Assignment 1: Car Prices

Àlex Martorell Locascio & Irene Fernández Rebollo

11/10/2021

Contents

Data loading

First, set working directory has to be fixed:

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 requered.

- 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

Data loading and union:

```
# Lecture of DataFrames:
df1 <- read.table("audi.csv",header=T, sep=",")
df1$manufacturer <- "Audi"
df2 <- read.table("bmw.csv",header=T, sep=",")
df2$manufacturer <- "BMW"
df3 <- read.table("merc.csv",header=T, sep=",")
df3$manufacturer <- "Mercedes"
df4 <- read.table("vw.csv",header=T, sep=",")
df4$manufacturer <- "VW"
```

```r
# Union by row:
df <- rbind(df1,df2,df3,df4)
dim(df)  # Size of data.frame
str(df) # Object class and description
names(df) # List of variable names

### Use birthday of 1 member of the group as random seed:
set.seed(130798)
# Random selection of x registers:
sam<-as.vector(sort(sample(1:nrow(df),5000)))
head(df)  # Take a look to the first rows/instances (6 rows)
df<-df[sam,]  # Subset of rows _ It will be my sample
summary(df)

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

Required packages:

```r
# Introduce required packages:

requiredPackages <- c("car","lmtest", "FactoMineR","car", "factoextra","RColorBrewer","ggplot2","dplyr"


#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()
```

Data preparation

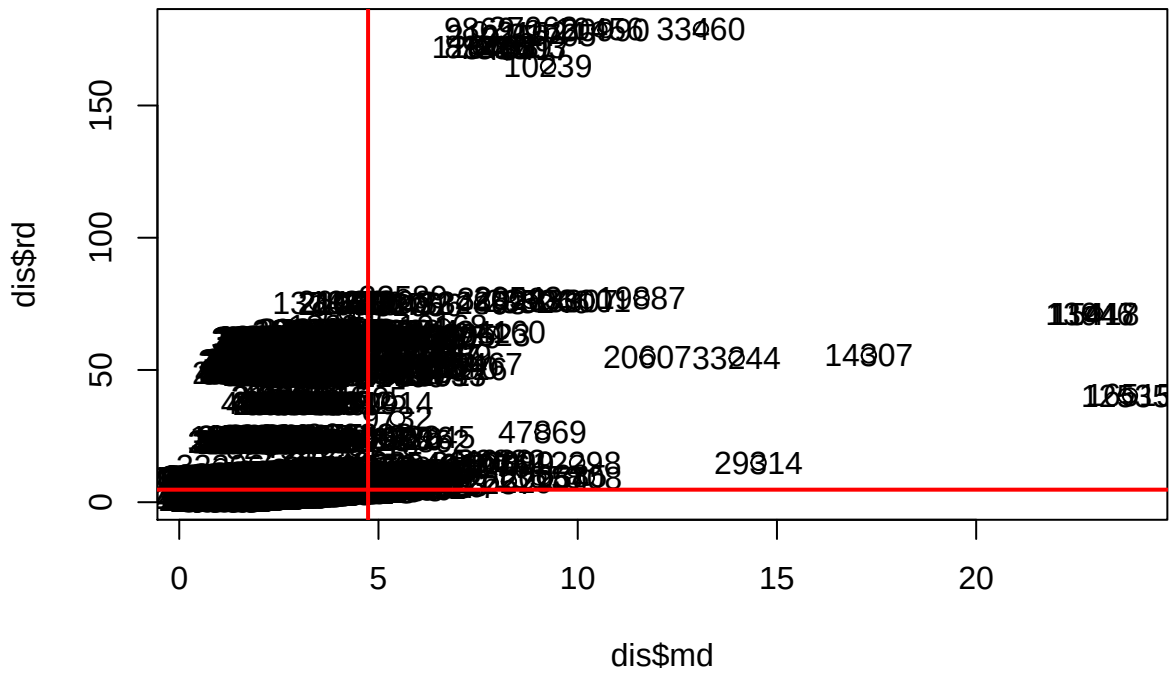
First, we load the raw data:

```
# Clean workspace
rm(list=ls())

load(paste0("MyOldCars-Raw.RData"))

summary(df)
```

```
##     model                year          price        transmission
##  Length:5000        Min.   :1998   Min.   :  1295   Length:5000
##  Class :character   1st Qu.:2016   1st Qu.: 13900   Class :character
##  Mode  :character   Median :2017   Median : 19000   Mode  :character
##                     Mean   :2017   Mean   : 21291
##                     3rd Qu.:2019   3rd Qu.: 25995
##                     Max.   :2020   Max.   :135124
##     mileage         fuelType              tax             mpg
##  Min.   :     1   Length:5000        Min.   :  0.0   Min.   :  1.10
##  1st Qu.:  5946   Class :character   1st Qu.:125.0   1st Qu.: 44.80
##  Median : 17421   Mode  :character   Median :145.0   Median : 53.30
##  Mean   : 23392                      Mean   :122.9   Mean   : 54.18
##  3rd Qu.: 33726                      3rd Qu.:145.0   3rd Qu.: 61.40
##  Max.   :170000                      Max.   :570.0   Max.   :470.80
##    engineSize    manufacturer
##  Min.   :0.000   Length:5000
##  1st Qu.:1.500   Class :character
##  Median :2.000   Mode  :character
##  Mean   :1.907
##  3rd Qu.:2.000
##  Max.   :6.600
```

Data Quality and Profiling

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

dis <- Moutlier(df[, c(2:3, 5, 7:9)], quantile = 0.999, plot=F)
plot(dis$md, dis$rd)
text(dis$md,dis$rd,labels=rownames(df))
abline(h=dis$cutoff, lwd=2, col="red")
abline(v=dis$cutoff, lwd=2, col="red")
```




```
list_mout <- which( ( dis$md > dis$cutoff ) & (dis$rd > dis$cutoff));
length(list_mout)
```

```
## [1] 167
```

```
df <- df[-list_mout, ]


count_na<-colSums(is.na(df))
count_na
```

```
##        model         year        price transmission      mileage     fuelType
##            0            0            0            0            0            0
##          tax          mpg   engineSize manufacturer
##            0            0            0            0
```

No NA's are recorded. We tried it multiple times with both our birthdays and no NA's were retrieved. The `colSums()` function helps us to identify NA's per column.

With respect to univariate outliers, the analysis is left to the plots shown below, as we choose the removal of outliers to be based on multivariate analysis.

Regarding multivariate outliers, as explained in the lectures, Mahalanobis distance (MD from now on) methods are applied. The Moutlier function is executed and a plot with MD and Robust MD on the axis is retrieved. The cutoff provided by the Moutlier function is used (set at 99.9%) and that gives a list of 167 multivariate outliers. They are removed from the data set, which will now have 4833 rows.

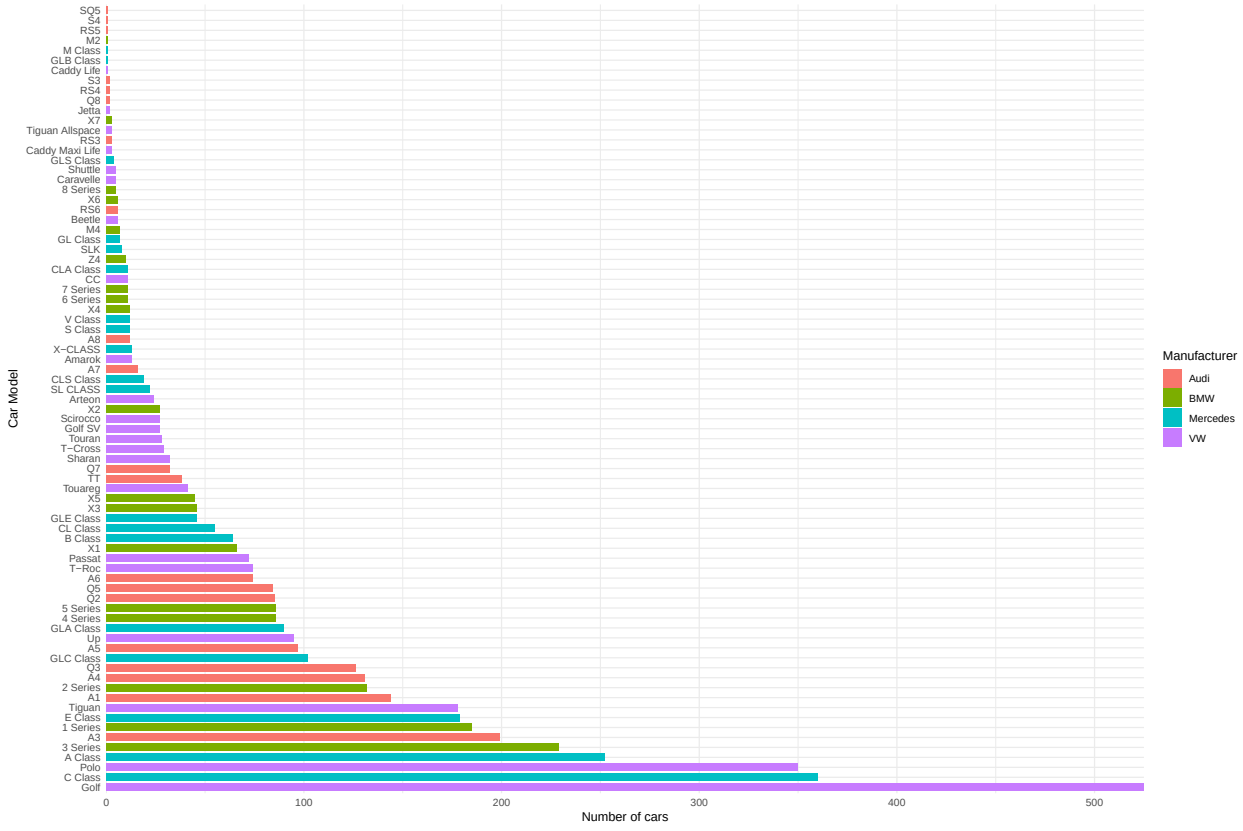
For Data Profiling, we use the `condes()` function. More information is found in Exercise 3.

Univariate Descriptive Analysis

Car Model

The car model is a qualitative variable, it has to be transformed to factor. There are 79 car models and the most popular is "Golf"; in the next plot, we can observe the number of cars for each model.

```
#summary(df$model)
#head(df$model)
#length(unique(df$model)) #79
#nrow(df[is.na(df$model),]) #0
df$model <- as.factor(df$model)
ggplot(df,aes(x = forcats::fct_infreq(model), fill = manufacturer)) +
    geom_bar(stat = 'count', width = 0.8) +
    coord_flip() +
  scale_y_continuous(expand = c(0, 0)) +
  xlab("Car Model") + ylab("Number of cars") +
  scale_fill_discrete(name = "Manufacturer") +
  theme_minimal()
```



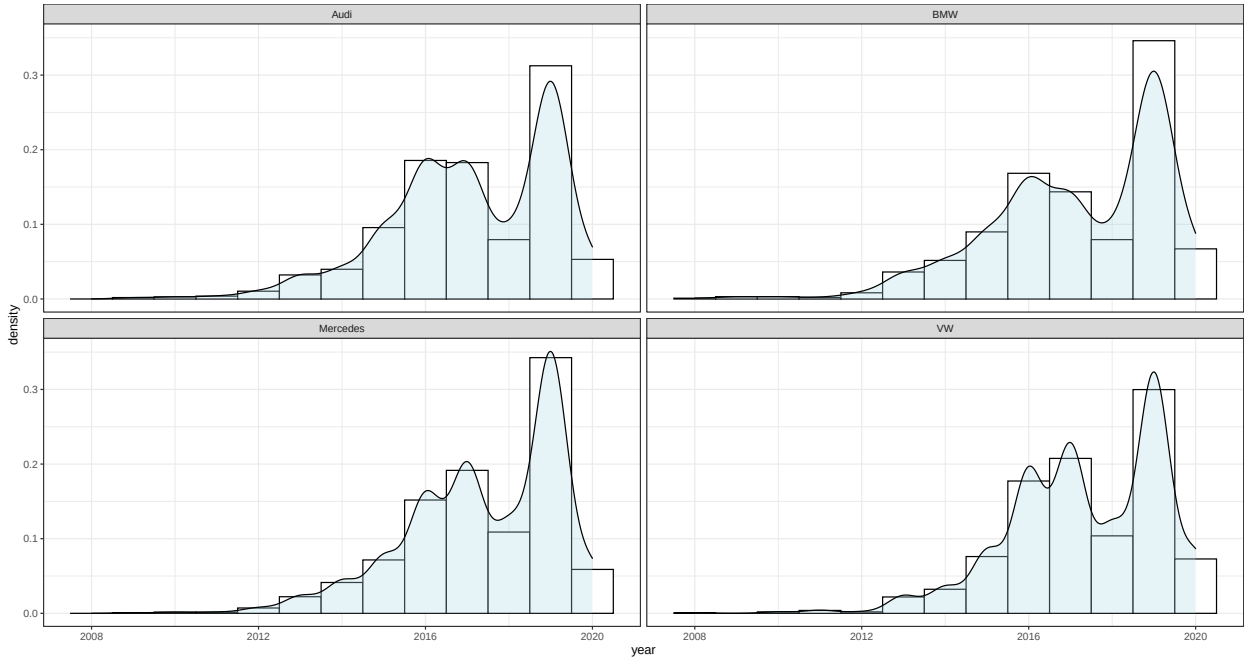
Registration Year

The registration year is a quantitative variable that goes from 1998 to 2020. The manufacturers have a similar distribution for this variable, with 2019 as the year with more registrations.

```
#summary(df$year)
#head(df$year)
table(df$year)
```

```
##
## 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020
##    2    6   11   14   31  131  194  396  825  895  459 1561  308
```

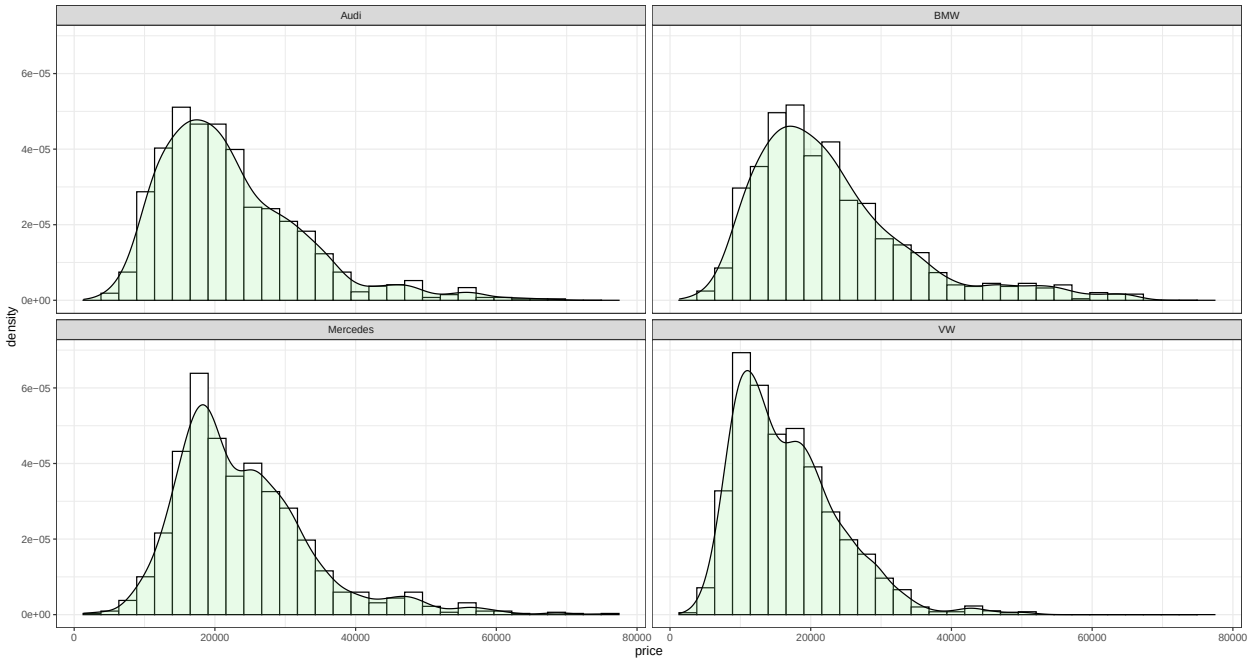
```
#nrow(df[is.na(df$year),]) #0
#boxplot(df$year, notch = TRUE)
ggplot(df,aes(x = year)) +
  geom_histogram(binwidth = 1, aes(y = ..density..), color = "black", fill = "white") +
  geom_density(alpha=.3, fill="lightblue") +
  facet_wrap(manufacturer ~ .) +
  theme_bw()
```



Price (£)

Price is the numeric target of the project, their values goes from 1295£ to 135124£ with a mean of 21291£.

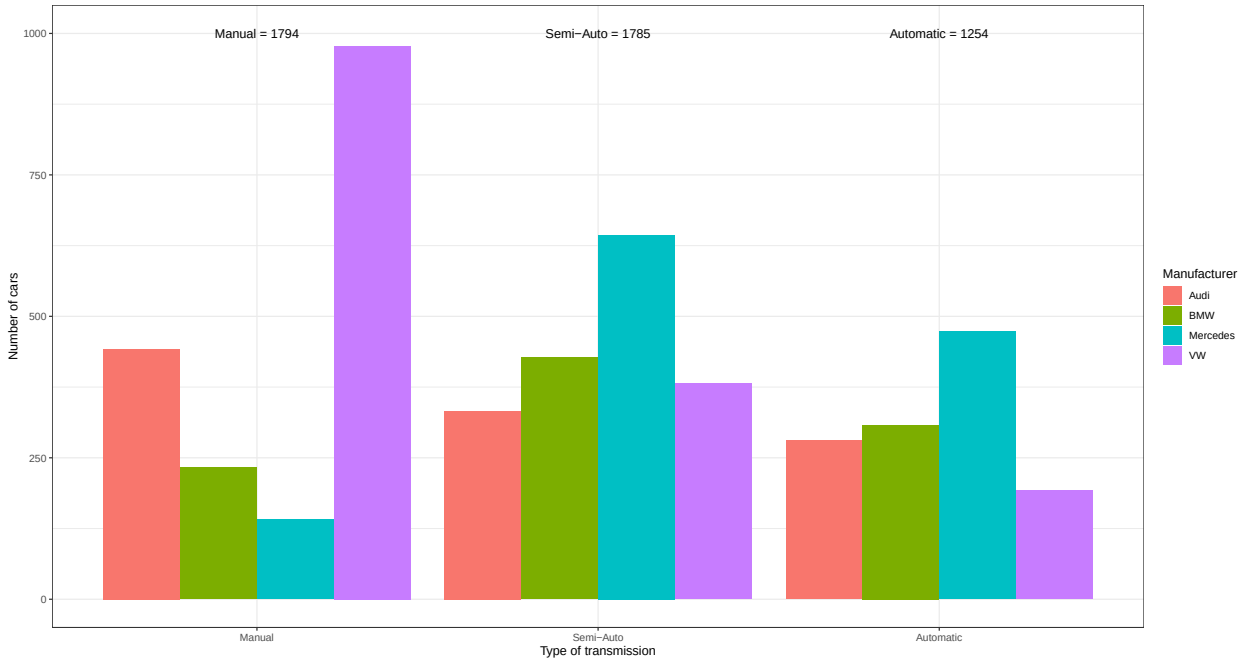
```
#summary(df$price)
#head(df$price)
#nrow(df[is.na(df$price),]) #0
#boxplot(df$price, notch = TRUE)
ggplot(df,aes(x = price)) +
  geom_histogram(aes(y = ..density..), color = "black", fill = "white", bins = 30) +
  geom_density(alpha=.2, fill="lightgreen") +
  facet_wrap(manufacturer ~ .) +
  theme_bw()
```



Type of Gearbox

The type of Gearbox is a qualitative variable with three possible categories: manual, semi-auto and automatic. More or less there are the same number of cars for the different transmissions. Highlight that VW has more manual cars compared with the other types of transmission, and that Mercedes has a lower number of manual cars compared with the rest of types.

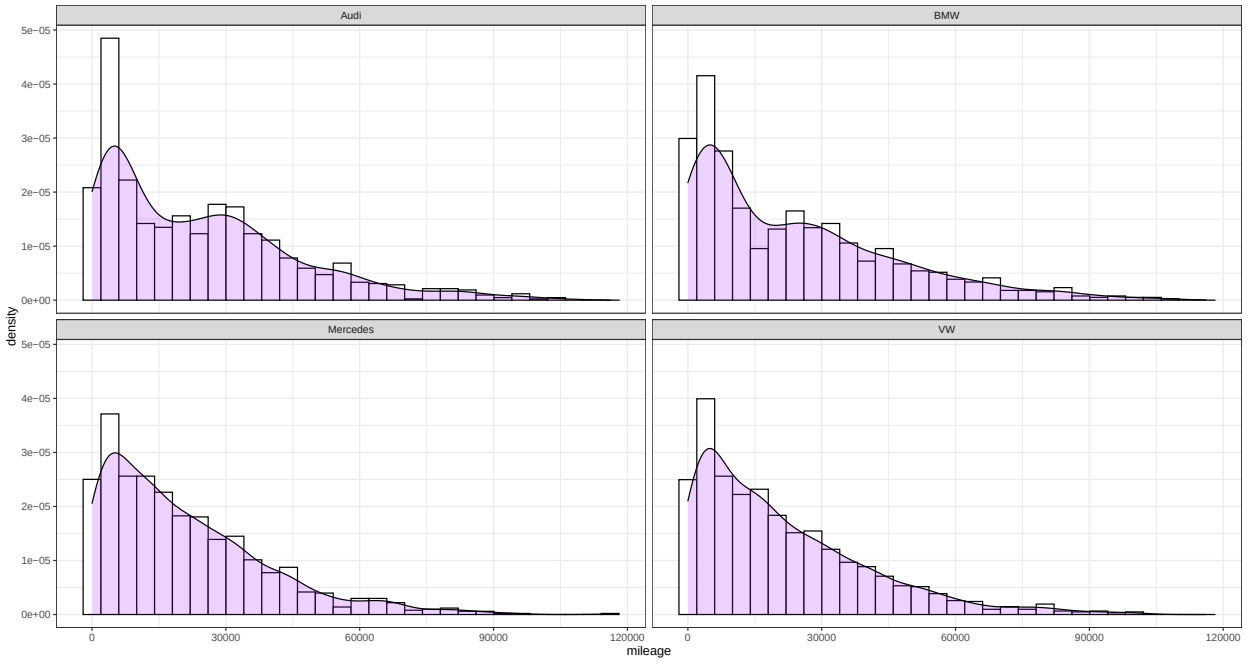
```
#summary(df$transmission)
#head(df$transmission)
#nrow(df[is.na(df$transmission),]) #0
#table(df$transmission)
df$transmission <- factor(df$transmission, levels = c("Manual", "Semi-Auto", "Automatic"))
ggplot(df, aes(x = transmission, fill = manufacturer)) +
  geom_bar(position = position_dodge()) +
  xlab("Type of transmission") + ylab("Number of cars") +
  scale_fill_discrete(name = "Manufacturer") +
  annotate(geom = "text", x = 1, y = 1000,
           label = paste0("Manual = ", nrow(df[df$transmission == "Manual",]))) +
  annotate(geom = "text", x = 2, y = 1000,
           label = paste0("Semi-Auto = ", nrow(df[df$transmission == "Semi-Auto",]))) +
  annotate(geom = "text", x = 3, y = 1000,
           label = paste0("Automatic = ", nrow(df[df$transmission == "Automatic",]))) +
  theme_bw()
```



Distance used

The distance used is a numerical varable that goes from 1 to 170000. Can be seen that as more distance used, less number of cars.

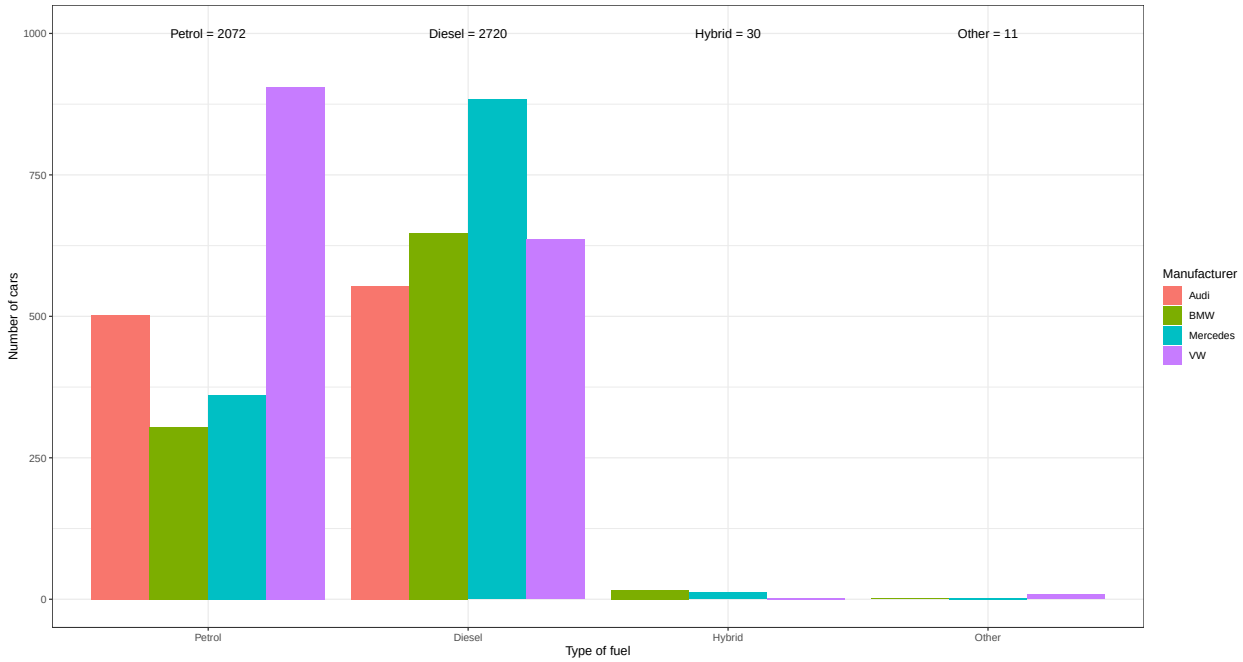
```
#summary(df$mileage)
#head(df$mileage)
#nrow(df[is.na(df$mileage),]) #0
#boxplot(df$mileage, notch = TRUE)
ggplot(df,aes(x = mileage)) +
  geom_histogram(aes(y = ..density..), color = "black", fill = "white", bins = 30) +
  geom_density(alpha=.2, fill="purple") +
  facet_wrap(manufacturer ~ .) +
  theme_bw()
```



Engine Fuel

Engine Fuel is a qualitative variable with 4 different types of fuel: petrol, diesel, hybrid and others. There are very few cars with the category hybrid or other fuel type and any cars for the Audi manufacturer.

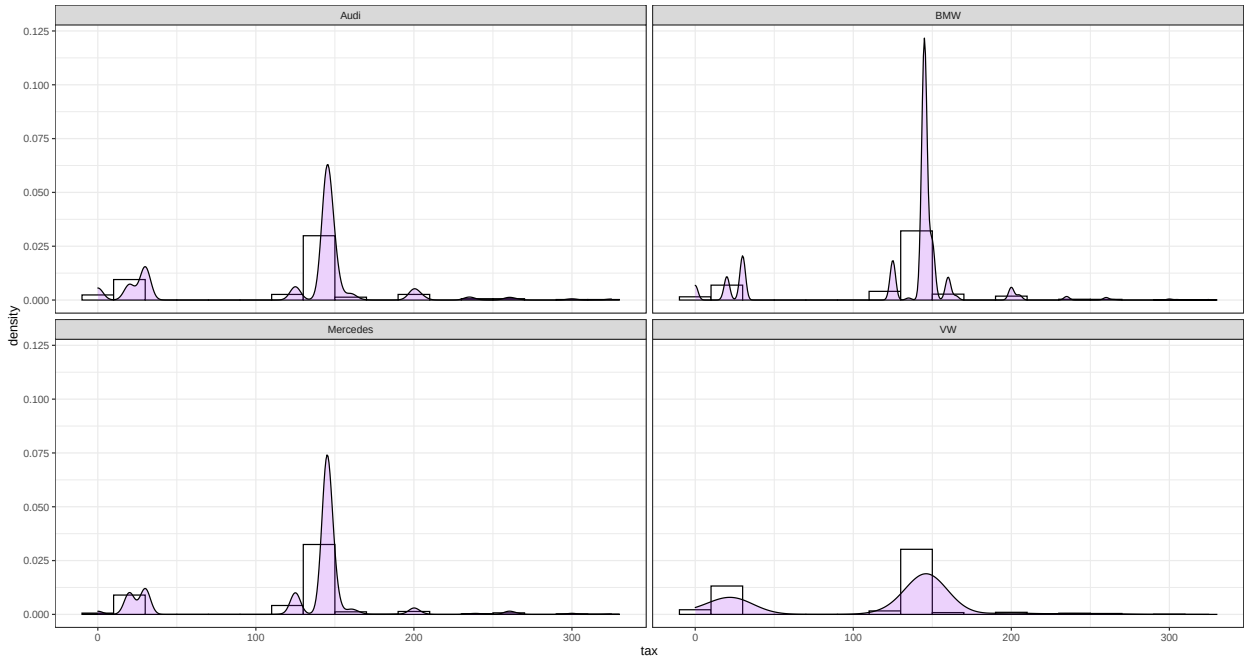
```
#summary(df$fuelType)
#head(df$fuelType)
#nrow(df[is.na(df$fuelType),]) #0
#table(df$fuelType)
df$fuelType <- factor(df$fuelType, levels = c("Petrol", "Diesel", "Hybrid", "Other"))
ggplot(df, aes(x = fuelType, fill = manufacturer)) +
  geom_bar(position = position_dodge()) +
  xlab("Type of fuel") + ylab("Number of cars") +
  scale_fill_discrete(name = "Manufacturer") +
  annotate(geom = "text", x = 1, y = 1000,
           label = paste0("Petrol = ", nrow(df[df$fuelType == "Petrol",]))) +
  annotate(geom = "text", x = 2, y = 1000,
           label = paste0("Diesel = ", nrow(df[df$fuelType == "Diesel",]))) +
  annotate(geom = "text", x = 3, y = 1000,
           label = paste0("Hybrid = ", nrow(df[df$fuelType == "Hybrid",]))) +
  annotate(geom = "text", x = 4, y = 1000,
           label = paste0("Other = ", nrow(df[df$fuelType == "Other",]))) +
  theme_bw()
```



Road Tax

The road tax is a quantitative variable that goes from 0 to 570. This variable has very different values, at this point cannot be seen any clear tendency.

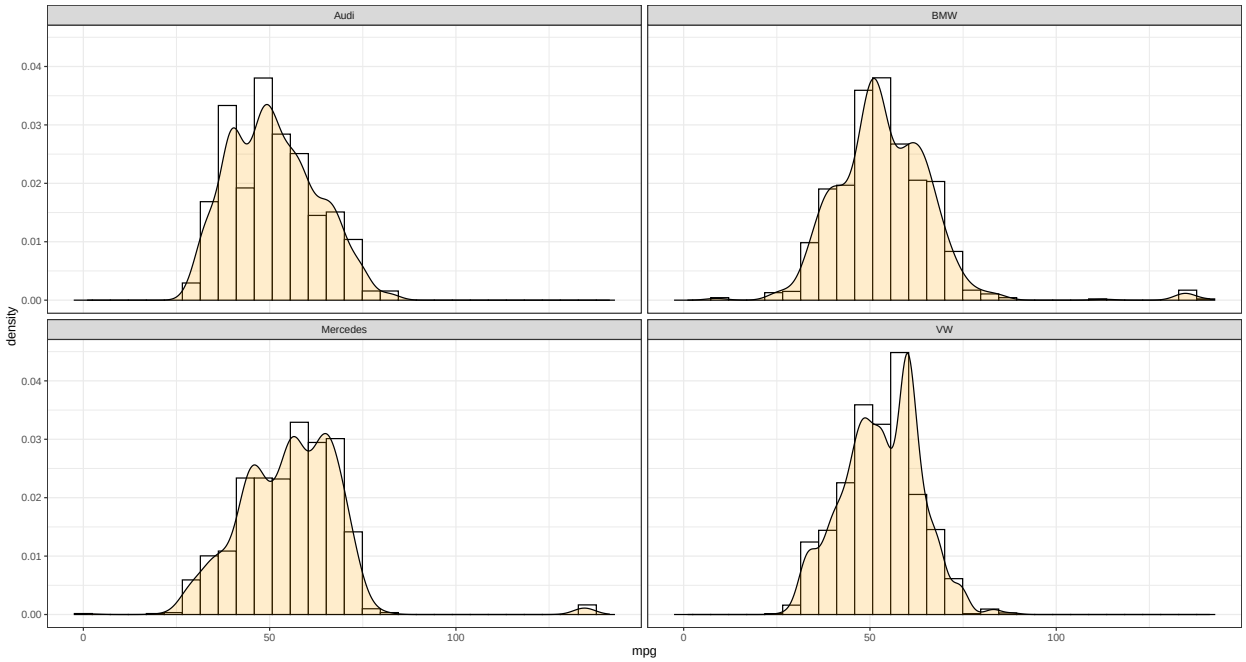
```
#summary(df$tax)
#head(df$tax)
#nrow(df[is.na(df$tax),]) #0
#table(df$tax)
#boxplot(df$tax)
ggplot(df,aes(x = tax)) +
  geom_histogram(binwidth = 20, aes(y = ..density..), color = "black", fill = "white") +
  geom_density(alpha=.2, fill="purple") +
  facet_wrap(manufacturer ~ .) +
  theme_bw()
```



Consumption in miles per gallon (mpg)

The consumption is a numerical value that goes from 1.10mpg to 470.80mpg. This variable seems to follow a normal distribution, similar between manufacturers.

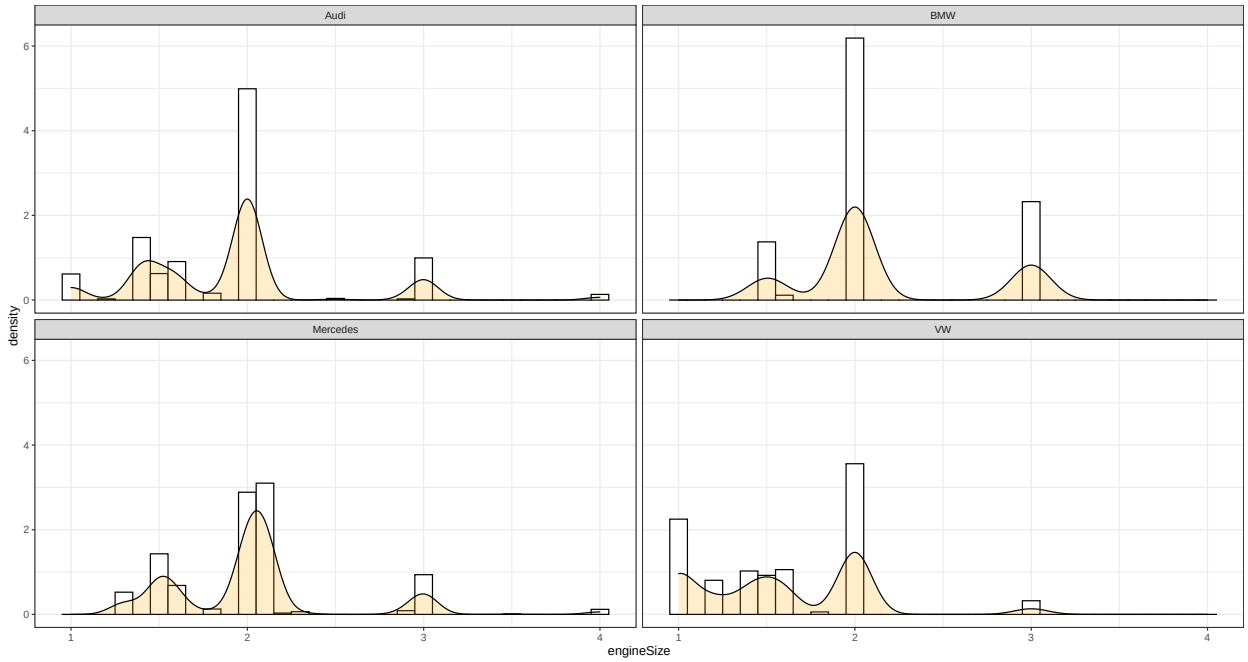
```
#summary(df$mpg)
#head(df$mpg)
#nrow(df[is.na(df$mpg),]) #0
#boxplot(df$mpg)
ggplot(df,aes(x = mpg)) +
  geom_histogram(aes(y = ..density..), color = "black", fill = "white", bins = 30) +
  geom_density(alpha=.2, fill="orange") +
  facet_wrap(manufacturer ~ .) +
  theme_bw()
```



Size in litres

The size in litres is the last quantitative variable that goes from 0 to 6.6. This variable has very different values, at this point cannot be seen any clear pattern.

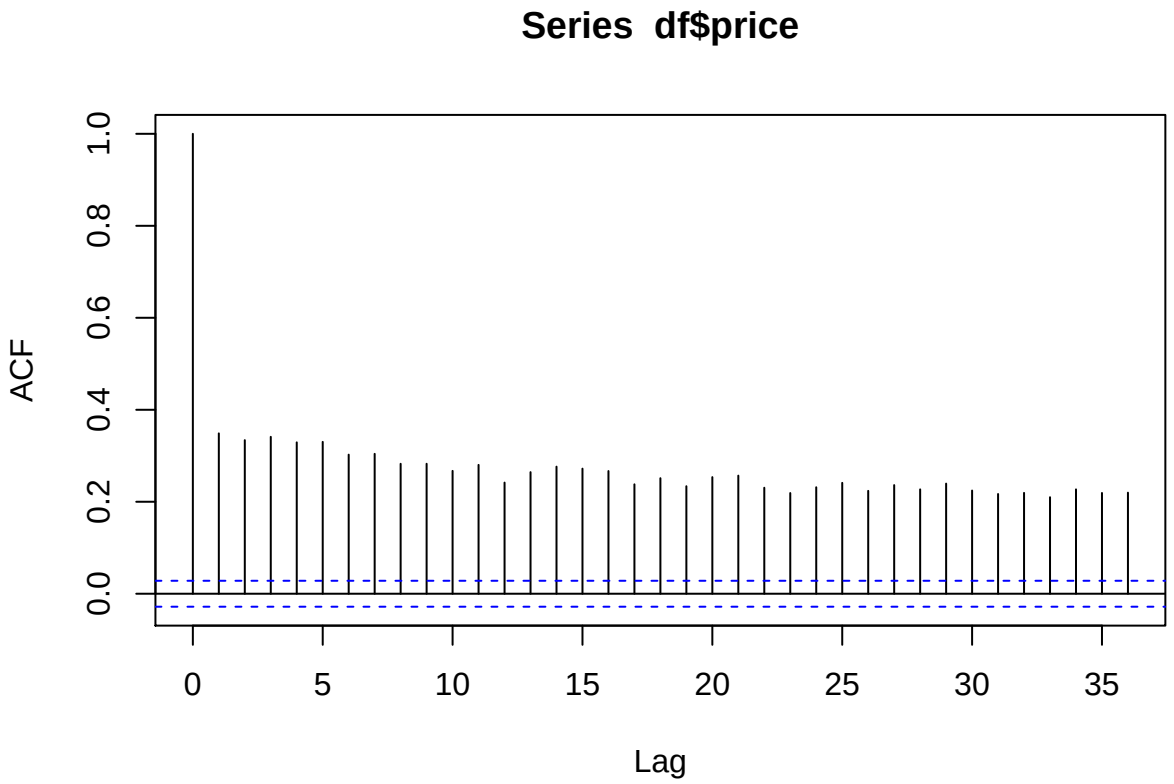
```
#summary(df$engineSize)
#head(df$engineSize)
#nrow(df[is.na(df$engineSize),]) #0
#table(df$engineSize)
#boxplot(df$engineSize)
ggplot(df,aes(x = engineSize)) +
  geom_histogram(binwidth = 0.1, aes(y = ..density..), color = "black", fill = "white") +
  geom_density(alpha=.2, fill="orange") +
  facet_wrap(manufacturer ~ .) +
  theme_bw()
```



Questions

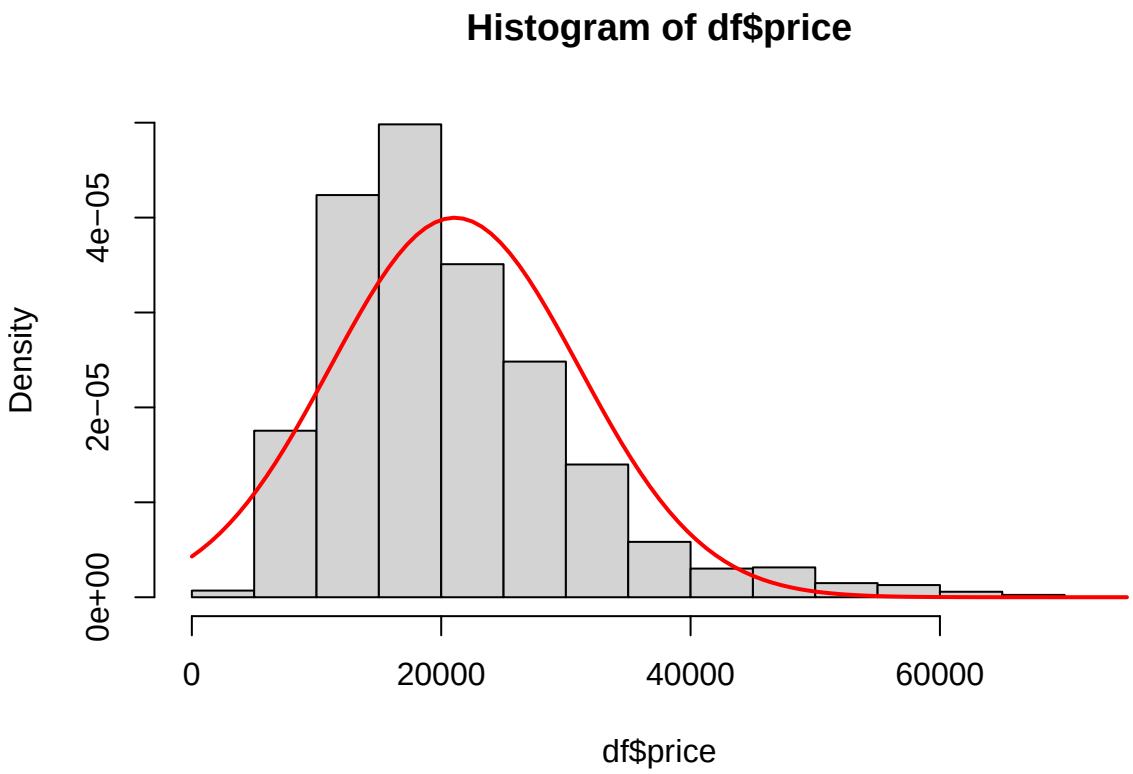
1. Determine if the response variable (price) has an acceptably normal distribution. Address test to discard serial correlation.

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



If we plot the ACF function, we can see that some autocorrelation is shown in the first 30 Lags, and it does seem significant.

```
mm <- mean(df$price)
ss <- sd(df$price)
hist(df$price, freq = F)
curve(dnorm(x, mm, ss), lwd=2, add=T, col="red")
```




```
shapiro.test(df$price)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  df$price
## W = 0.91302, p-value < 2.2e-16
```

Also, we want to assess if the variable is normal. We proceed graphically and The histogram of densities clearly does not seem to follow a normal distribution. The Shapiro-Wilk test returns a p-value of 0, which means we reject the null hypothesis of normality.

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

```
#names(df)
#Quantitative variables:
condes(df, 3)$quanti
```

```
##             correlation       p.value
## year          0.6069324  0.000000e+00
## engineSize    0.6004758  0.000000e+00
## tax           0.4396818 1.130472e-227
## mileage      -0.5409643  0.000000e+00
## mpg          -0.5706820  0.000000e+00
```

```
#Qualitative variables
condes(df, 3)$quali
```

```
##                         R2       p.value
## model         0.519108876  0.000000e+00
## transmission 0.243415984 2.754523e-293
## manufacturer 0.095451265 1.095643e-104
## fuelType     0.009369543  7.374765e-10
```

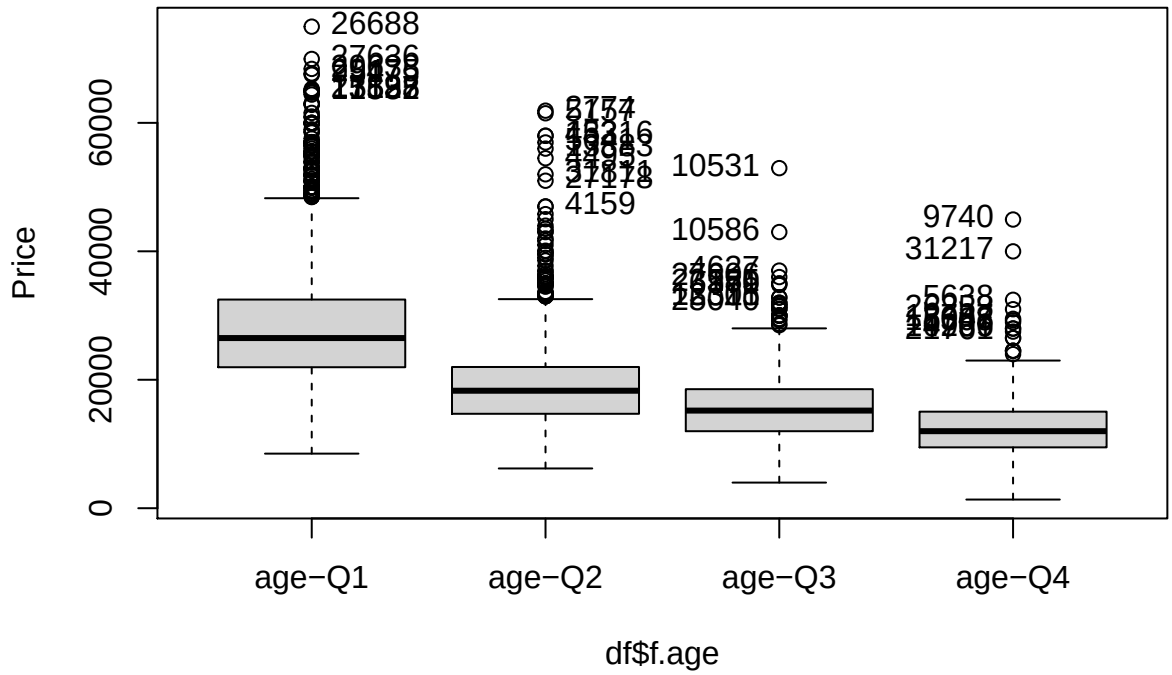
The most associated variables by exploration are year and engineSize for the quantitative variables and model for the qualitative variables.
If we order qualitative variables by highest *correlation* absolute value: $year > engineSize > mpg > mileage > tax$.
If we order qualitative variables by highest R^2 absolute value: $model > transmission > manufacturer > fuelType$

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.

```
maxyear<-max(df$year)

q1<-quantile(maxyear - df$year)[2]
q2<-quantile(maxyear - df$year)[3]
q3<-quantile(maxyear - df$year)[4]

df$f.age <- 0
df$f.age[maxyear - df$year > q3 ] <-3
df$f.age[maxyear - df$year <= q3  ] <-2
df$f.age[maxyear - df$year <= q2 ] <-1
df$f.age[maxyear - df$year <= q1 ] <-0
df$f.age<-factor(df$f.age, labels=c("age-Q1", "age-Q2", "age-Q3", "age-Q4"))

table(df$f.age)
```

```
##
## age-Q1 age-Q2 age-Q3 age-Q4
##   1869   1354    825    785
```

```
df$age<-maxyear - df$year

Boxplot(df$price ~ df$f.age, id=list(labels=row.names(df)) , ylab="Price")
```




```
##  [1] "26688" "27636" "29638" "29475" "9517"  "2783"  "13127" "15195" "17538"
## [10] "11582" "2774"  "5157"  "452"   "13316" "5641"  "13383" "4495"  "31811"
```

```
## [19] "27178" "4159"  "10531" "10586" "4627"  "27606" "551"   "23185" "6370"
## [28] "12301" "13315" "25040" "9740"  "31217" "5638"  "26253" "13787" "3043"
## [37] "15308" "10906" "4239"  "21761"
```

To create the factor that represents the car age based on the year quartile, the maximum in the registration year variable is computed and the actual registration year is subtracted. This will give us the age of each car.

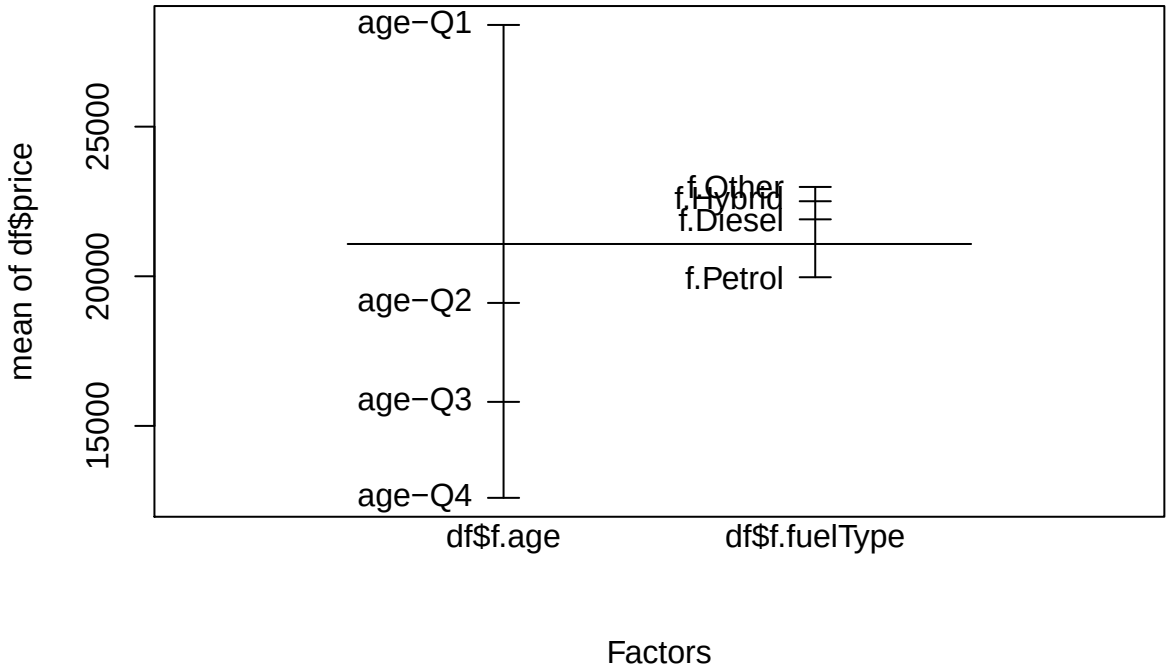
Once the factor is created, the boxplot clearly shows that the older the car, the lower the price. A significant number of outliers is also detected.

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

```
# We create a factor for Fuel Type

df$f.fuelType <- 0
df$f.fuelType[df$fuelType == "Petrol" ] <-1
df$f.fuelType[df$fuelType == "Hybrid" ] <-2
df$f.fuelType[df$fuelType == "Other" ] <-3
df$f.fuelType <-factor(df$f.fuelType, labels=c("f.Diesel", "f.Petrol", "f.Hybrid", "f.Other"))

table(df$f.fuelType)
```

```
##
## f.Diesel f.Petrol f.Hybrid  f.Other
##     2720     2072       30       11
```

```
plot.design(df$price ~ df$f.age + df$f.fuelType)
```




```
r1 <- with(df, tapply(price, f.age, mean))
r1
```

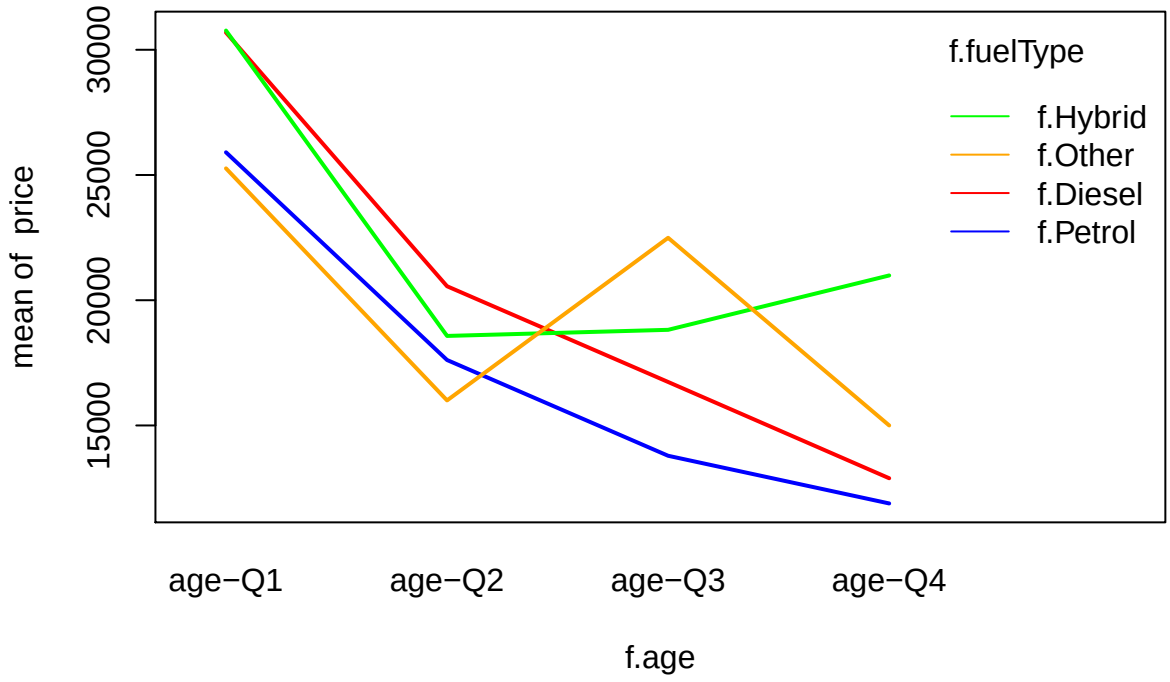
```
##   age-Q1   age-Q2   age-Q3   age-Q4
## 28398.70 19109.83 15802.93 12595.97
```

```
r2 <- with(df, tapply(price, f.fuelType, mean))
r2
```

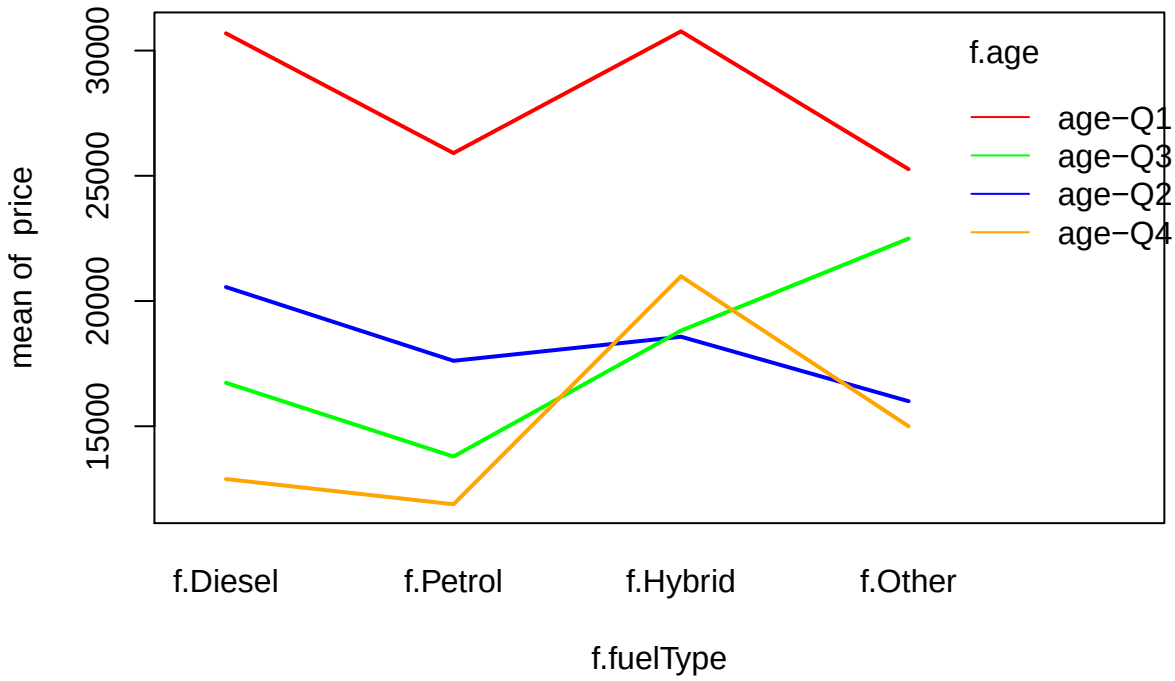
```
## f.Diesel f.Petrol f.Hybrid  f.Other
```

```
## 21902.94 19967.71 22507.47 22985.27
```

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

m0 <- lm(price ~ 1, data=df)
m1 <- lm(price ~ f.age*f.fuelType, data=df)
m2 <- lm(price ~ f.age + f.fuelType, data=df)
m3 <- lm(price ~ f.age, data=df)
m4 <- lm(price ~ f.fuelType, data=df)

#Interaction test
with(df, interaction.plot(f.age, f.fuelType, price,
                          col=c("red", "blue", "green", "orange"), lty = 1, lwd= 2))
```




```
with(df, interaction.plot(f.fuelType, f.age, price,
                          col=c("red", "blue", "green", "orange"), lty = 1, lwd= 2))
```




```
anova(m2,m1)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ f.age + f.fuelType
## Model 2: price ~ f.age * f.fuelType
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)
## 1   4826 2.8269e+11
## 2   4817 2.8040e+11  9 2291567135 4.3741 1.045e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We use the plot.design() function as a first approximation to our problem. It shows how the mean of the price is spread along both factors. It is noted that the Car Age factor seems to explain better the variation in price than the fuel type but we have to keep in mind that this factor is constructed using quartiles.

The First Interaction Plot shows some interaction between factors f.age and f.fuelType. Plus, both factors are significant (price varies according to fuel type and resgistration year). However, we must assess if this interaction is significant enough.

With regards to The Second Interaction Plot interaction between factors is also shown.

A low p-value indicates that there is an interaction between factors. In our case, the p-value < 0.05 (1.045e-05), so we have to reject the null hypothesis and conclude that there exists interaction. Hence, from now on the interaction model will be considered. (represented by m1)

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?

```
anova (m0, m3)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ 1
## Model 2: price ~ f.age
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)
## 1   4832 4.8098e+11
## 2   4829 2.9614e+11  3 1.8484e+11 1004.7 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova (m0, m4)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ 1
## Model 2: price ~ f.fuelType
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)
## 1   4832 4.8098e+11
## 2   4829 4.7648e+11  3 4506584290 15.225 7.375e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(m4, m1)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ f.fuelType
## Model 2: price ~ f.age * f.fuelType
##   Res.Df        RSS Df  Sum of Sq     F    Pr(>F)
## 1   4829 4.7648e+11
## 2   4817 2.8040e+11 12 1.9608e+11 280.7 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Using One-way ANOVA on the null model (no difference between group means) and on the age and the fuelType model (respectively). This is to assess whether the price variable depends on both factors or not. H_0 is the null model. In both cases, the p-value is a lot less than 0.05, even closer to 0 in the age factor. So it seems that the price has a stronger dependence from age, which leads to the second question: Whether the Fuel Type influences the relation between price and age factor. There is a specific two-way ANOVA test that can be done on the model with factor fuelType and the additive model. The p-value is almost zero, so the conclusion is that the models are not equivalent (H_0 rejected) so the interaction model must be considered.

Note. Questions 6 and 7 are joined.

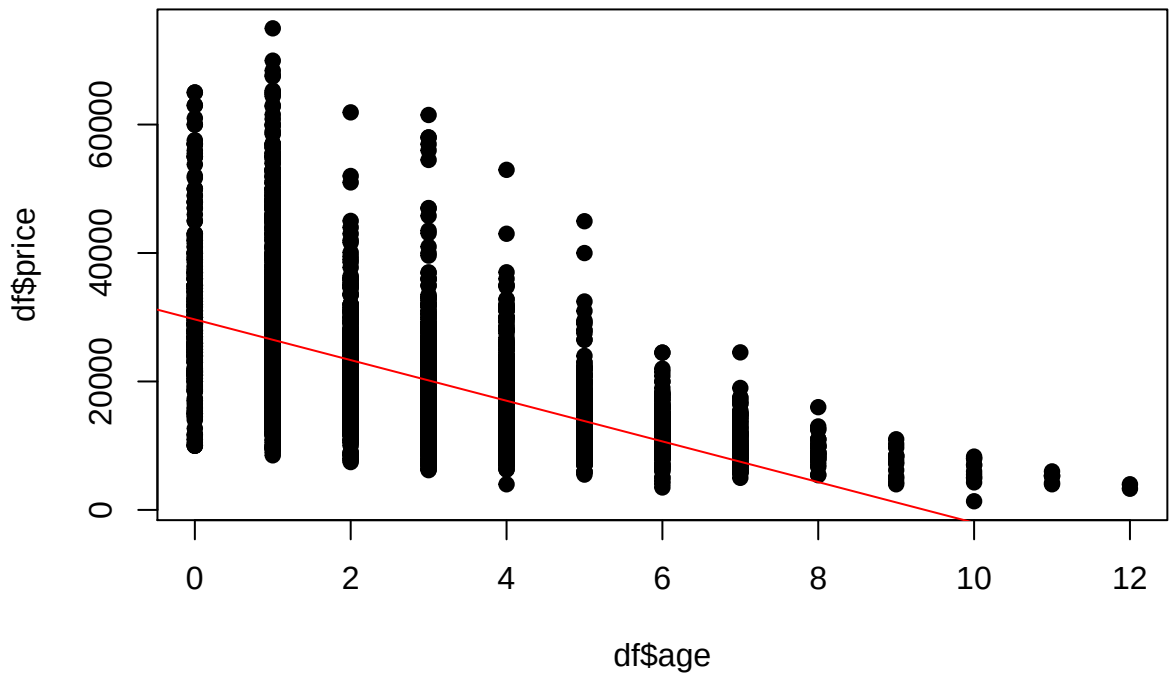
6. Calculate the linear regression model that explains the price from the age: interpret the regression line and assess its quality.

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

```
lm1 <- lm(price ~ df$age , data = df)
summary(lm1)
```

```
##
## Call:
## lm(formula = price ~ df$age, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -19668  -4959  -1007   3302  48487
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 29658.36     197.82  149.92   <2e-16 ***
## df$age      -3165.26      59.63  -53.08   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7930 on 4831 degrees of freedom
## Multiple R-squared:  0.3684, Adjusted R-squared:  0.3682
## F-statistic:  2817 on 1 and 4831 DF,  p-value: < 2.2e-16
```

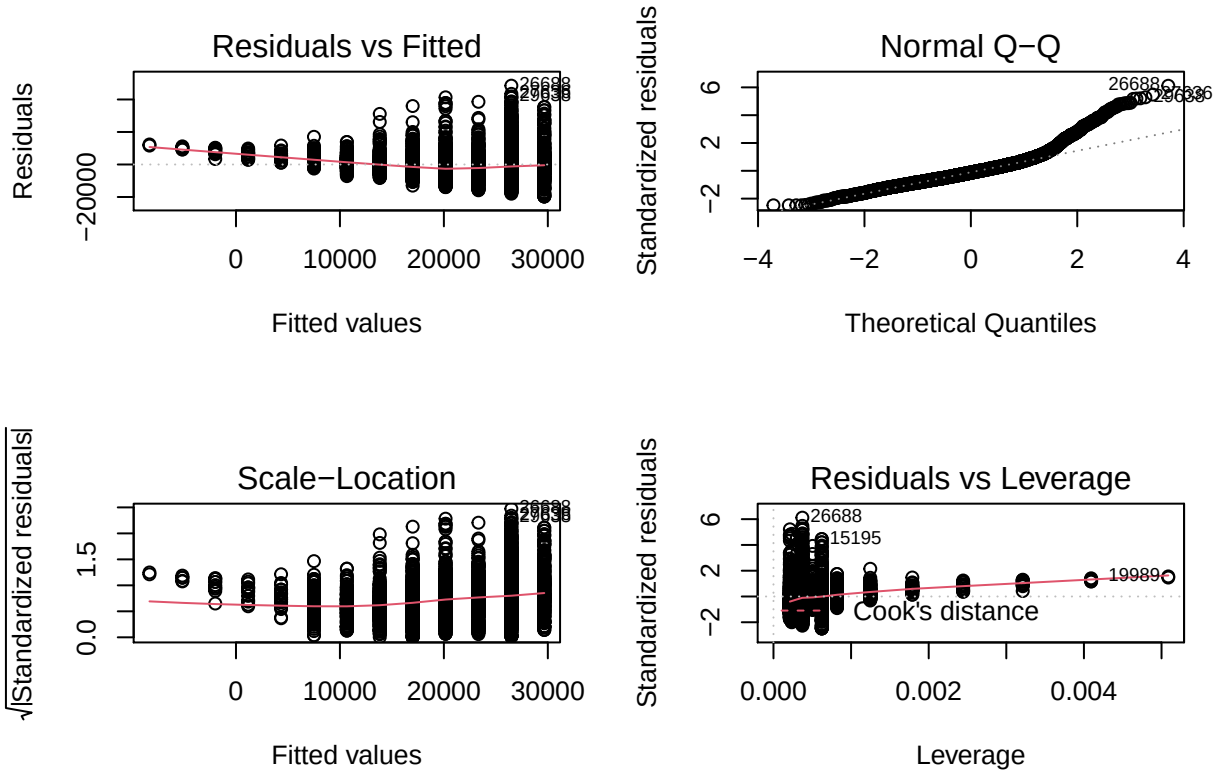
```
plot(df$price ~ df$age ,pch=19,col="black")
abline(lm1,col="red")
```



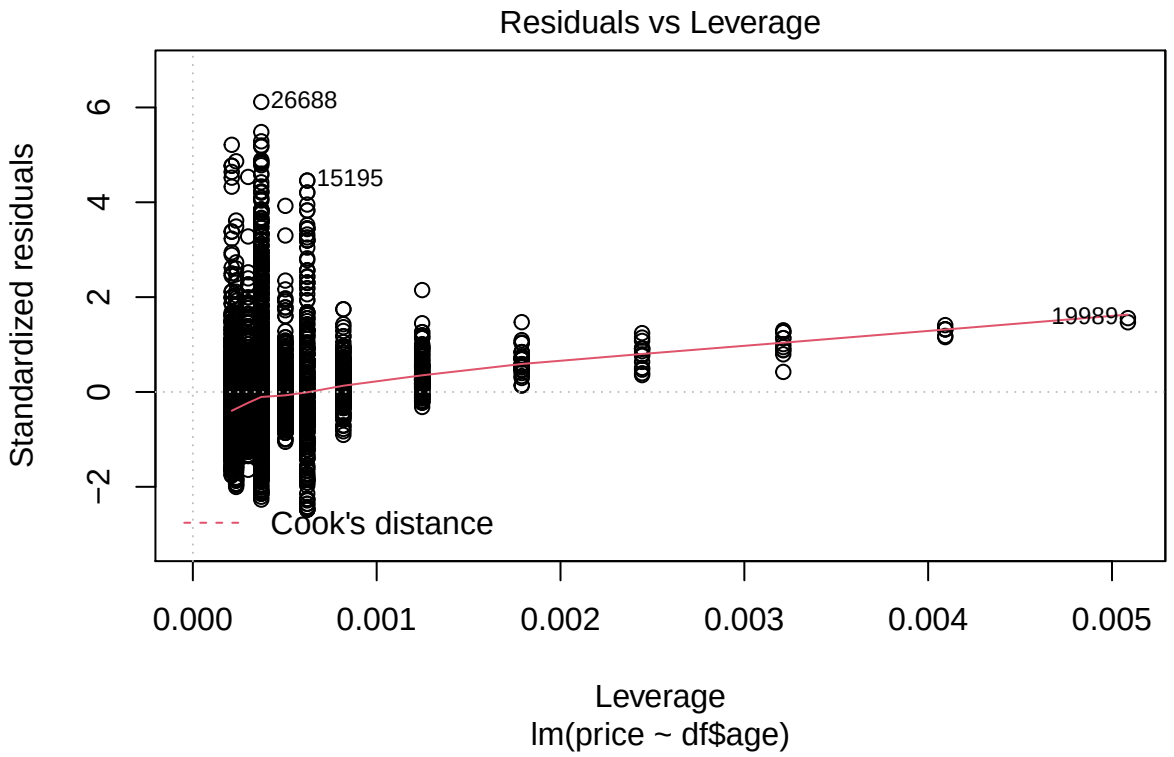

```
par(mfrow=c(1,1))
summary(lm1)
```

```
##
## Call:
## lm(formula = price ~ df$age, data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -19668  -4959  -1007   3302  48487
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 29658.36     197.82  149.92   <2e-16 ***
## df$age      -3165.26      59.63  -53.08   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7930 on 4831 degrees of freedom
## Multiple R-squared:  0.3684, Adjusted R-squared:  0.3682
## F-statistic:  2817 on 1 and 4831 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2,2))
plot(lm1)
```




```
par(mfrow=c(1,1))
plot(lm1, which=5)
```




```
bptest(lm1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  lm1
## BP = 175.56, df = 1, p-value < 2.2e-16
```

The first thing to notice is that the coefficients are well-estimated. The R^2 is approximately 0.37, meaning that the Car Registration Year explains 37% of the price variable. The p-value for both parameters is zero, which means that the null hypothesis for $\beta_1 = 0$ and $\beta_2 = 0$ is rejected.

With regards to **Model diagnostics**:

- The residuals vs fitted values plot deviates notably from the mean 0 line. Some of the residuals are extremly large, and the spread of the residuals is not the same, as the fitted value for the price is bigger, the residuals increase. This contradicts the constant variance hypothesis.
- In the normal QQ Plot, the normal assumption seems to fit quite well until the quantiles are large: This means that the residuals for the fitted price variable deviate from the normal assumption, as they are a lot larger than expected.
- In the Scale-Location Plot, recall that it is an easier way of checking for homoskedasticity (constant variance). The difference between the Residuals v. Fitted Plot is that in this case the residuals are standardized. The red line indicates the average magnitude of the std. residuals, which we want it to be constant. The spread of the different magnitudes in residuals gets larger as the fitted price is bigger.
- In the Residuals v. Leverage Plot, a further study into unsual observations is performed. As the leverage increases, the spread of the residuals decreases, indicating heteroskedasticity.

The plots have definitely pointed out the heteroskedasticity of the model. The Breusch-Pagan Test returns a p-value to 0, therefore rejecting the H_0 for homoskedasticity.

Observation. Looking at the plot of the regression line, it is easy to infer the conclusions from the more detailed study given above: If the car is less old, the price can be a lot higher, and the regression models fails to incorporate these values. This is why the residuals increase as the fitted value is higher.

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

```
lm11 <- lm(price ~ age + I(age^2) , data = df)
summary(lm11)
```

```
##
## Call:
## lm(formula = price ~ age + I(age^2), data = df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -21940  -4728   -997   2961  47941
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 31929.54     260.07  122.77   <2e-16 ***
## age         -5180.86     164.14  -31.56   <2e-16 ***
## I(age^2)      290.03      22.06   13.15   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7793 on 4830 degrees of freedom
## Multiple R-squared:  0.3902, Adjusted R-squared:  0.3899
## F-statistic:  1545 on 2 and 4830 DF,  p-value: < 2.2e-16
```

It does not seem extremely necessary because the Multiple R-square coefficient does not increase substantially (from 0.37 to 0.39).

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

```
round(cor(df[, c(2:3, 5, 7:9)]), 2)
```

```
##             year price mileage   tax   mpg engineSize
## year        1.00  0.61   -0.79  0.37 -0.37       0.00
## price       0.61  1.00   -0.54  0.44 -0.57       0.60
## mileage    -0.79 -0.54    1.00 -0.34  0.38       0.05
## tax         0.37  0.44   -0.34  1.00 -0.61       0.32
## mpg        -0.37 -0.57    0.38 -0.61  1.00      -0.30
## engineSize  0.00  0.60    0.05  0.32 -0.30       1.00
```

```
dfnum <- df[, c(2, 3, 5, 7:9)]

m5 <- lm(price ~  1, data=dfnum)
m5_forw_aic = step(m5,
scope = price ~ tax + mileage + mpg + year + engineSize,
direction = "forward", trace=0)

summary(m5_forw_aic)
```

```
##
## Call:
## lm(formula = price ~ year + engineSize + mpg + mileage + tax,
##     data = dfnum)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -15074  -2709   -366   2412  26203
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.263e+06  1.167e+05  -36.53   <2e-16 ***
## year         2.121e+03  5.780e+01   36.69   <2e-16 ***
## engineSize   1.059e+04  1.355e+02   78.19   <2e-16 ***
## mpg         -1.856e+02  7.173e+00  -25.87   <2e-16 ***
## mileage     -9.586e-02  5.533e-03  -17.32   <2e-16 ***
## tax         -1.605e+01  1.487e+00  -10.80   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4643 on 4827 degrees of freedom
## Multiple R-squared:  0.7836, Adjusted R-squared:  0.7834
## F-statistic:  3496 on 5 and 4827 DF,  p-value: < 2.2e-16
```

```
vif(m5_forw_aic)
```

```
##       year engineSize        mpg    mileage        tax
##   2.740577   1.193255   1.746089   2.772028   1.716757
```

First, the correlation between numeric factors is revised using the cor function. All variables have a correlation of at least 0.44 with the price variable either negative or positive. This implies that the price is dependent on all of them.

Collinearity is observed when a regressor is a function of the others. This can be seen using the correlation table, as the correlation between co-linear variables would be close to 1.

It is noted that no regressors are extremely correlated except ageg and mileage (0.79).

The `step()` function is applied in order to search for the best model using explanatory numeric variables. This method is based on calculating the AIC and finding its lowest value. The AIC decreases as more variables are included in the regression. The lowest value for AIC is when all the numerical variables are included.

m5 is the result of the linear regression between the price and all numeric variables.The results are quite surprising: All parameters seem to be different from zero (as is shown with the low p-values) and the F-statistic test returns 0 p-value which assesses that globally the model is good.

It is noted that no regressors are extremely correlated except year and mileage (0.79)

m5 is the result of the linear regression between the price and all numeric variables. The results are quite surprising: All paramaters seem to be different from zero (as is shown with the low p-values) and the F-statistic test returns 0 p-value which assesses that globally the model is good.

The `vif()` function measures the *Variance Inflation Factor* (VIF from now on). This measure calculates the effect of the collinearity on the variance of the estimated β_j parameters. It depends on R_j^2 which we define as the coefficient that states the variation of a regressor x_j explained by the other regressors. Since VIF is defined as:

$$\frac{1}{1 - R_j^2}$$

if R_j^2 is large then the VIF is large. This means that β_j varies significantly. A common threshold is if VIF is larger than 5. In the model obtained by the step function, all regressors show a VIF smaller than 3, so there is no cause for concern.

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

```
m6 <- lm(price ~  1, data=df)
m6_forw_aic = step(m6,
scope = price ~ model + transmission + mileage + fuelType + tax + mpg +
                engineSize + manufacturer + f.age, direction = "both", trace=0)

vif(m6_forw_aic)
```

```
##                   GVIF Df GVIF^(1/(2*Df))
## model        8.792462 78        1.014033
## f.age        4.147875  3        1.267567
## engineSize   3.572092  1        1.889998
## mileage      2.662095  1        1.631593
## fuelType     3.837405  3        1.251237
## tax          2.109481  1        1.452405
## mpg          4.820870  1        2.195648
## transmission 1.793149  2        1.157189
```

```
summary(m6_forw_aic)
```

```
##
## Call:
## lm(formula = price ~ model + f.age + engineSize + mileage + fuelType +
##     tax + mpg + transmission, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17842.7  -2074.6   -123.9   1812.5  24200.9
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             2.084e+04  6.867e+02  30.348  < 2e-16 ***
## model 2 Series          3.761e+02  3.859e+02   0.975 0.329816
## model 3 Series          2.323e+03  3.406e+02   6.819 1.03e-11 ***
## model 4 Series          2.282e+03  4.458e+02   5.118 3.21e-07 ***
## model 5 Series          3.537e+03  4.490e+02   7.878 4.10e-15 ***
## model 6 Series          2.450e+03  1.056e+03   2.319 0.020444 *
## model 7 Series          1.669e+04  1.062e+03  15.718  < 2e-16 ***
## model 8 Series          2.498e+04  1.540e+03  16.219  < 2e-16 ***
## model A Class           2.500e+03  3.312e+02   7.550 5.19e-14 ***
## model A1                1.015e+03  3.856e+02   2.633 0.008493 **
## model A3                2.228e+03  3.470e+02   6.420 1.49e-10 ***
## model A4                2.994e+03  3.870e+02   7.737 1.23e-14 ***
## model A5                3.986e+03  4.273e+02   9.328  < 2e-16 ***
## model A6                4.931e+03  4.734e+02  10.417  < 2e-16 ***
## model A7                6.978e+03  8.943e+02   7.802 7.45e-15 ***
## model A8                1.003e+04  1.022e+03   9.816  < 2e-16 ***
## model Amarok            3.746e+03  9.987e+02   3.751 0.000178 ***
## model Arteon            2.849e+03  7.403e+02   3.849 0.000120 ***
## model B Class           6.606e+02  4.931e+02   1.340 0.180414
## model Beetle           -2.132e+03  1.400e+03  -1.523 0.127940
## model C Class           4.105e+03  3.130e+02  13.116  < 2e-16 ***
## model Caddy Life       -1.109e+03  3.381e+03  -0.328 0.742934
```

```
## model Caddy Maxi Life -4.961e+03  1.966e+03  -2.523 0.011661 *
## model Caravelle        1.779e+04  1.542e+03  11.541  < 2e-16 ***
## model CC               2.186e+02  1.051e+03   0.208 0.835286
## model CL Class         4.840e+03  5.208e+02   9.292  < 2e-16 ***
## model CLA Class        3.876e+03  1.050e+03   3.691 0.000226 ***
## model CLS Class        4.493e+03  8.238e+02   5.454 5.18e-08 ***
## model E Class          5.609e+03  3.660e+02  15.326  < 2e-16 ***
## model GL Class         3.797e+03  1.305e+03   2.909 0.003645 **
## model GLA Class        2.743e+03  4.390e+02   6.249 4.51e-10 ***
## model GLB Class        1.404e+04  3.382e+03   4.150 3.38e-05 ***
## model GLC Class        1.012e+04  4.306e+02  23.497  < 2e-16 ***
## model GLE Class        1.581e+04  5.829e+02  27.119  < 2e-16 ***
## model GLS Class        1.580e+04  1.716e+03   9.209  < 2e-16 ***
## model Golf             3.551e+02  2.918e+02   1.217 0.223687
## model Golf SV         -1.119e+03  6.965e+02  -1.607 0.108107
## model Jetta           -1.570e+03  2.398e+03  -0.655 0.512651
## model M Class          8.084e+03  3.388e+03   2.386 0.017064 *
## model M2               1.243e+04  3.386e+03   3.671 0.000244 ***
## model M4               1.072e+04  1.312e+03   8.172 3.84e-16 ***
## model Passat           3.564e+02  4.716e+02   0.756 0.449914
## model Polo            -1.603e+03  3.295e+02  -4.864 1.19e-06 ***
## model Q2               2.772e+03  4.503e+02   6.156 8.07e-10 ***
## model Q3               5.160e+03  3.979e+02  12.967  < 2e-16 ***
## model Q5               9.179e+03  4.661e+02  19.695  < 2e-16 ***
## model Q7               1.727e+04  6.834e+02  25.270  < 2e-16 ***
## model Q8               3.015e+04  2.407e+03  12.522  < 2e-16 ***
## model RS3              1.172e+04  1.967e+03   5.960 2.70e-09 ***
## model RS4              2.236e+04  2.404e+03   9.301  < 2e-16 ***
## model RS5              2.114e+04  3.385e+03   6.244 4.63e-10 ***
## model RS6              2.414e+04  1.438e+03  16.785  < 2e-16 ***
## model S Class          1.464e+04  1.023e+03  14.314  < 2e-16 ***
## model S3               5.607e+03  2.400e+03   2.336 0.019548 *
## model S4               1.026e+04  3.386e+03   3.031 0.002447 **
## model Scirocco        -2.063e+02  6.994e+02  -0.295 0.767992
## model Sharan           6.596e+02  6.595e+02   1.000 0.317255
## model Shuttle          3.003e+03  1.539e+03   1.952 0.050996 .
## model SL CLASS         4.612e+03  7.644e+02   6.033 1.73e-09 ***
## model SLK              2.931e+03  1.224e+03   2.395 0.016666 *
## model SQ5              7.752e+03  3.390e+03   2.287 0.022257 *
## model T-Cross          2.620e+02  6.871e+02   0.381 0.703041
## model T-Roc            2.237e+03  4.719e+02   4.741 2.19e-06 ***
## model Tiguan           3.438e+03  3.711e+02   9.263  < 2e-16 ***
## model Tiguan Allspace  3.722e+03  1.967e+03   1.893 0.058430 .
## model Touareg          6.368e+03  6.165e+02  10.329  < 2e-16 ***
## model Touran           3.003e+03  6.892e+02   4.357 1.35e-05 ***
## model TT               4.049e+03  6.047e+02   6.696 2.39e-11 ***
## model Up              -3.817e+03  4.497e+02  -8.488  < 2e-16 ***
## model V Class          1.203e+04  1.020e+03  11.795  < 2e-16 ***
## model X-CLASS          5.289e+03  9.936e+02   5.323 1.07e-07 ***
## model X1               3.214e+03  4.880e+02   6.586 5.03e-11 ***
## model X2               3.279e+03  6.993e+02   4.688 2.83e-06 ***
## model X3               8.757e+03  5.664e+02  15.460  < 2e-16 ***
## model X4               1.081e+04  1.013e+03  10.670  < 2e-16 ***
## model X5               1.605e+04  5.913e+02  27.137  < 2e-16 ***
```

```
## model X6               2.250e+04  1.412e+03  15.933  < 2e-16 ***
## model X7               3.138e+04  1.975e+03  15.890  < 2e-16 ***
## model Z4               5.424e+03  1.099e+03   4.937 8.19e-07 ***
## f.ageage-Q2           -4.935e+03  1.584e+02 -31.152  < 2e-16 ***
## f.ageage-Q3           -7.710e+03  2.042e+02 -37.760  < 2e-16 ***
## f.ageage-Q4           -9.445e+03  2.283e+02 -41.371  < 2e-16 ***
## engineSize             6.486e+03  1.701e+02  38.137  < 2e-16 ***
## mileage               -1.115e-01  3.934e-03 -28.334  < 2e-16 ***
## fuelTypeDiesel        -1.144e+03  1.741e+02  -6.570 5.57e-11 ***
## fuelTypeHybrid         4.213e+03  7.643e+02   5.512 3.74e-08 ***
## fuelTypeOther          2.134e+02  1.045e+03   0.204 0.838245
## tax                   -2.152e+01  1.196e+00 -17.996  < 2e-16 ***
## mpg                   -1.120e+02  8.648e+00 -12.954  < 2e-16 ***
## transmissionSemi-Auto  1.455e+03  1.389e+02  10.475  < 2e-16 ***
## transmissionAutomatic  1.146e+03  1.522e+02   7.532 5.95e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3369 on 4742 degrees of freedom
## Multiple R-squared:  0.8881, Adjusted R-squared:  0.886
## F-statistic: 418.1 on 90 and 4742 DF,  p-value: < 2.2e-16
```

We add the factors and apply the step function again. The best model (lowest AIC) is with all the regressors except manufacturer. The Variance Inflation Factor is still low for most variables as shown.

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

The best model is shown in exercise 10. We have 90 regressors, most of them are model categories. A high number of them have high p-values (close to 1) which indicates not very good fitting. However, the R^2 statistic is close to 1 which indicates that the variance of the price variable is well explained by the predictors. The F-statistic test returns a very low p-value which shows a good fitting overall.

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

```
logm1 <- lm(log(price) ~ ., data=dfnum)
summary(logm1)
```

```
##
## Call:
## lm(formula = log(price) ~ ., data = dfnum)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.42441 -0.12315  0.00935  0.12869  0.78088
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.183e+02  4.805e+00 -45.434  < 2e-16 ***
## year         1.128e-01  2.380e-03  47.409  < 2e-16 ***
## mileage     -4.779e-06  2.278e-07 -20.976  < 2e-16 ***
## tax          1.669e-04  6.122e-05   2.726  0.00642 **
## mpg         -4.448e-03  2.954e-04 -15.060  < 2e-16 ***
## engineSize   4.568e-01  5.578e-03  81.880  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1912 on 4827 degrees of freedom
## Multiple R-squared:  0.8272, Adjusted R-squared:  0.827
## F-statistic:  4621 on 5 and 4827 DF,  p-value: < 2.2e-16
```

```
bptest(logm1)
```

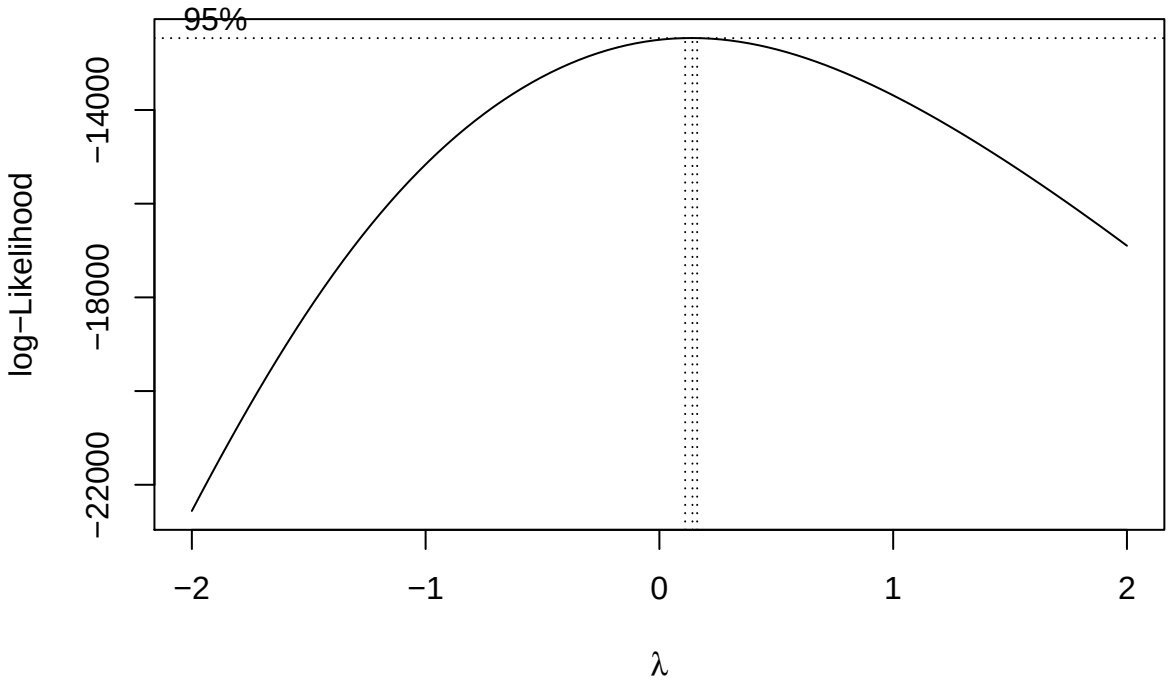
```
##
##  studentized Breusch-Pagan test
##
## data:  logm1
## BP = 66.521, df = 5, p-value = 5.418e-13
```

```
logm2 <- lm(price ~ boxCoxVariable(price) + year + mileage + tax + mpg + engineSize, data=dfnum)
summary(logm2)
```

```
##
## Call:
## lm(formula = price ~ boxCoxVariable(price) + year + mileage +
##     tax + mpg + engineSize, data = dfnum)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12182.3  -2186.5     96.6   2251.8  14372.9
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -3.878e+06  8.391e+04 -46.210  < 2e-16 ***
## boxCoxVariable(price)  9.260e-01  1.372e-02  67.480  < 2e-16 ***
## year                   1.936e+03  4.156e+01  46.577  < 2e-16 ***
## mileage               -8.175e-02  3.975e-03 -20.568  < 2e-16 ***
## tax                    6.526e+00  1.118e+00   5.838 5.64e-09 ***
## mpg                   -7.102e+01  5.418e+00 -13.107  < 2e-16 ***
```

```
## engineSize             7.514e+03  1.074e+02  69.990  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3331 on 4826 degrees of freedom
## Multiple R-squared:  0.8887, Adjusted R-squared:  0.8885
## F-statistic:  6420 on 6 and 4826 DF,  p-value: < 2.2e-16
```

```
boxcox(price ~ ., data=dfnum)
```



This improves the fit of the model. If we apply a Box-Cox transformation on the response variable, the ideal lambda value is $\lambda = 0.0926$ This improves the R squared up to 0.8924, higher than with $\lambda = 0$, the natural logarithm transformation.

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

As it is shown in question 10, the factors help to explain more variance in the response variable.

```
m7 <- lm(price ~  1, data=df)
m7_forw_aic = step(m7,
scope = price ~ boxCoxVariable(price) + model + transmission + mileage + fuelType + tax + mpg +engineSi
summary(m7_forw_aic)
```

```
##
## Call:
## lm(formula = price ~ model + f.age + boxCoxVariable(price) +
##     mileage + engineSize + transmission + fuelType + mpg, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14014.3  -1449.4    -39.2   1459.0   9578.1
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)             3.570e+04  4.944e+02  72.206  < 2e-16 ***
## model 2 Series          1.076e+03  2.631e+02   4.090 4.38e-05 ***
## model 3 Series          2.524e+03  2.321e+02  10.876  < 2e-16 ***
## model 4 Series          2.912e+03  3.039e+02   9.583  < 2e-16 ***
## model 5 Series          3.424e+03  3.060e+02  11.190  < 2e-16 ***
## model 6 Series          3.847e+03  7.202e+02   5.342 9.63e-08 ***
## model 7 Series          5.998e+03  7.367e+02   8.141 4.96e-16 ***
## model 8 Series          5.744e+03  1.080e+03   5.319 1.09e-07 ***
## model A Class           2.100e+03  2.257e+02   9.305  < 2e-16 ***
## model A1                3.786e+02  2.627e+02   1.441 0.149586
## model A3                2.155e+03  2.362e+02   9.125  < 2e-16 ***
## model A4                3.322e+03  2.635e+02  12.606  < 2e-16 ***
## model A5                4.250e+03  2.910e+02  14.604  < 2e-16 ***
## model A6                4.871e+03  3.225e+02  15.102  < 2e-16 ***
## model A7                5.653e+03  6.091e+02   9.281  < 2e-16 ***
## model A8                6.670e+03  6.979e+02   9.557  < 2e-16 ***
## model Amarok            3.396e+03  6.778e+02   5.010 5.63e-07 ***
## model Arteon            3.312e+03  5.044e+02   6.565 5.78e-11 ***
## model B Class          -2.415e+02  3.361e+02  -0.718 0.472491
## model Beetle           -2.605e+03  9.540e+02  -2.731 0.006344 **
## model C Class           3.731e+03  2.134e+02  17.489  < 2e-16 ***
## model Caddy Life       -1.231e+03  2.304e+03  -0.534 0.593122
## model Caddy Maxi Life  -2.722e+03  1.340e+03  -2.031 0.042306 *
## model Caravelle         9.626e+03  1.056e+03   9.115  < 2e-16 ***
## model CC               -1.731e+03  7.162e+02  -2.418 0.015664 *
## model CL Class          5.165e+03  3.547e+02  14.563  < 2e-16 ***
## model CLA Class         4.045e+03  7.157e+02   5.652 1.68e-08 ***
## model CLS Class         4.330e+03  5.612e+02   7.716 1.46e-14 ***
## model E Class           4.774e+03  2.495e+02  19.132  < 2e-16 ***
## model GL Class          4.278e+03  8.895e+02   4.809 1.56e-06 ***
## model GLA Class         3.148e+03  2.992e+02  10.522  < 2e-16 ***
## model GLB Class         8.017e+03  2.306e+03   3.476 0.000513 ***
## model GLC Class         7.373e+03  2.956e+02  24.944  < 2e-16 ***
## model GLE Class         7.687e+03  4.098e+02  18.759  < 2e-16 ***
```

```
## model GLS Class         7.285e+03  1.175e+03   6.199 6.16e-10 ***
## model Golf              5.482e+02  1.983e+02   2.764 0.005726 **
## model Golf SV          -1.109e+03  4.746e+02  -2.337 0.019466 *
## model Jetta            -2.511e+03  1.634e+03  -1.537 0.124385
## model M Class           8.085e+03  2.308e+03   3.503 0.000465 ***
## model M2                5.561e+03  2.309e+03   2.408 0.016080 *
## model M4                6.856e+03  8.952e+02   7.658 2.27e-14 ***
## model Passat            4.713e+02  3.212e+02   1.467 0.142350
## model Polo             -3.625e+03  2.262e+02 -16.025  < 2e-16 ***
## model Q2                3.172e+03  3.069e+02  10.336  < 2e-16 ***
## model Q3                4.862e+03  2.712e+02  17.930  < 2e-16 ***
## model Q5                7.103e+03  3.185e+02  22.300  < 2e-16 ***
## model Q7                7.719e+03  4.820e+02  16.016  < 2e-16 ***
## model Q8                5.934e+03  1.671e+03   3.551 0.000388 ***
## model RS3               9.329e+03  1.341e+03   6.959 3.89e-12 ***
## model RS4               6.380e+03  1.652e+03   3.862 0.000114 ***
## model RS5               6.402e+03  2.315e+03   2.765 0.005713 **
## model RS6               8.588e+03  9.975e+02   8.609  < 2e-16 ***
## model S Class           9.188e+03  6.995e+02  13.134  < 2e-16 ***
## model S3                5.567e+03  1.635e+03   3.405 0.000667 ***
## model S4                5.443e+03  2.308e+03   2.358 0.018404 *
## model Scirocco          1.987e+02  4.762e+02   0.417 0.676523
## model Sharan            1.905e+03  4.495e+02   4.238 2.30e-05 ***
## model Shuttle           3.800e+03  1.049e+03   3.624 0.000293 ***
## model SL CLASS          4.098e+03  5.209e+02   7.867 4.45e-15 ***
## model SLK               2.679e+03  8.338e+02   3.214 0.001320 **
## model SQ5               9.773e+03  2.310e+03   4.231 2.37e-05 ***
## model T-Cross           8.148e+02  4.683e+02   1.740 0.081953 .
## model T-Roc             3.042e+03  3.217e+02   9.454  < 2e-16 ***
## model Tiguan            3.595e+03  2.529e+02  14.214  < 2e-16 ***
## model Tiguan Allspace   4.126e+03  1.340e+03   3.079 0.002089 **
## model Touareg           4.656e+03  4.208e+02  11.065  < 2e-16 ***
## model Touran            2.264e+03  4.698e+02   4.818 1.49e-06 ***
## model TT                4.338e+03  4.120e+02  10.530  < 2e-16 ***
## model Up               -8.240e+03  3.120e+02 -26.408  < 2e-16 ***
## model V Class           7.035e+03  6.983e+02  10.075  < 2e-16 ***
## model X-CLASS           3.198e+03  6.745e+02   4.741 2.19e-06 ***
## model X1                3.033e+03  3.323e+02   9.125  < 2e-16 ***
## model X2                3.673e+03  4.766e+02   7.708 1.55e-14 ***
## model X3                5.731e+03  3.873e+02  14.796  < 2e-16 ***
## model X4                6.849e+03  6.926e+02   9.889  < 2e-16 ***
## model X5                7.544e+03  4.178e+02  18.056  < 2e-16 ***
## model X6                6.914e+03  9.836e+02   7.029 2.37e-12 ***
## model X7                5.082e+03  1.389e+03   3.658 0.000257 ***
## model Z4                4.403e+03  7.488e+02   5.880 4.37e-09 ***
## f.ageage-Q2            -2.879e+03  1.115e+02 -25.824  < 2e-16 ***
## f.ageage-Q3            -4.441e+03  1.400e+02 -31.726  < 2e-16 ***
## f.ageage-Q4            -7.395e+03  1.560e+02 -47.408  < 2e-16 ***
## boxCoxVariable(price)   9.555e-01  1.217e-02  78.523  < 2e-16 ***
## mileage                -1.179e-01  2.683e-03 -43.941  < 2e-16 ***
## engineSize              3.485e+03  1.215e+02  28.690  < 2e-16 ***
## transmissionSemi-Auto   2.188e+03  9.516e+01  22.992  < 2e-16 ***
## transmissionAutomatic   1.791e+03  1.041e+02  17.207  < 2e-16 ***
## fuelTypeDiesel         -3.618e+02  1.191e+02  -3.037 0.002401 **
```

```
## fuelTypeHybrid         4.016e+03  5.188e+02   7.741 1.19e-14 ***
## fuelTypeOther          1.828e+03  7.128e+02   2.564 0.010377 *
## mpg                   -5.996e+01  5.527e+00 -10.848  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2296 on 4742 degrees of freedom
## Multiple R-squared:  0.948,  Adjusted R-squared:  0.947
## F-statistic:   961 on 90 and 4742 DF,  p-value: < 2.2e-16
```

```
vif(m7_forw_aic)
```

```
##                             GVIF Df GVIF^(1/(2*Df))
## model                  13.799538 78        1.016967
## f.age                   4.020863  3        1.261014
## boxCoxVariable(price)   2.391123  1        1.546326
## mileage                 2.664842  1        1.632434
## engineSize              3.922664  1        1.980572
## transmission            1.813567  2        1.160469
## fuelType                3.843842  3        1.251587
## mpg                     4.240058  1        2.059140
```

Now, the model obtained is a combination from the results of questions 10 and 12, meaning that a transformation on the price variable is applied, as well as the addition of the factors. The `step()` function returns the best model which has an $R^2 = 0.948$, the best so far. Overall, the model is good fit. The `vif()` function shows no major causes for concern.

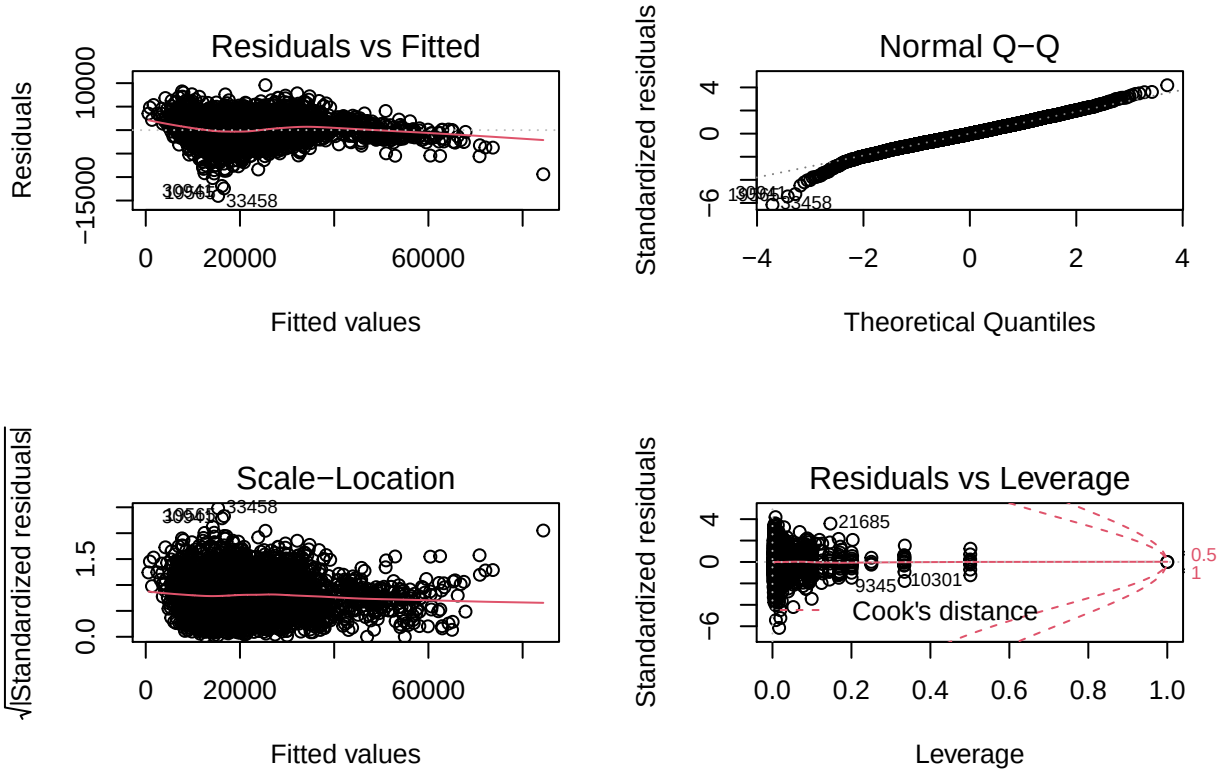
14. Graphically assess the best model obtained so far.

```
par(mfrow=c(2,2))
plot(m7_forw_aic)
```

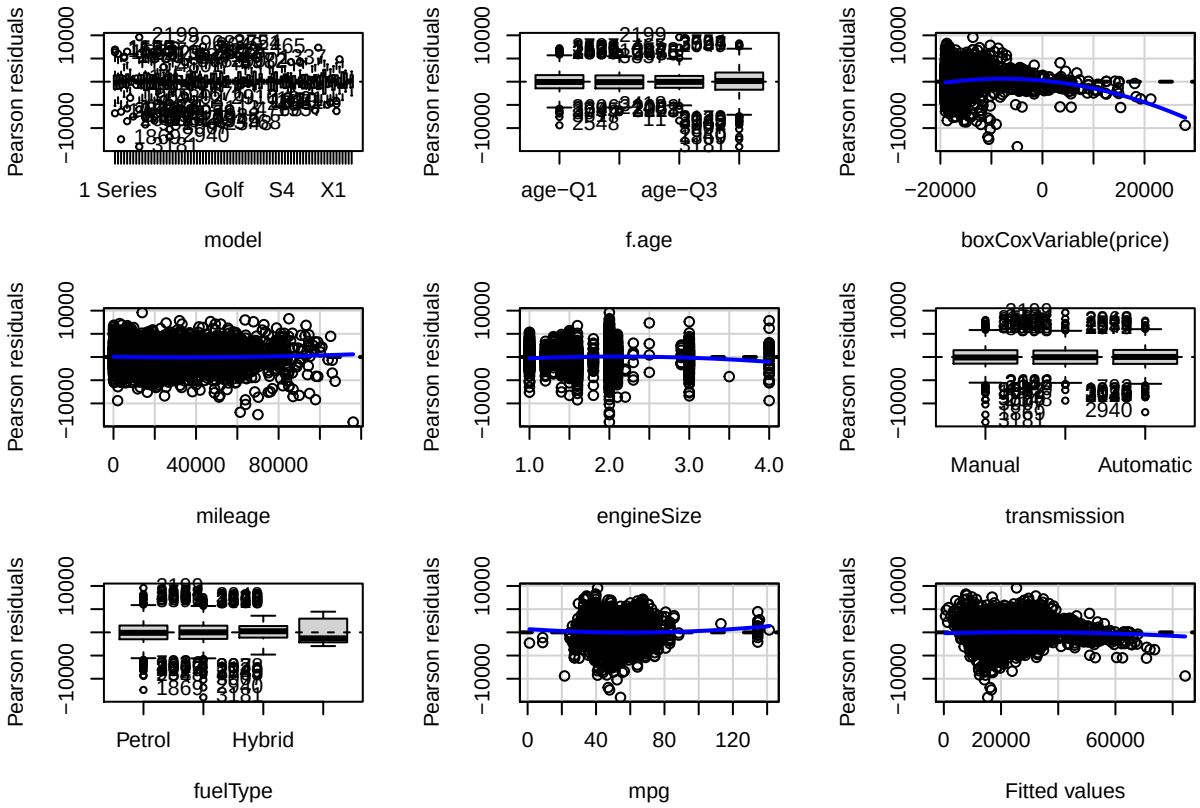
```
## Warning: not plotting observations with leverage one:
##   734, 995, 2561, 3194, 4833
```

```
## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced
```

```
## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced
```




```
residualPlots(m7_forw_aic)
```




```
##                          Test stat Pr(>|Test stat|)
## model
## f.age
## boxCoxVariable(price)  -10.6128        < 2.2e-16 ***
## mileage                  2.0594          0.03951 *
## engineSize              -6.8118        1.084e-11 ***
## transmission
## fuelType
## mpg                      4.6427        3.532e-06 ***
## Tukey test              -4.3106        1.628e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Residuals vs. Leverage Plot: In the plot we can see the Cook's distance (dashed lines); in this case, there are no points outside this distance except the observations reported in the warning, this means that those observations are influential points since have leverage one.
- Scale-Location Plot: This plot is used to check the homoscedasticity (equal variance) in the residuals of the model; in this case, the red line is more or less horizontal, which means that we can assume the equal variance for the residuals.
- Normal Q-Q Plot: With this plot we can determine if the residuals follow a normal distribution; in this case, the residuals are well distributed along the diagonal line except some points in the tails that deviate from the line.
- Residuals vs. Fitted Plot: This plot is used to determine if the residuals exhibit non-linear patterns; in this case, the red line is almost horizontal, so the linear regression model is appropriate for this dataset.

In the residual Plots we see that most variables seem to have well behaved residuals.

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

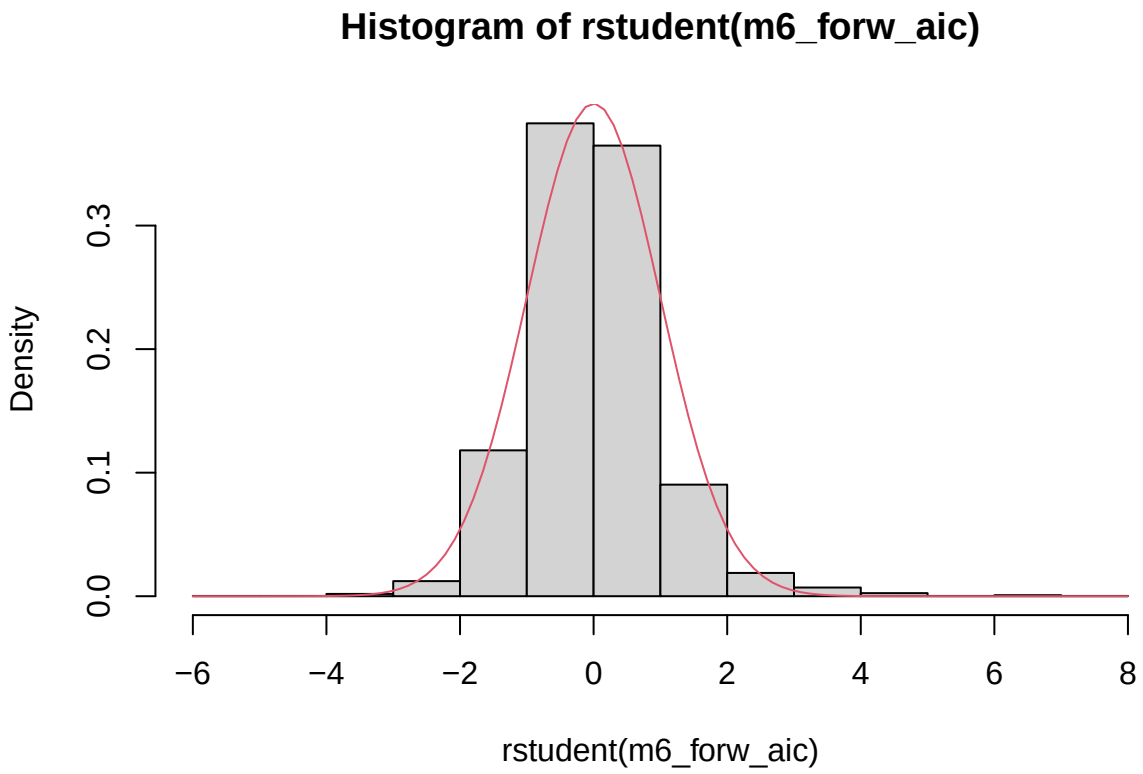
```
#qqPlot(m7_forw_aic, envelope = list(level = 0.99), labels=TRUE)

# Outliers at 99% CI

out1 <- which(studres(m6_forw_aic) < qt(0.005, 4742))
out2 <- which(studres(m6_forw_aic) > qt(0.995, 4742))

length(out1)
```

```
## [1] 19
```

```
length(out2)
```

```
## [1] 83
```

```
hist(rstudent(m6_forw_aic),freq=F)
curve(dt(x, m6_forw_aic$df),col=2,add=T)
```



First, we obtain the studentized residuals and plot them. From the plot, we can observe that there are several points that we consider outliers at a 99% confidence level. **Note:** The `qqPlot()` is commented, because we have problems when we render the file.

Also, in a more traditional way, we plot the t-student distribution and retrieve the outliers using a 99% CI. Counting the number of outliers we get a total of 102.

Finally, the plot of the histogram clearly shows the existence of such outliers.

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

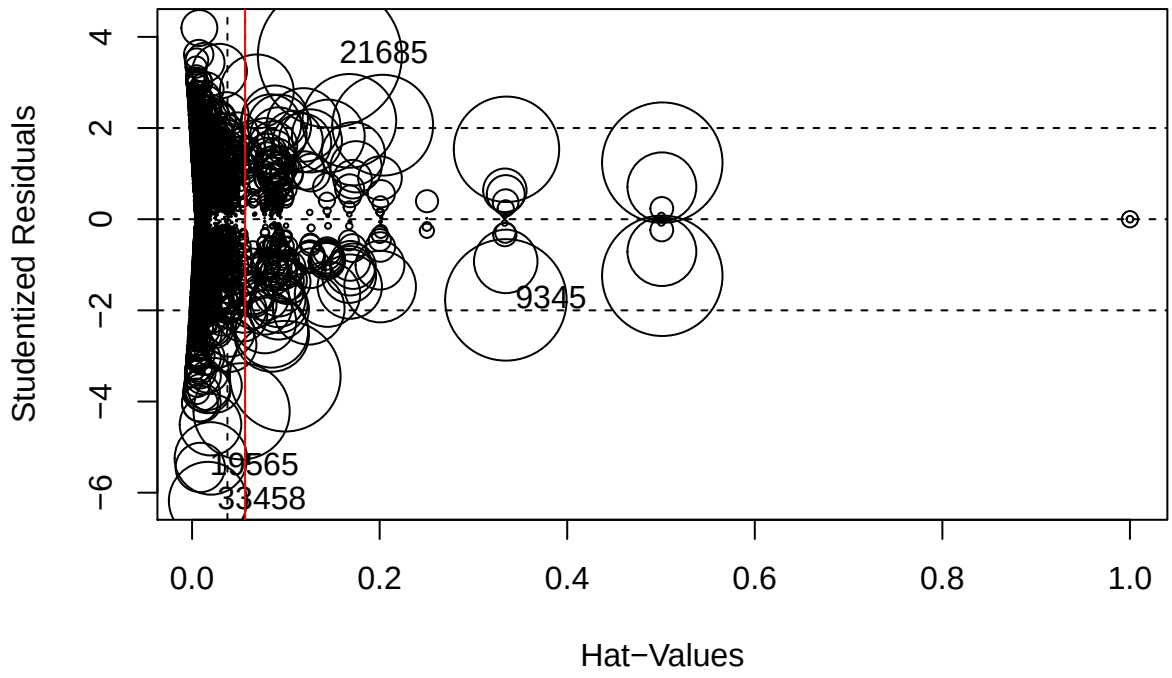
```
hat <- hatvalues(m7_forw_aic)
#Atypical values:
length(hat[hat > 3*mean(hat)])
```

```
## [1] 246
```

```
out3 <- hat[hat > 3*mean(hat)]
influencePlot(m7_forw_aic)
```

```
##         StudRes         Hat       CookD
## 7484        NaN 1.000000000         NaN
## 9345  -1.770094 0.334593046 0.017305563
## 10148       NaN 1.000000000         NaN
## 19565 -5.446947 0.008747211 0.002859778
## 21685  3.589420 0.146892554 0.024317350
## 33458 -6.179709 0.016840892 0.007132526
```

```
abline(v=3*mean(hat), col="red")
```



The hat-value is a leverage measure to study priori influential data observations. In this case, we can observe the atypical values that are outside the interval marked by the vertical dashed lines; 246 in total.

The threshold used appears in the Influence Plot. We note a large number of a priori influential data observations based on the hat values.

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

```
cooks <- cooks.distance(m7_forw_aic)
out4<-cooks[is.na(cooks)]

#Influential:
length(cooks[cooks > 1])
```

```
## [1] 5
```

It is observed that exactly 5 observations have a Cook's Distance higher than 1. This means they are atypical observations and should be eliminated from the model.

\textbf{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?}

```
p <- data.frame(model=" 1 Series", f.age= df$f.age[4], engineSize = mean(df$engineSize),  mileage = mea
```

```
m8 <- lm(price ~  1, data=df)
m8_forw_aic = step(m8,
scope = log(price) ~  model + transmission + mileage + fuelType + tax + mpg +engineSize + manufacturer
summary(m8_forw_aic)
```

```
##
## Call:
## lm(formula = price ~ model + f.age + engineSize + mileage + fuelType +
##     tax + mpg + transmission, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17842.7  -2074.6   -123.9   1812.5  24200.9
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             2.084e+04  6.867e+02  30.348  < 2e-16 ***
## model 2 Series          3.761e+02  3.859e+02   0.975 0.329816
## model 3 Series          2.323e+03  3.406e+02   6.819 1.03e-11 ***
## model 4 Series          2.282e+03  4.458e+02   5.118 3.21e-07 ***
## model 5 Series          3.537e+03  4.490e+02   7.878 4.10e-15 ***
## model 6 Series          2.450e+03  1.056e+03   2.319 0.020444 *
## model 7 Series          1.669e+04  1.062e+03  15.718  < 2e-16 ***
## model 8 Series          2.498e+04  1.540e+03  16.219  < 2e-16 ***
## model A Class           2.500e+03  3.312e+02   7.550 5.19e-14 ***
## model A1                1.015e+03  3.856e+02   2.633 0.008493 **
## model A3                2.228e+03  3.470e+02   6.420 1.49e-10 ***
## model A4                2.994e+03  3.870e+02   7.737 1.23e-14 ***
## model A5                3.986e+03  4.273e+02   9.328  < 2e-16 ***
## model A6                4.931e+03  4.734e+02  10.417  < 2e-16 ***
## model A7                6.978e+03  8.943e+02   7.802 7.45e-15 ***
## model A8                1.003e+04  1.022e+03   9.816  < 2e-16 ***
## model Amarok            3.746e+03  9.987e+02   3.751 0.000178 ***
## model Arteon            2.849e+03  7.403e+02   3.849 0.000120 ***
## model B Class           6.606e+02  4.931e+02   1.340 0.180414
## model Beetle           -2.132e+03  1.400e+03  -1.523 0.127940
## model C Class           4.105e+03  3.130e+02  13.116  < 2e-16 ***
## model Caddy Life       -1.109e+03  3.381e+03  -0.328 0.742934
## model Caddy Maxi Life -4.961e+03  1.966e+03  -2.523 0.011661 *
## model Caravelle         1.779e+04  1.542e+03  11.541  < 2e-16 ***
## model CC                2.186e+02  1.051e+03   0.208 0.835286
## model CL Class          4.840e+03  5.208e+02   9.292  < 2e-16 ***
## model CLA Class         3.876e+03  1.050e+03   3.691 0.000226 ***
## model CLS Class         4.493e+03  8.238e+02   5.454 5.18e-08 ***
## model E Class           5.609e+03  3.660e+02  15.326  < 2e-16 ***
## model GL Class          3.797e+03  1.305e+03   2.909 0.003645 **
## model GLA Class         2.743e+03  4.390e+02   6.249 4.51e-10 ***
## model GLB Class         1.404e+04  3.382e+03   4.150 3.38e-05 ***
```

```
## model GLC Class        1.012e+04  4.306e+02  23.497  < 2e-16 ***
## model GLE Class        1.581e+04  5.829e+02  27.119  < 2e-16 ***
## model GLS Class        1.580e+04  1.716e+03   9.209  < 2e-16 ***
## model Golf             3.551e+02  2.918e+02   1.217 0.223687
## model Golf SV         -1.119e+03  6.965e+02  -1.607 0.108107
## model Jetta           -1.570e+03  2.398e+03  -0.655 0.512651
## model M Class          8.084e+03  3.388e+03   2.386 0.017064 *
## model M2               1.243e+04  3.386e+03   3.671 0.000244 ***
## model M4               1.072e+04  1.312e+03   8.172 3.84e-16 ***
## model Passat           3.564e+02  4.716e+02   0.756 0.449914
## model Polo            -1.603e+03  3.295e+02  -4.864 1.19e-06 ***
## model Q2               2.772e+03  4.503e+02   6.156 8.07e-10 ***
## model Q3               5.160e+03  3.979e+02  12.967  < 2e-16 ***
## model Q5               9.179e+03  4.661e+02  19.695  < 2e-16 ***
## model Q7               1.727e+04  6.834e+02  25.270  < 2e-16 ***
## model Q8               3.015e+04  2.407e+03  12.522  < 2e-16 ***
## model RS3              1.172e+04  1.967e+03   5.960 2.70e-09 ***
## model RS4              2.236e+04  2.404e+03   9.301  < 2e-16 ***
## model RS5              2.114e+04  3.385e+03   6.244 4.63e-10 ***
## model RS6              2.414e+04  1.438e+03  16.785  < 2e-16 ***
## model S Class          1.464e+04  1.023e+03  14.314  < 2e-16 ***
## model S3               5.607e+03  2.400e+03   2.336 0.019548 *
## model S4               1.026e+04  3.386e+03   3.031 0.002447 **
## model Scirocco        -2.063e+02  6.994e+02  -0.295 0.767992
## model Sharan           6.596e+02  6.595e+02   1.000 0.317255
## model Shuttle          3.003e+03  1.539e+03   1.952 0.050996 .
## model SL CLASS         4.612e+03  7.644e+02   6.033 1.73e-09 ***
## model SLK              2.931e+03  1.224e+03   2.395 0.016666 *
## model SQ5              7.752e+03  3.390e+03   2.287 0.022257 *
## model T-Cross          2.620e+02  6.871e+02   0.381 0.703041
## model T-Roc            2.237e+03  4.719e+02   4.741 2.19e-06 ***
## model Tiguan           3.438e+03  3.711e+02   9.263  < 2e-16 ***
## model Tiguan Allspace  3.722e+03  1.967e+03   1.893 0.058430 .
## model Touareg          6.368e+03  6.165e+02  10.329  < 2e-16 ***
## model Touran           3.003e+03  6.892e+02   4.357 1.35e-05 ***
## model TT               4.049e+03  6.047e+02   6.696 2.39e-11 ***
## model Up              -3.817e+03  4.497e+02  -8.488  < 2e-16 ***
## model V Class          1.203e+04  1.020e+03  11.795  < 2e-16 ***
## model X-CLASS          5.289e+03  9.936e+02   5.323 1.07e-07 ***
## model X1               3.214e+03  4.880e+02   6.586 5.03e-11 ***
## model X2               3.279e+03  6.993e+02   4.688 2.83e-06 ***
## model X3               8.757e+03  5.664e+02  15.460  < 2e-16 ***
## model X4               1.081e+04  1.013e+03  10.670  < 2e-16 ***
## model X5               1.605e+04  5.913e+02  27.137  < 2e-16 ***
## model X6               2.250e+04  1.412e+03  15.933  < 2e-16 ***
## model X7               3.138e+04  1.975e+03  15.890  < 2e-16 ***
## model Z4               5.424e+03  1.099e+03   4.937 8.19e-07 ***
## f.ageage-Q2           -4.935e+03  1.584e+02 -31.152  < 2e-16 ***
## f.ageage-Q3           -7.710e+03  2.042e+02 -37.760  < 2e-16 ***
## f.ageage-Q4           -9.445e+03  2.283e+02 -41.371  < 2e-16 ***
## engineSize             6.486e+03  1.701e+02  38.137  < 2e-16 ***
## mileage               -1.115e-01  3.934e-03 -28.334  < 2e-16 ***
## fuelTypeDiesel        -1.144e+03  1.741e+02  -6.570 5.57e-11 ***
## fuelTypeHybrid         4.213e+03  7.643e+02   5.512 3.74e-08 ***
```

```
## fuelTypeOther          2.134e+02  1.045e+03   0.204 0.838245
## tax                   -2.152e+01  1.196e+00 -17.996  < 2e-16 ***
## mpg                   -1.120e+02  8.648e+00 -12.954  < 2e-16 ***
## transmissionSemi-Auto  1.455e+03  1.389e+02  10.475  < 2e-16 ***
## transmissionAutomatic  1.146e+03  1.522e+02   7.532 5.95e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3369 on 4742 degrees of freedom
## Multiple R-squared:  0.8881, Adjusted R-squared:  0.886
## F-statistic: 418.1 on 90 and 4742 DF,  p-value: < 2.2e-16
```

```
vif(m8_forw_aic)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## model        8.792462 78        1.014033
## f.age        4.147875  3        1.267567
## engineSize   3.572092  1        1.889998
## mileage      2.662095  1        1.631593
## fuelType     3.837405  3        1.251237
## tax          2.109481  1        1.452405
## mpg          4.820870  1        2.195648
## transmission 1.793149  2        1.157189
```

```
predict(m8_forw_aic, newdata=p, type="response", interval="confidence")
```

```
##        fit     lwr      upr
## 1 12590.65 11961.2 13220.09
```

```
dfnew <-df
names(out1)
```

```
##  [1] "93"    "2532"  "12320" "13315" "19281" "20272" "20596" "20798" "21269"
## [10] "22322" "22592" "22823" "22948" "23881" "24319" "31917" "33127" "33709"
## [19] "40983"
```

```
dfnew <- dfnew[!row.names(dfnew) %in% names(out1),]
length(out2)
```

```
## [1] 83
```

```
dfnew <- dfnew[!row.names(dfnew) %in% names(out2),]
length(out3)
```

```
## [1] 246
```

```
dfnew <- dfnew[!row.names(dfnew) %in% names(out3),]
length(out4)
```

```
## [1] 5
```

```
dfnew <- dfnew[!row.names(dfnew) %in% names(out4),]

m9 <- lm(price ~  1, data=dfnew)
m9_forw_aic = step(m9,
scope = log(price) ~  model + transmission + mileage + fuelType + tax + mpg +engineSize + manufacturer +
summary(m9_forw_aic)
```

```
##
```

```
## Call:
## lm(formula = price ~ model + f.age + engineSize + mileage + fuelType +
##     transmission + tax + mpg, data = dfnew)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8233.7 -1874.7   -66.5  1667.0 10537.8
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            2.186e+04  6.536e+02  33.440  < 2e-16 ***
## model 2 Series         4.534e+02  3.257e+02   1.392  0.16405
## model 3 Series         2.498e+03  2.877e+02   8.683  < 2e-16 ***
## model 4 Series         2.516e+03  3.762e+02   6.688 2.55e-11 ***
## model 5 Series         3.493e+03  3.808e+02   9.173  < 2e-16 ***
## model A Class          2.341e+03  2.795e+02   8.374  < 2e-16 ***
## model A1               8.769e+02  3.246e+02   2.702  0.00693 **
## model A3               2.188e+03  2.921e+02   7.490 8.23e-14 ***
## model A4               2.944e+03  3.290e+02   8.948  < 2e-16 ***
## model A5               4.094e+03  3.607e+02  11.348  < 2e-16 ***
## model A6               5.106e+03  4.018e+02  12.707  < 2e-16 ***
## model Arteon           3.358e+03  6.328e+02   5.307 1.17e-07 ***
## model B Class          5.545e+02  4.144e+02   1.338  0.18097
## model C Class          3.787e+03  2.652e+02  14.278  < 2e-16 ***
## model CL Class         4.559e+03  4.404e+02  10.351  < 2e-16 ***
## model CLS Class        3.714e+03  7.267e+02   5.111 3.34e-07 ***
## model E Class          5.395e+03  3.120e+02  17.292  < 2e-16 ***
## model GLA Class        2.774e+03  3.693e+02   7.512 6.98e-14 ***
## model GLC Class        1.014e+04  3.669e+02  27.633  < 2e-16 ***
## model GLE Class        1.595e+04  5.243e+02  30.420  < 2e-16 ***
## model Golf             3.501e+02  2.463e+02   1.421  0.15532
## model Golf SV         -1.127e+03  5.846e+02  -1.927  0.05406 .
## model Passat           3.661e+02  4.001e+02   0.915  0.36021
## model Polo            -1.891e+03  2.786e+02  -6.787 1.29e-11 ***
## model Q2               2.761e+03  3.800e+02   7.265 4.37e-13 ***
## model Q3               5.125e+03  3.370e+02  15.207  < 2e-16 ***
## model Q5               8.823e+03  4.011e+02  21.999  < 2e-16 ***
## model Q7               1.757e+04  5.916e+02  29.696  < 2e-16 ***
## model Scirocco        -2.779e+02  5.880e+02  -0.473  0.63651
## model Sharan           7.871e+02  5.572e+02   1.413  0.15784
## model SL CLASS         4.541e+03  6.547e+02   6.936 4.62e-12 ***
## model T-Cross          1.447e+02  5.779e+02   0.250  0.80222
## model T-Roc            2.363e+03  3.976e+02   5.943 3.01e-09 ***
## model Tiguan           3.329e+03  3.172e+02  10.496  < 2e-16 ***
## model Touareg          6.379e+03  5.368e+02  11.883  < 2e-16 ***
## model Touran           2.771e+03  5.870e+02   4.721 2.42e-06 ***
## model TT               3.827e+03  5.132e+02   7.457 1.06e-13 ***
## model Up              -4.159e+03  3.790e+02 -10.972  < 2e-16 ***
## model X1               3.155e+03  4.105e+02   7.687 1.84e-14 ***
## model X2               3.424e+03  5.969e+02   5.736 1.03e-08 ***
## model X3               7.662e+03  4.937e+02  15.519  < 2e-16 ***
## model X5               1.632e+04  5.484e+02  29.763  < 2e-16 ***
## f.ageage-Q2           -4.368e+03  1.397e+02 -31.261  < 2e-16 ***
## f.ageage-Q3           -6.994e+03  1.787e+02 -39.135  < 2e-16 ***
```

```
## f.ageage-Q4           -8.816e+03  1.980e+02 -44.533  < 2e-16 ***
## engineSize             5.405e+03  1.564e+02  34.564  < 2e-16 ***
## mileage               -1.054e-01  3.375e-03 -31.231  < 2e-16 ***
## fuelTypeDiesel        -6.368e+02  1.624e+02  -3.920 8.98e-05 ***
## fuelTypeHybrid         6.417e+03  7.979e+02   8.042 1.12e-15 ***
## transmissionSemi-Auto  1.576e+03  1.184e+02  13.305  < 2e-16 ***
## transmissionAutomatic  1.261e+03  1.309e+02   9.637  < 2e-16 ***
## tax                   -1.801e+01  1.058e+00 -17.028  < 2e-16 ***
## mpg                   -1.194e+02  8.540e+00 -13.986  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2825 on 4458 degrees of freedom
## Multiple R-squared:  0.8944, Adjusted R-squared:  0.8932
## F-statistic: 726.4 on 52 and 4458 DF,  p-value: < 2.2e-16
```

```
vif(m9_forw_aic)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## model        6.906005 41        1.023846
## f.age        4.264374  3        1.273432
## engineSize   3.482140  1        1.866049
## mileage      2.606095  1        1.614341
## fuelType     4.773057  2        1.478084
## transmission 1.708520  2        1.143286
## tax          2.123515  1        1.457228
## mpg          5.813026  1        2.411022
```

```
predict(m9_forw_aic, newdata=p, type="response", interval="confidence", trace=0)
```

```
##        fit      lwr      upr
## 1 12980.19 12438.43 13521.95
```

Observation. To simplify the equations, in this exercise we use models with the logarithm of the price only.

We note that if the car is 5 years old, the variable f.age is set to the last quantile. The reference factor levels are "1 Series" for the `model`, Diesel for the `fuelType` and Automatic for the `transmission` variable. The rest is straightforward.

We also want to consider the alternative in which we remove the different outliers obtained in Questions 15, 16, 17, to see if the prediction varies a lot. We end up removing all the outliers and retry the prediction (model m9).

We observe differences in the results, as well as a smaller interval for the result in the m9 model.

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

This assignment has been very useful for us to apply the techniques learned in class in a real case.

We have been able to apply a linear regression model to this dataset, analyse the results, calculate outliers, interpret residuals and try to improve the model. The important thing to note here is that since this is a real Data Science case study, we started by searching for missing values and identifying outliers in our data set. The removal of multivariate outliers will help to find a more fitting model. Also, we had a good picture of the variables in our data set thanks to the initial exploratory analysis.

Different techniques and methods have been used during the whole project. The first part fas centered in the comparison between two factors (ANOVA). Then, the rest of the project was focused on finding a compatible linear model that predicts the price as a function of its regressors. Different strategies were considered, as a real data set makes us realize how big the scope of the problem can be and how complicated it is to obtain the perfect model, the factors that we have to take into account and the different ways to manage them. Transformations were applied on the variables to see if this helped improve the fitting of the linear model.

Also, we dug deeper into the analysis of the results by interpreting the residuals using the graphical and mathematical tools available to us. This helps us to find a better model as some outliers may also be found in the residual analysis.