marcelo

2024-08-20

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:

```
library(MASS)
library(e1071)
library(nortest)
M1 = read.csv("downloads/food_data_g.csv")
```

Sodio

```
datos = M1$Sodium
```

Graficar el diagrama de caja y bigote

```
q1_cal = quantile(datos, 0.25)
q3_cal = quantile(datos, 0.75)
ri_cal = IQR(datos)

par(mfrow=c(2,1))
boxplot(datos, horizontal=TRUE, ylim=c(0, 8))
abline(v=q3_cal + 1.5*ri_cal, col="red")
abline(v=q1_cal + 1.5*ri_cal, col="green")
abline(v=mean(datos) + 1.5*ri_cal, col="blue")
```

Calcula las principales medidas que te ayuden a identificar datos atípicos (utilizar summary te puede abreviar el cálculo): Cuartil 1, Cuartil 3, Media, Cuartil 3, Rango intercuartílico y Desviación estándar

```
summary(datos)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.1000  0.4000  0.5732  0.9000  6.1000
```

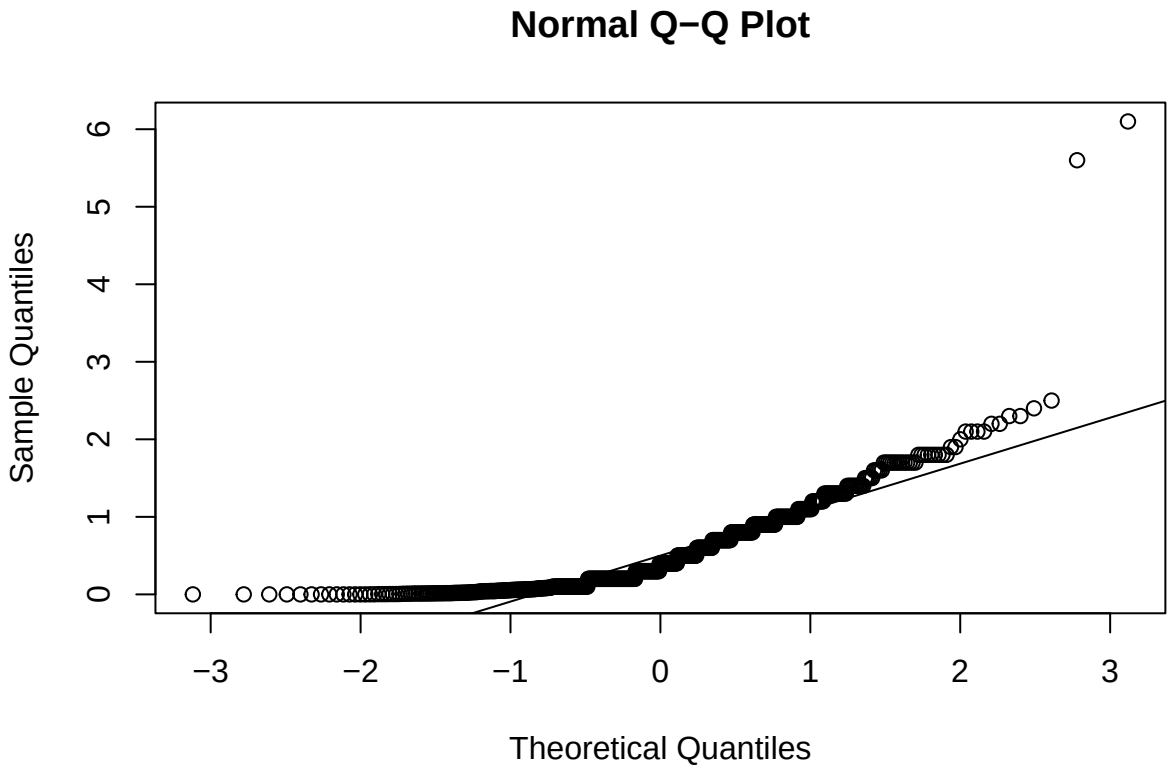
Realiza pruebas de normalidad univariada para la variable (utiliza las pruebas de Anderson-Darling y de Jarque Bera). No olvides incluir H0 y H1 para la prueba de normalidad.

```
ad.test(datos)
```

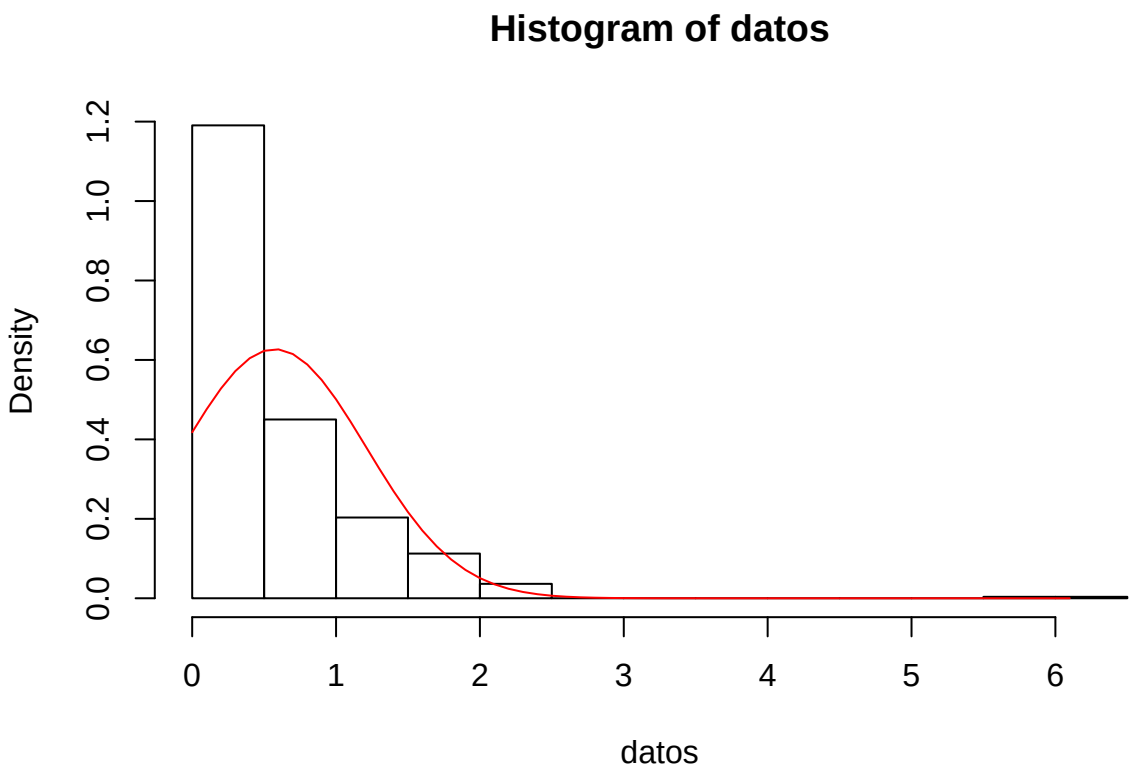
```
##
##  Anderson-Darling normality test
##
## data:  datos
## A = 24.827, p-value < 2.2e-16
```

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

```
qqnorm(datos)
qqline(datos)
```




```
hist(datos, freq=FALSE, col=0)
x_sod = seq(min(datos), max(datos), 0.1)
y_sod = dnorm(x_sod, mean=mean(datos), sd=sd(datos))
lines(x_sod, y_sod, col="red")
```



Calcula el coeficiente de sesgo y el coeficiente de curtosis

```
skewness(datos)
```

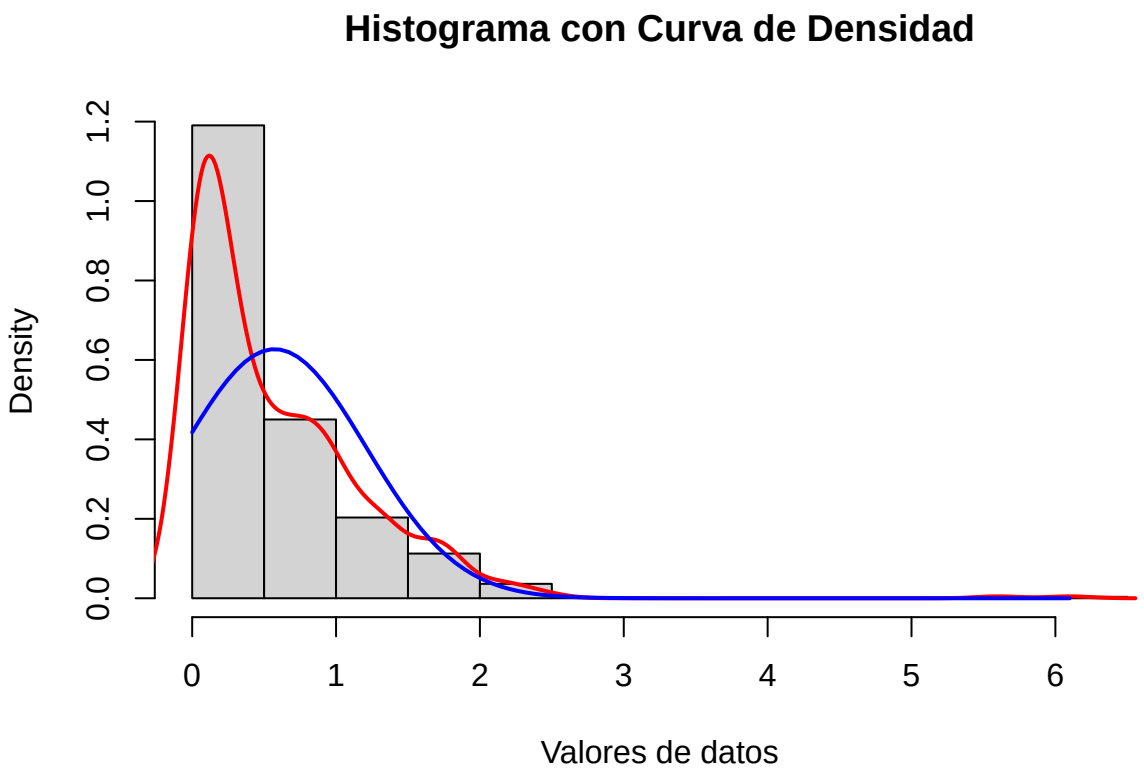
```
## [1] 2.728554
```

```
kurtosis(datos)
```

```
## [1] 16.29239
```

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

```
hist(datos, freq=FALSE, main="Histograma con Curva de Densidad", xlab="Valores de datos")

lines(density(datos), col="red", lwd=2)

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



Interpreta los gráficos y los resultados obtenidos en cada punto con vías a indicar si hay normalidad de los datos Comenta las características encontradas: Considera alejamientos de normalidad por simetría, curtosis Comenta si hay aparente influencia de los datos atípicos en la normalidad de los datos 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.

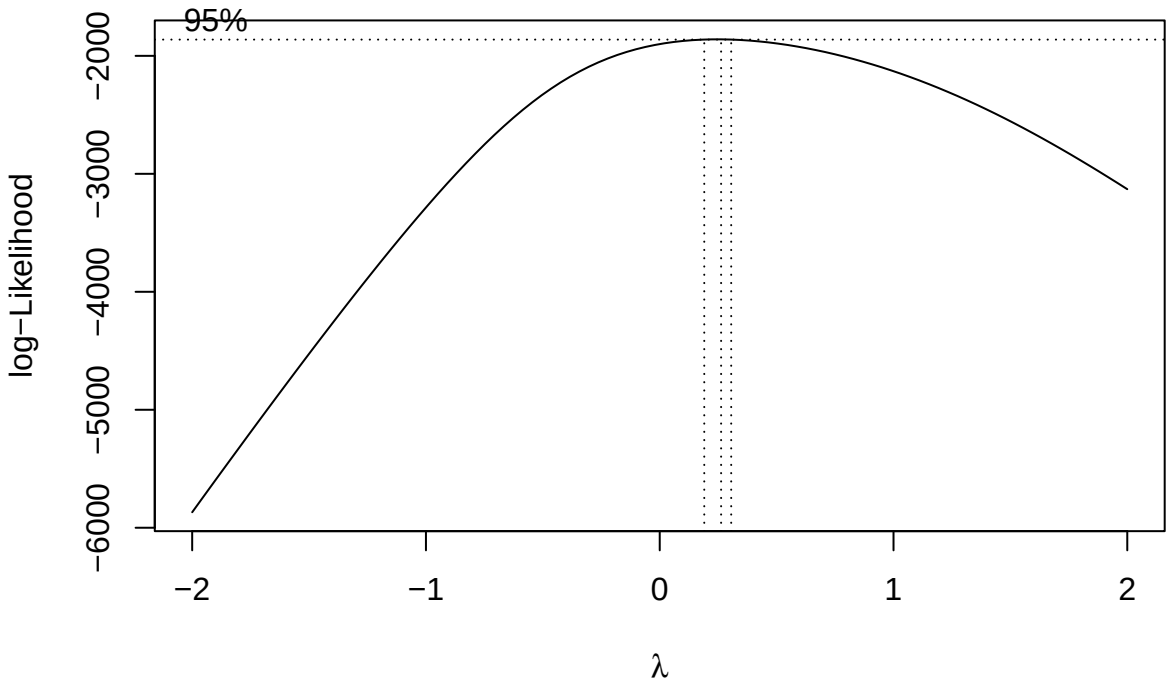
Resumen

Aqui si vemos como los datos atipicos influyen en los resultados ya que si tenemos algunos datos muy grandes, esto se aprecia mas en el boxcox donde tambien calculamos con los diferentes intercuartilicos y cada uno nos dieron varios datos atipicos diferentes, en el que se vieron mas fue cuando utilizabamos de 1.5 a comparacion de 3. Tambien en las graficas podemos ver nuestros resultados y en el summary vemos como es impactado por estos datos, para estos datos se recomienda quitar los 0 y tambien quitar los datos atipicos arriba de 3 o 4 para que no se vean tan afectados nuestros datos. Con estos datos vemos que no hay normalidad, tambien con los datos de curtosis y otros resultados que se nos dieron vemos que necesitamos hacer cambios para no alejarnos de normalidad.

2. Transformación a normalidad

Encuentra la mejor transformación de los datos para lograr normalidad. Puedes hacer uso de la transformación Box-Cox o de Yeo Johnson o el comando powerTransform para encontrar la mejor lambda para la transformación. Utiliza el modelo exacto y el aproximado de acuerdo con las sugerencias de Box y Cox para la transformación.

```
M2 = subset(M1, Sodium > 0)
sodP = M2$Sodium

bc = boxcox(sodP ~ 1)
```




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

lambda_opt
```

```
## [1] 0.2626263
```

Escribe las ecuaciones de los modelos de transformación encontrados.

$sqrt(X + 1)$ $X^lambda - 1/lambda$

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.

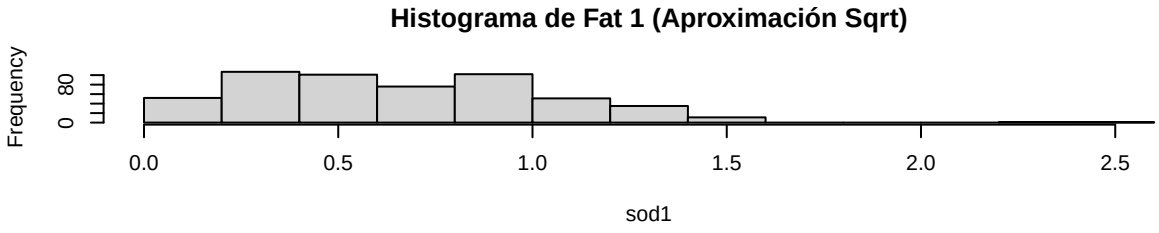
```
lambda_opt = bc$x[which.max(bc$y)]

sod1 = sqrt(sodP)
sod2 = (sodP^lambda_opt - 1) / lambda_opt

kurtosis_original = kurtosis(sodP)
skewness_original = skewness(sodP)

kurtosis_sod1 = kurtosis(sod1)
skewness_sod1 = skewness(sod1)

kurtosis_sod2 = kurtosis(sod2)
skewness_sod2 = skewness(sod2)

D0 = ad.test(sodP)
D1 = ad.test(sod1)
D2 = ad.test(sod2)

resumen_original = round(c(as.numeric(summary(sodP)), kurtosis_original, skewness_original, D0$p.value)
resumen_sod1 = round(c(as.numeric(summary(sod1)), kurtosis_sod1, skewness_sod1, D1$p.value), 3)
resumen_sod2 = round(c(as.numeric(summary(sod2)), kurtosis_sod2, skewness_sod2, D2$p.value), 3)
```

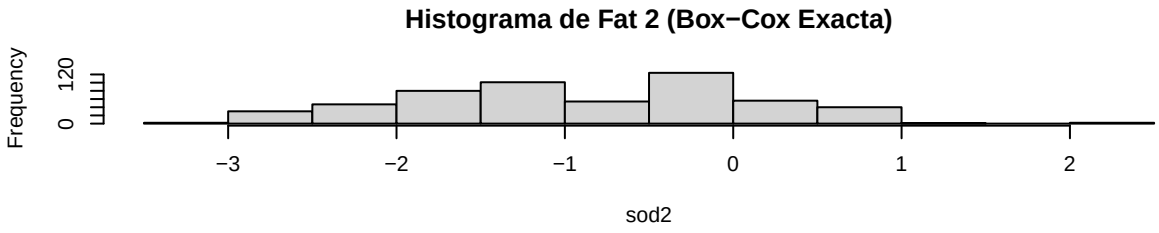
```r
resumen = as.data.frame(rbind(resumen_original, resumen_sod1, resumen_sod2))
row.names(resumen) = c("Original", "Fat 1 (sqrt)", "Fat 2 (Box-Cox)")
colnames(resumen) = c("Mínimo", "Q1", "Mediana", "Media", "Q3", "Máximo", "Curtosis", "Sesgo", "Valor p

print(resumen)
```

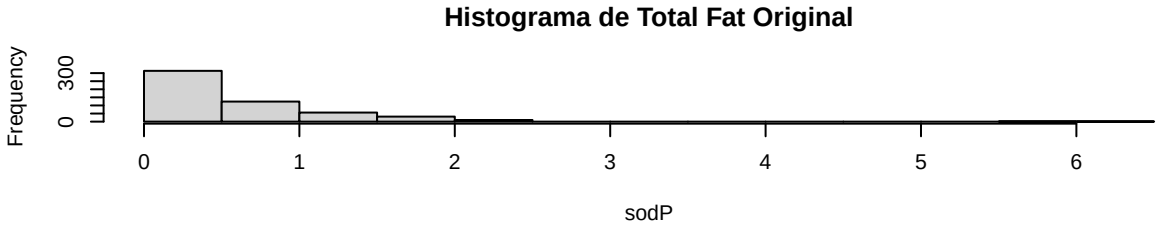
```
##                  Mínimo     Q1 Mediana  Media     Q3 Máximo Curtosis  Sesgo
## Original          0.001  0.100   0.400  0.588  0.900  6.100   16.348  2.730
## Fat 1 (sqrt)      0.032  0.316   0.632  0.661  0.949  2.470    0.301  0.571
## Fat 2 (Box-Cox) -3.187 -1.728  -0.814 -0.892 -0.104  2.315   -0.710 -0.056
##                  Valor p
## Original               0
## Fat 1 (sqrt)           0
## Fat 2 (Box-Cox)        0
```

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

```r
par(mfrow=c(3,1))
hist(sod1, main="Histograma de Fat 1 (Aproximación Sqrt)")
hist(sod2, main="Histograma de Fat 2 (Box-Cox Exacta)")
hist(sodP, main="Histograma de Total Fat Original")
```



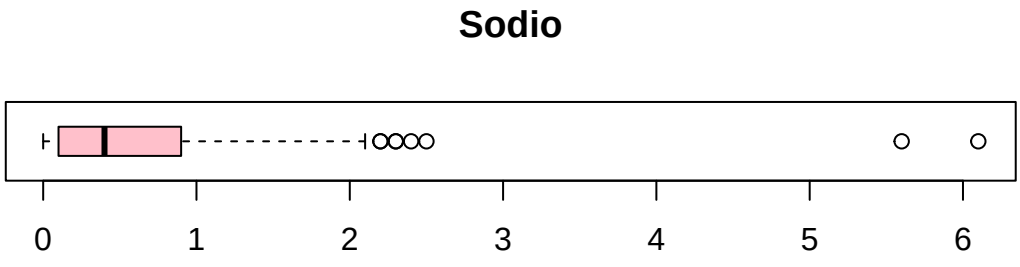


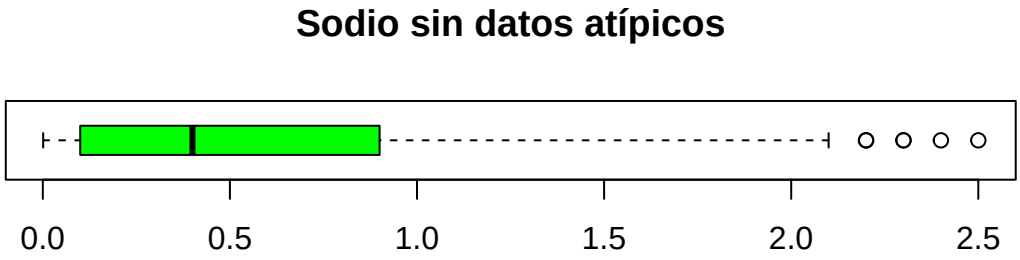


Realiza la prueba de normalidad de Anderson-Darling y de Jarque Bera para los datos transformados y los originales Detecta anomalías y corrige tu base de datos (datos atípicos, ceros anámalos, etc).

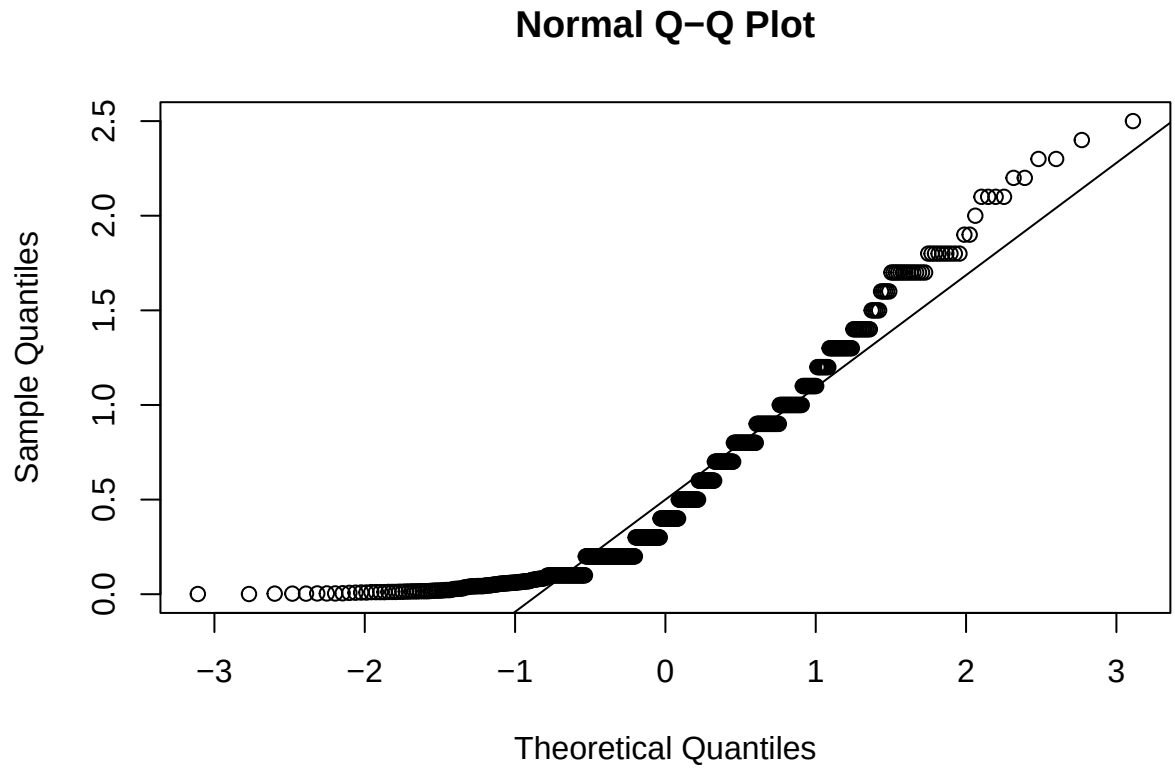
```r
sodP <- sodP[sodP <= 3]
par(mfrow=c(2,1))
boxplot(datos, horizontal = TRUE, col="pink", main="Sodio")
```

```r
boxplot(sodP, horizontal = TRUE, col="green", main="Sodio sin datos atípicos")
```



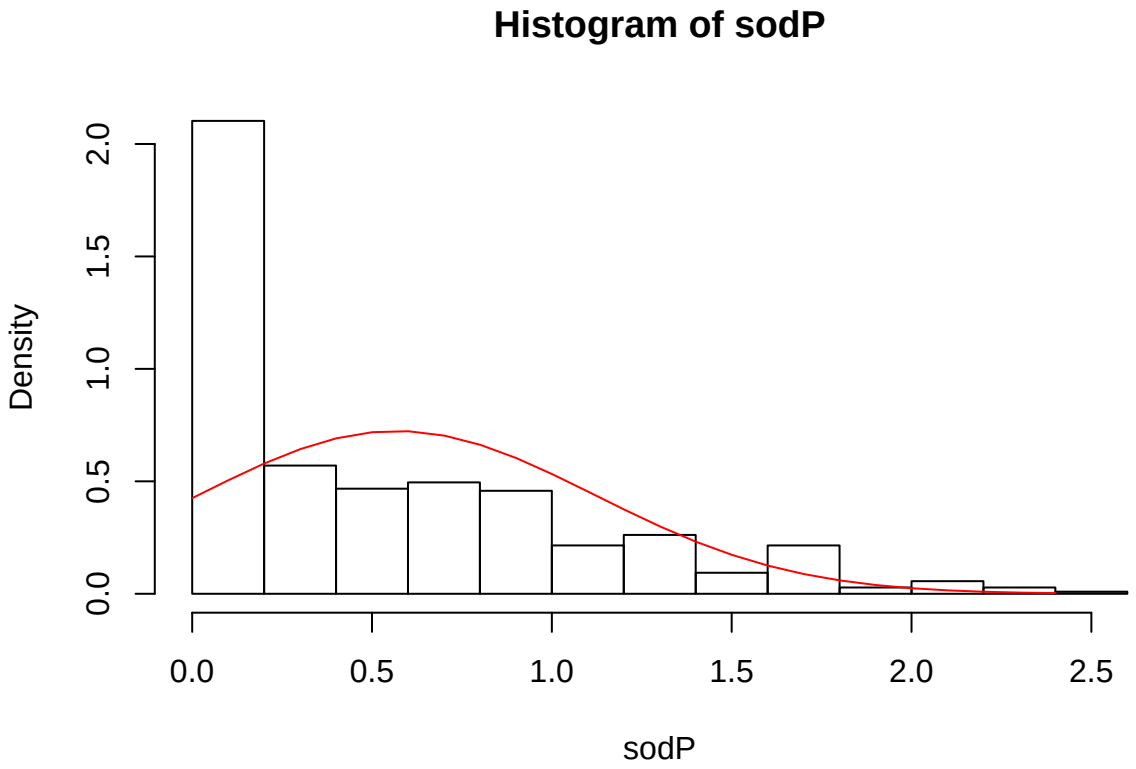



```r
qqnorm(sodP)
qqline(sodP)
```




```r
hist(sodP, freq=FALSE, col=0)
x_sod = seq(min(sodP), max(sodP), 0.1)
y_sod = dnorm(x_sod, mean=mean(sodP), sd=sd(sodP))
```

```r
lines(x_sod, y_sod, col="red")
```



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

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

Define la mejor transformación de los datos de acuerdo a las características de los modelos que encontraste. Toma en cuenta los criterios del inciso anterior para analizar normalidad y la economía del modelo.

Despues de nuestras pruebas y viendo los resultados y graficas de las diferentes cosas, no llegamos a tener una normalidad, ya que nuestro p-value necesita ser mayor a 0.05 pero vemos como no podemos llegar a esto, en un punto si llegue a tener un p-value de 0.085 pero luego trono mi programa y no pude replicar la respuesta.