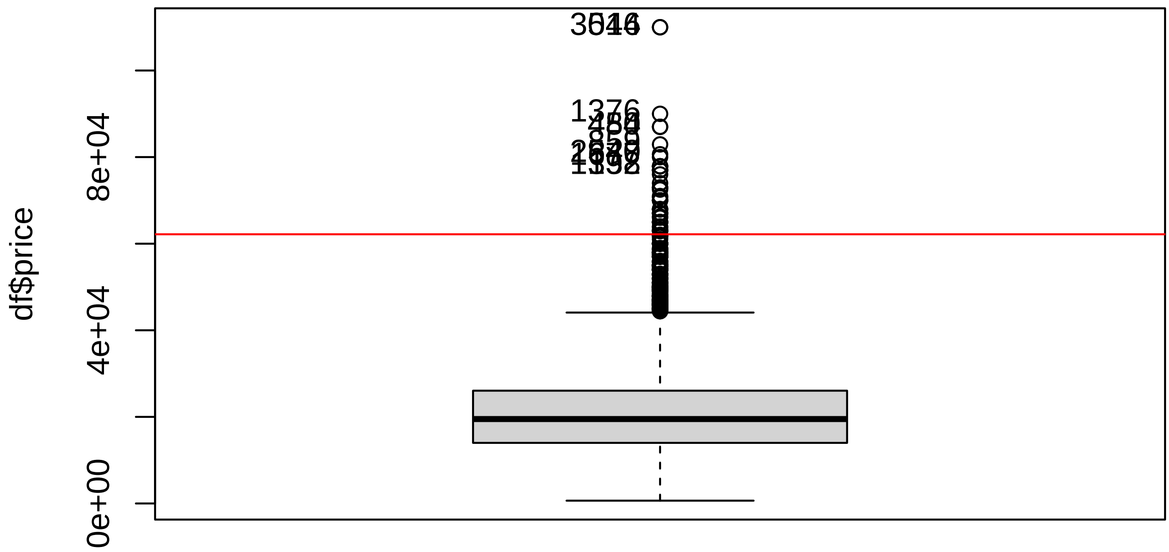
For each variable, we have executed calcQ in order to find the severe/extreme outliers (lower and upper). Then, we have recodified to/with NA the value of the variable of each indivual with a value in the variable less than the under severe outlier or greater than the upper severe outlier to later apply imputation.

4.3.1 price

```
Boxplot(df$price)
```

```
##  [1]  514 3046 1376  450  484  859 2540 1677  192 1338
```

```
var_out<-calcQ(df$price)
abline(h=var_out$souts,col="red")
abline(h=var_out$souti,col="red")
```



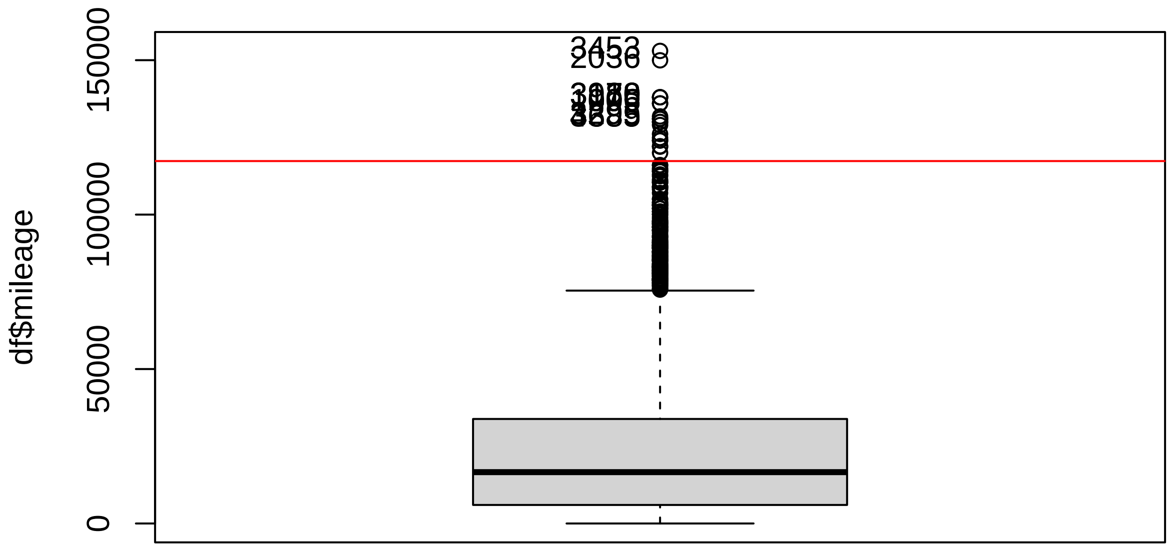
```r
# Outliers:
llout_price<-which((df$price<var_out$souti)|(df$price>var_out$souts))#souts abline
iouts[llout_price]<-iouts[llout_price]+1
jouts[3]<-length(llout_price)
```

4.3.2 mileage

```r
Boxplot(df$mileage)
```

```
##  [1] 3453 2036 3113 3170 3988  995 1006 3295 3385 4689
```

```r
var_out<-calcQ(df$mileage)
abline(h=var_out$souts,col="red")
abline(h=var_out$souti,col="red")
```



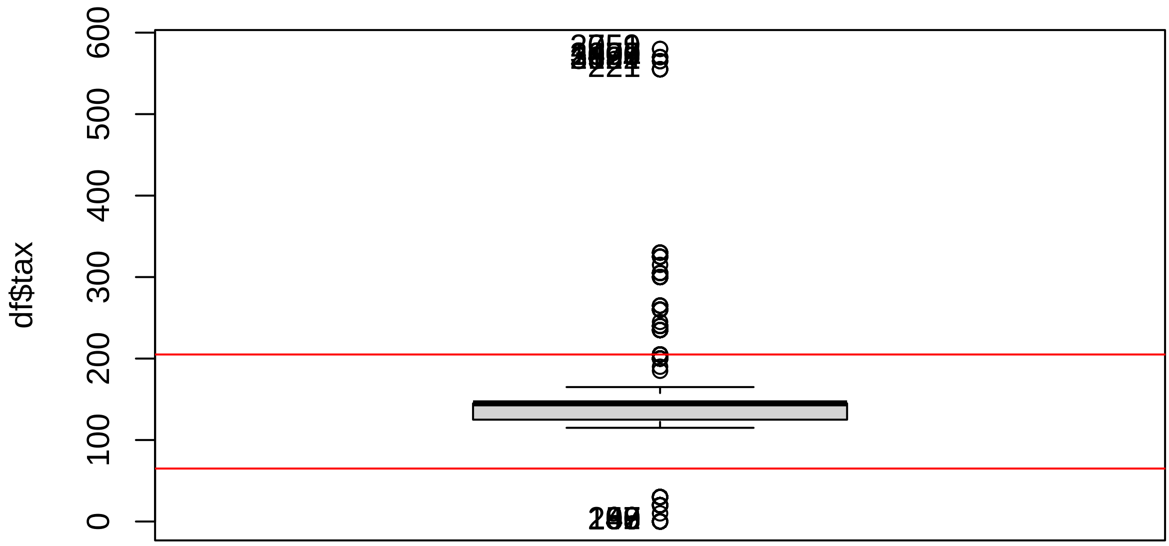

```r
# Outliers:
llout_mileage<-which((df$mileage<var_out$souti)|(df$mileage>var_out$souts))#souts abline
iouts[llout_mileage]<-iouts[llout_mileage]+1
jouts[5]<-length(llout_mileage)
df[llout_mileage,"mileage"]<-NA #llout
```

4.3.3 tax

```r
Boxplot(df$tax)
```

```
##  [1]    7   33   45   47  150  169  182  197  198  209  759 2051 1008 1077 2207
## [16] 3450 2094 3184 3322  221
```

```
var_out<-calcQ(df$tax)
abline(h=var_out$souts,col="red")
abline(h=var_out$souti,col="red")
```



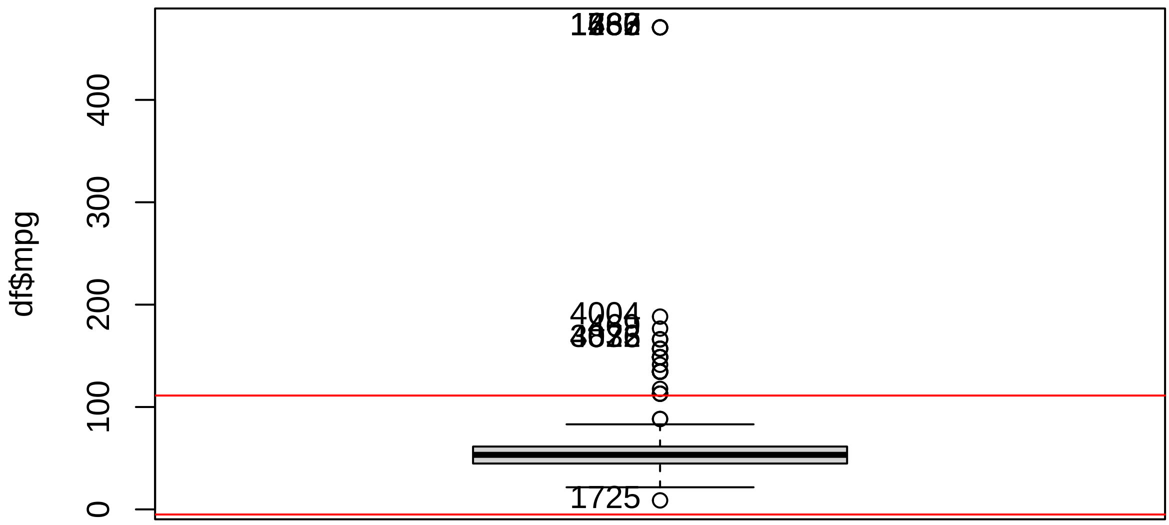

```
# Outliers:
llout_tax<-which((df$tax<var_out$souti)|(df$tax>var_out$souts))#souts abline
iouts[llout_tax]<-iouts[llout_tax]+1
jouts[7]<-length(llout_tax)
df[llout_tax,"tax"]<-NA #llout
```

4.3.4 mpg

```
Boxplot(df$mpg)
```

```
##  [1] 1725 1280 1487 1636 1762 1763 4004  469 3832 4026 4073
```

```
var_out<-calcQ(df$mpg)
abline(h=var_out$souts,col="red")
abline(h=var_out$souti,col="red")
```




```r
# Outliers:
llout_mpg<-which((df$mpg<var_out$souti)|(df$mpg>var_out$souts))#souts abline
iouts[llout_mpg]<-iouts[llout_mpg]+1
jouts[8]<-length(llout_mpg)
df[llout_mpg,"mpg"]<-NA #llout
```

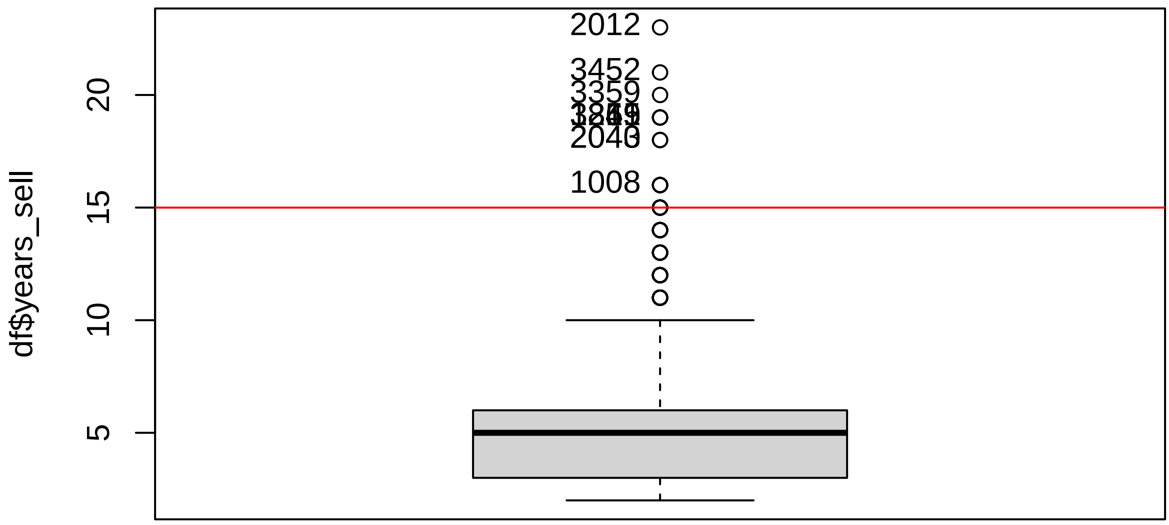
4.3.5 years_sell

He have decided not to assign NA to years_sell because is a special variable with a low number of univariate outliers.

```r
Boxplot(df$years_sell)
```

```
##  [1] 2012 3452 3359 1819 3249 3355 3361 2040 2043 1008
```

```r
var_out<-calcQ(df$years_sell)
abline(h=var_out$souts,col="red")
abline(h=var_out$souti,col="red")
```




```r
# Outliers:
llout_years_sell<-which((df$years_sell<var_out$souti)|(df$years_sell>var_out$souts))#souts abline
iouts[llout_years_sell]<-iouts[llout_years_sell]+1
jouts[11]<-length(llout_years_sell)
#df[llout_years_sell,"years_sell"]<-NA #llout
```

4.4 Number of errors, missings and outliers for individual and variable

4.4.1 Number of missing values of each variable

```r
jmis
```

```
##  [1] 0 0 0 0 0 0 0 0 0 0 0
```

4.4.2 Number of errors per each variable

```r
jerrs
```

```
##  [1] 0 0 0 0 0 0 0 0 0 0 0
```

4.4.3 Number of outilers per each variable

```r
outliers_ranking_sortlist <- sort.list(jouts, decreasing = TRUE)
for(j in outliers_ranking_sortlist) {
  if(!is.na(names(df)[j])) print(paste(names(df)[j], " : ", jouts[j]))
}
```

```
## [1] "tax  :  1292"
## [1] "mpg  :  48"
## [1] "price  :  42"
## [1] "mileage  :  20"
## [1] "years_sell  :  15"
## [1] "model  :  0"
## [1] "year  :  0"
## [1] "transmission  :  0"
## [1] "fuelType  :  0"
## [1] "engineSize  :  0"
## [1] "manufacturer  :  0"
```

4.4.4 Number of missing values individual

```
sum(imis)
```

```
## [1] 0
```

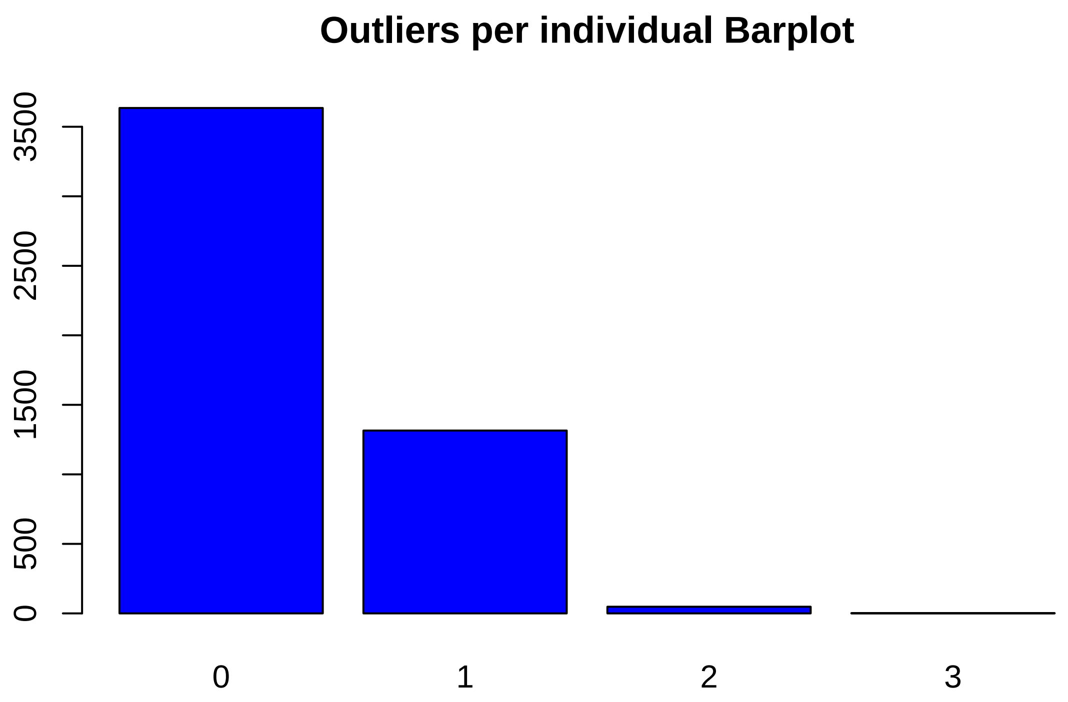
4.4.5 Number of errors individual

```
sum(ierrs)
```

```
## [1] 0
```

4.4.6 Number of outliers individual

```
barplot(table(iouts), main = "Outliers per individual Barplot", col = "blue")
```



4.4.7 New variable adding the total number missing values, outliers and errors

We have created a new variable to know the total missing values, outliers and errors per individual.

```
df$totalMOE<- imis+iouts+ierrs
```

4.5 Imputation

4.5.1 Imputation of numeric variables

We do imputation in order to give a value to the missings (NA's). The (regularized) iterative PCA algorithm first consists imputing missing values with initial values such as the mean of the variable.

```
library(missMDA)

names(df)
```

```
##  [1] "model"        "year"         "price"        "transmission" "mileage"
##  [6] "fuelType"     "tax"          "mpg"          "engineSize"   "manufacturer"
## [11] "years_sell"   "totalMOE"
```

```
vars_con<-names(df)[c(5,7,8)]
vars_dis<-names(df)[c(1:2, 4, 6, 9, 10)]
vars_res<-names(df)[c(3)]

summary(df[,vars_con])
```

```
##     mileage            tax             mpg
##  Min.   :     4   Min.   :115.0   Min.   : 8.80
##  1st Qu.:  5987   1st Qu.:145.0   1st Qu.:44.10
##  Median : 16508   Median :145.0   Median :52.30
##  Mean   : 22876   Mean   :146.8   Mean   :52.71
##  3rd Qu.: 33533   3rd Qu.:145.0   3rd Qu.:61.40
##  Max.   :116000   Max.   :205.0   Max.   :88.30
##  NA's   :20       NA's   :1292    NA's   :48
```

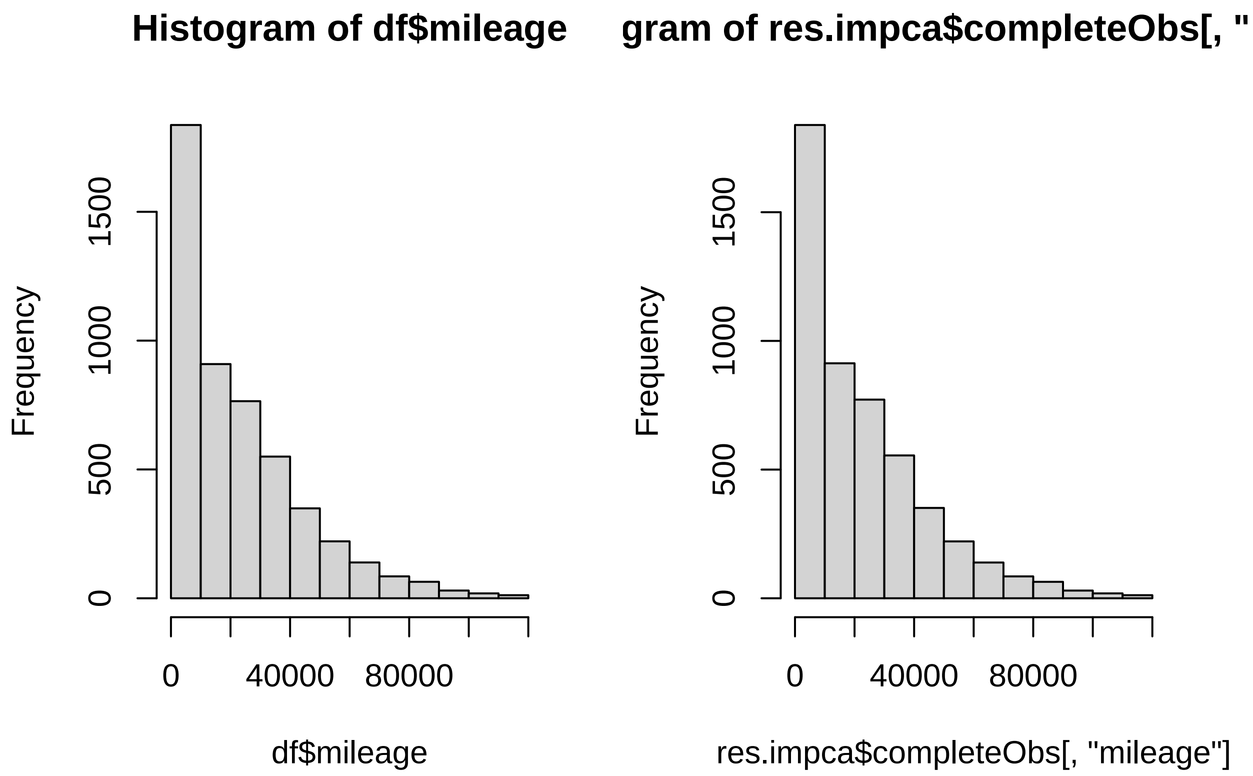
```
res.impca<-imputePCA(df[,vars_con],ncp=2)
summary(res.impca)
```

```
##             Length Class  Mode
## completeObs 15000  -none- numeric
## fittedX     15000  -none- numeric
```

We can notice a difference between the two plots, that means that the imputation has been/was correctly applied.

4.5.1.1 mileage Plot comparison for milage variable after imputation.

```
par(mfrow = c(1,2))
hist(df$mileage)
hist(res.impca$completeObs[,"mileage"])
```




```
quantile(df$mileage,seq(0,1,0.1),na.rm=T)
```

```
##        0%       10%       20%       30%       40%       50%       60%       70%
##       4.0    2000.0    4701.8    7370.7   11623.4   16508.5   22989.0   29712.1
##       80%       90%      100%
##   38260.8   52948.4  116000.0
```

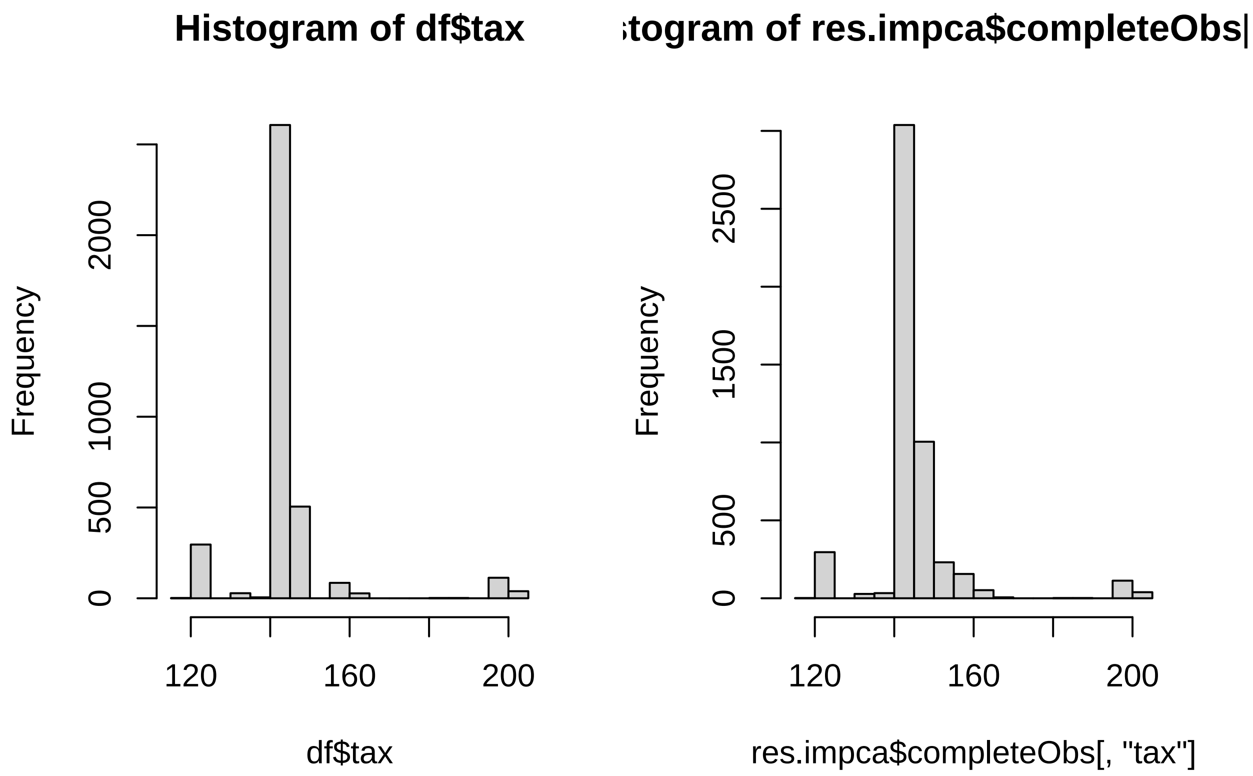
```
round(quantile(res.impca$completeObs[,"mileage"],seq(0,1,0.1),na.rm=T),dig=1)
```

```
##        0%       10%       20%       30%       40%       50%       60%       70%
##       4.0    2000.0    4714.4    7404.0   11648.6   16528.5   23000.0   29753.3
##       80%       90%      100%
##   38221.0   52770.7  116000.0
```

4.5.1.2 tax

Plot comparison for tax variable after imputation.

```
par(mfrow=c(1,2))
hist(df$tax)
hist(res.impca$completeObs[,"tax"])
```




```
quantile(df$tax,seq(0,1,0.1),na.rm=T)
```

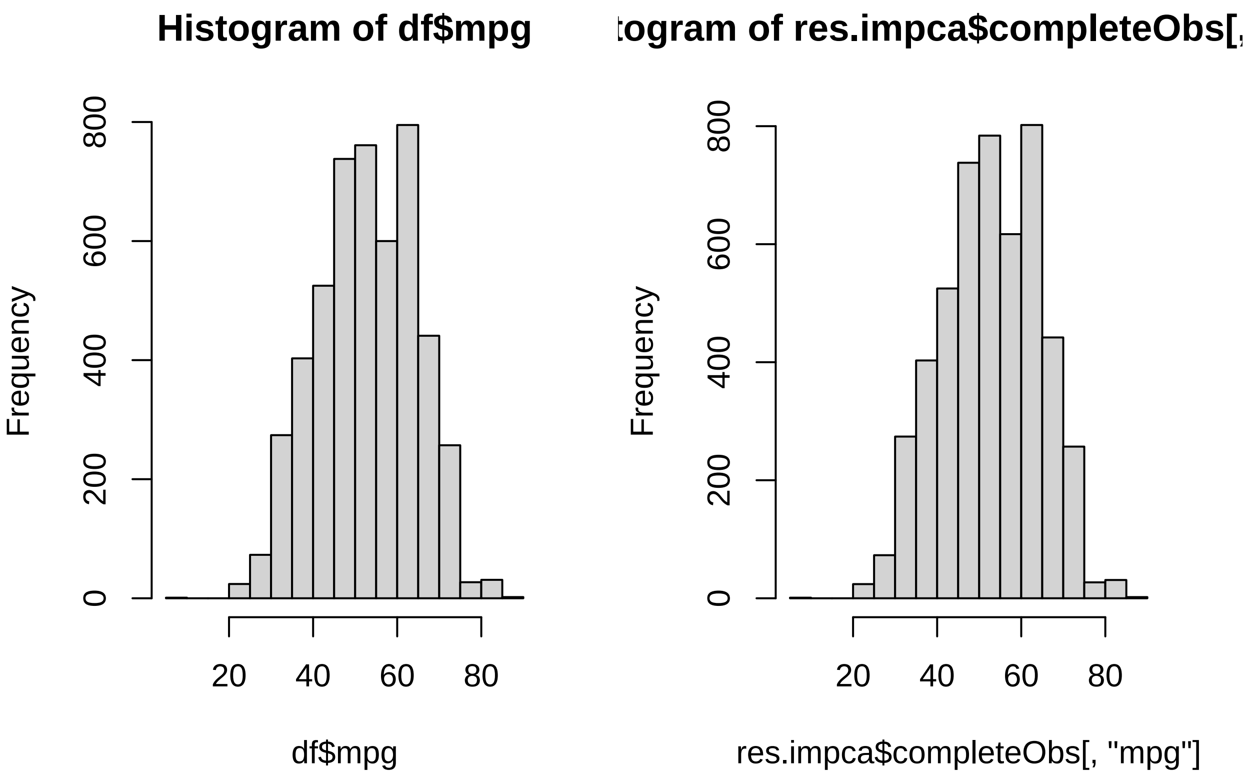
```
##   0%  10%  20%  30%  40%  50%  60%  70%  80%  90% 100%
##  115  145  145  145  145  145  145  145  150  150  205
```

```
round(quantile(res.impca$completeObs[,"tax"],seq(0,1,0.1),na.rm=T),dig=1)
```

```
##    0%   10%   20%   30%   40%   50%   60%   70%   80%   90%  100%
## 115.0 142.7 145.0 145.0 145.0 145.0 145.0 145.7 150.0 151.7 205.0
```

4.5.1.3 mpg Plot comparison for mpg variable after imputation.

```
par(mfrow = c(1,2))
hist(df$mpg)
hist(res.impca$completeObs[,"mpg"])
```




```
quantile(df$mpg,seq(0,1,0.1),na.rm=T)
```

```
##     0%    10%    20%    30%    40%    50%    60%    70%    80%    90%   100%
##   8.80  37.20  42.20  47.03  49.60  52.30  56.50  60.10  62.80  67.30  88.30
```

```
round(quantile(res.impca$completeObs[,"mpg"],seq(0,1,0.1),na.rm=T),dig=1)
```

```
##    0%  10%  20%  30%  40%  50%  60%  70%  80%  90% 100%
##   8.8 37.2 42.2 47.1 49.6 53.3 56.5 60.1 62.8 67.3 88.3
```

```
df[,vars_con]<-res.impca$completeObs
```

4.5.2 Imputation of factor variables

We do imputation in order to give a value to the missings (NA's). The (regularized) iterative MCA algorithm first consists in coding the categorical variables using the indicator matrix of dummy variables. Then, in the initialization step, missing values are imputed with initial values such as the proportion of the category for each category using the non-missing entries.

```
summary(df[,vars_dis])
```

```
##                model                    year                    transmission
##   VW-Golf          : 488   f.Year-2019:1597   f.Trans-Manual   :1795
##   Mercedes-C Class: 367   f.Year-2017: 887   f.Trans-SemiAuto :1892
##   VW-Polo          : 348   f.Year-2016: 862   f.Trans-Automatic:1313
##   Mercedes-A Class: 270   f.Year-2018: 493
##   BMW-3 Series     : 243   f.Year-2015: 419
##   Audi-A3          : 199   f.Year-2020: 321
##   (Other)          :3085   (Other)    : 421
##          fuelType             engineSize      manufacturer
##   f.Fuel-Diesel:2842   f.EngSize-2  :2079   Audi     :1085
##   f.Fuel-Hybrid:   55   f.EngSize-3  : 562   BMW      :1049
```

```
##   f.Fuel-Other :   12    f.EngSize-1.5: 553    Mercedes:1351
##   f.Fuel-Petrol:2091    f.EngSize-1  : 407    VW      :1515
##                         f.EngSize-2.1: 407
##                         f.EngSize-1.6: 353
##                         (Other)      : 639
```

```
res.immca<-imputeMCA(df[,vars_dis],ncp=10)
summary(res.immca$completeObs)
```

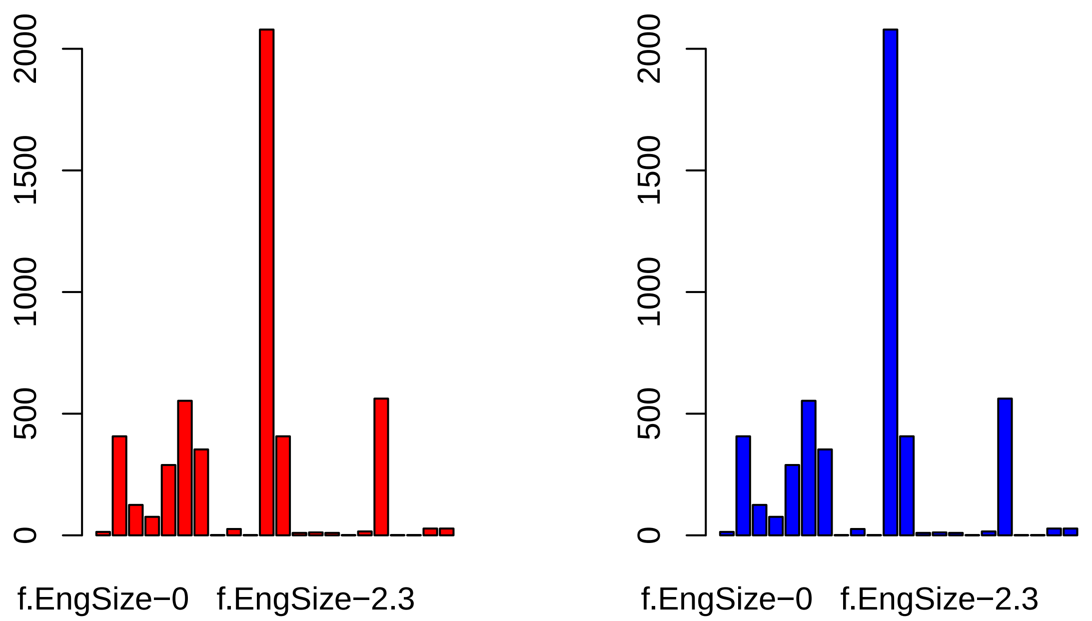
```
##                model                  year                    transmission
##   VW-Golf          : 488    f.Year-2019:1597    f.Trans-Manual   :1795
##   Mercedes-C Class: 367    f.Year-2017: 887    f.Trans-SemiAuto :1892
##   VW-Polo          : 348    f.Year-2016: 862    f.Trans-Automatic:1313
##   Mercedes-A Class: 270    f.Year-2018: 493
##   BMW-3 Series     : 243    f.Year-2015: 419
##   Audi-A3          : 199    f.Year-2020: 321
##   (Other)          :3085    (Other)    : 421
##           fuelType              engineSize      manufacturer
##   f.Fuel-Diesel:2842    f.EngSize-2  :2079    Audi    :1085
##   f.Fuel-Hybrid:   55    f.EngSize-3  : 562    BMW     :1049
##   f.Fuel-Other :   12    f.EngSize-1.5: 553    Mercedes:1351
##   f.Fuel-Petrol:2091    f.EngSize-1  : 407    VW      :1515
##                         f.EngSize-2.1: 407
##                         f.EngSize-1.6: 353
##                         (Other)      : 639
```

We can notice a difference between the two plots, that means that the imputation has been/was correctly applied.

4.5.2.1 engineSize

Plot comparison for engin size variable.

```
par(mfrow=c(1,2))
barplot(table(df$engineSize),col="red")
barplot(table(res.immca$completeObs[,"engineSize"]),col="blue")
```



```
df[ , vars_dis ]<-res.immca$completeObs
sum(countNA(df)$mis_ind)==0
```

```
## [1] TRUE
```

4.6 Correlation of numeric variables with MOE

```
cor(df[,c(3,5,7,8,11)], df$totalMOE)
```

```
##                   [,1]
## price      -0.29796314
## mileage     0.45404451
## tax         0.02667251
## mpg         0.32802523
## years_sell  0.52600573
```

As we can see years_sell is one of the variables with most correlation with total_MOE this means that as the time of a car being sold grows more tendency to have errors, outliers or/and missing increase.

4.7 Discretization

We do discretization in order to make it easier to understand the numeric variables.

4.7.1 price variable

```
quantile(df$price,seq(0,1,0.25),na.rm=TRUE)
```

```
##        0%       25%       50%       75%      100%
##    650.00  13995.00  19498.00  26039.25 109990.00
```

```
quantile(df$price,seq(0,1,0.1),na.rm=TRUE)
```

```
##       0%      10%      20%      30%      40%      50%      60%      70%
##    650.0  10318.0  12890.0  15145.5  17399.6  19498.0  21989.4  24904.5
##      80%      90%     100%
##  28000.0  33950.2 109990.0
```

```
df$aux_price<-factor(cut(df$price/1000,breaks=c(quantile(df$price,seq(0,1,0.25),na.rm=TRUE))/1000,inclu
summary(df$aux_price)
```

```
## [0.65,14] (14,19.5] (19.5,26]  (26,110]
##      1259      1246      1245      1250
```

```
tapply(df$price,df$aux_price,median)
```

```
## [0.65,14] (14,19.5] (19.5,26]  (26,110]
##   10995.0   16950.0   22646.0   31986.5
```

```
levels(df$aux_price)<-paste("f.price-",levels(df$aux_price),sep="")
table(df$aux_price,useNA="always")
```

```
##
## f.price-[0.65,14] f.price-(14,19.5] f.price-(19.5,26]  f.price-(26,110]
##              1259              1246              1245              1250
##              <NA>
##                 0
```

4.7.2 mileage variable

```
df$aux_mileage<-factor(cut(df$mileage,breaks=c(quantile(df$mileage,seq(0,1,0.25),na.rm=TRUE)),include.lo
summary(df$aux_mileage)
```

```
##            [4,6e+03]     (6e+03,1.65e+04] (1.65e+04,3.35e+04] (3.35e+04,1.16e+05]
##                 1252                 1248                 1250                 1250
```

```
tapply(df$mileage,df$aux_mileage,median)
```

```
##            [4,6e+03]     (6e+03,1.65e+04] (1.65e+04,3.35e+04] (3.35e+04,1.16e+05]
##               2753.5              10396.0              24443.5              48095.0
```

```
levels(df$aux_mileage)<-paste("f.mileage-",levels(df$aux_mileage),sep="")
table(df$aux_mileage,useNA="always")
```

```
##
##            f.mileage-[4,6e+03]     f.mileage-(6e+03,1.65e+04]
##                           1252                           1248
## f.mileage-(1.65e+04,3.35e+04] f.mileage-(3.35e+04,1.16e+05]
##                           1250                           1250
##                           <NA>
##                              0
```

4.7.3 tax variable

```
quantile(df$tax,seq(0,1,0.25),na.rm=TRUE)
```

```
##       0%      25%      50%      75%     100%
## 115.0000 145.0000 145.0000 148.0945 205.0000
```

```
df$aux_tax<-factor(cut(df$tax,breaks=c(0, 125, 145, 580),include.lowest = T ))
summary(df$aux_tax)
```

```
##   [0,125] (125,145] (145,580]
##       297      3099      1604
```

```
tapply(df$tax,df$aux_tax,median)
```

```
##   [0,125] (125,145] (145,580]
##       125       145       150
```

```
levels(df$aux_tax)<-paste("f.tax-",levels(df$aux_tax),sep="")
table(df$aux_tax,useNA="always")
```

```
##
##   f.tax-[0,125] f.tax-(125,145] f.tax-(145,580]            <NA>
##             297            3099            1604               0
```

4.7.4 mpg variable

```
quantile(df$mpg,seq(0,1,0.25),na.rm=TRUE)
```

```
##   0%  25%  50%  75% 100%
##  8.8 44.8 53.3 61.4 88.3
```

```
df$aux_mpg<-factor(cut(df$mpg,breaks=c(quantile(df$mpg,seq(0,1,0.25),na.rm=TRUE)),include.lowest = T ))
summary(df$aux_mpg)
```

```
##  [8.8,44.8] (44.8,53.3] (53.3,61.4] (61.4,88.3]
##        1300        1378        1188        1134
```

```
tapply(df$mpg,df$aux_mpg,median)
```

```
##  [8.8,44.8] (44.8,53.3] (53.3,61.4] (61.4,88.3]
##        39.2        49.6        57.7        67.3
```

```
levels(df$aux_mpg)<-paste("f.mpg-",levels(df$aux_mpg),sep="")
table(df$aux_mpg,useNA="always")
```

```
##
##  f.mpg-[8.8,44.8] f.mpg-(44.8,53.3] f.mpg-(53.3,61.4] f.mpg-(61.4,88.3]
##              1300              1378              1188              1134
##              <NA>
##                 0
```

4.7.5 years_sell variable

```
df$aux_years_sell<-factor(cut(df$years_sell,breaks=c(quantile(df$years_sell,seq(0,1,0.25),na.rm=TRUE)),i
summary(df$aux_years_sell)
```

```
##  [2,3]  (3,5]  (5,6] (6,23]
##   1918   1380    862    840
```

```
tapply(df$years_sell,df$aux_years_sell,median)
```

```
##  [2,3]  (3,5]  (5,6] (6,23]
##      3      5      6      8
```

```
levels(df$aux_years_sell)<-paste("f.years_sell-",levels(df$aux_years_sell),sep="")
table(df$aux_years_sell,useNA="always")
```

```
##
##  f.years_sell-[2,3]  f.years_sell-(3,5]  f.years_sell-(5,6] f.years_sell-(6,23]
##                1918                1380                 862                 840
##                <NA>
##                   0
```

4.8 Definition of binary outcome: Audi

Create binary target, define lists of numeric and qualitative variables and save your raw base database

We have created a binary target to know if the car is Audi or not. We will use this variable later to do profiling with it.

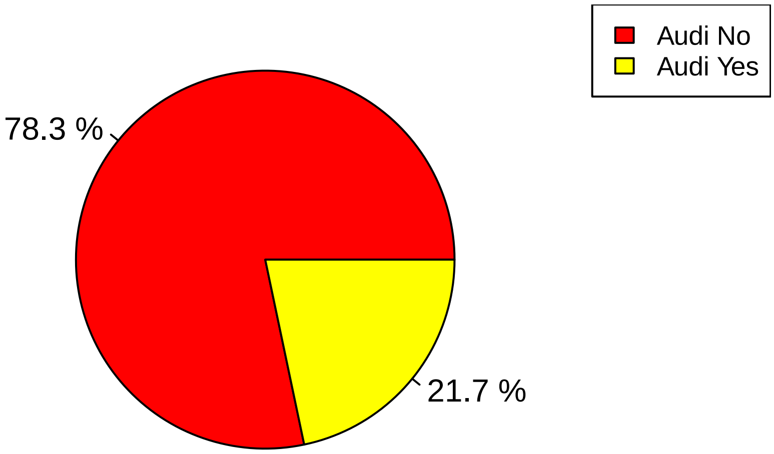
```
# Binary Target: Audi?
df$Audi<-ifelse(df$manufacturer == "Audi",1,0)
df$Audi<-factor(df$Audi,labels=paste("Audi",c("No","Yes")))
summary(df$Audi)
```

```
##  Audi No Audi Yes
##     3915     1085
```

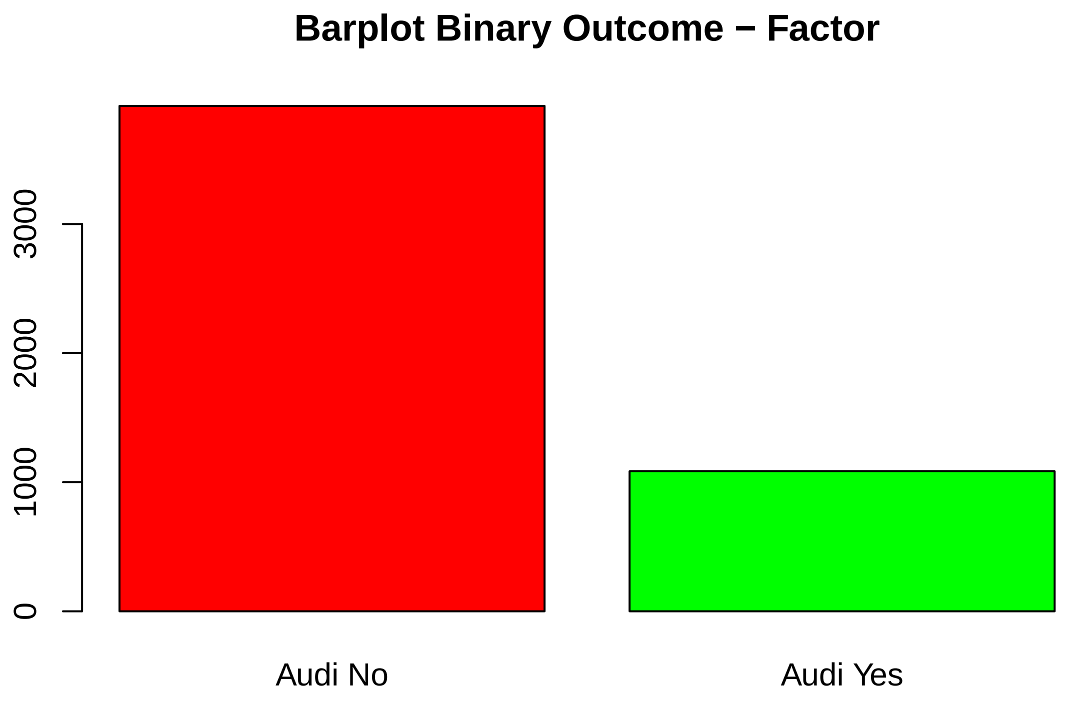
```r
# Pie
piepercent<-round(100*(table(df$Audi)/nrow(df)),dig=2); piepercent
```

```
##
##  Audi No Audi Yes
##     78.3     21.7
```

```r
pie(table(df$Audi),col=heat.colors(2),labels=paste(piepercent,"%"))
legend("topright", levels(df$Audi), cex = 0.8, fill = heat.colors(2))
```




```r
# Bar Chart
barplot(table(df$Audi),main="Barplot Binary Outcome - Factor",col=c("red","green"))
```



4.9 Multivariant outliers

We have executed Moutlier function in order to find the multivariant outliers. Then, we have created a new variable (df$mout) in order to distinguish/differentiate individuals that are multivariant outliers (they have a robust distance greater than the cutoff distance) and individuals that not.

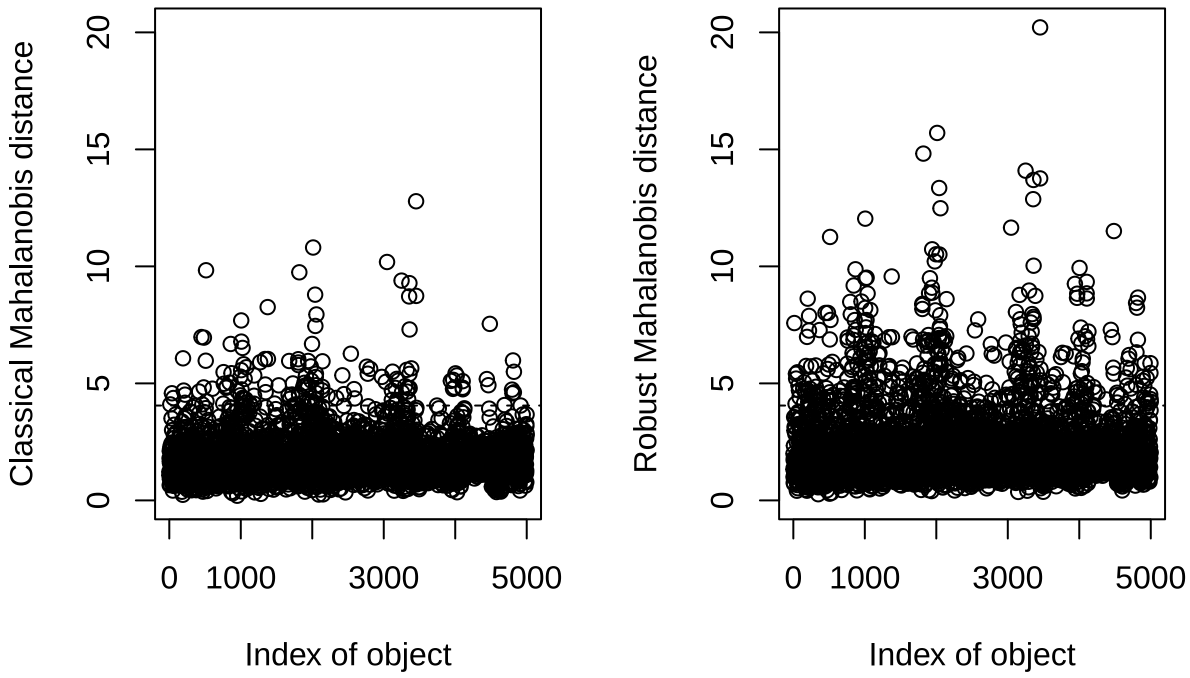
```
library(chemometrics)
```

```
## Loading required package: rpart
```

```
summary(df[,c(3, 5, 8, 11)])
```

```
##      price           mileage            mpg          years_sell
##  Min.   :   650   Min.   :     4   Min.   : 8.80   Min.   : 2.000
##  1st Qu.: 13995   1st Qu.:  5999   1st Qu.:44.80   1st Qu.: 3.000
##  Median : 19498   Median : 16528   Median :53.30   Median : 5.000
##  Mean   : 21470   Mean   : 22888   Mean   :52.74   Mean   : 4.787
##  3rd Qu.: 26039   3rd Qu.: 33516   3rd Qu.:61.40   3rd Qu.: 6.000
##  Max.   :109990   Max.   :116000   Max.   :88.30   Max.   :23.000
```

```
mout<-Moutlier(df[,c(3, 5, 8, 11)],quantile = 0.9975, plot = TRUE)
```




```
length(which(mout$rd>mout$cutoff))
```

```
## [1] 464
```

```
ll<-which(mout$rd>mout$cutoff)
Boxplot(mout$rd)
```

```
##  [1] 3452 2012 1819 3249 3453 3359 2040 3355 2058 1006
```

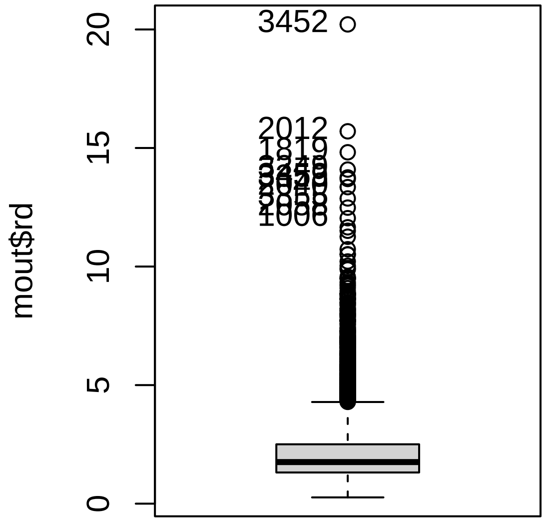
```
df[ll[1:3],c(3,5,8,11)]
```

```
##    price mileage  mpg years_sell
## 14 16200   89334 62.8          6
## 31 56985    1510 33.2          3
## 37 63985    8450 32.8          4
```

```
df$mout <- 0
df$mout[ ll ]<-1
df$mout <- factor( df$mout, labels=c( "NoMOut","YesMOut"))
table(df$mout)
```

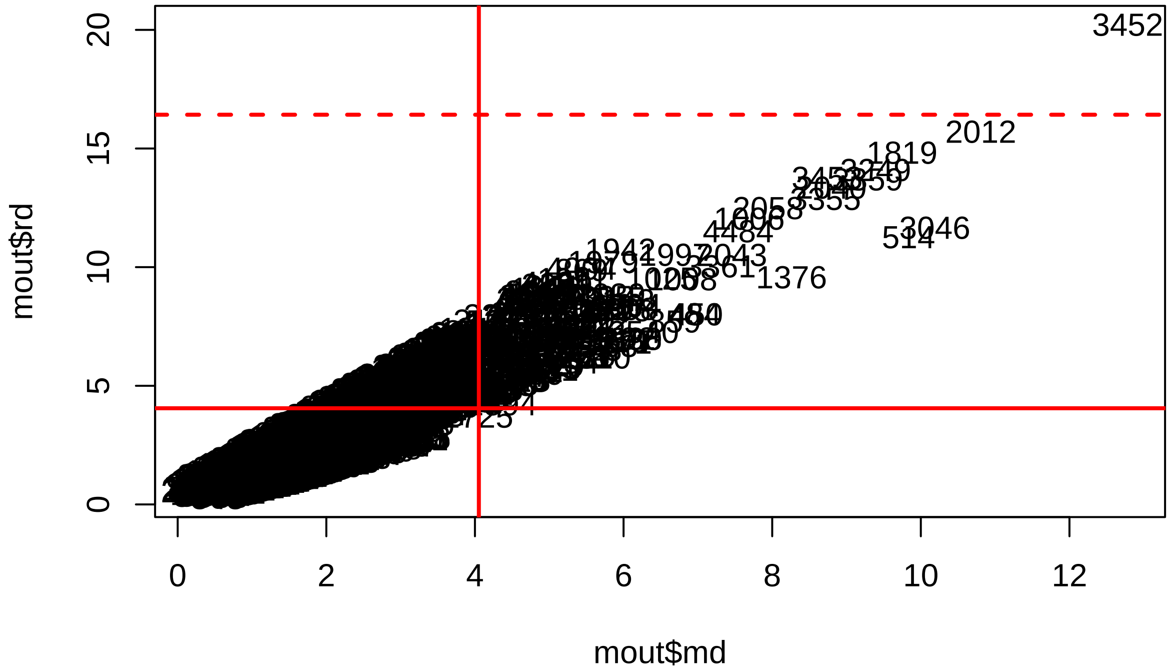
```
##
##  NoMOut YesMOut
##    4536     464
```

```
par(mfrow=c(1,1))
```




```
plot(mout$md,mout$rd, type="n")
text(mout$md,mout$rd,labels=rownames(df[,vars_con]))
abline(h=mout$cutoff,col="red",lwd=2)
abline(v=mout$cutoff,col="red",lwd=2)

abline(h=mout$cutof^2,col="red",lwd=2,lty=2)
abline(v=mout$cutoff^2,col="red",lwd=2,lty=2)
```



4.10 Profiling

```
library(FactoMineR)
summary(df$price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     650   13995   19498   21470   26039  109990
```

```
res.condes<-condes(df,3)
res.condes$quanti  # Global association to numeric variables
```

```
##            correlation       p.value
## totalMOE    -0.2979631 4.715577e-103
## mileage     -0.5131584  0.000000e+00
## years_sell  -0.5538620  0.000000e+00
## mpg         -0.5903613  0.000000e+00
```

```
res.condes$quali # Global association to factors
```

```
##                         R2       p.value
## model          0.515041082  0.000000e+00
## year           0.353483215  0.000000e+00
## engineSize     0.413280059  0.000000e+00
## aux_price      0.731460406  0.000000e+00
## aux_mileage    0.293743806  0.000000e+00
## aux_mpg        0.300971015  0.000000e+00
## aux_years_sell 0.321950885  0.000000e+00
## transmission   0.230475968 5.322731e-285
## manufacturer   0.092467374 9.441791e-105
## aux_tax        0.089992603 4.717795e-103
## Audi           0.004668061  1.327290e-06
## fuelType       0.003656164  3.811516e-04
```

```
res.condes$category  # Partial association to significative levels in factors
```

```
##                                               Estimate       p.value
## aux_years_sell=f.years_sell-[2,3]           9251.70250  0.000000e+00
## aux_mpg=f.mpg-[8.8,44.8]                    9723.17533  0.000000e+00
## aux_price=f.price-(26,110]                 14249.54562  0.000000e+00
## aux_mileage=f.mileage-[4,6e+03]             8087.75077 3.393062e-225
## year=f.Year-2019                           12374.45389 3.012776e-202
## engineSize=f.EngSize-3                     12016.91222 5.157298e-195
## transmission=f.Trans-SemiAuto               4505.05997 1.119939e-131
## aux_tax=f.tax-(125,145]                     4770.39249 1.736430e-103
## year=f.Year-2020                           16602.91086  2.019709e-77
## engineSize=f.EngSize-4                     36182.92607  1.924377e-73
## year=f.Year-2016                             469.10740  8.174011e-60
## engineSize=f.EngSize-4.2 or more           27624.31893  3.490423e-43
## model=Audi-Q8                              43576.84832  3.318259e-41
## manufacturer=Mercedes                       3039.49797  2.027962e-40
## model=Mercedes-GLE Class                   10820.23923  7.120933e-37
## model=BMW-X7                               42413.20546  1.314652e-34
## model=Audi-Q7                              13708.16082  6.673097e-29
## model=Mercedes-GLC Class                    3183.60042  8.847262e-28
## engineSize=f.EngSize-2.9                   28576.84571  8.454111e-27
## transmission=f.Trans-Automatic              2531.21056  7.568797e-26
## aux_mileage=f.mileage-(6e+03,1.65e+04]      2756.38254  2.237589e-25
## year=f.Year-2017                            2745.46170  1.557611e-21
## model=Audi-Q5                               2914.98273  6.957440e-21
## model=Mercedes-S Class                     17293.72332  3.925870e-20
```

```
## model=Mercedes-GLS Class                      18533.56260  1.822382e-19
## model=BMW-8 Series                            35992.74832  2.204029e-19
## model=BMW-X5                                   8463.71195  1.821812e-17
## model=BMW-M5                                  37683.09832  6.285862e-17
## model=Audi-R8                                 56033.34832  9.515733e-17
## model=BMW-M4                                  18969.53014  4.519729e-16
## model=Mercedes-SL CLASS                        4815.03253  2.006258e-12
## model=VW-Touareg                               4073.84832  1.374518e-09
## aux_years_sell=f.years_sell-(3,5]               645.29335  1.510423e-08
## model=BMW-X6                                  15456.63403  2.045855e-08
## model=BMW-X3                                    364.01498  4.555479e-07
## model=BMW-7 Series                             7855.09832  9.081694e-07
## engineSize=f.EngSize-2                          609.28213  1.057731e-06
## Audi=Audi Yes                                   898.85989  1.327290e-06
## manufacturer=Audi                              1120.63823  1.327290e-06
## model=BMW-M2                                  15534.34832  2.007301e-06
## aux_price=f.price-(19.5,26]                    1231.36722  2.727027e-06
## model=Audi-RS6                                20839.68165  5.941713e-06
## model=VW-Caravelle                             9357.72332  1.085753e-05
## model=Audi-A8                                  4406.53582  1.100877e-05
## model=Audi-RS5                                19173.34832  2.030061e-05
## manufacturer=BMW                                975.38554  2.200900e-05
## model=BMW-X4                                   3188.29276  2.841145e-05
## model=Audi-RS4                                23962.34832  4.062218e-05
## fuelType=f.Fuel-Diesel                           81.36076  6.524733e-05
## year=f.Year-2018                               7588.80081  2.276380e-04
## model=BMW-i8                                  15037.68165  3.187854e-04
## model=Audi-S4                                 11958.09832  3.326456e-04
## model=BMW-X2                                    696.07559  3.885212e-04
## model=VW-California                           29034.34832  7.553580e-04
## model=BMW-M3                                   4761.68165  5.636620e-03
## model=Audi-RS3                                 3637.14832  2.176207e-02
## engineSize=f.EngSize-2.5                       6487.23321  4.836521e-02
## model=VW-Tiguan Allspace                      -1746.22311  4.737339e-02
## model=VW-Jetta                               -19960.98502  4.630031e-02
## model=Mercedes-CLC Class                     -23029.15168  4.266624e-02
## model=VW-Touran                              -11006.95937  4.185280e-02
## model=Mercedes-X-CLASS                         -771.46986  3.982262e-02
## aux_mpg=f.mpg-(44.8,53.3]                      -258.09211  3.326596e-02
## model=VW-Arteon                               -2896.28131  2.745189e-02
## model=VW-Golf SV                             -12467.49168  2.132947e-02
## model=BMW-3 Series                            -9145.48707  1.449274e-02
## model=VW-Amarok                                -512.27668  9.982846e-03
## model=Mercedes-SLK                           -20110.85168  9.206574e-03
## model=Mercedes-V Class                         -452.42946  5.836196e-03
## model=Mercedes-CLS Class                       -370.31835  5.286254e-03
## model=VW-Beetle                              -18955.36597  5.108585e-03
## engineSize=f.EngSize-1.5                      -1747.80132  7.195178e-04
## model=VW-CC                                  -18679.10623  6.089175e-04
## model=VW-Scirocco                            -15278.40168  4.165586e-04
## model=Mercedes-A Class                       -10172.40353  2.833106e-05
## fuelType=f.Fuel-Petrol                        -1217.90283  2.275137e-05
## engineSize=f.EngSize-1.8                      -9426.92832  1.590542e-05
## engineSize=f.EngSize-2.1                      -2501.31936  1.564794e-05
## year=f.Year-2012                              -4887.18916  1.434448e-05
## model=Mercedes-C Class                        -4932.22934  2.721980e-06
## Audi=Audi No                                   -898.85989  1.327290e-06
## model=VW-Passat                              -12997.69424  6.406054e-07
## model=Audi-A3                                -11593.82254  4.796664e-08
## year=f.Year-2011                              -7439.76560  1.630019e-08
## model=Mercedes-E Class                        -2977.47390  1.281271e-08
## year=f.Year-2010                              -8897.47989  3.665675e-10
## model=Audi-A1                                -14095.87796  4.173040e-13
## model=BMW-1 Series                           -13141.11869  7.132624e-14
```

```
## aux_tax=f.tax-[0,125]                      -3090.21034  2.050124e-18
## year=f.Year-2009 or before                 -9958.24179  1.363765e-21
## model=VW-Golf                             -12379.53283  5.234941e-26
## engineSize=f.EngSize-1.4                   -7037.38028  4.806127e-28
## year=f.Year-2013                           -5024.10129  1.983361e-29
## model=VW-Up                               -20805.93577  1.276864e-31
## year=f.Year-2014                           -3095.45499  8.013091e-33
## engineSize=f.EngSize-1.2                  -11809.30679  7.807887e-34
## engineSize=f.EngSize-1.6                   -7001.12118  4.564378e-34
## year=f.Year-2015                            -478.50193  1.365347e-36
## aux_mileage=f.mileage-(1.65e+04,3.35e+04]  -3644.94145  1.161832e-43
## aux_mpg=f.mpg-(53.3,61.4]                  -3602.22515  5.344170e-46
## aux_years_sell=f.years_sell-(5,6]          -3361.32631  8.174011e-60
## engineSize=f.EngSize-1                     -8959.45696  1.684048e-65
## aux_price=f.price-(14,19.5]                -4657.16617  4.332775e-70
## aux_tax=f.tax-(145,580]                    -1680.18215  5.578448e-71
## model=VW-Polo                             -17500.47352  1.353249e-73
## manufacturer=VW                            -5135.52174 8.812113e-101
## aux_mpg=f.mpg-(61.4,88.3]                  -5862.85807 4.311250e-109
## aux_years_sell=f.years_sell-(6,23]         -6535.66954 4.910168e-149
## aux_mileage=f.mileage-(3.35e+04,1.16e+05]  -7199.19185 9.366109e-175
## transmission=f.Trans-Manual                -7036.27053 1.829258e-279
## aux_price=f.price-[0.65,14]               -10823.74668  0.000000e+00
```

As we can see in the output condes$quanti, price variable is related with tax, mpg, mileage and years_sell variables because their p-values are less than 0.05. More concretely, against more price less mileage, less years_sell and less mpg.

As we can see in the output condes$quali, model variable has a high correlation, i.e. certain models have high and/or low prices.

Regarding the output of condes$category, we can feature cars with model Audi-R8 which on average cost 56033.34832£ than baseline price, cars with model Audi-Q8 which on average cost 43576.84832£ more than baseline price and cars with model BMW-X7 which on average cost 42413.20546£ more than baseline price.

On the other hand, we can feature cars with model Mercedes-CLC Class which on average cost 23029.15168£ less than baseline price, cars with model Mercedes-SLK which on average cost 20110.85168£ less than baseline price and finally cars with model VW-Up which on average cost 20805.93577£ less than baseline price.

```r
summary(df$Audi)
```

```
##  Audi No Audi Yes
##     3915     1085
```

```r
res.catdes<-catdes(df,18)
res.catdes$quanti.var  # Global association to numeric variables
```

```
##                Eta2      P-value
## mpg     0.009703866 2.934540e-12
## price   0.004668061 1.327290e-06
## tax     0.004417699 2.552799e-06
## mileage 0.001260481 1.205183e-02
```

```r
res.catdes$quanti # Partial association of numeric variables to levels of outcome factor
```

```
## $`Audi No`
##            v.test Mean in category Overall mean sd in category  Overall sd
## mpg      6.964885         53.33278     52.73678       11.34995    11.49277
## mileage -2.510208      22486.99051  22887.97553    21188.80333 21454.13386
## tax     -4.699370        146.58458    146.98985       11.00856    11.58247
## price   -4.830697      21080.13921  21470.24440    10650.88867 10845.87254
##              p.value
## mpg     3.286717e-12
```

```
## mileage 1.206600e-02
## tax     2.609653e-06
## price   1.360558e-06
##
## $`Audi Yes`
##            v.test Mean in category Overall mean sd in category  Overall sd
## price    4.830697      22877.85899  21470.24440    11411.69143 10845.87254
## tax      4.699370        148.45220    146.98985       13.34934    11.58247
## mileage  2.510208      24334.84772  22887.97553    22325.59521 21454.13386
## mpg     -6.964885         50.58624     52.73678       11.74518    11.49277
##              p.value
## price   1.360558e-06
## tax     2.609653e-06
## mileage 1.206600e-02
## mpg     3.286717e-12
```

```
res.catdes$test.chi2 # Global association to factors
```

```
##                   p.value df
## model        0.000000e+00 87
## manufacturer 0.000000e+00  3
## engineSize   1.938026e-76 21
## aux_mpg      1.676014e-16  3
## aux_price    2.927530e-06  3
## fuelType     8.418028e-06  3
## mout         2.762592e-03  1
## transmission 4.235600e-03  2
## aux_tax      1.645093e-02  2
## aux_mileage  2.230200e-02  3
```

```
res.catdes$category  # Partial association to significative levels in factors
```

```
## $`Audi No`
##                                             Cla/Mod     Mod/Cla Global
## manufacturer=VW                           100.00000 38.69731801  30.30
## manufacturer=Mercedes                     100.00000 34.50830140  27.02
## manufacturer=BMW                          100.00000 26.79438059  20.98
## model=VW-Golf                             100.00000 12.46487867   9.76
## engineSize=f.EngSize-2.1                  100.00000 10.39591315   8.14
## model=Mercedes-C Class                    100.00000  9.37420179   7.34
## model=VW-Polo                             100.00000  8.88888889   6.96
## model=Mercedes-A Class                    100.00000  6.89655172   5.40
## model=BMW-3 Series                        100.00000  6.20689655   4.86
## model=BMW-1 Series                        100.00000  5.03192848   3.94
## model=Mercedes-E Class                    100.00000  4.59770115   3.60
## model=VW-Tiguan                           100.00000  4.18901660   3.28
## model=BMW-2 Series                        100.00000  3.52490421   2.76
## model=Mercedes-GLC Class                  100.00000  3.03959132   2.38
## model=BMW-4 Series                        100.00000  2.50319285   1.96
## model=VW-Passat                           100.00000  2.40102171   1.88
## model=VW-Up                               100.00000  2.24776501   1.76
## model=BMW-5 Series                        100.00000  2.19667944   1.72
## model=Mercedes-GLA Class                  100.00000  2.14559387   1.68
## engineSize=f.EngSize-1.3                  100.00000  1.94125160   1.52
## model=BMW-X1                              100.00000  1.89016603   1.48
## engineSize=f.EngSize-1.2                   96.00000  3.06513410   2.50
## engineSize=f.EngSize-1.5                   86.79928 12.26053640  11.06
## model=VW-T-Roc                            100.00000  1.60919540   1.26
## model=Mercedes-B Class                    100.00000  1.60919540   1.26
## aux_price=f.price-[0.65,14]                83.32010 26.79438059  25.18
## model=Mercedes-CL Class                   100.00000  1.45593870   1.14
## model=Mercedes-GLE Class                  100.00000  1.40485313   1.10
```

```
## aux_mpg=f.mpg-(61.4,88.3]                83.33333 24.13793103  22.68
## model=BMW-X3                            100.00000  1.22605364   0.96
## aux_mpg=f.mpg-(53.3,61.4]                82.65993 25.08301405  23.76
## model=VW-Touran                         100.00000  0.99616858   0.78
## model=Mercedes-SL CLASS                 100.00000  0.97062580   0.76
## fuelType=f.Fuel-Hybrid                   96.36364  1.35376756   1.10
## model=BMW-X5                            100.00000  0.84291188   0.66
## model=VW-Touareg                        100.00000  0.81736909   0.64
## engineSize=f.EngSize-1                   84.52088  8.78671775   8.14
## model=VW-Arteon                         100.00000  0.68965517   0.54
## model=VW-T-Cross                        100.00000  0.63856960   0.50
## model=VW-Golf SV                        100.00000  0.63856960   0.50
## model=VW-Scirocco                       100.00000  0.61302682   0.48
## fuelType=f.Fuel-Diesel                   79.80296 57.93103448  56.84
## mout=NoMOut                              78.85802 91.36653895  90.72
## model=VW-Sharan                         100.00000  0.56194125   0.44
## model=BMW-X2                            100.00000  0.56194125   0.44
## aux_tax=f.tax-[0,125]                    84.17508  6.38569604   5.94
## transmission=f.Trans-SemiAuto            80.23256 38.77394636  37.84
## model=Mercedes-V Class                  100.00000  0.45977011   0.36
## model=Mercedes-CLS Class                100.00000  0.45977011   0.36
## model=BMW-X4                            100.00000  0.45977011   0.36
## model=VW-Amarok                         100.00000  0.40868455   0.32
## model=Mercedes-S Class                  100.00000  0.40868455   0.32
## model=BMW-Z4                            100.00000  0.38314176   0.30
## model=VW-Tiguan Allspace                100.00000  0.35759898   0.28
## model=Mercedes-GLS Class                100.00000  0.35759898   0.28
## model=Mercedes-GL Class                 100.00000  0.35759898   0.28
## aux_mileage=f.mileage-(6e+03,1.65e+04]   80.28846 25.59386973  24.96
## model=Audi-RS4                            0.00000  0.00000000   0.04
## model=Audi-R8                             0.00000  0.00000000   0.04
## aux_years_sell=f.years_sell-(6,23]       75.11905 16.11749681  16.80
## year=f.Year-2015                         73.50835  7.86717752   8.38
## model=Audi-RS6                            0.00000  0.00000000   0.06
## model=Audi-RS5                            0.00000  0.00000000   0.06
## aux_mileage=f.mileage-(3.35e+04,1.16e+05]  75.36000 24.06130268  25.00
## mout=YesMOut                             72.84483  8.63346105   9.28
## model=Audi-S4                             0.00000  0.00000000   0.08
## engineSize=f.EngSize-2.5                 30.00000  0.07662835   0.20
## transmission=f.Trans-Manual              75.82173 34.76372925  35.90
## engineSize=f.EngSize-4                   50.00000  0.35759898   0.56
## aux_price=f.price-(26,110]               74.96000 23.93358876  25.00
## model=Audi-SQ5                            0.00000  0.00000000   0.10
## model=Audi-RS3                            0.00000  0.00000000   0.10
## fuelType=f.Fuel-Petrol                   75.65758 40.40868455  41.82
## model=Audi-Q8                             0.00000  0.00000000   0.16
## engineSize=f.EngSize-2                   74.36267 39.48914432  41.58
## model=Audi-A7                             0.00000  0.00000000   0.28
## model=Audi-A8                             0.00000  0.00000000   0.32
## aux_mpg=f.mpg-[8.8,44.8]                 70.53846 23.42273308  26.00
## model=Audi-TT                             0.00000  0.00000000   0.62
## model=Audi-Q7                             0.00000  0.00000000   0.64
## engineSize=f.EngSize-1.4                 47.40484  3.49936143   5.78
## model=Audi-A6                             0.00000  0.00000000   1.42
## model=Audi-Q2                             0.00000  0.00000000   1.60
## model=Audi-Q5                             0.00000  0.00000000   1.86
## model=Audi-A5                             0.00000  0.00000000   2.02
## model=Audi-A4                             0.00000  0.00000000   2.72
## model=Audi-A1                             0.00000  0.00000000   2.74
## model=Audi-Q3                             0.00000  0.00000000   2.84
## model=Audi-A3                             0.00000  0.00000000   3.98
## manufacturer=Audi                         0.00000  0.00000000  21.70
##                                              p.value     v.test
## manufacturer=VW                         6.200610e-198  30.015283
```

```
## manufacturer=Mercedes                     4.216696e-172  27.965963
## manufacturer=BMW                          5.589380e-128  24.066644
## model=VW-Golf                              1.159189e-55  15.716865
## engineSize=f.EngSize-2.1                   4.367102e-46  14.251826
## model=Mercedes-C Class                     1.985400e-41  13.482401
## model=VW-Polo                              3.119709e-39  13.104093
## model=Mercedes-A Class                     2.535101e-30  11.443480
## model=BMW-3 Series                         2.812726e-27  10.818523
## model=BMW-1 Series                         3.908271e-22   9.673488
## model=Mercedes-E Class                     3.008936e-20   9.218639
## model=VW-Tiguan                            1.765897e-18   8.771316
## model=BMW-2 Series                         1.278514e-15   7.996649
## model=Mercedes-GLC Class                   1.532980e-13   7.384265
## model=BMW-4 Series                         2.968357e-11   6.648132
## model=VW-Passat                            8.069916e-11   6.499291
## model=VW-Up                                3.611143e-10   6.269971
## model=BMW-5 Series                         5.947978e-10   6.191803
## model=Mercedes-GLA Class                   9.794760e-10   6.112719
## engineSize=f.EngSize-1.3                   7.186080e-09   5.786517
## model=BMW-X1                               1.181999e-08   5.702302
## engineSize=f.EngSize-1.2                   2.000762e-08   5.611935
## engineSize=f.EngSize-1.5                   7.077720e-08   5.389187
## model=VW-T-Roc                             1.817722e-07   5.217073
## model=Mercedes-B Class                     1.817722e-07   5.217073
## aux_price=f.price-[0.65,14]                3.496523e-07   5.094506
## model=Mercedes-CL Class                    8.047644e-07   4.934208
## model=Mercedes-GLE Class                   1.320844e-06   4.836592
## aux_mpg=f.mpg-(61.4,88.3]                  1.808258e-06   4.773751
## model=BMW-X3                               7.468166e-06   4.479890
## aux_mpg=f.mpg-(53.3,61.4]                  2.176387e-05   4.245981
## model=VW-Touran                            6.898687e-05   3.979753
## model=Mercedes-SL CLASS                    8.829322e-05   3.920697
## fuelType=f.Fuel-Hybrid                     1.981835e-04   3.721321
## model=BMW-X5                               3.029459e-04   3.612768
## model=VW-Touareg                           3.875960e-04   3.548389
## engineSize=f.EngSize-1                     1.055041e-03   3.275422
## model=VW-Arteon                            1.327652e-03   3.209935
## model=VW-T-Cross                           2.171678e-03   3.065691
## model=VW-Golf SV                           2.171678e-03   3.065691
## model=VW-Scirocco                          2.777248e-03   2.991374
## fuelType=f.Fuel-Diesel                     3.165296e-03   2.951212
## mout=NoMOut                                3.450461e-03   2.924468
## model=VW-Sharan                            4.541297e-03   2.837889
## model=BMW-X2                               4.541297e-03   2.837889
## aux_tax=f.tax-[0,125]                      9.288054e-03   2.601265
## transmission=f.Trans-SemiAuto              9.456906e-03   2.595079
## model=Mercedes-V Class                     1.213442e-02   2.508211
## model=Mercedes-CLS Class                   1.213442e-02   2.508211
## model=BMW-X4                               1.213442e-02   2.508211
## model=VW-Amarok                            1.982863e-02   2.329575
## model=Mercedes-S Class                     1.982863e-02   2.329575
## model=BMW-Z4                               2.534506e-02   2.236102
## model=VW-Tiguan Allspace                   3.239438e-02   2.139510
## model=Mercedes-GLS Class                   3.239438e-02   2.139510
## model=Mercedes-GL Class                    3.239438e-02   2.139510
## aux_mileage=f.mileage-(6e+03,1.65e+04]     4.802947e-02   1.977108
## model=Audi-RS4                             4.705501e-02  -1.985805
## model=Audi-R8                              4.705501e-02  -1.985805
## aux_years_sell=f.years_sell-(6,23]         1.539365e-02  -2.422983
## year=f.Year-2015                           1.482733e-02  -2.436569
## model=Audi-RS6                             1.019619e-02  -2.569104
## model=Audi-RS5                             1.019619e-02  -2.569104
## aux_mileage=f.mileage-(3.35e+04,1.16e+05]  3.928202e-03  -2.883871
## mout=YesMOut                               3.450461e-03  -2.924468
```

```
## model=Audi-S4                            2.207781e-03  -3.060757
## engineSize=f.EngSize-2.5                 1.574193e-03  -3.160647
## transmission=f.Trans-Manual              1.556181e-03  -3.163998
## engineSize=f.EngSize-4                   1.073648e-03  -3.270481
## aux_price=f.price-(26,110]               1.069335e-03  -3.271619
## model=Audi-SQ5                           4.777044e-04  -3.492956
## model=Audi-RS3                           4.777044e-04  -3.492956
## fuelType=f.Fuel-Petrol                   1.287964e-04  -3.828749
## model=Audi-Q8                            4.818120e-06  -4.572555
## engineSize=f.EngSize-2                   1.435083e-08  -5.669149
## model=Audi-A7                            4.805857e-10  -6.225316
## model=Audi-A8                            2.215780e-11  -6.691048
## aux_mpg=f.mpg-[8.8,44.8]                 1.250134e-14  -7.710818
## model=Audi-TT                            1.917911e-21  -9.509386
## model=Audi-Q7                            4.068178e-22  -9.669384
## engineSize=f.EngSize-1.4                 1.255775e-32 -11.895050
## model=Audi-A6                            1.226198e-48 -14.656391
## model=Audi-Q2                            7.866556e-55 -15.595047
## model=Audi-Q5                            7.969683e-64 -16.866251
## model=Audi-A5                            2.173682e-69 -17.607081
## model=Audi-A4                            5.313784e-94 -20.567976
## model=Audi-A1                            1.036756e-94 -20.647092
## model=Audi-Q3                            2.894022e-98 -21.038835
## model=Audi-A3                           1.877840e-139 -25.138712
## manufacturer=Audi                        0.000000e+00       -Inf
##
## $`Audi Yes`
##                                           Cla/Mod     Mod/Cla Global
## manufacturer=Audi                       100.000000 100.0000000  21.70
## model=Audi-A3                           100.000000  18.3410138   3.98
## model=Audi-Q3                           100.000000  13.0875576   2.84
## model=Audi-A1                           100.000000  12.6267281   2.74
## model=Audi-A4                           100.000000  12.5345622   2.72
## model=Audi-A5                           100.000000   9.3087558   2.02
## model=Audi-Q5                           100.000000   8.5714286   1.86
## model=Audi-Q2                           100.000000   7.3732719   1.60
## model=Audi-A6                           100.000000   6.5437788   1.42
## engineSize=f.EngSize-1.4                 52.595156  14.0092166   5.78
## model=Audi-Q7                           100.000000   2.9493088   0.64
## model=Audi-TT                           100.000000   2.8571429   0.62
## aux_mpg=f.mpg-[8.8,44.8]                 29.461538  35.2995392  26.00
## model=Audi-A8                           100.000000   1.4746544   0.32
## model=Audi-A7                           100.000000   1.2903226   0.28
## engineSize=f.EngSize-2                   25.637326  49.1244240  41.58
## model=Audi-Q8                           100.000000   0.7373272   0.16
## fuelType=f.Fuel-Petrol                   24.342420  46.9124424  41.82
## model=Audi-SQ5                          100.000000   0.4608295   0.10
## model=Audi-RS3                          100.000000   0.4608295   0.10
## aux_price=f.price-(26,110]               25.040000  28.8479263  25.00
## engineSize=f.EngSize-4                   50.000000   1.2903226   0.56
## transmission=f.Trans-Manual              24.178273  40.0000000  35.90
## engineSize=f.EngSize-2.5                 70.000000   0.6451613   0.20
## model=Audi-S4                           100.000000   0.3686636   0.08
## mout=YesMOut                             27.155172  11.6129032   9.28
## aux_mileage=f.mileage-(3.35e+04,1.16e+05] 24.640000  28.3870968  25.00
## model=Audi-RS6                          100.000000   0.2764977   0.06
## model=Audi-RS5                          100.000000   0.2764977   0.06
## year=f.Year-2015                         26.491647  10.2304147   8.38
## aux_years_sell=f.years_sell-(6,23]       24.880952  19.2626728  16.80
## model=Audi-RS4                          100.000000   0.1843318   0.04
## model=Audi-R8                           100.000000   0.1843318   0.04
## aux_mileage=f.mileage-(6e+03,1.65e+04]   19.711538  22.6728111  24.96
## model=VW-Tiguan Allspace                  0.000000   0.0000000   0.28
## model=Mercedes-GLS Class                  0.000000   0.0000000   0.28
```

```
## model=Mercedes-GL Class                      0.000000   0.0000000   0.28
## model=BMW-Z4                                 0.000000   0.0000000   0.30
## model=VW-Amarok                              0.000000   0.0000000   0.32
## model=Mercedes-S Class                       0.000000   0.0000000   0.32
## model=Mercedes-V Class                       0.000000   0.0000000   0.36
## model=Mercedes-CLS Class                     0.000000   0.0000000   0.36
## model=BMW-X4                                 0.000000   0.0000000   0.36
## transmission=f.Trans-SemiAuto               19.767442  34.4700461  37.84
## aux_tax=f.tax-[0,125]                       15.824916   4.3317972   5.94
## model=VW-Sharan                              0.000000   0.0000000   0.44
## model=BMW-X2                                 0.000000   0.0000000   0.44
## mout=NoMOut                                 21.141975  88.3870968  90.72
## fuelType=f.Fuel-Diesel                      20.197044  52.9032258  56.84
## model=VW-Scirocco                            0.000000   0.0000000   0.48
## model=VW-T-Cross                             0.000000   0.0000000   0.50
## model=VW-Golf SV                             0.000000   0.0000000   0.50
## model=VW-Arteon                              0.000000   0.0000000   0.54
## engineSize=f.EngSize-1                      15.479115   5.8064516   8.14
## model=VW-Touareg                             0.000000   0.0000000   0.64
## model=BMW-X5                                 0.000000   0.0000000   0.66
## fuelType=f.Fuel-Hybrid                       3.636364   0.1843318   1.10
## model=Mercedes-SL CLASS                      0.000000   0.0000000   0.76
## model=VW-Touran                              0.000000   0.0000000   0.78
## aux_mpg=f.mpg-(53.3,61.4]                   17.340067  18.9861751  23.76
## model=BMW-X3                                 0.000000   0.0000000   0.96
## aux_mpg=f.mpg-(61.4,88.3]                   16.666667  17.4193548  22.68
## model=Mercedes-GLE Class                     0.000000   0.0000000   1.10
## model=Mercedes-CL Class                      0.000000   0.0000000   1.14
## aux_price=f.price-[0.65,14]                 16.679905  19.3548387  25.18
## model=VW-T-Roc                               0.000000   0.0000000   1.26
## model=Mercedes-B Class                       0.000000   0.0000000   1.26
## engineSize=f.EngSize-1.5                    13.200723   6.7281106  11.06
## engineSize=f.EngSize-1.2                     4.000000   0.4608295   2.50
## model=BMW-X1                                 0.000000   0.0000000   1.48
## engineSize=f.EngSize-1.3                     0.000000   0.0000000   1.52
## model=Mercedes-GLA Class                     0.000000   0.0000000   1.68
## model=BMW-5 Series                           0.000000   0.0000000   1.72
## model=VW-Up                                  0.000000   0.0000000   1.76
## model=VW-Passat                              0.000000   0.0000000   1.88
## model=BMW-4 Series                           0.000000   0.0000000   1.96
## model=Mercedes-GLC Class                     0.000000   0.0000000   2.38
## model=BMW-2 Series                           0.000000   0.0000000   2.76
## model=VW-Tiguan                              0.000000   0.0000000   3.28
## model=Mercedes-E Class                       0.000000   0.0000000   3.60
## model=BMW-1 Series                           0.000000   0.0000000   3.94
## model=BMW-3 Series                           0.000000   0.0000000   4.86
## model=Mercedes-A Class                       0.000000   0.0000000   5.40
## model=VW-Polo                                0.000000   0.0000000   6.96
## model=Mercedes-C Class                       0.000000   0.0000000   7.34
## engineSize=f.EngSize-2.1                     0.000000   0.0000000   8.14
## model=VW-Golf                                0.000000   0.0000000   9.76
## manufacturer=BMW                             0.000000   0.0000000  20.98
## manufacturer=Mercedes                        0.000000   0.0000000  27.02
## manufacturer=VW                              0.000000   0.0000000  30.30
##                                                  p.value     v.test
## manufacturer=Audi                           0.000000e+00        Inf
## model=Audi-A3                              1.877840e-139  25.138712
## model=Audi-Q3                               2.894022e-98  21.038835
## model=Audi-A1                               1.036756e-94  20.647092
## model=Audi-A4                               5.313784e-94  20.567976
## model=Audi-A5                               2.173682e-69  17.607081
## model=Audi-Q5                               7.969683e-64  16.866251
## model=Audi-Q2                               7.866556e-55  15.595047
## model=Audi-A6                               1.226198e-48  14.656391
```

```
## engineSize=f.EngSize-1.4                     1.255775e-32  11.895050
## model=Audi-Q7                                4.068178e-22   9.669384
## model=Audi-TT                                1.917911e-21   9.509386
## aux_mpg=f.mpg-[8.8,44.8]                     1.250134e-14   7.710818
## model=Audi-A8                                2.215780e-11   6.691048
## model=Audi-A7                                4.805857e-10   6.225316
## engineSize=f.EngSize-2                       1.435083e-08   5.669149
## model=Audi-Q8                                4.818120e-06   4.572555
## fuelType=f.Fuel-Petrol                       1.287964e-04   3.828749
## model=Audi-SQ5                               4.777044e-04   3.492956
## model=Audi-RS3                               4.777044e-04   3.492956
## aux_price=f.price-(26,110]                   1.069335e-03   3.271619
## engineSize=f.EngSize-4                       1.073648e-03   3.270481
## transmission=f.Trans-Manual                  1.556181e-03   3.163998
## engineSize=f.EngSize-2.5                     1.574193e-03   3.160647
## model=Audi-S4                                2.207781e-03   3.060757
## mout=YesMOut                                 3.450461e-03   2.924468
## aux_mileage=f.mileage-(3.35e+04,1.16e+05]    3.928202e-03   2.883871
## model=Audi-RS6                               1.019619e-02   2.569104
## model=Audi-RS5                               1.019619e-02   2.569104
## year=f.Year-2015                             1.482733e-02   2.436569
## aux_years_sell=f.years_sell-(6,23]           1.539365e-02   2.422983
## model=Audi-RS4                               4.705501e-02   1.985805
## model=Audi-R8                                4.705501e-02   1.985805
## aux_mileage=f.mileage-(6e+03,1.65e+04]       4.802947e-02  -1.977108
## model=VW-Tiguan Allspace                     3.239438e-02  -2.139510
## model=Mercedes-GLS Class                     3.239438e-02  -2.139510
## model=Mercedes-GL Class                      3.239438e-02  -2.139510
## model=BMW-Z4                                 2.534506e-02  -2.236102
## model=VW-Amarok                              1.982863e-02  -2.329575
## model=Mercedes-S Class                       1.982863e-02  -2.329575
## model=Mercedes-V Class                       1.213442e-02  -2.508211
## model=Mercedes-CLS Class                     1.213442e-02  -2.508211
## model=BMW-X4                                 1.213442e-02  -2.508211
## transmission=f.Trans-SemiAuto                9.456906e-03  -2.595079
## aux_tax=f.tax-[0,125]                        9.288054e-03  -2.601265
## model=VW-Sharan                              4.541297e-03  -2.837889
## model=BMW-X2                                 4.541297e-03  -2.837889
## mout=NoMOut                                  3.450461e-03  -2.924468
## fuelType=f.Fuel-Diesel                       3.165296e-03  -2.951212
## model=VW-Scirocco                            2.777248e-03  -2.991374
## model=VW-T-Cross                             2.171678e-03  -3.065691
## model=VW-Golf SV                             2.171678e-03  -3.065691
## model=VW-Arteon                              1.327652e-03  -3.209935
## engineSize=f.EngSize-1                       1.055041e-03  -3.275422
## model=VW-Touareg                             3.875960e-04  -3.548389
## model=BMW-X5                                 3.029459e-04  -3.612768
## fuelType=f.Fuel-Hybrid                       1.981835e-04  -3.721321
## model=Mercedes-SL CLASS                      8.829322e-05  -3.920697
## model=VW-Touran                              6.898687e-05  -3.979753
## aux_mpg=f.mpg-(53.3,61.4]                    2.176387e-05  -4.245981
## model=BMW-X3                                 7.468166e-06  -4.479890
## aux_mpg=f.mpg-(61.4,88.3]                    1.808258e-06  -4.773751
## model=Mercedes-GLE Class                     1.320844e-06  -4.836592
## model=Mercedes-CL Class                      8.047644e-07  -4.934208
## aux_price=f.price-[0.65,14]                  3.496523e-07  -5.094506
## model=VW-T-Roc                               1.817722e-07  -5.217073
## model=Mercedes-B Class                       1.817722e-07  -5.217073
## engineSize=f.EngSize-1.5                     7.077720e-08  -5.389187
## engineSize=f.EngSize-1.2                     2.000762e-08  -5.611935
## model=BMW-X1                                 1.181999e-08  -5.702302
## engineSize=f.EngSize-1.3                     7.186080e-09  -5.786517
## model=Mercedes-GLA Class                     9.794760e-10  -6.112719
## model=BMW-5 Series                           5.947978e-10  -6.191803
```

```
## model=VW-Up                                          3.611143e-10  -6.269971
## model=VW-Passat                                      8.069916e-11  -6.499291
## model=BMW-4 Series                                   2.968357e-11  -6.648132
## model=Mercedes-GLC Class                             1.532980e-13  -7.384265
## model=BMW-2 Series                                   1.278514e-15  -7.996649
## model=VW-Tiguan                                      1.765897e-18  -8.771316
## model=Mercedes-E Class                               3.008936e-20  -9.218639
## model=BMW-1 Series                                   3.908271e-22  -9.673488
## model=BMW-3 Series                                   2.812726e-27 -10.818523
## model=Mercedes-A Class                               2.535101e-30 -11.443480
## model=VW-Polo                                        3.119709e-39 -13.104093
## model=Mercedes-C Class                               1.985400e-41 -13.482401
## engineSize=f.EngSize-2.1                             4.367102e-46 -14.251826
## model=VW-Golf                                        1.159189e-55 -15.716865
## manufacturer=BMW                                    5.589380e-128 -24.066644
## manufacturer=Mercedes                               4.216696e-172 -27.965963
## manufacturer=VW                                     6.200610e-198 -30.015283
```

As we can see in the output catdes$quanti.var, we can feature that µ_AudiNo, µ_AudiYes and µ_mpg are not the same because the p-value is less than 0.05.

As we can see in the output catdes$quanti, we can feature that Audi cars on average consume more, because on average they have less mpg.

As we can see in the output catdes$test.chi2, we can feature that Audi variable and model variable are not independent because the p-value is less than 0.05. Also, Audi variable and engineSize variable are not independent for the same reason.

As we can see in the output catdes$category, we can feature that VW, Mercedes and BMW cars are overrepresented in Audi No sample. Also, we can feature that engineSize 1.4, 2, 4 and 2.5 are overrepresented in Audi Yes sample and engineSize 1, 1.5, 1.2, 1.3, 2.1 are underrepresented in Audi Yes sample.