Laboratorio. Módulo 2

Franco Mendoza Muraira A01383399

2023-11-28

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: carData

## Loading required package: tibble

## Loading required package: bitops

## Rattle: A free graphical interface for data science with R.
## Versión 5.5.1 Copyright (c) 2006-2021 Togaware Pty Ltd.
## Escriba 'rattle()' para agitar, sacudir y  rotar sus datos.
```

Problema 1

```
data = Cars93
head(data)
```

```
##   Manufacturer   Model    Type Min.Price Price Max.Price MPG.city MPG.highway
## 1        Acura Integra   Small      12.9  15.9      18.8       25          31
## 2        Acura  Legend Midsize      29.2  33.9      38.7       18          25
## 3         Audi      90 Compact      25.9  29.1      32.3       20          26
## 4         Audi     100 Midsize      30.8  37.7      44.6       19          26
## 5          BMW    535i Midsize      23.7  30.0      36.2       22          30
## 6        Buick Century Midsize      14.2  15.7      17.3       22          31
##              AirBags DriveTrain Cylinders EngineSize Horsepower  RPM
## 1               None      Front         4        1.8        140 6300
## 2 Driver & Passenger      Front         6        3.2        200 5500
## 3        Driver only      Front         6        2.8        172 5500
## 4 Driver & Passenger      Front         6        2.8        172 5500
## 5        Driver only       Rear         4        3.5        208 5700
## 6        Driver only      Front         4        2.2        110 5200
```

```
##   Rev.per.mile Man.trans.avail Fuel.tank.capacity Passengers Length Wheelbase
## 1         2890             Yes               13.2          5    177       102
## 2         2335             Yes               18.0          5    195       115
## 3         2280             Yes               16.9          5    180       102
## 4         2535             Yes               21.1          6    193       106
## 5         2545             Yes               21.1          4    186       109
## 6         2565              No               16.4          6    189       105
##   Width Turn.circle Rear.seat.room Luggage.room Weight  Origin          Make
## 1    68          37           26.5           11   2705 non-USA Acura Integra
## 2    71          38           30.0           15   3560 non-USA  Acura Legend
## 3    67          37           28.0           14   3375 non-USA       Audi 90
## 4    70          37           31.0           17   3405 non-USA      Audi 100
## 5    69          39           27.0           13   3640 non-USA      BMW 535i
## 6    69          41           28.0           16   2880     USA Buick Century
```

A. Analice si existe una correlación entre el peso de un vehículo (Weight) y la potencia de su motor (Horsepower).

```
cor(data$Weight, data$Horsepower)
```

```
## [1] 0.7387975
```

Podemos ver que las 2 variables tienen una correlaicon de 0.74, lo cual indica que existe una correlación positiva entre el peso de un vehículo y la potencia de su motor, y es una correlación significativa.

B. Proponga un modelo de regresión simple para estas variables.

```
y= data$Horsepower
x= data$Weight
regresion = lm(y ~ x)
summary(regresion)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -93.017 -20.921  -1.515   8.356 136.028
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -57.738203  19.622752  -2.942  0.00413 **
## x             0.065595   0.006272  10.458  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.49 on 91 degrees of freedom
## Multiple R-squared:  0.5458, Adjusted R-squared:  0.5408
## F-statistic: 109.4 on 1 and 91 DF,  p-value: < 2.2e-16
```

Este modelo nos da una R^2 ajustada de 0.54, lo cual no es muy bueno, pero nos indica que el modelo es significativo. Tambien podemos ver que el valor p de los coeficientes es menor a un alfa de 0.05, lo que nos dice que se puede rechazar la hipotesis nula de que son igual a 0, y se concluye que los coeficientes son significativos. El modelo es el siguiente:

$Horsepower = -57.738203 + 0.065595 * Weight$

C. Realice la validación de los supuestos del modelo.

Normalidad de los residuos

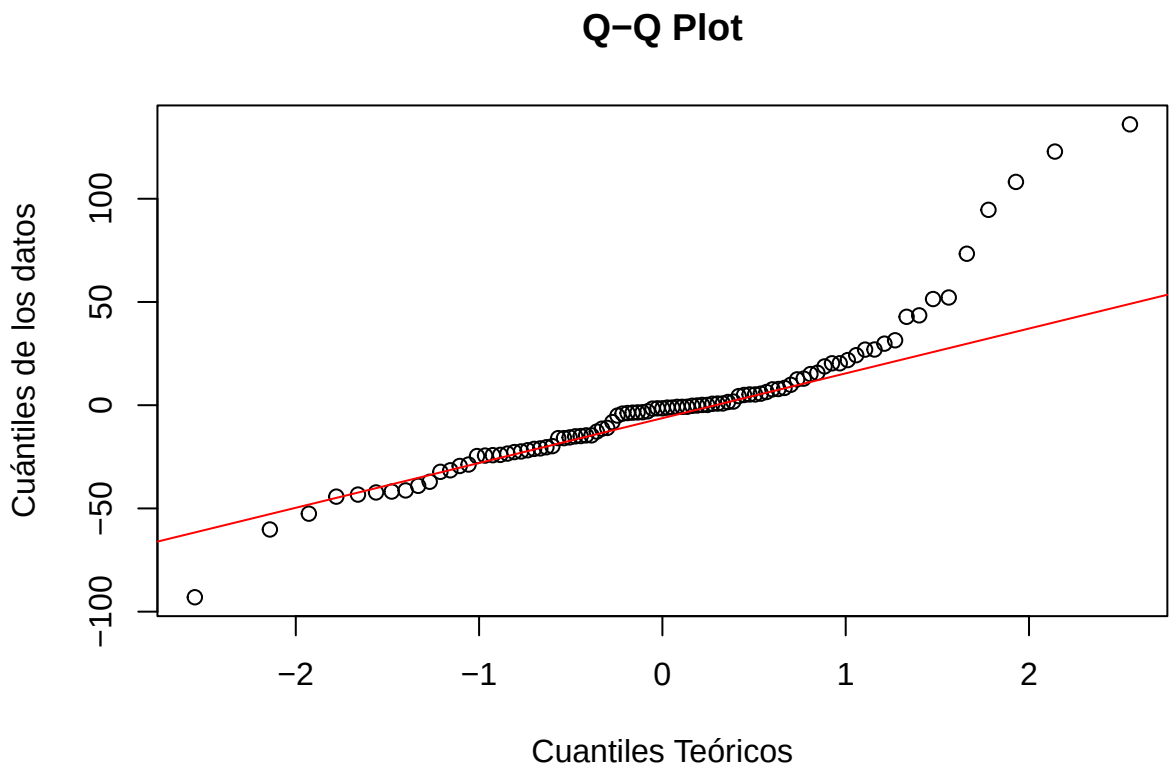
H_0 : Los residuos provienen de una distribución normal.

H_1 : Los residuos no provienen de una distribución normal.

```
ad.test(residuals(regresion))
```

```
##
##  Anderson-Darling normality test
##
## data:  residuals(regresion)
## A = 3.1687, p-value = 5.279e-08
```

```
qqnorm(regresion$residuals,main = "Q-Q Plot",xlab = "Cuantiles Teóricos",ylab= "Cuántiles de los datos")
qqline(regresion$residuals,col="red")
```



Debido al valor p bajo de la prueba de Anderson-Darling, se rechaza la hipotesis nula y se concluye que los residuos no provienen de una distribución normal. Esto se puede ver en el Q-Q plot, ya que los residuos no siguen la linea roja, ya que al final se dispersan mucho.

Homocedasticidad

H_0 : La varianza de los errores es constante.

H_1 : La varianza de los errores no es constante.

```
bptest(regresion)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  regresion
## BP = 7.7789, df = 1, p-value = 0.005286
```

Debido al valor p bajo de la prueba de Breusch-Pagan, se rechaza la hipotesis nula y se concluye que la varianza de los errores no es constante, lo cual es malo para el modelo ya que hay heterocedasticidad.

Independencia

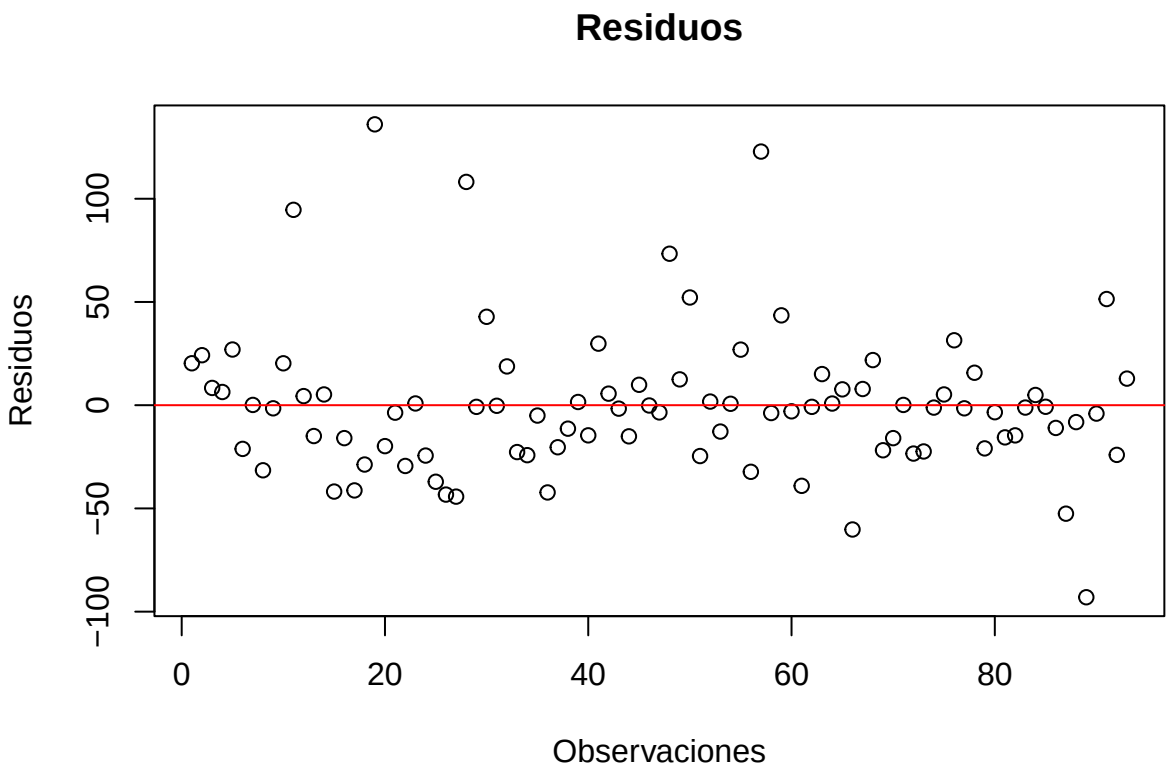
H_0 : Los residuos son independientes.

H_1 : Los residuos no son independientes.

```
dwtest(regresion)
```

```
##
##  Durbin-Watson test
##
## data:  regresion
## DW = 2.1052, p-value = 0.6894
## alternative hypothesis: true autocorrelation is greater than 0
```

```
plot(regresion$residuals,main = "Residuos",xlab = "Observaciones",ylab = "Residuos")
abline(h=0,col="red")
```

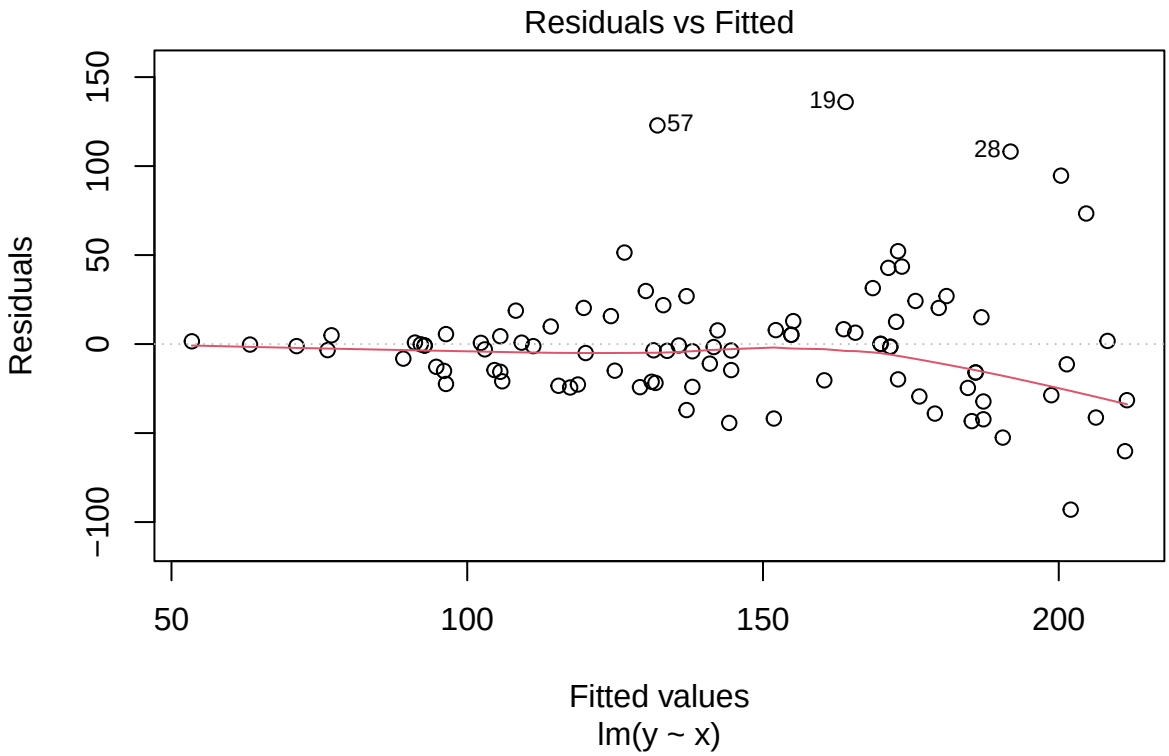


Debido al valor p alto de la prueba de Durbin-Watson, no se puede rechazar la hipotesis nula y se concluye que los residuos son independientes, esto se puede ver en la grafica anterior, con la aleatoriedad de los residuos.

D. Identifique los datos atípicos y datos influyentes y describa los criterios implementados para su determinación.

Datos atipicos

```
plot(regresion,which=1)
```



Con esta grafica podemos identificar facilmente los datos atipicos, ya que son los que estan muy alejados de la linea roja. En este caso, los datos atipicos son los siguientes: 57, 19, y 28.

Datos Influyentes

Segun los valores hat:

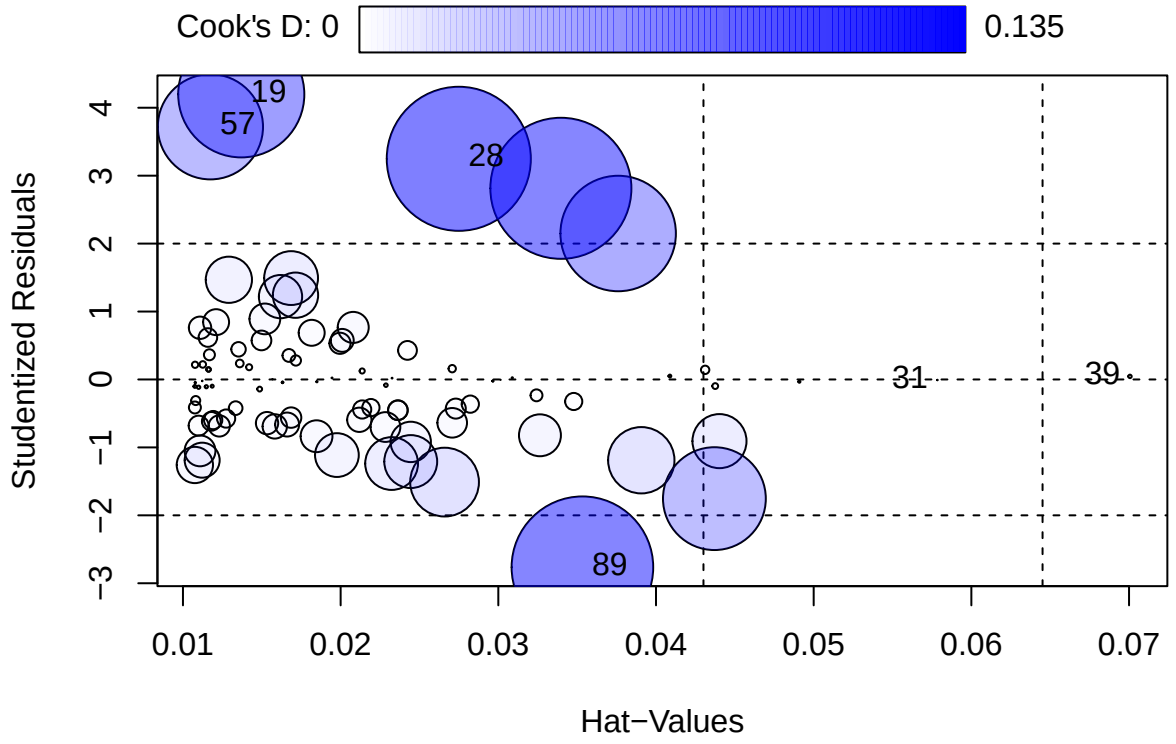
```
## Criterio de Influencia:  0.05376344
```

```
## Datos Influyentes:
```

```
##         31
## 0.05784921
##         39
## 0.0700586
```

Segun el metodo de Cook:

```
influencePlot(regresion,id=TRUE)
```




```
##         StudRes        Hat        CookD
## 19  4.196938831 0.01369854 1.034359e-01
## 28  3.248248107 0.02749431 1.349813e-01
## 31 -0.008199298 0.05784921 2.086887e-06
## 39  0.045191368 0.07005860 7.778142e-05
## 57  3.719207655 0.01174131 7.201528e-02
## 89 -2.764150849 0.03533386 1.304124e-01
```

E. Calcule:

1. Intervalos de confianza para los coeficientes de regresión

```
confint(regresion)
```

```
##                    2.5 %       97.5 %
## (Intercept) -96.71638922 -18.76001719
## x             0.05313529   0.07805411
```

2. Intervalos de confianza para la respuesta media de la regresión

```r
resmed=predict(regresion, interval = "confidence")
head(resmed)
```

```
##        fit      lwr      upr
## 1 119.6955 111.0670 128.3239
## 2 175.7789 166.2779 185.2800
## 3 163.6439 155.4216 171.8662
## 4 165.6117 157.2118 174.0117
## 5 181.0265 170.8598 191.1932
## 6 131.1745 123.4794 138.8696
```

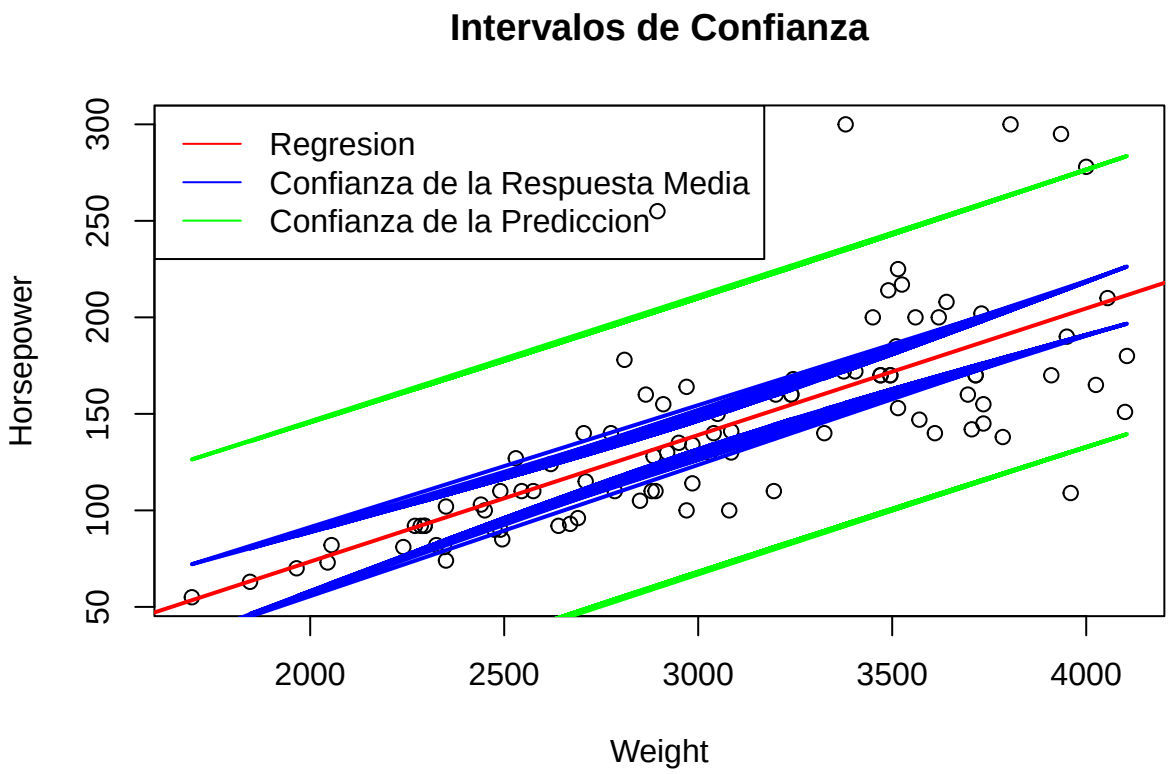
3. Intervalos de predicción

```r
pred=predict(regresion, interval = "prediction")
head(pred)
```

```
##        fit       lwr      upr
## 1 119.6955  48.67293 190.7180
## 2 175.7789 104.64511 246.9127
## 3 163.6439  92.66958 234.6182
## 4 165.6117  94.61662 236.6069
## 5 181.0265 109.80072 252.2523
## 6 131.1745  60.25934 202.0897
```

F. Realice un gráfico donde se ilustren los intervalos de confianza de la respuesta media y predicción.

```r
plot(data$Weight,data$Horsepower,main = "Intervalos de Confianza",xlab = "Weight",ylab = "Horsepower")
abline(regresion,col="red",lwd=2)
lines(data$Weight,resmed[,2],col="blue",lwd=2)
lines(data$Weight,resmed[,3],col="blue",lwd=2)
lines(data$Weight,pred[,2],col="green",lwd=2)
lines(data$Weight,pred[,3],col="green",lwd=2)
legend("topleft",legend = c("Regresion","Confianza de la Respuesta Media","Confianza de la Prediccion")
```



Podemos ver que la mayor parte de los datos quedan dentro de los intervalos de confianza de la respuesta media y de la prediccion, lo cual es bueno, hay pocos que salen de estos, que serían los datos atípicos.

G. Proponga un segundo modelo implementando una transformación a la variable Horsepower de modo que se satisfaga el supuesto de normalidad.

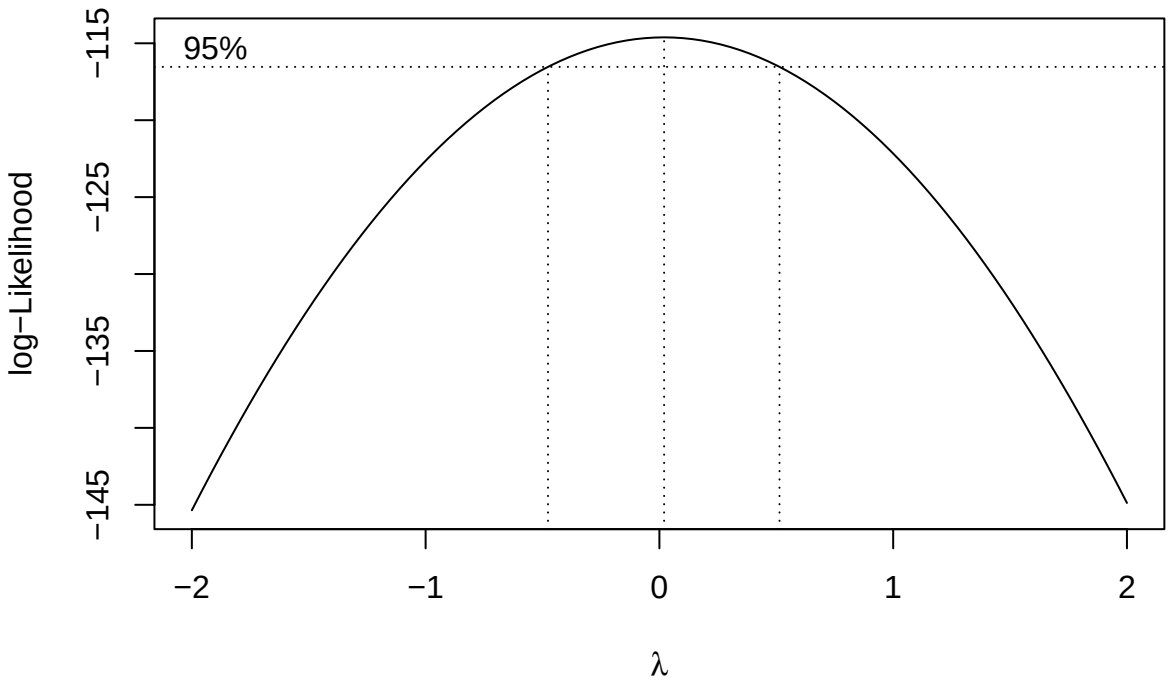
Normalidad

```
ad.test(y)
```

```
##
##  Anderson-Darling normality test
##
## data:  y
## A = 1.2873, p-value = 0.002276
```

Como el valor p de la prueba de Anderson-Darling es menor a 0.05, se rechaza la hipotesis nula y se concluye que los datos no provienen de una distribución normal.

```
bc = boxcox(lm(y~1))
```




```
l =bc$x[which.max(bc$y)]
cat("Lambda optimo: ",l)
```

```
## Lambda optimo:  0.02020202
```

Nuestro lambda optimo es de 0.202, por lo que la transformacion es la siguiente:

```
y2 = ((y^l)-1)/l
head(y2)
```

```
## [1] 5.196725 5.592268 5.424662 5.424662 5.635937 4.930892
```

Normalidad nuevos datos y

```
ad.test(y2)
```

```
##
##  Anderson-Darling normality test
##
## data:  y2
## A = 0.42449, p-value = 0.3114
```

Nuevo Modelo

```
new_model = lm(y2~x)
summary(new_model)
```

```
##
## Call:
## lm(formula = y2 ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7177 -0.1507  0.0049  0.1105  0.8012
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.495e+00  1.291e-01   27.07   <2e-16 ***
## x           5.412e-04  4.127e-05   13.12   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2335 on 91 degrees of freedom
## Multiple R-squared:  0.654,  Adjusted R-squared:  0.6502
## F-statistic:   172 on 1 and 91 DF,  p-value: < 2.2e-16
```

Igual concluimos que los coeficientes son significativos debido a sus valores p siendo muy cercanos a 0, y el modelo es un poco mejor debido al R^2 de 0.6502, que es mayor al anterior.

H. Contraste los resultados de la validación del segundo modelo con el obtenido inicialmente.

Normalidad de los residuos

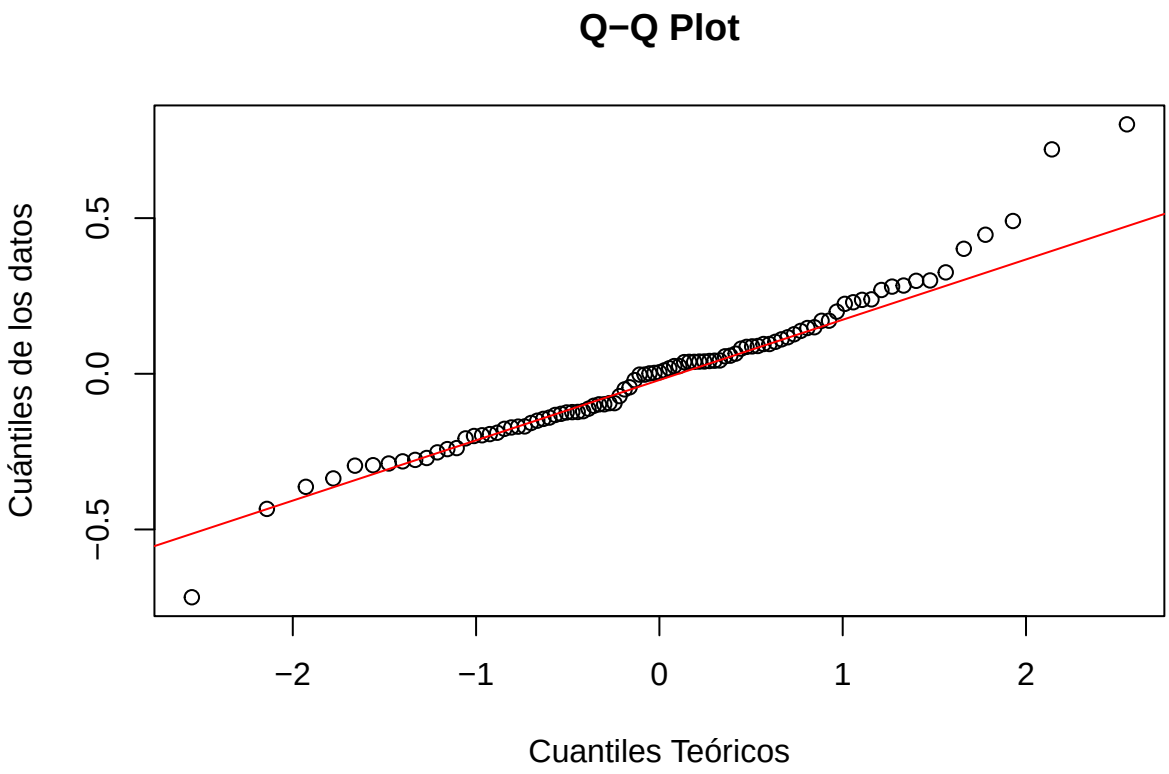
H_0 : Los residuos provienen de una distribución normal.

H_1 : Los residuos no provienen de una distribución normal.

```
ad.test(residuals(new_model))
```

```
##
##  Anderson-Darling normality test
##
## data:  residuals(new_model)
## A = 0.74795, p-value = 0.04972
```

```
qqnorm(new_model$residuals,main = "Q-Q Plot",xlab = "Cuantiles Teóricos",ylab= "Cuántiles de los datos")
qqline(new_model$residuals,col="red")
```



Debido a que el valor p sigue estando por debajo del alfa de 0.05 de la prueba de Anderson-Darling, se rechaza la hipotesis nula y se concluye que los residuos no provienen de una distribución normal. Esto se puede ver en el Q-Q plot, ya que los residuos no siguen la linea roja, aunque se ve un poco mejor este modelo que el anterior.

Homocedasticidad

H_0 : La varianza de los errores es constante.

H_1 : La varianza de los errores no es constante.

```
bptest(new_model)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  new_model
## BP = 4.7616, df = 1, p-value = 0.0291
```

Debido al valor p igual estando por debajo del alfa en la prueba de Breusch-Pagan, se rechaza la hipotesis nula y se concluye que la varianza de los errores no es constante.

Independencia

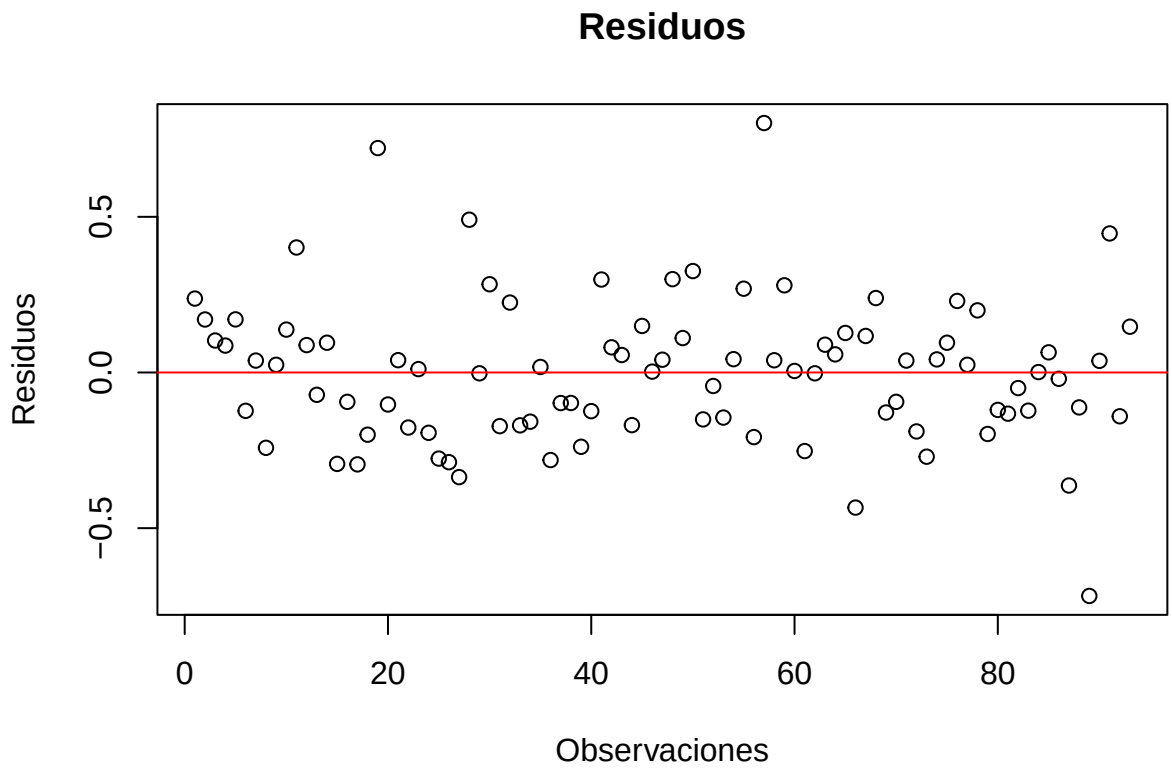
H_0 : Los residuos son independientes.

H_1 : Los residuos no son independientes.

```
dwtest(new_model)
```

```
##
##  Durbin-Watson test
##
## data:  new_model
## DW = 2.0333, p-value = 0.5578
## alternative hypothesis: true autocorrelation is greater than 0
```

```
plot(new_model$residuals,main = "Residuos",xlab = "Observaciones",ylab = "Residuos")
abline(h=0,col="red")
```



Debido al valor p por encima del alfa con 0.5578 en la prueba de Durbin-Watson, no se puede rechazar la hipotesis nula y se concluye que los residuos son independientes, esto se puede ver en la grafica anterior, con la aleatoriedad de los residuos.

Conclusión de los modelos

Se puede ver que los 2 modelos cumplieron los supuestos de la misma manera, los 2 modelos solo cumpliendo 1 supuesto. Esto nos dice que los modelos son parecidos en lo confiable que son, la diferencia es que el primer modelo explicó el 54% de variabilidad, mientras que el segundo modelo explicó el 65% de variabilidad, por lo que el segundo modelo podría concluirse que es mejor que el primero en este caso.

Problema 2

A. Realice el análisis de correlación entre las variables numéricas y seleccione un conjunto de variables numéricas puedan explicar la variabilidad del precio del vehículo.

```
number = data[sapply(data,is.numeric)]
data_num = number[,-c(16,17)]
head(data_num)
```

```
##   Min.Price Price Max.Price MPG.city MPG.highway EngineSize Horsepower  RPM
## 1      12.9  15.9      18.8       25          31        1.8        140 6300
## 2      29.2  33.9      38.7       18          25        3.2        200 5500
## 3      25.9  29.1      32.3       20          26        2.8        172 5500
## 4      30.8  37.7      44.6       19          26        2.8        172 5500
## 5      23.7  30.0      36.2       22          30        3.5        208 5700
## 6      14.2  15.7      17.3       22          31        2.2        110 5200
##   Rev.per.mile Fuel.tank.capacity Passengers Length Wheelbase Width Turn.circle
## 1         2890               13.2          5    177       102    68          37
## 2         2335               18.0          5    195       115    71          38
## 3         2280               16.9          5    180       102    67          37
## 4         2535               21.1          6    193       106    70          37
## 5         2545               21.1          4    186       109    69          39
## 6         2565               16.4          6    189       105    69          41
##   Weight
## 1   2705
## 2   3560
## 3   3375
## 4   3405
## 5   3640
## 6   2880
```

```
corr = as.data.frame(cor(data_num))
corr
```

```
##                     Min.Price        Price   Max.Price   MPG.city MPG.highway
## Min.Price           1.00000000  0.970601402  0.90675608 -0.6228754  -0.5799658
## Price               0.97060140  1.000000000  0.98158027 -0.5945622  -0.5606804
## Max.Price           0.90675608  0.981580272  1.00000000 -0.5478109  -0.5225607
## MPG.city           -0.62287544 -0.594562163 -0.54781090  1.0000000   0.9439358
## MPG.highway        -0.57996581 -0.560680362 -0.52256074  0.9439358   1.0000000
## EngineSize          0.64548767  0.597425392  0.53501197 -0.7100032  -0.6267946
## Horsepower          0.80244412  0.788217578  0.74444475 -0.6726362  -0.6190437
## RPM                -0.04259816 -0.004954931  0.02501478  0.3630451   0.3134687
## Rev.per.mile       -0.47039499 -0.426395113 -0.37402421  0.6958570   0.5874968
## Fuel.tank.capacity  0.63536902  0.619479981  0.58129439 -0.8131444  -0.7860386
## Passengers          0.06123644  0.057860074  0.05321592 -0.4168559  -0.4663858
## Length              0.55385881  0.503628440  0.44293341 -0.6662390  -0.5428974
## Wheelbase           0.51675786  0.500864163  0.46750079 -0.6671076  -0.6153842
## Width               0.49287830  0.456027866  0.40841435 -0.7205344  -0.6403592
## Turn.circle         0.42860290  0.392589927  0.34778485 -0.6663889  -0.5936833
```

```
## Weight              0.66655377  0.647179005  0.60514157 -0.8431385  -0.8106581
##                    EngineSize   Horsepower          RPM Rev.per.mile
## Min.Price           0.6454877  0.802444116 -0.042598158   -0.4703950
## Price               0.5974254  0.788217578 -0.004954931   -0.4263951
## Max.Price           0.5350120  0.744444746  0.025014782   -0.3740242
## MPG.city           -0.7100032 -0.672636151  0.363045129    0.6958570
## MPG.highway        -0.6267946 -0.619043685  0.313468728    0.5874968
## EngineSize          1.0000000  0.732119730 -0.547897805   -0.8240086
## Horsepower          0.7321197  1.000000000  0.036688212   -0.6003139
## RPM                -0.5478978  0.036688212  1.000000000    0.4947642
## Rev.per.mile       -0.8240086 -0.600313870  0.494764211    1.0000000
## Fuel.tank.capacity  0.7593062  0.711790317 -0.333345218   -0.6097098
## Passengers          0.3727212  0.009263668 -0.467137627   -0.3349756
## Length              0.7802831  0.550864666 -0.441249316   -0.6902333
## Wheelbase           0.7324842  0.486854213 -0.467812289   -0.6368238
## Width               0.8671102  0.644413421 -0.539721132   -0.7804604
## Turn.circle         0.7784636  0.561215737 -0.505650650   -0.7331596
## Weight              0.8450753  0.738797516 -0.427931473   -0.7352642
##                    Fuel.tank.capacity   Passengers     Length  Wheelbase
## Min.Price                   0.6353690  0.061236438  0.5538588  0.5167579
## Price                       0.6194800  0.057860074  0.5036284  0.5008642
## Max.Price                   0.5812944  0.053215917  0.4429334  0.4675008
## MPG.city                   -0.8131444 -0.416855859 -0.6662390 -0.6671076
## MPG.highway                -0.7860386 -0.466385827 -0.5428974 -0.6153842
## EngineSize                  0.7593062  0.372721168  0.7802831  0.7324842
## Horsepower                  0.7117903  0.009263668  0.5508647  0.4868542
## RPM                        -0.3333452 -0.467137627 -0.4412493 -0.4678123
## Rev.per.mile               -0.6097098 -0.334975577 -0.6902333 -0.6368238
## Fuel.tank.capacity          1.0000000  0.472095108  0.6904612  0.7576745
## Passengers                  0.4720951  1.000000000  0.4852941  0.6940544
## Length                      0.6904612  0.485294130  1.0000000  0.8236504
## Wheelbase                   0.7576745  0.694054395  0.8236504  1.0000000
## Width                       0.7987190  0.489978637  0.8221479  0.8072134
## Turn.circle                 0.6713431  0.449024715  0.7389545  0.7233244
## Weight                      0.8940181  0.553272980  0.8062743  0.8718953
##                         Width Turn.circle     Weight
## Min.Price           0.4928783   0.4286029  0.6665538
## Price               0.4560279   0.3925899  0.6471790
## Max.Price           0.4084144   0.3477849  0.6051416
## MPG.city           -0.7205344  -0.6663889 -0.8431385
## MPG.highway        -0.6403592  -0.5936833 -0.8106581
## EngineSize          0.8671102   0.7784636  0.8450753
## Horsepower          0.6444134   0.5612157  0.7387975
## RPM                -0.5397211  -0.5056506 -0.4279315
## Rev.per.mile       -0.7804604  -0.7331596 -0.7352642
## Fuel.tank.capacity  0.7987190   0.6713431  0.8940181
## Passengers          0.4899786   0.4490247  0.5532730
## Length              0.8221479   0.7389545  0.8062743
## Wheelbase           0.8072134   0.7233244  0.8718953
## Width               1.0000000   0.8178542  0.8749605
## Turn.circle         0.8178542   1.0000000  0.7780431
## Weight              0.8749605   0.7780431  1.0000000
```

B. A partir de las variables seleccionadas, ajuste un modelo de regresión lineal múltiple.

```
corr["Price"]
```

```
##                          Price
## Min.Price           0.970601402
## Price               1.000000000
## Max.Price           0.981580272
## MPG.city           -0.594562163
## MPG.highway        -0.560680362
## EngineSize          0.597425392
## Horsepower          0.788217578
## RPM                -0.004954931
## Rev.per.mile       -0.426395113
## Fuel.tank.capacity  0.619479981
## Passengers          0.057860074
## Length              0.503628440
## Wheelbase           0.500864163
## Width               0.456027866
## Turn.circle         0.392589927
## Weight              0.647179005
```

```
y= data_num$Price
x1= data_num$Horsepower
x2 = data_num$Wheelbase
x3 = data_num$MPG.highway
x4 = data_num$Rev.per.mile
```

Se eligieron las variables Horsepower, Wheelbase, MPG.highway y Rev.per.mile, ya que son las que tienen mayor correlacion con el precio, sin tener una alta correlación entre ellas.

```
regresion = lm(y~x1+x2+x3+x4)
summary(regresion)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3 + x4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.372  -3.386  -0.350   2.033  30.102
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -38.295005  17.351124  -2.207   0.0299 *
## x1            0.140482   0.015657   8.973 4.67e-14 ***
## x2            0.308611   0.123262   2.504   0.0141 *
## x3           -0.151387   0.162434  -0.932   0.3539
## x4            0.004255   0.001751   2.431   0.0171 *
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.733 on 88 degrees of freedom
## Multiple R-squared:  0.6631, Adjusted R-squared:  0.6478
## F-statistic:  43.3 on 4 and 88 DF,  p-value: < 2.2e-16
```

Podemos ver que el modelo tiene una variable con un valor p por encima de el alfa de 0.05, por lo que se puede eliminar esta variable y volver a hacer el modelo, en este caso es x3, MPG.highway.

```
regresion2 = lm(y~x1+x2+x4)
summary(regresion2)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.241  -3.328  -0.407   1.715  30.040
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -47.129885  14.521934  -3.245  0.00165 **
## x1            0.146037   0.014467  10.094  < 2e-16 ***
## x2            0.349141   0.115250   3.029  0.00321 **
## x4            0.004006   0.001729   2.317  0.02279 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.728 on 89 degrees of freedom
## Multiple R-squared:  0.6598, Adjusted R-squared:  0.6483
## F-statistic: 57.53 on 3 and 89 DF,  p-value: < 2.2e-16
```

Ahora tenemos que el R^2 ajustado es de 0.6483, y todas las variables son significativas.

C. Realice la validación de los supuestos del modelo.

Normalidad de los residuos

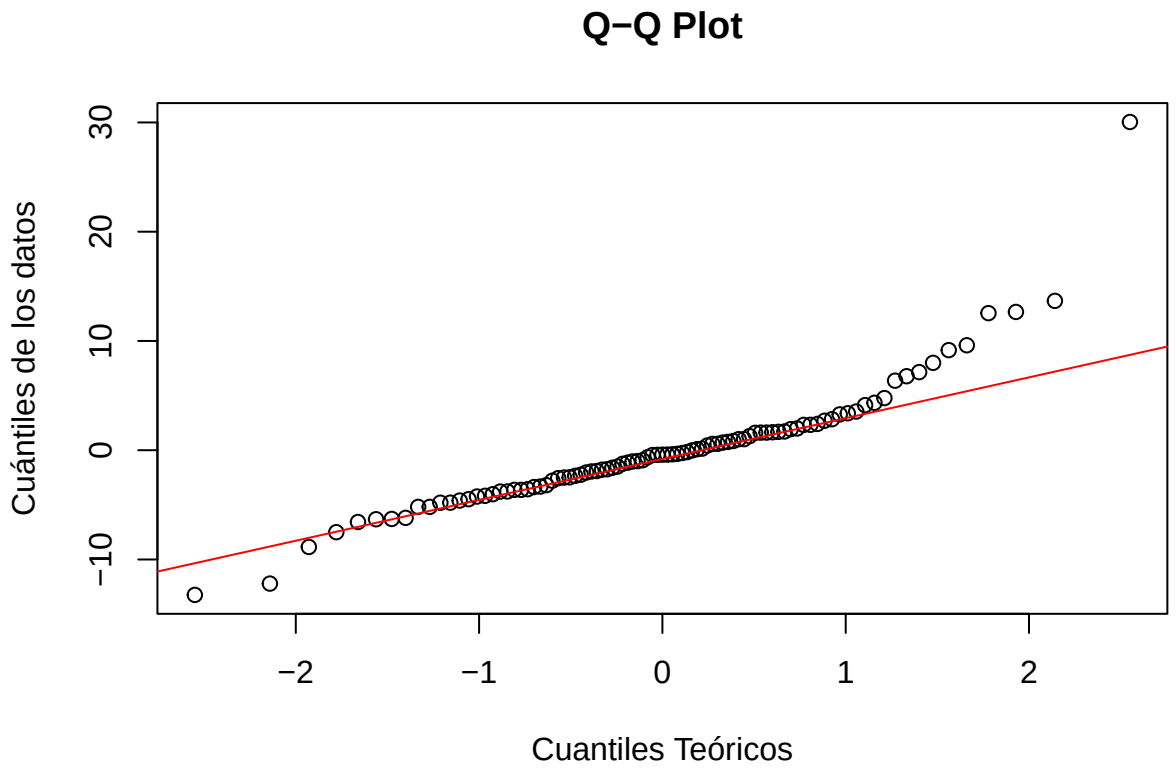
H_0 : Los residuos provienen de una distribución normal.

H_1 : Los residuos no provienen de una distribución normal.

```
ad.test(residuals(regresion2))
```

```
##
##  Anderson-Darling normality test
##
## data:  residuals(regresion2)
## A = 2.563, p-value = 1.615e-06
```

```r
qqnorm(regresion2$residuals,main = "Q-Q Plot",xlab = "Cuantiles Teóricos",ylab= "Cuántiles de los datos
qqline(regresion2$residuals,col="red")
```



Debido a que el valor p de la prueba de Anderson-Darling es menor a 0.05, y muy cercan a 0, se rechaza la hipotesis nula y se concluye que los residuos no provienen de una distribución normal. Esto se puede ver en el Q-Q plot, ya que los residuos no siguen la linea roja.

Homocedasticidad

H_0 : La varianza de los errores es constante.

H_1 : La varianza de los errores no es constante.

```r
bptest(regresion2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  regresion2
## BP = 7.5585, df = 3, p-value = 0.05607
```

Debido a que el valor p está por encima del alfa de 0.05 en la prueba de Breusch-Pagan, no se puede rechazar la hipotesis nula y se concluye que la varianza de los errores es constante, esto es bueno porque se concluye que si hay homocedasticidad.

Independencia

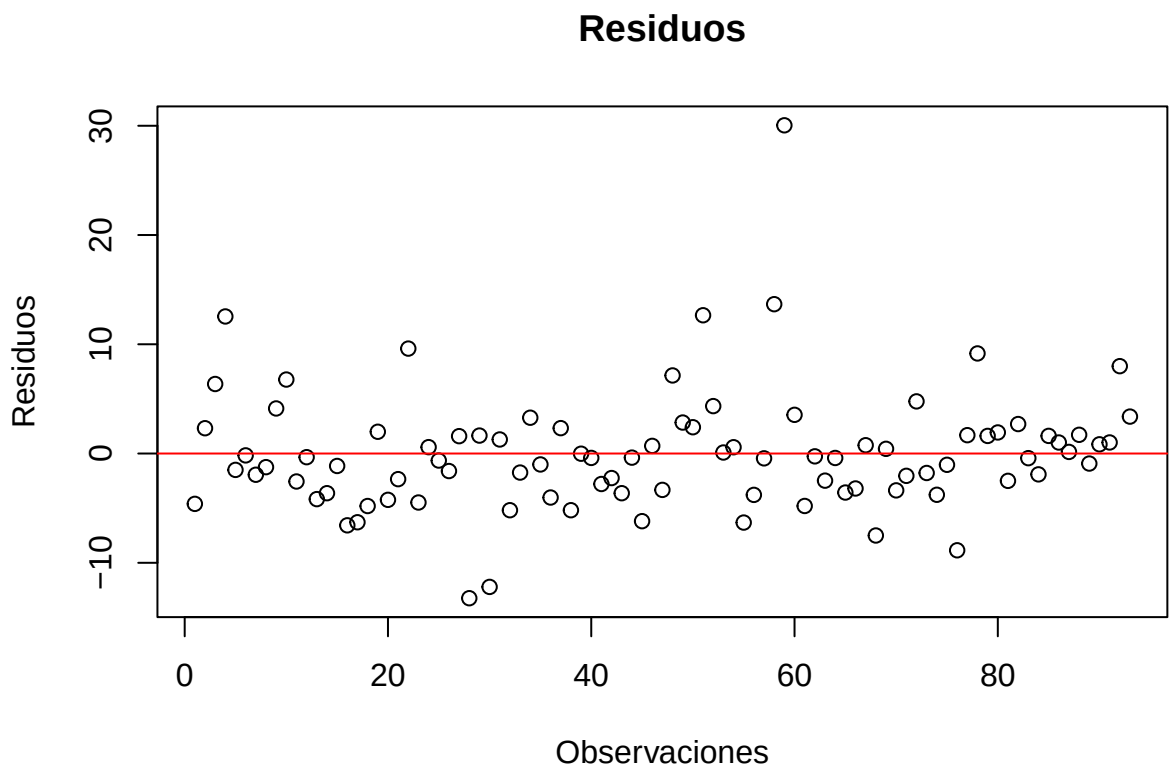
H_0 : Los residuos son independientes.

H_1 : Los residuos no son independientes.

```
dwtest(regresion2)
```

```
##
##  Durbin-Watson test
##
## data:  regresion2
## DW = 1.5357, p-value = 0.01113
## alternative hypothesis: true autocorrelation is greater than 0
```

```
plot(regresion2$residuals,main = "Residuos",xlab = "Observaciones",ylab = "Residuos")
abline(h=0,col="red")
```

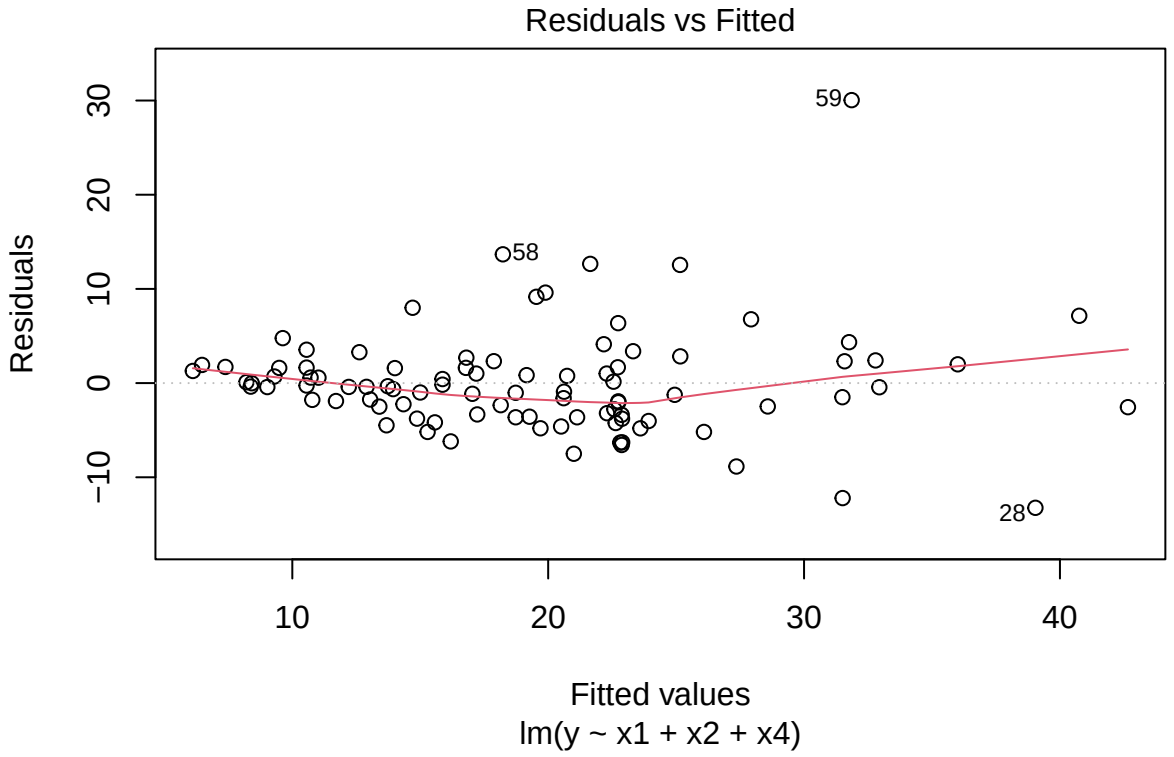


Debido al valor p de 0.011, que está por debajo del alfa de 0.05 en la prueba de Durbin-Watson, se rechaza la hipotesis nula y se concluye que los residuos no son independientes, esto se puede ver en la grafica anterior, ya que los residuos no son aleatorios, y se ven más cercanos siguiendo a la línea roja.

D. Identifique los datos atípicos y datos influyentes y describa los criterios implementados para su determinación.

Datos atípicos

```
plot(regresion2,which = 1)
```



Con esta gráfica podemos identificar que los datos atípicos son el 58, 59 y 28, ya que están muy alejados de los demás datos.

Datos influyentes

Segun los valores hat:

```
## Criterio de Influencia:  0.1075269
```

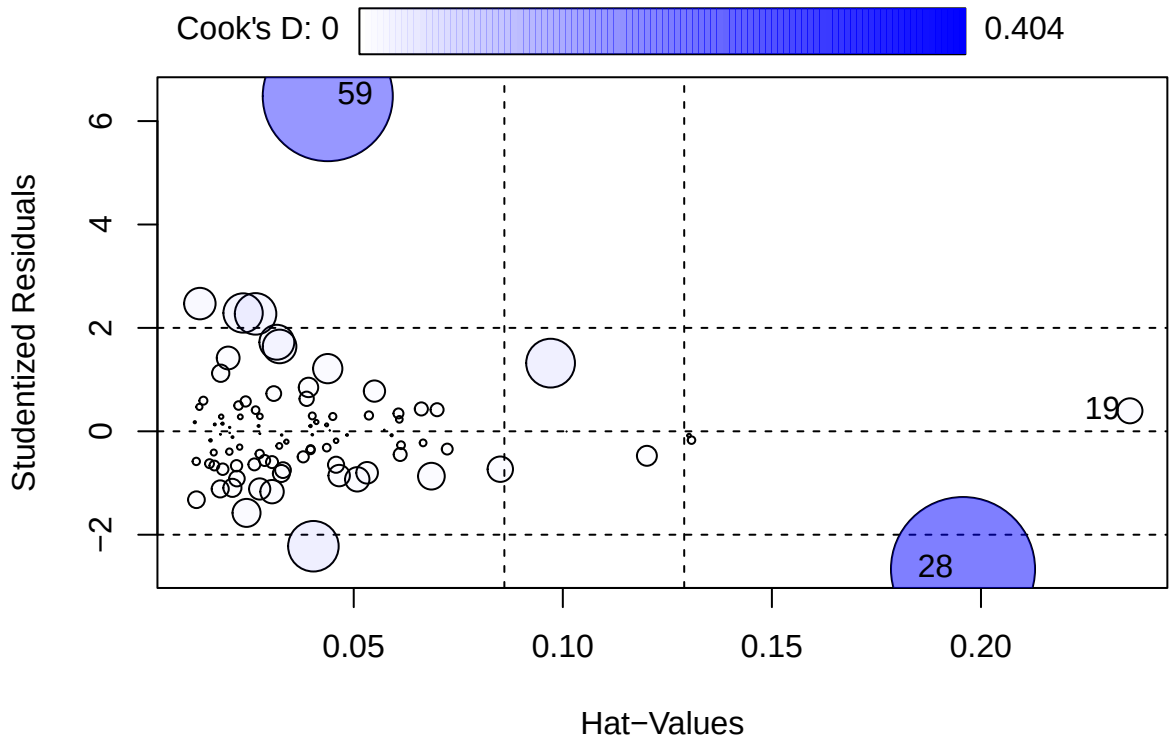
```
## Datos Influyentes:
```

```
##        11
## 0.1201075
##        19
## 0.2356269
##        28
```

```
## 0.1957038
##        57
## 0.1301909
##        89
## 0.130777
```

Segun el metodo de Cook:

```
influencePlot(regresion2,id=TRUE)
```




```
##       StudRes        Hat      CookD
## 19  0.3958737 0.23562685 0.01219291
## 28 -2.6643454 0.19570383 0.40412867
## 59  6.4819830 0.04378876 0.32927400
```

E. Calcule:

1. Intervalos de confianza para los coeficientes de regresión

```
confint(regresion2)
```

```
##                     2.5 %        97.5 %
```

```
## (Intercept) -7.598466e+01 -18.275111760
## x1           1.172915e-01   0.174783455
## x2           1.201408e-01   0.578140618
## x4           5.709513e-04   0.007441578
```

2. Intervalos de confianza para la respuesta media de la regresión

```
resmed=predict(regresion2, interval = "confidence")
head(resmed)
```

```
##        fit      lwr      upr
## 1 20.50582 18.44797 22.56366
## 2 31.58342 28.57358 34.59326
## 3 22.73520 21.20037 24.27002
## 4 25.15336 23.29978 27.00694
## 5 31.49819 28.67992 34.31646
## 6 15.87008 14.24652 17.49365
```

3. Intervalos de predicción

```
pred=predict(regresion2, interval = "prediction")
head(pred)
```

```
##        fit       lwr      upr
## 1 20.50582  8.939323 32.07232
## 2 31.58342 19.810219 43.35662
## 3 22.73520 11.250215 34.22018
## 4 25.15336 13.621450 36.68526
## 5 31.49819 19.772503 43.22388
## 6 15.87008  4.372904 27.36726
```

F. Interpreta los resultados desde la perspectiva estadística y en el contexto del problema.

Los intervalos de confianza para los coeficientes de regresión muestran la incertidumbre alrededor de estos valores estimados. Si los intervalos incluyen el cero, esos coeficientes podrían no ser significativos para predecir el precio de los carros.

Los intervalos de confianza para la respuesta media indican dónde se espera que estén los precios promedio de los carros con un 95% de confianza.

Los intervalos de predicción dan un rango esperado para los precios individuales de los carros con un 95% de confianza.

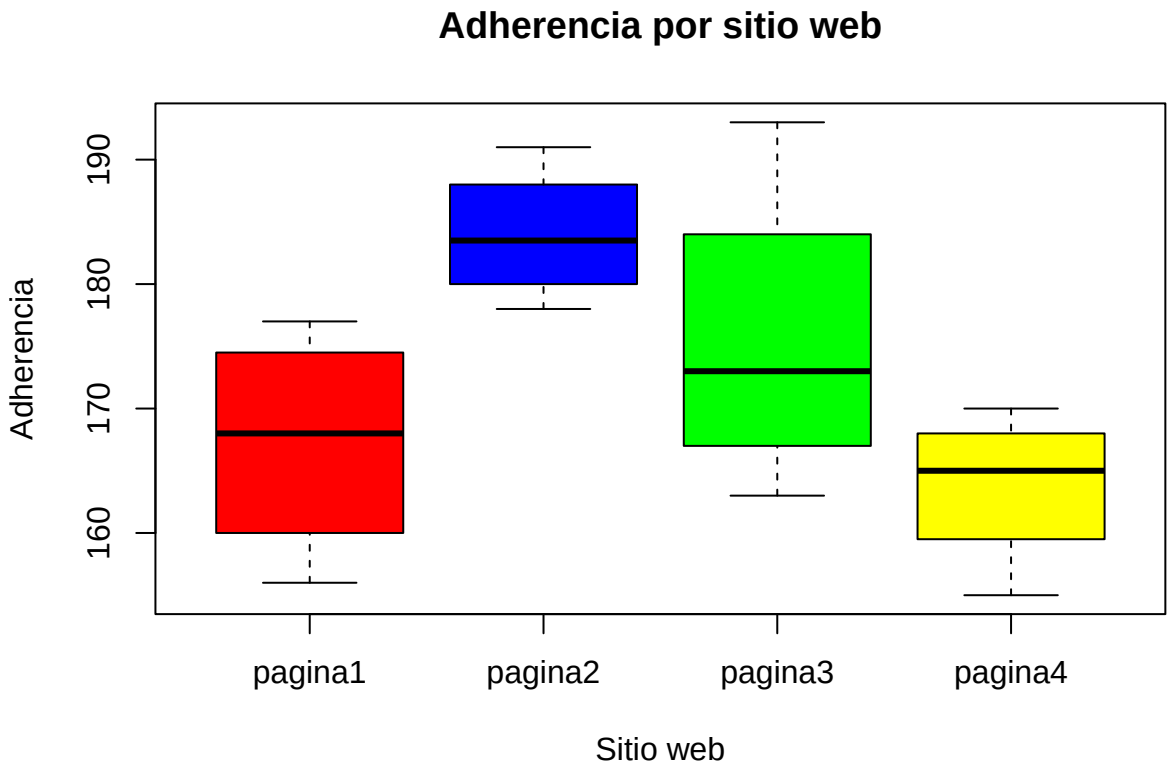
En resumen, estos intervalos revelan la incertidumbre alrededor de los coeficientes, ofrecen estimaciones para precios promedio y predicen el rango esperado de precios individuales de los carros.

Problema 3

```
##   pagina1 pagina2 pagina3 pagina4
## 1     164     178     175     155
## 2     172     191     193     166
## 3     177     182     171     164
## 4     156     185     163     170
```

A.Realice un gráfico de caja y bigotes para la adherencia por sitio web.

```
boxplot(datos,main = "Adherencia por sitio web",xlab = "Sitio web",ylab = "Adherencia",col = c("red","bl
```



B. Estime la media para la adherencia en cada sitio web.

```
cat("Media adherencia pagina 1:",mean(datos$pagina1))
```

```
## Media adherencia pagina 1: 167.25
```

```
cat("\nMedia adherencia pagina 2:",mean(datos$pagina2))
```

```
##
## Media adherencia pagina 2: 184
```

```
cat("\nMedia adherencia pagina 3:",mean(datos$pagina3))
```

```
##
## Media adherencia pagina 3: 175.5
```

```
cat("\nMedia adherencia pagina 4:",mean(datos$pagina4))
```

```
##
## Media adherencia pagina 4: 163.75
```

C. Obtenga los intervalos de confianza para la adherencia media en cada sitio.

```
cat("Intervalo de confianza adherencia pagina 1: [",t.test(datos$pagina1)$conf.int,"]")
```

```
## Intervalo de confianza adherencia pagina 1: [ 152.5868 181.9132 ]
```

D. Realice el análisis de varianza con un nivel de significancia de 0.05

```
anova_data = data.frame(ad = c(datos$pagina1,datos$pagina2,datos$pagina3,datos$pagina4),pagina = c(rep(
anova = aov(ad~pagina,data = anova_data)
summary(anova)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## pagina       3  981.2   327.1   4.138 0.0314 *
## Residuals   12  948.5    79.0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Debido a que se tiene un valor p de 0.0314, y es menor al alfa de 0.05, se rechaza la hipotesis nula y se concluye que al menos una de las medias es diferente.

E. Analiza la validez del modelo. Comprueba:

1. Normalidad

```
shapiro.test(anova$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  anova$residuals
## W = 0.98126, p-value = 0.9728
```

Debido a que el valor p es mucho mayor al alfa de 0.05 con un valor de 0.9728, no se rechaza la hipotesis nula y se concluye que los residuos siguen una distribución normal.

2. Homocedasticidad

```
bptest(anova)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  anova
## BP = 3.8696, df = 3, p-value = 0.2759
```

Debido a que el valor p es mayor al alfa de 0.05 con un valor de 0.2759, no se rechaza la hipotesis nula y se concluye que la varianza de los errores es constante.

3. Independencia

```
dwtest(anova)
```

```
##
##  Durbin-Watson test
##
## data:  anova
## DW = 1.9672, p-value = 0.1759
## alternative hypothesis: true autocorrelation is greater than 0
```

Debido a que el valor p es mayor al alfa de 0.05 con un valor de 0.1759, no se rechaza la hipotesis nula y se concluye que los residuos son independientes.

F. Interpreta el resultado desde la perspectiva estadística y en el contexto del problema.

Debido a que se rechazo la hipotesis nula en el analisis de varianza, se concluye que al menos una de las medias es diferente, y se puede ver en el gráfico de caja y bigotes que la media de la pagina 2 es mayor a las demás.

Tambien vemos que al validar los modelos, el modelo paso todos los supuestos por lo que es confiable.

Problema 4

```
data = wine

data$Class = ifelse(data$Type == 1,"one",ifelse(data$Type == 2,"two",ifelse(data$Type==3,"three",data$Ty

head(data)
```

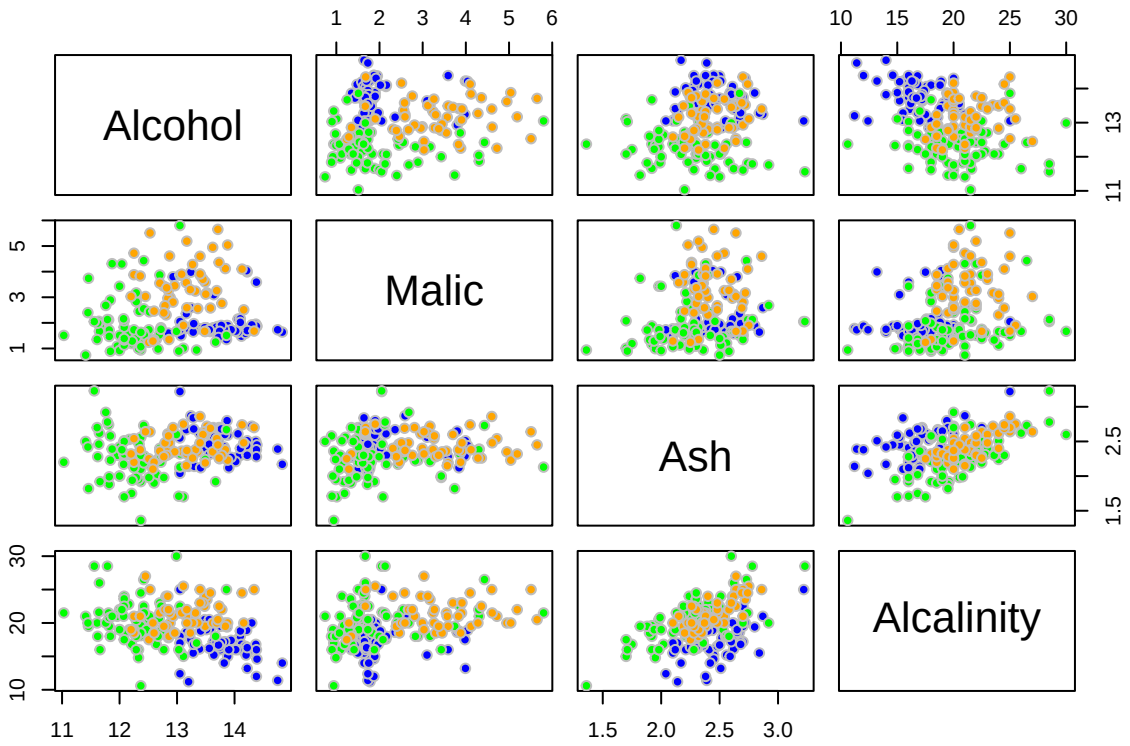
```
##   Type Alcohol Malic  Ash Alcalinity Magnesium Phenols Flavanoids Nonflavanoids
## 1    1   14.23  1.71 2.43       15.6       127    2.80       3.06          0.28
## 2    1   13.20  1.78 2.14       11.2       100    2.65       2.76          0.26
## 3    1   13.16  2.36 2.67       18.6       101    2.80       3.24          0.30
## 4    1   14.37  1.95 2.50       16.8       113    3.85       3.49          0.24
## 5    1   13.24  2.59 2.87       21.0       118    2.80       2.69          0.39
## 6    1   14.20  1.76 2.45       15.2       112    3.27       3.39          0.34
##   Proanthocyanins Color  Hue Dilution Proline Class
## 1            2.29  5.64 1.04     3.92    1065   one
## 2            1.28  4.38 1.05     3.40    1050   one
## 3            2.81  5.68 1.03     3.17    1185   one
## 4            2.18  7.80 0.86     3.45    1480   one
## 5            1.82  4.32 1.04     2.93     735   one
## 6            1.97  6.75 1.05     2.85    1450   one
```

A. Mediante un análisis discriminante realice una clasificación de la base de datos en los 3 diferentes grupos asociados los tipos de cultivares de vino.

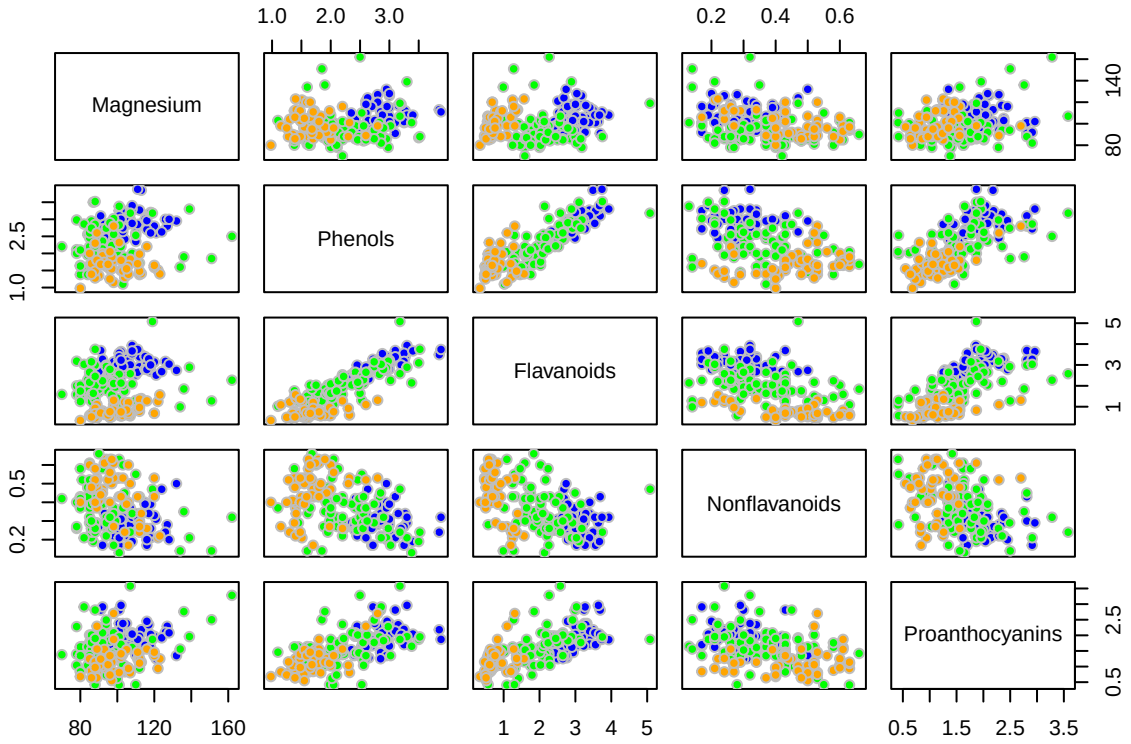
```
library(ggplot2)
#Asignamos un color a cada especie
color = c(one="blue",two="green",three="orange")
color
```

```
##      one      two    three
##   "blue"  "green" "orange"
```

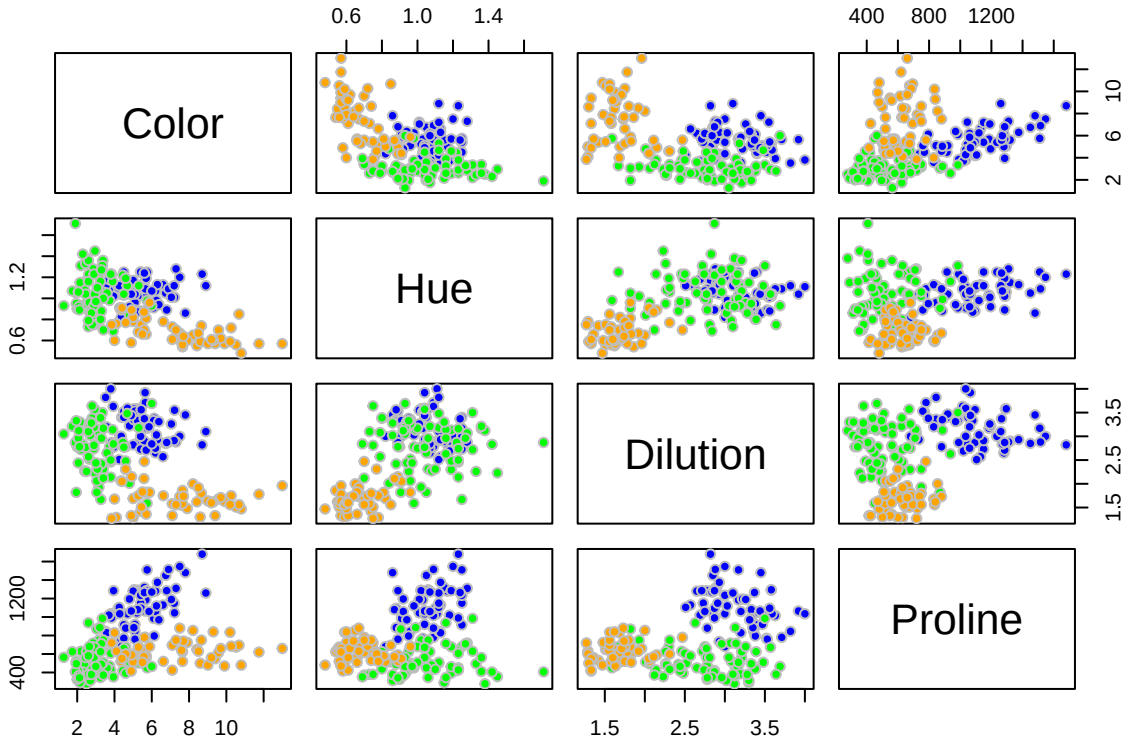
```
#Creamos un vector con el color corresponidente a cada observacion de acuerdo a la columna Species
col.ind=color[data$Class]
plot(data[2:5],pch=21,bg=col.ind,col="gray")
```




```
plot(data[6:10],pch=21,bg=col.ind,col="gray")
```




```
plot(data[11:14],pch=21,bg=col.ind,col="gray")
```



Debido a como se ve que las variables discriminan a los datos, usaremos las ultimas 4 variables para nuestro análisis discriminante. Las variables son Color, Hue, Dilution, Proline.

```
new_data = data[,c(1,11,12,13,14)]
head(new_data)
```

```
##   Type Color  Hue Dilution Proline
## 1    1  5.64 1.04     3.92    1065
## 2    1  4.38 1.05     3.40    1050
## 3    1  5.68 1.03     3.17    1185
## 4    1  7.80 0.86     3.45    1480
## 5    1  4.32 1.04     2.93     735
## 6    1  6.75 1.05     2.85    1450
```

```
discriminante = lda(Type~.,data = new_data)
discriminante
```

```
## Call:
## lda(Type ~ ., data = new_data)
##
## Prior probabilities of groups:
##         1         2         3
## 0.3314607 0.3988764 0.2696629
##
## Group means:
```

```
##      Color       Hue Dilution   Proline
## 1 5.528305 1.0620339 3.157797 1115.7119
## 2 3.086620 1.0562817 2.785352  519.5070
## 3 7.396250 0.6827083 1.683542  629.8958
##
## Coefficients of linear discriminants:
##                  LD1          LD2
## Color    -0.16631490 -0.313846263
## Hue       2.08901739  2.431407706
## Dilution  1.86101226  0.231141435
## Proline   0.00363052 -0.003900205
##
## Proportion of trace:
##    LD1    LD2
## 0.6886 0.3114
```

B. Escriba las funciones discriminantes implementadas por el modelo y el porcentaje de clasificación asociado a cada una de éstas.

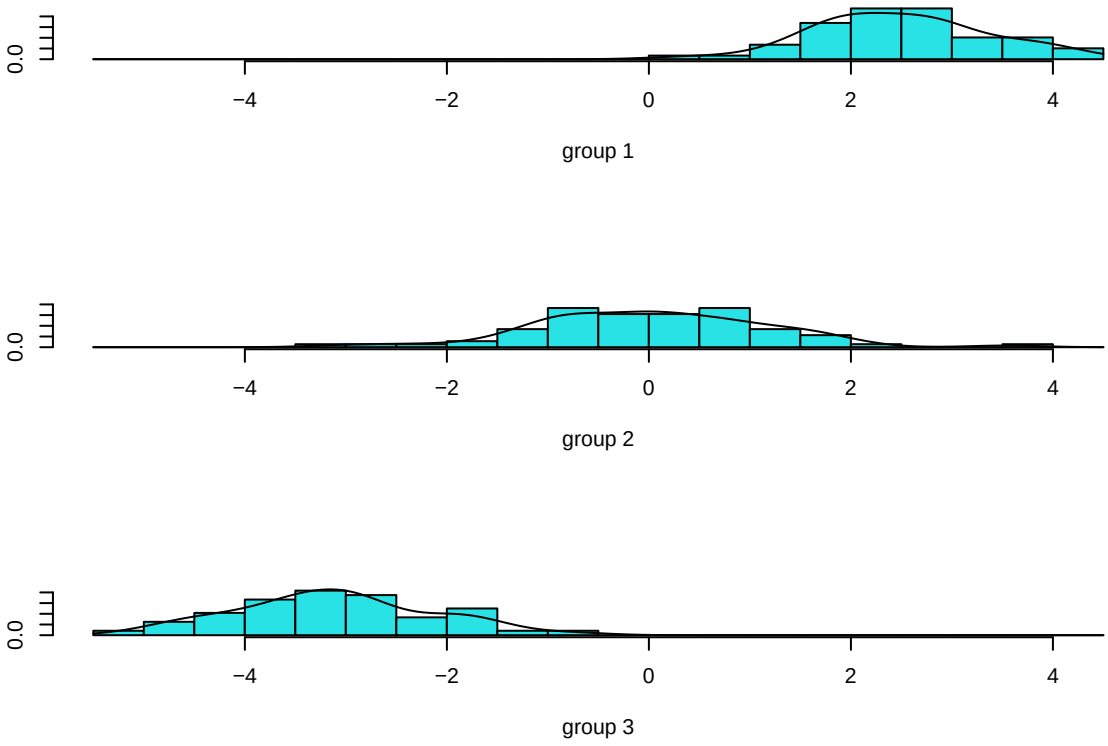
$LD1 = -0.166 * Color + 2.089 * Hue + 1.861 * Dilution + 0.004 * Proline$

$LD2 = -0.314 * Color + 2.431 * Hue + 0.231 * Dilution - 0.004 * Proline$

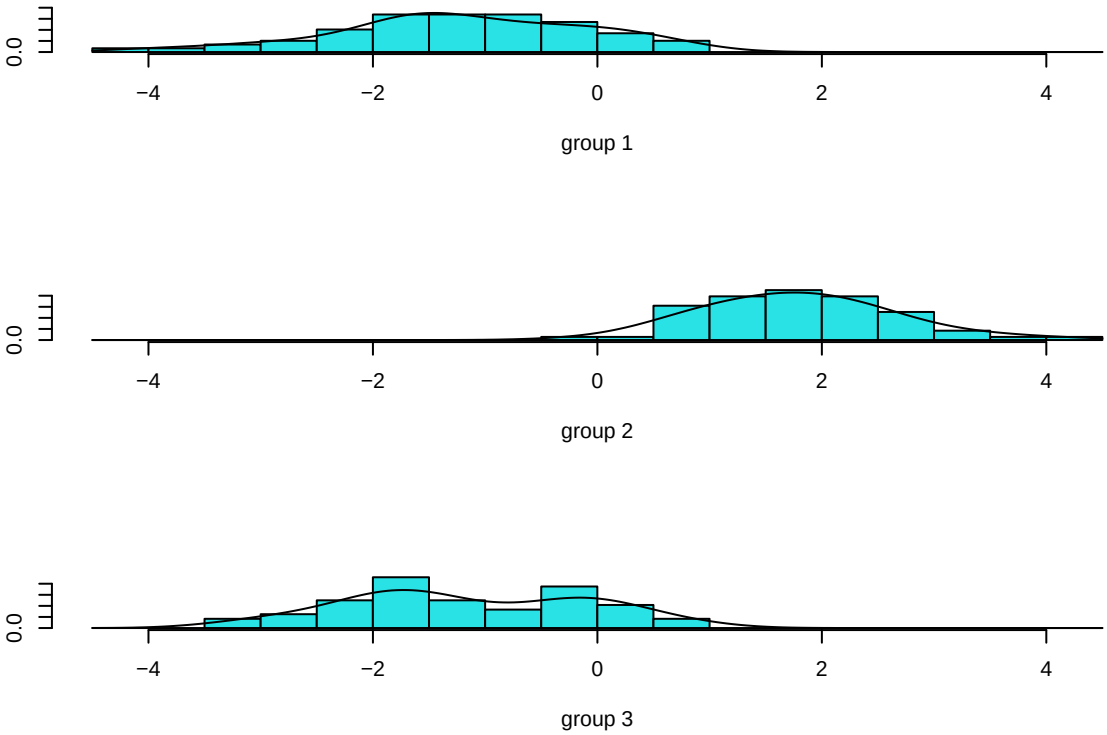
El LD1 representa el 68.8% de la clasificación y el LD2 el 31.1% de la clasificación.

C. Represente con histogramas la distribución de los valores asociados por cada función discriminante en cada categoría.

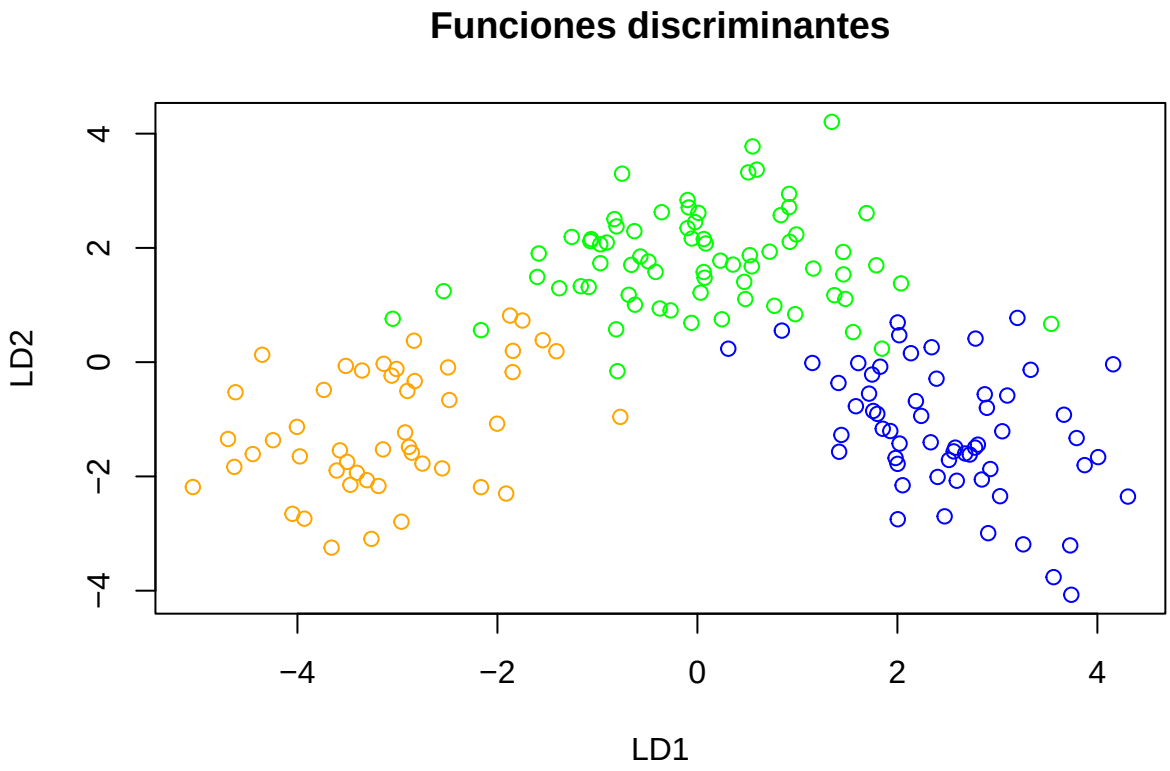
LD1



LD2



D. Represente visualmente sus resultados mediante un gráfico de dispersión con las funciones discriminantes.



Podemos ver en el gráfico que las funciones discriminantes separan bien los datos, por lo que podemos concluir que el análisis discriminante es bueno para este caso.

E. Determine la precisión del modelo.

```
table(data$Class,prediccion$class)
```

```
##
##           1  2  3
##   one    57  2  0
##   three  0  4 44
##   two    2 66  3
```

```
cat("El porcentaje de precision es: ",mean(prediccion$class == data$Class)*100,"%")
```

```
## El porcentaje de precision es:  0 %
```

Debido a que la precision es del 96%, este es un buen modelo.

Problema 5

```
##   X pregnant glucose pressure triceps insulin mass pedigree age diabetes
## 1 1        6     148       72      35      NA 33.6    0.627  50      pos
## 2 2        1      85       66      29      NA 26.6    0.351  31      neg
## 3 3        8     183       64      NA      NA 23.3    0.672  32      pos
## 4 4        1      89       66      23      94 28.1    0.167  21      neg
## 5 5        0     137       40      35     168 43.1    2.288  33      pos
## 6 6        5     116       74      NA      NA 25.6    0.201  30      neg
```

A. Prepare la base de datos omitiendo los datos faltantes.

```
data = na.omit(data)
head(data)
```

```
##     X pregnant glucose pressure triceps insulin mass pedigree age diabetes
## 4   4        1      89       66      23      94 28.1    0.167  21      neg
## 5   5        0     137       40      35     168 43.1    2.288  33      pos
## 7   7        3      78       50      32      88 31.0    0.248  26      pos
## 9   9        2     197       70      45     543 30.5    0.158  53      pos
## 14 14        1     189       60      23     846 30.1    0.398  59      pos
## 15 15        5     166       72      19     175 25.8    0.587  51      pos
```

```
# CHange diabetes values from neg and pos to 0 and 1
data$diabetes = ifelse(data$diabetes == "neg",0,1)
```

B. Divida el conjunto de datos en un conjunto de entrenamiento (80%) y un conjunto de prueba(20%)

```
library(caret)
```

```
## Loading required package: lattice
```

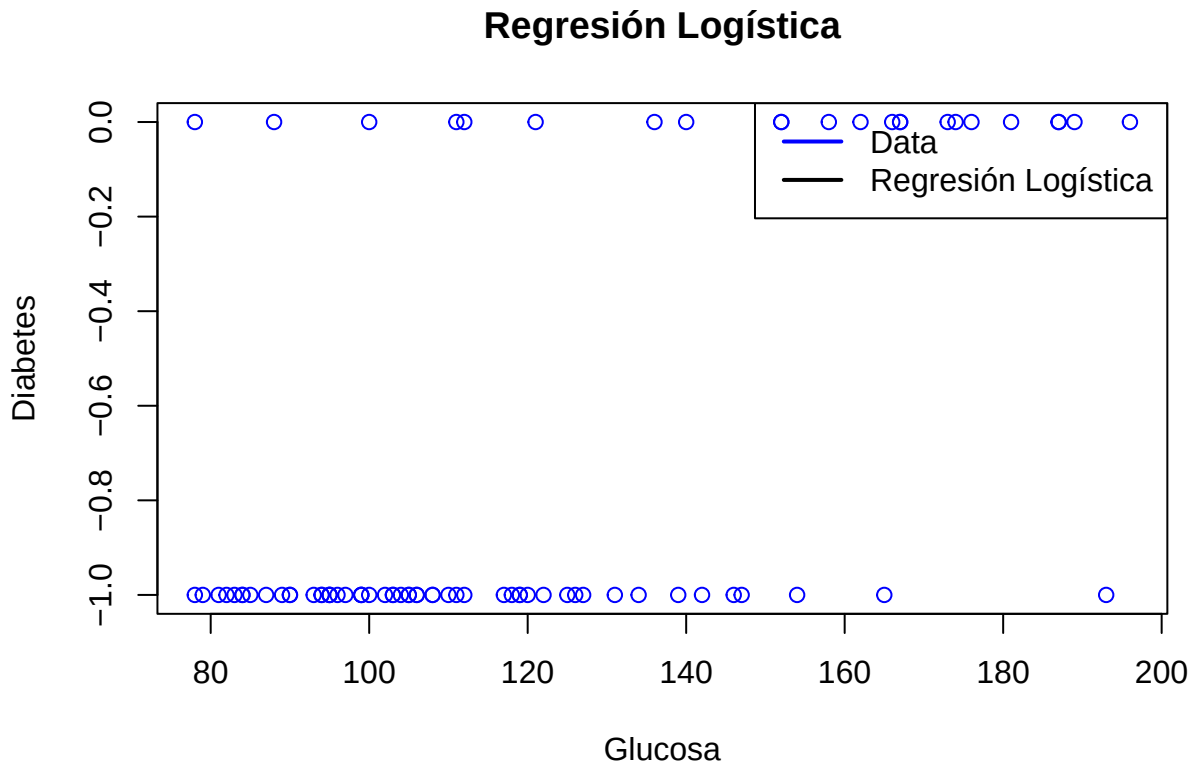
```
set.seed(123)
inTrain = createDataPartition(data$diabetes, p = 0.8, list = FALSE)
training = data[inTrain,]
testing = data[-inTrain,]
```

C. Considerando Diabetes como variable dependiente, formule un modelo de regresión logarítmica con el cual predecir la probabilidad de que un paciente sea positivo para diabetes basado en la concentración de glucosa.

```
modelo = glm(diabetes~glucose,family = binomial(link = "logit"),data = training)
summary(modelo)
```

```
##
## Call:
## glm(formula = diabetes ~ glucose, family = binomial(link = "logit"),
##     data = training)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.939733   0.706029  -8.413  < 2e-16 ***
## glucose      0.041555   0.005354   7.762 8.38e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 402.89  on 313  degrees of freedom
## Residual deviance: 319.14  on 312  degrees of freedom
## AIC: 323.14
##
## Number of Fisher Scoring iterations: 4
```

D. Grafique la curva de regresión logística.



E. Ajuste un modelo de regresión logística múltiple. Justifique la selección de las variables predictoras.

```
modelo2 = glm(diabetes~glucose+insulin+pressure+mass,family = binomial(link = "logit"),data = training)
summary(modelo2)
```

```
##
## Call:
## glm(formula = diabetes ~ glucose + insulin + pressure + mass,
##     family = binomial(link = "logit"), data = training)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.287e+00  1.200e+00  -6.907 4.95e-12 ***
## glucose      3.903e-02  6.296e-03   6.199 5.69e-10 ***
## insulin     -2.098e-05  1.403e-03  -0.015  0.98808
## pressure     5.753e-03  1.215e-02   0.473  0.63596
## mass         6.707e-02  2.289e-02   2.931  0.00338 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 402.89  on 313  degrees of freedom
## Residual deviance: 307.45  on 309  degrees of freedom
## AIC: 317.45
##
## Number of Fisher Scoring iterations: 4
```

Se usaron estas variables predictoras debido a la relacion que se cree que se tiene entre la insulina y la diabetes, y la concepcion que hay de que la presion y el peso tienen algo que ver con la diabetes.

F. Evalúe el rendimiento del modelo sobre los individuos del conjunto de prueba.

```
prediccion = predict(modelo2, newdata = testing, type = "response")
prediccion = ifelse(prediccion > 0.5, 1, 0)
conf =table(testing$diabetes, prediccion)
conf
```

```
##    prediccion
##      0  1
##   0 49  6
##   1  8 15
```

```
acc = sum(diag(conf))/sum(conf)
cat("El porcentaje de exactitud es: ",acc*100,"%")
```

```
## El porcentaje de exactitud es:  82.05128 %
```

Este no es un gran modelo, pero tiene algunas predicciones buenas. El porcentaje de exactitud es del 75%.