

4. Explorando bases

Adrian Pineda Sanchez

2024-08-13

```
M=read.csv("mc-donalds-menu.csv") #Leer La base de datos
```

Selección de las variables a explorar: datos atípicos y Normalidad

```
Proteinas <- M$Protein  
  
# Contar el número de datos  
Conteo_datos_proteinas <- length(Proteinas)  
  
print(Conteo_datos_proteinas)  
  
## [1] 260
```

Debido a que contamos con más de 260 datos podríamos omitir la de Shapiro debido a $n > 50$ pero igualmente observaremos su implicación

```
Calorias <- M$Calories  
  
# Contar el número de datos  
Conteo_datos_calorias <- length(Calorias)  
  
print(Conteo_datos_calorias)  
  
## [1] 260
```

De igual medida que las proteínas, calorías con $n > 50$

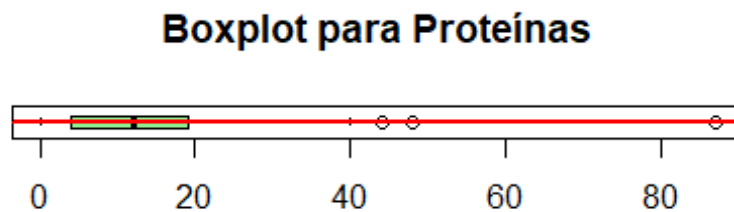
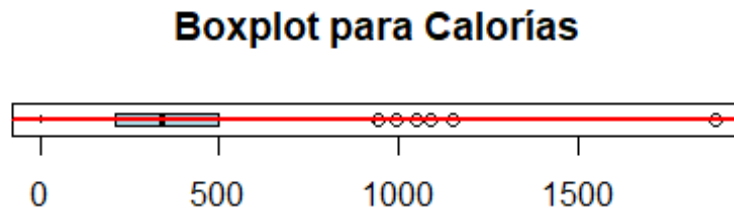
```
M <- data.frame(Calorias = Calorias, Proteinas = Proteinas )
```

Análisis Datos Atípicos

Graficar el diagrama de caja y bigote

```
par(mfrow=c(2,1)) # Para mostrar dos gráficos en una matriz de 2x1  
  
# Boxplot para Calorías  
boxplot(Calorias, horizontal = TRUE, main = "Boxplot para Calorías", col =  
"lightblue")  
abline(h=1, col="red", lwd=2) # Línea de referencia en el límite de los  
datos atípicos  
  
# Boxplot para Proteínas
```

```
boxplot(Proteinas, horizontal = TRUE, main = "Boxplot para Proteínas", col =
"lightgreen")
abline(h=1, col="red", lwd=2) # Línea de referencia en el límite de los
datos atípicos
```



Como podemos ver observamos unos outliers que se escapan dentro del rango establecido por los boxplots, aprox unos 6 en Calorías, y unos 3 en proteínas

Calcula el rango intercuartílico y los cuartiles

```
# Calcular cuartiles y rango intercuartílico para Calorías
q1_cal <- quantile(Calorías, 0.25)
q3_cal <- quantile(Calorías, 0.75)
iqr_cal <- IQR(Calorías)

cat("Cuartil 1 para Calorías:", q1_cal, "\n")
## Cuartil 1 para Calorías: 210

cat("Cuartil 3 para Calorías:", q3_cal, "\n")
## Cuartil 3 para Calorías: 500

cat("Rango intercuartílico para Calorías:", iqr_cal, "\n")
## Rango intercuartílico para Calorías: 290

# Calcular cuartiles y rango intercuartílico para Proteínas
q1_prot <- quantile(Proteinas, 0.25)
```

```

q3_prot <- quantile(Proteinas, 0.75)
iqr_prot <- IQR(Proteinas)

cat("Cuartil 1 para Proteínas:", q1_prot, "\n")
## Cuartil 1 para Proteínas: 4

cat("Cuartil 3 para Proteínas:", q3_prot, "\n")
## Cuartil 3 para Proteínas: 19

cat("Rango intercuartílico para Proteínas:", iqr_prot, "\n")
## Rango intercuartílico para Proteínas: 15

```

Identifica la cota de 1.5 rangos intercuartílicos para datos atípicos, ¿hay datos atípicos de acuerdo con este criterio?

```

# Para Calorías
cota_inf_cal_1_5iqr <- q1_cal - 1.5 * iqr_cal
cota_sup_cal_1_5iqr <- q3_cal + 1.5 * iqr_cal

# Imprimir Las cotas
cat("Cota inferior 1.5 IQR para Calorías:", cota_inf_cal_1_5iqr, "\n")
## Cota inferior 1.5 IQR para Calorías: -225

cat("Cota superior 1.5 IQR para Calorías:", cota_sup_cal_1_5iqr, "\n")
## Cota superior 1.5 IQR para Calorías: 935

# Identificar Los datos atípicos
outliers_inf_1_5iqr_cal <- Calorias[Calorias < cota_inf_cal_1_5iqr]
outliers_sup_1_5iqr_cal <- Calorias[Calorias > cota_sup_cal_1_5iqr]

# Imprimir Los datos atípicos
cat("\n")

cat("Datos atípicos inferiores según 1.5 IQR para Calorías:",
outliers_inf_1_5iqr_cal, "\n")
## Datos atípicos inferiores según 1.5 IQR para Calorías:

cat("Datos atípicos superiores según 1.5 IQR para Calorías:",
outliers_sup_1_5iqr_cal, "\n")
## Datos atípicos superiores según 1.5 IQR para Calorías: 1090 1150 990 1050
940 1880

```

Como podemos ver, la cota inferior no es relevante ni lógica en este contexto a menos que se tenga una base de datos sin limpiar, debido a que arrojaría un valor incongruente e ilógico debido a que el conteo de calorías no puede ser negativo en un alimento.

Ahora, entrando en cuestion con la cota superior, podemos ver outliers o valores atipicos detectados encima de 1.5 IQR que son: 1090 1150 990 1050 940 1880. Sin embargo, no me parece raro dejarlos debido a que si lo analizamos, muchos alimentos de comida rapida pueden tener un alto indice calorico, y debido a que ninguno arroja un valor ilogico dentro del marco alimenticio de comida rapida, podriamos dejarlos.

```
# Para Proteínas
cota_inf_prot_1_5iqr <- q1_prot - 1.5 * iqr_prot
cota_sup_prot_1_5iqr <- q3_prot + 1.5 * iqr_prot

# Imprimir Las cotas
cat("Cota inferior 1.5 IQR para Proteínas:", cota_inf_prot_1_5iqr, "\n")
## Cota inferior 1.5 IQR para Proteínas: -18.5

cat("Cota superior 1.5 IQR para Proteínas:", cota_sup_prot_1_5iqr, "\n")
## Cota superior 1.5 IQR para Proteínas: 41.5

# Identificar Los datos atípicos
outliers_inf_1_5iqr_prot <- Proteinas[Proteinas < cota_inf_prot_1_5iqr]
outliers_sup_1_5iqr_prot <- Proteinas[Proteinas > cota_sup_prot_1_5iqr]

# Imprimir Los datos atípicos
cat("\n")

cat("Datos atípicos inferiores según 1.5 IQR para Proteínas:",
outliers_inf_1_5iqr_prot, "\n")
## Datos atípicos inferiores según 1.5 IQR para Proteínas:

cat("Datos atípicos superiores según 1.5 IQR para Proteínas:",
outliers_sup_1_5iqr_prot, "\n")
## Datos atípicos superiores según 1.5 IQR para Proteínas: 48 44 87
```

Datos atípicos superiores según 1.5 IQR para Proteínas: 48 44 87, gramos de proteína. Aunque son mas altos y sobrepasan la cota superior, muy probablemente suceda lo mismo con respecto de los alimentos de las Calorias pasadas, en mi opinion, debido a la interpretacion del problema no me parece que deban ser removidos.

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

Cota 3 desviaciones estandar Calorias

```
# Para Calorías
cota_sup_cal_3sd <- mean(Calorias) + 3 * sd(Calorias)

# Imprimir La cota
cat("Cota superior 3 desviaciones estándar para Calorías:", cota_sup_cal_3sd,
"\n")
```

```
## Cota superior 3 desviaciones estándar para Calorías: 1089.079

# Identificar los datos atípicos
outliers_3sd_cal <- Calorias[Calorias > cota_sup_cal_3sd]

# Imprimir los datos atípicos
cat("Datos atípicos según 3 SD para Calorías:", outliers_3sd_cal, "\n")

## Datos atípicos según 3 SD para Calorías: 1090 1150 1880
```

Podemos observar que es un rango incluso mas permisible que el anterior, debido a que ahora son solo: 1090 1150 1880 los datos atipicos, y si bien estan alejados 3 desviaciones estandar aprox el 99.73% lo cual en otro contexto podria ser algo considerable para eliminarlos, no olvidemos que hablamos de alimentos de comida rapida, donde una comida completa aproximadamente se encuentra entre cerca de las 1200 calorías en McDonalds, por lo que perfectamente podria ser un paquete de comida completa incluso el de 1880, podriamos empezar a sospechar en torno a si excede en un rango superior a las 2000,2500 o incluso 3000 Calorias debido a que no seria lo habitual esos valores aunque podrian ocurrir.

Cota 3 desviaciones estandar Proteinas

```
# Para Proteínas
cota_sup_prot_3sd <- mean(Proteinas) + 3 * sd(Proteinas)

cat("Cota superior 3 desviaciones estándar para Proteínas:",
    cota_sup_prot_3sd, "\n")

## Cota superior 3 desviaciones estándar para Proteínas: 47.6169

outliers_3sd_prot <- Proteinas[Proteinas > cota_sup_prot_3sd]

cat("Datos atípicos según 3 SD para Proteínas:", outliers_3sd_prot, "\n")

## Datos atípicos según 3 SD para Proteínas: 48 87
```

Mismo Caso que con las Calorias, posiblemente adicional a Grasas y Carbs, los alimentos mostrados con las Calorias mas altas, tambien sean los de las proteinas mas altas, debido a que 1 gr proteina equivale a 4 Cal, posiblemente para el 87 gr estamos hablando que su aporte cal es aprox de 360 Cal solo en proteinas, donde seguramente predominara incluso mas en Grasas y Carbs su aporte calorico. No suena algo que no podria pasar en la vida real.

Toma una decisión de si conviene o no quitar los datos atípicos (para ello interpreta la variable en el contexto del problema y determina si es necesario quitarlos o no quitarlos)

En mi opinion, no conviene quitarlos ya que dichos valores tambien nos aportan informacion relevante del problema, sin mencionar que son pocos, y el hecho de que no caigan dentro de los intervalos establecidos no determina un error en su muestra, ya que en el contexto del problema sigue conservando sentido.

Sin embargo al final del documento hare una comparativa sin estos valores, aunque mi recomendacion seria si las pruebas salen negativas a normalidad, seria no interpretar erroneamente queriendo forzar una normalidad si no existe en los datos.

Analisis Normalidad

1. Realiza pruebas de normalidad univariada de las variables (selecciona entre los métodos vistos en clase)

Empezamos con el Supuesto de:

H_0 :La variable sigue una distribución normal

H_1 :La variable no sigue una distribución normal

Anderson Darling (n>50)

```
library(nortest)

ad.test(Calorias)

##
## Anderson-Darling normality test
##
## data: Calorias
## A = 2.5088, p-value = 2.369e-06

ad.test(Proteinas)

##
## Anderson-Darling normality test
##
## data: Proteinas
## A = 4.7515, p-value = 8.515e-12
```

cuando p-value < 0.05 se rechaza la hipotesis nula, por lo tanto descartamos normalidad por esta prueba

Cramer-Von Mises

```
cvm.test(Calorias)

##
## Cramer-von Mises normality test
##
## data: Calorias
## W = 0.38145, p-value = 4.102e-05

cvm.test(Proteinas)

##
## Cramer-von Mises normality test
```

```
##  
## data: Proteinas  
## W = 0.67776, p-value = 8.993e-08
```

cuando $p\text{-value} < 0.05$ se rechaza la hipótesis nula, por lo tanto descartamos normalidad por esta prueba

Kolmogorov-Smirnov

```
lillie.test(Calorias)
```

```
##  
## Lilliefors (Kolmogorov-Smirnov) normality test  
##  
## data: Calorias  
## D = 0.073753, p-value = 0.001611
```

```
lillie.test(Proteinas)
```

```
##  
## Lilliefors (Kolmogorov-Smirnov) normality test  
##  
## data: Proteinas  
## D = 0.12153, p-value = 5.806e-10
```

cuando $p\text{-value} < 0.05$ se rechaza la hipótesis nula, por lo tanto descartamos normalidad por esta prueba

Jarque-Bera

```
library(moments)
```

```
jarque.test(Calorias)
```

```
##  
## Jarque-Bera Normality Test  
##  
## data: Calorias  
## JB = 435.62, p-value < 2.2e-16  
## alternative hypothesis: greater
```

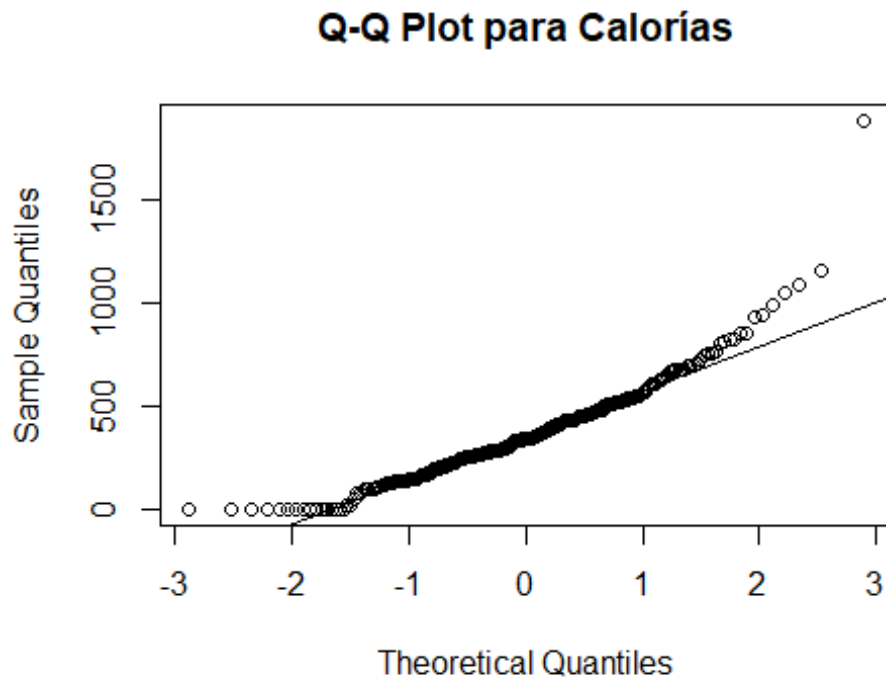
```
jarque.test(Proteinas)
```

```
##  
## Jarque-Bera Normality Test  
##  
## data: Proteinas  
## JB = 479.38, p-value < 2.2e-16  
## alternative hypothesis: greater
```

cuando $p\text{-value} < 0.05$ se rechaza la hipótesis nula, por lo tanto descartamos normalidad por esta prueba

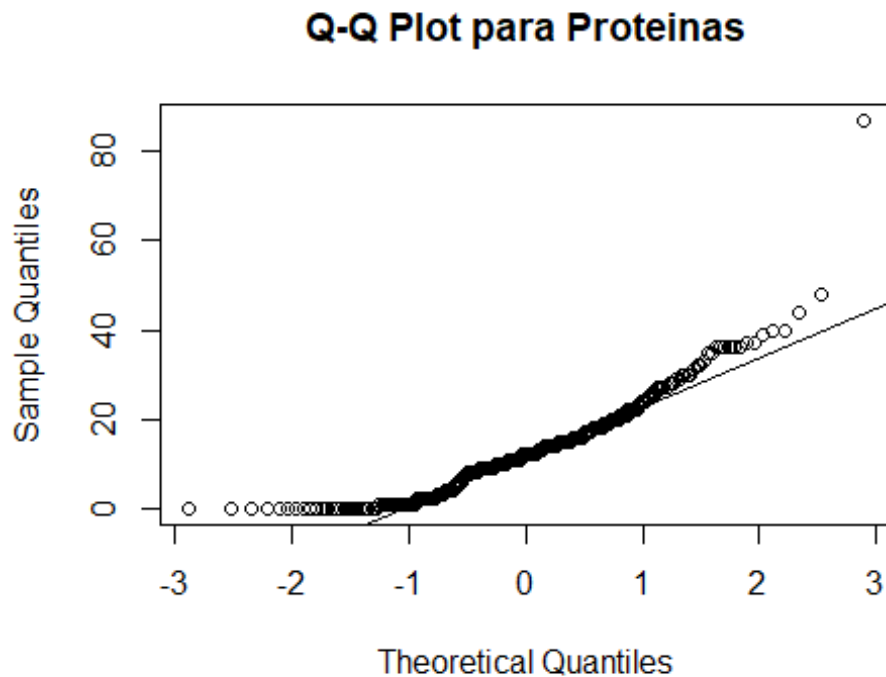
2. Grafica los datos y su respectivo QQPlot: `qqnorm(datos)` y `qqline(datos)` para cada variable

```
qqnorm(Calorias, main = "Q-Q Plot para Calorías")  
qqline(Calorias)
```



Podríamos juzgar por el gráfico que se muestra una aparente normalidad entre $[-1.5, 1]$ debido a que los datos están sumamente pegados a la línea de tendencia, y podríamos notar que la mayoría de los 260 datos se encuentran concentrados en dicha zona, sin embargo, los datos de los extremos son sumamente importantes para esto, y con lo que podríamos descartar normalidad debido a que en las colas parece desviarse de la misma.

```
qqnorm(Proteinas, main = "Q-Q Plot para Proteinas")  
qqline(Proteinas)
```

Al parecer por los datos podríamos notar hasta cierto punto entre $[-1,1]$ que los datos siguen una parente normalidad debido a la línea de tendencia en el Q-Q plot, sin embargo en sus valores extremos parece alejarse y desviarse de la misma línea, siendo al principio queda sin cambio aparente hasta el -1, y que a partir del 1, despega de la línea de tendencia, es muy importante notar las colas de los gráficos, ya que nos podría ayudar a negar normalidad.

3. Calcula el coeficiente de sesgo y el coeficiente de curtosis de cada variable.

```
# Calcular y mostrar el sesgo y la curtosis para Calorías
sesgo_calorias <- skewness(Calorias)
curtosis_calorias <- kurtosis(Calorias)

cat("Sesgo de Calorías:", sesgo_calorias, "\n")

## Sesgo de Calorías: 1.444105

cat("Curtosis de Calorías:", curtosis_calorias, "\n")

## Curtosis de Calorías: 8.645274

# Calcular y mostrar el sesgo y la curtosis para Proteínas
sesgo_proteinas <- skewness(Proteinas)
curtosis_proteinas <- kurtosis(Proteinas)

cat("Sesgo de Proteínas:", sesgo_proteinas, "\n")

## Sesgo de Proteínas: 1.570794
```

```
cat("Curtosis de Proteínas:", curtosis_proteinas, "\n")  
## Curtosis de Proteínas: 8.86355
```

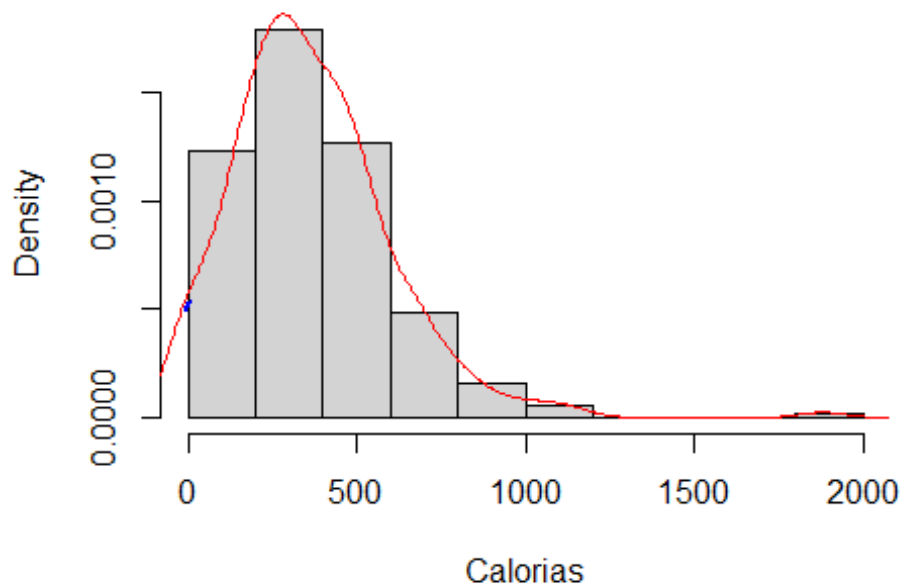
4. Compara las medidas de media, mediana y rango medio de cada variable.

```
# Comparar media, mediana y rango de Calorías  
media_calorias <- mean(Calorias)  
mediana_calorias <- median(Calorias)  
rango_calorias <- range(Calorias)  
  
cat("Media de Calorías:", media_calorias, "\n")  
## Media de Calorías: 368.2692  
  
cat("Mediana de Calorías:", mediana_calorias, "\n")  
## Mediana de Calorías: 340  
  
cat("Rango de Calorías:", rango_calorias, "\n")  
## Rango de Calorías: 0 1880  
  
# Comparar media, mediana y rango de Proteínas  
media_proteinas <- mean(Proteinas)  
mediana_proteinas <- median(Proteinas)  
rango_proteinas <- range(Proteinas)  
  
cat("Media de Proteínas:", media_proteinas, "\n")  
## Media de Proteínas: 13.33846  
  
cat("Mediana de Proteínas:", mediana_proteinas, "\n")  
## Mediana de Proteínas: 12  
  
cat("Rango de Proteínas:", rango_proteinas, "\n")  
## Rango de Proteínas: 0 87
```

5. Realiza el histograma y su distribución teórica de probabilidad

```
# Histograma y distribución teórica para Calorías  
hist(Calorias, freq = FALSE, main = "Histograma y Distribución Teórica para Calorías")  
lines(density(Calorias), col = "red")  
curve(dnorm(x, mean = mean(Calorias), sd = sd(Calorias)),  
      from = -6, to = 6, add = TRUE, col = "blue", lwd = 2)
```

Histograma y Distribución Teórica para Calorías

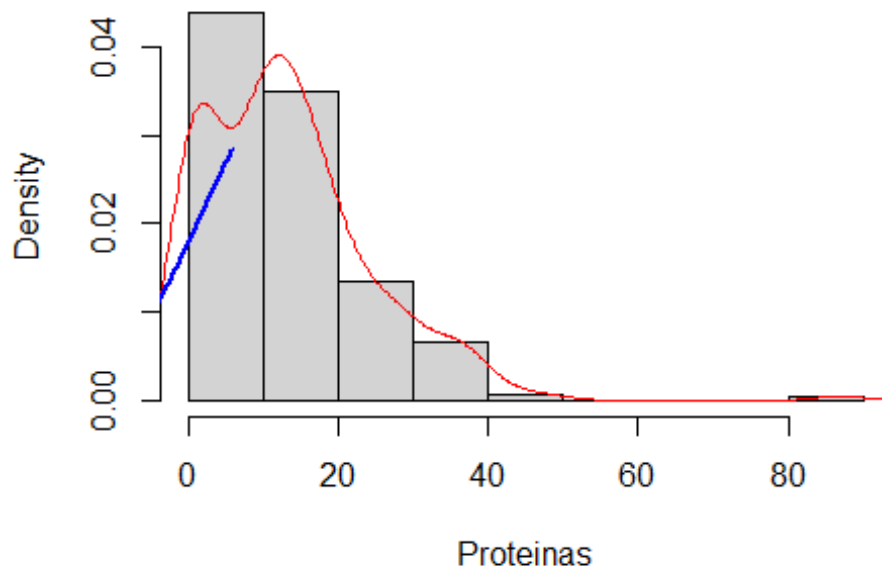


El histograma de las calorías muestra una distribución sesgada hacia la derecha (positivamente sesgada). Aunque parece que se ajusta por algún momento entre los intervalos aproximados de [0,600] en mi interpretación a una distribución normal, no se ajusta completamente a los datos observados. Aunque sigue la tendencia general, hay una desviación significativa en la cola derecha, lo que indica que los datos tienen más valores altos de lo que una distribución normal esperaría.

No Normalidad: El gráfico y las medidas indican que la distribución de las calorías no es normal.

```
# Histograma y distribución teórica para Proteínas
hist(Proteinas, freq = FALSE, main = "Histograma y Distribución Teórica para Proteínas")
lines(density(Proteinas), col = "red")
curve(dnorm(x, mean = mean(Proteinas), sd = sd(Proteinas)),
      from = -6, to = 6, add = TRUE, col = "blue", lwd = 2)
```

Histograma y Distribución Teórica para Proteínas



La curva de distribución normal (en rojo) no se ajusta bien a los datos observados, al menos de forma uniforme y esto es evidente debido a que el histograma de las proteínas muestra una distribución claramente sesgada hacia la derecha (positivamente sesgada), donde la mayoría de los datos están concentrados en el lado izquierdo del histograma, mientras que la distribución normal esperaría una mayor dispersión alrededor de la media.

No Normalidad: El gráfico y las medidas sugieren que la distribución de proteínas no es normal. Esto se evidencia por la falta de ajuste de la curva normal y la diferencia significativa entre la media y la mediana.

Extra, Comparativa con Valores Atípicos y sin valores Atípicos

Para Calorías

Calcular cuartiles y rango intercuartílico (IQR) para Calorías

`q1_cal <- quantile(Calorias, 0.25)` *# Cuantil 1 (Q1)*

`q3_cal <- quantile(Calorias, 0.75)` *# Cuantil 3 (Q3)*

`ri_cal <- IQR(Calorias)` *# Rango intercuartílico (IQR)*

Definir las cotas inferior y superior utilizando 1.5 veces el IQR

`cota_inf_iqr_cal <- q1_cal - 1.5 * ri_cal`

`cota_sup_iqr_cal <- q3_cal + 1.5 * ri_cal`

Calcular la cota superior utilizando 3 desviaciones estándar

`cota_sup_3sd_cal <- mean(Calorias) + 3 * sd(Calorias)`

Identificar y eliminar los datos atípicos utilizando IQR

```

Calorias_iqr <- Calorias[Calorias >= cota_inf_iqr_cal & Calorias <=
cota_sup_iqr_cal]

# Identificar y eliminar los datos atípicos utilizando 3 desviaciones
estándar (sobre el conjunto filtrado por IQR)
Calorias_clean <- Calorias_iqr[Calorias_iqr <= cota_sup_3sd_cal]

# Mostrar el resumen de los datos antes y después de eliminar outliers para
Calorías
cat("Resumen de Calorías original:\n")

## Resumen de Calorías original:

summary(Calorias)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      0.0   210.0   340.0   368.3   500.0   1880.0

cat("\nResumen de Calorías después de eliminar outliers con IQR y 3
desviaciones estándar:\n")

##
## Resumen de Calorías después de eliminar outliers con IQR y 3 desviaciones
estándar:

summary(Calorias_clean)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      0.0   202.5   335.0   349.0   480.0   930.0

# Graficar los boxplots y los histogramas antes y después de eliminar
outliers para Calorías
par(mfrow = c(2, 2)) # Matriz de gráficos de 2x2

# Boxplot antes de eliminar outliers
boxplot(Calorias, horizontal = TRUE, main = "Boxplot Original - Calorías")
abline(v = c(cota_inf_iqr_cal, cota_sup_iqr_cal), col = "red")
abline(v = cota_sup_3sd_cal, col = "blue")

# Boxplot después de eliminar outliers
boxplot(Calorias_clean, horizontal = TRUE, main = "Boxplot sin Outliers -
Calorías")

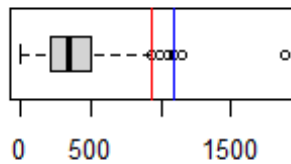
# Histograma antes de eliminar outliers
hist(Calorias, prob = TRUE, col = "lightgray", main = "Histograma Original -
Calorías")
lines(density(Calorias), col = "red")

# Histograma después de eliminar outliers
hist(Calorias_clean, prob = TRUE, col = "lightgray", main = "Histograma sin

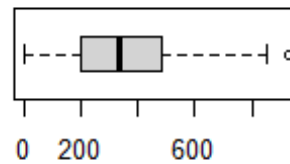
```

```
Outliers - Calorías")
lines(density(Calorias_clean), col = "red")
```

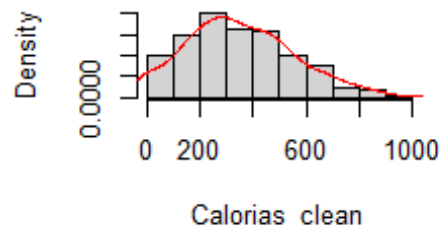
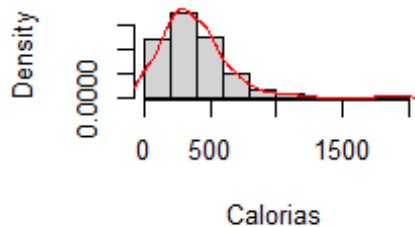
Boxplot Original - Calorías



Boxplot sin Outliers - Caloría



Histograma Original - Caloría Histograma sin Outliers - Calor



Efectivamente, aunque sin duda la eliminacion de los Outliers, muestra una aparente ayuda hacia la normalizacion de los datos, la realidad es que no deberiamos querer forzar una normalizacion, cuando algunos outliers nos aportan sentido y contexto al problema, en mi opinion, seria mejor conservarlos debido a todas las razones que epxlique anteriormente en el primer punto, ya que nos ayuda a visualizar de mejor manera la verdadera naturaleza de los datos, y que ese sesgo a la derecha es parte de ella.

Para Proteínas

Calcular cuartiles y rango intercuartílico (IQR) para Proteínas

```
q1_prot <- quantile(Proteinas, 0.25) # Cuantil 1 (Q1)
```

```
q3_prot <- quantile(Proteinas, 0.75) # Cuantil 3 (Q3)
```

```
ri_prot <- IQR(Proteinas) # Rango intercuartílico (IQR)
```

Definir las cotas inferior y superior utilizando 1.5 veces el IQR

```
cota_inf_iqr_prot <- q1_prot - 1.5 * ri_prot
```

```
cota_sup_iqr_prot <- q3_prot + 1.5 * ri_prot
```

Calcular la cota superior utilizando 3 desviaciones estándar

```
cota_sup_3sd_prot <- mean(Proteinas) + 3 * sd(Proteinas)
```

Identificar y eliminar Los datos atípicos utilizando IQR

```
Proteinas_iqr <- Proteinas[Proteinas >= cota_inf_iqr_prot & Proteinas <=
```

```

cota_sup_iqr_prot]

# Identificar y eliminar los datos atípicos utilizando 3 desviaciones
estándar (sobre el conjunto filtrado por IQR)
Proteinas_clean <- Proteinas_iqr[Proteinas_iqr <= cota_sup_3sd_prot]

# Mostrar el resumen de los datos antes y después de eliminar outliers para
Proteínas
cat("Resumen de Proteínas original:\n")

## Resumen de Proteínas original:

summary(Proteinas)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      0.00   4.00   12.00   13.34   19.00   87.00

cat("\nResumen de Proteínas después de eliminar outliers con IQR y 3
desviaciones estándar:\n")

##
## Resumen de Proteínas después de eliminar outliers con IQR y 3 desviaciones
estándar:

summary(Proteinas_clean)

##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0.0     4.0    12.0    12.8    18.0    40.0

# Graficar los boxplots y los histogramas antes y después de eliminar
outliers para Proteínas
par(mfrow = c(2, 2)) # Matriz de gráficos de 2x2

# Boxplot antes de eliminar outliers
boxplot(Proteinas, horizontal = TRUE, main = "Boxplot Original - Proteínas")
abline(v = c(cota_inf_iqr_prot, cota_sup_iqr_prot), col = "red")
abline(v = cota_sup_3sd_prot, col = "blue")

# Boxplot después de eliminar outliers
boxplot(Proteinas_clean, horizontal = TRUE, main = "Boxplot sin Outliers -
Proteínas")

# Histograma antes de eliminar outliers
hist(Proteinas, prob = TRUE, col = "lightgray", main = "Histograma Original -
Proteínas")
lines(density(Proteinas), col = "red")

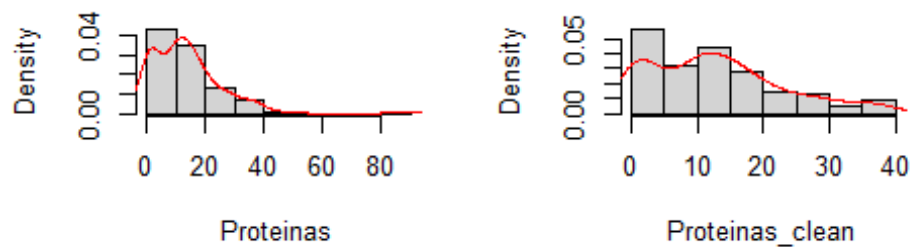
# Histograma después de eliminar outliers
hist(Proteinas_clean, prob = TRUE, col = "lightgray", main = "Histograma sin
Outliers - Proteínas")
lines(density(Proteinas_clean), col = "red")

```

Boxplot Original - Proteínas Boxplot sin Outliers - Proteínas



Histograma Original - Proteínas Histograma sin Outliers - Proteínas



Un perfecto ejemplo de la información que nos aportaban estos outliers, al perderlos y eliminarlos podemos observar como disminuye el sesgo a la derecha del histograma, y aunque tal vez no muestra una normalidad completa, La curva de densidad ahora se ajusta mejor a los datos restantes, una distribución más centrada, pero que tal vez no nos diga mucho de la verdadera naturaleza de los datos completos, misma razón que el punto anterior sigo apoyando el conservarlos para concluir la no normalidad de los mismos.