Actividad Integradora 1

Juan Bernal

2024-08-20

Trabaja con el set de datos Nutrición Mundial, que contiene diversas características del alimentos que se consumen en el mundo. Pueden encontrar más información sobre ella en: Utsav Dey. (2024). Food Nutrition Dataset [Data set]. Kaggle. https://doi.org/10.34740/KAGGLE/DSV/8820139Links to an external site.. El resumen de su contenido es:

The Comprehensive Nutritional Food Database provides detailed nutritional information for a wide range of food items commonly consumed around the world. This dataset aims to support dietary planning, nutritional analysis, and educational purposes by providing extensive data on the macro and micronutrient content of foods.

Punto 1. Análisis descriptivo de la variable

Analiza una de las siguientes variables en cuanto a sus datos atípicos y normalidad. La variable que te corresponde analizar te será asignada por tu profesora al inicio de la actividad:

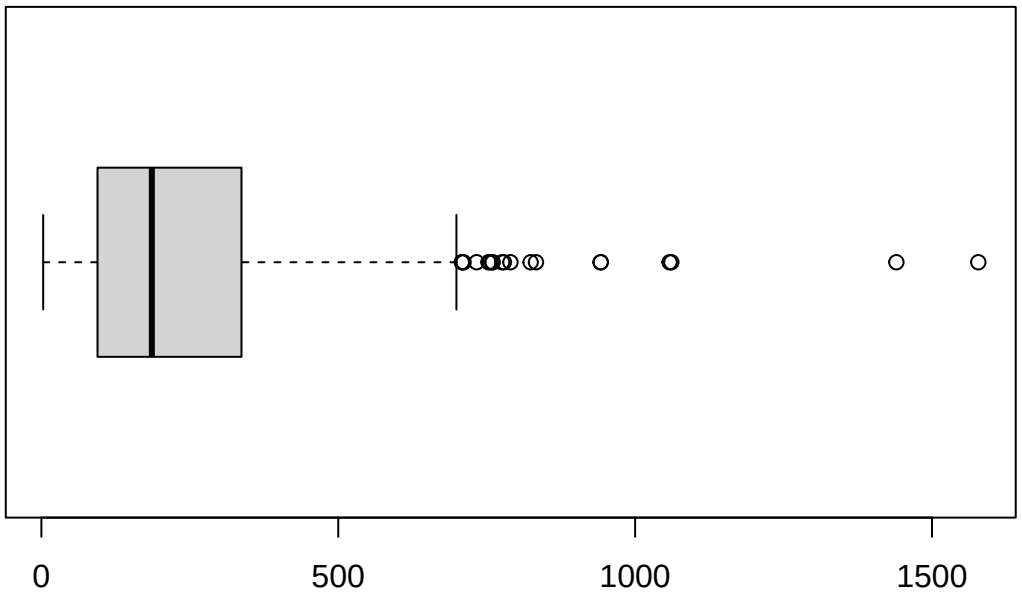
`Calorías`

```
data=read.csv("food_data_g.csv") #leer la base de datos
dataset = data$Caloric.Value
```

Para analizar datos atípicos se te sugiere:

`Graficar el diagrama de caja y bigote`

```
boxplot(dataset, horizontal = TRUE)
```



En la gráfica se observa un gran número de datos atípicos y extremos, además de un gran sesgo hacia la derecha.

Calcula las principales medidas que te ayuden a identificar datos atípicos (utilizar summary te puede al

```
summary(dataset)
```

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

```
print('Desviación estándar')
```

```
## [1] "Desviación estándar"
```

```
sd(dataset)
```

```
## [1] 199.2356
```

```
print('Rango intercuartílico')
```

```
## [1] "Rango intercuartílico"
```

```
quantile(dataset,c(0.75)) - quantile(dataset,c(0.25))
```

```
##   75%
## 242.5
```

Identifica la cota de 1.5 rangos intercuartílicos para datos atípicos, ¿hay datos atípicos de acuerdo c

```
q15s = quantile(dataset,c(0.75)) + 1.5*(quantile(dataset,c(0.75)) - quantile(dataset,c(0.25)))
q15i = quantile(dataset,c(0.25)) - 1.5*(quantile(dataset,c(0.75)) - quantile(dataset,c(0.25)))
cat("Cota superior de 1.5 rangos intercuartílicos de las Calorías: ", q15s,'\n')
```

```
## Cota superior de 1.5 rangos intercuartílicos de las Calorías:  700.75
```

```
cat("Cota inferior de 1.5 rangos intercuartílicos de las Calorías", q15i,'\n')
```

```
## Cota inferior de 1.5 rangos intercuartílicos de las Calorías -269.25
```

```
print("Los datos atípicos son")
```

```
## [1] "Los datos atípicos son"
```

```
dataset[dataset > q15s]
```

```
##  [1] 1058  708  779 1440 1578  776  942 1061  833  790  942  733  709  755  760
## [16]  824  711  759  753
```

```
dataset[dataset < q15i]
```

```
## integer(0)
```

Identifica la cota de 3 desviaciones estándar alrededor de la media, ¿hay datos atípicos de acuerdo con

```
ds3s = 3*sd(dataset)+mean(dataset)
ds3i = 3*sd(dataset)-mean(dataset)
cat("Cota superior de 3 desviaciones estándar alrededor de la media en las Calorías: ", ds3s, '\n')
```

```
## Cota superior de 3 desviaciones estándar alrededor de la media en las Calorías:  835.0661
```

```
cat("Cota inferior de 3 desviaciones estándar alrededor de la media en las Calorías", ds3i,'\n')
```

```
## Cota inferior de 3 desviaciones estándar alrededor de la media en las Calorías 360.3474
```

```
print("Los datos atípicos son")
```

```
## [1] "Los datos atípicos son"
```

```
dataset[dataset>ds3s]
```

```
## [1] 1058 1440 1578  942 1061  942
```

```
dataset[dataset<ds3i]
```

```
##   [1]  51 215  49  30  30  19 116 113  71  19  21  98 128 100  75 106 136 103
##  [19] 111  90 310 314 315  75  90  98 100 316  80 159  64  35  56 100  88 101
##  [37]  88  86  49  46  48  94  37  27  65 309 223 231 323 110 298 225 285 350
##  [55] 189 155 260 224 100 161 158 153  67 344  38 160 107  86  94 185 230 328
##  [73] 222  46 209 271 223 304 212 213 103 102 285  84 340 300 145 336 180 228
##  [91] 283 331 178 347 189 172 352  35  72 315 223 247 274 310 220 156 165  30
## [109]  57 193  93 247 231 167 229 338  60 182 176 255 273 332 227 166 310 316
## [127] 211 316 176  37 166 198 227 139  65 197 268  10 162 287 168  17 126 120
## [145] 234 189 199 320  78 235 148  72  71  31 186  70 186 127 100 149 308 174
## [163] 188 161 135 201  91 203 186 106  40  17 139  98  39 283 167 154  54 127
## [181] 181 188 240 128 281 118 145  11 174 139 181 335  12   8 180  61  86 248
## [199]  99 113 113  95 148 164 125  70 156 185  97 207 158 296 148 250  73 358
## [217] 104 105 214 339  27  79 163 195  70  98 287 310 129 113  48 151 261  36
## [235] 200 259 186 313 124 173 292  57 258 255 251 308 325 313 269 260 285 278
## [253] 332 146 348 135 104 356  27 265 148 275 293 218  58  37 291 189 111  84
## [271]  65 118 196 119  21 139 353 213  30  77 186 218 143  51  45 189 248 114
## [289]  79 148  55  43 231 111 236 192  42 294 276  68 290 186 234  91 279 172
## [307] 174 255  20 125 349 265 350  28 140   3  47 125 184 211 111 278  35  22
## [325] 117  37 104 173  25 173  54 106  41  54 277  30  22 199  22  90 109  42
## [343] 189  78 254  27  37 168  67  60  47  42 355 206  83  31 230 106  82 148
## [361]  79 236   3 186 238 257  87 238  48 168 329 184 162 247 182 310   8 165
## [379] 102 328 110 331 320 128 144  64   7  19 130  11 140  41  89  25  78  41
## [397]   8 344  23 233 129  98 140 140  52   4  71  29 165  26  10 116  28  18
## [415]  39 234 227 134  29 195 139   8  61  49 159  50  33 147
```

Identifica la cota de 3 rangos intercuartílicos para datos extremos, ¿hay datos extremos de acuerdo con

```
q3s = quantile(dataset,c(0.75)) + 3*(quantile(dataset,c(0.75)) - quantile(dataset,c(0.25)))
q3i = quantile(dataset,c(0.25)) - 3*(quantile(dataset,c(0.75)) - quantile(dataset,c(0.25)))
cat("Cota superior de 3 rangos intercuartílicos de las Calorías: ", q3s,'\n')
```

```
## Cota superior de 3 rangos intercuartílicos de las Calorías:  1064.5
```

```
cat("Cota inferior de 3 rangos intercuartílicos de las Calorías", q3i,'\n')
```

```
## Cota inferior de 3 rangos intercuartílicos de las Calorías -633
```

```
print("Los datos atípicos son")
```

```
## [1] "Los datos atípicos son"
```

```
dataset[dataset > q3s]
```

```
## [1] 1440 1578
```

```
dataset[dataset < q3i]
```

```
## integer(0)
```

Interpreta los resultados obtenidos y argumenta sobre el comportamiento de los datos atípicos y extremos

De la boxplot pudimos observar que hay una gran concentración de datos a la izquierda, lo que causa que un gran número de observaciones sean tratadas como datos atípicos, siendo que están demasiado cerca del quartile 3. Además, la cota de 3 desviaciones estándar alrededor de la media deja fuera a una gran cantidad

de datos debido a la concentración/sesgo antes mencionada. La mejor decisión sería eliminar los datos atípicos a partir de la cota 1.5 rangos intercuartílicos para no perder una gran cantidad de datos importantes.

Para analizar normalidad se te sugiere:

Realiza pruebas de normalidad univariada para la variable (utiliza las pruebas de Anderson-Darling y de

```
library(nortest)
library(moments)
ad.test(dataset)
```

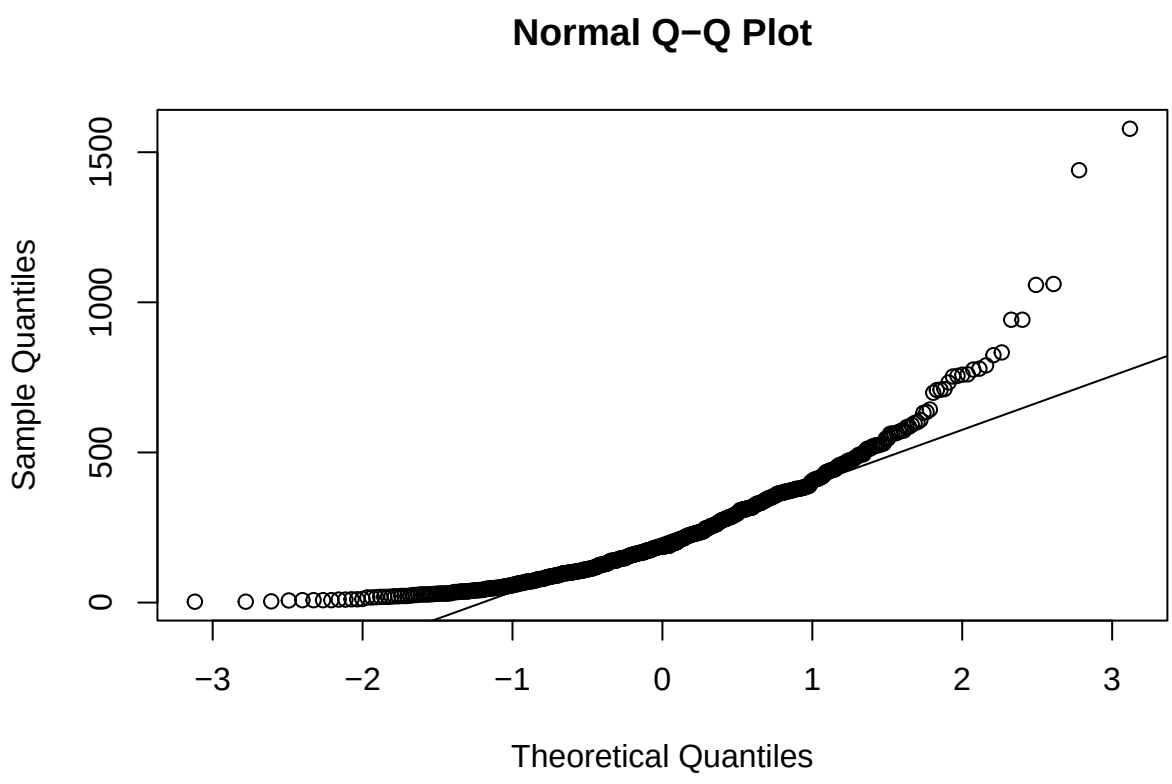
```
##
##  Anderson-Darling normality test
##
## data:  dataset
## A = 15.326, p-value < 2.2e-16
```

```
jarque.test(dataset)
```

```
##
##  Jarque-Bera Normality Test
##
## data:  dataset
## JB = 1388.9, p-value < 2.2e-16
## alternative hypothesis: greater
```

Grafica los datos y su respectivo QQPlot: qqnorm(datos) y qqline(datos)

```
qqnorm(dataset)
qqline(dataset)
```



Calcula el coeficiente de sesgo y el coeficiente de curtosis

```r
skewness(dataset) #Sesgo de Calorías
```

```
## [1] 1.922735
```

```r
kurtosis(dataset) #Curtosis de Calorías
```

```
## [1] 9.760844
```

Compara las medidas de media, mediana y rango medio de cada variable

```r
summary(dataset) #Media y mediana de las Calorías
```

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

```r
print("Rango medio de las Calorías")
```

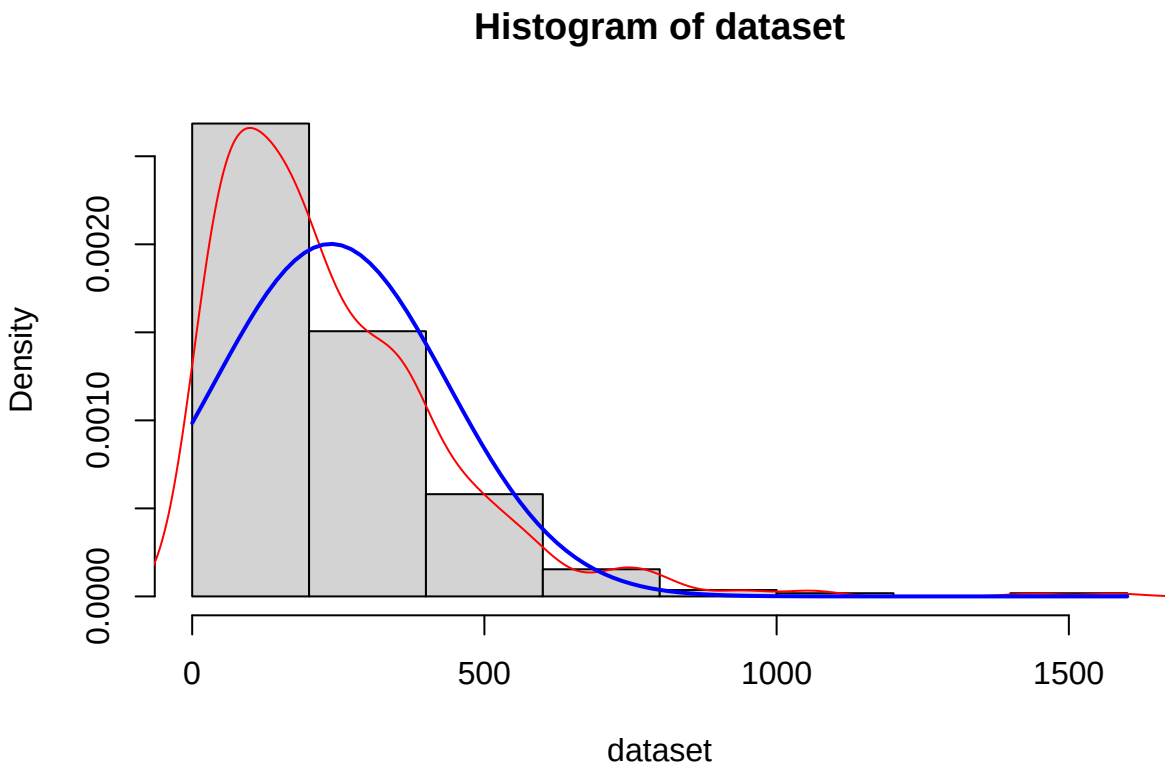
```
## [1] "Rango medio de las Calorías"
```

```r
(max(dataset) - min(dataset))/2 #Rango medio de las Calorías
```

```
## [1] 787.5
```

Realiza el gráfico de densidad empírica y teórica suponiendo normalidad en la variable. Adapta el códig

```r
hist(dataset,freq=FALSE)
lines(density(dataset),col="red")
curve(dnorm(x,mean=mean(dataset),sd=sd(dataset)), add=TRUE, col="blue",lwd=2)
```



Interpreta los gráficos y los resultados obtenidos en cada punto con vías a indicar si hay normalidad d

De las pruebas de normalidad obtenemos que el p-value es igual a 2e-16, por lo que es muchísimo menor al valor estándar de $\alpha = 0.05$ en una prueba de hipótesis, llevándonos a rechazar la hipótesis de que la distribución es normal. De la QQplot y el histograma, aunado al coeficiente de sesgo, logramos observar que la distribución de los datos muestra un sesgo inmenso hacia la derecha. Como última evidencia de que la

distribución de los datos de Calorías no es normal, tomemos los datos de la media, mediana y rango medio. La mediana, es decir, el 50% de los datos se encuentra debajo de 186 calorías, mientras que la media es de 237 calorías. Si tomamos en consideración el rango medio que está 787.5 calorías, nos daremos cuenta de que hay muchos datos por encima de la mediana y la media con demasiada varianza, lo que causa que la distribución empiece a estirarse hacia la derecha, generando un gran sesgo y una gigantesca curtosis.

`Comenta las características encontradas:`

Considera alejamientos de normalidad por simetría, curtosis

La curtosis es de 9.76, lo que indica que la distribución es muy asimétrica, y lo podemos notar en las gráficas, donde la distribución se ve muy estirada debido a los datos atípicos y extremos.

Comenta si hay aparente influencia de los datos atípicos en la normalidad de los datos

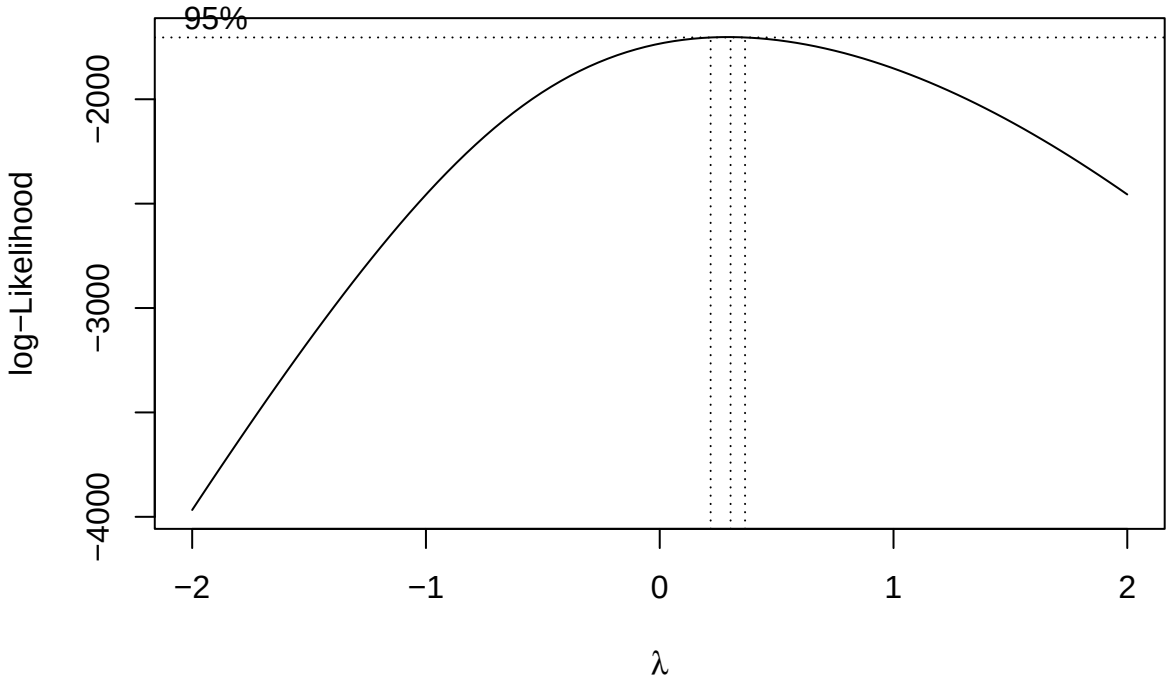
Los datos atípicos estiran la distribución a un sesgo hacia la derecha, por lo que afecta a la simetría de la distribución y la evita de ser normal.

Emite una conclusión sobre la normalidad de los datos. Se debe argumentar en términos de los 3 puntos analizados: las pruebas de normalidad, los gráficos y las medidas.

Punto 2. Transformación a normalidad

`Encuentra la mejor transformación de los datos para lograr normalidad. Puedes hacer uso de la transforma`

```
library(MASS)
bc = boxcox(lm(dataset+1~1))
```




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

```
cat('El mejor valor de lambda encontrado es ',l)
```

```
## El mejor valor de lambda encontrado es  0.3030303
```

`Escribe las ecuaciones de los modelos de transformación encontrados.`

El modelo aproximado queda como $x_1 = \sqrt{x+1}$, y el modelo exacto queda como $x_2 = \frac{(x+1)^{0.3030}-1}{0.3030}$.

```r
x1 = sqrt(dataset+1) # Modelo 1
x2 = ((dataset+1)^l - 1)/l # Modelo 2
```

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

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

```r
library(e1071)
```

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

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

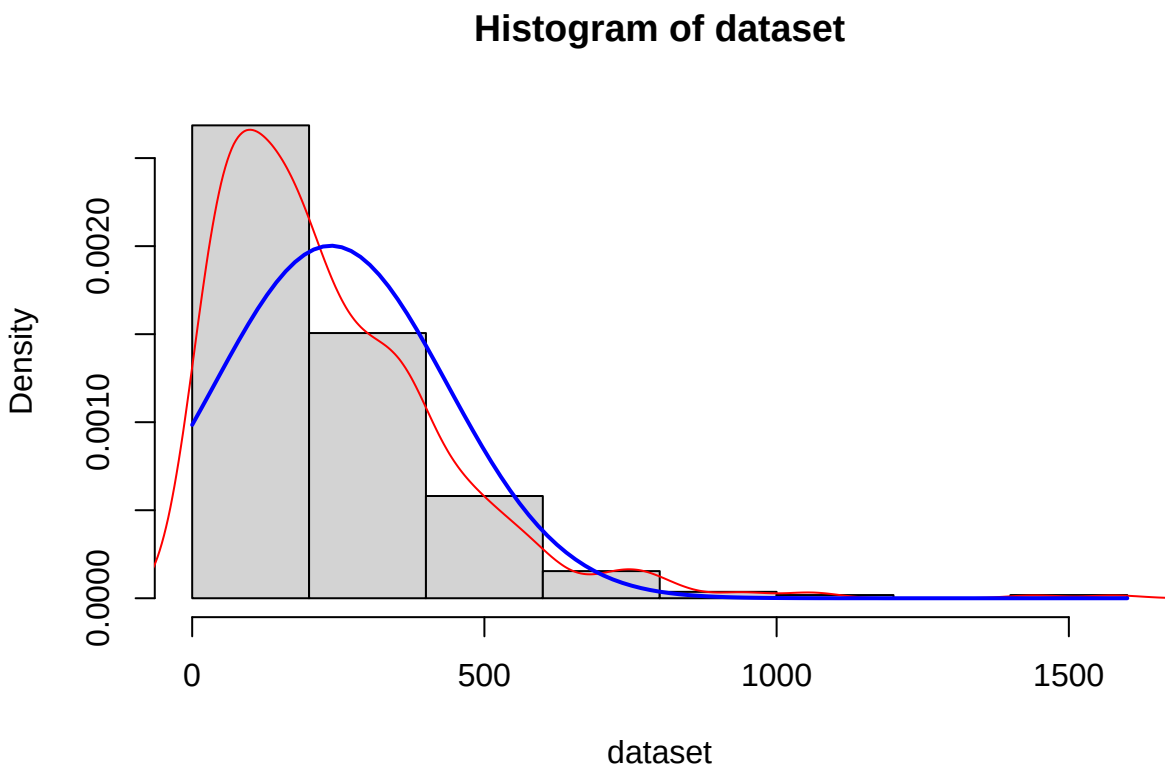
```r
m0=round(c(as.numeric(summary(dataset)),kurtosis(dataset),skewness(dataset)),3)
m1=round(c(as.numeric(summary(x1)),kurtosis(x1),skewness(x1)),3)
m2=round(c(as.numeric(summary(x2)),kurtosis(x2),skewness(x2)),3)

m<-as.data.frame(rbind(m0,m1,m2))
row.names(m)=c("Original","Modelo aproximado","Modelo exacto")
names(m)=c("Minimo","Q1","Mediana","Media","Q3","Máximo","Curtosis","Sesgo")
m
```

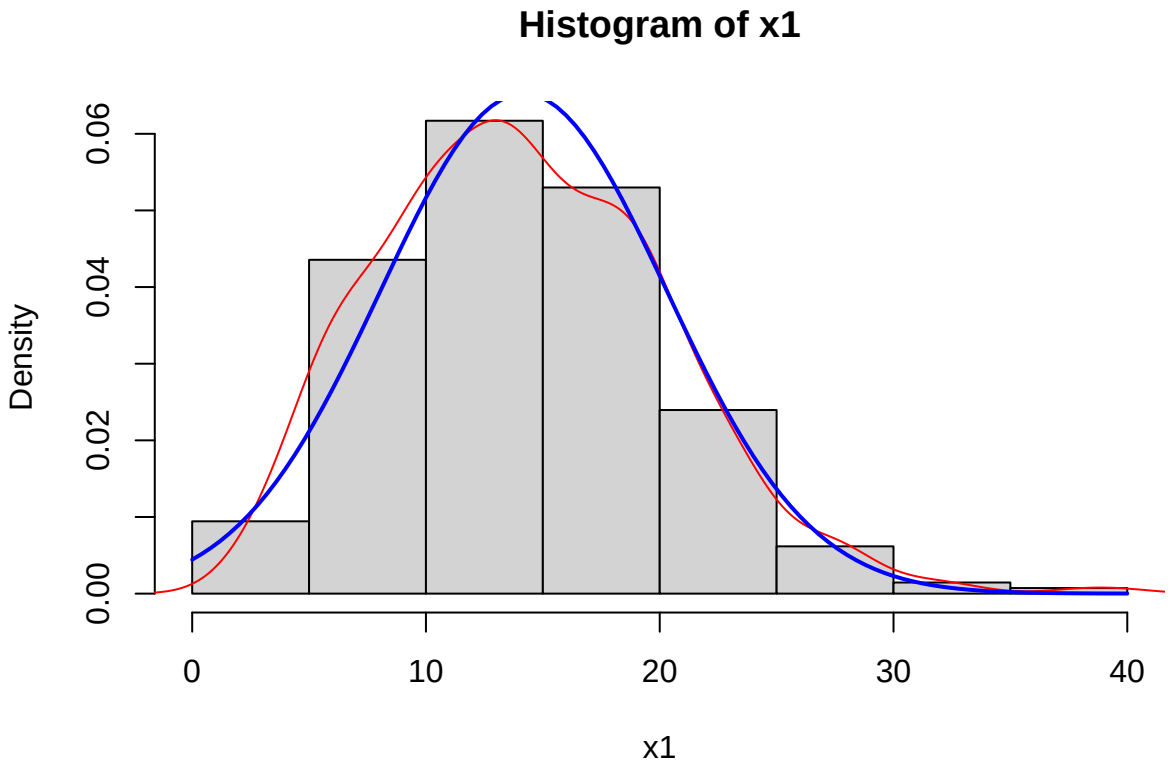
```
##                   Minimo     Q1 Mediana   Media      Q3   Máximo Curtosis
## Original           3.000 94.500 186.000 237.359 337.000 1578.000    6.725
## Modelo aproximado  2.000  9.772  13.675  14.178  18.385   39.737    0.350
## Modelo exacto      1.723  9.838  12.805  12.775  15.969   27.441   -0.205
##                    Sesgo
## Original           1.918
## Modelo aproximado  0.486
## Modelo exacto     -0.002
```

Grafica las funciones de densidad empírica y teórica de los 2 modelos obtenidos (exacto y aproximado) y los datos originales.

```r
hist(dataset,freq=FALSE)
lines(density(dataset),col="red")
curve(dnorm(x,mean=mean(dataset),sd=sd(dataset)), add=TRUE, col="blue",lwd=2)
```

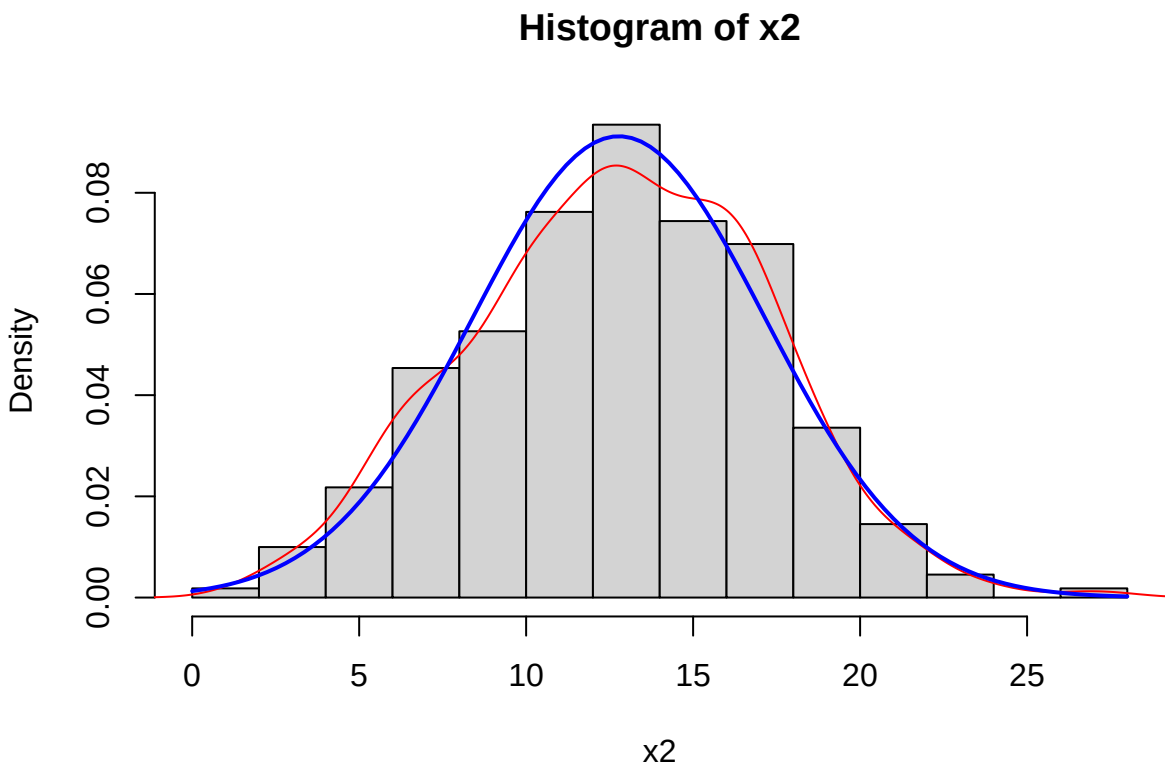



```
hist(x1,freq=FALSE)
lines(density(x1),col="red")
curve(dnorm(x,mean=mean(x1),sd=sd(x1)), add=TRUE, col="blue",lwd=2)
```




```
hist(x2,freq=FALSE)
lines(density(x2),col="red")
```

```
curve(dnorm(x,mean=mean(x2),sd=sd(x2)), add=TRUE, col="blue",lwd=2)
```



Realiza la prueba de normalidad de Anderson-Darling y de Jarque Bera para los datos transformados y los originales

```
D2=ad.test(x2)
d2 = jarque.test(x2)
cat("P-value de Modelo Exacto con Anderson-Darling",D2$p.value, '\n')
```

```
## P-value de Modelo Exacto con Anderson-Darling 0.1382034
```

```
cat("P-value de Modelo Exacto con Jarque-Bera",d2$p.value, '\n')
```

```
## P-value de Modelo Exacto con Jarque-Bera 0.6454385
```

```
D1=ad.test(x1)
d1 = jarque.test(x1)
cat("P-value de Modelo Aproximado con Anderson-Darling",D1$p.value, '\n')
```

```
## P-value de Modelo Aproximado con Anderson-Darling 0.002053753
```

```
cat("P-value de Modelo Aproximado con Jarque-Bera",d1$p.value, '\n')
```

```
## P-value de Modelo Aproximado con Jarque-Bera 4.013594e-06
```

```
D=ad.test(dataset)
d = jarque.test(dataset)
cat("P-value de Modelo Aproximado con Anderson-Darling",D$p.value, '\n')
```

```
## P-value de Modelo Aproximado con Anderson-Darling 3.7e-24
```

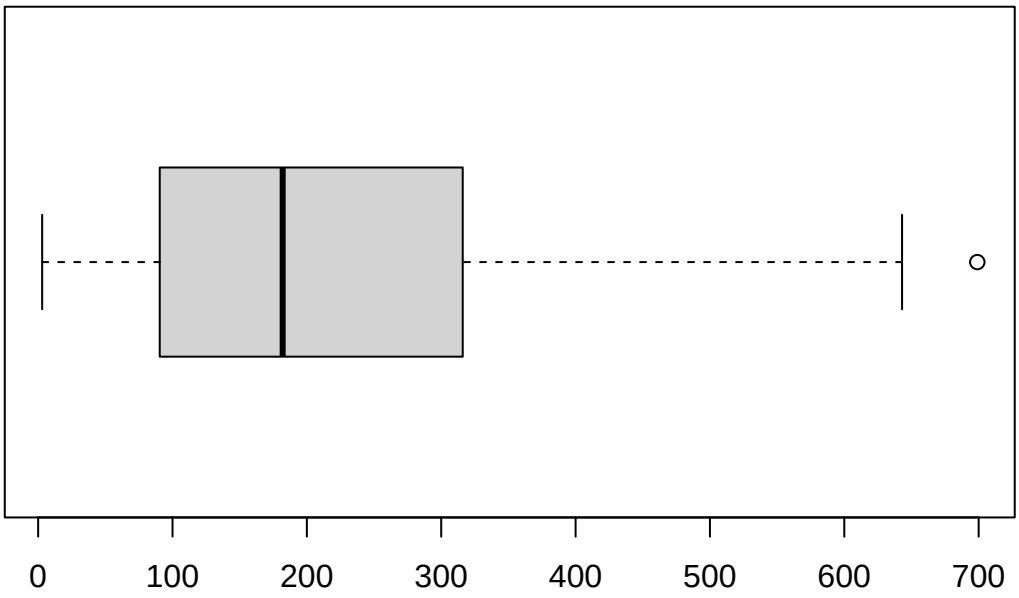
```
cat("P-value de Modelo Aproximado con Jarque-Bera",d$p.value, '\n')
```

```
## P-value de Modelo Aproximado con Jarque-Bera 0
```

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

Como se mencionó en puntos anteriores, se borrarán los datos atípicos a partir de las cotas 1.5 intercuartílicas.

```
calories <- dataset[dataset > q15i & dataset <= q15s]
boxplot(calories, horizontal = TRUE)
```




```
x11 = sqrt(calories+1) # Modelo 1
x22 = ((calories+1)^l - 1)/l # Modelo 2
```

Comenta la normalidad de las transformaciones obtenidas. Utiliza como argumento de normalidad:

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

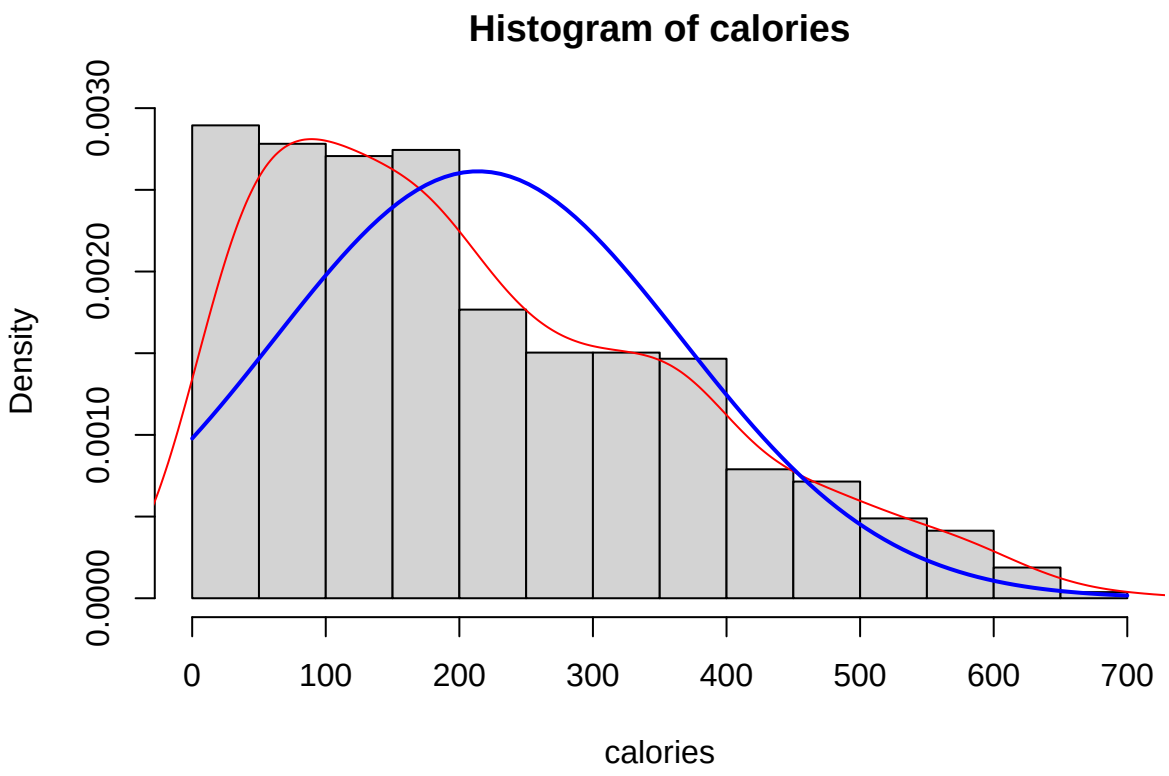
```
m01=round(c(as.numeric(summary(calories)),kurtosis(calories),skewness(calories)),3)
m11=round(c(as.numeric(summary(x11)),kurtosis(x11),skewness(x11)),3)
m21=round(c(as.numeric(summary(x22)),kurtosis(x22),skewness(x22)),3)

mm<-as.data.frame(rbind(m01,m11,m21))
row.names(mm)=c("Original","Modelo aproximado","Modelo exacto")
names(mm)=c("Minimo","Q1","Mediana","Media","Q3","Máximo","Curtosis","Sesgo")
mm
```

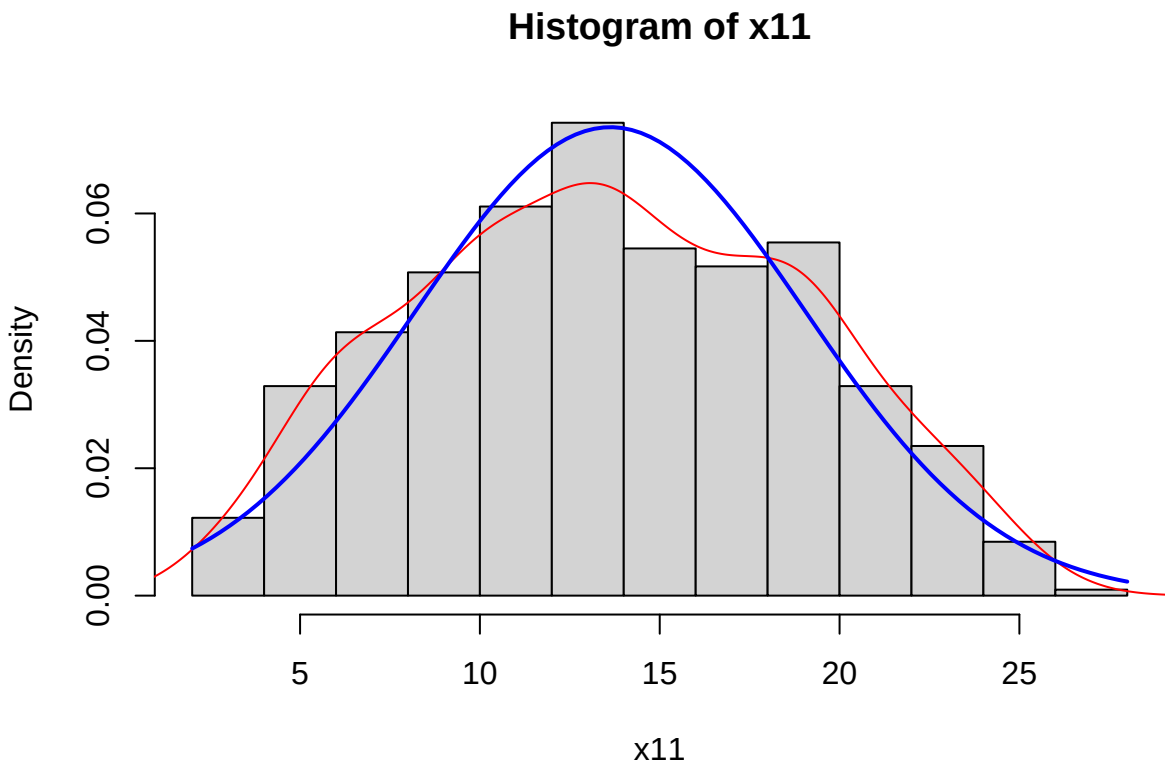
```
##                   Minimo     Q1 Mediana   Media      Q3  Máximo Curtosis  Sesgo
## Original           3.000 90.750 182.000 214.049 316.000 699.000   -0.252  0.725
## Modelo aproximado  2.000  9.579  13.528  13.627  17.804  26.458   -0.806  0.047
## Modelo exacto      1.723  9.679  12.699  12.431  15.598  20.725   -0.606 -0.289
```

Obten el histograma de los 2 modelos obtenidos (exacto y aproximado) y de los datos originales.

```
hist(calories,freq=FALSE)
lines(density(calories),col="red")
curve(dnorm(x,mean=mean(calories),sd=sd(calories)), add=TRUE, col="blue",lwd=2)
```

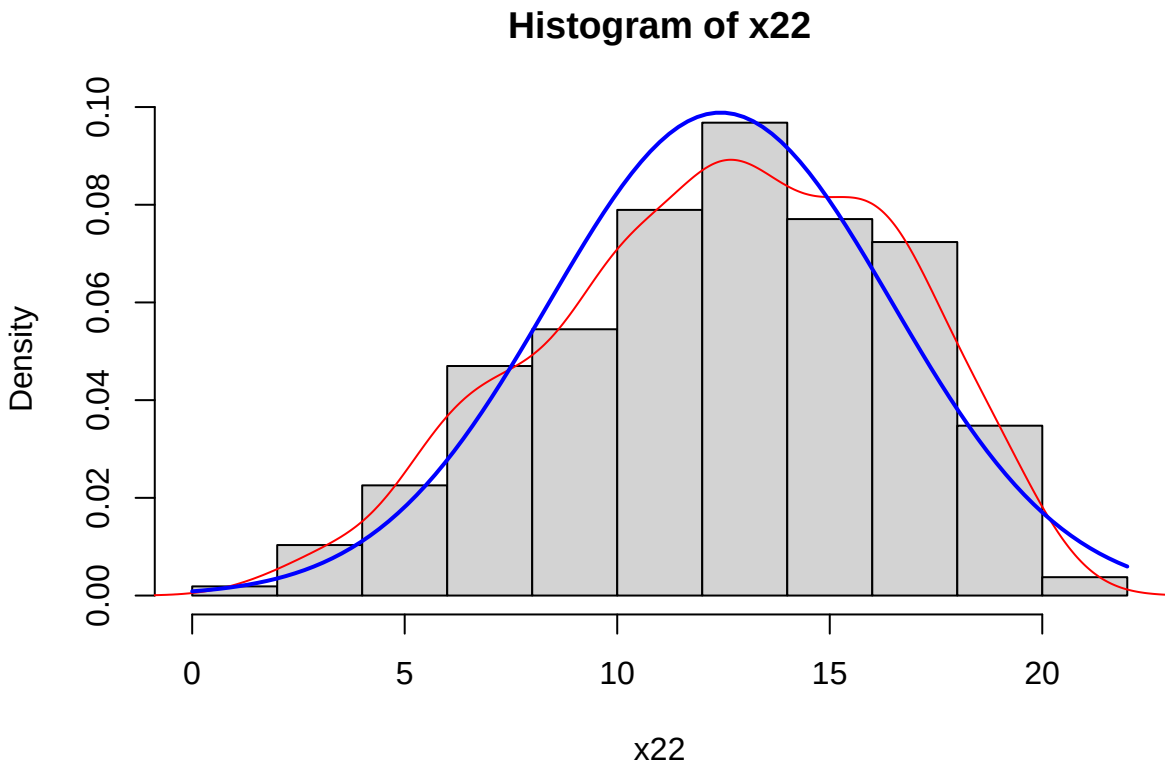



```
hist(x11,freq=FALSE)
lines(density(x11),col="red")
curve(dnorm(x,mean=mean(x11),sd=sd(x11)), add=TRUE, col="blue",lwd=2)
```




```
hist(x22,freq=FALSE)
lines(density(x22),col="red")
```

```
curve(dnorm(x,mean=mean(x22),sd=sd(x22)), add=TRUE, col="blue",lwd=2)
```



Interpreta la prueba de normalidad de Anderson-Darling y Jarque Bera para los datos transformados y los originales

```
D22=ad.test(x22)
d22 = jarque.test(x22)
cat("P-value de Modelo Exacto con Anderson-Darling",D22$p.value, '\n')
```

```
## P-value de Modelo Exacto con Anderson-Darling 3.680295e-05
```

```
cat("P-value de Modelo Exacto con Jarque-Bera",d22$p.value, '\n')
```

```
## P-value de Modelo Exacto con Jarque-Bera 0.0004612335
```

```
D11=ad.test(x11)
d11 = jarque.test(x11)
cat("P-value de Modelo Aproximado con Anderson-Darling",D11$p.value, '\n')
```

```
## P-value de Modelo Aproximado con Anderson-Darling 0.0004355751
```

```
cat("P-value de Modelo Aproximado con Jarque-Bera",d11$p.value, '\n')
```

```
## P-value de Modelo Aproximado con Jarque-Bera 0.0007817594
```

```
Dd=ad.test(calories)
dd = jarque.test(calories)
cat("P-value de Modelo Aproximado con Anderson-Darling",Dd$p.value, '\n')
```

```
## P-value de Modelo Aproximado con Anderson-Darling 1.521316e-22
```

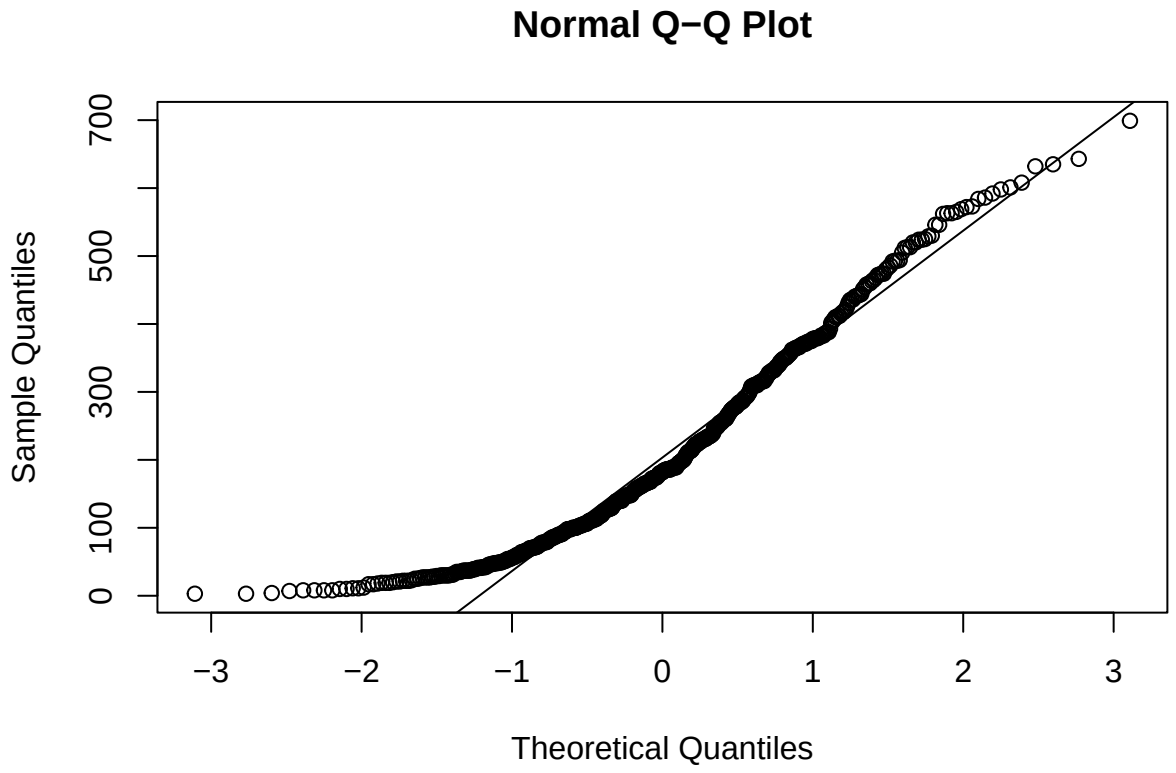
```
cat("P-value de Modelo Aproximado con Jarque-Bera",dd$p.value, '\n')
```

```
## P-value de Modelo Aproximado con Jarque-Bera 3.379474e-11
```

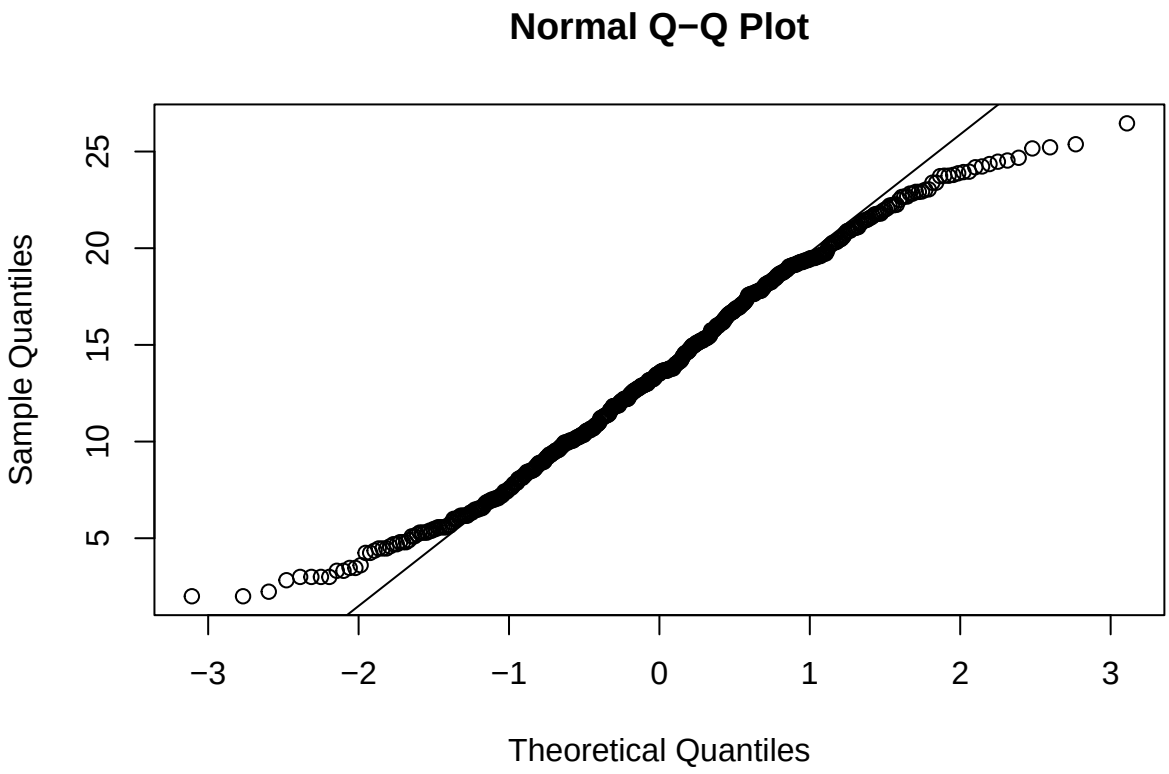
La prueba de normalidad para todos los modelos y los datos originales (ya corregidos) prueba que ninguna de las distribuciones es normal, pero la más cercan a serlo es el modelo exacto, que queda unas centécimas por debajo del valor de $\alpha = 0.05$. Pero si vamos a las pruebas de normalidad de los datos sin corregir, el modelo exacto logra la normalidad, pues sus p-values en ambas pruebas son mayores a α, es de esta forma que es el único modelo con evidencia de ser normal.

Indica posibilidades de motivos de alejamiento de normalidad (sesgo, curtosis, datos atípicos, etc)

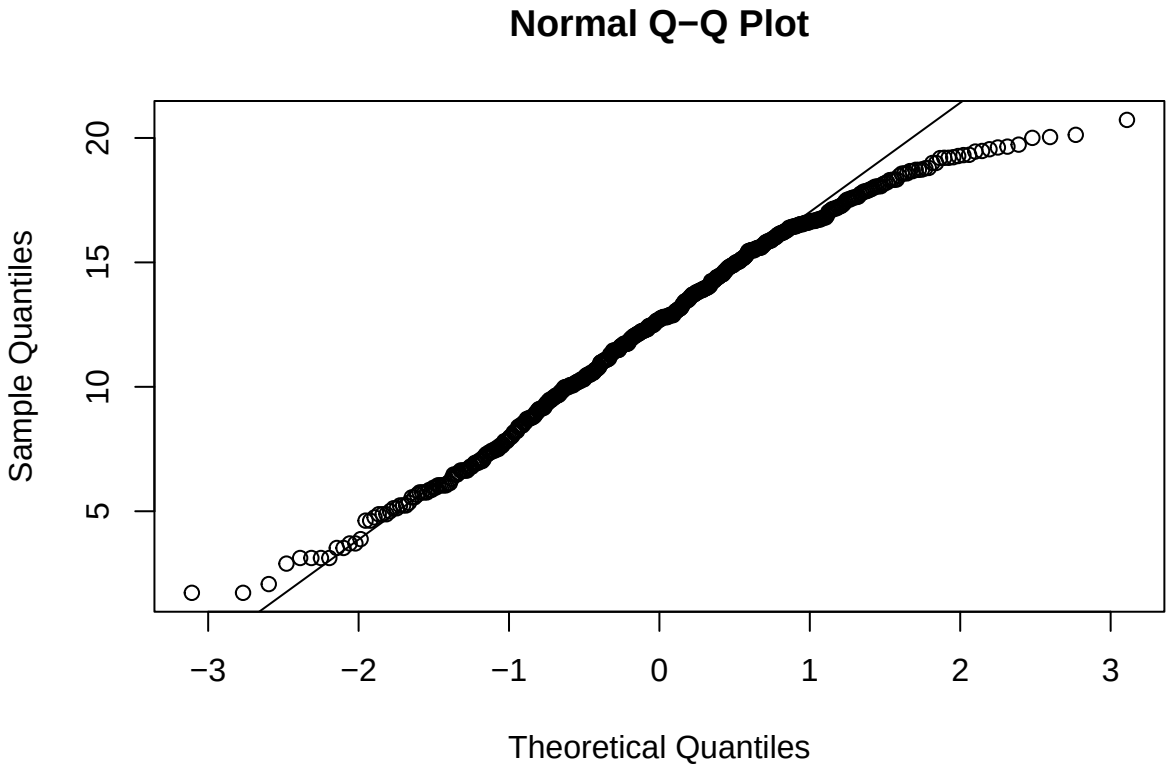
```
qqnorm(calories)
qqline(calories)
```




```
qqnorm(x11)
qqline(x11)
```




```
qqnorm(x22)
qqline(x22)
```



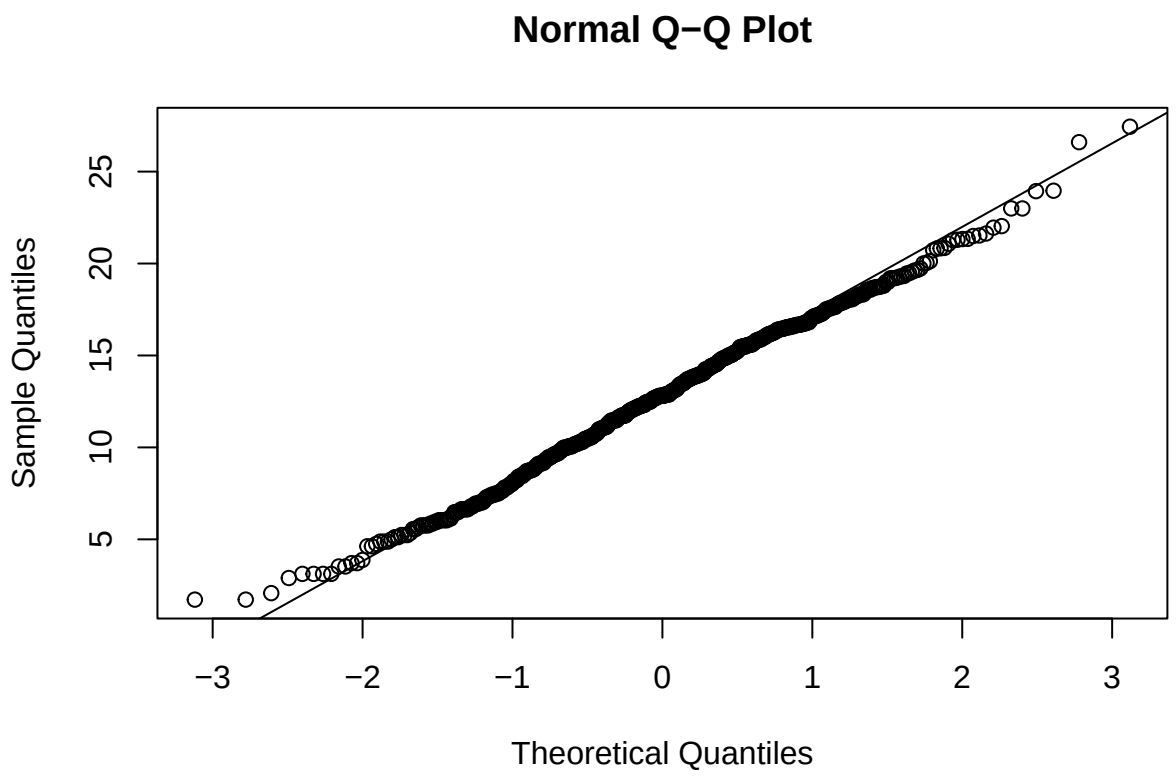
Después de corregir la base de datos, la curtosis y el sesgo de todos los modelos se redujo considerablemente, que hasta aparentan ser normales, pero después de observar detenidamente los histogramas y de ver que las líneas de la QQplot quedan abajo, podemos notar que hay un sesgo hacia la izquierda, es decir, los datos

empiezan a acumularse a la derecha debido a la corrección de la base de datos.

```
Define la mejor transformación de los datos de acuerdo a las características de los modelos que encontra
```

La mejor transformación fue sin duda la de del modelo exacto sin correción de los datos $x_2 = \frac{(x+1)^{0.3030}-1}{0.3030}$. Pues obtuvo un sesgo y curtosis muy cercanos a 0, además de que fue el único modelo que alcanzo la normalidad, pues su resultado en ambas pruebas de normalidad (Anderson-Darling y Jarque Bera) fueron mayores al valor estándar de las pruebas de hipótesis $\alpha = 0.05$.

```
qqnorm(x2)
qqline(x2)
```



En el QQplot observamos que el modelo exacto sin corregir se pega casi perfectamente a la línea, y no muestra ningún sesgo aparente.