AEMOD: Ejercicio 1 (Fichero Auto)

Inmaculada Perea Fernández

Febrero 2017

Con el fichero Auto de la librería ISLR seleccionar los vehículos con mpg>=13

Proponer un modelo que identifique qué variables influyen en la nueva variable de conteo: m_13=round(mpg-13).

1. Carga de librerías necesarias

```
library(ISLR)
library(ggplot2)
library(MASS)
```

2. Obtención e inspección del conjunto de datos para el estudio

El fichero Auto tiene las siguientes variables:

- mpg: miles per gallon
- cylinders: Number of cylinders between 4 and 8
- displacement: Engine displacement (cu. inches)
- horsepower: Engine horsepower
- weight: Vehicle weight (lbs.)
- acceleration: Time to accelerate from 0 to 60 mph (sec.)
- year: Model year (modulo 100)
- origin: Origin of car (1. American, 2. European, 3. Japanese)
- name: Vehicle name

```
data(Auto)
str(Auto)
```

```
## 'data.frame':    392 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : num  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : num  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : num  3504 3693 3436 3433 3449 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : num  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ name        : Factor w/ 304 levels "amc ambassador brougham",..: 49 36 231 14 161 141 54 223 241 
```

```
head(Auto)
```

```
##   mpg cylinders displacement horsepower weight acceleration year origin
## 1  18         8          307        130   3504         12.0   70      1
## 2  15         8          350        165   3693         11.5   70      1
## 3  18         8          318        150   3436         11.0   70      1
## 4  16         8          304        150   3433         12.0   70      1
```

```
## 5  17         8          302        140   3449          10.5   70      1
## 6  15         8          429        198   4341          10.0   70      1
##                        name
## 1 chevrolet chevelle malibu
## 2         buick skylark 320
## 3        plymouth satellite
## 4             amc rebel sst
## 5               ford torino
## 6          ford galaxie 500
```

```
dim(Auto)
```

```
## [1] 392   9
```

```
summary(Auto)
```

```
##       mpg          cylinders      displacement     horsepower
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0
##
##      weight      acceleration        year           origin
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
##  1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000
##  Median :2804   Median :15.50   Median :76.00   Median :1.000
##  Mean   :2978   Mean   :15.54   Mean   :75.98   Mean   :1.577
##  3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000
##
##                  name
##  amc matador       :  5
##  ford pinto        :  5
##  toyota corolla    :  5
##  amc gremlin       :  4
##  amc hornet        :  4
##  chevrolet chevette:  4
##  (Other)           :365
```

2.1 Construcción del conjunto de datos sobre el que realiza el análisis.

Filtramos el conjunto de datos original para quedarnos con el subconjunto correspondiente al consumo mpg>= 13 y eliminar las variables mpg y name

```
mpg_ge_13<-Auto[I(Auto$mpg>=13),]
head(mpg_ge_13)
```

```
##   mpg cylinders displacement horsepower weight acceleration year origin
## 1  18         8          307        130   3504         12.0   70      1
## 2  15         8          350        165   3693         11.5   70      1
## 3  18         8          318        150   3436         11.0   70      1
## 4  16         8          304        150   3433         12.0   70      1
## 5  17         8          302        140   3449         10.5   70      1
```

```
## 6  15         8          429        198   4341         10.0   70      1
##                         name
## 1 chevrolet chevelle malibu
## 2         buick skylark 320
## 3        plymouth satellite
## 4             amc rebel sst
## 5               ford torino
## 6          ford galaxie 500
```

```r
dim(mpg_ge_13)
```

```
## [1] 379   9
```

```r
summary(mpg_ge_13)
```

```
##       mpg          cylinders      displacement     horsepower
##  Min.   :13.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:18.00   1st Qu.:4.000   1st Qu.: 99.5   1st Qu.: 75.0
##  Median :23.00   Median :4.000   Median :141.0   Median : 92.0
##  Mean   :23.87   Mean   :5.385   Mean   :188.3   Mean   :101.5
##  3rd Qu.:29.25   3rd Qu.:6.000   3rd Qu.:258.0   3rd Qu.:115.0
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0
##
##      weight      acceleration        year           origin
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
##  1st Qu.:2220   1st Qu.:14.00   1st Qu.:73.00   1st Qu.:1.000
##  Median :2745   Median :15.50   Median :76.00   Median :1.000
##  Mean   :2921   Mean   :15.63   Mean   :76.12   Mean   :1.596
##  3rd Qu.:3512   3rd Qu.:17.20   3rd Qu.:79.00   3rd Qu.:2.000
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000
##
##                 name
##  amc matador       :  5
##  ford pinto        :  5
##  toyota corolla    :  5
##  amc gremlin       :  4
##  amc hornet        :  4
##  chevrolet chevette:  4
##  (Other)           :352
```

```r
data_auto = data.frame(m_13=round(mpg_ge_13$mpg-13), mpg_ge_13[2:8])
head(data_auto)
```

```
##   m_13 cylinders displacement horsepower weight acceleration year origin
## 1    5         8          307        130   3504         12.0   70      1
## 2    2         8          350        165   3693         11.5   70      1
## 3    5         8          318        150   3436         11.0   70      1
## 4    3         8          304        150   3433         12.0   70      1
## 5    4         8          302        140   3449         10.5   70      1
## 6    2         8          429        198   4341         10.0   70      1
```

```r
dim(data_auto)
```

```
## [1] 379   8
```

```r
str(data_auto)
```

```
## 'data.frame':    379 obs. of  8 variables:
```

```
##  $ m_13        : num  5 2 5 3 4 2 1 1 1 2 ...
##  $ cylinders   : num  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : num  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : num  3504 3693 3436 3433 3449 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : num  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : num  1 1 1 1 1 1 1 1 1 1 ...
```

```r
summary(data_auto)
```

```
##       m_13         cylinders      displacement     horsepower   
##  Min.   : 0.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0  
##  1st Qu.: 5.00   1st Qu.:4.000   1st Qu.: 99.5   1st Qu.: 75.0  
##  Median :10.00   Median :4.000   Median :141.0   Median : 92.0  
##  Mean   :10.85   Mean   :5.385   Mean   :188.3   Mean   :101.5  
##  3rd Qu.:16.00   3rd Qu.:6.000   3rd Qu.:258.0   3rd Qu.:115.0  
##  Max.   :34.00   Max.   :8.000   Max.   :455.0   Max.   :230.0  
##      weight      acceleration        year           origin     
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000  
##  1st Qu.:2220   1st Qu.:14.00   1st Qu.:73.00   1st Qu.:1.000  
##  Median :2745   Median :15.50   Median :76.00   Median :1.000  
##  Mean   :2921   Mean   :15.63   Mean   :76.12   Mean   :1.596  
##  3rd Qu.:3512   3rd Qu.:17.20   3rd Qu.:79.00   3rd Qu.:2.000  
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000  
```

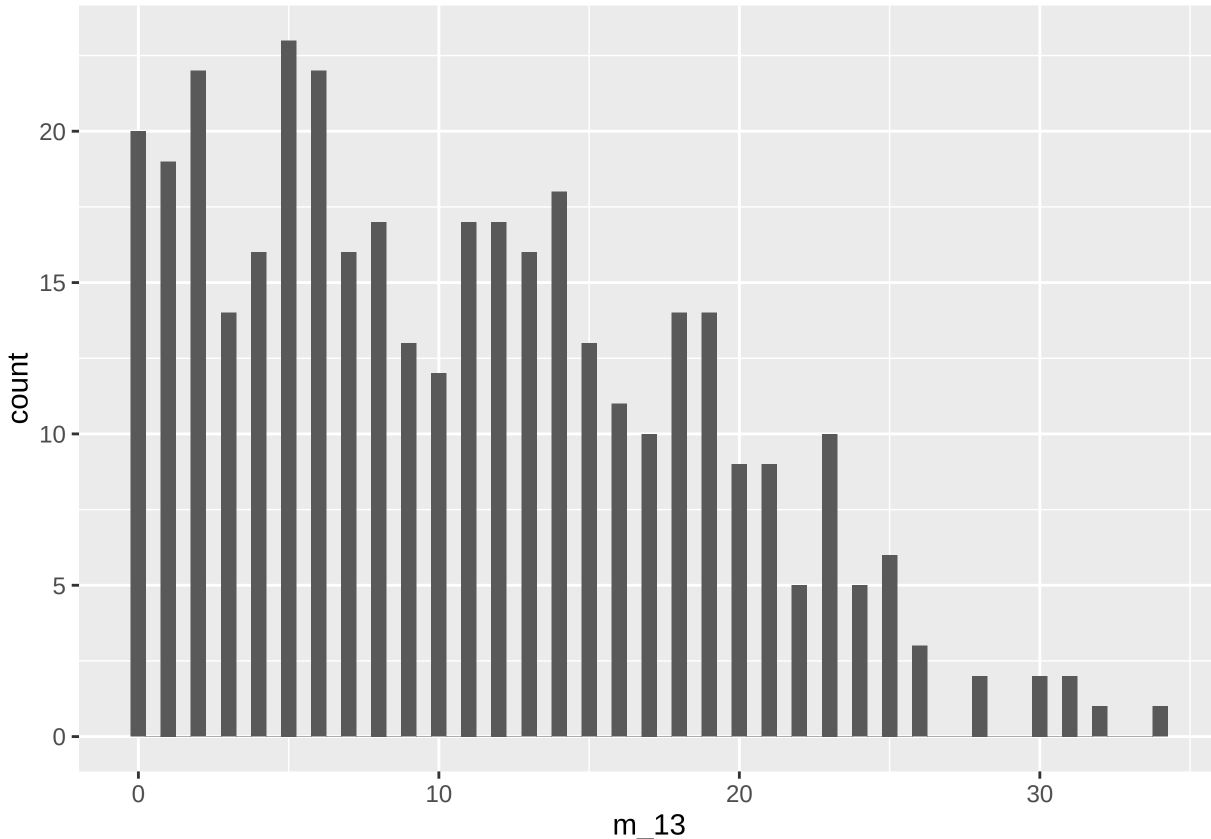
Observamos que la variable *origin* es tipo categórica pero en el fichero no está representada como tal, vamos a usar la función factor para representarla correctamente. Consultando la ayuda de R para el dataset Auto, vemos que la categorización es la siguiente: 1 = American 2 = European 3 = Japanese)

```r
data_auto <- within(data_auto, {origin <- factor(origin, levels=1:3,
                                labels=c("American", "European", "Japanese"))})

summary(data_auto)
```

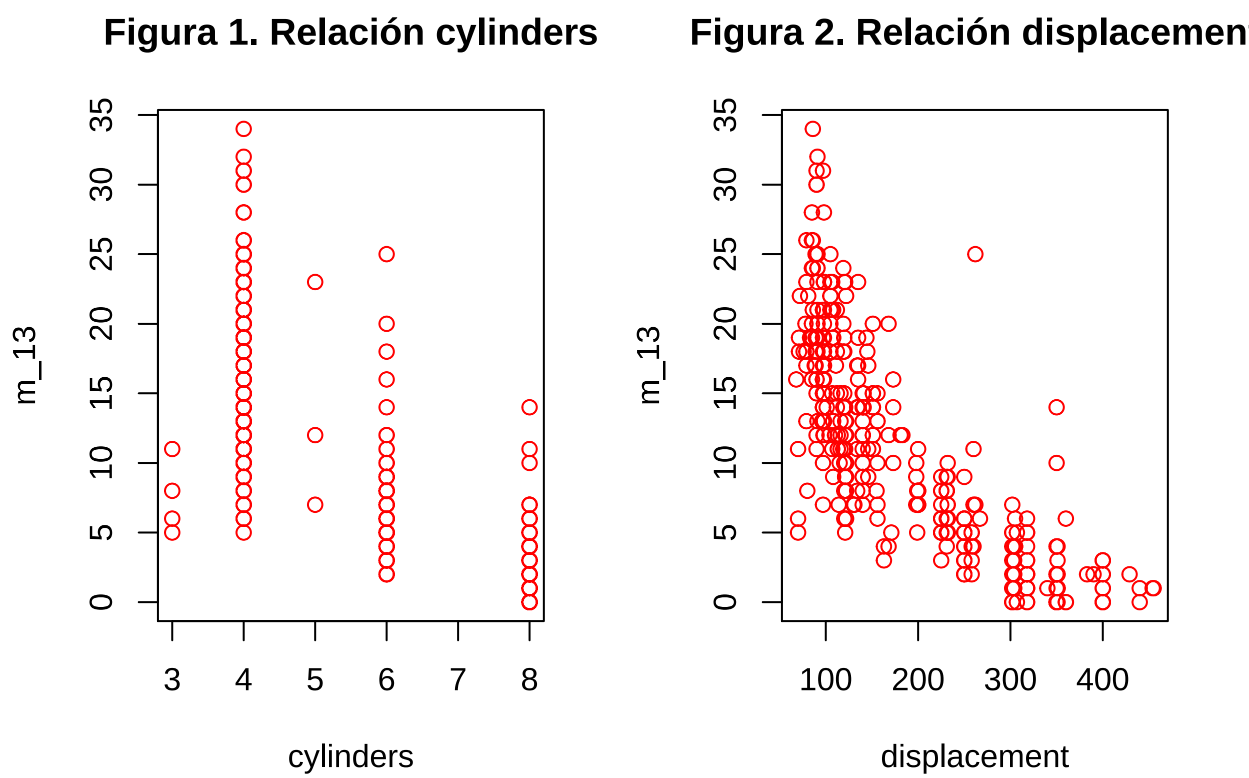
```
##       m_13         cylinders      displacement     horsepower   
##  Min.   : 0.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0  
##  1st Qu.: 5.00   1st Qu.:4.000   1st Qu.: 99.5   1st Qu.: 75.0  
##  Median :10.00   Median :4.000   Median :141.0   Median : 92.0  
##  Mean   :10.85   Mean   :5.385   Mean   :188.3   Mean   :101.5  
##  3rd Qu.:16.00   3rd Qu.:6.000   3rd Qu.:258.0   3rd Qu.:115.0  
##  Max.   :34.00   Max.   :8.000   Max.   :455.0   Max.   :230.0  
##      weight      acceleration        year            origin   
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   American:232  
##  1st Qu.:2220   1st Qu.:14.00   1st Qu.:73.00   European: 68  
##  Median :2745   Median :15.50   Median :76.00   Japanese: 79  
##  Mean   :2921   Mean   :15.63   Mean   :76.12                 
##  3rd Qu.:3512   3rd Qu.:17.20   3rd Qu.:79.00                 
##  Max.   :5140   Max.   :24.80   Max.   :82.00                 
```

2.2 Inspección gráfica de la relación con la variable objetivo *m_13*

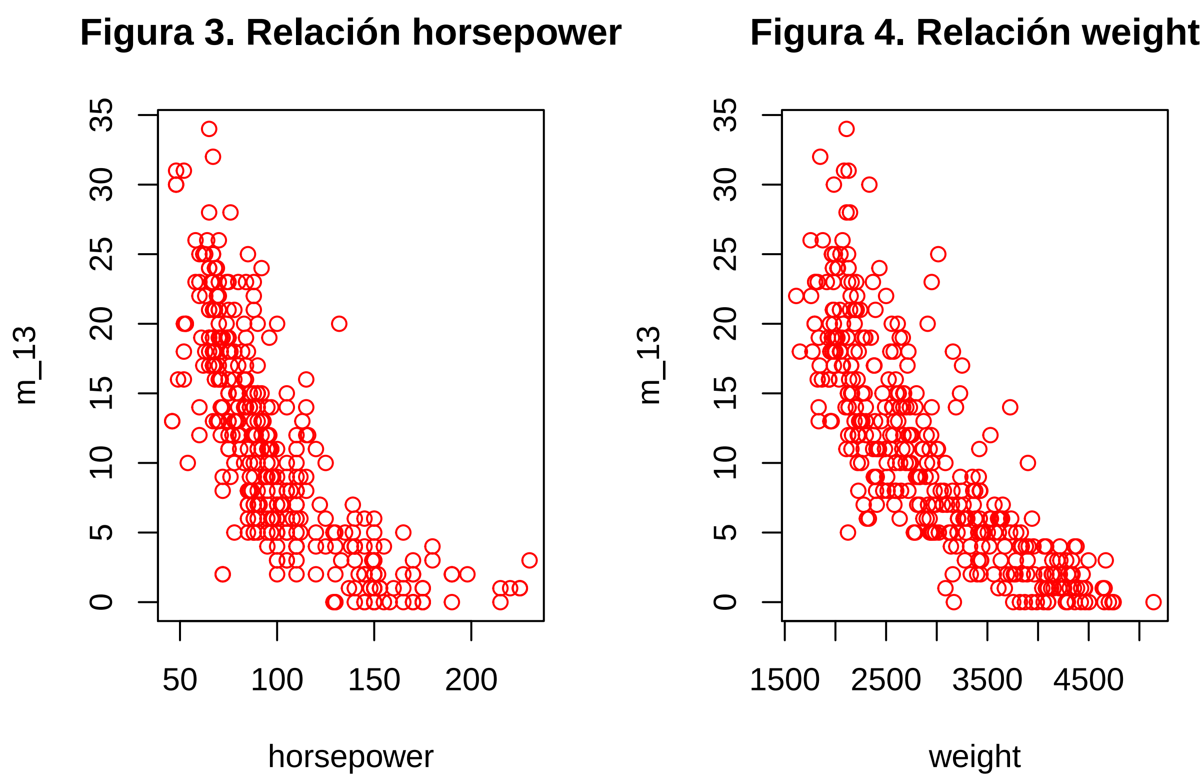
```r
ggplot(data_auto, aes(m_13)) +  geom_histogram(binwidth=.5, position="dodge")
```



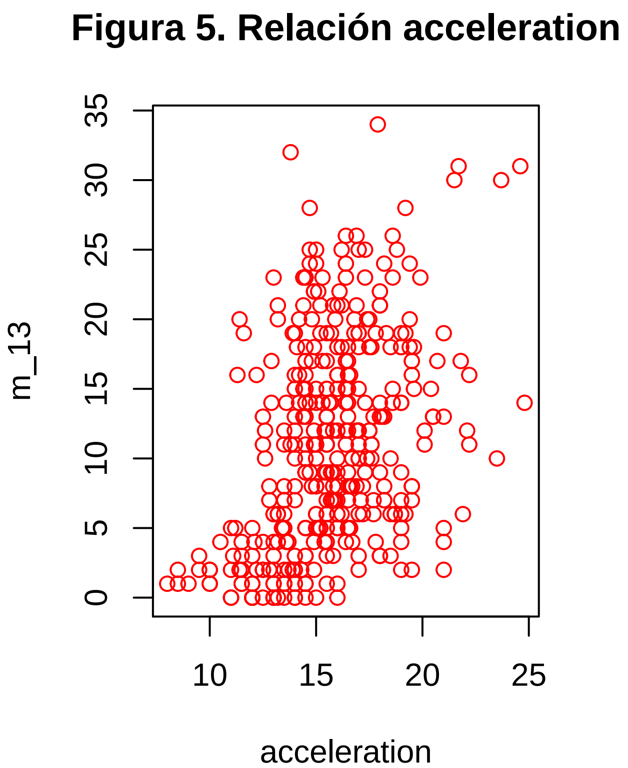

```
par(mfrow=c(1,2))
plot(data_auto$cylinders, data_auto$m_13,
     xlab="cylinders",  ylab="m_13",
     main="Figura 1. Relación cylinders",
     col="red")

plot(data_auto$displacement, data_auto$m_13,
     xlab="displacement",  ylab="m_13",
     main="Figura 2. Relación displacement",
     col="red")
```

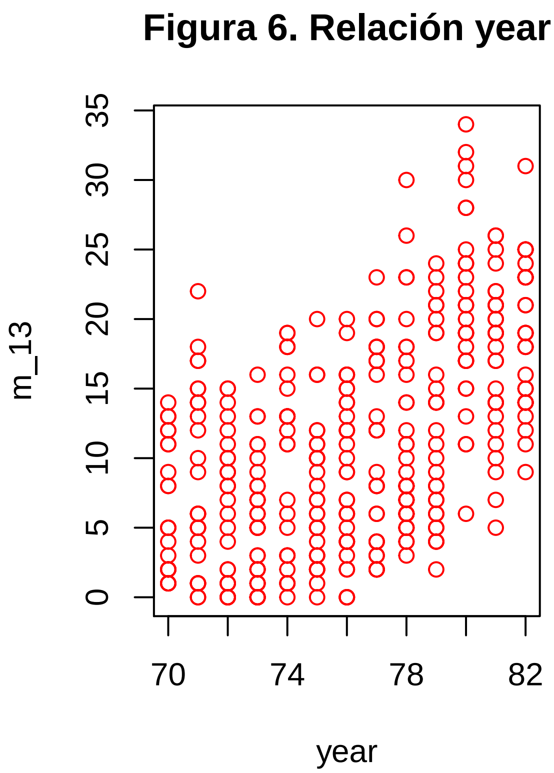



```
par(mfrow=c(1,2))

plot(data_auto$horsepower, data_auto$m_13,
     xlab="horsepower",  ylab="m_13",
     main="Figura 3. Relación horsepower",
     col="red")

plot(data_auto$weight, data_auto$m_13,
     xlab="weight",  ylab="m_13",
     main="Figura 4. Relación weight",
     col="red")
```

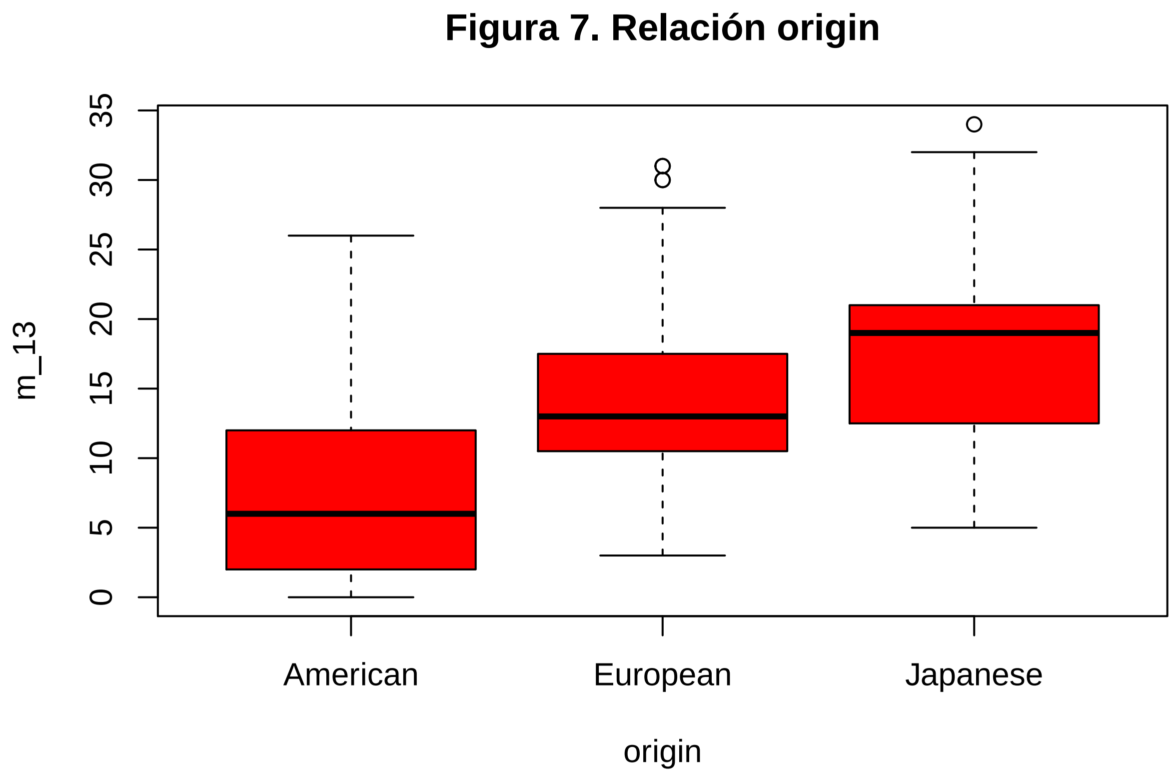



```
par(mfrow=c(1,2))

plot(data_auto$acceleration, data_auto$m_13,
     xlab="acceleration",  ylab="m_13",
     main="Figura 5. Relación acceleration",
     col="red")

plot(data_auto$year, data_auto$m_13,
     xlab="year",  ylab="m_13",
     main="Figura 6. Relación year",
     col="red")
```

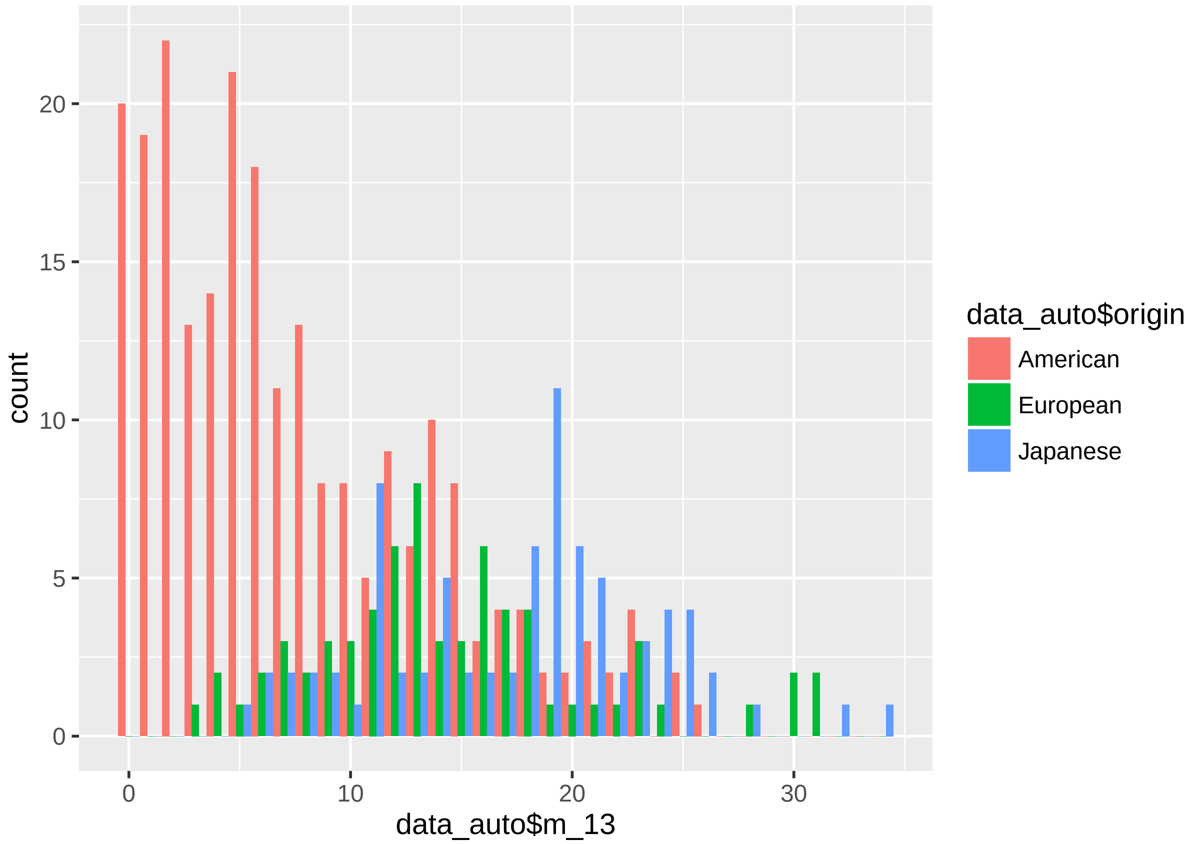





```
plot(data_auto$origin, data_auto$m_13,
     xlab="origin",  ylab="m_13",
     main="Figura 7. Relación origin",
     col="red")
```




```
ggplot(data_auto, aes(data_auto$m_13, fill = data_auto$origin)) +
  geom_histogram(binwidth=1, position="dodge")
```



3. Construcción del modelo

La variable objetivo m_13 es una variable de conteo, dos de las distribuciones empleadas para modelar datos de conteo son la Poisson y la Binomial Negativa.

Cuando la variable puede tomar valores desde cero y no tiene una cota superior una posible distribuciónn es la de Poisson.

Si no se cumple la condición de igualdad entre la media y la varianza de la distribución la Binomial negativa puede ser el modelo más adecuado.

Vamos a comparar la media y la varianza de la variable objetivo m_13

```
media=mean(data_auto$m_13)
varianza=var(data_auto$m_13)

cat("varianza =", round(varianza, 2))
```

```
## varianza = 57.49
```

```
cat("media =", round(media,2))
```

```
## media = 10.85
```

Observamos que no es razonable asumir que la media y la varianza son semejantes. La varianza y la media son distintas. La varianza es mayor que la media, por tanto existe sobredispersión y es mas adecuado aplicar el modelo binommial negativo.

A continuación estudiaremos la influencia o no de cada una de las variables con el modelo binomial negativo y finalmente lo compararemos con el modelo de Poisson.

3.1 Modelo binomial negativo con todas las variables

Construimos el modelo binomial negativo con todas las variables del conjunto de datos

```r
mfull = glm.nb(m_13 ~ ., data = data_auto)
(sum.mfull=summary(mfull))
```

```
##
## Call:
## glm.nb(formula = m_13 ~ ., data = data_auto, init.theta = 115437.6735,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0553  -0.6566  -0.0408   0.4888   3.5579
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -8.381e-01  4.591e-01  -1.826  0.06792 .
## cylinders      -1.419e-02  3.906e-02  -0.363  0.71643
## displacement   -1.220e-03  1.011e-03  -1.207  0.22729
## horsepower     -4.964e-03  1.805e-03  -2.751  0.00594 **
## weight         -5.193e-04  8.385e-05  -6.193 5.91e-10 ***
## acceleration    3.951e-03  9.384e-03   0.421  0.67370
## year            6.799e-02  4.825e-03  14.090  < 2e-16 ***
## originEuropean  8.189e-02  4.774e-02   1.716  0.08625 .
## originJapanese  3.219e-02  4.481e-02   0.718  0.47256
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(115437.7) family taken to be 1)
##
##     Null deviance: 2218.33  on 378  degrees of freedom
## Residual deviance:  336.74  on 370  degrees of freedom
## AIC: 1800.1
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  115438
##           Std. Err.:  685324
## Warning while fitting theta: iteration limit reached
##
##  2 x log-likelihood:  -1780.059
```

3.2 Paso 1: Modelo eliminando 1 variable del modelo completo

En este paso tomaré el modelo completo (*mfull*) e iré eliminando en cada etapa una de las variables para medir qué influencia tiene sobre la variable objetivo

3.2.1 Eliminando la variable *cylinders* del modelo completo

```
mfull.cylinders <- update(mfull, . ~ . - cylinders)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
sum.cilynders=summary(mfull.cylinders)
(anov.cilynders=anova(mfull, mfull.cylinders))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                                                 Model
## 1              displacement + horsepower + weight + acceleration + year + origin
## 2 cylinders + displacement + horsepower + weight + acceleration + year + origin
##      theta Resid. df    2 x log-lik.   Test    df  LR stat.   Pr(Chi)
## 1 115122.7       371       -1780.191
## 2 115437.7       370       -1780.059 1 vs 2     1 0.1321171 0.7162469
```

Observamos que la variable *cylinders* no influye en el consumo.

3.2.2 Eliminando la variable *displacement* del modelo completo

```
mfull.displacement <- update(mfull, . ~ . - displacement)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
sum.displacement=summary(mfull.displacement)
(anov.displacement=anova(mfull, mfull.displacement))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                                                 Model
## 1                 cylinders + horsepower + weight + acceleration + year + origin
## 2 cylinders + displacement + horsepower + weight + acceleration + year + origin
##      theta Resid. df    2 x log-lik.   Test    df LR stat.   Pr(Chi)
## 1 116028.0       371       -1781.520
## 2 115437.7       370       -1780.059 1 vs 2     1 1.461143 0.2267484
```

La variable *displacement* no influye en el consumo.

3.2.3 Eliminando la variable *horsepower* del modelo completo

```
mfull.horsepower <- update(mfull, . ~ . - horsepower)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
sum.horsepower=summary(mfull.horsepower)
(anov.horsepower=anova(mfull, mfull.horsepower))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                                              Model
## 1              cylinders + displacement + weight + acceleration + year + origin
## 2 cylinders + displacement + horsepower + weight + acceleration + year + origin
##      theta Resid. df    2 x log-lik.   Test    df LR stat.     Pr(Chi)
## 1 112969.5       371       -1787.721
## 2 115437.7       370       -1780.059 1 vs 2     1 7.661591 0.005640875
```

La variable *horsepower* si es significativa

3.2.4 Eliminando la variable *weight* del modelo completo

```
mfull.weight <- update(mfull, . ~ . - weight)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
sum.weight=summary(mfull.weight)
(anov.weight=anova(mfull, mfull.weight))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                                              Model
## 1          cylinders + displacement + horsepower + acceleration + year + origin
## 2 cylinders + displacement + horsepower + weight + acceleration + year + origin
##       theta Resid. df    2 x log-lik.   Test    df LR stat.     Pr(Chi)
## 1  79479.07       371       -1817.567
## 2 115437.67       370       -1780.059 1 vs 2     1 37.50738 9.10678e-10
```

La variable *weight* influye significativamente sobre el consumo

3.2.5 Eliminando la variable *acceleration* del modelo completo

```
mfull.acceleration <- update(mfull, . ~ . - acceleration)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```r
sum.acceleration=summary(mfull.acceleration)
(anov.acceleration=anova(mfull, mfull.acceleration))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                                           Model
## 1                cylinders + displacement + horsepower + weight + year + origin
## 2 cylinders + displacement + horsepower + weight + acceleration + year + origin
##      theta Resid. df    2 x log-lik.   Test    df  LR stat.   Pr(Chi)
## 1 114637.0       371       -1780.236
## 2 115437.7       370       -1780.059 1 vs 2     1 0.1770871 0.6738886
```

Confirmamos que la variable *acceleration* no es significativa sobre el consumo

3.2.6 Eliminando la variable *year* del modelo completo

```r
mfull.year <- update(mfull, . ~ . - year)
sum.year=summary(mfull.year)
(anov.year=anova(mfull, mfull.year))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                                           Model
## 1        cylinders + displacement + horsepower + weight + acceleration + origin
## 2 cylinders + displacement + horsepower + weight + acceleration + year + origin
##          theta Resid. df    2 x log-lik.   Test    df LR stat. Pr(Chi)
## 1     36.91911       371       -1966.785
## 2 115437.67352       370       -1780.059 1 vs 2     1 186.7256       0
```

La variable *year* si es significativo para el consumo

3.2.7 Eliminando la variable *origin* del modelo completo

```r
mfull.origin <- update(mfull, . ~ . - origin)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```r
sum.origin=summary(mfull.origin)
(anov.origin=anova(mfull, mfull.origin, test="Chisq"))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                                           Model
## 1          cylinders + displacement + horsepower + weight + acceleration + year
## 2 cylinders + displacement + horsepower + weight + acceleration + year + origin
##      theta Resid. df    2 x log-lik.   Test    df LR stat.   Pr(Chi)
## 1 111167.6       372       -1783.061
```

```
## 2 115437.7       370       -1780.059 1 vs 2     2 3.002137 0.2228919
```

La variable *origin* no es significativo

3.2.8 Tabla resumen paso 1

A continuación construiremos una tabla resumen con el resultado de eliminar del modelo completo cada una de las variables

```
comp_mfull=c(sum.mfull$aic, sum.mfull$deviance, 0)

comp_mfull.cilynders=c(sum.cilynders$aic,
                       sum.cilynders$deviance,
                       anov.acceleration$`Pr(Chi)`[2])

comp_mfull.displacement=c(sum.displacement$aic,
                          sum.displacement$deviance,
                          anov.displacement$`Pr(Chi)`[2])

comp_mfull.horsepower=c(sum.horsepower$aic,
                        sum.horsepower$deviance,
                        anov.horsepower$`Pr(Chi)`[2])

comp_mfull.weight=c(sum.weight$aic,
                    sum.weight$deviance,
                    anov.weight$`Pr(Chi)`[2])

comp_mfull.acceleration=c(sum.acceleration$aic,
                          sum.acceleration$deviance,
                          anov.acceleration$`Pr(Chi)`[2])

comp_mfull.year=c(sum.year$aic,
                  sum.year$deviance,
                  anov.year$`Pr(Chi)`[2])

comp_mfull.origin=c(sum.origin$aic,
                    sum.origin$deviance,
                    anov.origin$`Pr(Chi)`[2])


tabla_step1 = data.frame (round(rbind(comp_mfull, comp_mfull.cilynders,
                                      comp_mfull.displacement,comp_mfull.horsepower,
                                      comp_mfull.weight, comp_mfull.acceleration,
                                      comp_mfull.year, comp_mfull.origin), 3),
                                      row.names=c("mfull (modelo completo)",
                                                  "mfull-cylinder",
                                                  "mfull-displacement",
                                                  "mfull-horsepower",
                                                  "mfull-weight",
                                                  "mfull-acceleration",
                                                  "mfull-year",
                                                  "mfull-origin"))

print(knitr::kable(tabla_step1, format = "pandoc",
```

```r
                    col.names = c("AIC", "Deviance", "Pr(Chi)"), align='c'))
```

```
##
##
##                               AIC       Deviance    Pr(Chi)
## ------------------------  ----------  ----------  ---------
## mfull (modelo completo)    1800.059    336.742      0.000
## mfull-cylinder             1798.191    336.874      0.674
## mfull-displacement         1799.520    338.203      0.227
## mfull-horsepower           1805.721    344.403      0.006
## mfull-weight               1835.567    374.233      0.000
## mfull-acceleration         1798.236    336.919      0.674
## mfull-year                 1984.785    430.146      0.000
## mfull-origin               1799.061    339.743      0.223
```

A la vista de los resultados, podemos concluir que las variables menos significativas son por este orden: cylinder, acceleration, displacement, origin, ya que hemos comprobado que eliminarlas no influye, y que el AIC del modelo resultante es menor que el que contempla todas las variables.

3.3 Paso 2: Eliminando 2 variables al modelo completo

Nos quedamos con el mejor modelo del paso 1 y repetimos el proceso.

El mejor modelo del paso anterior es el modelo resultante de eliminar la variable *cylinders* al modelo completo, este modelo lo hemos nombrado como *mfull.cylinders*, cuyo summary es el siguiente:

```r
sum.cilynders
```

```
##
## Call:
## glm.nb(formula = m_13 ~ displacement + horsepower + weight +
##     acceleration + year + origin, data = data_auto, init.theta = 115122.7427,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0685  -0.6538  -0.0457   0.4949   3.5982
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -8.724e-01  4.493e-01  -1.942  0.05217 .
## displacement   -1.500e-03  6.545e-04  -2.292  0.02191 *
## horsepower     -4.957e-03  1.806e-03  -2.744  0.00607 **
## weight         -5.150e-04  8.314e-05  -6.194 5.85e-10 ***
## acceleration    3.743e-03  9.372e-03   0.399  0.68960
## year            6.802e-02  4.826e-03  14.094  < 2e-16 ***
## originEuropean  7.827e-02  4.669e-02   1.676  0.09367 .
## originJapanese  2.830e-02  4.352e-02   0.650  0.51547
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(115122.7) family taken to be 1)
##
##     Null deviance: 2218.33  on 378  degrees of freedom
## Residual deviance:  336.87  on 371  degrees of freedom
```

```
## AIC: 1798.2
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  115123
##           Std. Err.:  683371
## Warning while fitting theta: iteration limit reached
##
##  2 x log-likelihood:  -1780.191
```

Vemos que la variable *acceleration* no es significativa del modelo mfull.cylinders, probamos a eliminar dicha variable

```
mfull.cylinders.acc <- update(mfull.cylinders, . ~ . - acceleration)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
sum.cylinders.acc=summary(mfull.cylinders.acc)
(anov.cylinders.acc=anova(mfull.cylinders, mfull.cylinders.acc, test="Chisq"))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                                 Model
## 1                 displacement + horsepower + weight + year + origin
## 2 displacement + horsepower + weight + acceleration + year + origin
##      theta Resid. df    2 x log-lik.   Test    df  LR stat.   Pr(Chi)
## 1 114370.5       372       -1780.351
## 2 115122.7       371       -1780.191 1 vs 2     1 0.1593497 0.6897559
```

Vemos que la variable *acceleration* no es significativa en el modelo mfull.cylinders, en el paso 3 haremos pruebas para ver si es posible simplificar aun mas.

3.4 Paso 3: Eliminando 3 variables al modelo completo

Nos quedamos con el mejor modelo del paso 2 y repetimos el proceso.

El mejor modelo del paso anterior es el modelo resultante de eliminar la variable *acceleration* al modelo mfull.cylinders, este modelo lo hemos nombrado como *mfull.cylinders.acc*, cuyo summary es el siguiente:

```
sum.cylinders.acc
```

```
##
## Call:
## glm.nb(formula = m_13 ~ displacement + horsepower + weight +
##     year + origin, data = data_auto, init.theta = 114370.5017,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0683  -0.6480  -0.0564   0.4917   3.5995
##
```

```
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -7.802e-01  3.856e-01  -2.023   0.0431 *
## displacement   -1.535e-03  6.494e-04  -2.363   0.0181 *
## horsepower     -5.466e-03  1.280e-03  -4.270 1.96e-05 ***
## weight         -4.973e-04  7.049e-05  -7.055 1.72e-12 ***
## year            6.765e-02  4.741e-03  14.268  < 2e-16 ***
## originEuropean  7.852e-02  4.671e-02   1.681   0.0927 .
## originJapanese  2.984e-02  4.335e-02   0.688   0.4913
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(114370.5) family taken to be 1)
##
##     Null deviance: 2218.32  on 378  degrees of freedom
## Residual deviance:  337.03  on 372  degrees of freedom
## AIC: 1796.4
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  114371
##           Std. Err.:  678503
## Warning while fitting theta: iteration limit reached
##
##  2 x log-likelihood:  -1780.351
```

Observamos que todas las variables son significativas, vamos a probar a eliminar *displacement*

```
mfull.cylinders.acc.dis <- update(mfull.cylinders.acc, . ~ . - displacement)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
sum.cylinders.acc.dis=summary(mfull.cylinders.acc.dis)
(anov.cylinders.acc.dis=anova(mfull.cylinders.acc, mfull.cylinders.acc.dis, test="Chisq"))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                Model    theta Resid. df
## 1                horsepower + weight + year + origin 112852.4       373
## 2 displacement + horsepower + weight + year + origin 114370.5       372
##      2 x log-lik.   Test    df LR stat.    Pr(Chi)
## 1       -1785.962
## 2       -1780.351 1 vs 2     1 5.611691 0.01784104
```

La variable *displacement* si que es significativo en el modelo mfull.cylinders.acc

Probamos ahora a eliminar *origin*

```
mfull.cylinders.acc.orig <- update(mfull.cylinders.acc, . ~ . - origin)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```
sum.cylinders.acc.orig=summary(mfull.cylinders.acc.orig)
(anov.cylinders.acc.orig=anova(mfull.cylinders.acc, mfull.cylinders.acc.orig, test="Chisq"))
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                                Model    theta Resid. df
## 1          displacement + horsepower + weight + year 110370.1       374
## 2 displacement + horsepower + weight + year + origin 114370.5       372
##      2 x log-lik.   Test    df LR stat.   Pr(Chi)
## 1       -1783.220
## 2       -1780.351 1 vs 2     2 2.869548 0.2381692
```

Vemos que la variable *origin* no es significativa, por tanto la eliminamos del modelo

3.4 Paso 4: Eliminando 4 variables al modelo completo

Repetimos el proceso con el modelo resultante del paso 3, *mfull.cylinders.acc.orig*

```
sum.cylinders.acc.orig
```

```
##
## Call:
## glm.nb(formula = m_13 ~ displacement + horsepower + weight +
##     year, data = data_auto, init.theta = 110370.0581, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0420  -0.6593  -0.0704   0.4409   3.6686
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.959e-01  3.694e-01  -1.613  0.10674
## displacement -1.907e-03  6.079e-04  -3.137  0.00171 **
## horsepower   -5.488e-03  1.263e-03  -4.347 1.38e-05 ***
## weight       -4.786e-04  6.908e-05  -6.928 4.28e-12 ***
## year          6.571e-02  4.567e-03  14.388  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(110370.1) family taken to be 1)
##
##     Null deviance: 2218.3  on 378  degrees of freedom
## Residual deviance:  339.9  on 374  degrees of freedom
## AIC: 1795.2
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  110370
##           Std. Err.:  654885
```

```
## Warning while fitting theta: iteration limit reached
##
##  2 x log-likelihood:  -1783.22
```

En este modelo todas las variables son significativas, pero probaremos a eliminar *displacement* que es la que presenta un mayor p-valor

```r
mfull.cylinders.acc.orig.dis <- update(mfull.cylinders.acc.orig, . ~ . - displacement)
```

```
## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached

## Warning in theta.ml(Y, mu, sum(w), w, limit = control$maxit, trace =
## control$trace > : iteration limit reached
```

```r
sum.cylinders.acc.orig.dis=summary(mfull.cylinders.acc.orig.dis)
(anov.cylinders.acc.orig.dis=anova(mfull.cylinders.acc.orig, mfull.cylinders.acc.orig.dis, test="Chisq")
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: m_13
##                                        Model    theta Resid. df
## 1                 horsepower + weight + year 106422.2       375
## 2 displacement + horsepower + weight + year 110370.1       374
##      2 x log-lik.   Test    df LR stat.    Pr(Chi)
## 1       -1793.072
## 2       -1783.220 1 vs 2     1 9.851991 0.00169648
```

Tal y com suponíamos a la vista del summary del modelo mfull.cylinders.acc.orig vemos que no podemos simplificar más el modelo.

3.5 Tabla resumen

Vamos a actualizar la tabla comparativa con todos los modelos calculados

```r
comp_mfull.cylinders.acc = c(sum.cylinders.acc$aic,
                             sum.cylinders.acc$deviance,
                             anov.cylinders.acc$`Pr(Chi)`[2])

comp_mfull.cylinders.acc.dis= c(sum.cylinders.acc.dis$aic,
                                sum.cylinders.acc.dis$deviance,
                                anov.cylinders.acc.dis$`Pr(Chi)`[2])

comp_mfull.cylinders.acc.orig= c(sum.cylinders.acc.orig$aic,
                                 sum.cylinders.acc.orig$deviance,
                                 anov.cylinders.acc.orig$`Pr(Chi)`[2])

comp_mfull.cylinders.acc.orig.dis= c(sum.cylinders.acc.orig.dis$aic,
                                     sum.cylinders.acc.orig.dis$deviance,
                                     anov.cylinders.acc.orig.dis$`Pr(Chi)`[2])



tabla_step2 = data.frame (rbind(tabla_step1,
                                comp_mfull.cylinders.acc,
                                comp_mfull.cylinders.acc.dis,
                                comp_mfull.cylinders.acc.orig,
```

```r
                                  comp_mfull.cylinders.acc.orig.dis),
                                  row.names=c("mfull (modelo completo)",
                                              "mfull-cylinder",
                                              "mfull-displacement",
                                              "mfull-horsepower",
                                              "mfull-weight",
                                              "mfull-acceleration",
                                              "mfull-year",
                                              "mfull-origin",
                                              "mfull-cylinder-acceleration",
                                              "mfull-cylinder-acceleration-displacement",
                                              "mfull-cylinder-acceleration-origin (BEST)",
                                              "mfull-cylinder-acceleration-origin-displacement"))




print(knitr::kable(tabla_step2, format = "pandoc",
                   col.names = c("AIC", "Deviance", "Pr(Chi)"), align='c'))
```

```
##
##
##                                                        AIC        Deviance      Pr(Chi)
## ------------------------------------------------  ----------  ----------  -----------
## mfull (modelo completo)                            1800.059    336.7420    0.0000000
## mfull-cylinder                                     1798.191    336.8740    0.6740000
## mfull-displacement                                 1799.520    338.2030    0.2270000
## mfull-horsepower                                   1805.721    344.4030    0.0060000
## mfull-weight                                       1835.567    374.2330    0.0000000
## mfull-acceleration                                 1798.236    336.9190    0.6740000
## mfull-year                                         1984.785    430.1460    0.0000000
## mfull-origin                                       1799.061    339.7430    0.2230000
## mfull-cylinder-acceleration                        1796.351    337.0329    0.6897559
## mfull-cylinder-acceleration-displacement           1799.962    342.6441    0.0178410
## mfull-cylinder-acceleration-origin (BEST)          1795.220    339.9012    0.2381692
## mfull-cylinder-acceleration-origin-displacement    1803.072    349.7518    0.0016965
```

4. Modelo resultante

El mejor modelo obtenido es el resultante de eliminar las variables *cylinder*, *acceleration* y *origin* al modelo binomial negativo con todas las variables. Este modelo presenta un AIC = 1795.220, y en la tabla anterior corresponde a la fila *mfull-cylinder-acceleration-origin (BEST)*

Las estimaciones de los coeficientes y sus intervalos de confianza son las siguientes.

```r
(est <- cbind(Estimate = coef(mfull.cylinders.acc.orig),
              confint(mfull.cylinders.acc.orig)))
```

```
## Waiting for profiling to be done...

##                   Estimate         2.5 %        97.5 %
## (Intercept)  -0.5958602722 -1.3221685189  0.1259281664
## displacement -0.0019067969 -0.0030986750 -0.0007158221
## horsepower   -0.0054883335 -0.0079656021 -0.0030162977
```

```
## weight       -0.0004785529 -0.0006141509 -0.0003434097
## year          0.0657082423  0.0567764642  0.0746787024
```

Siendo los valores de las exponenciales

```r
(exp(est))
```

```
##                Estimate     2.5 %    97.5 %
## (Intercept)  0.5510883 0.2665566 1.1342007
## displacement 0.9980950 0.9969061 0.9992844
## horsepower   0.9945267 0.9920660 0.9969882
## weight       0.9995216 0.9993860 0.9996566
## year         1.0679151 1.0584192 1.0775379
```

El modelo propuesto es el siguiente:

ln(m_13) = -0.5959 - 0.0019 · displacement - 0.0055 · horsepower -0.0005 · weight + 0.0657 · year

El consumo de un vehículo depende de las siguientes variables:

- **displacement**: relación directamente proporcional con el consumo del vehículo
- **horsepower**: relación directamente proporcional con el consumo del vehículo (cuanta mas potencia tiene el motor más consume)
- **weight**: relación directamente proporcional con el consumo del vehículo (cuanto más pesado es el vehículo más consume)
- **year**: relación inversalmente proporcional con el consumo del vehículo (cuanto más nuevo es el vehículo menos consume)

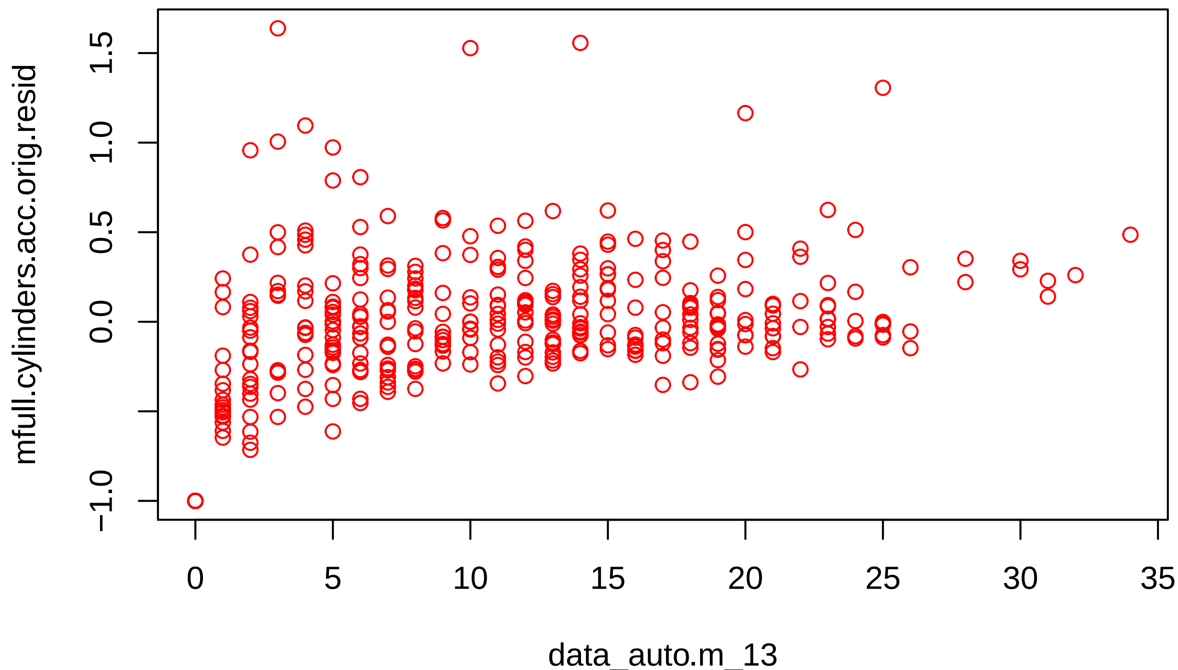
La bondad de ajuste global viene dada por

```r
with(mfull.cylinders.acc.orig,
     cbind(res.deviance = deviance,
           df = df.residual,
           p = pchisq(deviance, df.residual, lower.tail=FALSE)))
```

```
##      res.deviance  df         p
## [1,]     339.9012 374 0.8965707
```

Gráfica de los residuos frente a la variable de estudio (m_13)

```r
plot(data.frame(data_auto$m_13, mfull.cylinders.acc.orig$resid), col="red")
```



El gráfico de valores observados contra residuos pone de manifiesto como aumenta la varianza de los residuos

5. Comparación con el modelo Poisson

A continuación compararemos el modelo propuesto construido usando el modelo binomial negativo con el construido utilizando Poisson.

```r
mPoisson <- glm(data_auto$m_13 ~ displacement + horsepower + weight + year,
                data=data_auto, family=poisson)

summary(mPoisson)
```

```
##
## Call:
## glm(formula = data_auto$m_13 ~ displacement + horsepower + weight +
##     year, family = poisson, data = data_auto)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0420  -0.6593  -0.0704   0.4409   3.6688
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.958e-01  3.694e-01  -1.613  0.10673
## displacement -1.907e-03  6.079e-04  -3.137  0.00171 **
```

```
## horsepower   -5.488e-03  1.263e-03  -4.347 1.38e-05 ***
## weight       -4.785e-04  6.907e-05  -6.928 4.27e-12 ***
## year          6.571e-02  4.566e-03  14.389  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 2218.51  on 378  degrees of freedom
## Residual deviance:  339.93  on 374  degrees of freedom
## AIC: 1793.2
##
## Number of Fisher Scoring iterations: 5
```

```r
X2 <- 2 * (logLik(mfull) - logLik(mPoisson))
X2
```

```
## 'log Lik.' 3.152601 (df=10)
```

```r
pchisq(X2, df = 1, lower.tail=FALSE)
```

```
## 'log Lik.' 0.07580602 (df=10)
```

El valor obtenido de Chi es algo mayor que 0.05, por tanto vemos que el modelo binomial negativo es más apropiado que el modelo de Poisson, aunque realmente ambos modelos son similares, puesto que los p-valores obtenidos con el modelo de Poisson sugieren un modelo muy similar, las variables significativas son similares a las obtenidas con el modelo binomial negativo, por tanto podemos concluir que aunque no se cumpla la hipótesis de varianza igual a media para aplicar el modelo de Poisson el modelo resultante con aplicando Poisson también da buenos resultados sobre este conjunto de datos.