Tarea3. Transformaciones
Marcelo Márquez Murillo
A01720588

```r
library(moments)
library(MASS)
library(e1071)
```

```
##
## Attaching package: 'e1071'
```

```
## The following objects are masked from 'package:moments':
##
##     kurtosis, moment, skewness
```

```r
library(nortest)
library(VGAM)
```

```
## Loading required package: stats4
```

```
## Loading required package: splines
```

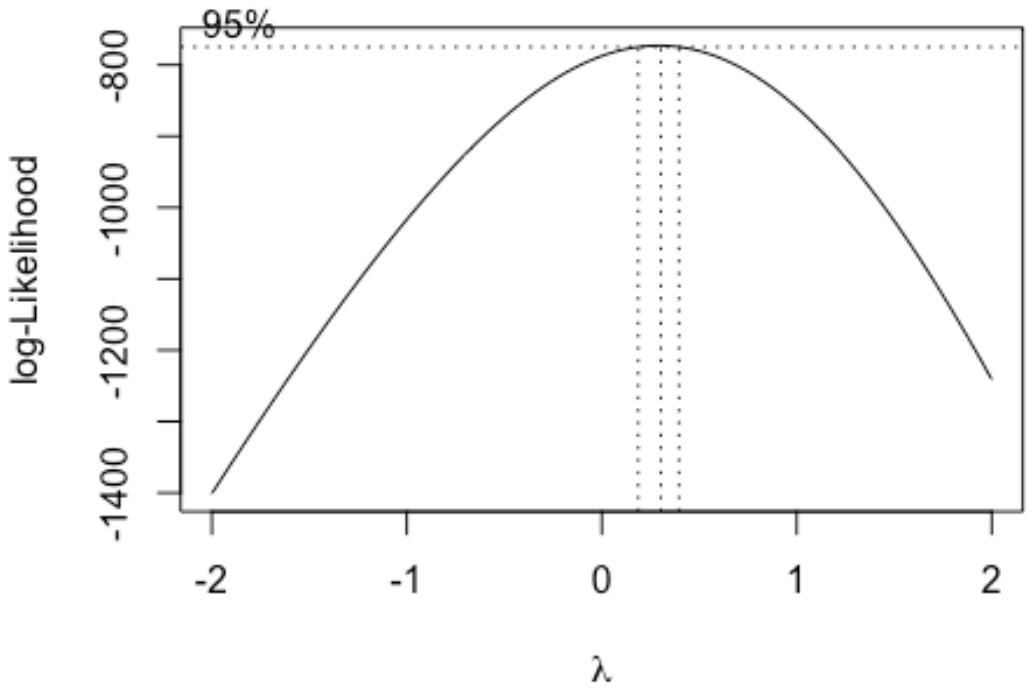
```r
M=read.csv("mc-donalds-menu-1.csv") #Leer la base de datos
fat = M$Total.Fat
print(fat)
```

```
##   [1]  13.0   8.0  23.0  28.0  23.0  23.0  26.0  30.0  20.0  25.0  27.0
31.0
##  [13]  33.0  37.0  27.0  32.0  20.0  24.0  32.0  21.0  15.0  22.0  31.0
26.0
##  [25]  31.0  25.0  35.0  48.0  52.0  37.0  41.0  56.0  60.0  46.0  50.0
9.0
##  [37]  24.0  16.0   9.0  19.0   4.0   4.0  27.0  26.0  29.0  31.0  27.0
43.0
##  [49]   8.0  11.0  21.0  40.0  17.0  22.0  22.0  23.0  26.0  22.0   9.0
33.0
##  [61]  20.0  28.0  15.0  38.0  25.0  19.0  16.0  24.0  21.0  16.0  32.0
19.0
##  [73]  31.0  18.0  33.0  20.0  23.0  10.0  12.0  18.0  30.0  59.0 118.0
19.0
##  [85]   7.0  21.0   8.0   4.5  22.0   8.0  15.0   8.0  15.0   8.0  20.0
13.0
##  [97]  11.0  16.0  24.0   5.0   0.0   0.0   2.0  13.0   8.0   6.0   1.5
9.0
## [109]   8.0   6.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
0.0
## [121]   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   2.5
```

```
0.0
## [133]   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
0.0
## [145]   0.0   0.0   0.0   0.0   9.0  10.0  14.0   9.0  10.0  14.0   9.0
10.0
## [157]  14.0   9.0  10.0  14.0   9.0  10.0  14.0   0.0   0.0   0.5   0.0
0.0
## [169]   0.5   0.0   0.0   0.5   0.0   0.0   0.5   0.0   0.0   0.5  11.0
14.0
## [181]  17.0   3.5   3.5   4.0  11.0  14.0  17.0   3.5   3.5   3.5  13.0
16.0
## [193]  20.0   3.5   3.5   3.5   4.5   7.0   9.0   4.5   7.0   9.0   4.5
7.0
## [205]   9.0   4.5   7.0   9.0   4.5   7.0   9.0  11.0  13.0  16.0   5.0
5.0
## [217]   6.0  11.0  13.0  16.0   5.0   5.0   6.0  18.0  22.0  26.0  19.0
23.0
## [229]  27.0  23.0  26.0  31.0   0.5   1.0   1.0   0.5   1.0   1.0   0.5
1.0
## [241]   1.0  15.0  19.0  23.0  16.0  20.0  24.0  16.0  20.0  23.0  19.0
23.0
## [253]  23.0  33.0  15.0  17.0  23.0  11.0  32.0  16.0
```

1. Utiliza la transformación Box-Cox. Utiliza el modelo exacto y el aproximado de acuerdo con las sugerencias de Box y Cox para la transformación

```
bc = boxcox((fat+1)~1)
```




```
lambda = bc$x[which.max(bc$y)]
cat("Lambda:", lambda, "\n")

## Lambda: 0.3030303

fat1 = sqrt(fat+1)
fat2 = (((fat + 1)^lambda) - 1) / lambda
```

2. Escribe las ecuaciones de los modelos encontrados.

$$fat1 = \sqrt{x+1}$$
$$fat2 = \frac{(x+1)^{0.3} - 1}{0.3l}$$

3. Analiza la normalidad de las transformaciones obtenidas con los datos originales. Utiliza como argumento de normalidad:

- Compara las medidas: Mínimo, máximo, media, mediana, cuartil 1 y cuartil 3, sesgo y curtosis.
- Obten el histograma de los 2 modelos obtenidos (exacto y aproximado) y los datos originales.
- Realiza la prueba de normalidad de Anderson-Darling o de Jarque Bera para los datos transformados y los originales

```r
# Fat
D = ad.test(fat)
summary(fat)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   2.375  11.000  14.165  22.250 118.000
```

```r
cat("Curtosis:", kurtosis(fat), "\n")
```

```
## Curtosis: 10.35171
```

```r
cat("Sesgo:", skewness(fat), "\n")
```

```
## Sesgo: 2.128023
```

```r
cat("P-value:", D$p.value, "\n")
```

```
## P-value: 1.46366e-16
```

```r
# Fat1
D1 = ad.test(fat1)
summary(fat1)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.836   3.464   3.450   4.822  10.909
```

```r
cat("Curtosis:", kurtosis(fat1), "\n")
```

```
## Curtosis: -0.08053187
```

```r
cat("Sesgo:", skewness(fat1), "\n")
```

```
## Sesgo: 0.3078819
```

```r
cat("P-value:", D1$p.value, "\n")
```

```
## P-value: 6.861263e-10
```

```r
# Fat2
D2 = ad.test(fat2)
summary(fat2)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.469   3.707   3.433   5.262  10.743
```

```r
cat("Curtosis:", kurtosis(fat2), "\n")
```

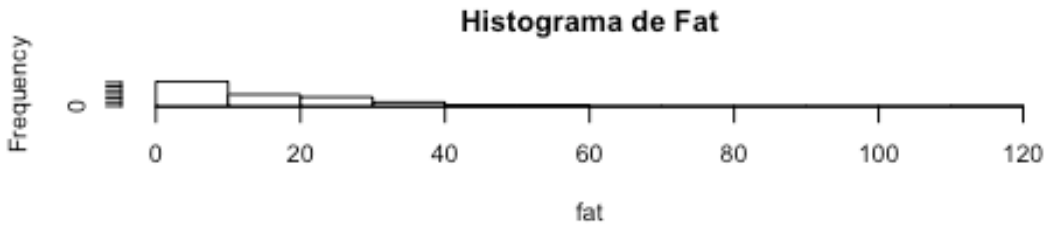
```
## Curtosis: -0.851942
```

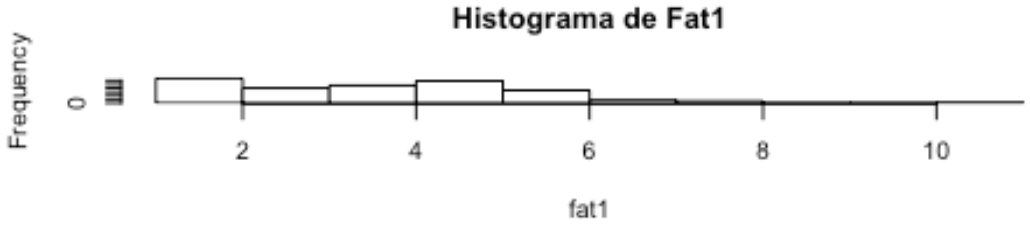
```r
cat("Sesgo:", skewness(fat2), "\n")
```

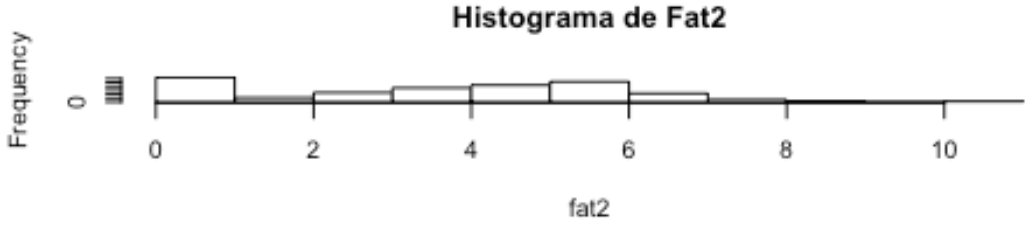
```
## Sesgo: -0.1151632
```

```r
cat("P-value:", D2$p.value, "\n")
```

```
## P-value: 1.380766e-14

# Histogramas de las variables
par(mfrow = c(3, 1))
hist(fat, col = 0, main = "Histograma de Fat")
hist(fat1, col = 0, main = "Histograma de Fat1")
hist(fat2, col = 0, main = "Histograma de Fat2")
```

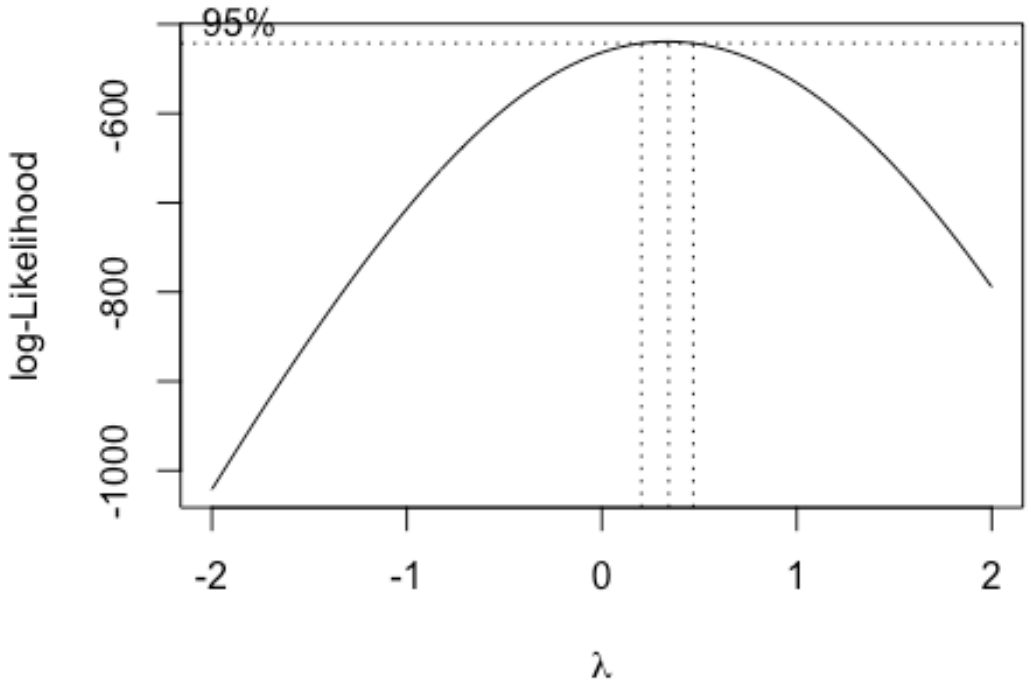




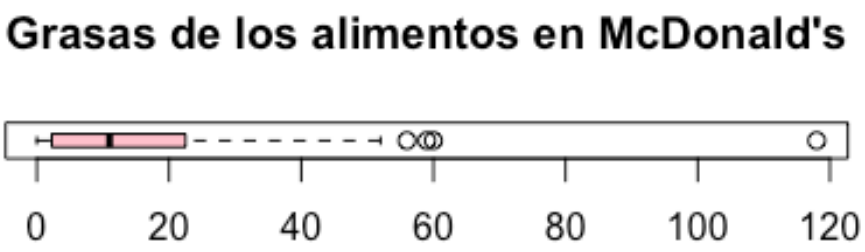


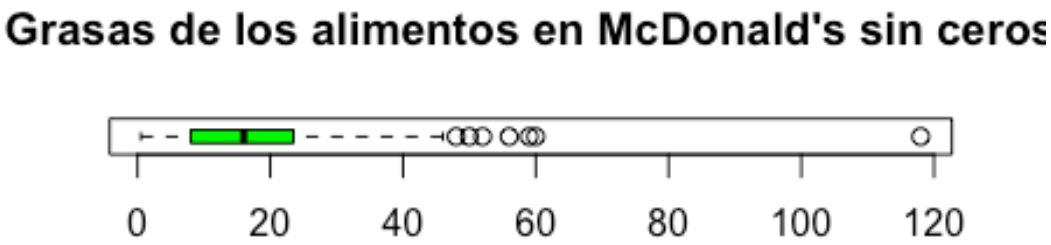
4. Detecta anomalías y corrige tu base de datos (datos atípicos, ceros anámalos, etc).

```
# Quitaremos esos datos que son igual a 0, estos pueden incluir bebidas
dieteticas o agua
fatWoZeros = fat[fat > 0]
bcWoZeros = boxcox((fatWoZeros+1)~1)
```

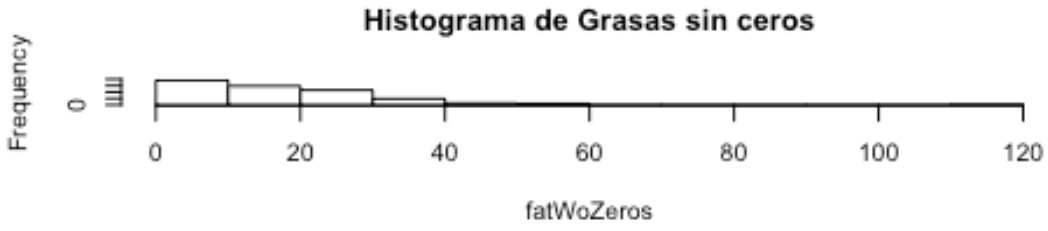


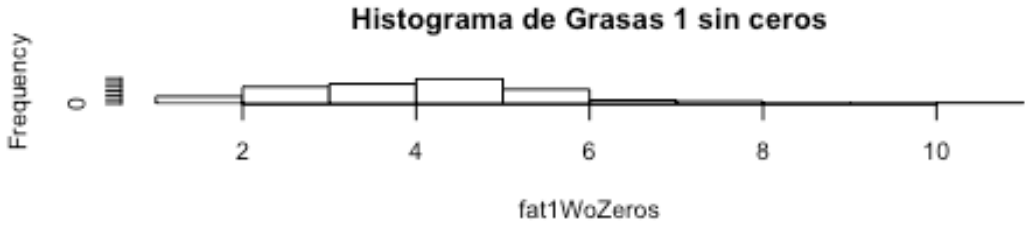

```
lambdaWoZeros = bcWoZeros$x[which.max(bcWoZeros$y)]

cat("Lambda sin ceros:", lambdaWoZeros, "\n")

## Lambda sin ceros: 0.3434343

par(mfrow = c(2, 1))
boxplot(fat, horizontal = TRUE, col = "pink", main = "Grasas de los alimentos
en McDonald's")
boxplot(fatWoZeros, horizontal = TRUE, col = "green", main = "Grasas de los
alimentos en McDonald's sin ceros")
```

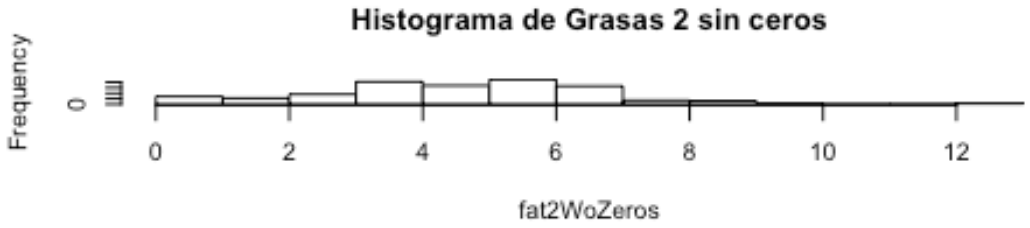





```r
fat1WoZeros = sqrt(fatWoZeros + 1)
fat2WoZeros = (((fatWoZeros + 1) ^ lambdaWoZeros) - 1) / lambdaWoZeros

par(mfrow = c(3, 1))
hist(fatWoZeros, col = 0, main = "Histograma de Grasas sin ceros")
hist(fat1WoZeros, col = 0, main = "Histograma de Grasas 1 sin ceros")
hist(fat2WoZeros, col = 0, main = "Histograma de Grasas 2 sin ceros")
```



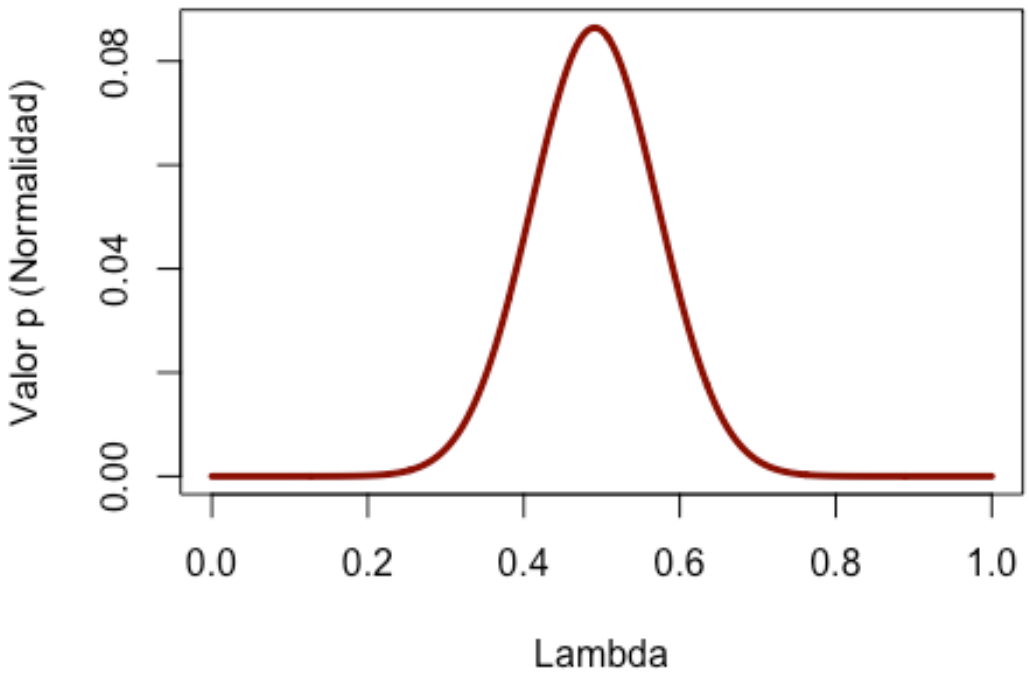




##5. Utiliza la transformación de Yeo Johnson y encuentra el valor de lambda que maximiza el valor p de la prueba de normalidad que hayas utilizado (Anderson-Darling o Jarque Bera).

```
fatYeo = yeo.johnson(fatWoZeros, lambda = lambda)

lp = seq(0, 1, 0.001)
nlp = length(lp)
n = length(fatWoZeros)
D3 = matrix(as.numeric(NA), ncol = 2, nrow = nlp)
d = NA
for (i in 1:nlp) {
  d = yeo.johnson(fatWoZeros, lambda = lp[i])
  p = ad.test(d)
  D3[i, ] = c(lp[i], p$p.value)
}

N = as.data.frame(D3)
plot(
  N$V1,
  N$V2,
  type = "l",
  col = "darkred",
```

```
  lwd = 3 ,
  xlab = "Lambda",
  ylab = "Valor p (Normalidad)"
)
```




```
# El valor de lambda que maximisa valor-p
G = data.frame(subset(N,N$V2==max(N$V2)))
lambdaYeo = G$V1
cat("Lambda de Yeo:", lambdaYeo, "\n")

## Lambda de Yeo: 0.491

fat3 = (((fatYeo + 1)^lambdaYeo) - 1) / lambdaYeo
```

6. Escribe la ecuación del modelo encontrado.

$$fat3 = \frac{(x+1)^{0.49} - 1}{0.49}$$

7. Analiza la normalidad de las transformaciones obtenidas con los datos originales. Utiliza como argumento de normalidad:

- Compara las medidas: Mínimo, máximo, media, mediana, cuartil 1 y cuartil 3, sesgo y curtosis.

- Obten el histograma de los 2 modelos obtenidos (exacto y aproximado) y los datos originales.
- Realiza la prueba de normalidad de Anderson-Darling para los datos transformados y los originales.

```
# Fat
D = ad.test(fat)
summary(fat)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   2.375  11.000  14.165  22.250 118.000
```

```
cat("Curtosis:", kurtosis(fat), "\n")
```

```
## Curtosis: 10.35171
```

```
cat("Sesgo:", skewness(fat), "\n")
```

```
## Sesgo: 2.128023
```

```
cat("P-value:", D$p.value, "\n")
```

```
## P-value: 1.46366e-16
```

```
# Fat 3
D3 = ad.test(fat3)
summary(fat3)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.3922  2.0460  2.6616  2.4741  3.0298  4.7896
```

```
cat("Curtosis:", kurtosis(fat3), "\n")
```

```
## Curtosis: 0.2935619
```

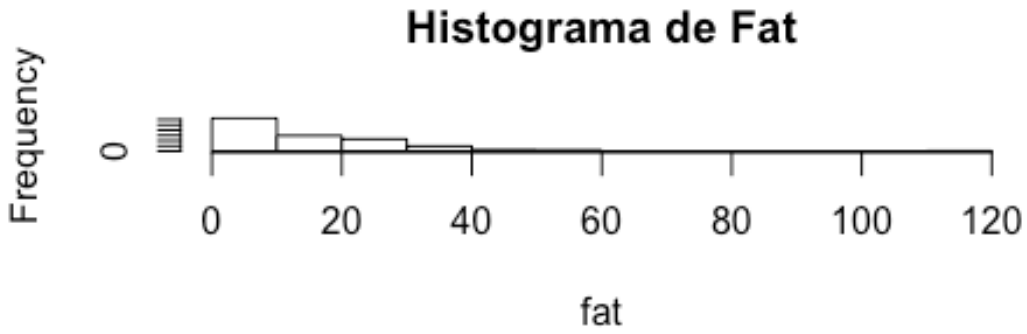
```
cat("Sesgo:", skewness(fat3), "\n")
```

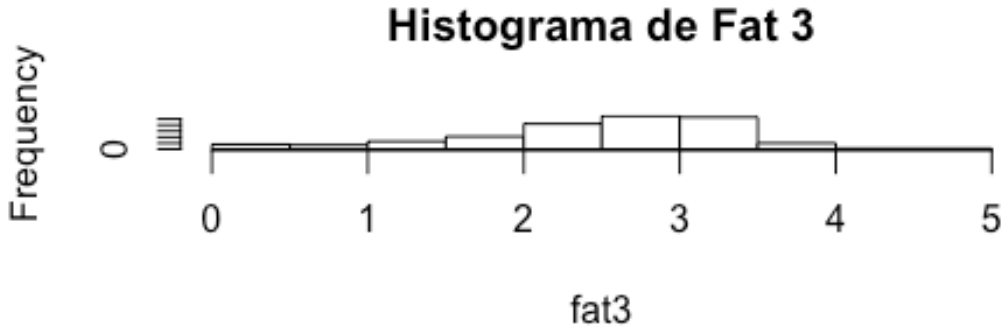
```
## Sesgo: -0.662302
```

```
cat("P-value:", D3$p.value, "\n")
```

```
## P-value: 7.124648e-08
```

```
# Histogramas de las variables
par(mfrow = c(2, 1))
hist(fat, col = 0, main = "Histograma de Fat")
hist(fat3, col = 0, main = "Histograma de Fat 3")
```





8. Define la mejor transformación de los datos de acuerdo a las características de los modelos que encontraste.

Ambos modelos funcionan bien para cumplir con el supuesto de normalidad, no obstante, tomando en cuenta que ninguna de las variables tiene valores negativos, el modelo de BoxCox se ajusta mejor para transformar los datos que el de Yeo Johsnon de acuerdo a sus características.

9. Concluye sobre las ventajas y desventajes de los modelos de Box Cox y de Yeo Johnson.

Uno de los beneficios de utilizar la stransformaciones de Box Cox y Yeo Johnson es que los datos son más sencillos de utilizar ya que los datos son más fáciles de comparar. Además, utilizar este tipos de transformaciones ayuda a mejorar la calidad de los datos y evitar impurezas en los mismos como lo son datos nulos y duplicados.

Algunas desventajas que tienen estas transformaciones es que BoxCox no funciona adecuadamente con datos negativos ni con datos discretos lo cual limita las capacidades de uno para utilizar estas herramientas.

10. Analiza las diferencias entre la transformación y el escalamiento de los datos:

```
- Escribe al menos 3 diferencias entre lo que es la transformación y el
escalamiento de los datos
- Indica cuándo es necesario utilizar cada uno
```

Una de las diferencias entre la transformación y el escalamiento de los datos es que la transformación tiene como objetivo que se cumplan supuestos como lo es la normalidad y el escalamiento busca ajustar los valores a un determinado rango.

Otra diferencia es que la transformación puede cambiar la distribución de datos mientrsas que el escalamiento sólo puede (como dice su nombre) cambiar la escala. Finalmente, los métodos para realizar el escalamiento y las trasnformaciones son distintos. Para las tranformaciones se puede utilizar logaritmos, explonenciales y raíces cuadradas pero para el escalimiento se utilizan operaciones como la división para realizar el ajuste.

La transofmración se debe utilizar cuando se necesita que los datos se ajusten a los supuestos de un modelo. El escalamiento se debe utilizar primordialmente cuando se quiere realizar una comparación de datos que están en diferentes escalas y que, por su rango, no son comparables.