Actividad 2.1. Regresión lineal simple: Método de mínimos cuadrados

Franco Mendoza Muraira A01383399

2023-11-01

```
library(lmtest)
```

```
## Warning: package 'lmtest' was built under R version 4.2.3
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.2.3
```

```
##
## Attaching package: 'zoo'
```

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

```
library(nortest)
```

```
df = read.csv(file="mcdonaldsmenu.csv")
head(df,3)
```

```
##    Category              Item   Serving.Size Calories Calories.from.Fat
## 1 Breakfast      Egg McMuffin 4.8 oz (136 g)      300               120
## 2 Breakfast Egg White Delight 4.8 oz (135 g)      250                70
## 3 Breakfast  Sausage McMuffin 3.9 oz (111 g)      370               200
##   Total.Fat Total.Fat....Daily.Value. Saturated.Fat
## 1        13                        20             5
## 2         8                        12             3
## 3        23                        35             8
##   Saturated.Fat....Daily.Value. Trans.Fat Cholesterol
## 1                            25         0         260
## 2                            15         0          25
## 3                            42         0          45
##   Cholesterol....Daily.Value. Sodium Sodium....Daily.Value. Carbohydrates
## 1                          87    750                    31            31
## 2                           8    770                    32            30
## 3                          15    780                    33            29
##   Carbohydrates....Daily.Value. Dietary.Fiber Dietary.Fiber....Daily.Value.
```

```
## 1                           10             4                           17
## 2                           10             4                           17
## 3                           10             4                           17
##   Sugars Protein Vitamin.A....Daily.Value. Vitamin.C....Daily.Value.
## 1      3      17                       10                         0
## 2      3      18                        6                         0
## 3      2      14                        8                         0
##   Calcium....Daily.Value. Iron....Daily.Value.
## 1                      25                   15
## 2                      25                    8
## 3                      25                   10
```

1. Analisis Exploratorio

Matriz de Covarianza de las variables elegidas: ("Protein", "Calories", "Cholesterol", "Total.Fat", "Carbohydrates", "Sodium")

```
cuant_df = df[,c("Protein","Calories","Cholesterol","Total.Fat","Carbohydrates","Sodium")]
cov(cuant_df)
```

```
##                 Protein  Calories Cholesterol Total.Fat Carbohydrates
## Protein        130.5568  2162.924    559.9617  131.1176      113.6700
## Calories      2162.9240 57729.618  12505.4017 3086.9958     5305.2153
## Cholesterol    559.9617 12505.402   7615.9233  843.7065      668.1089
## Total.Fat      131.1176  3086.996    843.7065  201.8104      185.1086
## Carbohydrates  113.6700  5305.215    668.1089  185.1086      798.1886
## Sodium        5734.7645 98755.936  31440.7770 6936.1593     3273.4266
##                   Sodium
## Protein         5734.764
## Calories       98755.936
## Cholesterol    31440.777
## Total.Fat       6936.159
## Carbohydrates   3273.427
## Sodium        332959.377
```

Matriz de correlacion de las variables elegidas

```
cor(cuant_df)
```

```
##                 Protein  Calories Cholesterol Total.Fat Carbohydrates    Sodium
## Protein       1.0000000 0.7878475   0.5615614 0.8077730     0.3521222 0.8698016
## Calories      0.7878475 1.0000000   0.5963992 0.9044092     0.7815395 0.7123087
## Cholesterol   0.5615614 0.5963992   1.0000000 0.6805474     0.2709775 0.6243619
## Total.Fat     0.8077730 0.9044092   0.6805474 1.0000000     0.4612135 0.8461584
## Carbohydrates 0.3521222 0.7815395   0.2709775 0.4612135     1.0000000 0.2007956
## Sodium        0.8698016 0.7123087   0.6243619 0.8461584     0.2007956 1.0000000
```

Se elige de variable predictora "Protein", y su variable independiente "Sodium", esto debido a la alta correlación entre las 2 variables.

2. Método de mínimos cuadrados:

Pendiente (b1)

Primero separamos los datos a un dataframe que tenga solo estas 2 variables, este dataframe se llama data.

```
data=cuant_df[,c("Protein","Sodium")]
b1=cov(data)['Protein','Sodium']/var(data[,'Sodium'])
b1
```

Ahora para sacar b1 dividimos la covarianza de las 2 varianza entre la varianza de "x", en este caso "Sodium".

```
## [1] 0.01722362
```

beta0 = Protein - 0.01722362*Sodium

Para sacar b0 usamos la formula anterior, usando las medias de ambas variables.

Intercept (b0)

```
b0 = mean(cuant_df$Protein)-b1*mean(cuant_df$Sodium)
b0
```

```
## [1] 4.799854
```

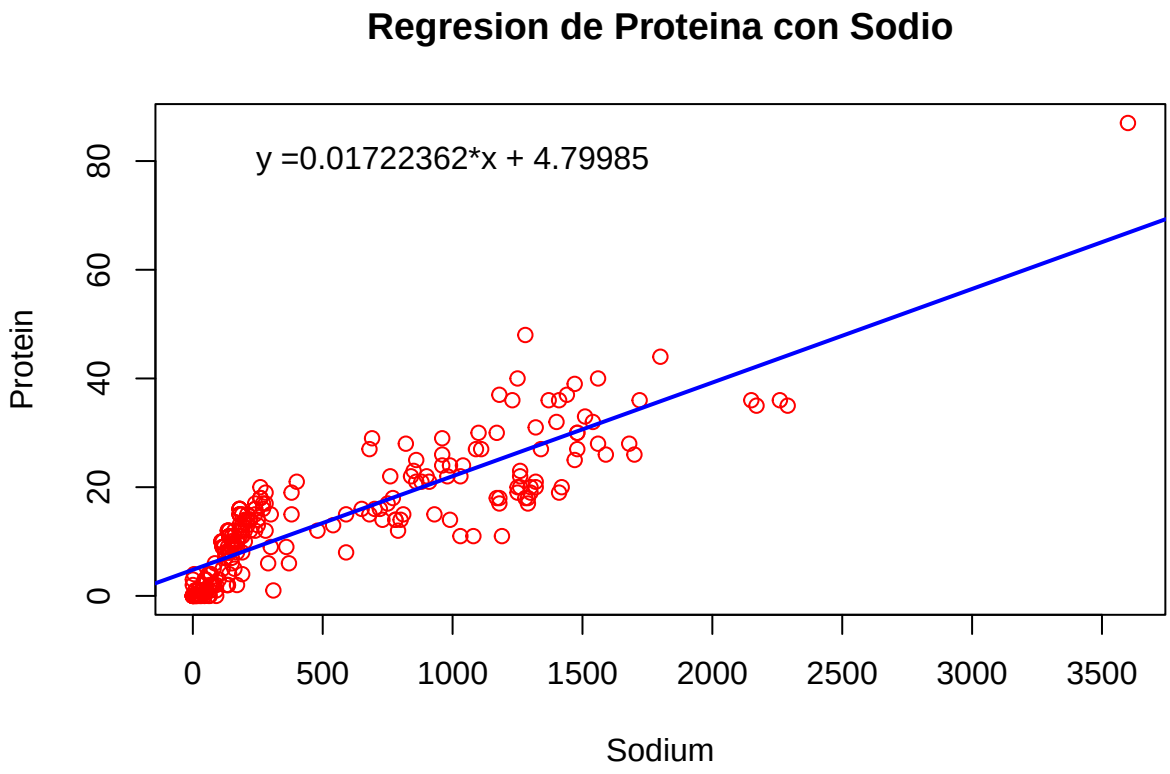
3. Regresion lineal en R

```
A = lm(data$Protein~data$Sodium)
A
```

```
##
## Call:
## lm(formula = data$Protein ~ data$Sodium)
##
## Coefficients:
## (Intercept)  data$Sodium
##     4.79985      0.01722
```

4. Representacion Grafica

```
plot( data$Sodium, data$Protein ,col='red' ,ylab='Protein' ,xlab='Sodium' ,main='Regresion de Proteina c
abline(A, col='blue' , lwd=2)
text(1000,80, "y =0.01722362*x + 4.79985")
```



5. Coeficiente de determinacion

```
summary(A)
```

```
##
## Call:
## lm(formula = data$Protein ~ data$Sodium)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.2960  -4.7999  -0.0672   3.7888  21.1539
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.7998539  0.4622223   10.38   <2e-16 ***
## data$Sodium 0.0172236  0.0006083   28.32   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.649 on 258 degrees of freedom
## Multiple R-squared:  0.7566, Adjusted R-squared:  0.7556
## F-statistic: 801.8 on 1 and 258 DF,  p-value: < 2.2e-16
```

Terminamos con un R cuadrado de 0.7566 lo cual nos dice que la variable Sodium nos explica en un 75% de la variabilidad de Protein en el modelo.

6. Validacion del modelo

Usando un α de 0.05

Significancia de los coeficientes de regresion

```
summary(A)
```

```
##
## Call:
## lm(formula = data$Protein ~ data$Sodium)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.2960  -4.7999  -0.0672   3.7888  21.1539
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.7998539  0.4622223   10.38   <2e-16 ***
## data$Sodium 0.0172236  0.0006083   28.32   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.649 on 258 degrees of freedom
## Multiple R-squared:  0.7566, Adjusted R-squared:  0.7556
## F-statistic: 801.8 on 1 and 258 DF,  p-value: < 2.2e-16
```

Pendiente $H_0 : \beta_1 = 0$ Si no se rechaza H_0, el valor de β_1 no es significante, la pendiente seria 0.

$H_1 : \beta_1 \neq 0$ Si se rechaza H_0 el valor de β_1 es significante, la pendiente si afectaria el modelo.

Intercept $H_0 : \beta_0 = 0$ Si no se rechaza H_0, el valor de β_0 no es significante, la interseccion en el eje y seria en 0.

$H_1 : \beta_0 \neq 0$ Si se rechaza H_0 el valor de β_0 es significante, la interseccion si afectaria el modelo.

Los pvalue que nos dieron de 2e-16 son menor que el alfa que pusimos ya sea de de 0.05 o de 0.1 por lo que se rechaza H_0, por ende los coeficientes de la regresion lineal son significativos.

Media cero de los residuos

```
t.test(A$residuals)
```

```
##
##  One Sample t-test
##
## data:  A$residuals
## t = 2.6414e-16, df = 259, p-value = 1
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.6884874  0.6884874
## sample estimates:
##    mean of x
## 9.235234e-17
```

H_0 : La media de los residuos es igual a 0. H_1 : La media de los residuos no es igual a 0.

El pvalue obtenido es 1, lo cual es muy alto. Este valor nos indica que no hay suficiente evidencia para rechazar la hipótesis nula. Esto sugiere que no hay diferencias estadísticamente significativas entre la media de los residuos y el valor de 0.

Distribucion Normal de los residuos

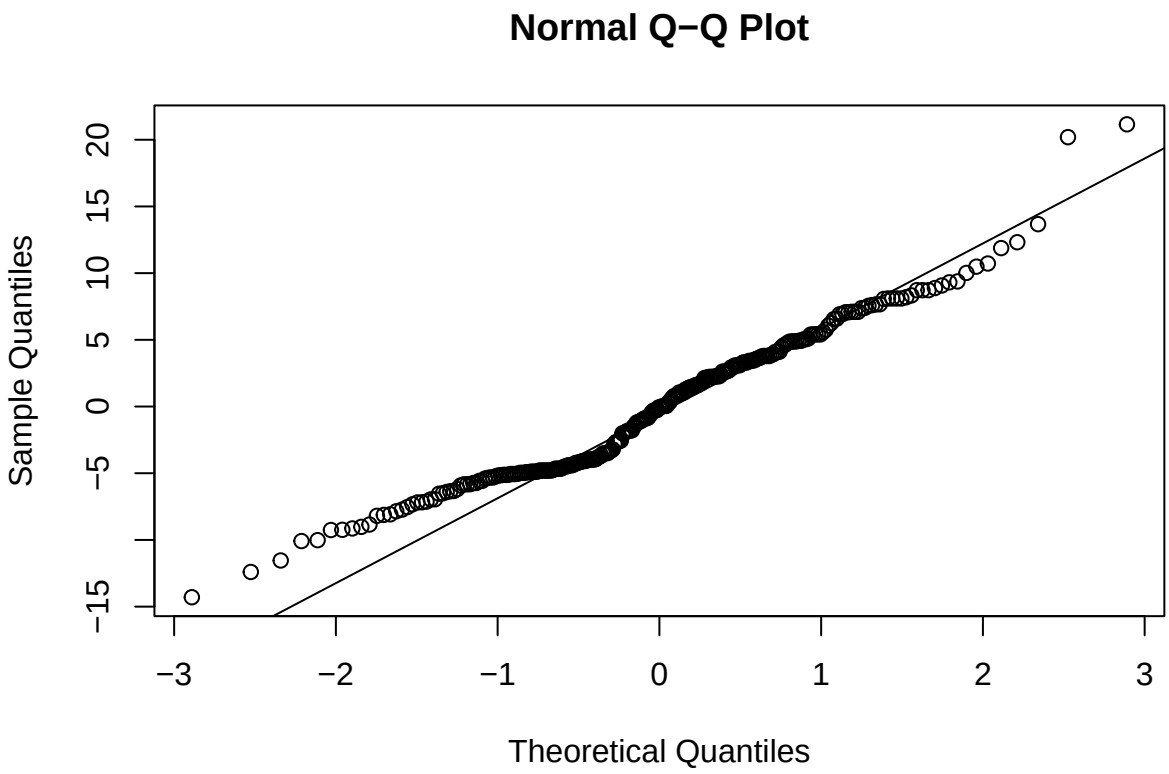
```
ad.test(A$residuals)
```

```
##
##  Anderson-Darling normality test
##
## data:  A$residuals
## A = 2.3721, p-value = 5.117e-06
```

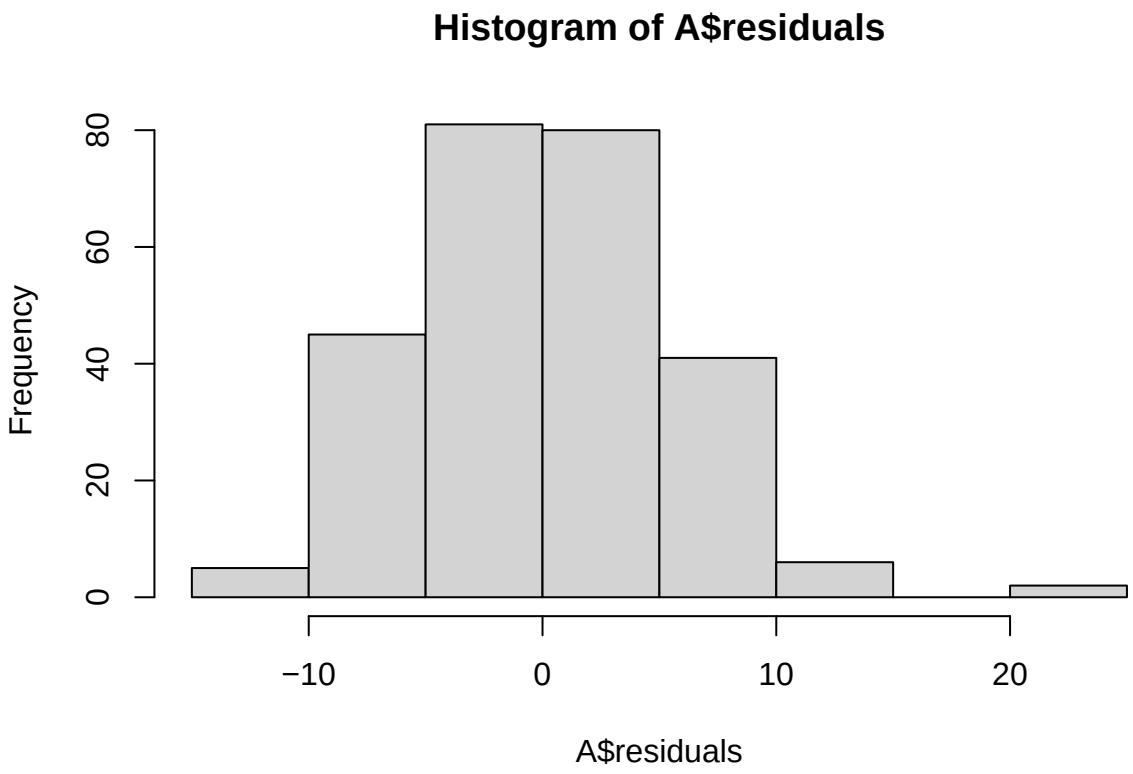
H_0 : Los residuos siguen una distribución normal. H_1 : Los residuos no siguen una distribución normal.

El pvalue tan bajo obtenido sugiere evidencia en contra de la hipótesis nula. Por lo tanto, se rechaza la hipótesis nula, lo que indica que hay suficiente evidencia para concluir que los residuos no siguen una distribución normal, lo que se puede ver en las siguientes graficas.

```
qqnorm(A$residuals)
qqline(A$residuals)
```




```r
hist(A$residuals)
```



Se puede ver en las graficas que no hay normalidad

Homocedasticidad (varianza constante)

```
#install.packages("lmtest")
library(lmtest)
```

```
bptest(A)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  A
## BP = 49.05, df = 1, p-value = 2.496e-12
```

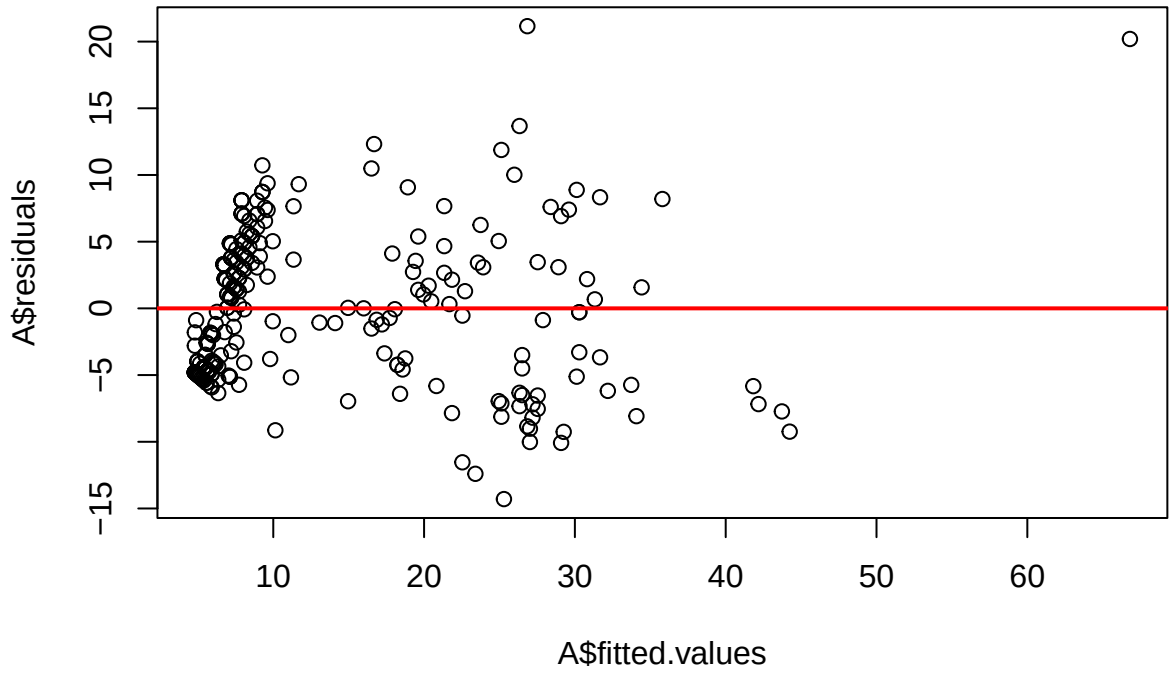
```
bptest(A,  varformula= ~ data$Sodium + I(data$Sodium^2) , data=data)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  A
## BP = 68.477, df = 2, p-value = 1.35e-15
```

H_0 : No hay heterocedasticidad en los residuos. H_1 : Existe heterocedasticidad en los residuos.

En ambos casos, los pvalue que estan por debajo del α sugieren la presencia de heterocedasticidad en los residuos del modelo de regresión, lo que lleva al rechazo de la hipótesis nula. Esto significa que la varianza de los errores es constante y que depende de los coeficientes.

```
plot(A$fitted.values,A$residuals)
abline(h=0,col='red',lwd=2)
```



Independecia

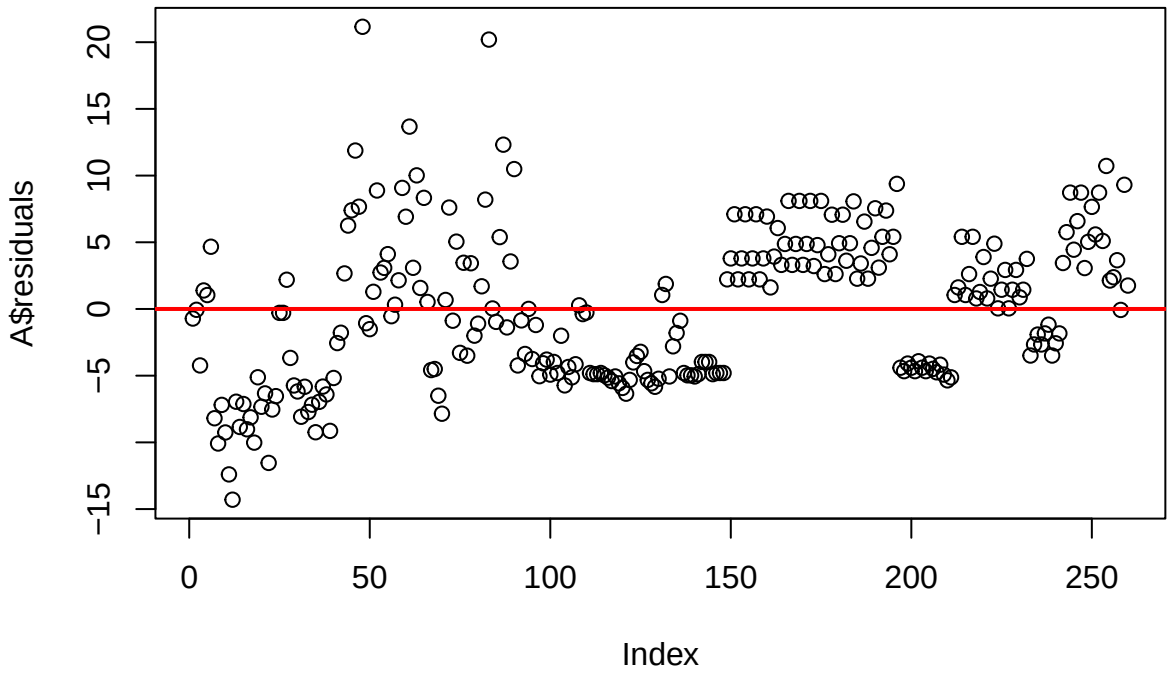
```
dwtest(A)
```

```
##
##  Durbin-Watson test
##
## data:  A
## DW = 0.62965, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

H_0 : autocorrelacion en los residuos $= 0$ H_1 : autocorrelacion en los residuos $\neq 0$

Se rechaza H_0 debido al pvalue que es menor que el α, lo que nos dice que hay evidencia de autocorrelación en los residuos del modelo de regresión. Esto significa que los errores sí están correlacionados.

```r
plot(A$residuals)
abline(h=0,col="red",lwd=2)
```



7. Conclusiones

Se pudo conseguir el modelo de regresion correctamente usando la variable predictora de el sodio, con la variable dependiente siendo la proteina. Usando las formulas vistas en clase, el modelo salió de la misma forma que con la función lm de R. En el modelo en sí que se pudo ver que tiene coeficientes altamente significativos debido a sus pvalues tan pequeños, lo que nos dice que hay una relación alta entre las 2 variables. Se pudo observar que la relación fue significativa, con un error estándar residual de 5.649. Podemos concluir que el modelo se ajusta bien a los datos, aunque no es un modelo perfecto ya que hay muchas predicciones erróneas, y lejas de los datos originales, pero aún así se ve una tendencia positiva correcta de parte del modelo de regresión.