Pràctica 1, Eines d'ajuda a la presa de decisions

Oscar Galera i Alfaro

15 d'Octubre, 2018

Anàlisi del Data Set *Auto MPG*

En aquesta pràctica s'analitzara el fitxer de dades que hi ha disponible en el següent enllaç

https://archive.ics.uci.edu/ml/machine-learning-databases/auto-mpg/auto-mpg.data

Nota: El *dataset* original també incorpora la variable *origin*, però com que no hi ha una descripció clara del seu significat ni es pot deduir a partir de la seva distribució de valors, s'ha optat per excloure-la d'aquest anàlisi.

Carregar les dades

El primer que cal fer és carregar el fitxer que conté les dades, per això s'executa la següent comanda

```
#Carregar les dades
dades <- read.csv(file="dades.csv", header=FALSE, sep = ",")
```

Preprocessament de les dades

Si tot va bé, s'ha de generar un nou dataset amb **398 registres i 12 variables** (la variable original "car name" s'ha dividit en les variables "model" i "marca").

El tipus d'aquestes variables correspon a:

- V1. *mpg*: Consum en *miles per galon* - V. Quantitativa continua
- V2. *cylinders*: Nombre de cilindres - V. Quantitativa discreta
- V3. *displacement*: Distància necessaria per frenar - V. Quantitativa continua
- V4. *horsepower*: Potència - V. Quantitativa continua
- V5. *weight*: Pes - V. Quantitativa continua
- V6. *acceleration*: Acceleració - V. Quantitativa continua
- V7. *model year*: Any del model - V. Quantitativa discreta
- V8. *model*: Nom del model - V. Qualitativa
- V9 *marca*: Marca - V. Qualitativa
- V10 *marca_1*: Marca del primer subconjunt - V. Qualitativa
- V11 *marca_2*: Marca del segon subconjunt - V. Qualitativa
- V12 *marca_3*: Marca del tercer subconjunt - V. Qualitativa

Com que el fitxer de dades no conté el nom de les variables i perquè sigui més fàcil la seva interpretació, es fa l'assignació de noms amb la següent comanda.

```
#Renombrar les columnes
names(dades) <- c("mpg", "cylinders", "displ", "hp", "weight", "accel", "model_year"
                  , "model", "marca", "marca_1", "marca_2", "marca_3")
```

Per veure com ha interpretat les variables *R*

```
#Mostrar el tipus de les variables
str(dades)
```

```
## 'data.frame':    398 obs. of  12 variables:
##  $ mpg       : num  10 13 31 9 29 31.9 41.5 44.3 43.4 44 ...
##  $ cylinders : int  8 8 4 8 4 4 4 4 4 4 ...
##  $ displ     : num  307 350 119 304 90 89 98 90 90 97 ...
##  $ hp        : Factor w/ 94 levels "?","100.0","102.0",..: 43 26 78 41 67 68 72 51 51 53 ...
##  $ weight    : num  4376 4055 2720 4732 1937 ...
##  $ accel     : num  15 12 19.4 18.5 14.2 14 14.7 21.7 23.7 24.6 ...
##  $ model_year: int  70 76 82 70 76 79 80 80 80 82 ...
##  $ model     : Factor w/ 189 levels "'cuda","100",..: 57 56 161 7 151 151 151 151 89 147 ...
##  $ marca     : Factor w/ 32 levels "amc","audi","bmw",..: 8 8 8 14 32 32 32 32 32 32 ...
##  $ marca_1   : Factor w/ 11 levels "bmw","buick",..: 3 3 3 5 11 11 11 11 11 11 ...
##  $ marca_2   : Factor w/ 12 levels "audi","capri",..: 8 8 8 8 8 8 8 8 8 8 ...
##  $ marca_3   : Factor w/ 12 levels "amc","cadillac",..: 8 8 8 8 8 8 8 8 8 8 ...
```

Variable *hp*

Dels resultats obtinguts, es pot veure que *R* classifica *hp* com una variable qualitativa quan es tracta d'una variable quantitativa, amb la següent comanda es mostra la distribució de valors que pren la variable

```
#Mostrar la distribució de valors que pren la variable hp
levels(dades$hp)
```

```
##  [1] "?"      "100.0" "102.0" "103.0" "105.0" "107.0" "108.0" "110.0"
##  [9] "112.0" "113.0" "115.0" "116.0" "120.0" "122.0" "125.0" "129.0"
## [17] "130.0" "132.0" "133.0" "135.0" "137.0" "138.0" "139.0" "140.0"
## [25] "142.0" "145.0" "148.0" "149.0" "150.0" "152.0" "153.0" "155.0"
## [33] "158.0" "160.0" "165.0" "167.0" "170.0" "175.0" "180.0" "190.0"
## [41] "193.0" "198.0" "200.0" "208.0" "210.0" "215.0" "220.0" "225.0"
## [49] "230.0" "46.00" "48.00" "49.00" "52.00" "53.00" "54.00" "58.00"
## [57] "60.00" "61.00" "62.00" "63.00" "64.00" "65.00" "66.00" "67.00"
## [65] "68.00" "69.00" "70.00" "71.00" "72.00" "74.00" "75.00" "76.00"
## [73] "77.00" "78.00" "79.00" "80.00" "81.00" "82.00" "83.00" "84.00"
## [81] "85.00" "86.00" "87.00" "88.00" "89.00" "90.00" "91.00" "92.00"
## [89] "93.00" "94.00" "95.00" "96.00" "97.00" "98.00"
```

Hi ha observacions amb valor absent (*?*) i per aquest motiu el inferidor de *R* no ha detectat correctament el tipus. Quantes d'aquestes observacions tenen valor absent?

```
#Nombre d'observacions amb valor abscent (?) en la variable hp
nrow(dades[dades$hp == "?",])
```

```
## [1] 6
```

De les diferents estratègies per resoldre aquesta situació, s'opta per eliminar les observació afectades perquè es recomana complir la regla $N > 20p$ (on N correspon al nombre d'observacions i p al nombre de variables) per a poder fer l'anàlisi sense problemes, i en aquest cas s'assoleix el valor mínim, ja que $392 > 240$

```r
#Filtrar les observacions que tenen valor absent en la variable hp
dades = dades[dades$hp != "?", ]
```

Fet això, ja es pot convertir la variable *hp* de qualitativa a quantitativa

```r
#Convertir hp de qualitativa a quantitativa
dades$hp = as.numeric(dades$hp)
str(dades)
```

```
## 'data.frame':    392 obs. of  12 variables:
##  $ mpg       : num  10 13 31 9 29 31.9 41.5 44.3 43.4 44 ...
##  $ cylinders : int  8 8 4 8 4 4 4 4 4 4 ...
##  $ displ     : num  307 350 119 304 90 89 98 90 90 97 ...
##  $ hp        : num  43 26 78 41 67 68 72 51 51 53 ...
##  $ weight    : num  4376 4055 2720 4732 1937 ...
##  $ accel     : num  15 12 19.4 18.5 14.2 14 14.7 21.7 23.7 24.6 ...
##  $ model_year: int  70 76 82 70 76 79 80 80 80 82 ...
##  $ model     : Factor w/ 189 levels "'cuda","100",..: 57 56 161 7 151 151 151 151 89 147 ...
##  $ marca     : Factor w/ 32 levels "amc","audi","bmw",..: 8 8 8 14 32 32 32 32 32 32 ...
##  $ marca_1   : Factor w/ 11 levels "bmw","buick",..: 3 3 3 5 11 11 11 11 11 11 ...
##  $ marca_2   : Factor w/ 12 levels "audi","capri",..: 8 8 8 8 8 8 8 8 8 8 ...
##  $ marca_3   : Factor w/ 12 levels "amc","cadillac",..: 8 8 8 8 8 8 8 8 8 8 ...
```

Variables *marca*, *marca_1*, *marca_2* i *marca_3*

En un inici la variable *marca* contenia moltes categories (37 en total), algunes de les quals mal escrites (32 després de la seva correcció) o amb molt pocs valors. Per facilitar la posterior representació, s'ha decidit fer una agrupació disjunta i equitativa de les categories en les variables *marca_1*, *marca_2* i *marca_3* (valor *other* per la resta), d'aquesta manera la distribució de les categories és:

```r
#Categories de la variable marca
table(dades$marca)
```

```
## 
##           amc          audi           bmw         buick      cadillac 
##            27             7             2            17             2 
##         capri     chevrolet         chevy      chrysler        datsun 
##             1            44             3             6            23 
##         dodge          fiat          ford            hi         honda 
##            28             8            48             1            13 
##         mazda mercedes-benz       mercury        nissan    oldsmobile 
##            12             3            11             1            10 
##          opel       peugeot      plymouth       pontiac       renault 
##             4             8            31            16             3 
##          saab        subaru        toyota       triumph    volkswagen 
##             4             4            26             1            16 
##         volvo            vw 
##             6             6
```

```r
#Categories de la variable marca_1
table(dades$marca_1)
```

```
## 
##     bmw   buick   chevy    ford      hi   mazda mercury   other pontiac 
##       2      17       3      48       1      12      11     273      16 
## renault      vw
```

```
##        3       6
```

```r
#Categories de la variable marca_2
table(dades$marca_2)
```

```
##
##          audi          capri         datsun          dodge           fiat
##             7              1             23             28              8
##         honda mercedes-benz          other        peugeot           saab
##            13              3            280              8              4
##       triumph     volkswagen
##             1             16
```

```r
#Categories de la variable marca_3
table(dades$marca_3)
```

```
##
##        amc   cadillac  chevrolet   chrysler     nissan oldsmobile
##         27          2         44          6          1         10
##       opel      other   plymouth     subaru     toyota      volvo
##          4        231         31          4         26          6
```

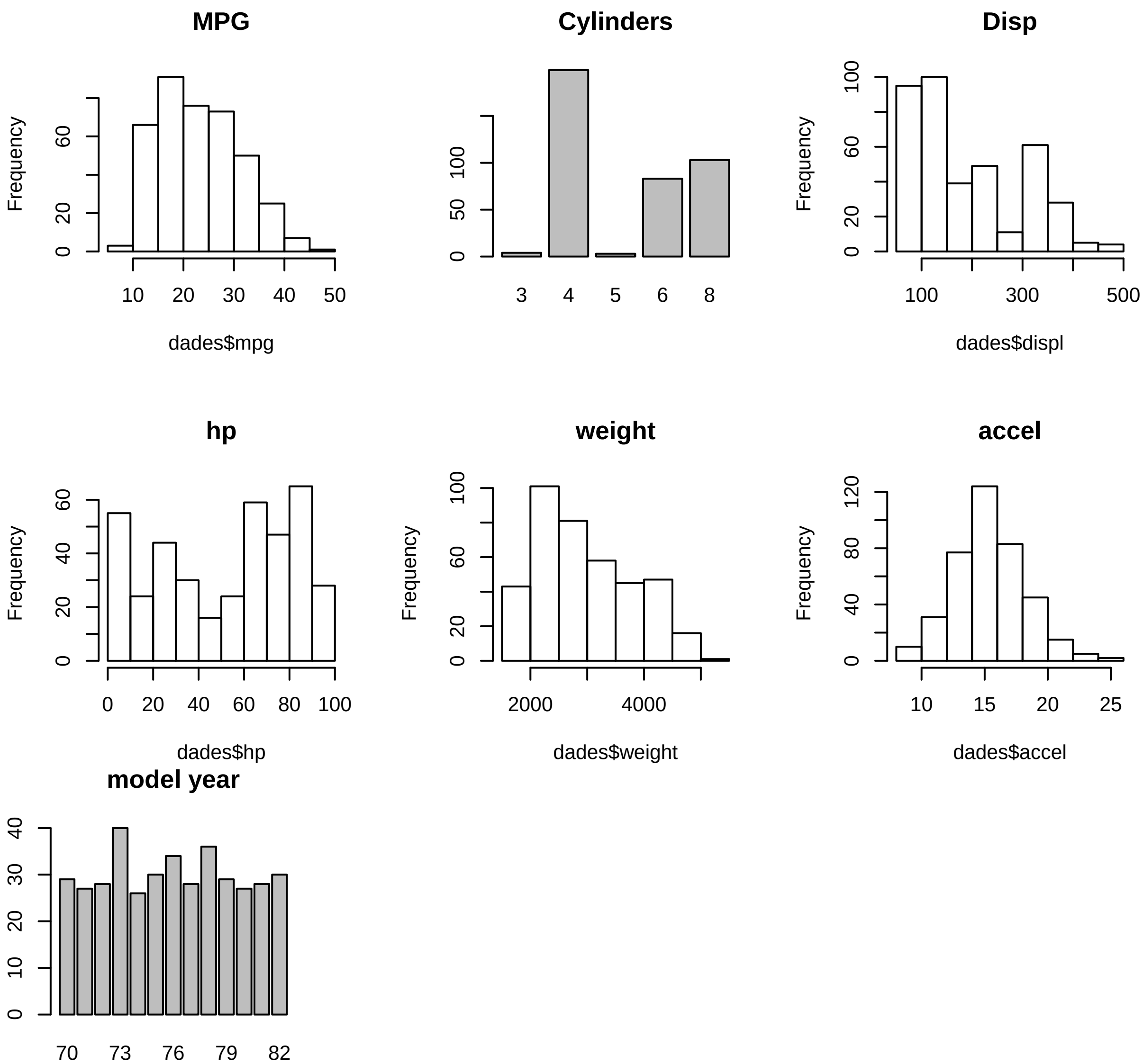
Arribats a aquest punt ja es tenen les dades preparades per començar a treballar.

Anàlisi descriptiu bàsic

Es vol veure els valors de centralitat i dispersió per cada variable. En els següents diagrames es mostra la distribució de valors.

Histogrames

```r
hNumerics <- function(){
  #Agrupar els gràfics en tripletes
  attach(mtcars)
  par(mfrow=c(2,3))

  hist(x = dades$mpg, main="MPG")
  barplot(table(dades$cylinders), main="Cylinders")
  hist(x = dades$displ, main = "Disp")
  hist(x = dades$hp, main="hp")
  hist(x = dades$weight, main="weight")
  hist(x = dades$accel, main="accel")
  barplot(table(dades$model_year), main="model year")
}
hNumerics()
```

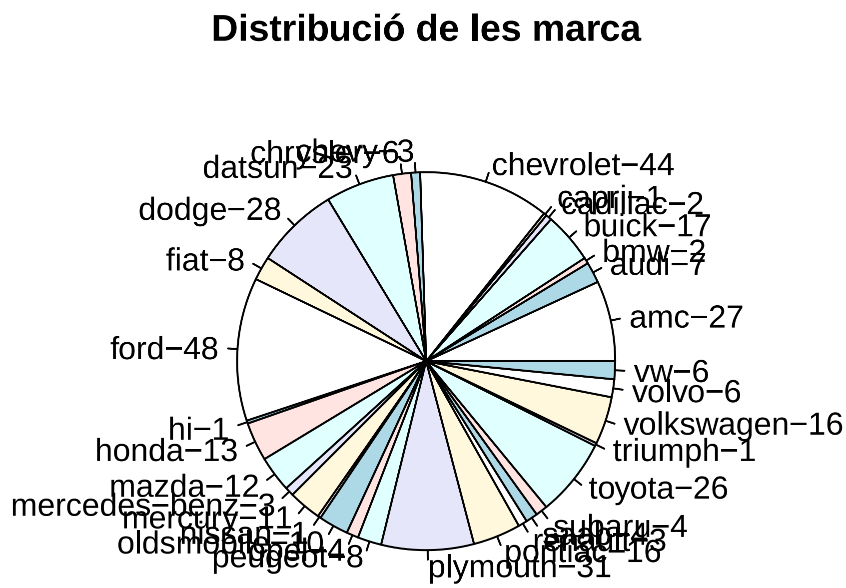


Es destaca que les variables *accel* té un alt grau de simetria, i que la distribució de la variable *model year* és força uniforme.

Diagrama de sectors

També es pot utilitzar un diagrama de sectors per veure com es distribueixen els vehicles de la mostra en base a les marques

```
pintarPie <- function (dades, titol){
  taula <- table(dades)
  etiquetes <- paste(names(taula), "-", taula, sep="")
  pie(taula, labels = etiquetes, main=titol)
}
pintarPie(dades$marca, "Distribució de les marca")
```

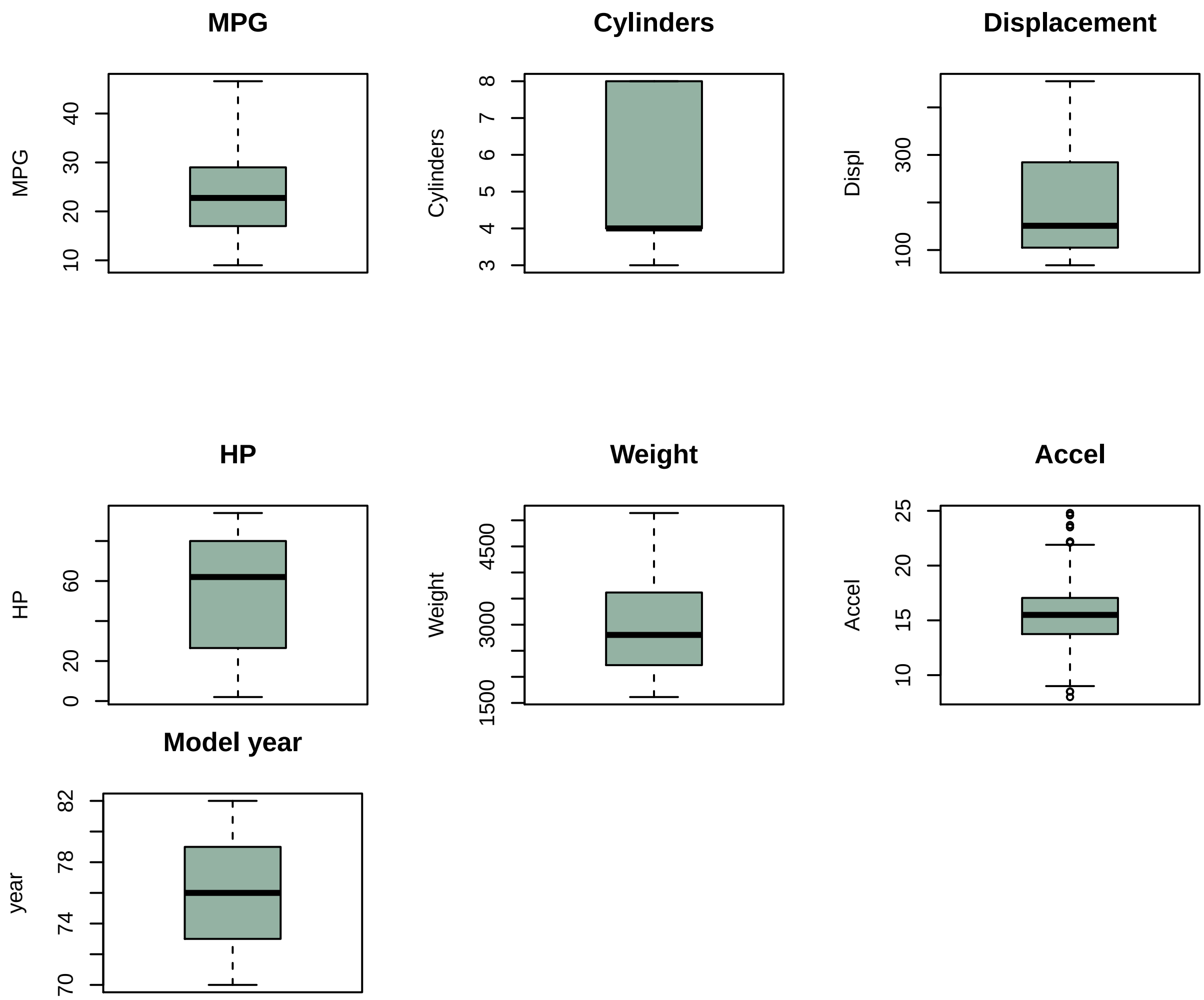


Les marques amb més representació són: *ford* (48), *chervolet* (44) i *plymouth* (31).

Diagrames de caixa

Un altre tipus de gràfic molt útil per aquesta finalitat és el diagrama de caixa.

```
bplotNumerics <- function(){
  attach(mtcars)
  par(mfrow=c(2,3))

  #Bloxplots
  boxplot(dades$mpg, col=rgb(0.3, 0.5, 0.4, 0.6), ylab="MPG", main="MPG")
  boxplot(dades$cylinders, col=rgb(0.3, 0.5, 0.4, 0.6), ylab="Cylinders", main="Cylinders")
  boxplot(dades$displ, col=rgb(0.3, 0.5, 0.4, 0.6), ylab="Displ", main="Displacement")
  boxplot(dades$hp, col=rgb(0.3, 0.5, 0.4, 0.6), ylab="HP", main="HP")
  boxplot(dades$weight, col=rgb(0.3, 0.5, 0.4, 0.6), ylab="Weight", main="Weight")
  boxplot(dades$accel, col=rgb(0.3, 0.5, 0.4, 0.6), ylab="Accel", main="Accel")
  boxplot(dades$model_year, col=rgb(0.3, 0.5, 0.4, 0.6), ylab="year", main="Model year")
}
bplotNumerics()
```



Aquí es pot veure com hi ha un clar biaix a la dreta per la variable *cylinders*, també es pot observar com les variable *accel* té certa simetria (com s'ha dit anteriorment) i com també conté dades atípiques.

Les dades atípies inferiors són aquelles que tenen un valor menor a $1Q - 1.5RIQ$ i les superiors aquelles amb un valor major a $3Q + 1.5RIQ$.

```r
#Valors atípics per la variable Accel.
#Inferiors
dades[dades$accel <= 13.78 - 1.5 * IQR(dades$accel), ][c("accel")]
```

```
##     accel
## 300   8.5
## 332   8.5
## 333   8.0
```

```r
#Superiors
dades[dades$accel >= 17.02 + 1.5 * IQR(dades$accel), ][c("accel")]
```

```
##     accel
## 9    23.7
## 10   24.6
## 136  21.9
## 138  24.8
## 154  23.5
## 245  22.2
```

```
## 280   22.2
## 281   22.1
```

Resum de les variables

En les següents taules es mostren els estadístics de centralitat i dispersió classificats com a robustos i no robustos

```r
#install.packages("psych")
library(knitr)
library(kableExtra)
taulaResum <- function(dades, nom, fw = TRUE){
  library(psych)
  #Estadístics de centralitat
  dades <- na.omit(dades)
  mitja <- mean(dades)
  mediana <- median(dades)
  mitjana_retallada_05 <- mean(dades, trim=0.5, na.rm = TRUE)

  if(fw){
    mitjana_winsoritzada_05 <- winsor.mean(dades, trim=0.5, na.rm=TRUE)
  }else{
    mitjana_winsoritzada_05 <- NA
  }

  #Estadístics de dispersió
  sd <- sd(dades)
  iqr <- IQR(dades)
  mad <- max(dades)

  df <- data.frame("Estadistic" =
                   c("Mediana", "Mit. Retallada", "Mit. Winsoritzada", "Mitjana"
                     , "RIC", "MAD", "D. Estàndard"), "Valor" =
                   c(mediana, mitjana_winsoritzada_05,
                     mitjana_winsoritzada_05, mitja, iqr, mad, sd))
  kable(df, caption = paste("Variable ",nom)) %>%
    kable_styling("striped", full_width = F) %>%
    group_rows("C. robustos", 1, 4) %>%
    group_rows("C. NO robustos", 4, 4) %>%
    group_rows("D. robustos", 5, 6) %>%
    group_rows("D. NO robustos", 7, 7)
}

taulaResum(dades$mpg, "mpg")
```

```r
taulaResum(dades$displ, "disp")
```

```r
taulaResum(dades$hp, "hp")
```

```r
taulaResum(dades$weight, "weight")
```

```r
taulaResum(dades$accel, "accel")
```

Table 1: Variable mpg

Estadistic	Valor
C. robustos	
Mediana	22.750000
Mit. Retallada	22.750000
Mit. Winsoritzada	22.750000
C. NO robustos	
Mitjana	23.445918
D. robustos	
RIC	12.000000
MAD	46.600000
D. NO robustos	
D. Estàndard	7.805008

Table 2: Variable disp

Estadistic	Valor
C. robustos	
Mediana	151.000
Mit. Retallada	151.000
Mit. Winsoritzada	151.000
C. NO robustos	
Mitjana	194.412
D. robustos	
RIC	170.750
MAD	455.000
D. NO robustos	
D. Estàndard	104.644

Table 3: Variable hp

Estadistic	Valor
C. robustos	
Mediana	62.00000
Mit. Retallada	62.00000
Mit. Winsoritzada	62.00000
C. NO robustos	
Mitjana	52.16071
D. robustos	
RIC	53.25000
MAD	94.00000
D. NO robustos	
D. Estàndard	29.49805

Table 4: Variable weight

Estadistic	Valor
C. robustos	
Mediana	2803.5000
Mit. Retallada	2803.5000
Mit. Winsoritzada	2803.5000
C. NO robustos	
Mitjana	2977.5842
D. robustos	
RIC	1389.5000
MAD	5140.0000
D. NO robustos	
D. Estàndard	849.4026

Table 5: Variable accel

Estadistic	Valor
C. robustos	
Mediana	15.500000
Mit. Retallada	15.500000
Mit. Winsoritzada	15.500000
C. NO robustos	
Mitjana	15.541327
D. robustos	
RIC	3.250000
MAD	24.800000
D. NO robustos	
D. Estàndard	2.758864

Anàlisi de components principals

L'anàlisi de components principals o *ACP* permet descriure un conjunt de dades, resumint-lo i reduir la seva dimensionalitat. En aquest cas, s'usarà per interpretar la relació que hi ha entre les variables.

Del conjunt de variables disponibles, s'ha decidit utilitzar les variables *marca*, *marca_1*, *marca_2* i *marca_3* com a variables suplementaries en la representació, i per tant, queda exclosa de les variables actives.

```
#Seleccionar varialbes actives
dades.PCA <- dades[, c("mpg", "cylinders", "displ", "hp", "weight", "accel", "model_year")]

names(dades.PCA)
```

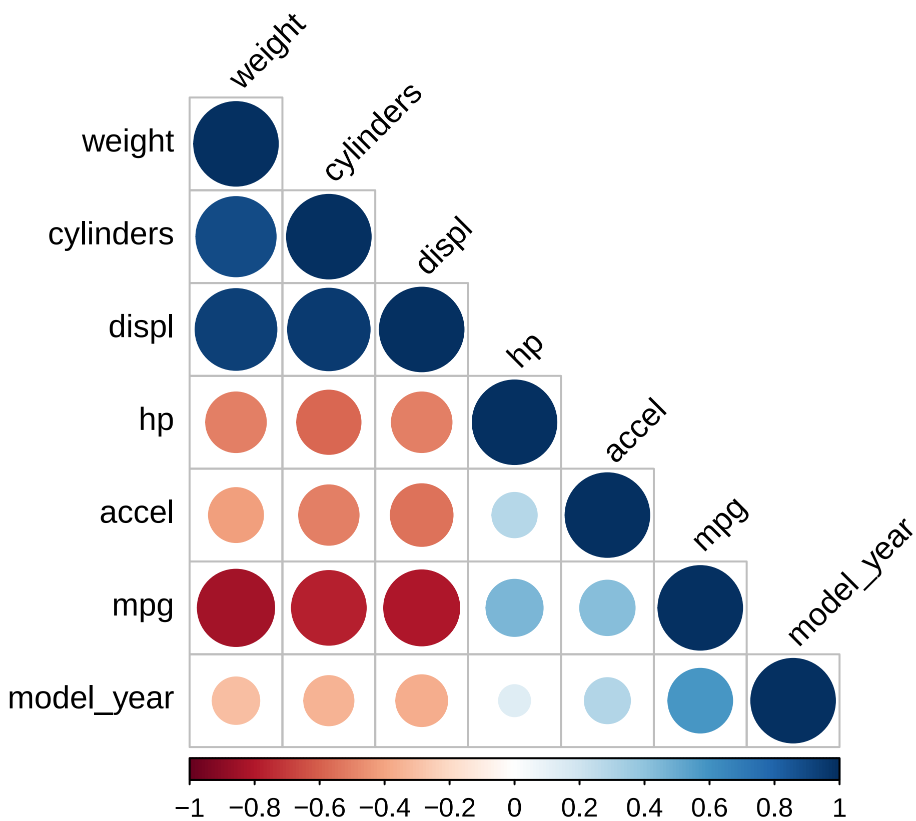
```
## [1] "mpg"        "cylinders"  "displ"      "hp"         "weight"
## [6] "accel"      "model_year"
```

Es pot aplicar l'anàlisi de components principals?

Abans de fer l'anàlisi però, cal comprovar que aquest es pugui realitzar. Això serà així si la correlació entre variables és significativa.

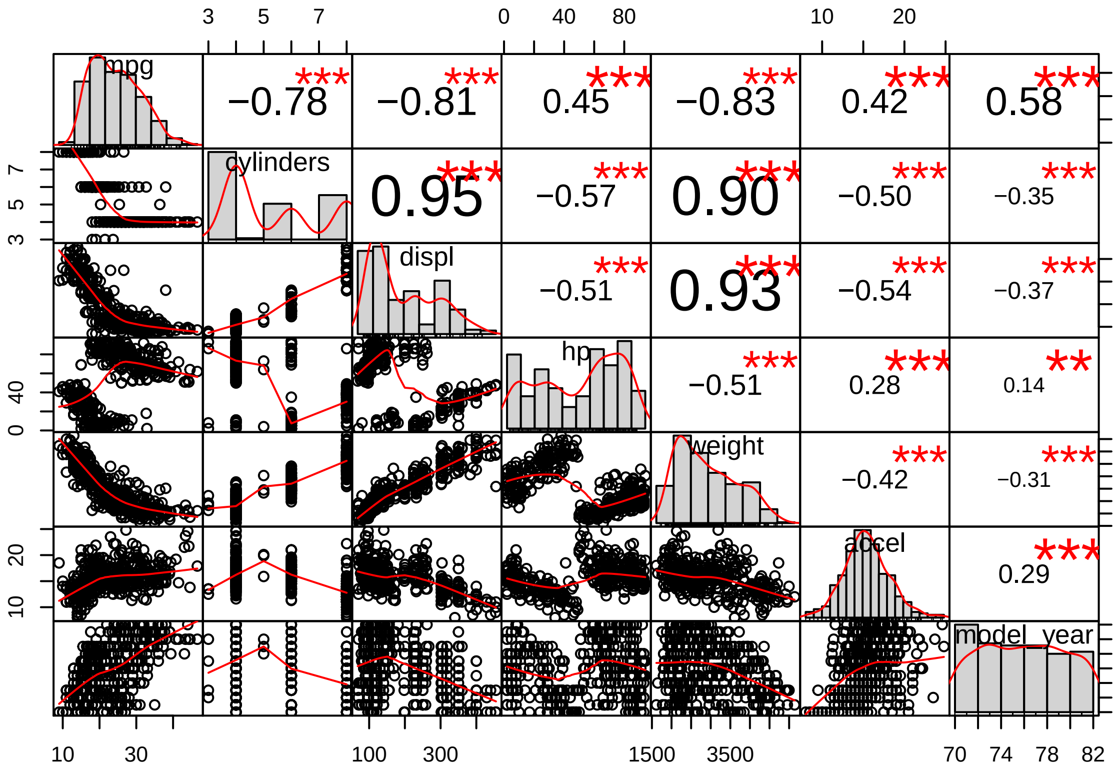
Matriu esquemàtica de correlació entre variables.

```
library("corrplot")
cor.mat <- cor(dades.PCA)
corrplot(cor.mat, type="lower", order="hclust", tl.col="black", tl.srt=45)
```



Matriu de gràfics bivariants i coeficients de correlació de *Pearson*.

```r
library("PerformanceAnalytics")
chart.Correlation(dades[,1:7], histogram = TRUE, pch = 19)
```



Seguidament es fa el test d'esfericitat de *Barlett*, que comprova si com a mínim dues de les variables de treball tenen diferent variància, és a dir, s'aplica el següent contrast d'hipòtesis:

$$H_0 : s_1^2 = s_2^2 \wedge s_3^2 ... \wedge s_k^2$$

$$H_1 : s_i^2 \neq s_j^2 | parella(i,j)$$

On k correspon al nombre de variables i *parella(i,j)* a alguna parella de variables.

```r
library(psych)
cortest.bartlett(cor.mat, n=100 )
```

```
## $chisq
## [1] 680.7449
##
## $p.value
## [1] 1.547461e-130
##
## $df
## [1] 21
```

Degut a que el $p - value$ és molt petit (pròxim a 0) es rebutja la hipòtesis nul · la i s'accepta amb un nivell de confiança del 95% que com a mínim hi ha una variable amb una variància diferent a la resta.

I també es pot aplicar un test de $Kaiser - Mayer - Olkin$

```r
library(psych)
KMO(cor.mat)
```

```
## Kaiser-Meyer-Olkin factor adequacy
```

```
## Call: KMO(r = cor.mat)
## Overall MSA =  0.8
## MSA for each item =
##        mpg  cylinders      displ         hp     weight      accel
##       0.81       0.84       0.77       0.88       0.79       0.83
## model_year
##       0.61
```

Com s'obté un resultat de 0,8 s'assumeix que el test és positiu i que es pot aplicar l'anàlisi de components principals.

Aplicant l'anàlisi de components principals

S'aplica l'anàlisi de components principals amb la funció *PCA* del paquet *FactoMineR*.

```r
library(FactoMineR)
res <- PCA(dades.PCA, scale.unit=TRUE, ncp=7, graph=FALSE)
```

Eixos factorials

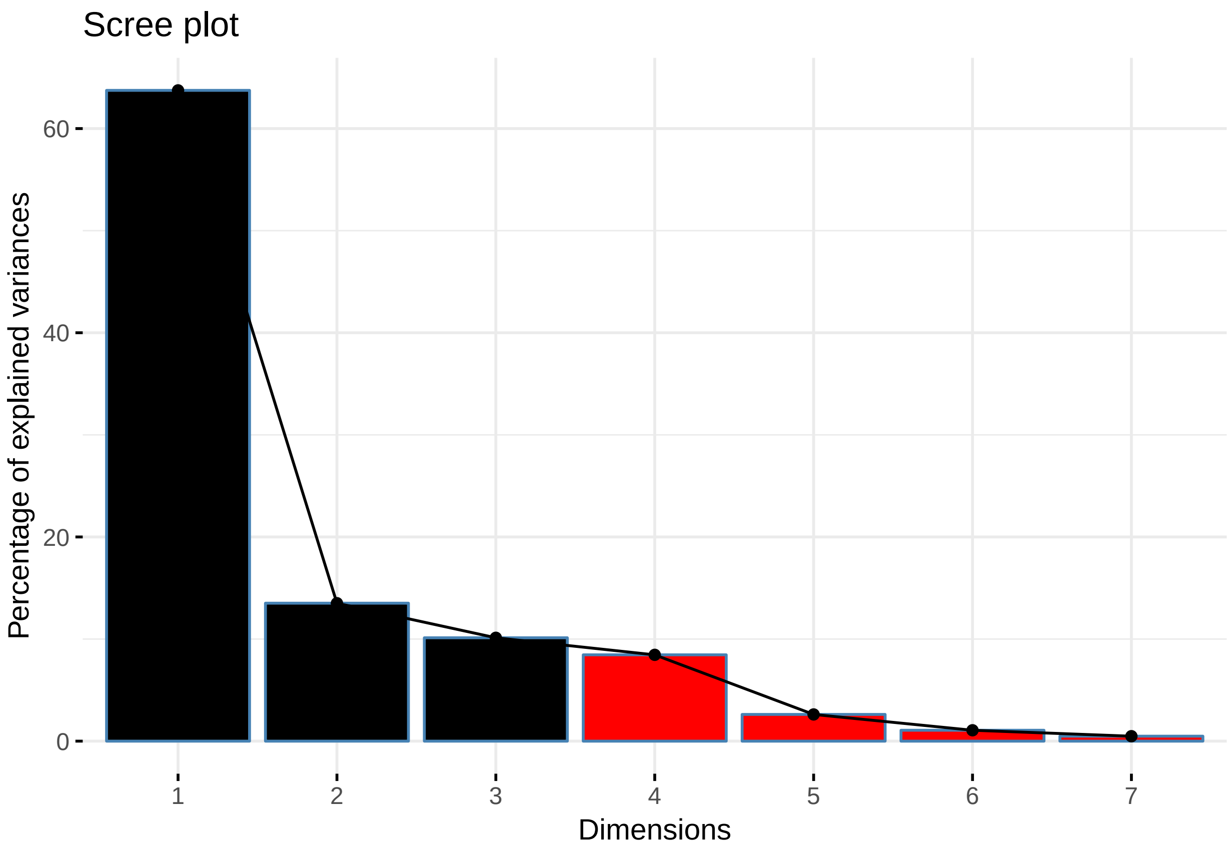
Com que hi ha set variables actives, s'han trobat set eixos factorials. En la següent llista es mostra la inèrcia que conté cada un d'aquests eixos, aquesta inèrcia ve representada pels valors propis de la matriu de variàncies.

```r
res$eig
```

```
##        eigenvalue percentage of variance cumulative percentage of variance
## comp 1 4.46178005             63.7397149                          63.73971
## comp 2 0.94578664             13.5112377                          77.25095
## comp 3 0.70881007             10.1258581                          87.37681
## comp 4 0.59180203              8.4543147                          95.83113
## comp 5 0.18316342              2.6166203                          98.44775
## comp 6 0.07471212              1.0673160                          99.51506
## comp 7 0.03394567              0.4849382                         100.00000
```

Però és clar, no serveix de gaire utilitzar-los tots, i així doncs, quins són els eixos més representatius? Segons el criteri *Latent Root* es poden considerar tots aquells eixos amb un valor propi superior a 1, és a dir, que tenen més inèrcia que qualsevol de les variables originals.

```r
library("factoextra")
fviz_screeplot(res, ncp=7, barfill=c(rep(1,3),rep(2,4)))
```



En aquest cas, tot i que un 63,74% de la variabilitat ve explicada pel primer eix, s'ha decidit agafar els tres primers per tenir més marge en els exemples. Així doncs, amb aquests tres eixos s'obté un 87,38% de la variabilitat de les dades.

També es pot comprovar quines variables han tingut un major impacte alhora de determinar els eixos, això ve dictat pel factor *contribution*. Pel primer eix, aquestes variables són: *displ*, *cylinders*, *weight* i *mpg*, i les de menor impacte són: *model_year*, *accel*. Pel segon eix les variables més impactants són *model_year* i *hp*, i pel tercer eix la variable *accel*.

```
res[["var"]][["contrib"]]
```

```
##                Dim.1      Dim.2       Dim.3      Dim.4      Dim.5
## mpg        17.971051  4.1289649  5.46536743  0.6485902 58.2654241
## cylinders  19.973778  2.4770942  0.06248921  2.5388709 20.6006934
## displ      20.484216  0.8996707  0.03251121  6.5770159  9.2294562
## hp          8.555233 22.3649622  2.06135113 66.1509514  0.1105271
## weight     19.212769  2.4323281  2.30976764 10.0299317  0.8491002
## accel       8.102681  2.0046966 85.29490851  1.8815834  1.8451240
## model_year  5.700273 65.6922833  4.77360487 12.1730567  9.0996750
##                 Dim.6      Dim.7
## mpg        12.8340210  0.6865811
## cylinders  33.6956323 20.6514423
## displ       0.3971941 62.3799362
## hp          0.5061954  0.2507799
## weight     49.9390296 15.2270741
## accel       0.3106581  0.5603486
## model_year  2.3172694  0.2438378
```

Les variables més ben representades tenen un major valor en el camp *cos2*, pel primer eix són: *displ*, *cylinders*, *weight* i *mpg*, i les menys ben representades: *model_year* i *accel*.

```r
res[["var"]][["cos2"]]
```

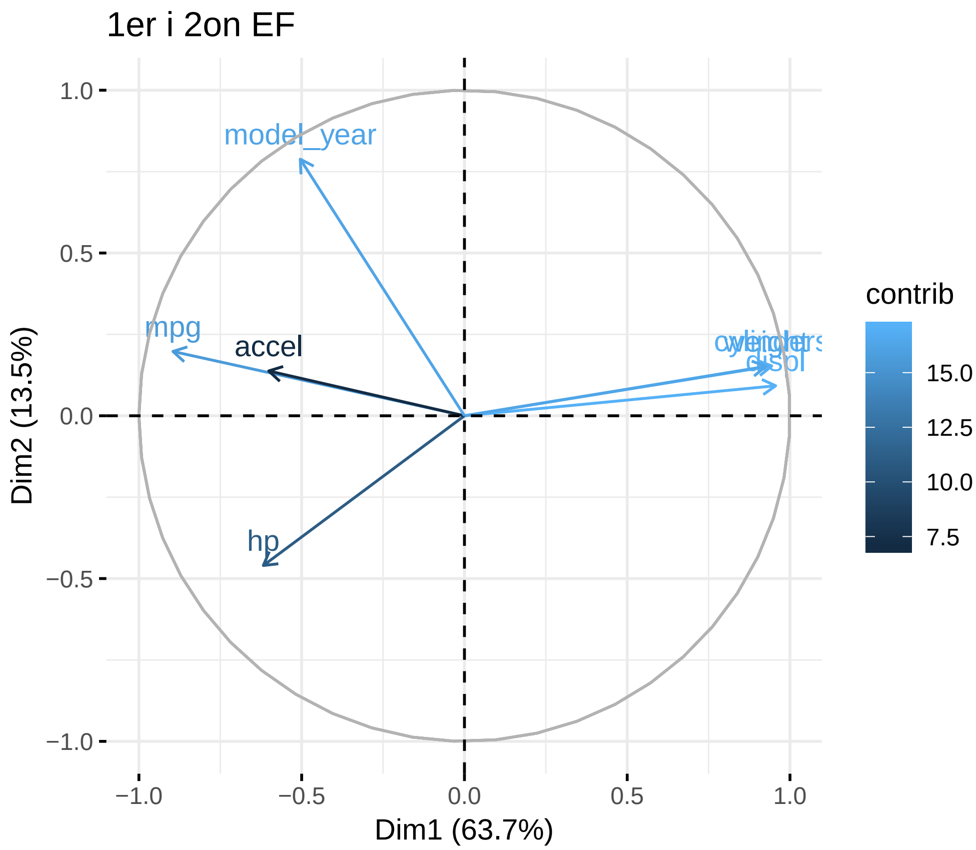
```
##                Dim.1       Dim.2        Dim.3      Dim.4        Dim.5
## mpg        0.8018288 0.039051199 0.0387390745 0.00383837 0.1067209428
## cylinders  0.8911860 0.023428026 0.0004429298 0.01502509 0.0377329344
## displ      0.9139606 0.008508965 0.0002304427 0.03892291 0.0169049876
## hp         0.3817157 0.211524825 0.0146110643 0.39148267 0.0002024452
## weight     0.8572315 0.023004635 0.0163718656 0.05935734 0.0015552410
## accel      0.3615238 0.018960152 0.6045788985 0.01113525 0.0033795922
## model_year 0.2543336 0.621308839 0.0338357919 0.07204040 0.0166672758
##                   Dim.6        Dim.7
## mpg        0.0095885694 2.330646e-04
## cylinders  0.0251747217 7.010271e-03
## displ      0.0002967521 2.117529e-02
## hp         0.0003781893 8.512893e-05
## weight     0.0373105084 5.168933e-03
## accel      0.0002320993 1.902141e-04
## model_year 0.0017312811 8.277239e-05
```

Ara per veure millor la representació de les variables sobre els eixos factorials, es poden plasmar sobre una circumferència de radi u, on els eixos de coordenades corresponen als dos eixos factorials amb més inèrcia i la tonalitat de blau al nivell de contribució.

```r
#Coordenades
res[["var"]][["cor"]]
```

```
##                 Dim.1       Dim.2       Dim.3       Dim.4       Dim.5
## mpg        -0.8954489  0.19761376 -0.19682244 -0.06195458  0.32668171
## cylinders   0.9440265  0.15306216  0.02104590  0.12257687  0.19424967
## displ       0.9560129  0.09224405 -0.01518034  0.19728891  0.13001918
## hp         -0.6178314 -0.45991828 -0.12087624  0.62568576  0.01422833
## weight      0.9258680  0.15167279  0.12795259  0.24363362 -0.03943654
## accel      -0.6012685  0.13769587  0.77754672  0.10552369  0.05813426
## model_year -0.5043150  0.78823146 -0.18394508  0.26840342 -0.12910180
##                  Dim.6        Dim.7
## mpg         0.09792124  0.015266453
## cylinders  -0.15866544  0.083727364
## displ       0.01722649 -0.145517320
## hp         -0.01944709  0.009226534
## weight      0.19315928  0.071895292
## accel      -0.01523480 -0.013791814
## model_year -0.04160867 -0.009097933
```

```r
#Gràfic
fviz_pca_var(res, axes = c(1, 2), col.var="contrib", title="1er i 2on EF")
```

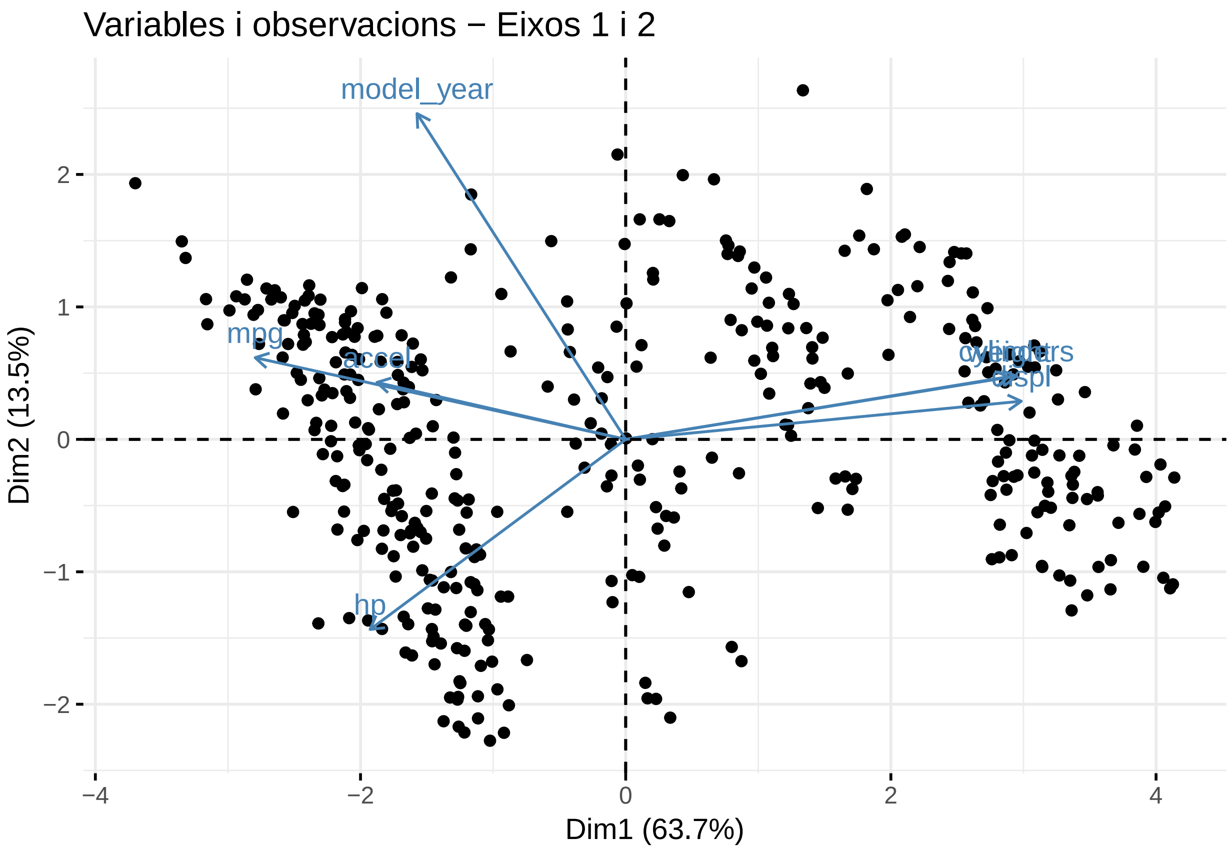


En el primer eix es representen els vehicles més pesats (*weight*) que solien tenir una major cilindrada (*cylinder*), i que tendien a consumir més combustible per milla recorreguda (*mpg*). També es pot veure com a mesura que augmentava el pes i la cilindrada dels cotxes, aquest necessitaven més espai per frenar (*displ*).

El segon eix explica que a mesura que augmentava l'any del model (*model_year*), hi havia tendència a disminuir la potència (*hp*), disminuir el consum (*mpg*) i augmentar l'acceleració (*accel*).

Ara es poden representar les observacions sobre els eixos factorials

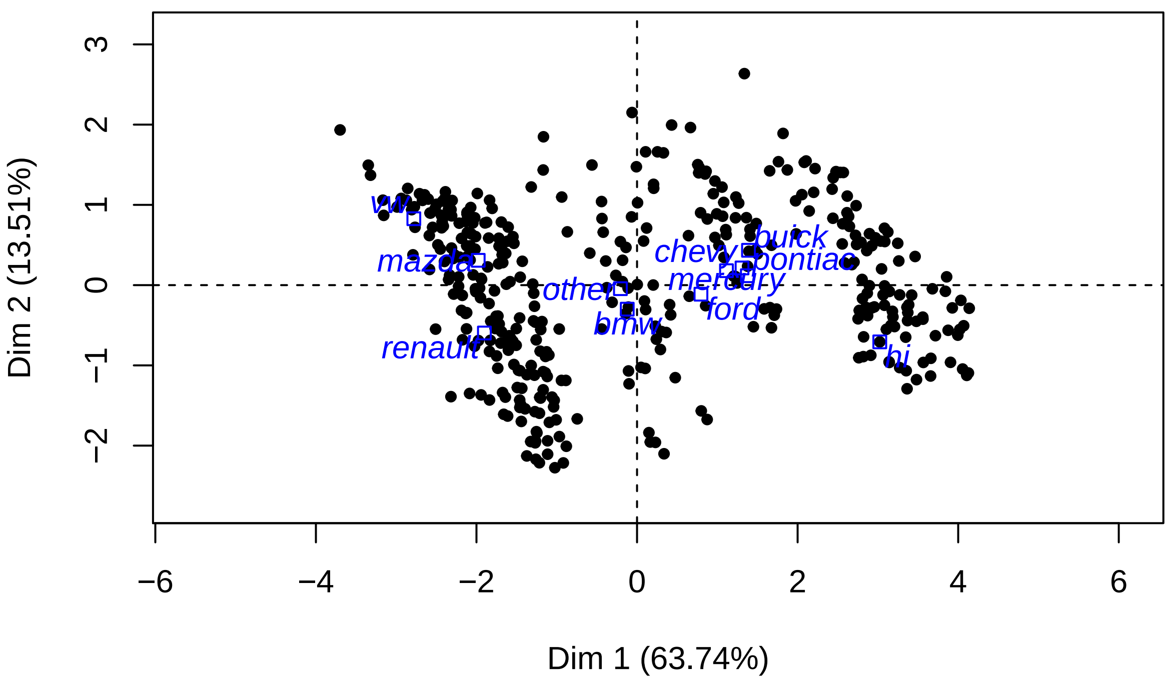
```
fviz_pca_biplot(res, axes = c(1, 2), geom="point", title="Variables i observacions - Eixos 1 i 2")
```



Projectar les categories de les variables *marca_1*, *marca_2* i *marca_3* que són suplementaries, és a dir, es mostraran les propietats característiques dels vehicles de cada marca en tres tandes.

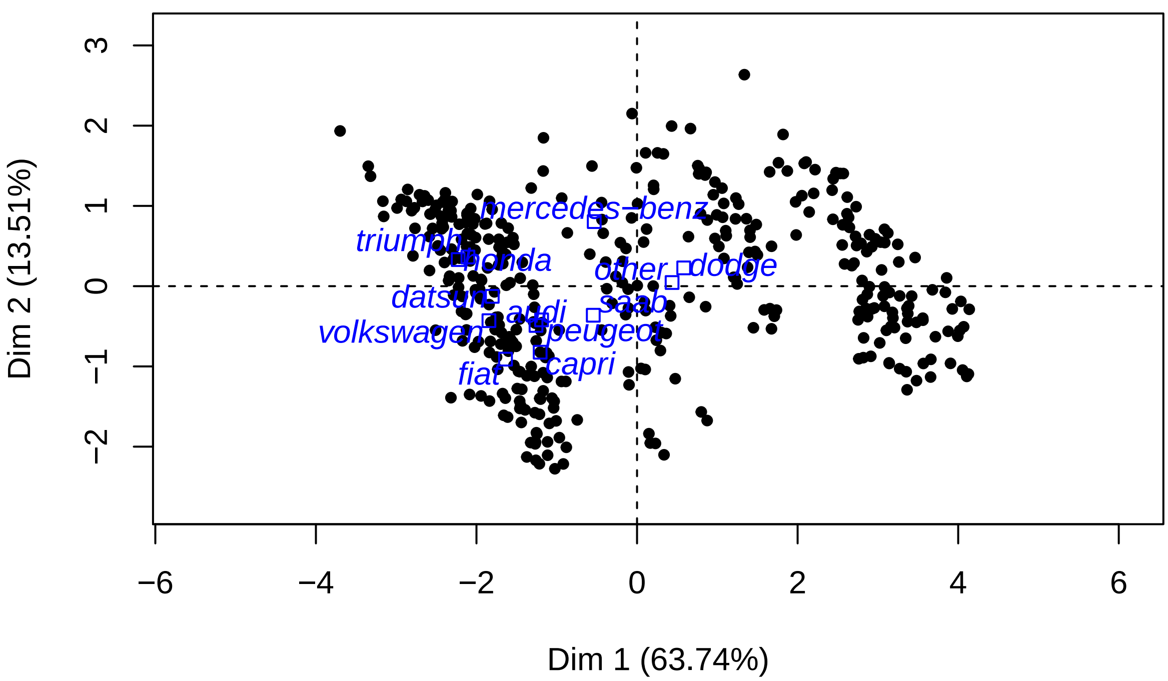
```
library(FactoMineR)
representacio <- function(dades, marca, titol, eix1 = 1, eix2 = 2){
  dades.PCA = dades[, c("mpg", "cylinders", "displ", "hp", "weight", "accel", "model_year",
                   marca)]
  res<-PCA(dades.PCA , scale.unit=TRUE, ncp=5, quali.sup=c(8:8), graph = FALSE)
  plot.PCA(res, axes=c(eix1, eix2), choix="ind", habillage="none", col.var="white",
           col.quali="blue", new.plot=TRUE, label=c("var", "quali"), title = titol)
}

#Marca 1
representacio(dades, "marca_1", "Marca_1 - Eixos: 1 i 2")
```

Marca_1 − Eixos: 1 i 2

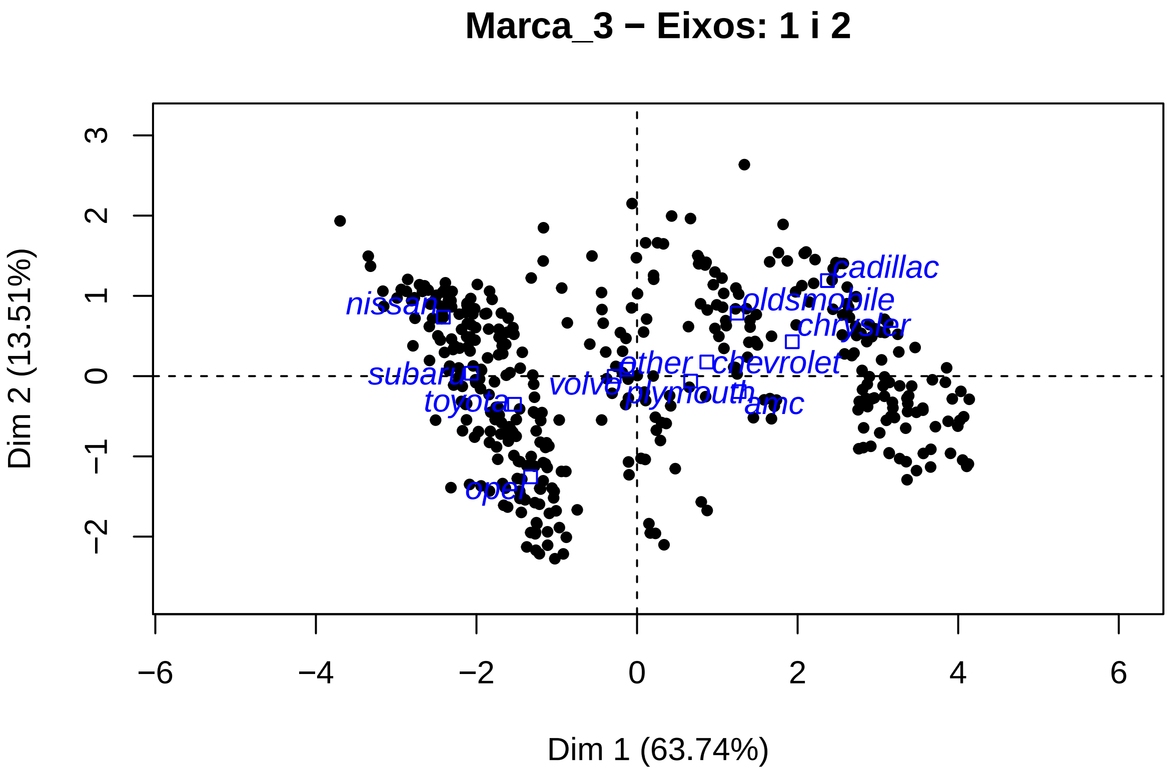



```r
#Marca 2
representacio(dades, "marca_2", "Marca_2 - Eixos: 1 i 2")
```

Marca_2 − Eixos: 1 i 2




```r
#Marca 3
representacio(dades, "marca_3", "Marca_3 - Eixos: 1 i 2")
```

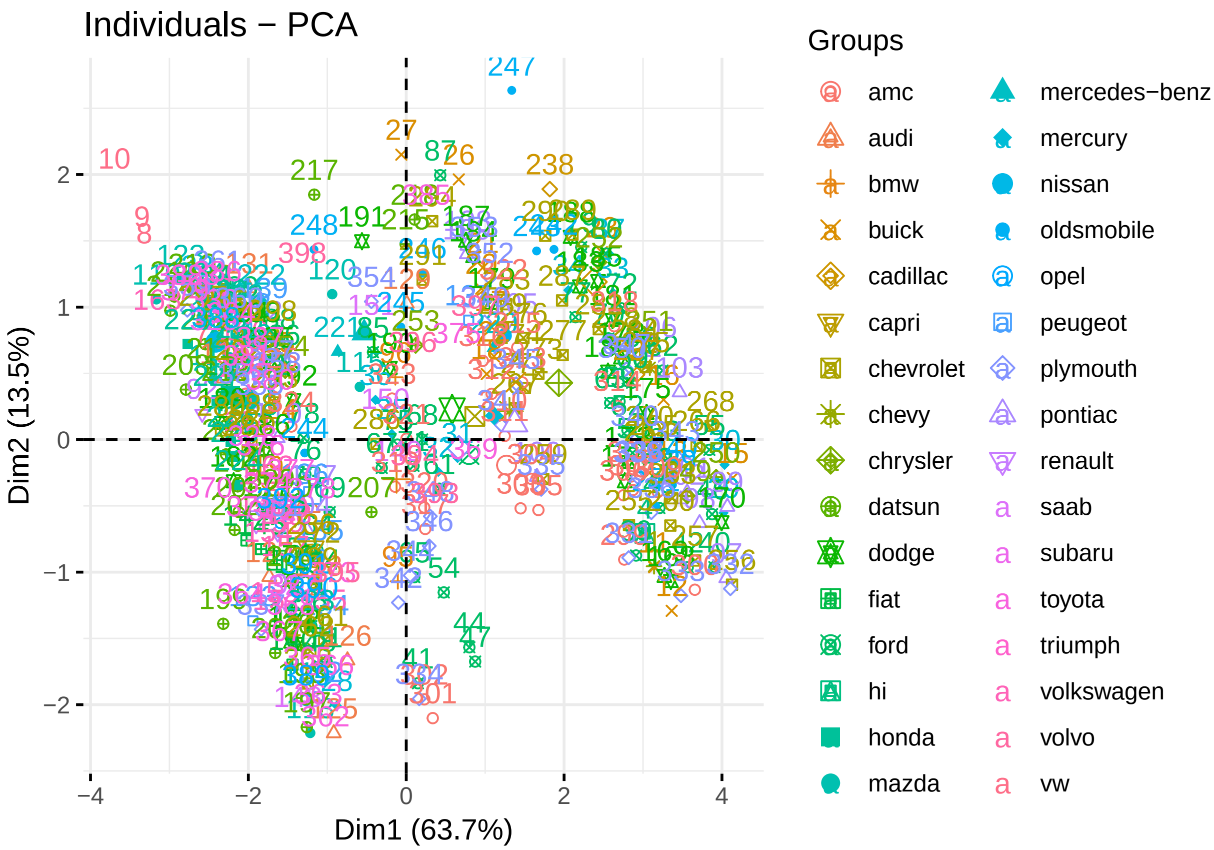


Amb aquests tres gràfics i a partir del primer eix factorial, es pot veure que les marques *vw*, *mazda*, *honda*, *triumph*, *nissan*, *renault* i *subaru* tendien (anys 70) a fabricar vehicles poc pesants, amb pocs cilindres i amb un consum baix. Les marques *volvo*, *bmw* i *ford* buscaven un cert equilibri entre pes i consum, i finalment, fabricants com *hi*, *chrysler* o *cadillac* tendien a treballar de forma contraria i fabricaven cotxes més pesants i que consumien més.

A partir del segon eix, es pot veure com les marques *vw* i *nissan* tenen representació de veichles mes moderns (*model year*) mentre que els vehicles de les marques *fiat*, *capri* o *opel* són més antics.

En el següent gràfic es mostra la projecció de tots els vehicles sobre els dos primers eixos factorials, utilitzant diferents símbols i colors en funció de la marca del vehicle.

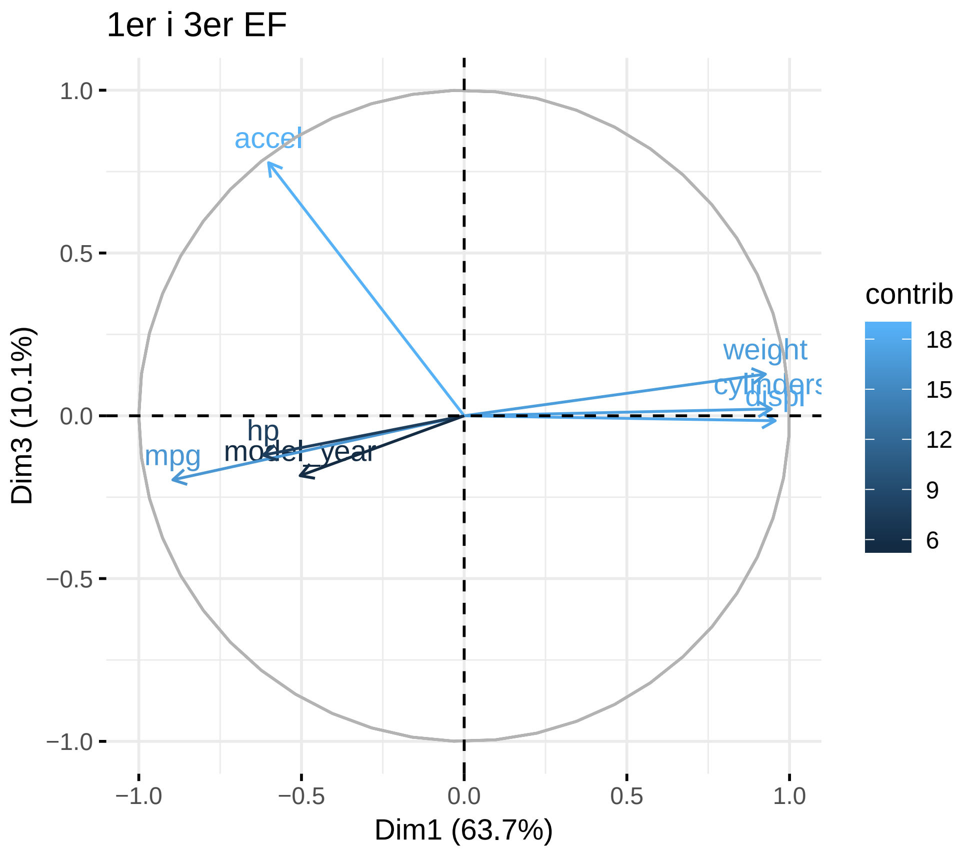
```
grafic_pca_individus <- function(dades, v_marca, marca, eix1=1, eix2=2){
  dades.PCA = dades[, c("mpg", "cylinders", "displ", "hp", "weight", "accel", "model_year",
                    marca)]
  res<-PCA(dades.PCA , scale.unit=TRUE, ncp=5, quali.sup=c(8:8), graph = FALSE)
  fviz_pca_ind(res, axes = c(eix1, eix2), habillage = v_marca)
}
grafic_pca_individus(dades, dades$marca, "marca")
```



Primer i tercer eix factorial

Com que la variable *accel* no esta ben representada en cap dels dos primers eixos, però si ho esta millor en el tercer. Es torna a dibuixar la circumferència però aquesta vegada amb el primer i tercer eix.

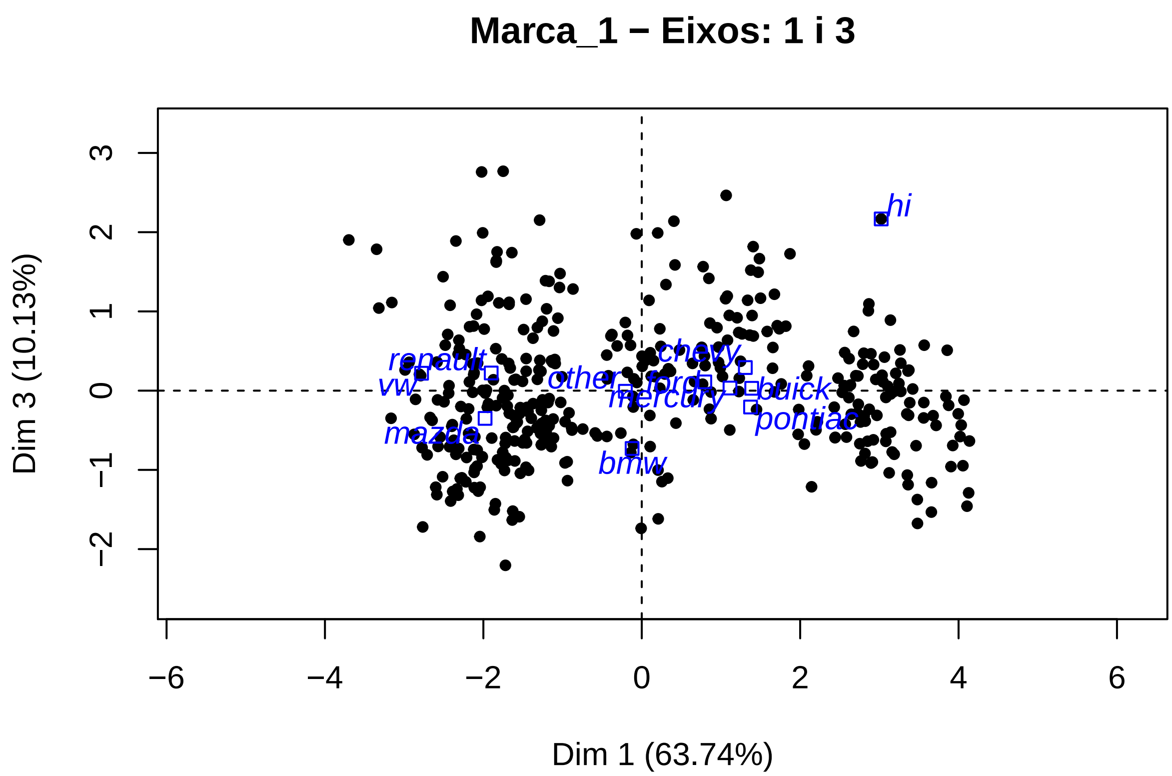
```
#Gràfic
fviz_pca_var(res, axes = c(1, 3), col.var="contrib", title="1er i 3er EF")
```



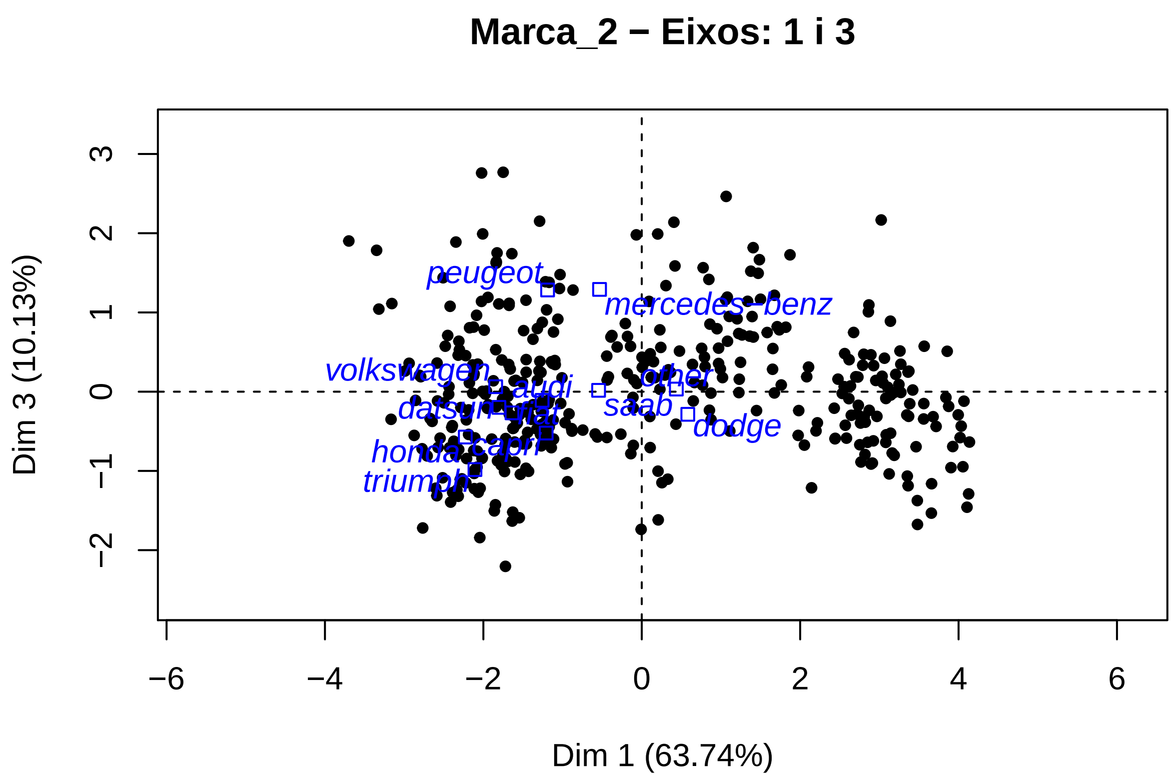
En el tercer eix es pot intuir que a mesura que augmentava l' acceleració (*accel*) augmentava el consum de combustible (*mpg*).

Distribuint les variables *marca_1*, *marca_2* i *marca_3* sobre aquests eixos factorials s'obté:

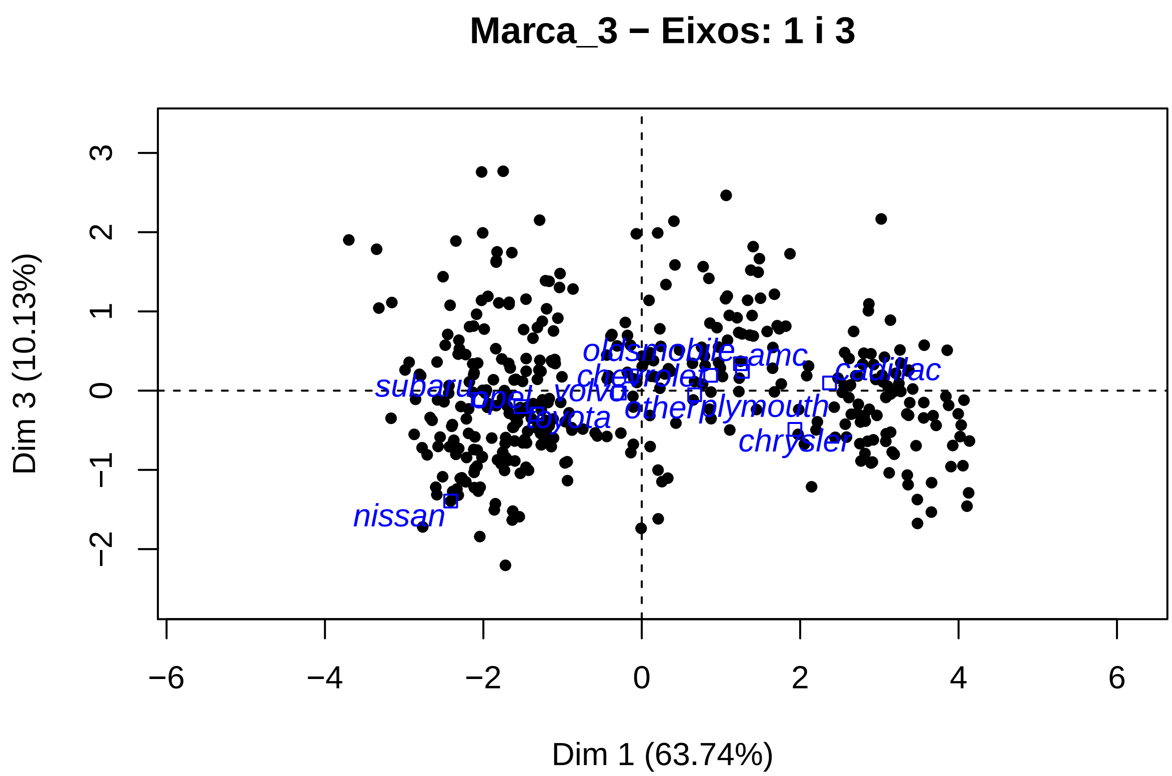
```
#Marca 1
representacio(dades, "marca_1", "Marca_1 - Eixos: 1 i 3", eix1 = 1, eix2=3)
```




```r
#Marca 2
representacio(dades, "marca_2", "Marca_2 - Eixos: 1 i 3", eix1 = 1, eix2=3)
```



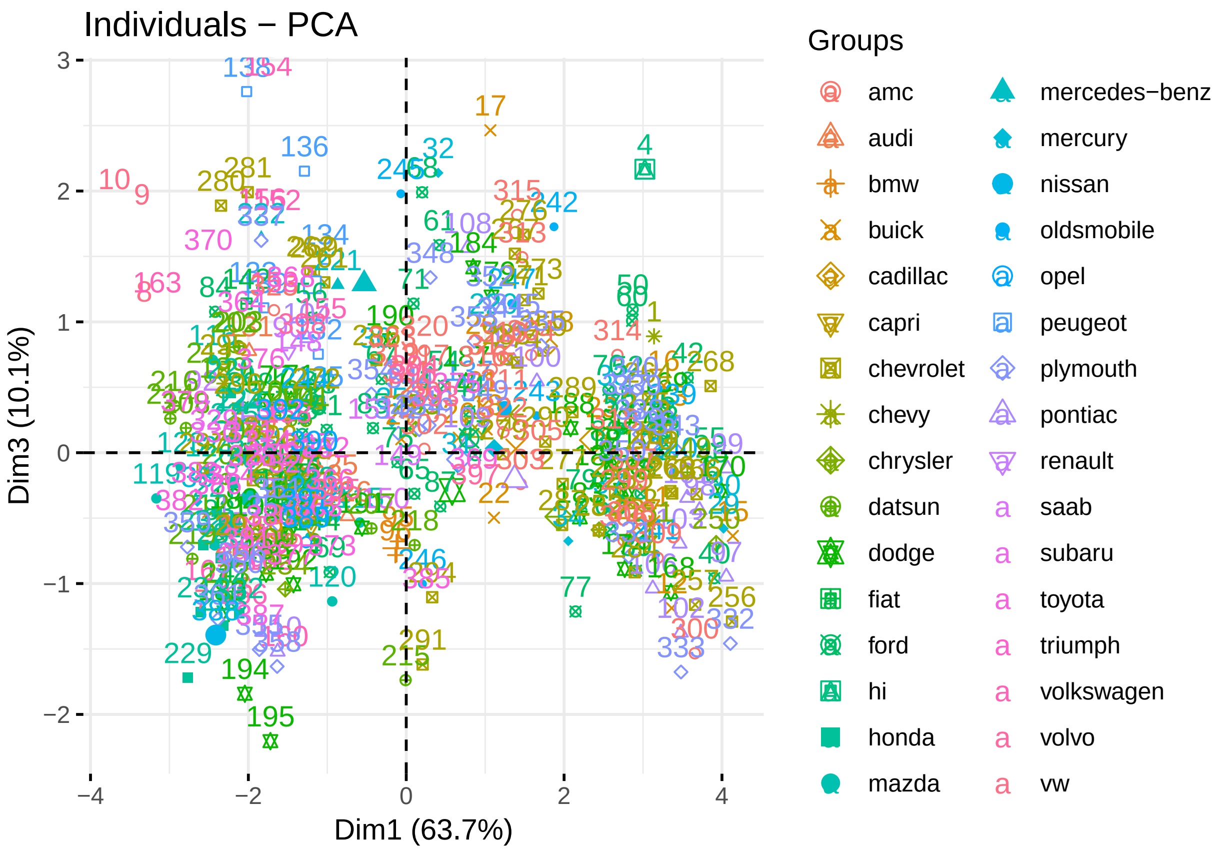

```r
#Marca 3
representacio(dades, "marca_3", "Marca_3 - Eixos: 1 i 3", eix1 = 1, eix2=3)
```



Amb aquests tres gràfics es pot veure com les marques *hi*, *mercedes-benz* i *peugeot* tendien a fabricar cotxes amb més acceleració (*accel*), mentre que les marques *triumph*, *bmw* o *nissan* fabricaven cotxes amb menys acceleració.

Distribució dels vehicles en el primer i tercer eix diferenciats per la marca

```
grafic_pca_individus(dades, dades$marca, "marca", eix1 = 1, eix2=3)
```




```
remove(dades.PCA)
remove(res)
```

Anàlisi predictiu

Per aquest anàlisi, es farà una regressió lineal simple per predir el valor de la variable *MPG*, es començarà el model amb la variable més correlacionada amb aquesta (*weight*) i s'afegiran més variables per intentar millorar el model.

Nota: Per construir el model s'ha decidit no dividir el conjunt de dades, posteriorment per fer la seva validació s'utilitzarà la tècnica del *k-cross fold validation*.

Regressió lineal simple

Per aquest tipus de models s'han de complir les següents propietats

- 1. Independència (les observacions de la mostra s'han extret amb independència entre elles).

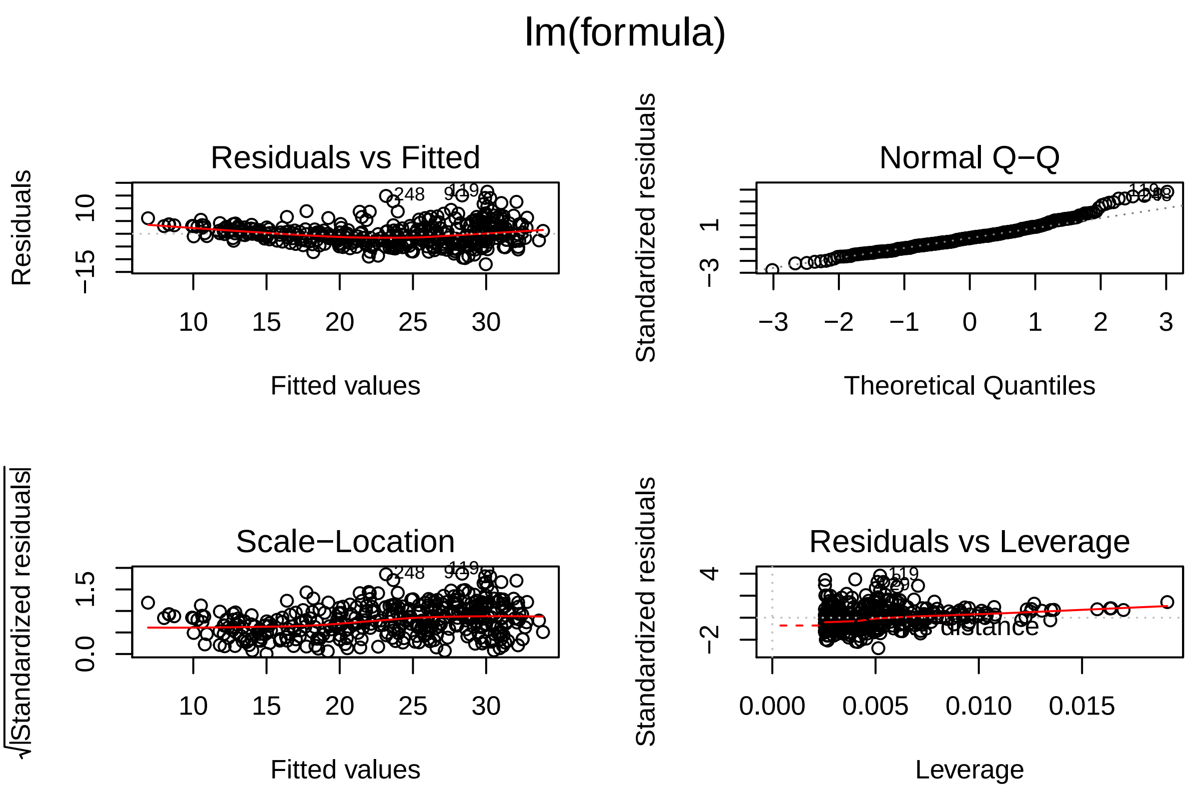
- 2. Linealitat (la variable resposta es pot ajustar linealment a partir de les variables explicatives).

- 3. Normalitat en els residus (els residus segueixen una distribució normal).

- 4. Homocedasticitat (la variància dels residus és constant).

El primer i segon punt s'assumeixen, per comprovar el tercer i quart punt es mostren les gràfiques dels residus que relacionen la variable *weight* amb la variable *mpg*.

```
parametres_reg <- function(formula, dades, titol){
  res <- lm(formula, data = dades)
  oldpar <- par(oma=c(0,0,3,0), mfrow=c(2,2))
  plot(res)
  par(oldpar)
}
parametres_reg(mpg ~ weight, dades)
```



El gràfic *Residuals vs Fitted* mostra les variàncies dels residus (punt 4) mentre que el gràfic *Normal Q-Q* mostra la distribució dels errors (Punt 3).

Per assegurar que es compleix l'homocedasticitat, es pot aplicar un test *Breuch Pagane* on les hipòtesis són:

H_0 : *Hi ha homocedasticitat*

H_1 : *No hi ha homocedasticitat*

```
#install.packages("lmtest")
library(lmtest)
bptest(mpg~weight,data=dades)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  mpg ~ weight
## BP = 22.456, df = 1, p-value = 2.15e-06
```

Com que el *p-value* del test és molt inferior al nivell de significació ($\alpha = 0.05$), amb un nivell de confiança del 95% **es pot rebutjar la hipòtesi nul·la i considerar que no hi ha homocedasticitat** (hi ha heterocedasticitat).

Per comprovar la constància en els errors, es pot aplicar un test de *Shapiro Wilk* i un altre d' *Anderson-Daling*, on en tots dos casos, les hipòtesis plantejades són:

H_0 : *Hi ha normalitat en els residus*

H_1 : *No hi ha normalitat en els residus*

```
res <- lm(mpg~weight, data = dades)
#Shapiro Wilk test
shapiro.test(residuals(res))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(res)
## W = 0.96938, p-value = 2.525e-07
```

```r
#Anderson-Daling test
#install.packages("nortest")
library(nortest)
ad.test(residuals(res))
```

```
##
##  Anderson-Darling normality test
##
## data:  residuals(res)
## A = 2.6013, p-value = 1.428e-06
```

En el test de *Shapiro Wilk* s'ha obtingut un *p-value* de $2.038 \cdot 10^{-6}$ i en el test d'*Anderson-Daling* un p-value de $2.169 \cdot 10^{-6}$, i per tant, en tots dos casos es pot **descartar la hipòtesi nul·la amb un nivell de confiança del 95%, i es pot dir que els residus no segueixen una distribució normal.**

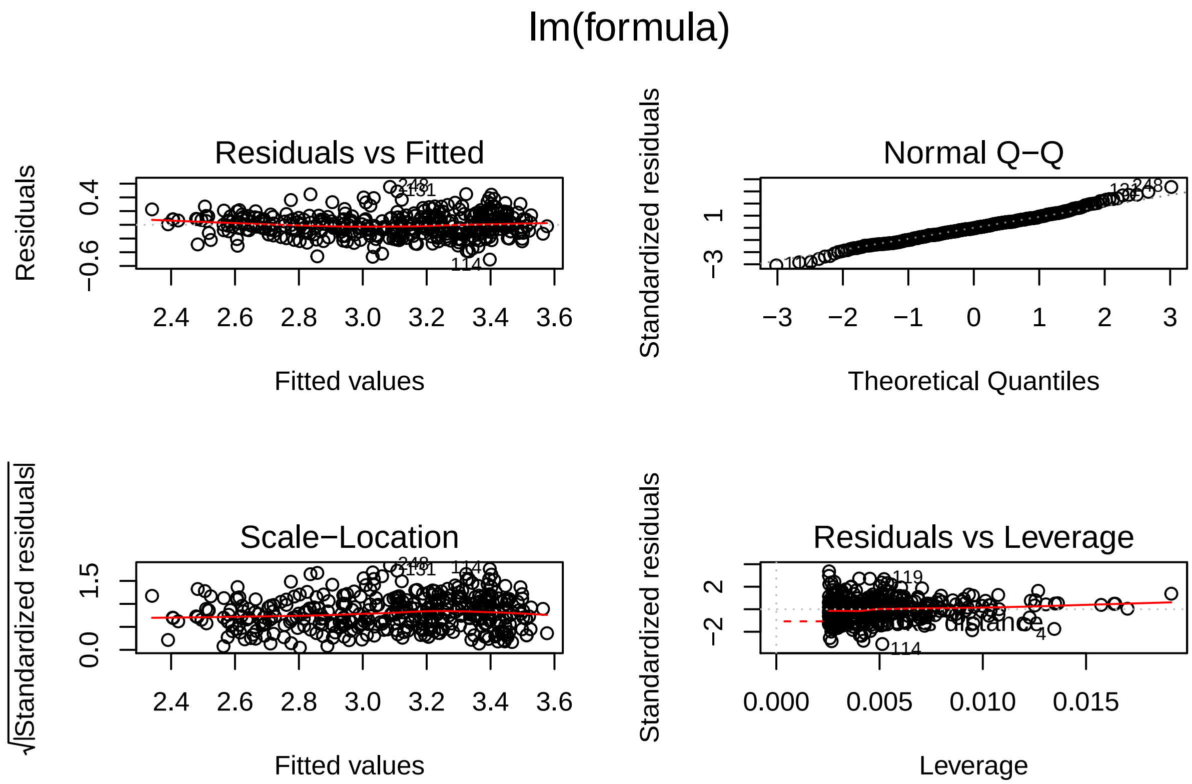
Com que s'ha demostrat que en aquest cas no es compleix ni el 3er ni el 4rt punt anunciats, la regressió que s'obtindria seria de baixa qualitat i no es podria donar un interval de confiança que acotés la predicció.

Observant la variable *mpg* **es pot veure que hi ha un biaix a la dreta, per aquest motiu pot ser que aplicant una correcció logarítmica sobre la variable resposta s'aconsegueixi normalitat en els residus i homocedasticitat**.

```r
#Correcció logarítmica en la variable mpg
dades$log_mpg <- log(dades$mpg)
```

Un cop aplicada la transformació, es torna a fer la representació gràfica, i els testos *Breuch Pagane* per l'homocedasticitat i *Shapiro Wilk* per la normalitat en els errors.

```r
#Regressió corregida
res_log <- lm(log_mpg ~ weight, data = dades)

#Gràfic
parametres_reg(formula = log_mpg ~ weight, dades = dades)
```




```r
#Homocedasticitat
bptest(log_mpg~weight,data=dades)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  log_mpg ~ weight
## BP = 2.6409, df = 1, p-value = 0.1041
```

```r
#Normalitat en els errors
shapiro.test(residuals(res_log))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(res_log)
## W = 0.99375, p-value = 0.1059
```

En aquesta ocasió s'obté un *p-value* de 0,1041 pel test de *Breuch Pagane*, i per tant, **amb un nivell de confiança del 95% no hi ha suficients evidències per rebutjar la hipòtesi nul · la, i per això s'assumeix homocedasticitat**. Per altre banda, en el test de *Shapiro Wilk* s'obté un *p-value* de 0,1059, i per tant, **amb un nivell de confiança del 95% no es pot rebutjar la hipòtesi nul · la, i s'assumeix que els errors segueixen una distribució normal**.

La forma que pren aquesta primera regressió és:

$$\hat{y} = 4.142 + -3.505 \cdot 10^{-4} \cdot x_1 + \epsilon$$

És a dir:

$$mpg = 4.142 + -3.505 \cdot 10^{-4} \cdot weight + \epsilon$$

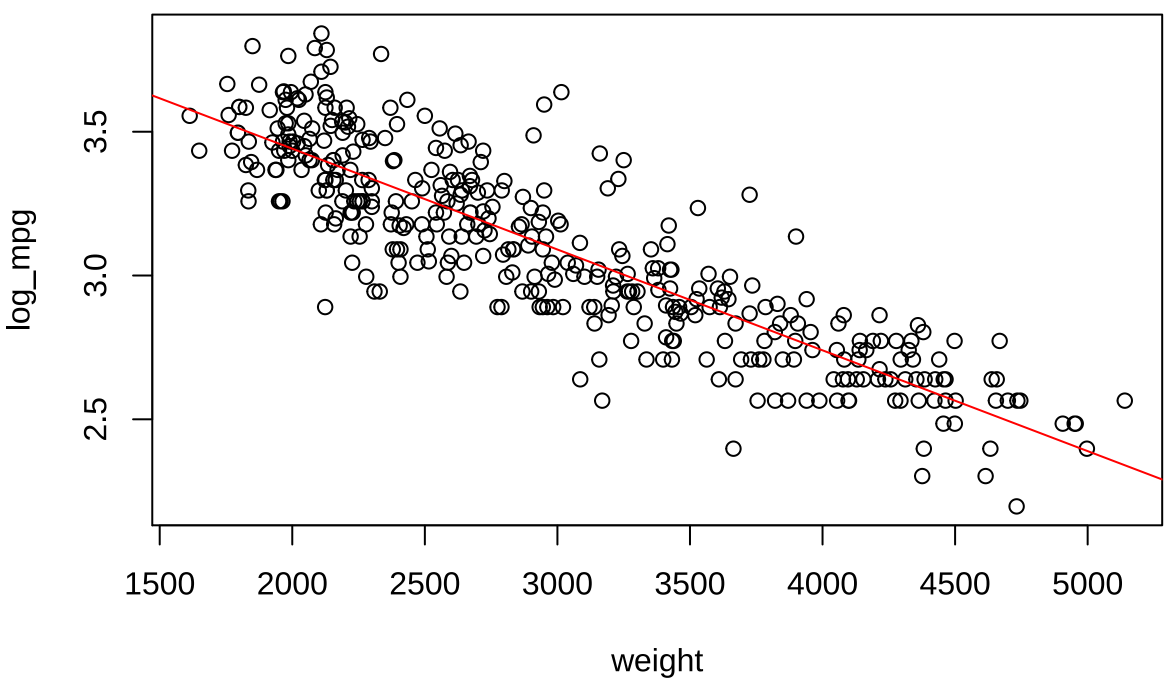
Tot i que el primer coeficient de la regressió (β_1) és molt pròxim a 0, s'ha obtingut un *Adjusted* R^2 de 0,7662.

```r
#Dades resultants de la regressió
summary(res_log)
```

```
##
## Call:
## lm(formula = log_mpg ~ weight, data = dades)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50716 -0.09966 -0.00621  0.09973  0.55239
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.142e+00  3.031e-02  136.66   <2e-16 ***
## weight      -3.505e-04  9.790e-06  -35.81   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1644 on 390 degrees of freedom
## Multiple R-squared:  0.7668, Adjusted R-squared:  0.7662
## F-statistic:  1282 on 1 and 390 DF,  p-value: < 2.2e-16
```

Ara es representa la regressió amb un *scatter plot*

```r
#Representació de la regressió
res_log = lm(log_mpg~weight,dades)
plot(log_mpg~weight, data=dades)
abline(res_log, col='red')
```

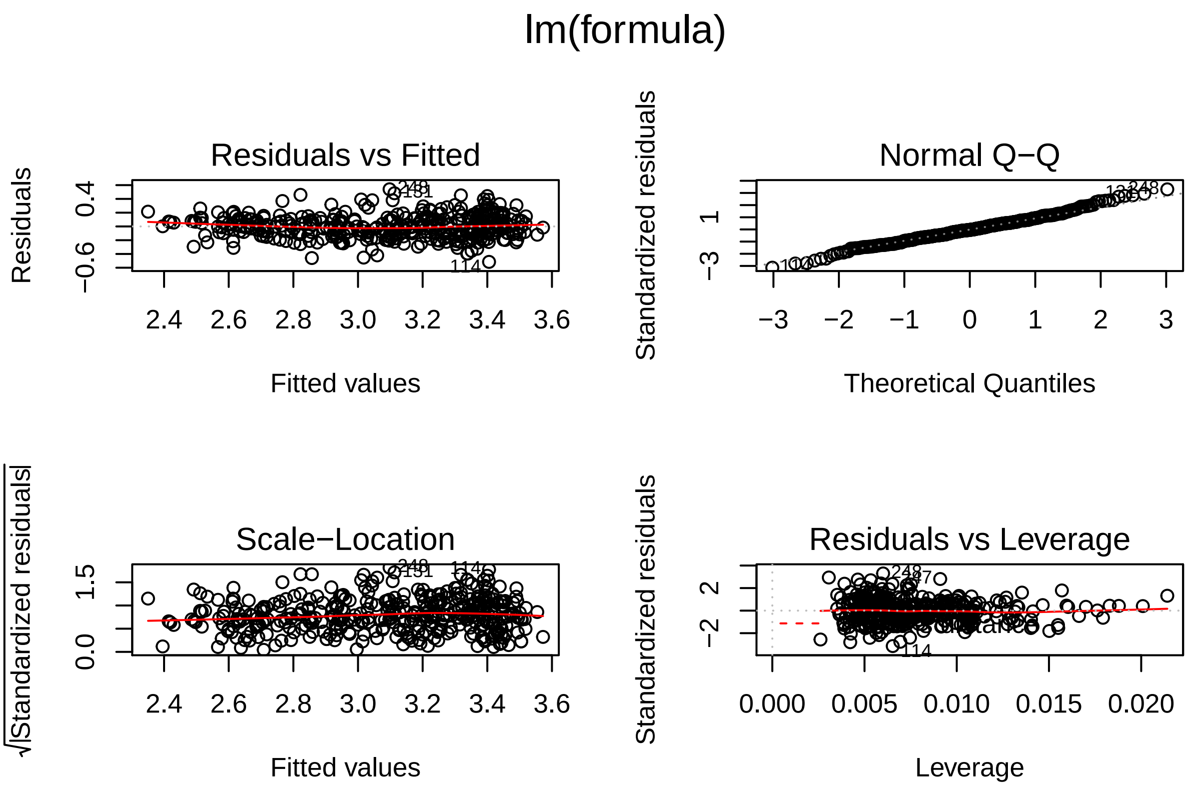


Amb aquest gràfic es pot tornar a veure com a mesura que augmenta el pes, la variable *mpg* tendeix a decréixer.

Ara es poden afegir més variables al model (passarà de ser una regressió lineal simple a una regressió lineal múltiple) per tal de millorar-lo, per això, s'ha de provar d'afegir aquelles variables poc relacionades amb les que ja inclou el model (que intentin explicar el que encara no està explicat) i que estiguin el màxim de correlacionades amb la variable resposta.

Es prova d'afegir la variable *hp* i es tornen a fer els testos de homocedasticitat i normalitat d'errors (cal tornar a fer-los per assegurar que no es perd qualitat amb les noves variables).

```
#Nova regressió
res_log <- lm(log_mpg ~ weight + hp, dades)

#Gràfics
parametres_reg(log_mpg ~ weight + hp, dades)
```




```
#Homocedasticitat
bptest(res_log)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  res_log
## BP = 3.1615, df = 2, p-value = 0.2058
```

```
#Normalitat en els errors
shapiro.test(residuals(res_log))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(res_log)
## W = 0.99327, p-value = 0.07722
```

En el test de constància en els errors s'obté un *p-value* de 0,2058, i per tant, es continua assumint homocedasticitat, i en el test de normalitat dels residus s'obté un *p-value* de 0,07722 i es segueix assumint normalitat en els residus.

El nou model queda:

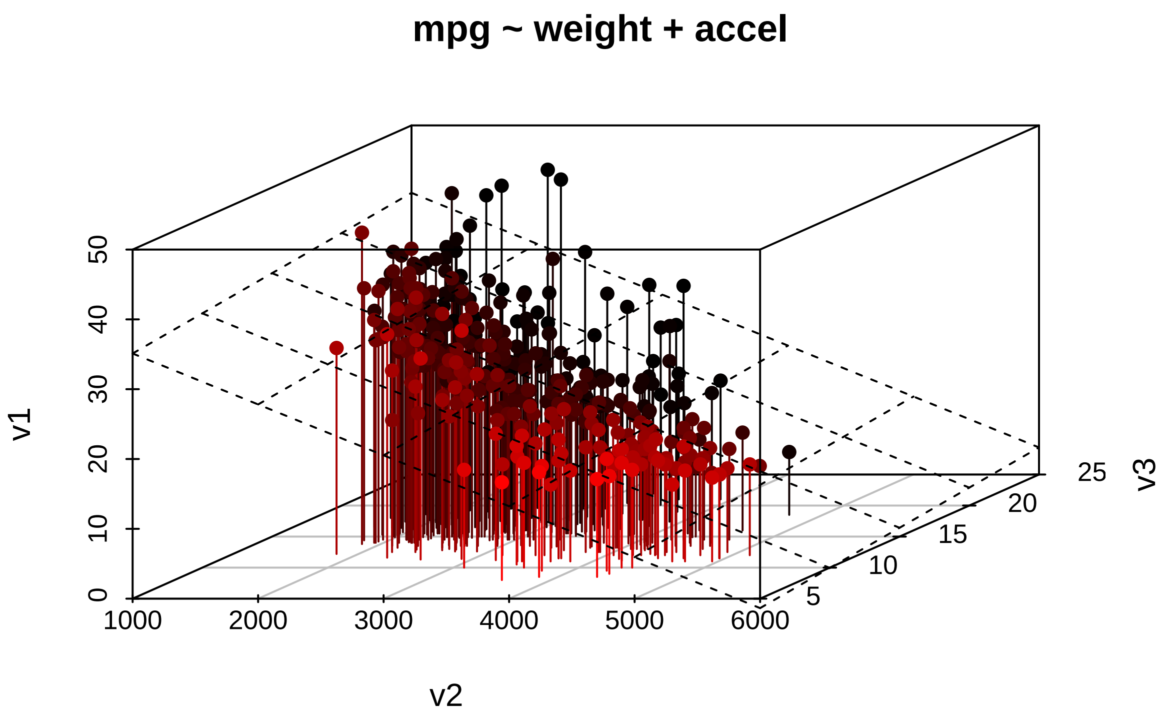
$$\hat{y} = 4.098 + -3.431 \cdot 10^{-4} \cdot x_1 + 4.2 \cdot 10^{-4} \cdot x_2 + \epsilon$$

És a dir:

$$mpg = 4.098 + -3.431 \cdot 10^{-4} \cdot weight + 4.2 \cdot 10^{-4} \cdot hp + \epsilon$$

Com ara hi ha tres variables involucrades, cal fer un gràfic 3D per representar-lo.

```
library(ggplot2)
plot_reg_3d <- function(v1,v2,v3){
  library(scatterplot3d)
  attach(mtcars)
  s3d <-scatterplot3d(v2,v3,v1, pch=16, highlight.3d=TRUE,
    type="h", main="mpg ~ weight + accel")
  fit <- lm(v1 ~ v2+v3)
  s3d$plane3d(fit)
}
plot_reg_3d(dades$mpg, dades$weight, dades$accel)
```



Per veure com de bo és aquest model, es consulta altre vegada el valor del paràmetre $Adjusted R^2$

```
summary(res_log)
```

```
##
## Call:
## lm(formula = log_mpg ~ weight + hp, data = dades)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.51505 -0.10357 -0.00648  0.09882  0.54000
##
## Coefficients:
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.098e+00  4.573e-02  89.622   <2e-16 ***
## weight      -3.431e-04  1.136e-05 -30.206   <2e-16 ***
## hp           4.200e-04  3.271e-04   1.284      0.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1643 on 389 degrees of freedom
## Multiple R-squared:  0.7678, Adjusted R-squared:  0.7666
## F-statistic:   643 on 2 and 389 DF,  p-value: < 2.2e-16
```

Aquest valor ara és de 0,7666 (en vers al 0,7662 obtingut només amb la variable *weight*), per veure si aquesta millora és significativa, es pot aplicar un test *ANOVA* on les hipòtesis són:

$$H_0 : \beta_2 = 0$$

$$H_1 : \beta_2 \neq 0$$

```r
anova(lm(mpg~weight, dades), lm(mpg~weight+hp, dades))
```

```
## Analysis of Variance Table
##
## Model 1: mpg ~ weight
## Model 2: mpg ~ weight + hp
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    390 7321.2
## 2    389 7297.7  1    23.577 1.2568  0.263
```

Com que el *p-value* és 0,263, **amb un 95% de confiança no es pot rebutjar la hipòtesi nul · la, i per tant, es considerà que la contribució de la variable *hp* al model és 0**.

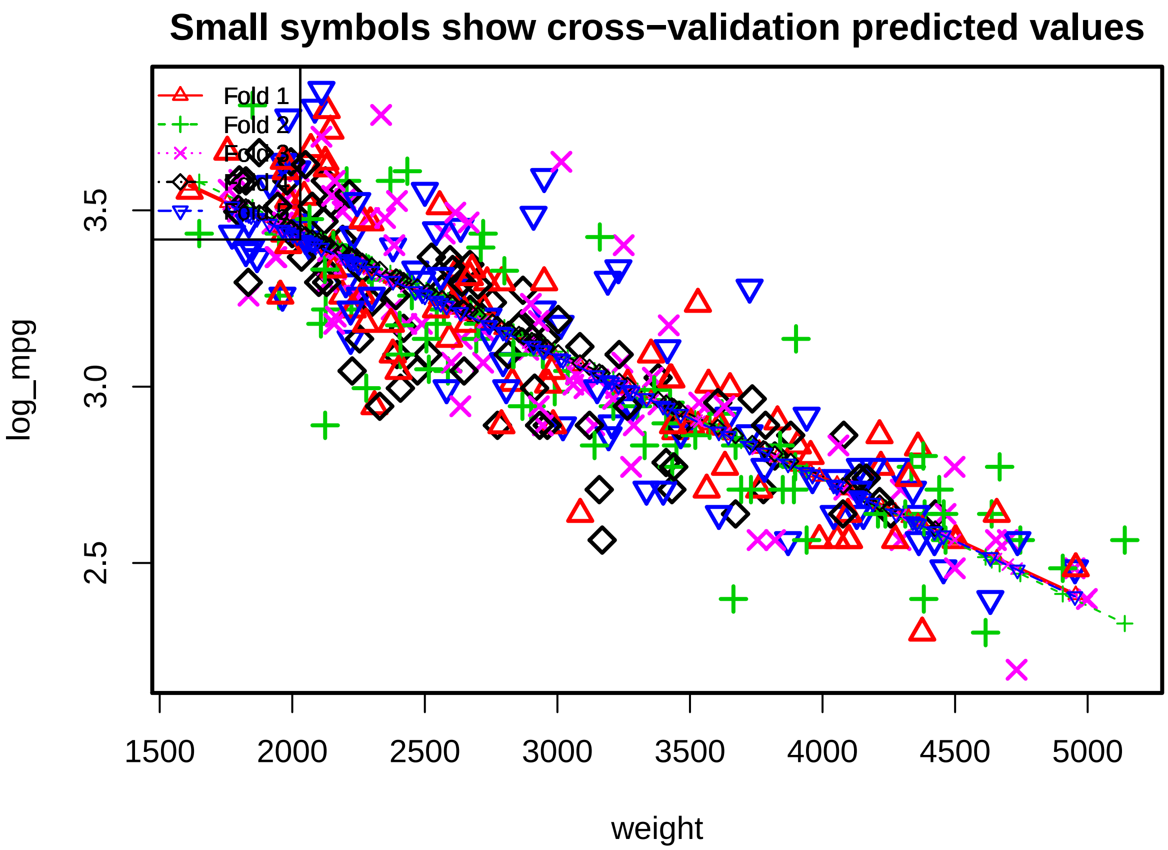
Degut que la inclusió de la variable *hp* no ha portat una millora significant, es decideix utilitzar el model sense aquesta variable i aquest finalment queda de la següent manera:

$$mpg = 4.142 + -3.505 \cdot 10^{-4} \cdot weight + \epsilon$$

Eficiència del model

Ara es vol comprovar de forma empírica com de bé, el model prediu el consum a partir del pes i per això s'utilitza la tècnica del *k-fold cross validate*. Com que s'aconsella utilitzar un 80% de les dades (314 observacions) per construir el model i el 20% per testejar-lo (78 observacions), el valor del k serà 5.

```r
library(DAAG)

#k-fold cross validation amb k=5
kfold = cv.lm(data=dades, lm(log_mpg ~ weight, dades), m=5, printit = FALSE)
```



Es calcula l'error comés en la validació.

```
errors_log = mean(sqrt(sum(kfold$log_mpg-kfold$cvpred)^2))

#En unitats mpg
errors = exp(errors_log)

#mitja de la variable mpg
mitja_real = mean(dades$mpg)

#Error relatiu
(mitja_real-errors)/mitja_real
```

```
## [1] 0.950176
```

```
remove(cor.mat); remove(dades); remove(res); remove(res_log); remove(kfold); remove(errors_log); remove
```

L'error comés en aquesta predicció ronda 1,179 milles per galó, tenint en compte que la mitja de totes les observacions és 23,45, i per tant s'ha comés un error relatiu del 0.95.