```
install.packages("dplyr")
install.packages("MASS")
install.packages("e1071")
install.packages("nortest")
install.packages("VGAM")
```

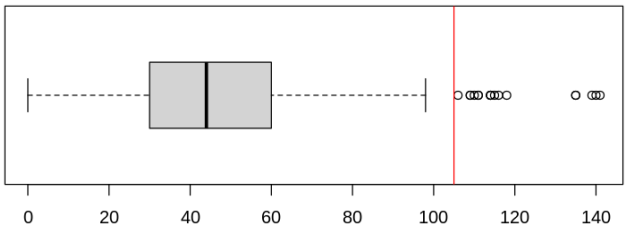
```
Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)

Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)

Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)

also installing the dependency ‘proxy’


Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)

Installing package into ‘/usr/local/lib/R/site-library’
(as ‘lib’ is unspecified)
```

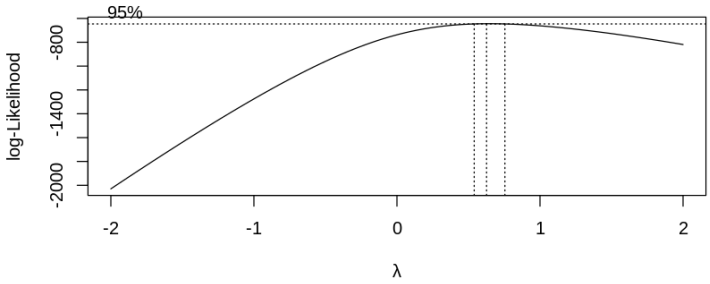
```
library(dplyr)
library(ggplot2)
library(MASS)
library(e1071)
library(nortest)
```



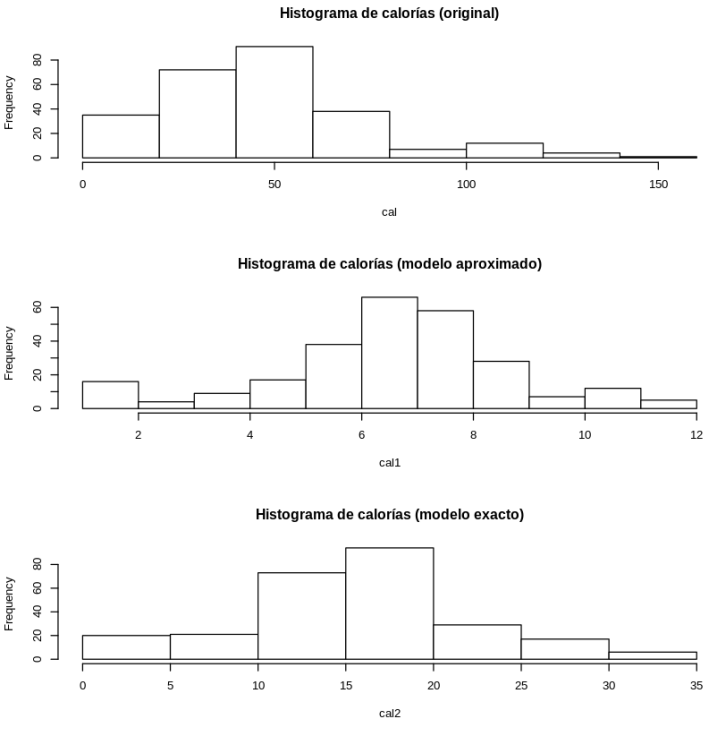
```
{
  result = ((data + 1)^lda - 1)/lda
  return(result)  # Return the result
}

# LOADING A DATASET
data = read.csv("mc-donalds-menu.csv")
cal = data$Carbohydrates

q1c = quantile(cal, 0.25) # First quartile of the "Calories" variable
q3c = quantile(cal, 0.75) # Third quartile of the "Calories" variable
ri_c = IQR(cal) # Interquartile range of "Calories" variable

par(mfrow=c(2,1)) # Create a 2x1 grid for plots
boxplot(cal, horizontal=TRUE)
abline(v=q3c + 1.5*ri_c, col="red")

# BOX-COX TRANSFORMATION
bc = boxcox((cal+1)~1)
lda = bc$x[which.max(bc$y)]
cat("Optimal lambda: ",lda,"\n")
cat("Best transformation equation: ((x-1)^",lda,"-",lda,")/",lda)
```

```
Optimal lambda:  0.6262626
Best transformation equation: ((x-1)^ 0.6262626 - 0.6262626 )/ 0.6262626
```






```
1 # HISTOGRAM COMPARATION
2 cal1 = sqrt(cal + 1)
3 cal2 = BoxCox(cal, lda)
4
5 par(mfrow=c(3,1))
6 hist(cal, col=0, main="Histograma de calorías (original)")
```


```
1 # STATISTICAL SUMMARY
```

```
# STATISTICAL SUMMARY
# Normality test (Anderson-Darling)
D0 = ad.test(cal)
D1 = ad.test(cal1)
D2 = ad.test(cal2)

# Summary
m0 = round(c(as.numeric(summary(cal)),kurtosis(cal),skewness(cal),D0$p.value),5)
m1 = round(c(as.numeric(summary(cal1)),kurtosis(cal1),skewness(cal1),D1$p.value),5)
m2 = round(c(as.numeric(summary(cal2)),kurtosis(cal2),skewness(cal2),D2$p.value),5)

# Print results
m = as.data.frame(rbind(m0,m1,m2))
row.names(m) = c("Original","Primer modelo","Segundo Modelo")
names(m) = c("Minimo","Q1","Mediana","Media","Q3","Máximo","Curtosis","Sesgo","Valor p")
m
print(m$'Valor p')
```

A data.frame: 3 × 9

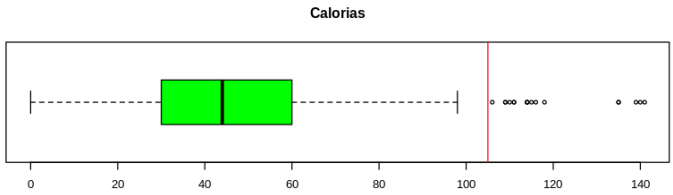
	Minimo	Q1	Mediana	Media	Q3	Máximo	Curtosis	Sesgo	Valor
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl
Original	0	30.00000	44.00000	47.34615	60.00000	141.00000	1.32408	0.90220	
Primer modelo	1	5.56776	6.70820	6.58324	7.81025	11.91638	0.90923	-0.49396	
Segundo Modelo	0	12.11923	15.72485	15.66877	19.36021	33.97793	0.63820	-0.08250	

```
[1] 0 0 0
```

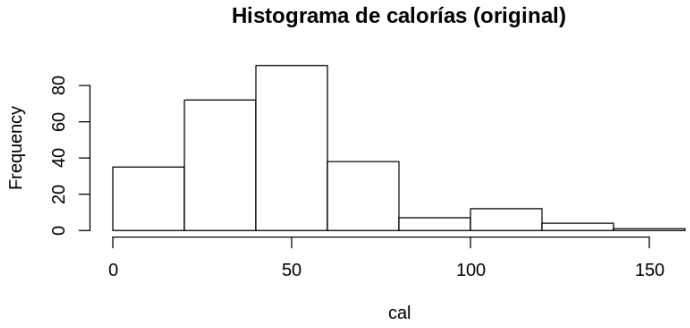
```
# REMOVING OUTLIERS AND ZEROS
threshold_c = q3c + 1.5*ri_c
```

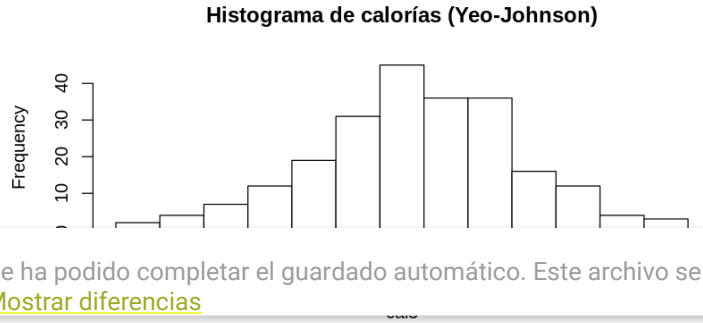


```

par(mfrow=c(3,1)) # Create a 2x1 grid for plots
boxplot(cal, horizontal = TRUE, col="green", main="Calorias")
abline(v=threshold_c, col="red")
boxplot(cal_NoOutliers, horizontal = TRUE, col="yellow", main="Calorias sin outliers")
abline(v=threshold_c, col="red")
hist(cal_NoOutliers, col=0, main="Histograma de calorías (no outliers)")
```




```
cal3 = yeo.johnson(cal_NoOutliers, lambda = lda)
par(mfrow=c(2,1))
hist(cal, col=0, main="Histograma de calorías (original)")
hist(cal3, col=0, main="Histograma de calorías (Yeo-Johnson)")
```





No se ha podido completar el guardado automático. Este archivo se ha actualizado de forma remota o en otra pestaña.
Mostrar diferencias

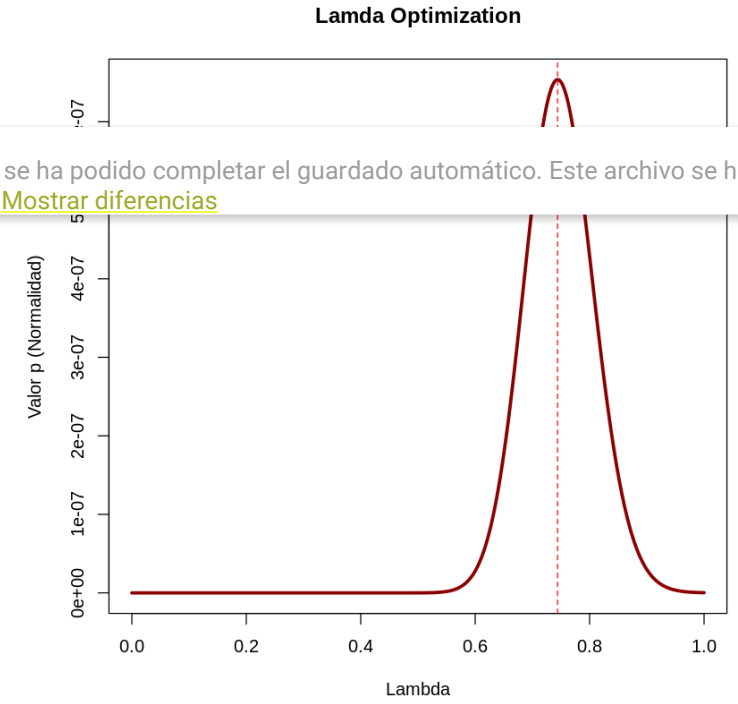
```
# STATISTICAL SUMMARY
# Normality test (Anderson-Darling)
D0 = ad.test(cal)
D3 = ad.test(cal3)

# Summary
m0 = round(c(as.numeric(summary(cal)),kurtosis(cal),skewness(cal),D0$p.value),5)
m3 = round(c(as.numeric(summary(cal3)),kurtosis(cal3),skewness(cal3),D3$p.value),5)

# Print results
M = as.data.frame(rbind(m0,m3))
row.names(M) = c("Original","Modelo Yeo-Johnson")
names(M) = c("Minimo","Q1","Mediana","Media","Q3","Máximo","Curtosis","Sesgo","Valor p")
M
print(M2$`Valor p`)
```

A data.frame: 2 × 9

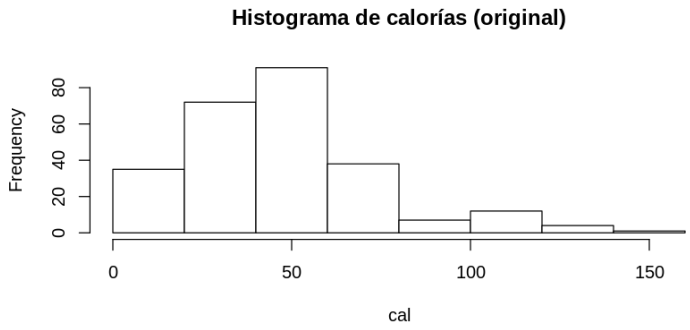
	Minimo	Q1	Mediana	Media	Q3	Máximo	Curtosis	Sesgo	Val
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<

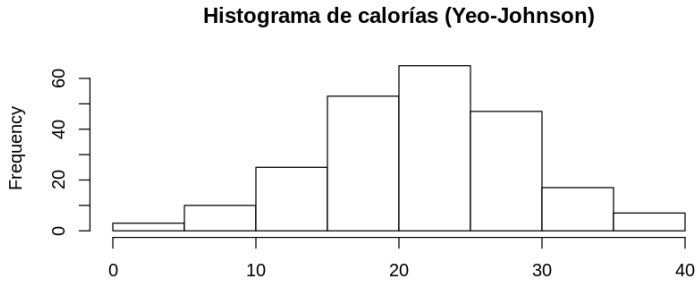
```
lp = seq(0, 1, 0.001) # Proposed lambda values
nlp = length(lp)
n = length(cal)
D = matrix(as.numeric(NA), ncol = 2, nrow = nlp)

for (i in 1:nlp) {
  d = yeo.johnson(cal, lambda = lp[i])
  p = ad.test(d)
  D[i,] = c(lp[i], p$p.value)
}

N = as.data.frame(D)
names(N) <- c("Lambda","Valor-p")
plot(N$Lambda, N$`Valor-p`, type = "l", col = "darkred", lwd = 3,
xlab = "Lambda", ylab = "Valor p (Normalidad)", main="Lamda Optimization")

G = data.frame(subset(N,N$`Valor-p`==max(N$`Valor-p`)))
lda2 = G$Lambda
abline(v = lda2, col = "red", lty = 2)
cat("Optimal lambda: ",lda2,"\n")
```

```
Optimal lambda:  0.744
```




```
cal3 = yeo.johnson(cal_NoOutliers, lambda = lda2)
par(mfrow=c(2,1))
hist(cal, col=0, main="Histograma de calorías (original)")
hist(cal3, col=0, main="Histograma de calorías (Yeo-Johnson)")
```






```
# STATISTICAL SUMMARY
# Normality test (Anderson-Darling)
D0 = ad.test(cal)
D3 = ad.test(cal3)

# Summary
m0 = round(c(as.numeric(summary(cal)),kurtosis(cal),skewness(cal),D0$p.value),5)
m3 = round(c(as.numeric(summary(cal3)),kurtosis(cal3),skewness(cal3),D3$p.value),5)

# Print results
```



```
names(M2) = c("Mínimo","Q1","Mediana","Media","Q3","Máximo","Curtosis","Sesgo","Valor p")
M2
print(M2$`Valor p`)
```

A data.frame: 2 × 9

	Minimo	Q1	Mediana	Media	Q3	Máximo	Curtosis	Sesgo	Va
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	
Original	0.00000	30.00000	44.00000	47.34615	60.00000	141.0000	1.32408	0.90220	0
Modelo Yeo-Johnson 2	3.10695	16.77772	21.48125	21.56740	26.22469	39.6933	-0.01462	-0.03104	0

```
[1] 0.00000 0.82772
```

La mejor transformación de los datos de acuerdo a las características de los modelos que encontraste:

• La mejor transformación de acuerdo los valores-p obtenidos para cada distribución fue la transformación Yeo-Johnson, para el caso de los carbohidratos de la base de datos.

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

Ventajas de modelo Cox-Box:

• Es muy simple

• Es fácil de interpretar

Desventajas de modelo Cox-Box:

• No acepta valores negativos

• No acepta valores ceros

Ventajas de modelo Yeo-Johnson:

• Acepta valores ceros y negativos

• Es un modelo robusto

Desventajas de modelo Yeo-Johnson:

• Es más complejo de programar

• Se debe estimar los parámetros cuidadosamente

Diferencias entre la transformación y el escalamiento de los datos:

• Cambios en la Distribución: Mientras que las transformaciones pueden hacer que una distribución no normal se aproxime a la normalidad, el escalamiento no cambia la forma de la distribución ni su normalidad.

• Una transformación puede convertir una relación lineal en una relación no lineal con sus variables y viceversa. El escalamiento no afecta la relación entre las variables.

• Las transformaciones como Yeo-Johnson o Box-Cox pueden manejar o no valores negativos y ceros (se requiere hacer un ajustamiento). El escalamiento no modifica la naturaleza de los valores negativos.

Cuándo Utilizar Cada Uno:

• Cuando se desee ajustar los datos a una escala determinada y se requieren mantener las relaciones relativas entre las variables, lo mejor será usar un escalamiento de datos.

✓ 0 s completado a las 20:57

No se ha podido completar el guardado automático. Este archivo se ha actualizado de forma remota o en otra pestaña.
Mostrar diferencias