Actividad Integradora 1

José Carlos Sánchez Gómez

2024-08-20

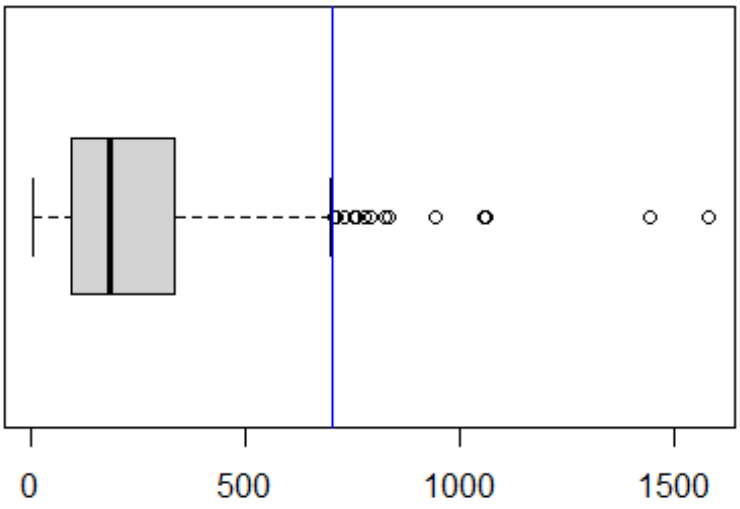
Actividad Integradora 1

```
data =
read.csv("C:\\Users\\jcsg6\\Documentos\\Uni\\SeptimoSemestre\\Estadistica
\\food_data_g.csv")
calorias = data$Caloric.Value
```

Punto 1

Interpretación de datos atipicos y normalidad

```
q1 = quantile(calorias, 0.25)
q3 = quantile(calorias, 0.75)
q2 = quantile(calorias, 0.50)
ri = q3 - q1
# ri2 = IQR(calorias) comprobar que si da lo mismo

boxplot(calorias, horizontal = TRUE)
abline(v = q3 + 1.5 * ri, col="blue")  #linea vertical en el límite de
los datos atípicos o extremos
```




```
cat("Cuartil 1: ", q1, " ")
```

```
## Cuartil 1:  94.5
```

```
cat("Cuartil 2: ", q2, " ")
```

```
## Cuartil 2:  186
```

```
cat("Cuartil 3: ", q3, " ")
```

```
## Cuartil 3:  337
```

```
cat("Rango intercuartilico: ", ri, "\n")
```

```
## Rango intercuartilico:  242.5
```

```
sd = sd(calorias)
cat("Desviación estandar: ", sd, "\n")
```

```
## Desviación estandar:  199.2356
```

```
summary(calorias)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     3.0    94.5   186.0   237.4   337.0  1578.0
```

```
# Cota de 1.5 rangos intercuartilicos
cota_inter_inf = q1 - 1.5 * ri
cota_inter_sup = q1 + 1.5 * ri
```

```r
calorias_rango_1.5 = calorias[calorias < cota_inter_inf | calorias > cota_inter_sup]
cat("Hay", length(calorias_rango_1.5), "datos atípicos en la cota de 1.5 rangos intercuartilicos")
```

```
## Hay 65 datos atípicos en la cota de 1.5 rangos intercuartilicos
```

```r
# Cota de 3 desviaciones estandar
u = mean(calorias)
cota_sd_inf = u - 3 * sd
cota_sd_sup = u + 3 * sd
calorias_rango_sd = calorias[calorias < cota_sd_inf | calorias > cota_sd_sup]

cat("Hay", length(calorias_rango_sd), "datos atípicos en la cota de 3 desviaciones estandar alrededor de la media")
```

```
## Hay 6 datos atípicos en la cota de 3 desviaciones estandar alrededor de la media
```

```r
# Cota de 3 rangos intercuartilicos
cota_inter_inf_3 = q1 - 3 * ri
cota_inter_sup_3 = q1 + 3 * ri
calorias_rango_3 = calorias[calorias < cota_inter_inf_3 | calorias > cota_inter_sup_3]
cat("Hay", length(calorias_rango_3), "datos atípicos en la cota de 3 rangos intercuartilicos")
```

```
## Hay 8 datos atípicos en la cota de 3 rangos intercuartilicos
```

Viendo el resumen de los datos de calorias podemos inferir que no parece tener una distribución normal, puesto que el valor de su desviación estandar es muy elevado, su media esta alejada de su mediana, y los valores atipicos son varios con respecto al tamaño de los datos.

```r
library(nortest)
library(tseries)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

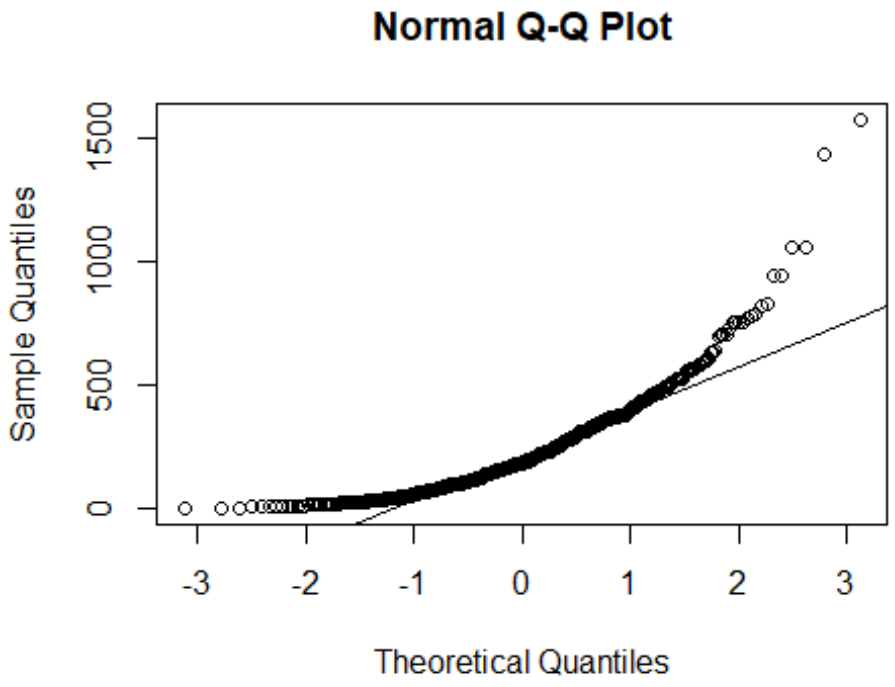
```r
ad_test = ad.test(calorias)
ad_test
```

```
##
##  Anderson-Darling normality test
##
## data:  calorias
## A = 15.326, p-value < 2.2e-16
```

```r
jb_test = jarque.bera.test(calorias)
jb_test
```

```
##
##  Jarque Bera Test
##
## data:  calorias
## X-squared = 1388.9, df = 2, p-value < 2.2e-16
```

```r
qqnorm(calorias)
qqline(calorias)
```




```r
library(e1071)
sesgo = skewness(calorias)
curtosis = kurtosis(calorias)
cat("Sesgo de los datos de calorias: ", sesgo, "\n")
```

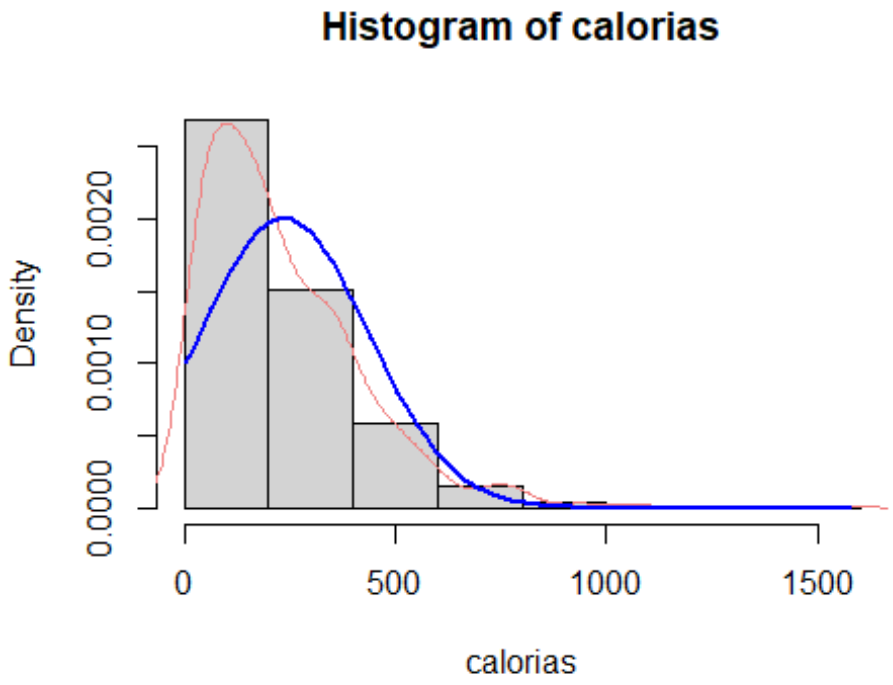
```
## Sesgo de los datos de calorias:  1.917503
```

```r
cat("Curtosis de los datos de calorias: ", curtosis, "\n")
```

```
## Curtosis de los datos de calorias:  6.725447
```

```r
media = mean(calorias)
mediana = median(calorias)
rango_medio = (max(calorias) + min(calorias)) / 2
cat("Media de los valores: ", media, "\n")
```

```
## Media de los valores:  237.3593

cat("Mediana de los valores: ", mediana, "\n")

## Mediana de los valores:  186

cat("Rango medio de los valores: ", rango_medio, "\n")

## Rango medio de los valores:  790.5

hist(calorias, freq = FALSE)
lines(density(calorias), col="lightcoral")
curve(dnorm(x, mean = mean(calorias), sd = sd(calorias)), from =
min(calorias), to = max(calorias),
      add = TRUE, col = "blue", lwd = 2)
```

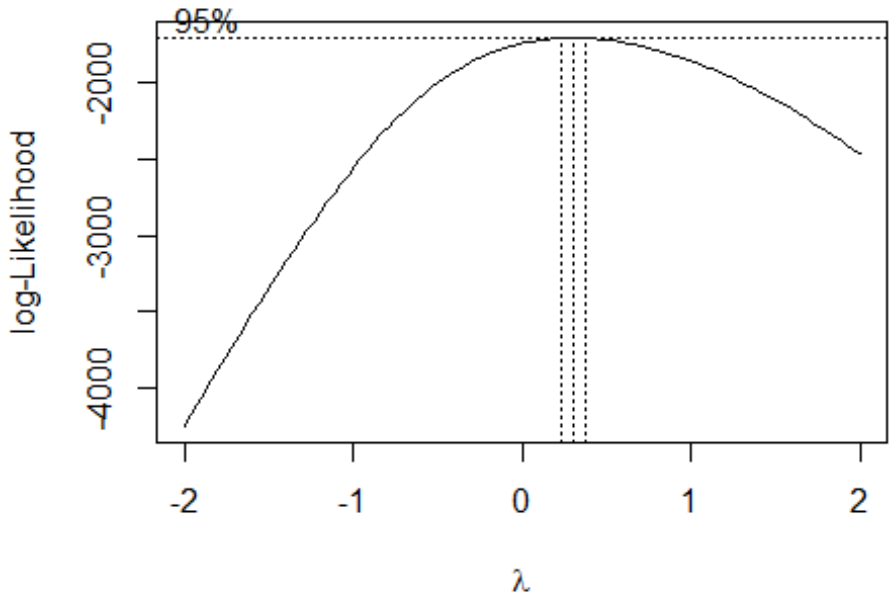


Viendo el histograma de nuestros datos podemos concluir sencillamente que no se sigue una distribución normal. Apoyandonos en los datos estadisticos podemos ver que tiene un sesgo a la derecha muy grande, el cual tiene un valor de casi 2 (1.9), además de que tiene una curtosis muy elevada lo cual indica que tiene un pico muy elevado; aunado a esto las pruebas de normalidad de Anderson-Darling y Jarque-Bera nos proporcionan valores muy bajos de p, el cual es otro indicador de que nuestros datos no tienen normalidad. Esta información nos ayuda a concluir que efectivamente se rechaza la h0 la cual dice que los datos siguen una distribución normal, por consiguiente se acepta la h1.

Punto 2

Transformar a normalidad

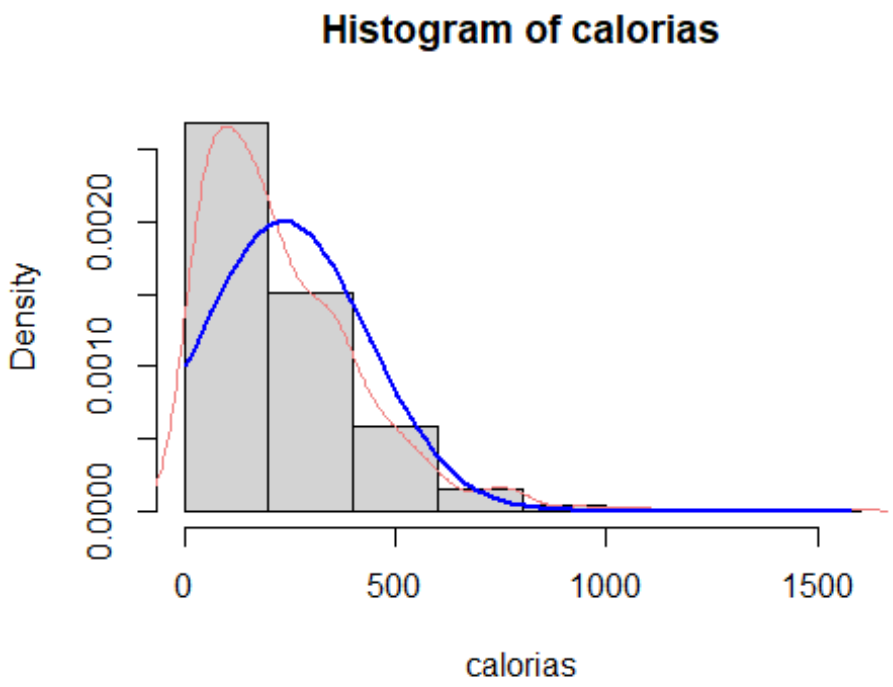
```
# Obteniendo lambda para boxcox
library(MASS)
bc = boxcox((calorias) ~ 1)
```



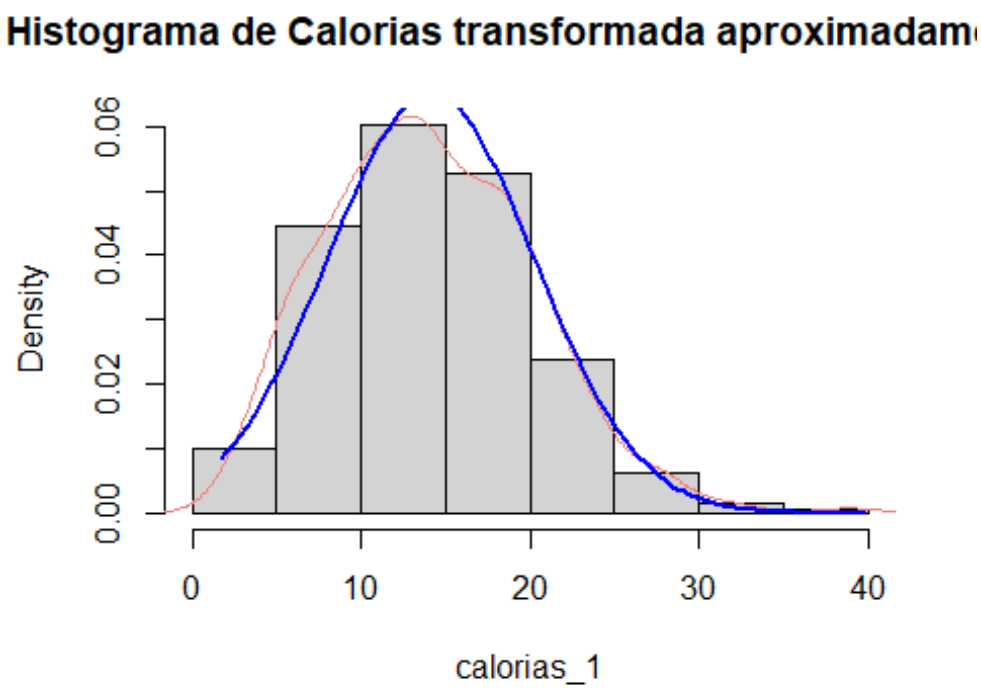

```
l = bc$x[which.max(bc$y)]
l

## [1] 0.3030303
```

Dado que nuestra lambda es de 0.3030, la función recomendada para la aproximación es $\sqrt{(x)}$, y para la exacta será $\frac{x^{\lambda}-1}{\lambda}$, que quedaria como $\frac{x^{(0.3030)}-1}{0.3030}$

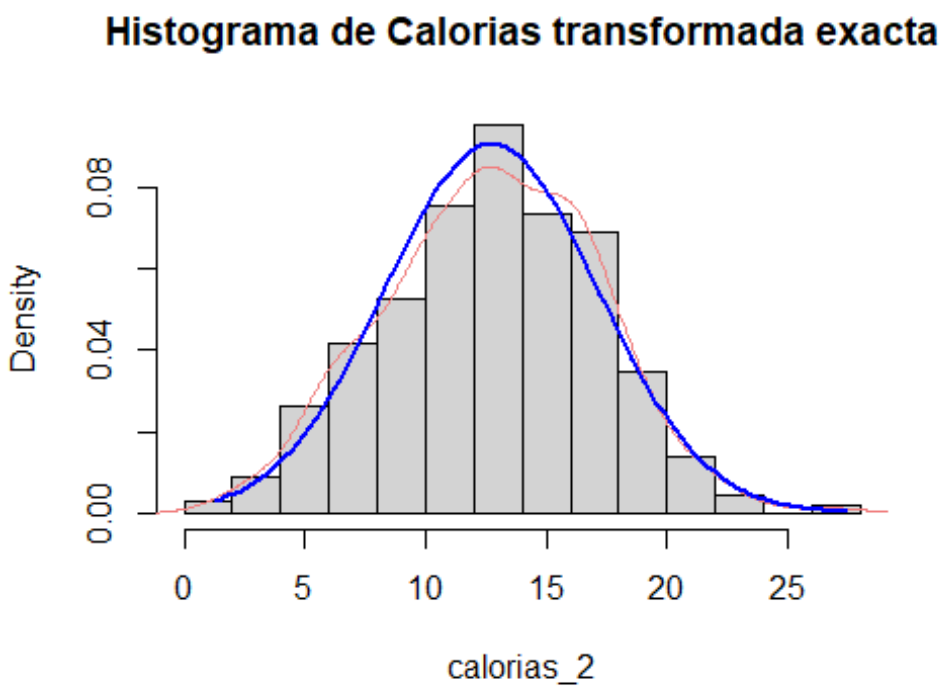
```
# Histograma con la información original
hist(calorias, freq = FALSE)
lines(density(calorias), col="lightcoral")
curve(dnorm(x, mean = mean(calorias), sd = sd(calorias)), from =
min(calorias), to = max(calorias),
      add = TRUE, col = "blue", lwd = 2)
```




```r
# Histograma con boxcox aproximado
calorias_1 = sqrt(calorias)
hist(calorias_1, freq = FALSE, main="Histograma de Calorias transformada
aproximadamente")
lines(density(calorias_1), col="lightcoral")
curve(dnorm(x, mean = mean(calorias_1), sd = sd(calorias_1)), from =
min(calorias_1), to = max(calorias_1),
      add = TRUE, col = "blue", lwd = 2)
```




```r
# Histograma con boxcox exacto
calorias_2 = ((calorias)^l - 1) / l
hist(calorias_2, freq = FALSE, main="Histograma de Calorias transformada
exacta")
lines(density(calorias_2), col="lightcoral")
curve(dnorm(x, mean = mean(calorias_2), sd = sd(calorias_2)), from =
min(calorias_2), to = max(calorias_2),
      add = TRUE, col = "blue", lwd = 2)
```

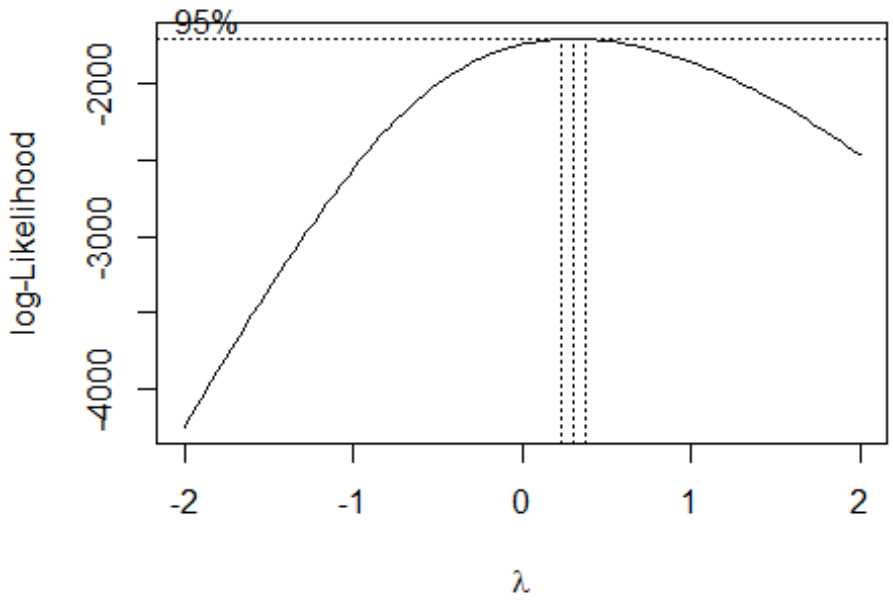



```
library(e1071)
library(nortest)
library(tseries)

# resumen de los datos normales
calorias_aproximado_summary = summary(calorias_1)
calorias_exacto_summary = summary(calorias_2)
p_value_normal = ad.test(calorias)$p.value
p_value_aproximado = ad.test(calorias_1)$p.value
p_value_exacto = ad.test(calorias_2)$p.value

data.frame(
  Estadistico = c(names(summary(calorias)), "Curtosis", "Sesgo", "P-
Value_ad", "P_Value_jb"),
  Orginal = c(as.numeric(summary(calorias)), curtosis, sesgo,
p_value_normal, jarque.bera.test(calorias)$p.value),
  "Modelo Aproximado" = c(as.numeric(calorias_aproximado_summary),
kurtosis(calorias_1), skewness(calorias_1), p_value_aproximado,
jarque.bera.test(calorias_1)$p.value),
  "Modelo Exacto" = c(as.numeric(calorias_exacto_summary),
kurtosis(calorias_2), skewness(calorias_2), p_value_exacto,
jarque.bera.test(calorias_2)$p.value)
)

##    Estadistico      Orginal Modelo.Aproximado Modelo.Exacto
## 1         Min. 3.000000e+00      1.732051e+00    1.30358470
## 2      1st Qu. 9.450000e+01      9.721077e+00    9.79568908
```

```
## 3       Median 1.860000e+02      1.363818e+01   12.77848538
## 4         Mean 2.373593e+02      1.413279e+01   12.73576346
## 5      3rd Qu. 3.370000e+02      1.835754e+01   15.95138551
## 6         Max. 1.578000e+03      3.972405e+01   27.43528700
## 7     Curtosis 6.725447e+00      3.416900e-01   -0.18683613
## 8        Sesgo 1.917503e+00      4.763660e-01   -0.02223906
## 9   P-Value_ad 3.700000e-24      2.964427e-03    0.13284227
## 10  P_Value_jb 0.000000e+00      6.696789e-06    0.68329473
```

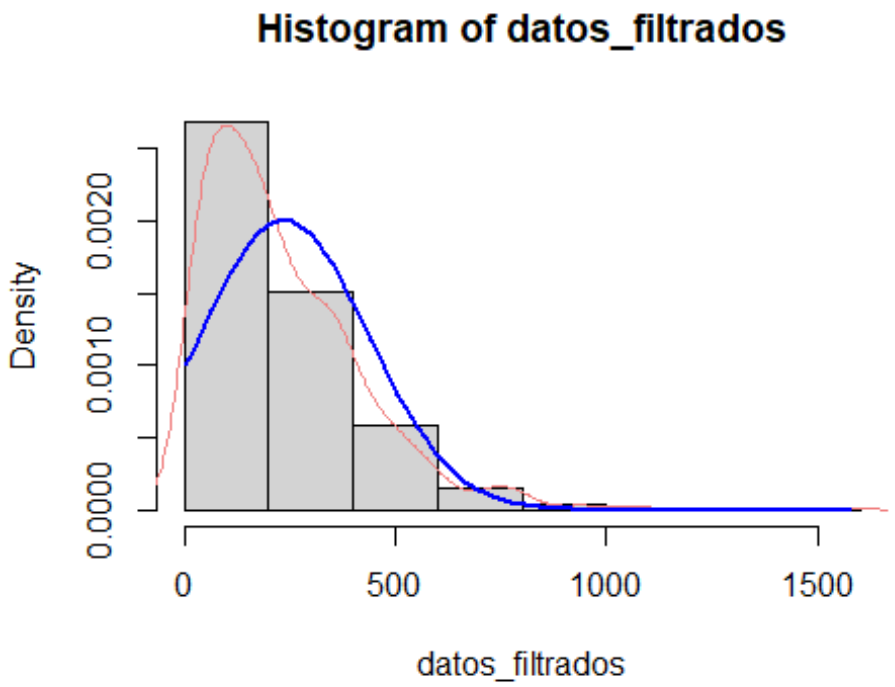
A primera vista podemos observar que los datos proporcionados por la transformación del modelo exacto es mucho mejor a la aproximada o a la normal. Su media es casi la misma a su mediana, exceptuando por unas decimas. Tiene un valor de curtosis y sesgo casi nulo, además de que sus valores de pruebas de normalidad son buenas. Viendo los datos podemos concluir que la transformación exacta es mejor. Sin emabrgo, hay algunos datos atípicos en los extremos de los datos que serán limpiados para tener una mejor distribución.

```
library(MASS)
datos_filtrados = calorias[calorias > cota_inter_inf_3 | calorias <
cota_inter_sup_3]
bc = boxcox((datos_filtrados) ~ 1)
```

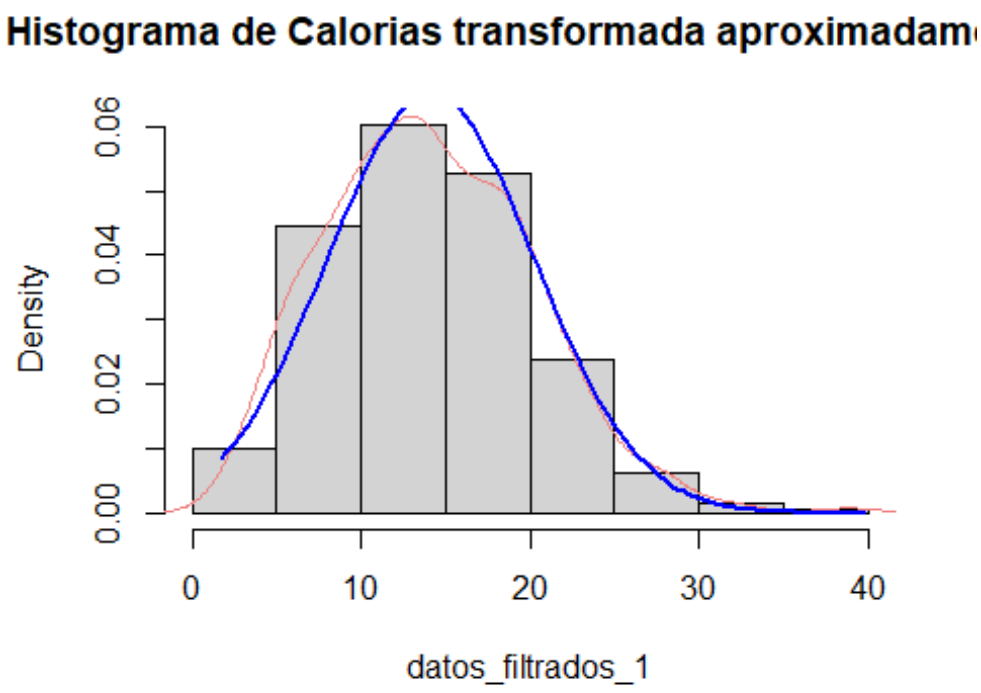



```
l = bc$x[which.max(bc$y)]

# Histograma con la información original
hist(datos_filtrados, freq = FALSE)
lines(density(datos_filtrados), col="lightcoral")
```

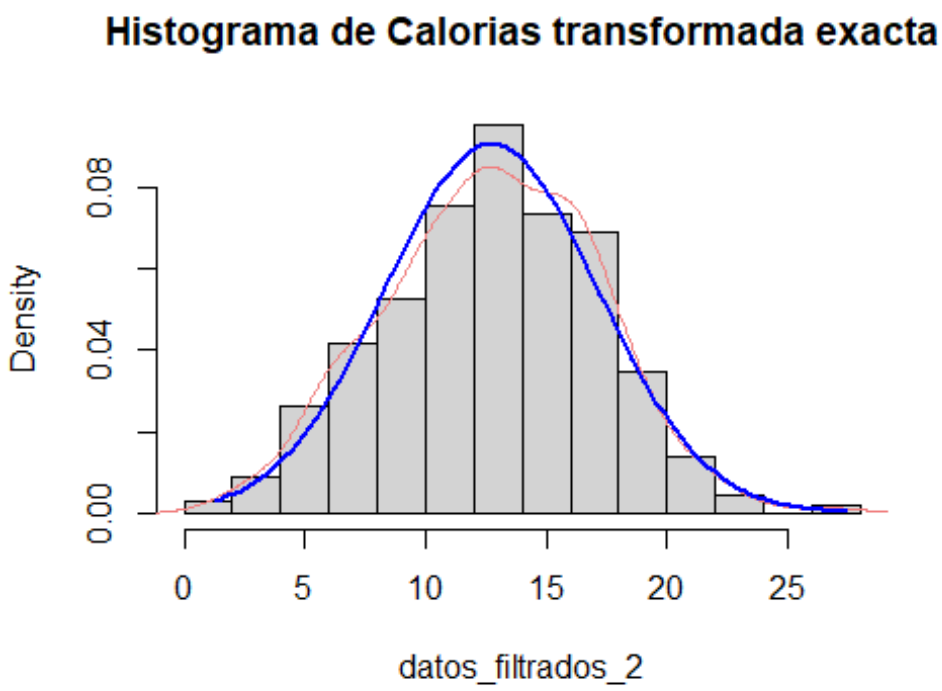
```r
curve(dnorm(x, mean = mean(datos_filtrados), sd = sd(datos_filtrados)),
from = min(datos_filtrados), to = max(datos_filtrados),
      add = TRUE, col = "blue", lwd = 2)
```




```r
# Histograma con boxcox aproximado
datos_filtrados_1 = sqrt(datos_filtrados)
hist(datos_filtrados_1, freq = FALSE, main="Histograma de Calorias
transformada aproximadamente")
lines(density(datos_filtrados_1), col="lightcoral")
curve(dnorm(x, mean = mean(datos_filtrados_1), sd =
sd(datos_filtrados_1)), from = min(datos_filtrados_1), to =
max(datos_filtrados_1),
      add = TRUE, col = "blue", lwd = 2)
```




```r
# Histograma con boxcox exacto
datos_filtrados_2 = ((calorias)^l - 1) / l
hist(datos_filtrados_2, freq = FALSE, main="Histograma de Calorias
transformada exacta")
lines(density(datos_filtrados_2), col="lightcoral")
curve(dnorm(x, mean = mean(datos_filtrados_2), sd =
sd(datos_filtrados_2)), from = min(datos_filtrados_2), to =
max(datos_filtrados_2),
      add = TRUE, col = "blue", lwd = 2)
```




```
library(e1071)
library(nortest)
library(tseries)

# resumen de los datos normales
calorias_aproximado_summary = summary(datos_filtrados_1)
calorias_exacto_summary = summary(datos_filtrados_2)
p_value_normal = ad.test(datos_filtrados)$p.value
p_value_aproximado = ad.test(datos_filtrados_1)$p.value
p_value_exacto = ad.test(datos_filtrados_2)$p.value

data.frame(
  Estadistico = c(names(summary(calorias)), "Curtosis", "Sesgo", "P-
Value_ad", "P_Value_jb"),
  Orginal = c(as.numeric(summary(datos_filtrados)), curtosis, sesgo,
p_value_normal, jarque.bera.test(datos_filtrados)$p.value),
  "Modelo Aproximado" = c(as.numeric(calorias_aproximado_summary),
kurtosis(datos_filtrados_1), skewness(datos_filtrados_1),
p_value_aproximado, jarque.bera.test(datos_filtrados_1)$p.value),
  "Modelo Exacto" = c(as.numeric(calorias_exacto_summary),
kurtosis(datos_filtrados_2), skewness(datos_filtrados_2), p_value_exacto,
jarque.bera.test(datos_filtrados_2)$p.value)
)

##    Estadistico      Orginal Modelo.Aproximado Modelo.Exacto
## 1         Min. 3.000000e+00      1.732051e+00    1.30358470
## 2      1st Qu. 9.450000e+01      9.721077e+00    9.79568908
```

```
## 3       Median 1.860000e+02      1.363818e+01   12.77848538
## 4         Mean 2.373593e+02      1.413279e+01   12.73576346
## 5      3rd Qu. 3.370000e+02      1.835754e+01   15.95138551
## 6         Max. 1.578000e+03      3.972405e+01   27.43528700
## 7     Curtosis 6.725447e+00      3.416900e-01   -0.18683613
## 8        Sesgo 1.917503e+00      4.763660e-01   -0.02223906
## 9   P-Value_ad 3.700000e-24      2.964427e-03    0.13284227
## 10  P_Value_jb 0.000000e+00      6.696789e-06    0.68329473
```

Aún así transformando los datos, la transformación exacta es la mejor transformación de todas. Sigue teniendo un sesgo y una curtosis casi nula, y sus valores de p siguen siendo muy buenos. Podemos concluir que esta transformación es la mejor de todas.

```
qqnorm(datos_filtrados_2)
qqline(datos_filtrados_2)
```