Assignment 1

Diana Rocio Galindo Gonzalez

Introduction (from *Data_loadingintroduction* document)

This data dictionary describes data (https://www.kaggle.com/adityadesai13/used-car-dataset-ford-and-mercedes) - A sample of 5000 trips has been randomly selected from Mercedes, BMW, Volkwagen and Audi manufacturers. So, firstly you have to combine used car from the 4 manufacturers into 1 dataframe.

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 Factor: Audi, BMW, Mercedes or Volkswagen
- model Car model
- year registration year
- price price in £
- transmission type of gearbox
- mileage distance used
- fuelType engine fuel
- tax road tax
- mpg Consumption in miles per gallon
- engineSize size in litres

```r
knitr::opts_chunk$set(echo = TRUE)

# Required packages
pkgs<-c("car","chemometrics","corrplot","corrplot","dplyr","data.table","fitdistrplus",
        "dygraphs","DT","factoextra","FactoMineR","ggcorrplot","ggplot2","lmtest","GGally",
        "ggspatial","googleway","grid","gridExtra","heatmaply","htmlwidgets",
        "knitr", "lattice","leaflet","lubridate","magrittr","missMDA","naniar",
        "plotly","rnaturalearth","rnaturalearthdata","rstudioapi","sf","sm",
        "mice","tidyr","tidyverse","VIM","visdat","xtable")

# Non-installed packages
inspkgs<-pkgs[!pkgs %in% installed.packages()]
for(libs in inspkgs) install.packages(libs,
                                      repos = "http://cran.us.r-project.org")

# Loading required
sapply(pkgs,require,character=TRUE)

# Loading files
current_path <- getActiveDocumentContext()$path
current_path
setwd(dirname(current_path ))
getwd()
```

Data loading and data cleaning

The first step to develop this assignment, as it was explained in the *Data_loadingintroduction* document is to load the data, unification of data frames and subsetting the random sample as input to the data cleaning procedure.

```
rm(list=ls())
set.seed(310883)

# Import initial dataset
csvf<-list.files(path=".",pattern=".csv")
dfs<-lapply(csvf, read.delim,stringsAsFactors = TRUE,header=T, sep=",")

# Data subsetting by manufacturer
names(dfs)<-c("Audi","BMW","MERCEDES","VM")
df0<-dplyr::bind_rows(dfs, .id = "manufacturer")

# Random selection of x registers:
sam<-as.vector(sort(sample(1:nrow(df0),1000)))
#head(df0)
df1<-df0[sam,] # Subset of rows _ It will be my sample

# Converting char variables as factors
df1[sapply(df1, is.character)] <- lapply(df1[sapply(df1, is.character)],
                                          as.factor)

# Data frame structure
data.frame(Variable = names(df1),
 Class = sapply(df1, class),
 Head = sapply(df1, function(x) paste0(head(x), collapse = ", ")),
 row.names = NULL) %>% kable()
```

Variable	Class	Head
manufacturer	factor	Audi, Audi, Audi, Audi, Audi, Audi
model	factor	A4, A4, A4, Q3, A6, A4
year	integer	2017, 2019, 2019, 2014, 2013, 2019
price	integer	16000, 26985, 30995, 14500, 12495, 28485
transmission	factor	Automatic, Semi-Auto, Automatic, Semi-Auto, Manual, Automatic
mileage	integer	58028, 476, 10, 35423, 49000, 2250
fuelType	factor	Petrol, Petrol, Petrol, Diesel, Diesel, Diesel
tax	integer	145, 145, 145, 200, 325, 145
mpg	numeric	52.3, 39.2, 38.7, 47.9, 29.7, 51.4
engineSize	numeric	2, 2, 2, 2, 2, 2

According to the instructions, it is necessary to clean the data to perform an optimal analysis. The cleaning process includes: remove duplicate data, validation of consistency and fix structural errors. Regarding duplicated data, the 743 duplicated rows present in the whole dataset were removed. For consistency, the chunk of code shows a verification of the levels and removing leading, and duplicated spaces.

```
# Checking and removing duplicates
kable(table(duplicated(df1)), col.names = c("Duplicated","Freq"))
```

Level
Audi
BMW
MERCEDES
VM

Level
Automatic
Manual
Semi-Auto
Other

Level
Diesel
Hybrid
Petrol
Electric
Other

Level	Level	Level	Level	Level	Level	Level
A1	RS4	5 Series	X5	GL Class	220	Golf SV
A2	RS5	6 Series	X6	GLA Class	230	Jetta
A3	RS6	7 Series	X7	GLB Class	Amarok	Passat
A4	RS7	8 Series	Z3	GLC Class	Arteon	Polo
A5	S3	i3	Z4	GLE Class	Beetle	Scirocco
A6	S4	i8	A Class	GLS Class	Caddy	Sharan
A7	S5	M2	B Class	M Class	Caddy Life	Shuttle
A8	S8	M3	C Class	R Class	Caddy Maxi	T-Cross
Q2	SQ5	M4	CL Class	S Class	Caddy Maxi Life	T-Roc
Q3	SQ7	M5	CLA Class	SL CLASS	California	Tiguan
Q5	TT	M6	CLC Class	SLK	Caravelle	Tiguan Allspace
Q7	1 Series	X1	CLK	V Class	CC	Touareg
Q8	2 Series	X2	CLS Class	X-CLASS	Eos	Touran
R8	3 Series	X3	E Class	180	Fox	Up
RS3	4 Series	X4	G Class	200	Golf	

Duplicated	Freq
FALSE	998
TRUE	2

```
df2<-df1[!duplicated(df1), ]
#kable(table(duplicated(df1)), col.names = c("Duplicated","Freq"))

# Checking levels of factor variables
kable(sapply(df2[,c(1,5,7)], levels), col.names = c("Level"))
```

```
mod_lev<-levels(df2$model)
vis<-split(mod_lev, ceiling(seq_along(mod_lev)/15))
kable(vis,col.names = c("Level"))
```

```
# Removing leading, trailing, multiple spaces on levels
for (i in c(1,2,5,7)){ df2[,i] <-gsub(" +$", " ", df2[,i]) }
for (i in c(1,2,5,7)){ df2[,i] <-trimws(df2[,i])}
#sapply(df1[,c(1,2,5,7)], table)
```

Data preparation

To define missing data related with electric cars and engine 0, in this report are considered three assumptions based on the data available:

- Electric car has no *transmission* and *engine size* is equal or less than 0.6. *Fuel* can be hybrid
- Based on the previous condition, electric model references were taken from in the website wattev2buy. According to this specialized web portal, the models related to the manufacturers of the analysis are:

According to the information provided by the portal, potencial electric cars per manufacturer in the data set would be:

-Audi: A3, A6, A7, A8, Q2, Q3, Q5, Q7, Q8, R8 -BMW:7 Series, i3, i8, X1, X2, X3, X5 -Mercedes: GLC, C Class, C300e, CLA, GLE350de -VW: Arteon, Passat, Tiguan, Touareg, Touran

According to the assumptions, cars with engine lower or equal to 0.6, automatic transmission, with *hybrid* or *"other"* type of fuel and intersecting with the list from the portal could be recategorized as *"fuelType"* electric.

The price was converted to miles of £ and the mileage to miles to facilite the results interpretation.

```
elecweb<-c("A3","A6","A7","A8","Q2","Q3","Q5", "Q7","Q8","R8","7 Series","i3",
           "i8","X1","X2","X3","X5","GLC","C Class","C300e","CLA","GLE350de",
           "Arteon","Passat","Tiguan","Touareg","Touran")

#table(df0$fuel,df0$engineSize)
#table(df0$manufacturer,df0$model)
#table(df0$model)
#df1[df1$engineSize <= 0.6 & df1$transmission != , ]
#df1[df1$model %in% elecweb, ]

df3<-df2
df3$fuelType<-ifelse((df3$engineSize <= 0.6 & (df3$fuelType == "Hybrid"|
                                                 df2$fuelType == "Other") &
                        df3$model %in% elecweb), "Electric", df3$fuelType)


df3$price<-df3$price*0.001
df3$mileage<-df3$mileage*0.001
df<-df3
df[sapply(df, is.character)] <- lapply(df[sapply(df, is.character)], as.factor)

df<-df[,c("price","year","mileage","tax","mpg","engineSize",
          "manufacturer","model","transmission","fuelType")]

numy_var<-select_if(df, is.numeric)
caty_var<-select_if(df, is.factor)

#Keep information in an .Rdata file:
save(list=c("df"),file="MyOldCars-RawDiana.RData")
```

Exploratory analysis

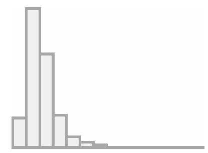
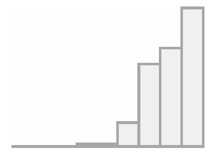
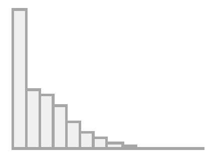
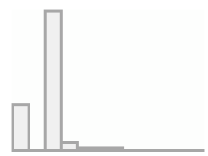
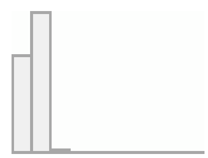
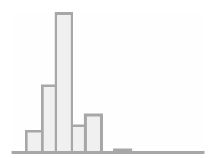
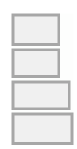
To have a better understanting of data behavior, some univariate exploration tools were used. A summary of selected dataset is displayed:

```
define_keywords(title.dfSummary = "Data Frame Summary in PDF Format")
dfSummary(df)
```

Type of model (wattev2buy.com)	model (wattev2buy)	Closest model in dataset
Audi Plug-in Hybrid Electric Models	A3 ‘Sportback 40 TFSI e	A3
	A3 Sportback 30 g-tron	A3
	A6 55 TFSI e	A6
	A7 Sportback	A7
	A8 60 TFSI e	A8
	Q3 TFSI E	Q3
	Q5 55 TFSI e	Q5
	Q5 55 TFSI e	Q5
	Q7	Q7
	Q8 55 TFSI e	Q8
Audi Pure Electric Models	Audi R8 e-tron	R8
	Q2 L 30 e-tron	Q2
BMW Plug-in Hybrid Electric Models	7 Series	7 Series
	i3 REx	i3
	i8	i8
	X1 xDrive25e	X1
	X2 xDrive25e	X2
	X3 xDrive30e PHEV	X3
	X5 xDrive45e	X5
BMW Pure Electric Models	BMW iX	iX
	i3 120AH	i3
	i4	i4
	iX3	iX3
Mercedes Fuel Cell Electric Models	GLC F CELL	GLC
Mercedes Plug-in Hybrid Electric Models	A250e 4Matic	A250e
	A250e L 4Matic	A250e
	C Class PHEV	C Class
	C300e Estate	C300e
	CLA250 Coupe	CLA
	CLA250 Shootingbrake	CLA
	GLA 250e SUV	CLA
	GLC 300e 4MATIC	GLC
	GLC 300e 4MATIC Coupé	GLC
	GLE350de 4MATIC	GLE350de
Mercedes Pure Electric Models	B250e ED	B250e
VW Plug-in Hybrid Electric Models	Arteon eHybrid	Arteon
	Arteon Estate eHybrid	Arteon
	GTE	GTE
	Passat GTE	Passat
	Passat GTE Estate	Passat
	Tiguan eHybrid PHEV	Tiguan
	Touareg R	Touareg
	Touran	Touran
VW Pure Electric Models	e-Golf	e-Golf
	ID 3 1ST	ID

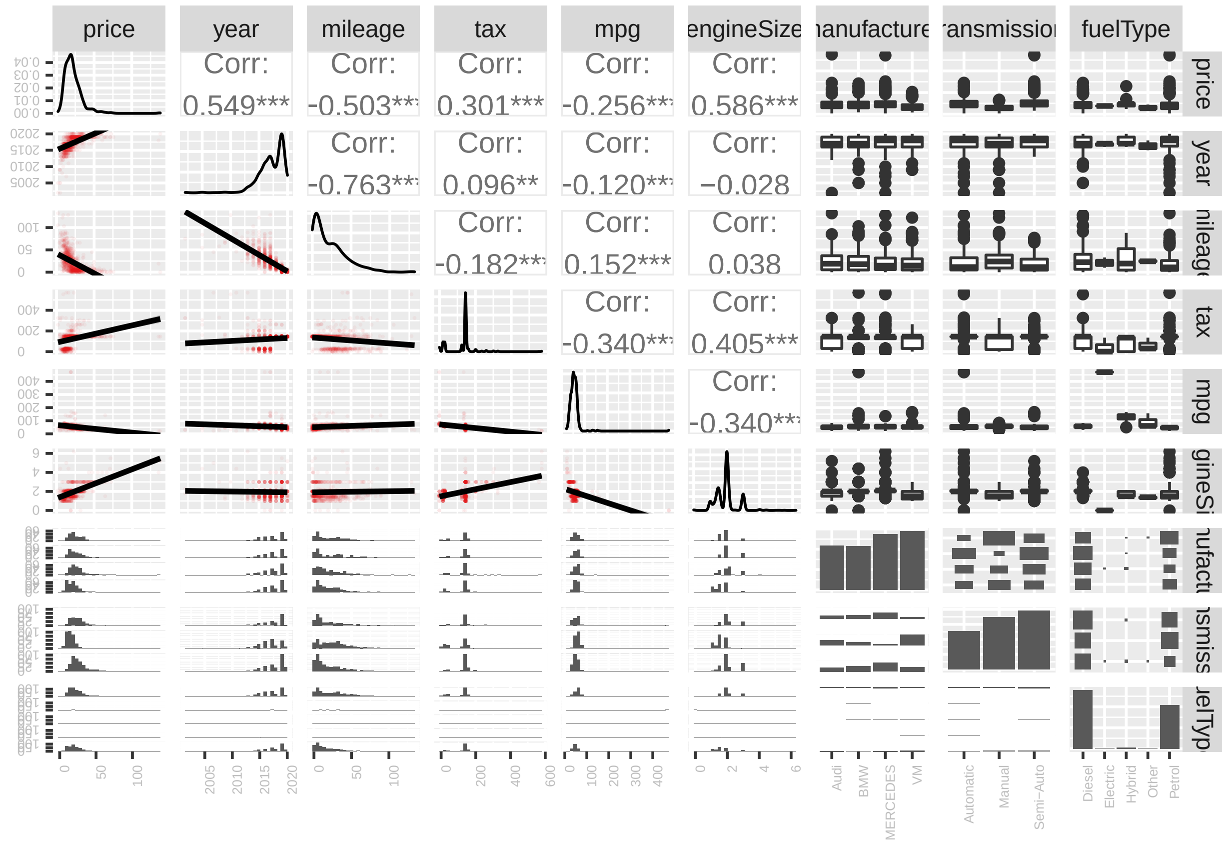
Data Frame Summary in PDF Format

df
Dimensions: 998 x 10
Duplicates: 0

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	price [numeric]	Mean (sd) : 22 (12) min < med < max: 2 < 19.9 < 138 IQR (CV) : 12.7 (0.5)	740 distinct values		0 (0.0%)
2	year [integer]	Mean (sd) : 2017.2 (2.2) min < med < max: 2002 < 2017 < 2020 IQR (CV) : 3 (0)	16 distinct values		0 (0.0%)
3	mileage [numeric]	Mean (sd) : 22.4 (21.3) min < med < max: 0 < 16.2 < 131.9 IQR (CV) : 28.6 (1)	890 distinct values		0 (0.0%)
4	tax [integer]	Mean (sd) : 123.8 (66.4) min < med < max: 0 < 145 < 570 IQR (CV) : 20 (0.5)	23 distinct values		0 (0.0%)
5	mpg [numeric]	Mean (sd) : 54.7 (27.4) min < med < max: 21.1 < 53.3 < 470.8 IQR (CV) : 17.3 (0.5)	90 distinct values		0 (0.0%)
6	engineSize [numeric]	Mean (sd) : 1.9 (0.6) min < med < max: 0 < 2 < 6.2 IQR (CV) : 0.5 (0.3)	21 distinct values		0 (0.0%)
7	manufacturer [factor]	1. Audi 2. BMW 3. MERCEDES 4. VM	217 (21.7%) 216 (21.6%) 275 (27.6%) 290 (29.1%)		0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
8	model [factor]	1. 1 Series 2. 2 Series 3. 3 Series 4. 4 Series 5. 5 Series 6. 6 Series 7. 7 Series 8. 8 Series 9. A Class 10. A1 [64 others]	47 (4.7%) 24 (2.4%) 41 (4.1%) 20 (2.0%) 20 (2.0%) 3 (0.3%) 1 (0.1%) 2 (0.2%) 46 (4.6%) 32 (3.2%) 762 (76.4%)		0 (0.0%)
9	transmission [factor]	1. Automatic 2. Manual 3. Semi-Auto	255 (25.6%) 350 (35.1%) 393 (39.4%)		0 (0.0%)
10	fuelType [factor]	1. Diesel 2. Electric 3. Hybrid 4. Other 5. Petrol	562 (56.3%) 3 (0.3%) 12 (1.2%) 3 (0.3%) 418 (41.9%)		0 (0.0%)

```r
p <- ggpairs(df[,-8],
        lower = list(continuous = wrap("smooth",size=0.01,alpha = 0.05,
                                       col='#e31a1c')))

p + theme(axis.text.x = element_text(angle = 90, hjust = 1, size=5,color="gray"),
          axis.text.y = element_text(angle = 180, hjust = 1, size=5,color="gray"))
```



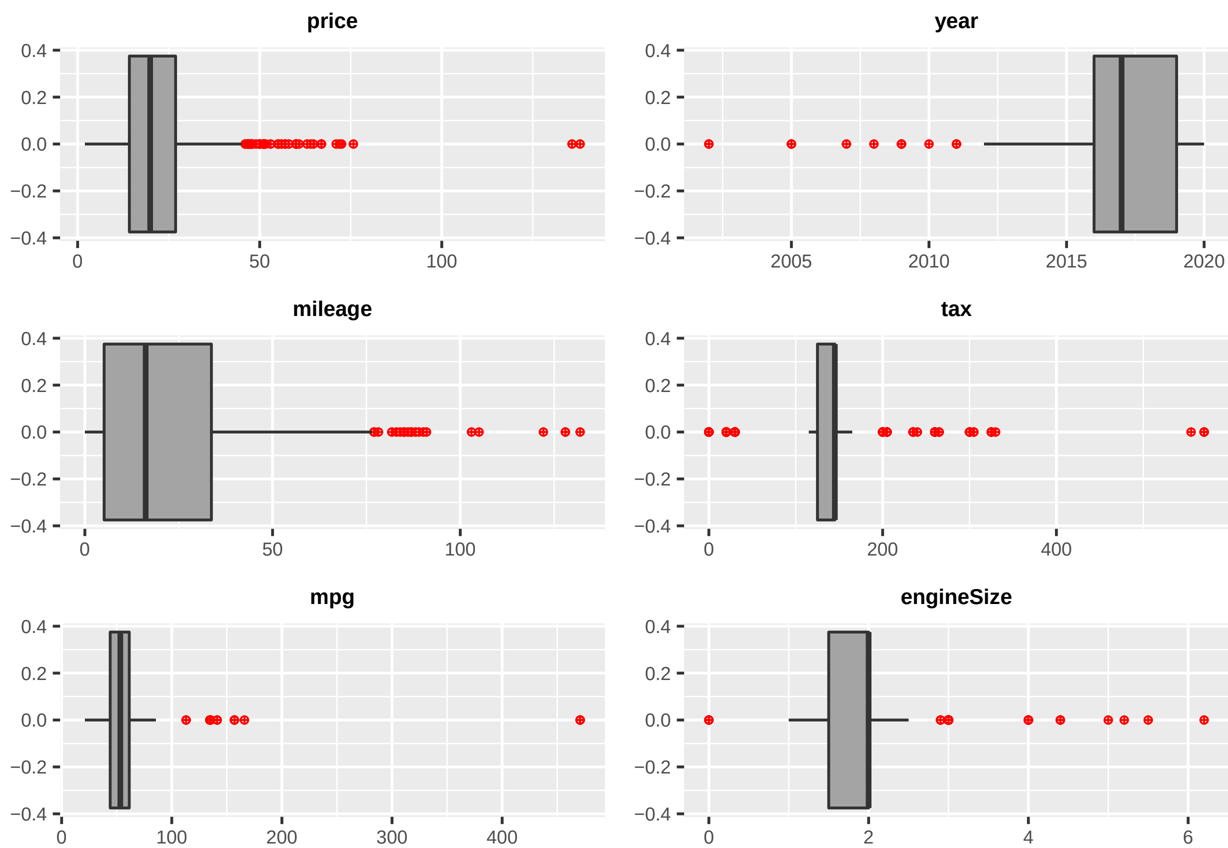
From this summary:

- In this particular dataset, there is an approximate range of price between £2.000 and £138.000. The mean price of a car is around £19.900 and this variable does not present a normal distribution, has a trend to lower values.
- The oldest car is from 2002 and the newest car is model 2020. The average is 2017. As well, most of the cars are recent from recent years.
- The mileage of the cars varies between 5 and 131925 miles. The average per car is 16.197 miles. Majority of cars have lower values of mileage.
- Tax figures are between 0 an 570, with a mean value o f 145. However, the most of cars have a road tax between 115 and 125.
- Regarding the miles per galon per car, the mean value is 53,3 values, but the most of values are below the mean.
- The majority of cars in data set are fueled by diesel.
- All the numeric variables including price, present outlier data.

Outliers

In accordance with the summary table, there is presence of univariate outlier values per column.

```
a<-names(numy_var)
a<-as.list(a)
fun02<-function(i){index=grep(i,names(numy_var))
                   nm=paste0(i)
                   assign(paste("g",i,sep=""),
                   ggplot(numy_var, aes(numy_var[,index])) +
                   geom_boxplot(fill='#A4A4A4', outlier.colour="red",
                                outlier.shape=10, outlier.size=1)+
                   labs(title=nm, x=NULL, y=NULL)) +
                   theme(plot.title = element_text(size = rel(0.7),face ="bold",
                                                   hjust = 0.5),
                        axis.title.y = element_text(size = rel(0.6)),
                        axis.text = element_text(size = rel(0.6)))
                   }
boxplots<-lapply(a,fun02)
do.call(grid.arrange, boxplots)
```




```
fun03<-function(x){out <- boxplot.stats(x)$out # identifying outlier values
                   out_ind <- which(x %in% c(out)) #identifying rows with them
                   print(out_ind)
                   print(length(out_ind))}
sapply(numy_var,fun03)
```

```
##  [1]   7  57  75 122 153 166 167 212 293 299 311 318 329 331 335 350 351 355 356
## [20] 441 492 499 504 522 523 532 549 560 563 566 567 587 593 597 602 603 608 611
```

```
## [39] 613 621 623 631 645 963
## [1] 44
##  [1] 208 404 405 406 419 647 682 684 691 696 722 814 887
## [1] 13
##  [1]  34 208 215 366 367 384 392 403 406 407 592 647 659 682 786 787 800 814 922
## [20] 953 998
## [1] 21
##   [1]   4   5  12  14  15  19  23  24  31  32  34  35  38  39  41  42  43  45
##  [19]  49  50  51  52  53  54  59  60  64  66  69  71  72  73  74  78  83  94
##  [37]  95 102 107 108 109 112 120 126 146 147 155 160 168 172 174 175 180 186
##  [55] 188 190 191 195 196 201 204 206 207 208 209 210 214 215 217 218 229 233
##  [73] 234 237 239 269 270 273 279 283 290 294 297 307 315 327 347 349 358 364
##  [91] 366 368 369 370 374 379 381 384 385 387 389 390 396 399 403 406 407 418
## [109] 419 420 421 424 427 428 429 430 433 435 439 442 446 451 458 461 462 463
## [127] 467 470 471 472 479 480 482 483 489 493 495 497 510 513 518 529 533 534
## [145] 537 542 543 547 553 559 564 569 570 577 584 592 601 625 629 640 646 647
## [163] 650 654 656 657 658 663 664 667 668 669 671 672 673 678 679 680 682 684
## [181] 685 687 689 691 692 694 696 698 703 706 721 724 729 730 735 737 739 740
## [199] 741 743 744 746 749 753 755 759 764 766 773 778 779 780 782 783 784 786
## [217] 787 789 790 794 796 800 802 803 806 808 811 814 828 829 831 832 833 836
## [235] 837 838 843 845 846 847 850 851 854 858 864 866 867 871 872 874 876 877
## [253] 879 881 888 889 890 891 892 898 917 932 936 940 942 944 945 946 948 949
## [271] 953 962 964 977 985 986 987 988 997 998
## [1] 280
##  [1] 231 241 311 317 327 386 387 402 403 592 698 767 801 806
## [1] 14
##   [1]   7   9  24  51  57  60  68  82 108 122 127 130 139 140 150 153 156 159
##  [19] 160 165 166 167 169 195 212 225 228 235 237 238 242 248 251 253 255 256
##  [37] 262 267 269 277 279 288 289 291 293 297 299 310 313 318 321 327 328 329
##  [55] 331 332 333 335 338 345 347 350 351 355 356 358 360 365 371 372 386 387
##  [73] 388 391 401 406 411 419 423 432 439 440 441 457 458 470 480 492 499 504
##  [91] 511 522 523 537 542 543 549 560 563 566 567 580 585 587 593 596 597 598
## [109] 601 602 608 610 611 613 621 623 627 628 631 637 645 647 654 657 665 684
## [127] 687 694 696 955 956 957 958 959 960 961 962 963 964 965 985 986 987 988
## [1] 144
```

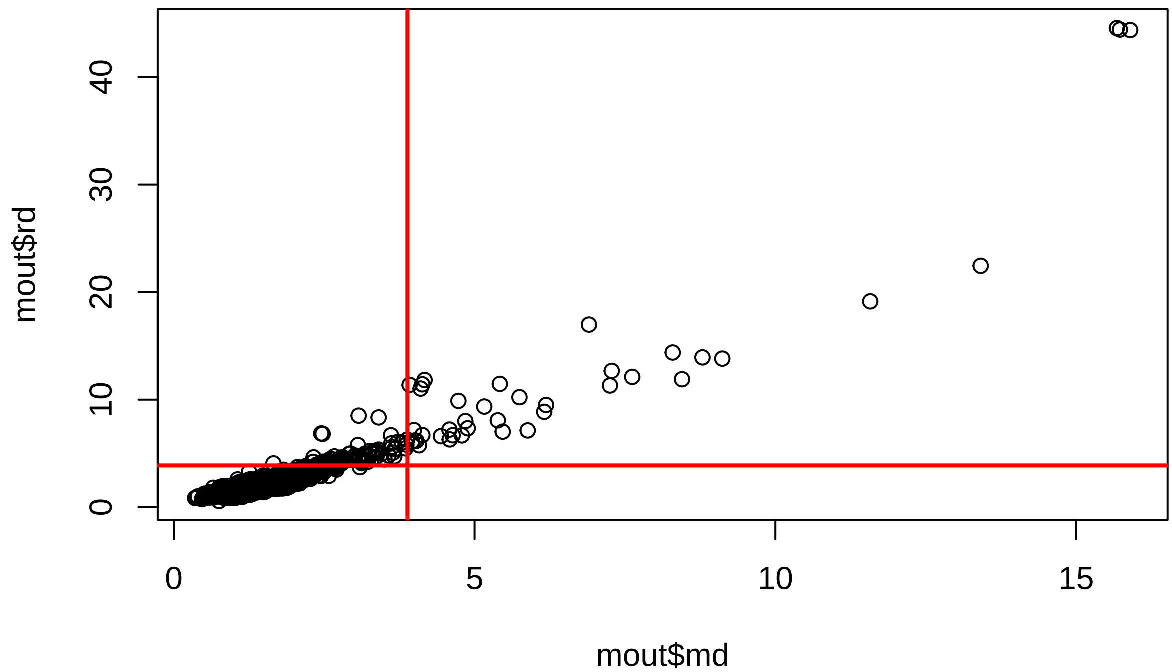
```
##      price       year    mileage        tax        mpg engineSize
##         44         13         21        280         14        144
```

Outliers are present in all the numerical variables as stated previously:

- High outliers values associated to variables: price (44 cars), mileage (21 cars) and mpg (14 cars).
- Low outliers values associated to variables: year (13 cars).
- Variables with both, low and high values: tax (280) and engine size (144).

To identify multivariate outliers was necessary to ommit variable tax. This variable is highly correlated positively with engineSize and negatively with mpg, and present high number of univariate outliers. As well in minimum correlated with price variable.

```
mout <- Moutlier( numy_var[,-c(4)], quantile = 0.99, plot=F )
par(mfrow=c(1,1))
plot( mout$md, mout$rd )
abline( h=mout$cutoff, lwd=2, col="red")
abline( v=mout$cutoff, lwd=2, col="red")
```




```r
llmout <- which((mout$md>mout$cutoff) & (mout$rd > mout$cutoff) )
llmout
```

```
##  2256  7643 10109 10340 11316 15053 16171 16257 17261 19044 19069 19346 19867
##    57   165   208   212   231   311   327   329   350   386   387   392   403
## 19945 19953 20060 20151 20679 27352 27668 28898 29631 29900 30291 31119 31612
##   404   405   406   407   419   549   560   592   608   613   621   647   659
## 32833 33000 33246 33447 33463 35344 38164 39193 40143 40317 44402 46158 47805
##   682   684   691   694   696   722   767   787   801   806   887   922   953
```

```r
kable(df[llmout,],table.attr = "style='width:30%;'")
```

	price	year	mileage	tax	mpg	engineSize	manufacturer	model	transmission	fuelType
2256	137.995	2020	0.070	145	21.1	5.2	Audi	R8	Semi-Auto	Petrol
7643	32.000	2019	4.000	145	31.4	0.0	Audi	Q3	Automatic	Petrol
10109	1.990	2002	131.925	325	30.1	1.8	Audi	TT	Manual	Petrol
10340	72.500	2020	0.010	150	32.8	3.0	Audi	Q8	Automatic	Diesel
11316	25.498	2017	20.279	135	156.9	2.0	BMW	5 Series	Semi-Auto	Hybrid
15053	64.750	2019	2.277	140	141.2	1.5	BMW	i8	Automatic	Hybrid
16171	18.995	2017	33.021	0	470.8	0.0	BMW	i3	Automatic	Electric
16257	66.991	2020	0.123	145	33.2	3.0	BMW	8 Series	Semi-Auto	Petrol
17261	70.995	2019	0.023	145	24.1	4.4	BMW	M5	Semi-Auto	Petrol
19044	18.999	2017	20.321	135	470.8	0.0	BMW	i3	Automatic	Electric
19069	18.999	2016	9.990	0	470.8	0.0	BMW	i3	Automatic	Electric

	price	year	mileage	tax	mpg	engineSize	manufacturer	model	transmission	fuelType
19346	9.995	2014	103.000	125	60.1	2.0	BMW	3 Series	Automatic	Diesel
19867	12.000	2017	88.100	0	141.2	1.5	BMW	2 Series	Automatic	Hybrid
19945	4.675	2009	70.000	165	47.9	2.0	BMW	3 Series	Manual	Petrol
19953	4.375	2005	55.000	160	50.4	2.0	BMW	3 Series	Manual	Diesel
20060	4.995	2005	84.000	305	24.6	4.4	BMW	6 Series	Automatic	Petrol
20151	11.269	2016	86.128	30	65.7	2.0	BMW	3 Series	Automatic	Diesel
20679	15.980	2011	46.000	570	22.6	4.4	BMW	X5	Automatic	Petrol
27352	135.771	2018	19.000	145	21.4	4.0	MERCEDES	G Class	Semi-Auto	Petrol
27668	63.999	2019	0.618	145	52.3	3.0	MERCEDES	S Class	Automatic	Diesel
28898	14.990	2017	76.982	0	134.5	2.0	MERCEDES	C Class	Semi-Auto	Hybrid
29631	66.899	2019	0.391	145	22.4	4.0	MERCEDES	GLC Class	Semi-Auto	Petrol
29900	75.729	2019	1.000	145	22.1	4.0	MERCEDES	GLC Class	Semi-Auto	Petrol
30291	71.899	2019	3.574	145	23.7	5.5	MERCEDES	GLE Class	Automatic	Petrol
31119	8.990	2010	128.000	555	32.5	3.0	MERCEDES	M Class	Automatic	Diesel
31612	15.491	2017	128.000	150	65.7	2.0	MERCEDES	E Class	Automatic	Diesel
32833	1.995	2005	105.000	260	43.5	2.1	MERCEDES	CLK	Automatic	Diesel
33000	12.995	2007	45.000	570	23.3	5.0	MERCEDES	SL CLASS	Automatic	Petrol
33246	4.990	2002	75.034	325	30.0	2.3	MERCEDES	SLK	Automatic	Petrol
33447	22.948	2013	39.000	570	23.5	6.2	MERCEDES	C Class	Automatic	Petrol
33463	7.495	2008	58.000	330	29.7	3.0	MERCEDES	SLK	Automatic	Petrol
35344	4.998	2009	66.000	165	44.8	1.4	VM	Golf	Manual	Petrol
38164	23.990	2017	9.444	140	156.9	1.4	VM	Golf	Semi-Auto	Hybrid
39193	9.399	2016	85.144	20	68.9	2.0	VM	Golf	Manual	Diesel
40143	22.495	2018	26.982	135	156.9	1.4	VM	Golf	Automatic	Other
40317	19.698	2017	25.088	0	166.0	1.4	VM	Passat	Semi-Auto	Hybrid
44402	3.999	2009	65.621	145	48.7	1.2	VM	Polo	Manual	Petrol
46158	14.240	2017	87.000	150	58.9	2.0	VM	Tiguan	Manual	Diesel
47805	8.000	2015	122.150	20	55.4	2.0	VM	Scirocco	Manual	Diesel

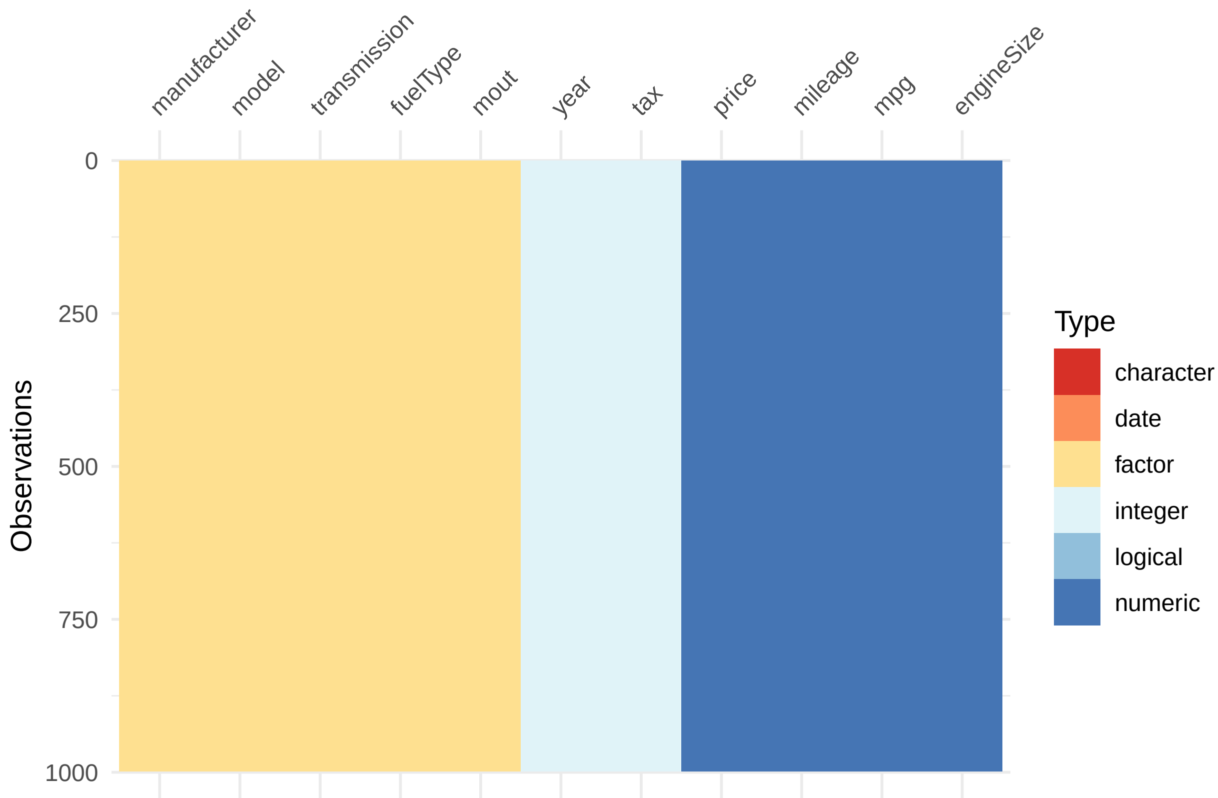
```
mout$md[llmout]
```

```
##      2256      7643     10109     10340     11316     15053     16171     16257
## 11.576425  5.467496  8.447850  4.847008  4.103400  6.900944 15.676894  4.132313
##     17261     19044     19069     19346     19867     19945     19953     20060
##  4.586751 15.726730 15.898612  4.441224  5.419477  4.033843  7.250232  7.279305
##     20151     20679     27352     27668     28898     29631     29900     30291
##  4.008143  5.746268 13.412620  3.991689  4.731184  4.075859  4.885811  5.883292
##     31119     31612     32833     33000     33246     33447     33463     35344
##  5.385830  7.620749  6.157573  8.293189  8.788397  9.117580  5.160824  4.635923
##     38164     39193     40143     40317     44402     46158     47805
##  4.130445  3.958322  3.919988  4.168888  4.788275  4.581712  6.188509
```

```
df$mout <- 0
df$mout[ llmout ] <- 1
df$mout <- factor( df$mout, labels = c("MvOut.No","MvOut.Yes"))
```

Checking missing data in the selected data frame:

```
vis_dat(df, sort_type = TRUE, palette = "cb_safe")
```

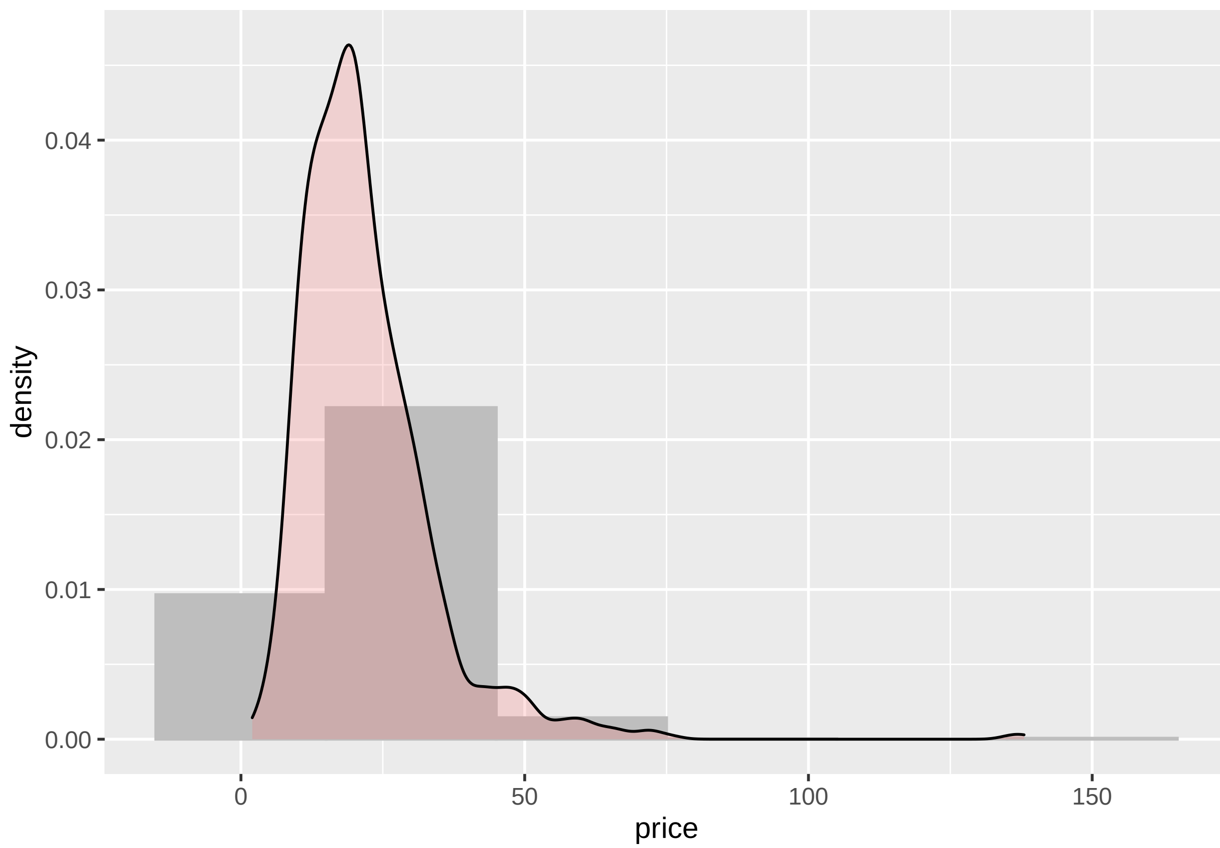


As shown, there are no duplicated data.

Responses

1. Determine if the response variable (price) has an acceptably normal distribution. Address test to discard serial correlation.

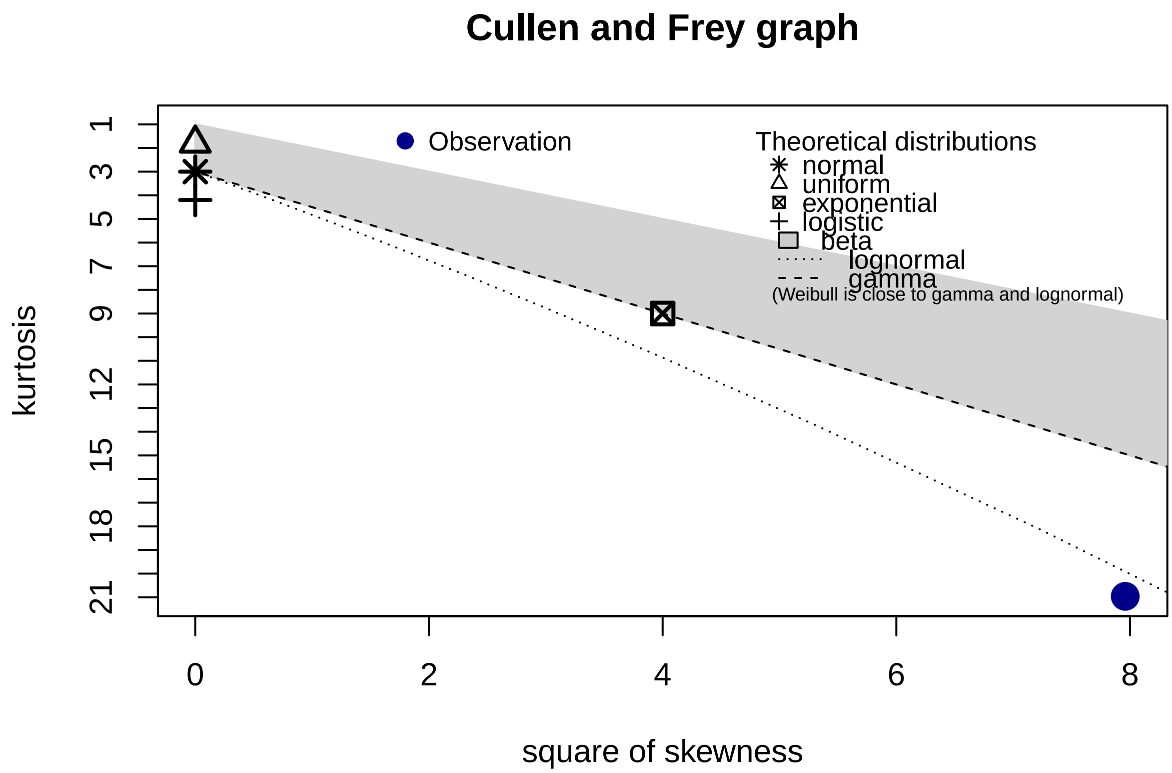
```
# Histogram with density plot price variable
ggplot(df, aes(x=price)) + geom_histogram(aes(y=..density..),
                                          colour="gray",
                                          fill="gray",binwidth=30 ) +
  geom_density(alpha=.2, fill="#FF6666")
```




```
#Normality and serial correlation test
shapiro.test(df$price)
```

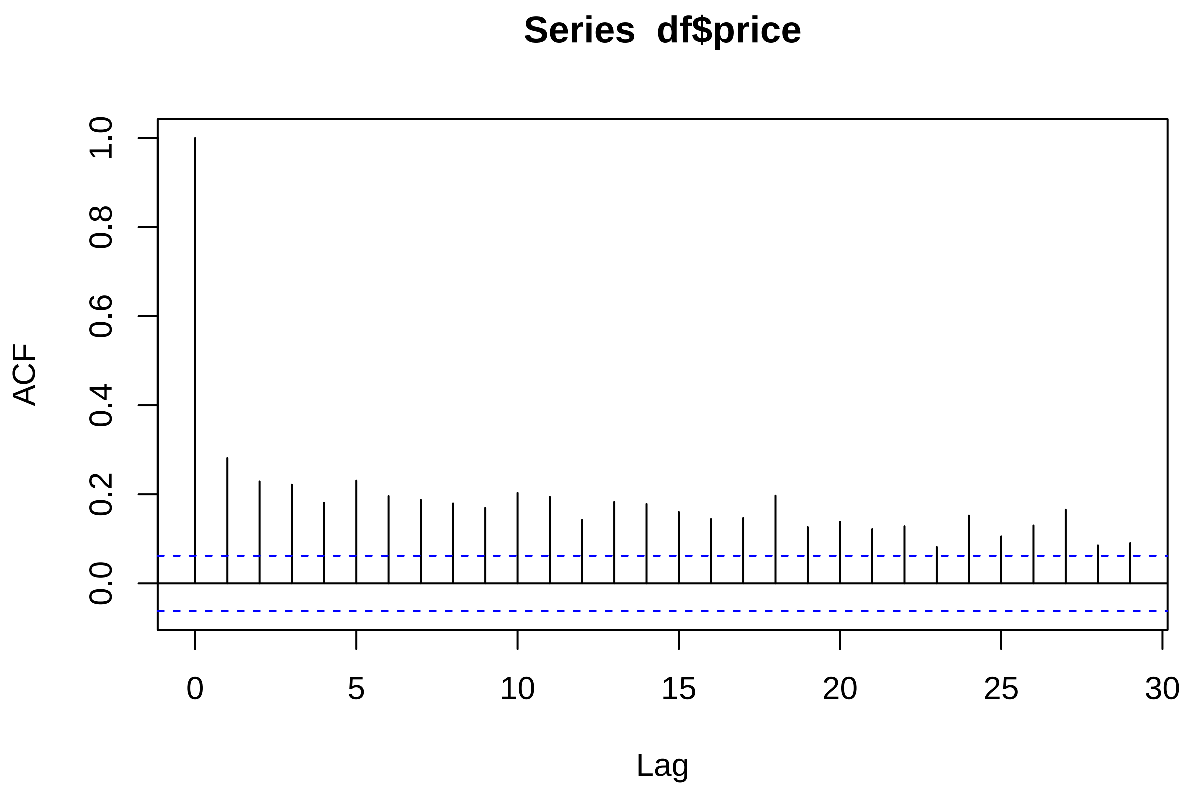
```
##
##  Shapiro-Wilk normality test
##
## data:  df$price
## W = 0.81687, p-value < 2.2e-16
```

```
descdist(df$price)
```




```
## summary statistics
## ------
## min:  1.99   max:  137.995
## median:  19.9205
## mean:  21.98812
## estimated sd:  12.03886
## estimated skewness:  2.821271
## estimated kurtosis:  20.96248
```

```
acf(df$price)
```




```
dwtest(df$price~1)
```

```
##
##  Durbin-Watson test
##
## data:  df$price ~ 1
## DW = 1.4355, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

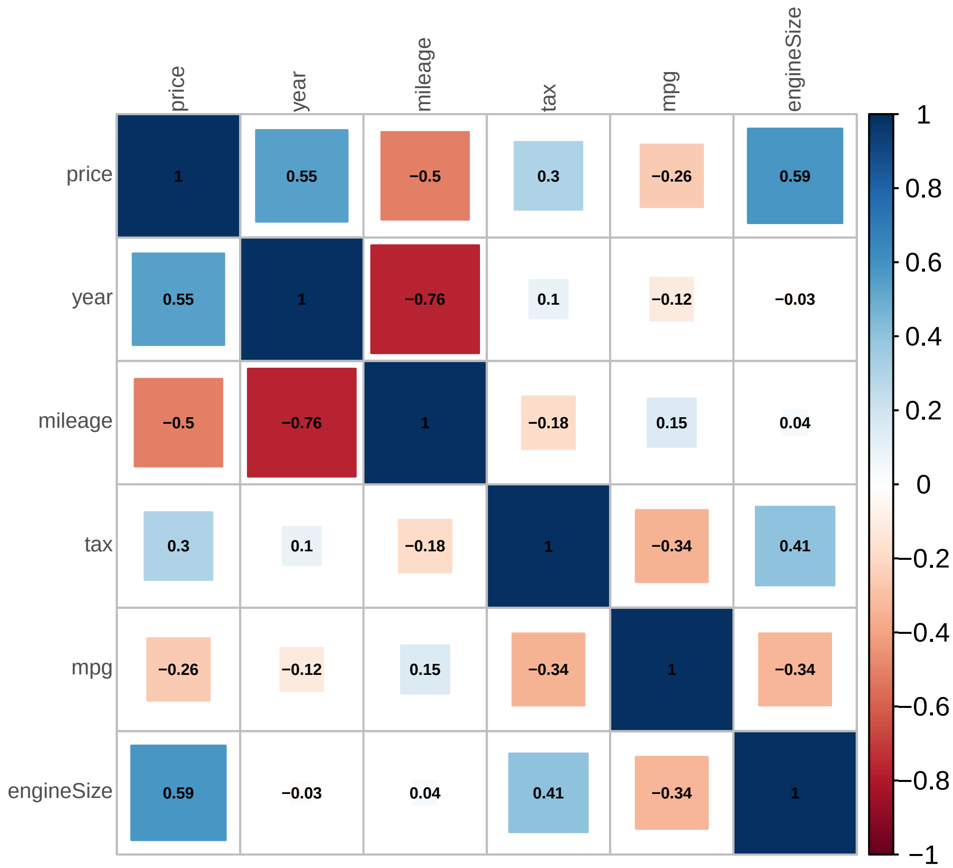
According to the graphic analysis and the shapiro-wilk test, the price variable does not follow a normal distribution and also present serial correlation. The Cullen and Frey graph suggests a lognormal distribution.

Hence, the response variable is transformed using Box-Cox, which suggests a better approach for modelling.

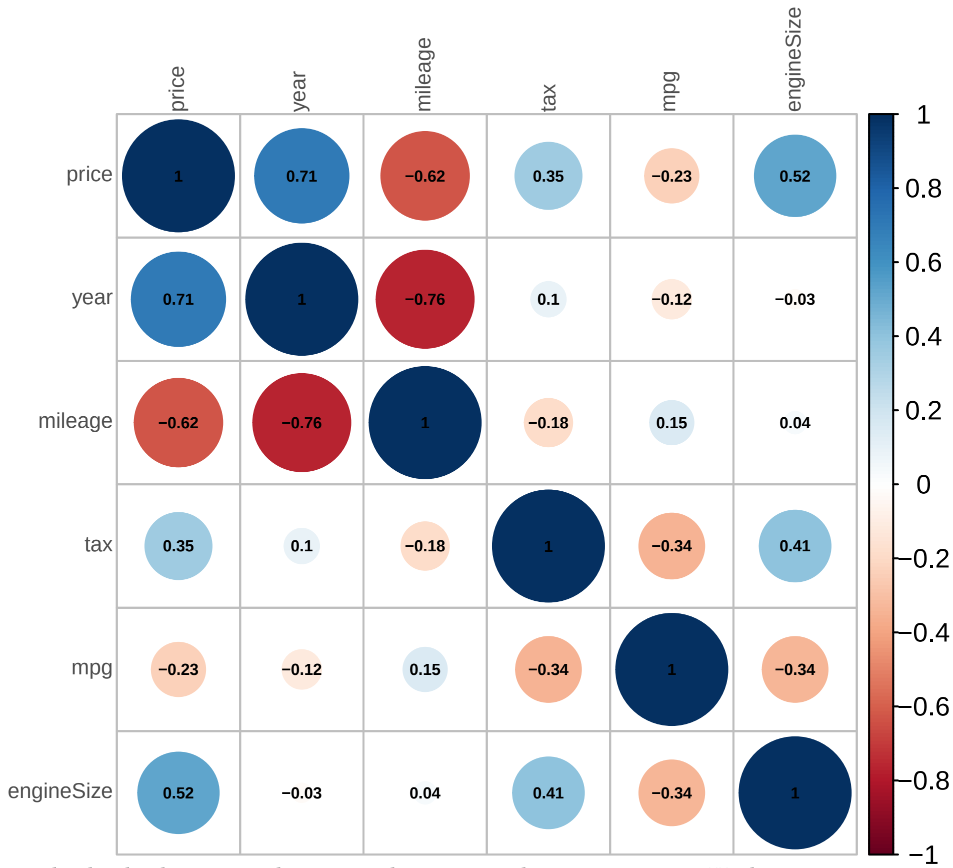
2. Indicate by exploration of the data which are apparently the variables most associated with the response variable (use only the indicated variables).

To identify the relationship of the response variable with the explanatory, the first tool used is the correlation matrix for numerical data.

```
corr<-cor(numy_var)

corrplot(corr,cex.main=0.7,method = c("square"),
         number.cex = 0.5,tl.cex=0.7,tl.col="gray31",
         cl.align="c",tl.offset = 0.1,addCoef.col=TRUE)
```




```
numlogy_var<-numy_var
numlogy_var$price<-log(numlogy_var$price)
#head(numlogy_var)
corrlog<-cor(numlogy_var)

corrplot(corrlog,cex.main=0.7,method = c("circle"),
         number.cex = 0.5,tl.cex=0.7,tl.col="gray31",cl.align="c",
         tl.offset = 0.1,addCoef.col=TRUE)
```



The variable with the highest correlation with price is the engine size. Higher engine cars present higher prices, followed by year and the tax. On the opposite, cars with lower mileage or miles per gallon values present higher prices.

As well, is used the continuous variable description of the package *factoMineR* to obtain insights for numerical and categorical variables:

```
con <- condes(df, num.var=1, proba = 0.01 )
con$quanti
```

```
##              correlation      p.value
## engineSize   0.5860008 4.475107e-93
## year         0.5493553 9.258183e-80
## tax          0.3005822 2.753923e-22
## mpg         -0.2558329 2.232036e-16
## mileage     -0.5025224 5.627401e-65
```

```
con$quali
```

```
##                      R2       p.value
## model        0.64418401 2.006737e-160
## transmission 0.20743078  5.902112e-51
## manufacturer 0.06228360  8.396730e-14
## mout         0.02216786  2.312233e-06
```

The most correlated qualitative variable with the response variable is model with 64,4%.

3. Define a polytomic factor f.age for the covariate car age according to its quartiles and argue if the average price depends on the level of age. Statistically justify the answer.

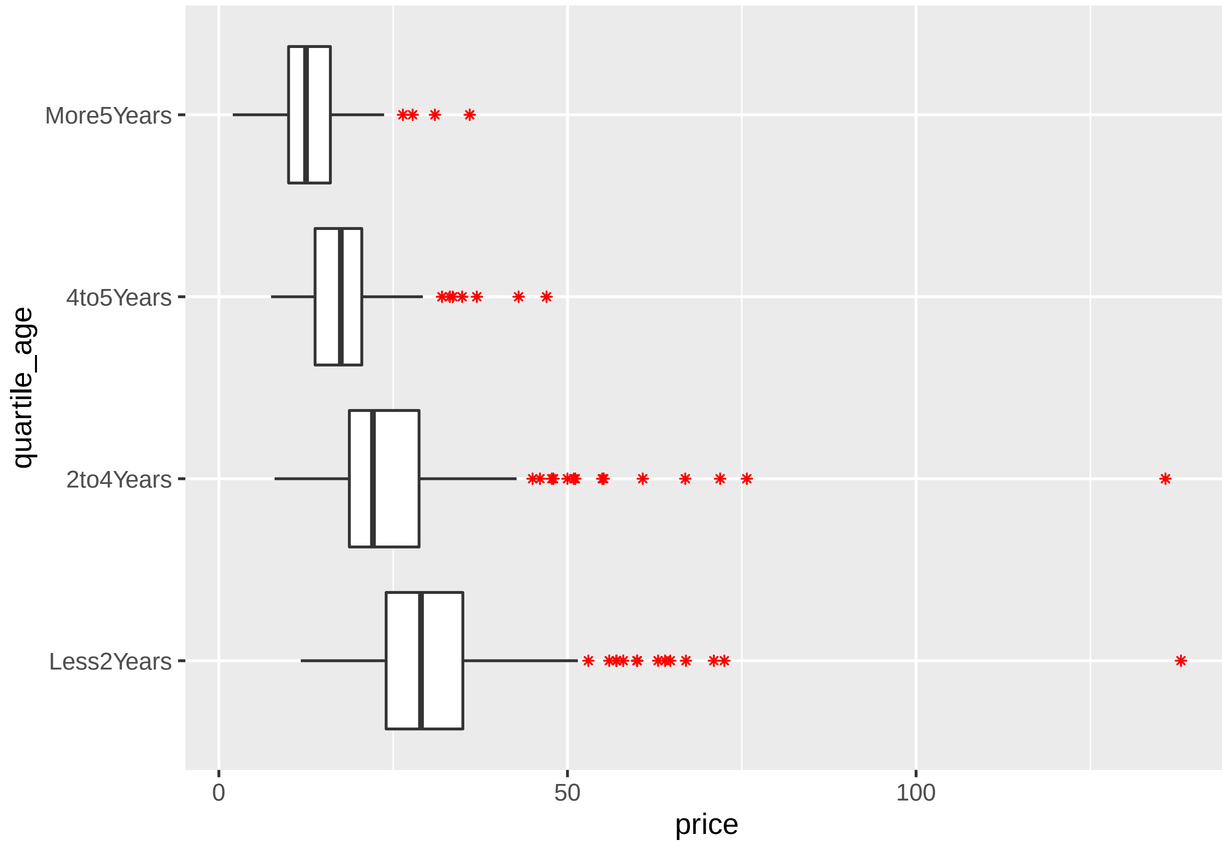
```
# Defining the factor variable required
df$age<-2021-df$year
df$quartile_age <- ntile(df$age, 4)
df$quartile_age<-as.factor(df$quartile_age)
#kable(table(df$quartile_age, df$year))
levels(df$quartile_age)<-c("Less2Years","2to4Years","4to5Years","More5Years")
#ggpairs(df[,c(4,12)], aes(color = df$quartile_age, alpha = 0.5))

res.cat <- catdes(df[,c("price","quartile_age")], num.var=2, proba = 0.01 )
res.cat$quanti
```

```
## $Less2Years
##         v.test Mean in category Overall mean sd in category Overall sd
## price 14.89157         31.80428     21.98812       12.92082   12.03283
##            p.value
## price 3.738599e-50
##
## $`2to4Years`
##         v.test Mean in category Overall mean sd in category Overall sd
## price 4.796973         25.15016     21.98812       12.36416   12.03283
##            p.value
## price 1.610816e-06
##
## $`4to5Years`
##         v.test Mean in category Overall mean sd in category Overall sd
## price -6.35706         17.78649     21.98812       5.730995   12.03283
##            p.value
## price 2.056523e-10
##
## $More5Years
##          v.test Mean in category Overall mean sd in category Overall sd
## price -13.35781         13.15942     21.98812       4.897734   12.03283
##           p.value
## price 1.06657e-40
```

```
tapply(df$price, df$quartile_age, mean )
```

```
## Less2Years  2to4Years  4to5Years More5Years
##   31.80428   25.15016   17.78649   13.15942
```

```
ggplot(df, aes(x=price, y=quartile_age)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
               outlier.size=1, notch=FALSE)
```




```r
kruskal.test(price~quartile_age, data = df )
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  price by quartile_age
## Kruskal-Wallis chi-squared = 515.54, df = 3, p-value < 2.2e-16
```

The mean price for cars with more than 4 years of age seem to be less than the others, taking into account mean summary across levels of age (Less than 2 years, 2 to 4 years, 4 to 5 years and more than five years). As expected Less than 2 years old cars have higher mean prices. All the levels present outlier price data.

Mean prices remarkable higher than the rest hypothesis is absolutely rejected according to the non-parametric Kruskal-Wallis homogeneity test for means (pvalue 2.2e-16).

4. Calculate and interpret the anova model that explains car price according to the age factor and the fuel type.

```r
m1 <- lm( price ~ ., data=df[,c("price","quartile_age","fuelType")])
summary(m1)
```

```
##
## Call:
## lm(formula = price ~ ., data = df[, c("price", "quartile_age",
##     "fuelType")])
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.973  -4.969  -1.434   2.767 111.614
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               32.7289     0.6874  47.613  < 2e-16 ***
## quartile_age2to4Years     -6.5660     0.8686  -7.560  9.2e-14 ***
## quartile_age4to5Years    -14.3745     0.8766 -16.398  < 2e-16 ***
## quartile_ageMore5Years   -18.8089     0.8732 -21.540  < 2e-16 ***
## fuelTypeElectric           0.6433     5.6357   0.114  0.90915
## fuelTypeHybrid             2.3958     2.8477   0.841  0.40038
## fuelTypeOther             -2.4080     5.6217  -0.428  0.66850
## fuelTypePetrol            -2.0062     0.6324  -3.172  0.00156 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.694 on 990 degrees of freedom
## Multiple R-squared:  0.3562, Adjusted R-squared:  0.3516
## F-statistic: 78.24 on 7 and 990 DF,  p-value: < 2.2e-16
```

```r
summary(Anova(m1))
```

```
##      Sum Sq            Df           F value           Pr(>F)
##  Min.   : 1075   Min.   :  3.0   Min.   :  2.861   Min.   :0.000000
##  1st Qu.:26114   1st Qu.:  3.5   1st Qu.: 47.509   1st Qu.:0.005629
##  Median :51153   Median :  4.0   Median : 92.156   Median :0.011257
##  Mean   :48420   Mean   :332.3   Mean   : 92.156   Mean   :0.011257
##  3rd Qu.:72092   3rd Qu.:497.0   3rd Qu.:136.804   3rd Qu.:0.016886
##  Max.   :93031   Max.   :990.0   Max.   :181.451   Max.   :0.022515
##                                  NA's   :1         NA's   :1
```

The model including both factor variables shows a low proportion of the variance in the response variable explained by age and the fueltype of the cars with The R-squared of the the 35%. The car prices down related with their age.

The ANOVA Fisher tests finds significant both variables with a level of significance of 95%. This could be given by the level of petrol of fuelType as the most common value in the variable.

5. Do you think that the variability of the price depends on both factors? Does the relation between price and age factor depend on fuel type?

```r
options(contrasts=c("contr.treatment","contr.treatment"))  # Set parametrization for factors
kruskal.test(df$price, df$fuelType)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  df$price and df$fuelType
## Kruskal-Wallis chi-squared = 6.6231, df = 4, p-value = 0.1572
```

```r
m0 <- lm( price ~ 1, data = df)
m1 <- lm( price ~ fuelType+quartile_age, data = df)
m2 <- lm( price ~ fuelType*quartile_age, data = df)


summary(m1)
```

```
##
## Call:
## lm(formula = price ~ fuelType + quartile_age, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.973  -4.969  -1.434   2.767 111.614
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               32.7289     0.6874  47.613  < 2e-16 ***
## fuelTypeElectric           0.6433     5.6357   0.114  0.90915
## fuelTypeHybrid             2.3958     2.8477   0.841  0.40038
## fuelTypeOther             -2.4080     5.6217  -0.428  0.66850
## fuelTypePetrol            -2.0062     0.6324  -3.172  0.00156 **
## quartile_age2to4Years     -6.5660     0.8686  -7.560  9.2e-14 ***
## quartile_age4to5Years    -14.3745     0.8766 -16.398  < 2e-16 ***
## quartile_ageMore5Years   -18.8089     0.8732 -21.540  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.694 on 990 degrees of freedom
## Multiple R-squared:  0.3562, Adjusted R-squared:  0.3516
## F-statistic: 78.24 on 7 and 990 DF,  p-value: < 2.2e-16
```

```r
summary(m2)
```

```
##
## Call:
## lm(formula = price ~ fuelType * quartile_age, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.145  -5.019  -1.492   2.869 111.189
##
## Coefficients: (7 not defined because of singularities)
##                                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)                              32.4946     0.8650  37.564  < 2e-16
## fuelTypeElectric                          0.2237     5.6594   0.040    0.968
## fuelTypeHybrid                            4.8516     4.9315   0.984    0.325
## fuelTypeOther                            -1.8787     6.9102  -0.272    0.786
## fuelTypePetrol                           -1.6000     1.2386  -1.292    0.197
## quartile_age2to4Years                    -6.7507     1.2283  -5.496 4.95e-08
## quartile_age4to5Years                   -13.7207     1.1614 -11.814  < 2e-16
## quartile_ageMore5Years                  -18.4740     1.1647 -15.861  < 2e-16
```

```
## fuelTypeElectric:quartile_age2to4Years        NA         NA      NA       NA
## fuelTypeHybrid:quartile_age2to4Years          NA         NA      NA       NA
## fuelTypeOther:quartile_age2to4Years      -1.3702    11.9498  -0.115    0.909
## fuelTypePetrol:quartile_age2to4Years      0.4384     1.7460   0.251    0.802
## fuelTypeElectric:quartile_age4to5Years        NA         NA      NA       NA
## fuelTypeHybrid:quartile_age4to5Years     -3.9862     6.0585  -0.658    0.511
## fuelTypeOther:quartile_age4to5Years           NA         NA      NA       NA
## fuelTypePetrol:quartile_age4to5Years     -1.5294     1.8162  -0.842    0.400
## fuelTypeElectric:quartile_ageMore5Years       NA         NA      NA       NA
## fuelTypeHybrid:quartile_ageMore5Years         NA         NA      NA       NA
## fuelTypeOther:quartile_ageMore5Years          NA         NA      NA       NA
## fuelTypePetrol:quartile_ageMore5Years    -0.6902     1.7797  -0.388    0.698
##
## (Intercept)                             ***
## fuelTypeElectric
## fuelTypeHybrid
## fuelTypeOther
## fuelTypePetrol
## quartile_age2to4Years                   ***
## quartile_age4to5Years                   ***
## quartile_ageMore5Years                  ***
## fuelTypeElectric:quartile_age2to4Years
## fuelTypeHybrid:quartile_age2to4Years
## fuelTypeOther:quartile_age2to4Years
## fuelTypePetrol:quartile_age2to4Years
## fuelTypeElectric:quartile_age4to5Years
## fuelTypeHybrid:quartile_age4to5Years
## fuelTypeOther:quartile_age4to5Years
## fuelTypePetrol:quartile_age4to5Years
## fuelTypeElectric:quartile_ageMore5Years
## fuelTypeHybrid:quartile_ageMore5Years
## fuelTypeOther:quartile_ageMore5Years
## fuelTypePetrol:quartile_ageMore5Years
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.71 on 985 degrees of freedom
## Multiple R-squared:  0.3573, Adjusted R-squared:  0.3495
## F-statistic: 45.63 on 12 and 985 DF,  p-value: < 2.2e-16
```

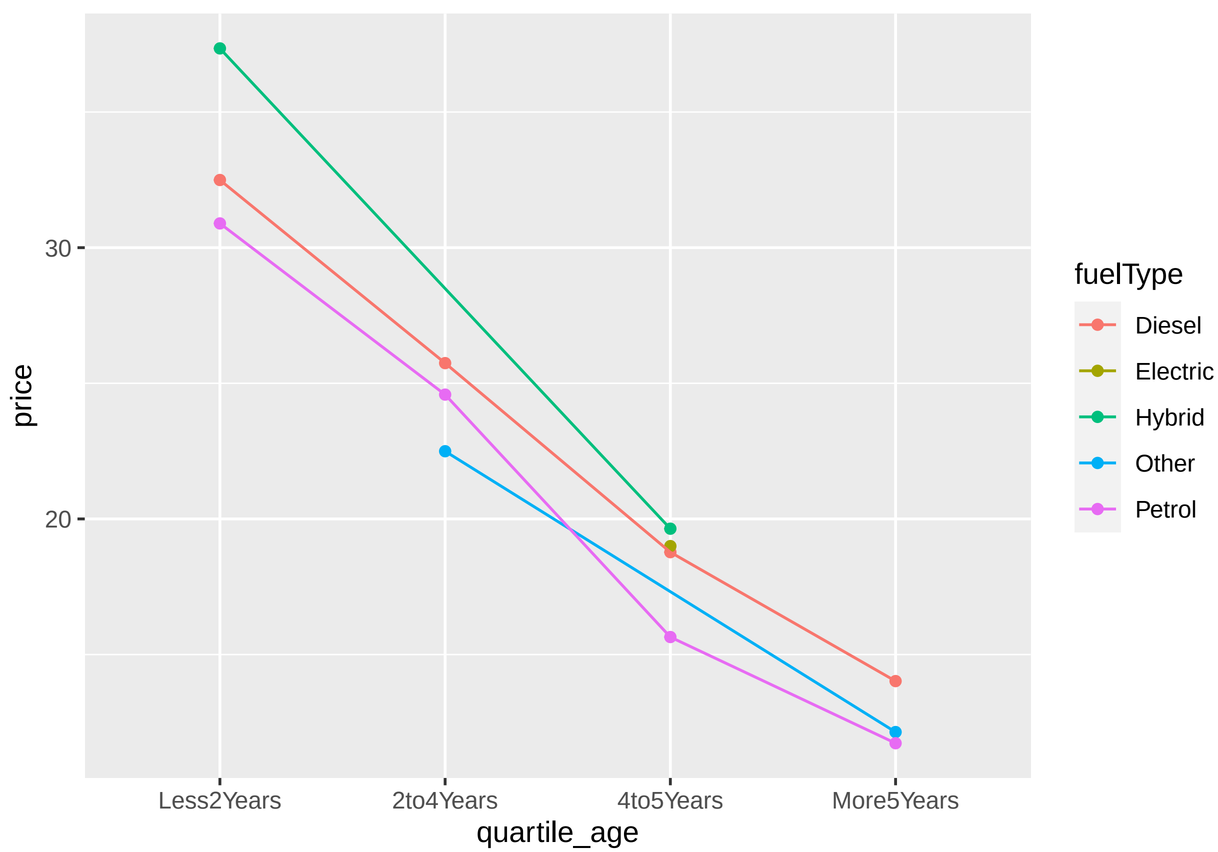
```
anova(m2)
```

```
## Analysis of Variance Table
##
## Response: price
##                       Df Sum Sq Mean Sq  F value Pr(>F)
## fuelType               4    315    78.9   0.8364 0.5021
## quartile_age           3  51153 17051.0 180.8442 <2e-16 ***
## fuelType:quartile_age  5    159    31.9   0.3379 0.8901
## Residuals            985  92872    94.3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

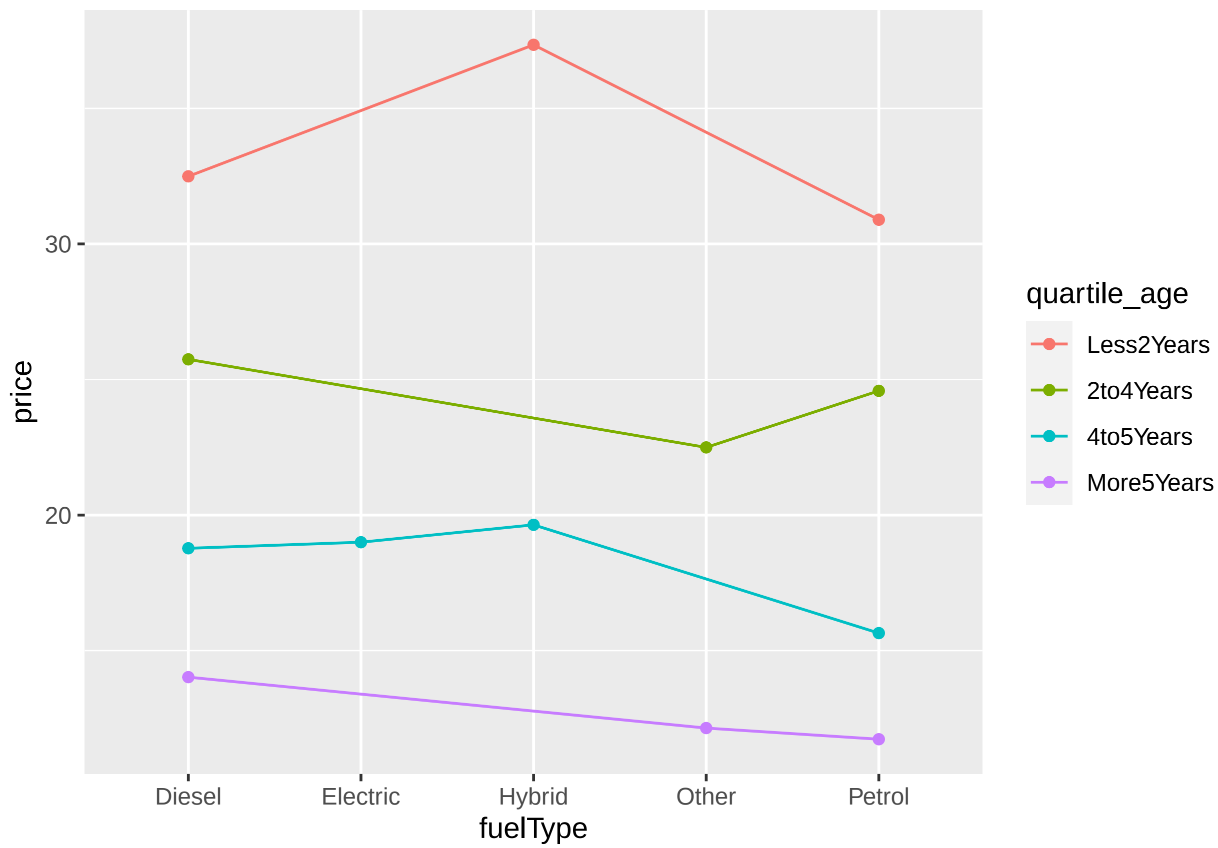
```r
# Interactions needed?
anova(m2,m1)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ fuelType * quartile_age
## Model 2: price ~ fuelType + quartile_age
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1    985 92872
## 2    990 93031 -5   -159.27 0.3379 0.8901
```

```r
par(mfrow=c(1,2))
df[,c("price","quartile_age","fuelType")] %>%
  ggplot() +
  aes(x = quartile_age, color = fuelType, group = fuelType, y = price) +
  stat_summary(fun = mean, geom = "point") +
  stat_summary(fun = mean, geom = "line")
```




```r
df[,c("price","quartile_age","fuelType")] %>%
  ggplot() +
  aes(x = fuelType, color = quartile_age, group = quartile_age, y = price) +
  stat_summary(fun = mean, geom = "point") +
  stat_summary(fun = mean, geom = "line")
```



According to the Kruskal-Wallis sum test for the fuelType factor is not significant for modelling mean prices (0.1572). Comparing the three models including a constant, an interaction between fuelType and age and one including both variables.

The summary of the model including both variables shows no significance of the variable fuelType with the exception of the Petrol category and a multiple R-squared of the 35%. The ANOVA analysis of the interaction model, reinforce this indicating there is no significance of the fuelType variable as is on the age variable.

Finally, the interaction plots show paralell behavior of the levels of the factors with an exception of the Petrol level in cars with 4 to 5 years old.

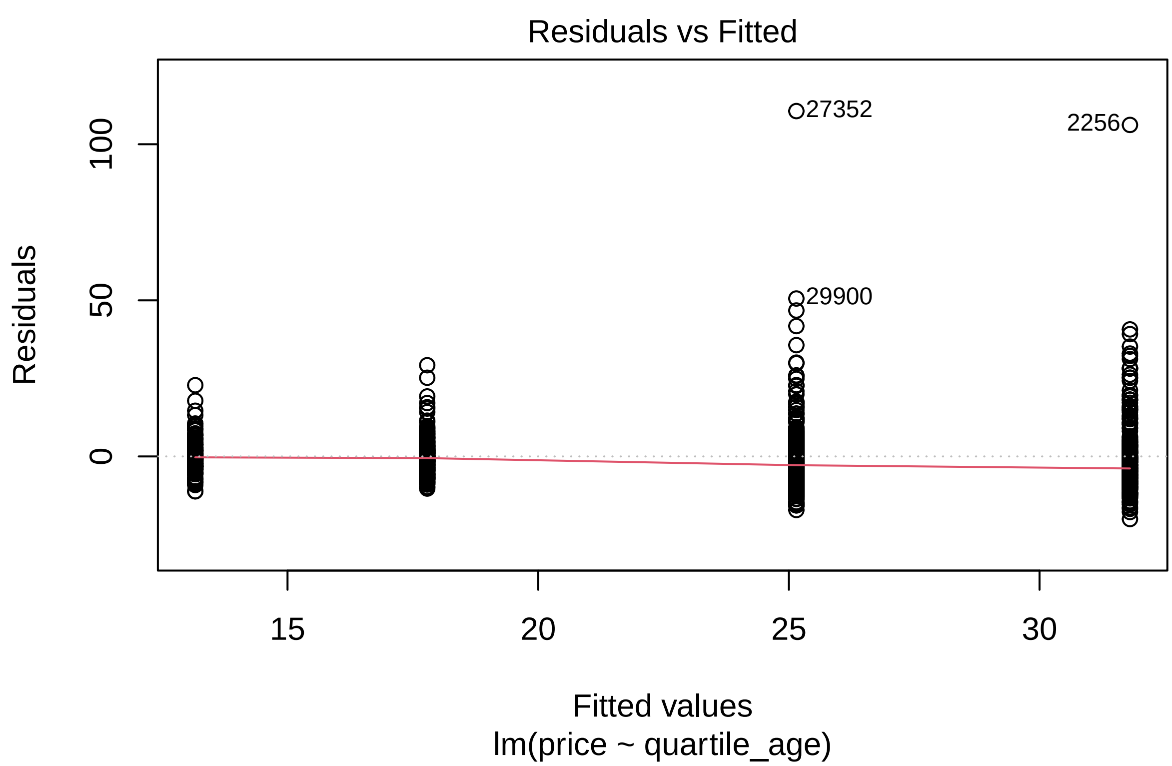
6. Calculate the linear regression model that explains the price from the age: interpret the regression line and assess its quality.

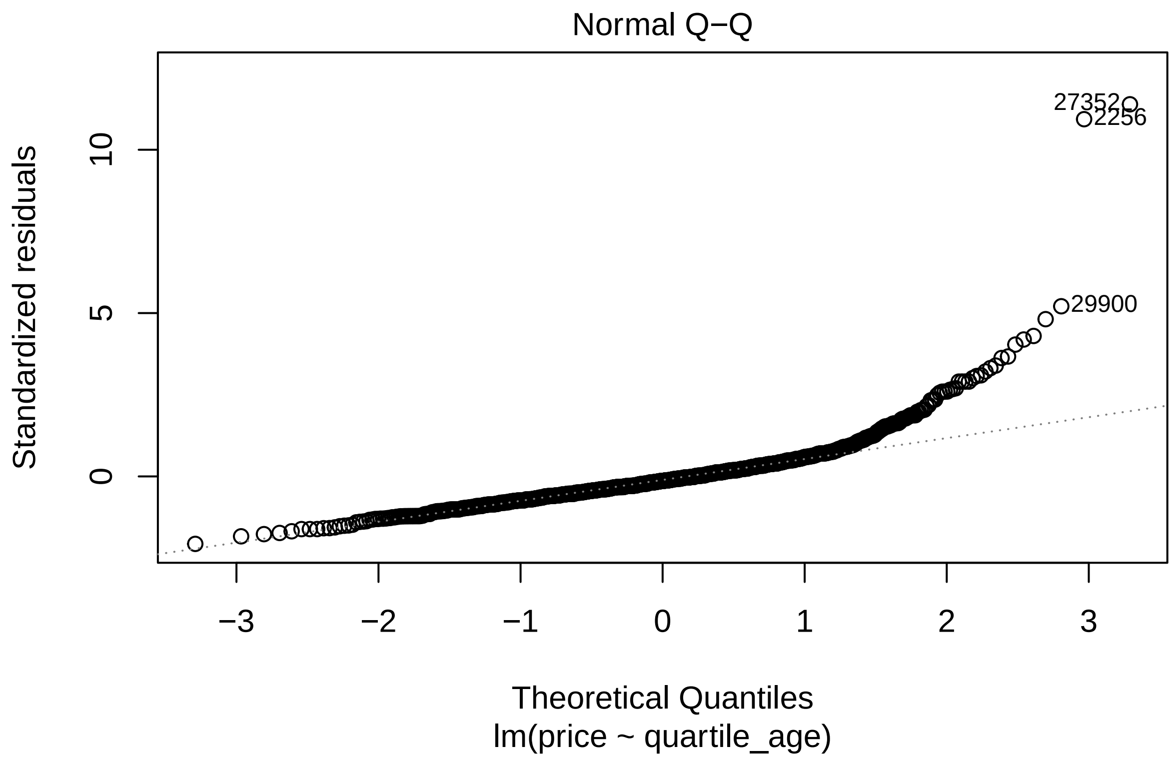
```
m3 <- lm( price ~ quartile_age, data=df[,c("price","quartile_age","fuelType")])
summary(m3)
```

```
##
## Call:
## lm(formula = price ~ quartile_age, data = df[, c("price", "quartile_age",
##     "fuelType")])
##
## Residuals:
##     Min       1Q   Median       3Q      Max
## -20.054   -5.236   -1.331    3.149 110.621
##
```

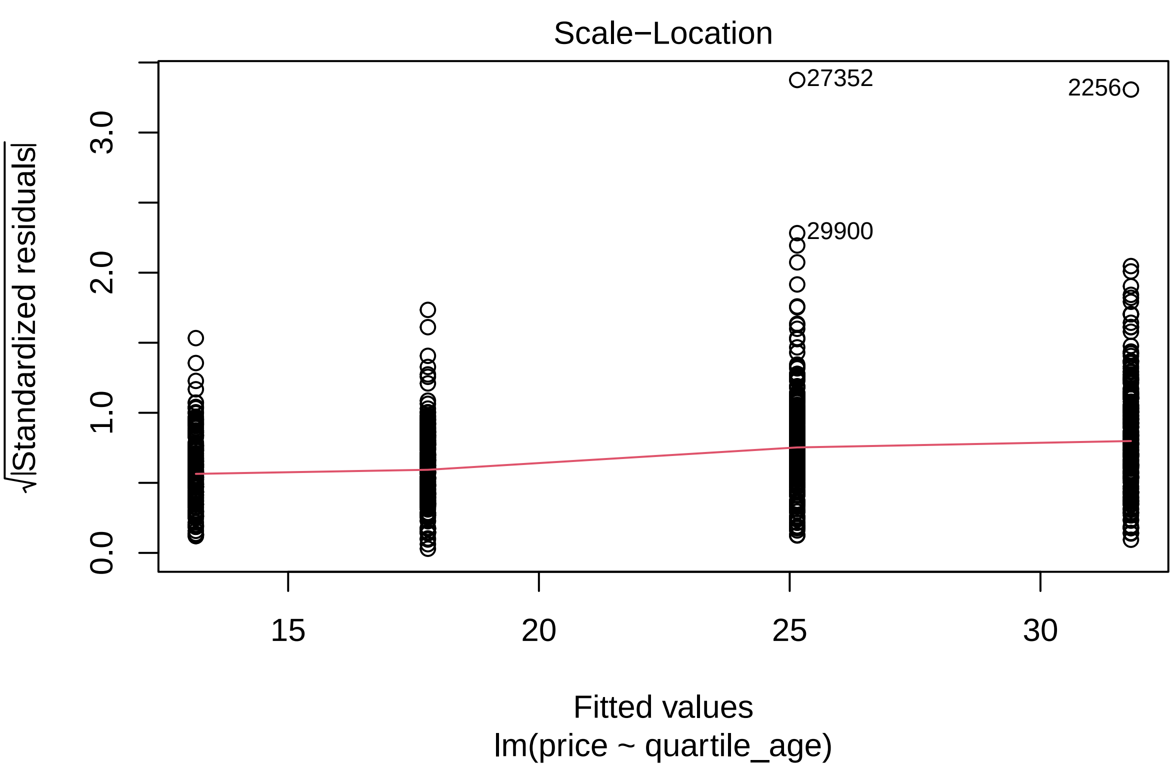
```
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)    
## (Intercept)               31.8043     0.6154  51.682  < 2e-16 ***
## quartile_age2to4Years     -6.6541     0.8703  -7.646 4.88e-14 ***
## quartile_age4to5Years    -14.0178     0.8712 -16.091  < 2e-16 ***
## quartile_ageMore5Years   -18.6449     0.8712 -21.402  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.73 on 994 degrees of freedom
## Multiple R-squared:  0.3487, Adjusted R-squared:  0.3468 
## F-statistic: 177.4 on 3 and 994 DF,  p-value: < 2.2e-16
```

```
#Residual analysis
plot(m3)
```

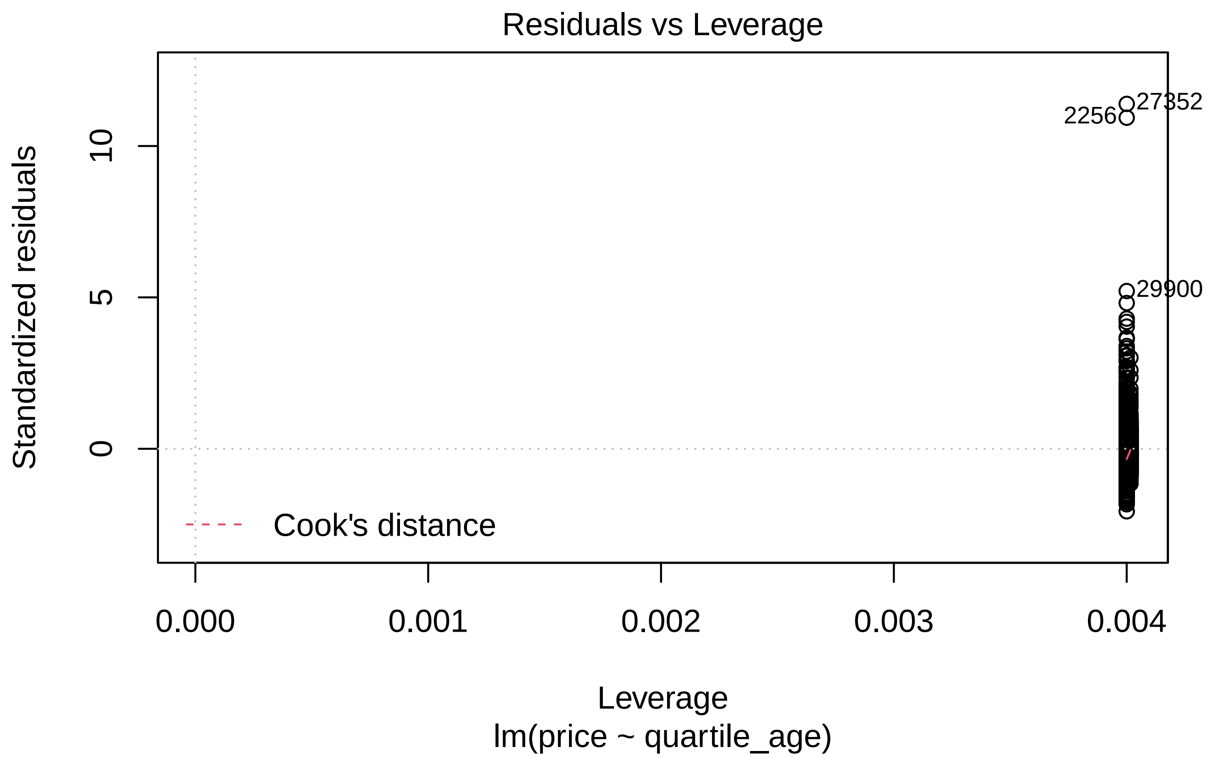




Normal Q−Q
Standardized residuals
10
5
0
27352
2256
29900
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(price ~ quartile_age)



Scale−Location
√|Standardized residuals|
3.0
2.0
1.0
0.0
27352
2256
29900
15
20
25
30
Fitted values
lm(price ~ quartile_age)




```
bptest(m3)# Null Hypothesis: Homoskedasticity holds - BP = 0.57522, df = 2, p-value = 0.7501
```

```
##
##  studentized Breusch-Pagan test
##
## data:  m3
## BP = 14.039, df = 3, p-value = 0.002853
```

The model including the age factor shows a significance of 99.9% per level. The residual plot displays not a normal distribution in the higher values related residuals and according to p-value of Breusch-pagan the null hypothesis of Homoskedastic test should be rejected.

7. What is the percentage of the price variability that is explained by the age of the car?

```
summary(m3)
```

```
##
## Call:
## lm(formula = price ~ quartile_age, data = df[, c("price", "quartile_age",
##     "fuelType")])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.054  -5.236  -1.331   3.149 110.621
```

```
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               31.8043     0.6154  51.682  < 2e-16 ***
## quartile_age2to4Years     -6.6541     0.8703  -7.646 4.88e-14 ***
## quartile_age4to5Years    -14.0178     0.8712 -16.091  < 2e-16 ***
## quartile_ageMore5Years   -18.6449     0.8712 -21.402  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.73 on 994 degrees of freedom
## Multiple R-squared:  0.3487, Adjusted R-squared:  0.3468
## F-statistic: 177.4 on 3 and 994 DF,  p-value: < 2.2e-16
```

```
af <- anova(m3)
 afss <- af$"Sum Sq"
 print(cbind(af,PctExp=afss/sum(afss)*100))
```

```
##              Df   Sum Sq     Mean Sq  F value       Pr(>F)   PctExp
## quartile_age   3 50393.16 16797.72094 177.4265 4.054335e-92 34.87432
## Residuals    994 94106.20    94.67424       NA           NA 65.12568
```

The model using the age factor explains the 34.87% of the variance in the price variable.

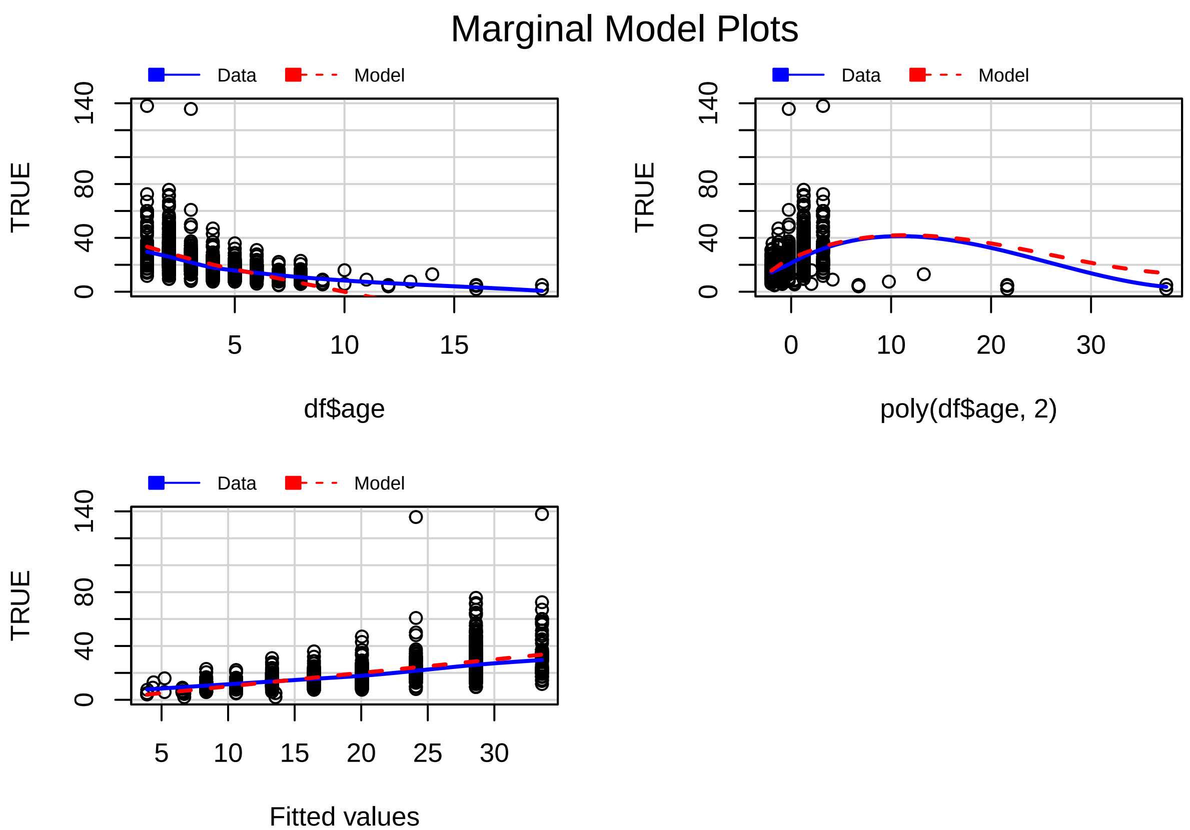
8. Do you think it is necessary to introduce a quadratic term in the equation that relates the price to its age?

```
m4 <- lm( price ~ df$age + poly(df$age,2), data=df[,c(1,12)])
summary(m4)
```

```
##
## Call:
## lm(formula = price ~ df$age + poly(df$age, 2), data = df[, c(1,
##     12)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.828  -5.116  -1.107   2.864 111.663
##
## Coefficients: (1 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        33.4036     0.6133  54.467   <2e-16 ***
## df$age             -3.0195     0.1404 -21.509   <2e-16 ***
## poly(df$age, 2)1        NA         NA      NA       NA
## poly(df$age, 2)2   84.2921     9.7086   8.682   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.709 on 995 degrees of freedom
## Multiple R-squared:  0.351,  Adjusted R-squared:  0.3497
## F-statistic:   269 on 2 and 995 DF,  p-value: < 2.2e-16
```

```
marginalModelPlots(m4)
```

```
## Warning in mmps(...): Splines and/or polynomials replaced by a fitted linear
## combination
```



A model using a quadratic term of the age shows a significance at the 99.9% of confidence of the squared age. This is confirmed by the marginal plots of the model using this variable.

9. Are there any additional explanatory numeric variables needed to the car price? Study collinearity effects.

To response this question, a model including each numerical variable is performed and compared with the one with factor age. In accordance with the exploratory analysis tax variable is not taking into account.

```
m0<-lm(price~1,data = df[,c("price","quartile_age","mileage","mpg","engineSize")])

m4<-lm(price ~., data = df[,c("price","quartile_age","mileage")])
anova(m0,m4)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ 1
## Model 2: price ~ quartile_age + mileage
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    997 144499
```

```
## 2    993  91930  4     52570 141.96 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(m3,m4)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ quartile_age
## Model 2: price ~ quartile_age + mileage
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    994 94106
## 2    993 91930  1    2176.6 23.511 1.441e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
vif(m4)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## quartile_age 2.081007  3        1.129915
## mileage      2.081007  1        1.442570
```

```r
m5<-lm(price ~., data = df[,c("price","quartile_age","mileage","mpg")])
anova(m0,m5)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ 1
## Model 2: price ~ quartile_age + mileage + mpg
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    997 144499
## 2    992  89280  5     55219 122.71 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(m4,m5)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ quartile_age + mileage
## Model 2: price ~ quartile_age + mileage + mpg
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    993 91930
## 2    992 89280  1    2649.4 29.438 7.255e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
vif(m5)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## quartile_age 2.203199  3        1.140711
## mileage      2.081009  1        1.442570
## mpg          1.083665  1        1.040992
```

```r
m6<-lm(price ~., data = df[,c("price","quartile_age","mileage","mpg","engineSize")])
anova(m0,m6)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ 1
## Model 2: price ~ quartile_age + mileage + mpg + engineSize
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    997 144499
## 2    991  45250  6     99249 362.27 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(m5,m6)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ quartile_age + mileage + mpg
## Model 2: price ~ quartile_age + mileage + mpg + engineSize
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    992 89280
## 2    991 45250  1     44030 964.29 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
vif(m6)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## quartile_age 2.235603  3        1.143490
## mileage      2.111424  1        1.453074
## mpg          1.228385  1        1.108325
## engineSize   1.157811  1        1.076016
```

According to the methods, the resting of numerical variables has significance inside the model. The Variance Inflation Factor close to one per each model indicates there is no correlation between the given predictors.

10. After controlling by numerical variables, indicate whether the additive effect of the available factors on the price are statistically significant.

```r
options(contrasts=c("contr.treatment","contr.treatment"))  # Set parametrization for factors

m7 <- lm(price~.,data = df[,c("price","mileage","mpg","engineSize")])

# Net-effects: For numerical: numerical | numerical+age factor or
# for age factor: age factor | numerical
anova( m7, m6 )
```

```
## Analysis of Variance Table
##
## Model 1: price ~ mileage + mpg + engineSize
```

```
## Model 2: price ~ quartile_age + mileage + mpg + engineSize
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    994 54922
## 2    991 45250  3    9671.6 70.605 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova( m6, m7 )
```

```
## Analysis of Variance Table
##
## Model 1: price ~ quartile_age + mileage + mpg + engineSize
## Model 2: price ~ mileage + mpg + engineSize
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    991 45250
## 2    994 54922 -3   -9671.6 70.605 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

According to the Fisher Test in the ANOVA analysis, for both cases, numerical variables adding age factor variable and, age factor adding the numerical variable adding the correspondant variables are significant with a p-value of 99.9%.

11. Select the best model available so far. Interpret the equations that relate the explanatory variables to the answer (rate).

So far the best model includes the age factor variable and numerical variables: mileage, mpg and engineSize. However, taking into account the rest of categorical variables available in the data set again is used the step-wise regression method to evaluate the best model. The mpg variable is removed according to a considerable increment in the Variance Inflation Factor.

```r
data<-df[,c("price","quartile_age","mileage","engineSize","model","transmission")]

full.model <- lm(price ~., data = data)
# Stepwise regression model
step.model <- stepAIC(full.model, direction = "both", trace = FALSE)
summary(step.model)
```

```
##
## Call:
## lm(formula = price ~ quartile_age + mileage + engineSize + model +
##     transmission, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.421  -2.123   0.000   1.972  28.646
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             15.142782   0.979452  15.460  < 2e-16 ***
## quartile_age2to4Years   -3.618440   0.439711  -8.229 6.43e-16 ***
## quartile_age4to5Years   -7.274837   0.491151 -14.812  < 2e-16 ***
```

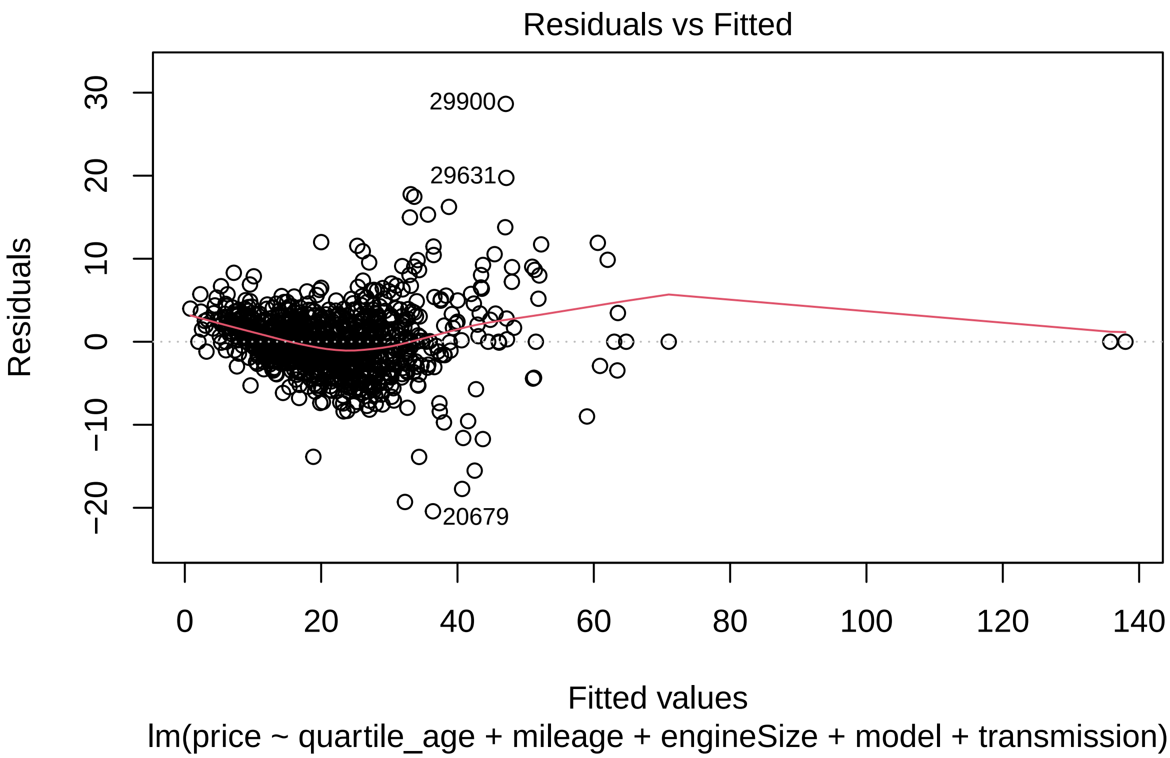
```
## quartile_ageMore5Years  -9.661225   0.570951 -16.921  < 2e-16 ***
## mileage                 -0.132892   0.009886 -13.442  < 2e-16 ***
## engineSize               5.883152   0.344874  17.059  < 2e-16 ***
## model2 Series            0.325660   1.037108   0.314 0.753586    
## model3 Series            1.114507   0.896748   1.243 0.214247    
## model4 Series            1.138819   1.113933   1.022 0.306890    
## model5 Series            4.867871   1.120355   4.345 1.55e-05 ***
## model6 Series           -1.351762   2.493918  -0.542 0.587934    
## model7 Series           18.196301   4.188534   4.344 1.55e-05 ***
## model8 Series           30.169434   3.011512  10.018  < 2e-16 ***
## modelA Class             2.418303   0.887087   2.726 0.006531 ** 
## modelA1                  1.441862   0.974369   1.480 0.139273    
## modelA3                  1.683212   1.053400   1.598 0.110413    
## modelA4                  1.343027   0.933103   1.439 0.150403    
## modelA5                  3.821356   1.340485   2.851 0.004460 ** 
## modelA6                  4.800245   1.153762   4.161 3.47e-05 ***
## modelA7                  2.677693   2.181569   1.227 0.219981    
## modelA8                  6.636187   3.002006   2.211 0.027311 *  
## modelAmarok              2.447142   2.203469   1.111 0.267038    
## modelArteon              5.991507   1.544213   3.880 0.000112 ***
## modelB Class             0.419254   1.355189   0.309 0.757111    
## modelC Class             3.903896   0.780514   5.002 6.81e-07 ***
## modelCaravelle          23.423815   4.185462   5.596 2.89e-08 ***
## modelCC                  3.561450   3.005988   1.185 0.236409    
## modelCL Class            3.033074   1.227211   2.472 0.013635 *  
## modelCLA Class          10.469905   4.183157   2.503 0.012492 *  
## modelCLK                -1.887468   4.225222  -0.447 0.655187    
## modelCLS Class           2.698867   2.471713   1.092 0.275163    
## modelE Class             4.563513   0.912626   5.000 6.85e-07 ***
## modelG Class           102.652992   4.235222  24.238  < 2e-16 ***
## modelGL Class            6.569368   2.173358   3.023 0.002575 ** 
## modelGLA Class           2.416849   1.269420   1.904 0.057236 .  
## modelGLB Class           8.170686   4.172589   1.958 0.050512 .  
## modelGLC Class          11.572438   1.113307  10.395  < 2e-16 ***
## modelGLE Class          18.625237   1.182405  15.752  < 2e-16 ***
## modelGolf                2.492579   0.789993   3.155 0.001656 ** 
## modelGolf SV            -0.507275   3.010819  -0.168 0.866240    
## modeli3                 13.935170   2.577904   5.406 8.24e-08 ***
## modeli8                 41.085085   4.179907   9.829  < 2e-16 ***
## modelM Class             2.869222   4.276479   0.671 0.502434    
## modelM2                 11.776775   4.188448   2.812 0.005033 ** 
## modelM3                 30.408389   4.188493   7.260 8.26e-13 ***
## modelM4                 12.979950   3.001607   4.324 1.70e-05 ***
## modelM5                 29.383391   4.257801   6.901 9.62e-12 ***
## modelPassat              0.842012   1.276163   0.660 0.509548    
## modelPolo                0.113692   0.847817   0.134 0.893353    
## modelQ2                  1.474600   1.103314   1.337 0.181711    
## modelQ3                  5.391691   0.931921   5.786 9.91e-09 ***
## modelQ5                  6.423177   1.157654   5.548 3.77e-08 ***
## modelQ7                 16.397129   2.197145   7.463 1.96e-13 ***
## modelQ8                 27.796938   2.495178  11.140  < 2e-16 ***
## modelR8                 91.683115   4.321146  21.217  < 2e-16 ***
## modelRS3                11.324407   3.000377   3.774 0.000171 ***
## modelS Class            19.564044   3.000709   6.520 1.16e-10 ***
```

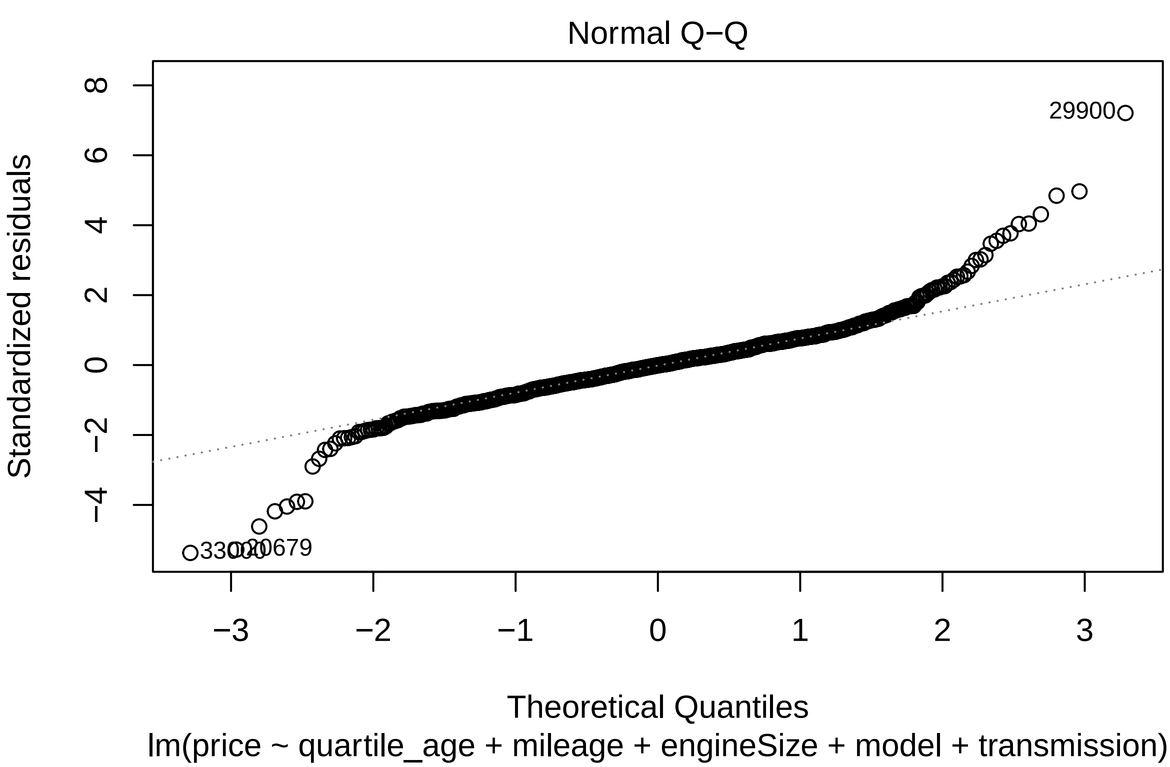
```
## modelScirocco            2.675680   2.158814   1.239 0.215507
## modelSharan              3.106787   1.960828   1.584 0.113442
## modelShuttle             2.614115   2.178221   1.200 0.230405
## modelSL CLASS            3.377386   1.964424   1.719 0.085902 .
## modelSLK                -5.989803   3.016129  -1.986 0.047339 *
## modelSQ5                 7.332651   4.205732   1.743 0.081583 .
## modelT-Cross             1.038703   1.711268   0.607 0.544016
## modelT-Roc               1.744368   1.379374   1.265 0.206333
## modelTiguan              4.944705   0.941388   5.253 1.87e-07 ***
## modelTiguan Allspace     4.465661   2.174277   2.054 0.040272 *
## modelTouareg            10.626324   1.443610   7.361 4.06e-13 ***
## modelTouran              3.260239   1.589638   2.051 0.040557 *
## modelTT                  6.051856   1.675622   3.612 0.000321 ***
## modelUp                 -2.641095   1.214345  -2.175 0.029892 *
## modelV Class             0.636110   4.181940   0.152 0.879135
## modelX-CLASS             6.346124   2.492738   2.546 0.011064 *
## modelX1                  1.879496   1.223876   1.536 0.124959
## modelX2                  1.303141   1.522910   0.856 0.392392
## modelX3                  6.622529   1.448508   4.572 5.49e-06 ***
## modelX4                  7.942290   2.985785   2.660 0.007950 **
## modelX5                 11.146789   1.495717   7.452 2.12e-13 ***
## modelX6                 12.735055   2.486033   5.123 3.67e-07 ***
## transmissionManual      -1.413395   0.431003  -3.279 0.001080 **
## transmissionSemi-Auto    0.586012   0.353551   1.658 0.097760 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.117 on 917 degrees of freedom
## Multiple R-squared:  0.8924, Adjusted R-squared:  0.8831
## F-statistic:  95.1 on 80 and 917 DF,  p-value: < 2.2e-16
```

```
vif(step.model)
```

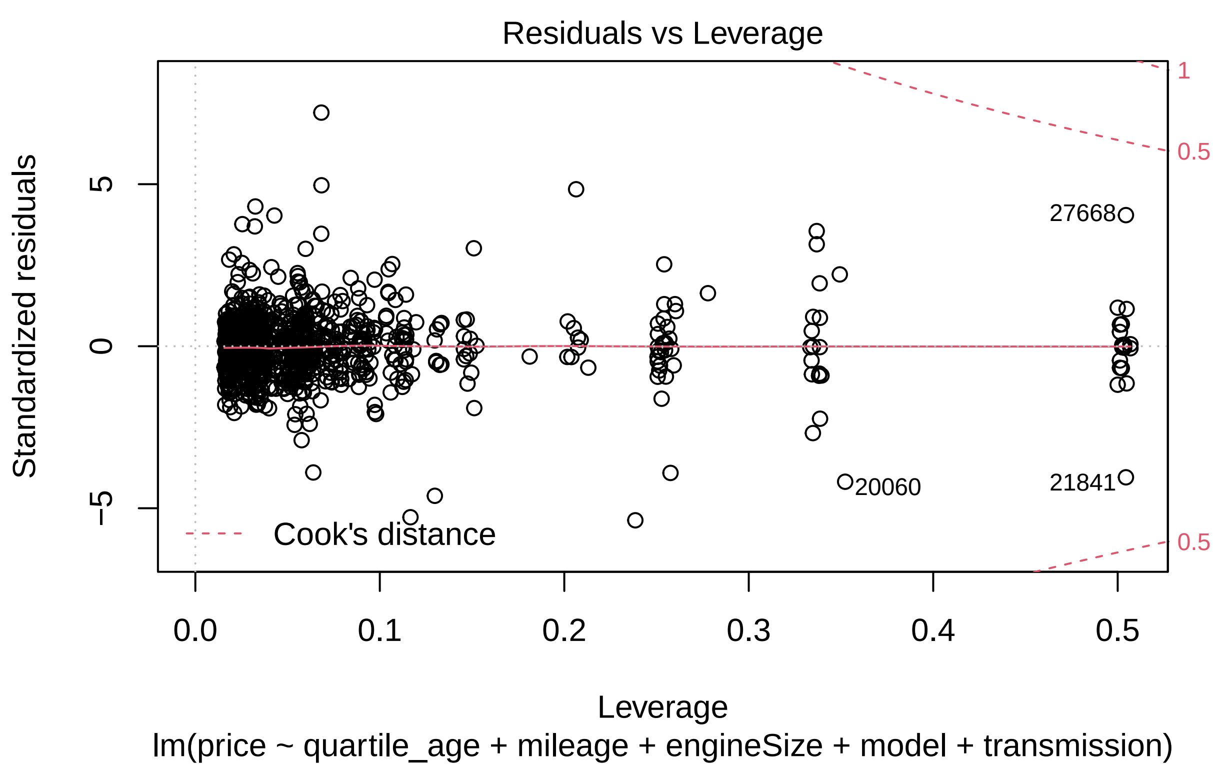
```
##                  GVIF Df GVIF^(1/(2*Df))
## quartile_age 4.104442  3        1.265345
## mileage      2.611955  1        1.616154
## engineSize   2.759691  1        1.661232
## model        9.364033 73        1.015439
## transmission 2.031400  2        1.193848
```

```
plot(step.model)
```


Residuals vs Fitted
Residuals
Fitted values
lm(price ~ quartile_age + mileage + engineSize + model + transmission)
29900
29631
20679
30
20
10
0
−10
−20
0
20
40
60
80
100
120
140


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(price ~ quartile_age + mileage + engineSize + model + transmission)
29900
20679
8
6
4
2
0
−2
−4
−3
−2
−1
0
1
2
3

Scale−Location
29900
33000
20679
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
0
20
40
60
80
100
120
140
Fitted values
lm(price ~ quartile_age + mileage + engineSize + model + transmission)



Residuals vs Leverage
27668
20060
21841
Cook's distance
1
0.5
0.5
Standardized residuals
−5
0
5
0.0
0.1
0.2
0.3
0.4
0.5
Leverage
lm(price ~ quartile_age + mileage + engineSize + model + transmission)

In general terms, the equation of the selected model is explained:

- The mean price of the cars decreases when age increases, per each level of the category established in the variable, the mean price: -3.62 miles of £ if age is between 2 to 4 years, -7.27 £ if age is between 4 to 5 years, and -9.66 £ if age is more than 5 years, taking as fixed the rest of the variables in the model.
- The mean price of the cars decreases when one unit of mileage increases in 0.63 miles of £ being fixed the rest of the variables in the model.
- The mean price of the cars increases when one unit of engine increases in 5.88 miles of £ being fixed the rest of the variables in the model.
- Depending on the model of the car, the price of the cars would varied between -5.98 and 102.65 miles £, being fixed the rest of the variables in the model.

12. Study the model that relates the logarithm of the price to the numerical variables.

```
data$price<-log(data$price)
m8 <- lm( price ~ .,data=data)
summary(m8)
```

```
##
## Call:
## lm(formula = price ~ ., data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.01077 -0.08314 -0.00196  0.08862  0.55498
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             2.8609829  0.0382221  74.852  < 2e-16 ***
## quartile_age2to4Years  -0.1113706  0.0171592  -6.490 1.40e-10 ***
## quartile_age4to5Years  -0.2168238  0.0191666 -11.313  < 2e-16 ***
## quartile_ageMore5Years -0.3981582  0.0222807 -17.870  < 2e-16 ***
## mileage                -0.0082206  0.0003858 -21.308  < 2e-16 ***
## engineSize              0.1874617  0.0134583  13.929  < 2e-16 ***
## model2 Series           0.0240385  0.0404720   0.594 0.552689
## model3 Series           0.0801932  0.0349946   2.292 0.022155 *
## model4 Series           0.1114625  0.0434700   2.564 0.010502 *
## model5 Series           0.2437256  0.0437206   5.575 3.26e-08 ***
## model6 Series          -0.1365169  0.0973225  -1.403 0.161037
## model7 Series           0.4976608  0.1634530   3.045 0.002396 **
## model8 Series           0.7042597  0.1175210   5.993 2.96e-09 ***
## modelA Class            0.1183379  0.0346176   3.418 0.000658 ***
## modelA1                 0.0038808  0.0380237   0.102 0.918729
## modelA3                 0.1072171  0.0411078   2.608 0.009250 **
## modelA4                 0.0989061  0.0364133   2.716 0.006728 **
## modelA5                 0.2176940  0.0523110   4.162 3.46e-05 ***
## modelA6                 0.2599758  0.0450243   5.774 1.06e-08 ***
## modelA7                 0.2522905  0.0851333   2.963 0.003121 **
## modelA8                 0.3609695  0.1171500   3.081 0.002123 **
## modelAmarok             0.1884560  0.0859880   2.192 0.028655 *
```

```
## modelArteon             0.2826233  0.0602612   4.690 3.15e-06 ***
## modelB Class            0.0124487  0.0528848   0.235 0.813956    
## modelC Class            0.1878455  0.0304587   6.167 1.04e-09 ***
## modelCaravelle          0.7462949  0.1633331   4.569 5.57e-06 ***
## modelCC                 0.1450398  0.1173054   1.236 0.216616    
## modelCL Class           0.2003272  0.0478906   4.183 3.15e-05 ***
## modelCLA Class          0.4081141  0.1632432   2.500 0.012592 *  
## modelCLK               -1.3026908  0.1648847  -7.901 7.92e-15 ***
## modelCLS Class          0.2276300  0.0964559   2.360 0.018487 *  
## modelE Class            0.2184307  0.0356142   6.133 1.28e-09 ***
## modelG Class            1.5425365  0.1652749   9.333  < 2e-16 ***
## modelGL Class           0.3432724  0.0848129   4.047 5.62e-05 ***
## modelGLA Class          0.1772922  0.0495377   3.579 0.000363 ***
## modelGLB Class          0.3220818  0.1628308   1.978 0.048226 *  
## modelGLC Class          0.4167182  0.0434456   9.592  < 2e-16 ***
## modelGLE Class          0.6004911  0.0461421  13.014  < 2e-16 ***
## modelGolf               0.1051860  0.0308286   3.412 0.000673 ***
## modelGolf SV           -0.0788544  0.1174939  -0.671 0.502303    
## modeli3                 0.4736987  0.1005999   4.709 2.88e-06 ***
## modeli8                 1.0470765  0.1631163   6.419 2.19e-10 ***
## modelM Class            0.2231355  0.1668849   1.337 0.181534    
## modelM2                 0.3874756  0.1634496   2.371 0.017965 *  
## modelM3                 0.7434925  0.1634514   4.549 6.12e-06 ***
## modelM4                 0.4910771  0.1171345   4.192 3.03e-05 ***
## modelM5                 0.5518194  0.1661560   3.321 0.000932 ***
## modelPassat             0.0258066  0.0498009   0.518 0.604447    
## modelPolo              -0.1674702  0.0330851  -5.062 5.02e-07 ***
## modelQ2                 0.1179120  0.0430556   2.739 0.006290 ** 
## modelQ3                 0.2722820  0.0363672   7.487 1.65e-13 ***
## modelQ5                 0.3332200  0.0451762   7.376 3.65e-13 ***
## modelQ7                 0.5941361  0.0857412   6.929 7.96e-12 ***
## modelQ8                 0.6955788  0.0973716   7.144 1.85e-12 ***
## modelR8                 1.0668444  0.1686280   6.327 3.91e-10 ***
## modelRS3                0.5678090  0.1170864   4.849 1.45e-06 ***
## modelS Class            0.5474179  0.1170994   4.675 3.38e-06 ***
## modelScirocco           0.0931425  0.0842453   1.106 0.269185    
## modelSharan             0.1659666  0.0765192   2.169 0.030343 *  
## modelShuttle            0.2436470  0.0850027   2.866 0.004247 ** 
## modelSL CLASS           0.1567591  0.0766595   2.045 0.041152 *  
## modelSLK               -0.6019548  0.1177011  -5.114 3.84e-07 ***
## modelSQ5                0.6141861  0.1641241   3.742 0.000194 ***
## modelT-Cross            0.0973827  0.0667804   1.458 0.145113    
## modelT-Roc              0.1390113  0.0538286   2.582 0.009963 ** 
## modelTiguan             0.2685341  0.0367366   7.310 5.83e-13 ***
## modelTiguan Allspace    0.2379614  0.0848488   2.805 0.005145 ** 
## modelTouareg            0.4124094  0.0563353   7.321 5.39e-13 ***
## modelTouran             0.1490878  0.0620339   2.403 0.016444 *  
## modelTT                 0.1119111  0.0653893   1.711 0.087335 .  
## modelUp                -0.4688882  0.0473885  -9.895  < 2e-16 ***
## modelV Class            0.1731977  0.1631957   1.061 0.288838    
## modelX-CLASS            0.2964350  0.0972764   3.047 0.002375 ** 
## modelX1                 0.1442661  0.0477604   3.021 0.002593 ** 
## modelX2                 0.1152155  0.0594299   1.939 0.052847 .  
## modelX3                 0.2932937  0.0565264   5.189 2.61e-07 ***
```

```
## modelX4                 0.3663813  0.1165170   3.144 0.001718 **
## modelX5                 0.4591083  0.0583687   7.866 1.03e-14 ***
## modelX6                 0.5102928  0.0970147   5.260 1.79e-07 ***
## transmissionManual     -0.1287629  0.0168194  -7.656 4.87e-14 ***
## transmissionSemi-Auto   0.0251648  0.0137969   1.824 0.068486 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1607 on 917 degrees of freedom
## Multiple R-squared:  0.9023, Adjusted R-squared:  0.8937
## F-statistic: 105.8 on 80 and 917 DF,  p-value: < 2.2e-16
```

```r
Anova(m8)
```

```
## Anova Table (Type II tests)
##
## Response: price
##              Sum Sq  Df F value    Pr(>F)
## quartile_age  8.5763   3 110.754 < 2.2e-16 ***
## mileage      11.7191   1 454.024 < 2.2e-16 ***
## engineSize    5.0079   1 194.018 < 2.2e-16 ***
## model        26.2795  73  13.947 < 2.2e-16 ***
## transmission  2.7736   2  53.727 < 2.2e-16 ***
## Residuals    23.6693 917
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The log transformation of the variable response price, increases model that explains all the variation in the response variable around its mean. Usually, the larger the R2, the better the regression model fits your observations. However, this guideline has important caveats that I'll discuss in both this post and the next post.

13. Once explanatory numerical variables are included in the model, are there any main effects from factors needed?

```r
# Gross-effects: Adding numeric variables and factors to a model without any variable
anova( m0, m8)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ 1
## Model 2: price ~ quartile_age + mileage + engineSize + model + transmission
##   Res.Df    RSS Df Sum of Sq     F    Pr(>F)
## 1    997 144499
## 2    917     24 80    144476 69966 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova( m3, m8)
```

```
## Analysis of Variance Table
```

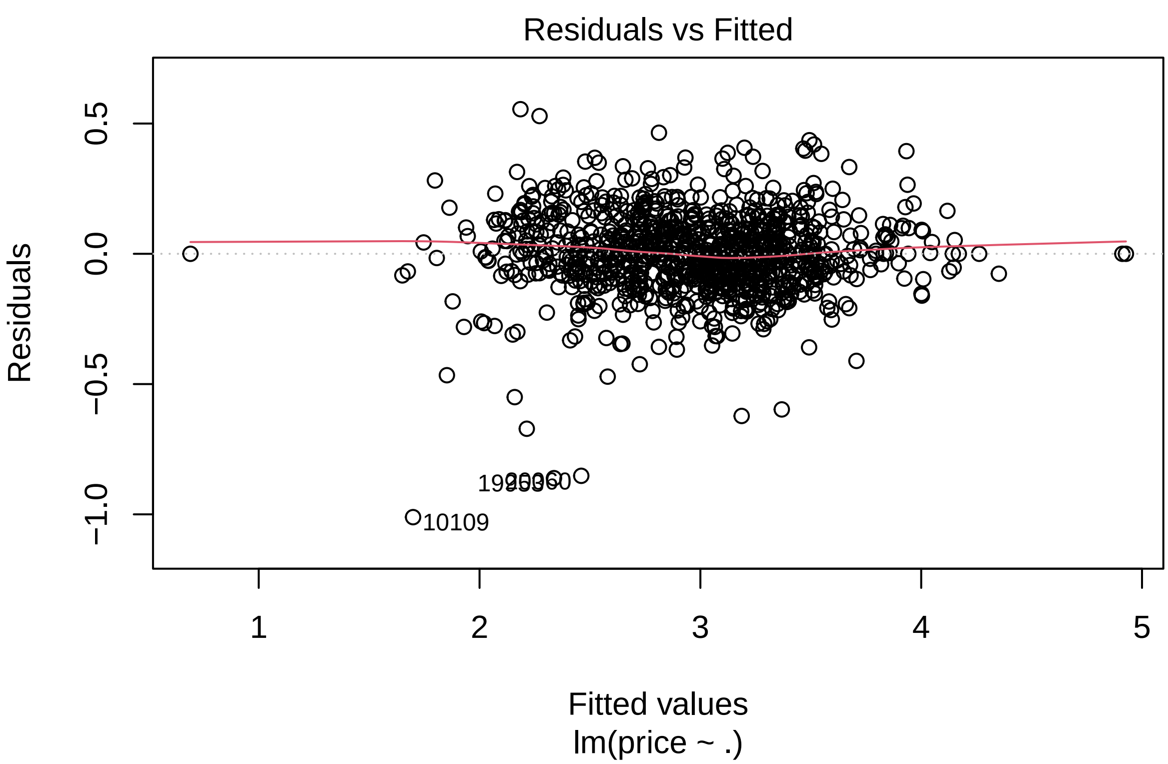
```
##
## Model 1: price ~ quartile_age
## Model 2: price ~ quartile_age + mileage + engineSize + model + transmission
##   Res.Df   RSS Df Sum of Sq     F    Pr(>F)
## 1    994 94106
## 2    917    24 77     94083 47337 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

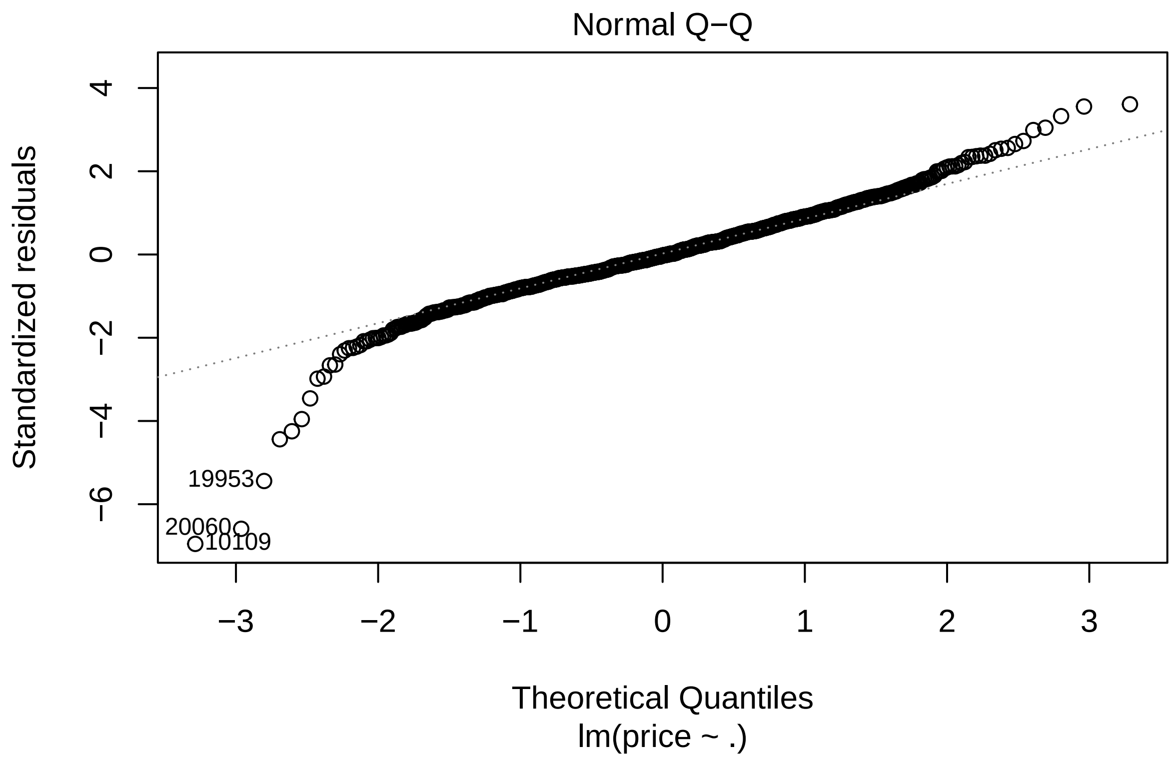
Including the factor variables model and transmission increases the R squared of the model. Even though some levels of this categorical variables are not significant, the Fisher test, shows the variables are significant compared with a constant model and as well considering numeric variables with an alpha =0.01.

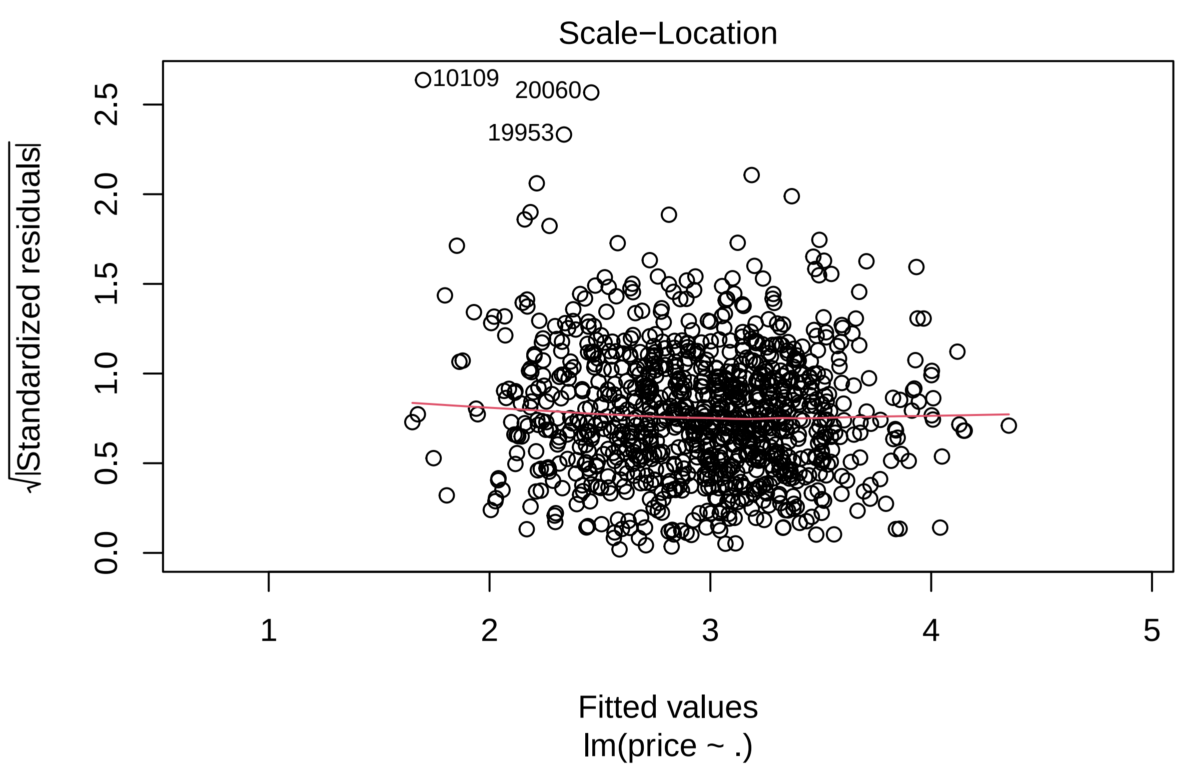
14. Graphically assess the best model obtained so far.

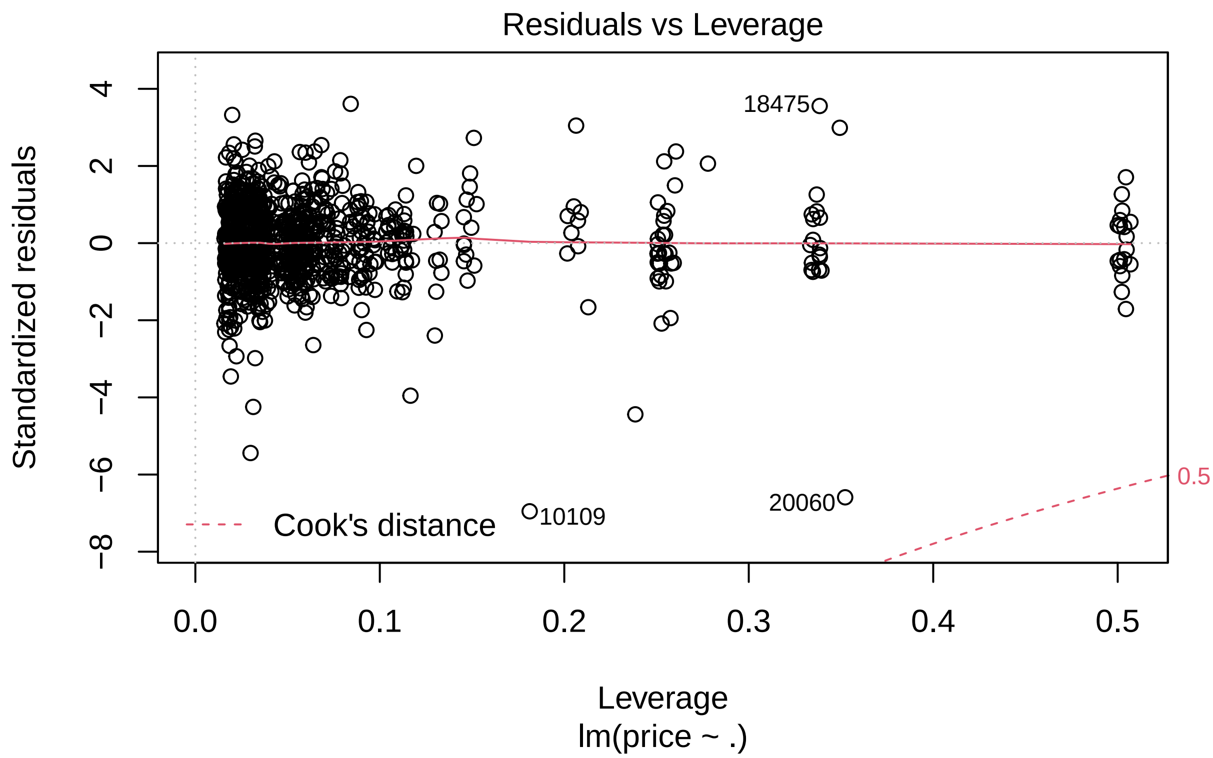
```
plot(m8)
```

```
## Warning: not plotting observations with leverage one:
##   57, 195, 293, 311, 331, 333, 350, 443, 465, 549, 647, 682, 705, 954
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(price ~ .)
19953
20060
10109
−3
−2
−1
0
1
2
3
4
−6
−4


Scale−Location
√|Standardized residuals|
Fitted values
lm(price ~ .)
10109
20060
19953
0.0
0.5
1.0
1.5
2.0
2.5
1
2
3
4
5

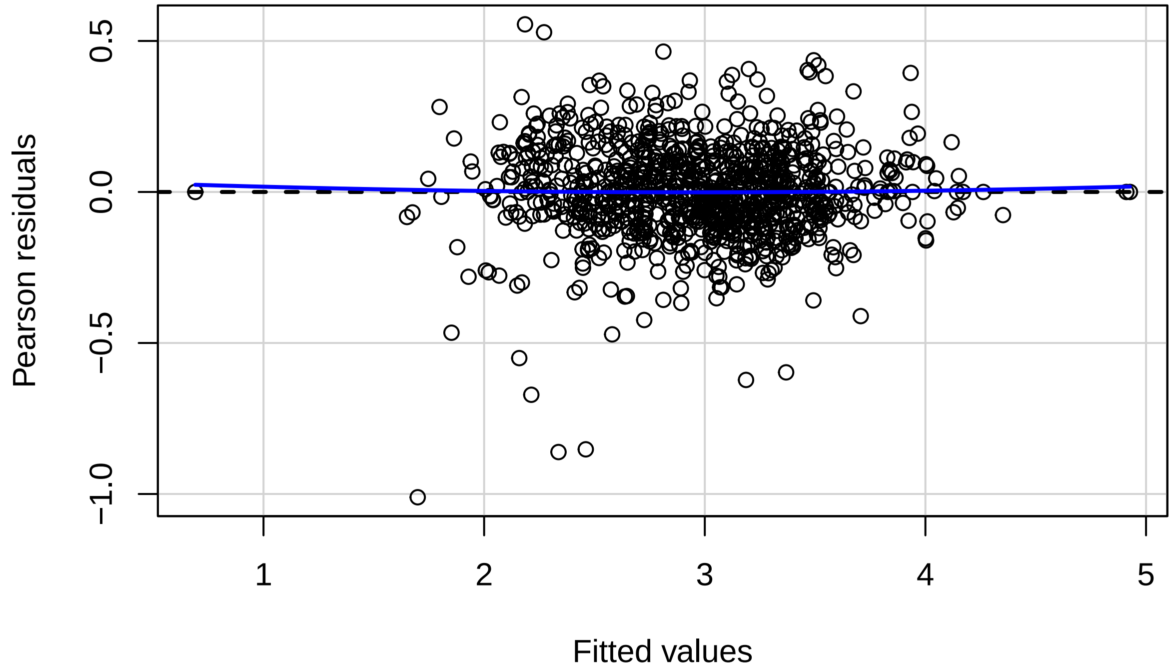


The selected model corresponding to: log-transformed response variable, numeric and categorical variables shows: for the scaled location plot the red line is approximately horizontal, then the average magnitude of the standardized residuals isn't changing much as a function of the fitted values. The QQPlot shows a general normal distribution with a deviation in high values. Cook-s distances graph shows there is low number of influential residuals in this model.

15. Assess the presence of outliers in the studentized residuals at a 99% confidence level. Indicate what those observations are.

In the selected model there is presence of leverage and influential data as shown in the graphs and listed in the next output.

```
# Default residual analysis:
par(mfrow=c(2,2))

# Metrics related to residuals:
par(mfrow=c(1,1))
residualPlot(m8)
```

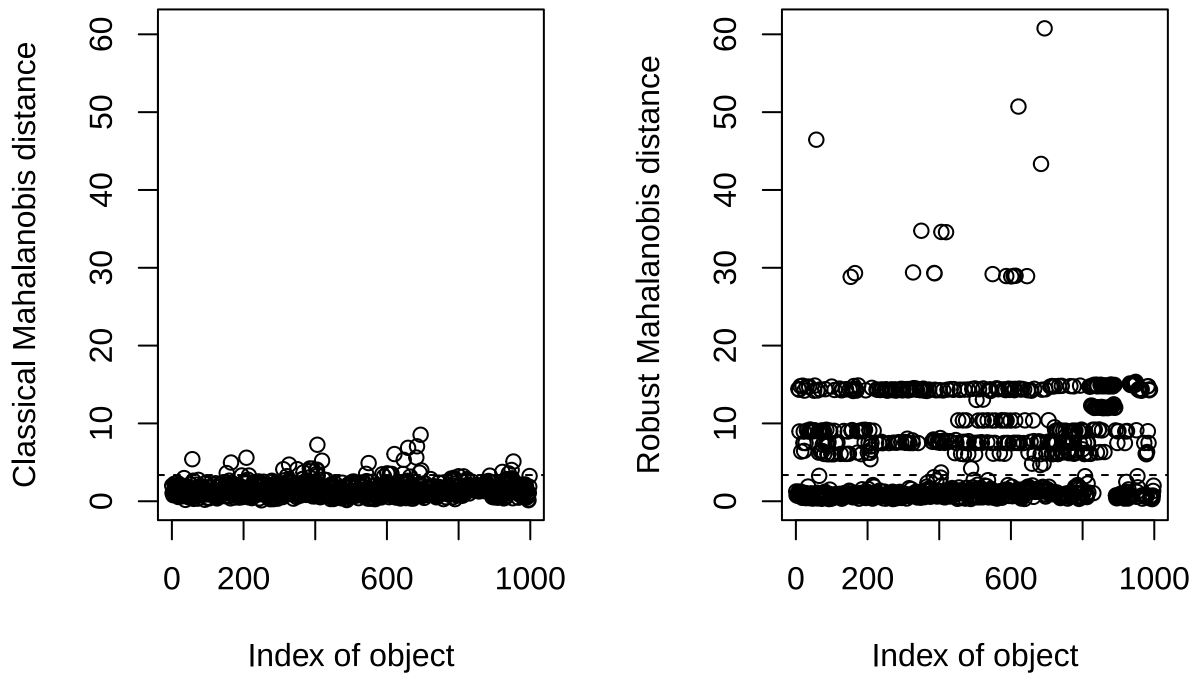



```r
rstan <- rstandard(m8) #Standardized residuals
rstud <- rstudent(m8) #Studentized residuals
```

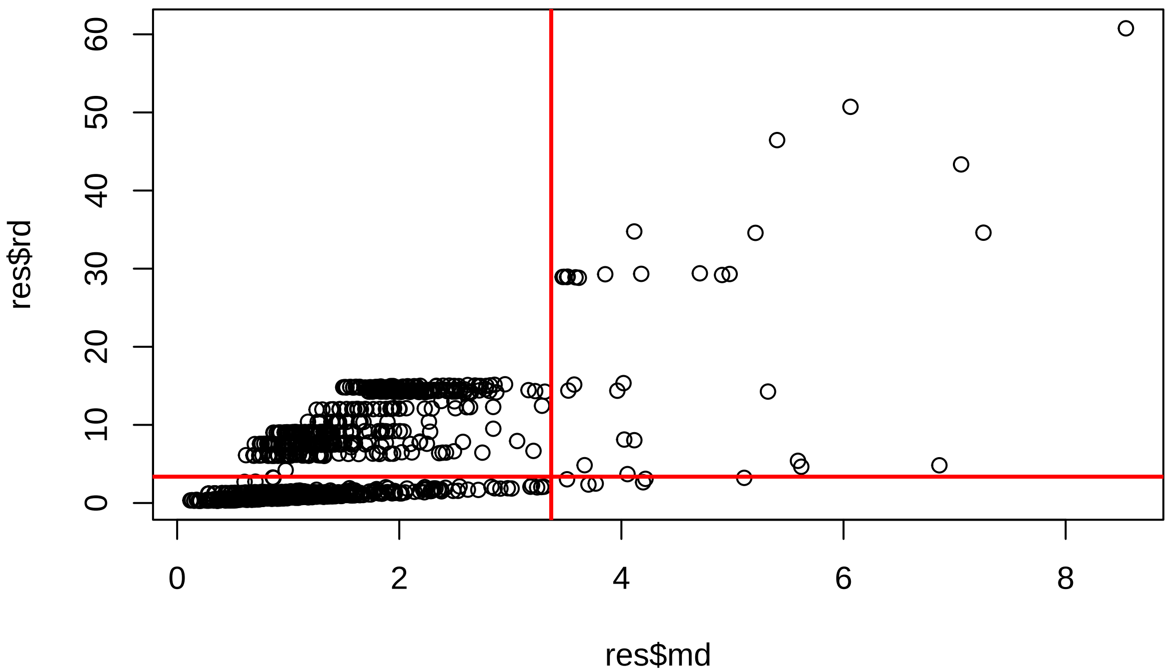
16. Study the presence of a priori influential data observations, indicating their number according to the criteria studied in class.

Using the moutlier function from chemometrics package. with a significance of 99%, there are 19 influential data observations a priori.

```r
res <- Moutlier(data[,c(1,3,4)],quantile=0.99)
```




```
par(mfrow=c(1,1))
plot( res$md, res$rd )
abline( h=res$cutoff, lwd=2, col="red")
abline( v=res$cutoff, lwd=2, col="red")
```




```
llmout <- which((res$md>mout$cutoff) & (res$rd > mout$cutoff) )

res$md[llmout]
```

```
##     2256     7643    10109    15053    16171    17261    19044    19867
## 5.401896 4.974438 5.589870 4.116304 4.706158 4.115692 4.178692 4.025602
##    20060    20679    27352    30291    31119    31612    32833    33000
## 7.260066 5.207268 4.906495 6.062669 5.319759 6.863033 5.619520 7.058692
##    33447    33463    47437
## 8.542743 3.963464 4.018830
```

```
data$mout <- 0
data$mout[ llmout ] <- 1
data$mout <- factor( df$mout, labels = c("MvOut.No","MvOut.Yes"))

kable(data[llmout,],table.attr = "style='width:30%;'")
```

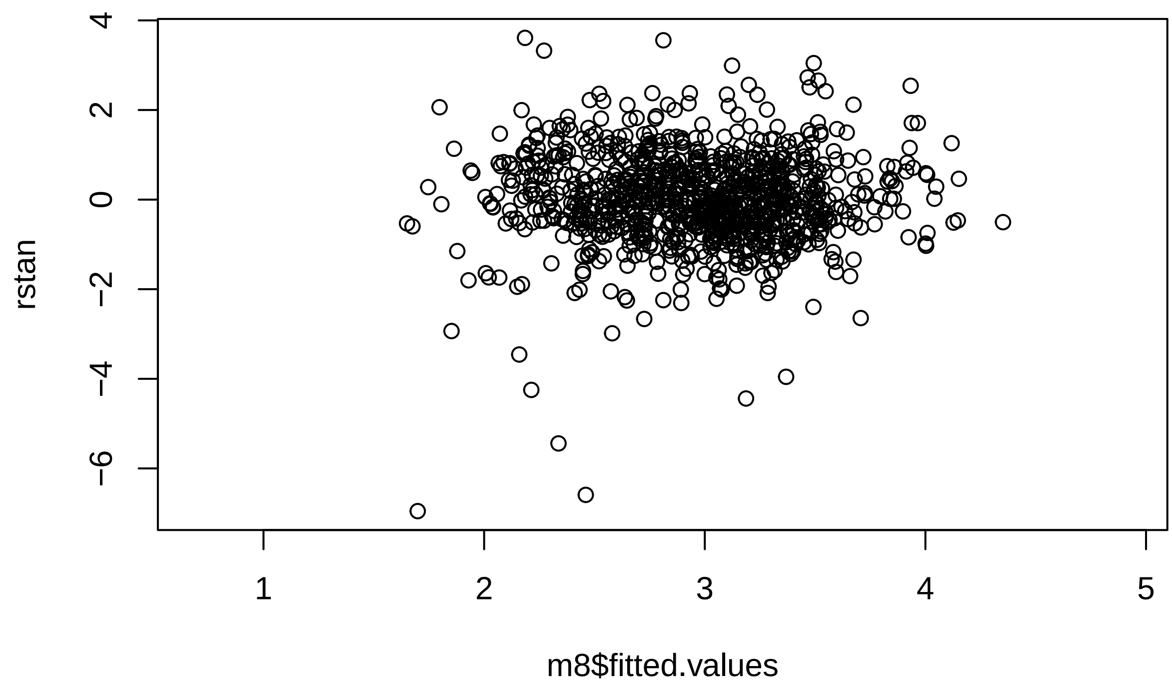
	price	quartile_age	mileage	engineSize	model	transmission	mout
2256	4.9272175	Less2Years	0.070	5.2	R8	Semi-Auto	MvOut.Yes
7643	3.4657359	Less2Years	4.000	0.0	Q3	Automatic	MvOut.Yes
10109	0.6881346	More5Years	131.925	1.8	TT	Manual	MvOut.Yes
15053	4.1705337	Less2Years	2.277	1.5	i8	Automatic	MvOut.Yes
16171	2.9441758	4to5Years	33.021	0.0	i3	Automatic	MvOut.Yes
17261	4.2626095	Less2Years	0.023	4.4	M5	Semi-Auto	MvOut.Yes

	price	quartile_age	mileage	engineSize	model	transmission	mout
19044	2.9443863	4to5Years	20.321	0.0	i3	Automatic	MvOut.Yes
19867	2.4849066	4to5Years	88.100	1.5	2 Series	Automatic	MvOut.Yes
20060	1.6084374	More5Years	84.000	4.4	6 Series	Automatic	MvOut.Yes
20679	2.7713379	More5Years	46.000	4.4	X5	Automatic	MvOut.Yes
27352	4.9109696	2to4Years	19.000	4.0	G Class	Semi-Auto	MvOut.Yes
30291	4.2752624	2to4Years	3.574	5.5	GLE Class	Automatic	MvOut.Yes
31119	2.1961128	More5Years	128.000	3.0	M Class	Automatic	MvOut.Yes
31612	2.7402592	4to5Years	128.000	2.0	E Class	Automatic	MvOut.Yes
32833	0.6906441	More5Years	105.000	2.1	CLK	Automatic	MvOut.Yes
33000	2.5645647	More5Years	45.000	5.0	SL CLASS	Automatic	MvOut.Yes
33447	3.1332308	More5Years	39.000	6.2	C Class	Automatic	MvOut.Yes
33463	2.0142361	More5Years	58.000	3.0	SLK	Automatic	MvOut.Yes
47437	1.6486586	More5Years	15.000	1.0	Up	Manual	MvOut.No

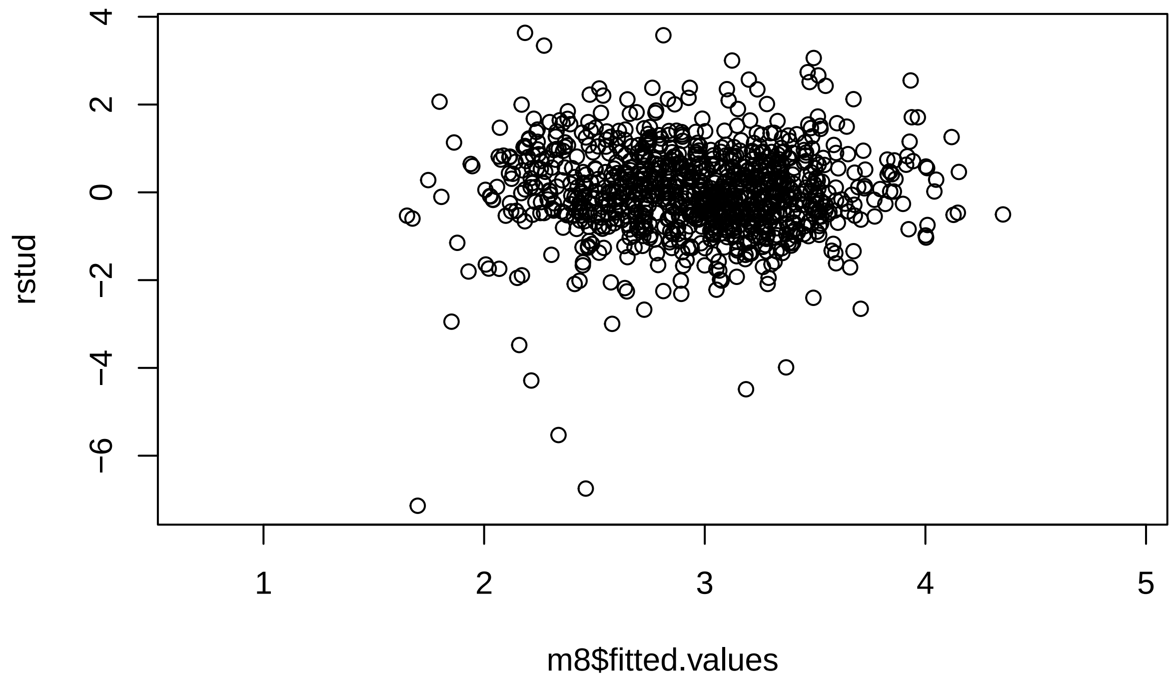
17. Study the presence of a posteriori influential values, indicating the criteria studied in class and the actual atypical observations.

For the a posteriori influential values it is used the Cook's distance and according to the DFBeta

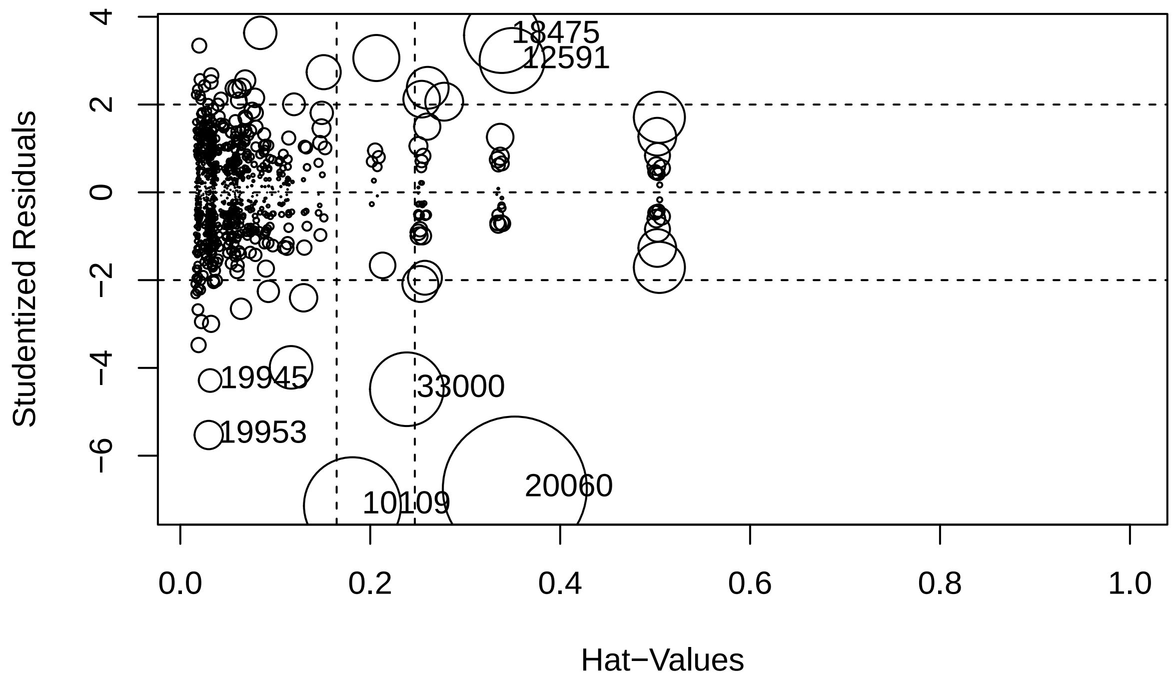
```
dcook <- cooks.distance(m8) #Cook distance
#dcook
leverage <- hatvalues (m8) #Leverage of observations
#leverage

plot(m8$fitted.values, rstan) #Standardized residuals vs fitted values
```




```
plot(m8$fitted.values, rstud) #Studentized residuals vs fitted values
```

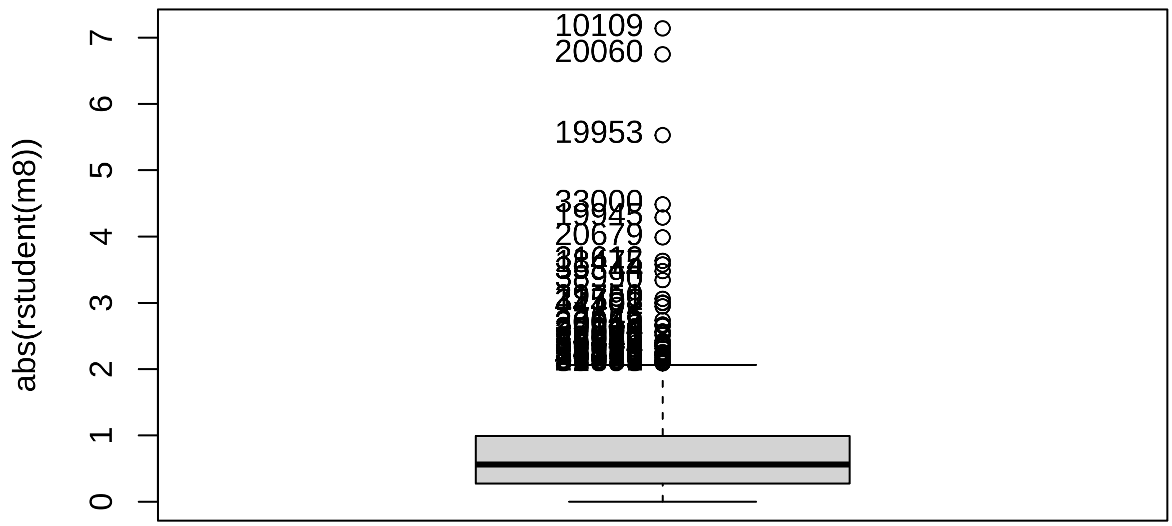



```
influencePlot (m8, id= list (n=5, method = "noteworthy"))
```

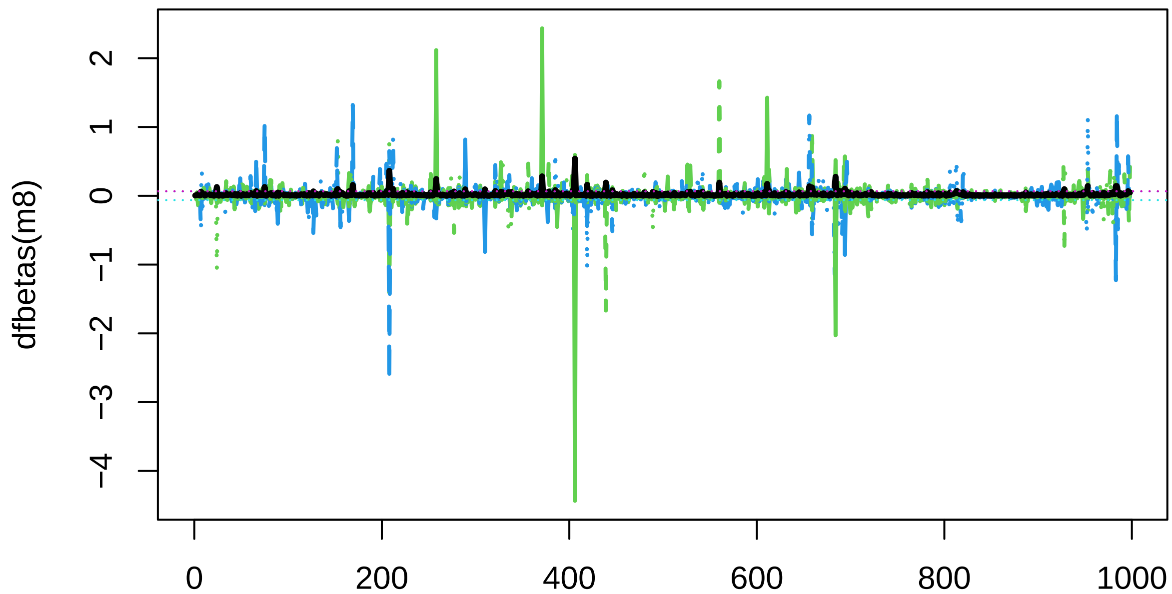



```
##            StudRes        Hat       CookD
## 2256           NaN 1.00000000         NaN
## 9252           NaN 1.00000000         NaN
## 10109 -7.139857 0.18123830 0.132111133
## 12591  3.003374 0.34935900 0.059276500
## 14512       NaN 1.00000000         NaN
## 15053       NaN 1.00000000         NaN
## 16395       NaN 1.00000000         NaN
## 18475  3.578613 0.33844727 0.079857408
## 19945 -4.286945 0.03139741 0.007217814
## 19953 -5.529136 0.02993364 0.011282455
## 20060 -6.748687 0.35225792 0.291616488
## 33000 -4.485505 0.23850832 0.076210344
```

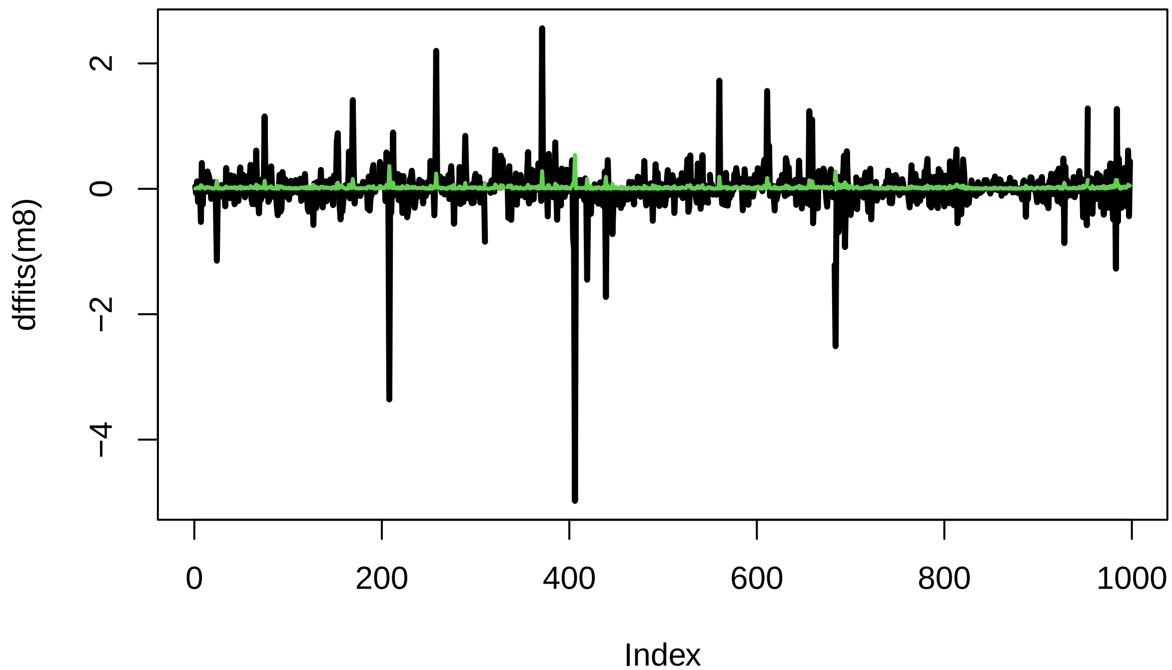
```
llaux<-Boxplot (abs(rstudent (m8)), id=list(n=Inf, labels = row.names (df)))
```




```
# Detection of influential data:
matplot(dfbetas(m8), type="l", col=3:4,lwd=2)
lines(sqrt(cooks.distance(m8)),col=1,lwd=3)
abline(h=2/sqrt(dim(data)[1]), lty=3,lwd=1,col=5)
abline(h=-2/sqrt(dim(data)[1]), lty=3,lwd=1,col=5)
abline(h=sqrt(4/(dim(data)[1]-length(names(coef(m8))))), lty=3,lwd=1,col=6)
```




```
llegenda<-c("Cook d", names(coef(m8)), "DFBETA Cut-off", "Ch-H Cut-off")


# Dffits: another metric for influential data:
par(mfrow=c(1,1))
plot(dffits(m8),type="l",lwd=3)
pp=length(names(coef(m8)))
lines(sqrt(cooks.distance(m8)),col=3,lwd=2)
abline(h=2*(sqrt(pp/(nrow(m8)-pp))),lty=3,lwd=1,col=2)
abline(h=-2*(sqrt(pp/(nrow(m8)-pp))),lty=3,lwd=1,col=2)
```




```
llegenda<-c("DFFITS","DFFITS Cut-off","Cooks D")

# AIC and BIC:
AIC(m8)
```

```
## [1] -737.8895
```

```
AIC(m8, k=log(nrow(data)))
```

```
## [1] -335.6177
```

18. Given a 5-year old car, the rest of numerical variables on the mean and factors on the reference level, what would be the expected price with a 95% confidence interval?

```
newdata = data.frame(quartile_age="4to5Years", mileage=22.402, engineSize=1.937, model="1 Series",transm
predict(m8, newdata, interval='prediction', alpha=0.05)
```

```
##        fit      lwr      upr
## 1 2.823115 2.503228 3.143002
```

£ 2,283. According to the confidence interval a value between £2500 and £3140.

19. Summarize what you have learned by working with this interesting real dataset

The average price of the cars of the subset can be modelled from key variables as the age, the model of the car and engine size however, the market of cars is quiet variable but still linear models could characterize its behavior.