Actividad 3: Transformaciones

Jorge Eduardo de León Reyna - A00829759

```
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)
```

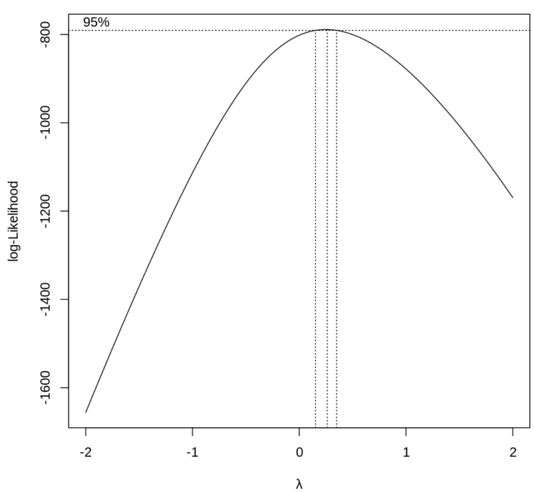
Importando datos

```
data = read.csv("/content/mc-donalds-menu-1.csv")
x = data[["Sugars"]]
```

1. Utiliza la transformación Box-Cox. Utiliza el modelo exacto y el aproximado de acuerdo con las sugerencias de Box y Cox para la transformación

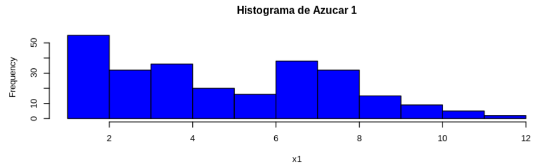
```
library(MASS)
bc = boxcox((x+1)~1)
bc$x[which.max(bc$y)]
```

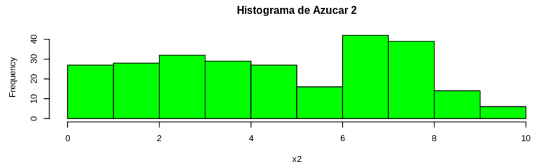
0.262626262626263



Se obtiene que el valor optimo para lambda = 0.2626

```
x1 = sqrt(x + 1)
l = 0.2626
x2 = ((x + 1)^l - 1) / l
par(mfrow = c(3, 1))
hist(x1, col = "blue", main = "Histograma de Azucar 1")
hist(x2, col = "green", main = "Histograma de Azucar 2")
hist(x, col = "red", main = "Histograma de Azucar")
```





2. Escribe las ecuaciones de los modelos encontrados.

$$x_{aproximada} = \sqrt{x+1}$$
$$x_{exacta} = (x+1)^{0.2626} - \frac{1}{0.2626}$$

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

1. Compara las medidas: Mínimo, máximo, media, mediana, cuartil 1 y cuartil 3, sesgo y curtosis.
2. Obten el histograma de los 2 modelos obtenidos (exacto y aproximado) y los datos originales.
3. Realiza la prueba de normalidad de Anderson-Darling o de Jarque Bera para los datos transformados y los originales

Prueba de normalidad

```
D0=ad.test(x)
D1=ad.test(x1)
D2=ad.test(x2)
```

Resumen de medidas

```
m0=round(c(as.numeric(summary(x)),kurtosis(x),skewness(x),D0$p.value),3)
m1=round(c(as.numeric(summary(x1)),kurtosis(x1),skewness(x1),D1$p.value),3)
m2=round(c(as.numeric(summary(x2)),kurtosis(x2),skewness(x2),D2$p.value),3)
```

Tabla comparativa

```
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
```

A data.frame: 3 × 9

	Minimo	Q1	Mediana	Media	Q3	Máximo	Curtosis	Sesgo	Valor p
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Original	0	5.750	17.500	29.423	48.000	128.000	0.461	1.020	0
Primer modelo	1	2.597	4.301	4.825	7.000	11.358	-1.014	0.279	0
Segundo Modelo	0	2.477	4.385	4.519	6.774	9.836	-1.113	-0.106	0

```
D0
D1
D2
```

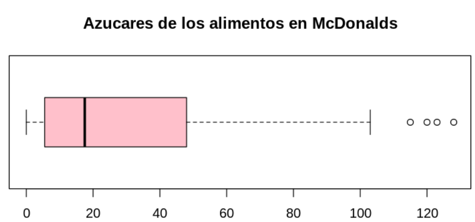
```
        Anderson-Darling normality test

data:  x
A = 9.9899, p-value < 2.2e-16

        Anderson-Darling normality test

data:  x1
A = 4.0816, p-value = 3.531e-10

        Anderson-Darling normality test

data:  x2
A = 3.3722, p-value = 1.857e-08
```

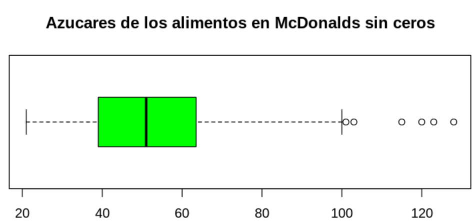
4. Detecta anomalías y corrige tu base de datos (datos atípicos, ceros anámalos, etc).

Se identifica una alta frecuencia en el rango de valores cercano a 0 que pueden afetctar al modelo por lo que en base a lo observado en el historgrama se eliminaran los valores de dicho rango para facilitar el normalizacion de los datos.

Boxplot comparativo eliminando 0s

```
data_1 = subset(data,data$Sugars>20)
x_clean = data_1[["Sugars"]]
par(mfrow=c(2,1))
boxplot(x, horizontal = TRUE,col="pink", main="Azucares de los alimentos en McDonalds")
boxplot(x_clean, horizontal = TRUE,col="green", main="Azucares de los alimentos en McDonalds sin ceros")

```

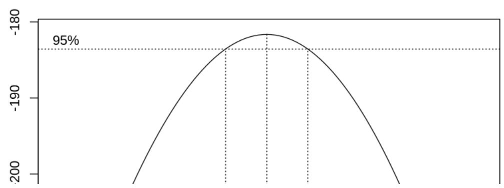




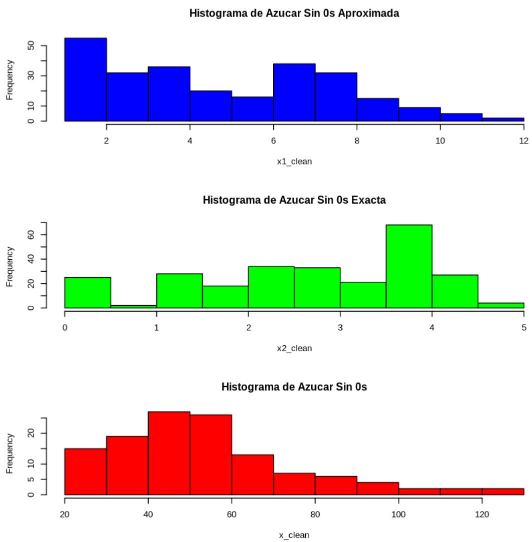
Box-cox quitando 0s

```
bc_clean = boxcox((x_clean+1)~1)
optimal_lambda_index <- which.max(bc_clean$y)
optimal_lambda <- bc_clean$x[optimal_lambda_index]
optimal_lambda
```

-0.0202020202020201




```
x1_clean = sqrt(x + 1)
l = -0.0202020202020201
x2_clean = ((x + 1)^l - 1) / l
par(mfrow = c(3, 1))
hist(x1_clean, col = "blue", main = "Histograma de Azucar Sin 0s Aproximada")
hist(x2_clean, col = "green", main = "Histograma de Azucar Sin 0s Exacta")
hist(x_clean, col = "red", main = "Histograma de Azucar Sin 0s")
```



Prueba de normalidad

```
D0_clean =ad.test(x_clean)
D1_clean =ad.test(x1_clean)
D2_clean =ad.test(x2_clean)
```

Resumen de medidas

```
m0=round(c(as.numeric(summary(x_clean)),kurtosis(x_clean),skewness(x_clean),D0_clean$p.value),3)
m1=round(c(as.numeric(summary(x1_clean)),kurtosis(x1_clean),skewness(x1_clean),D1_clean$p.value),3)
m2=round(c(as.numeric(summary(x2_clean)),kurtosis(x2_clean),skewness(x2_clean),D2_clean$p.value),3)
```

Tabla comparativa

```
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
```

A data.frame: 3 × 9

	Minimo	Q1	Mediana	Media	Q3	Máximo	Curtosis	Sesgo	Valor p
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Original	21	39.000	51.000	54.407	63.500	128.000	0.984	1.045	0
Primer modelo	1	2.597	4.301	4.825	7.000	11.358	-1.014	0.279	0
Segundo Modelo	0	1.871	2.833	2.664	3.743	4.629	-0.609	-0.636	0

```
D0_clean
D1_clean
D2_clean
```

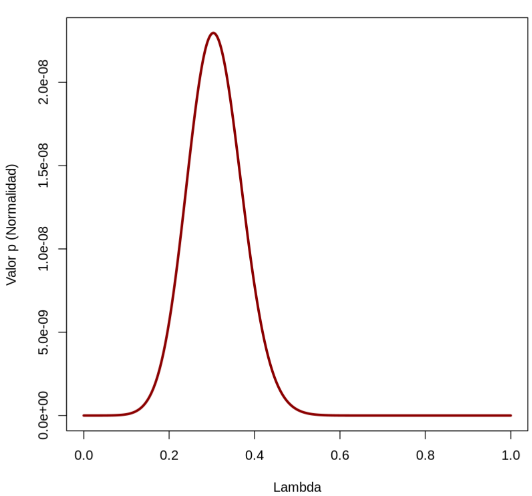
```
        Anderson-Darling normality test

data:  x_clean
A = 2.3631, p-value = 5.15e-06

        Anderson-Darling normality test

data:  x1_clean
A = 4.0816, p-value = 3.531e-10

        Anderson-Darling normality test

data:  x2_clean
A = 6.0816, p-value = 5.492e-15
```

5. Utiliza la transformación de Yeo Johnson y encuentra el valor de lambda que maximiza el valor p de la prueba de normalidad que hayas utilizado (Anderson-Darling o Jarque Bera).

```
library(VGAM)
x_yeo = yeo.johnson(x_clean, lambda = 1)
```

```
lp <- seq(0,1,0.001) # Valores de lambda propuestos
nlp <- length(lp)
n=length(x)
D <- matrix(as.numeric(NA),ncol=2,nrow=nlp)
d <-NA
for (i in 1:nlp){
  d= yeo.johnson(x, lambda = lp[i])
  p=ad.test(d)
  D[i,]=c(lp[i],p$p.value)}
```

```
N=as.data.frame(D)
plot(N$V1,N$V2,
type="l",col="darkred",lwd=3,
xlab="Lambda",
ylab="Valor p (Normalidad)")
```



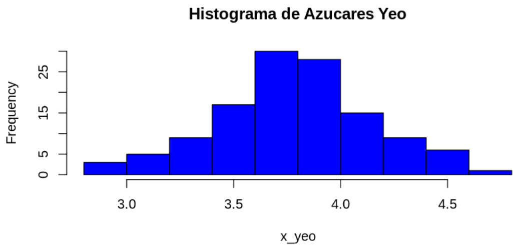

```
G=data.frame(subset(N,N$V1==max(N$V1)))
G
```

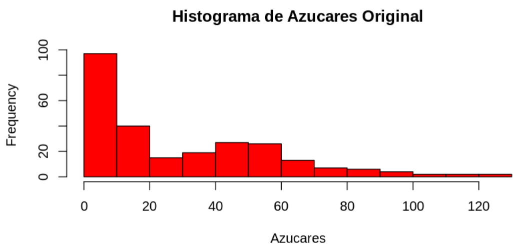
A data.frame: 1 × 2

6. Escribe la ecuación del modelo encontrado.

$$x_{exacta} = (x+1)^{3.7e-24} - \frac{1}{3.7e-24}$$

```
par(mfrow=c(2,1))
hist(x_yeo,col="blue",main="Histograma de Azucares Yeo")
hist(x,col="red",main="Histograma de Azucares Original",xlab="Azucares")
```





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

1. Compara las medidas: Mínimo, máximo, media, mediana, cuartil 1 y cuartil 3, sesgo y curtosis.
2. Obten el histograma de los 2 modelos obtenidos (exacto y aproximado) y los datos originales.
3. Realiza la prueba de normalidad de Anderson-Darling para los datos transformados y los originales

```
D0_clean =ad.test(x)
D1_clean =ad.test(x_yeo)
```

```
m0=round(c(as.numeric(summary(x)),kurtosis(x),skewness(x),D0_clean$p.value),3)
m1=round(c(as.numeric(summary(x_yeo)),kurtosis(x_yeo),skewness(x_yeo),D1_clean$p.value),3)
```

```
m_yeo = as.data.frame(rbind(m0,m1))
row.names(m_yeo)=c("Original","Yeo")
names(m_yeo)=c("Minimo","Q1","Mediana","Media","Q3","Máximo","Curtosis","Sesgo","Valor p")
m
m_yeo
```

A data.frame: 3 × 9

	Minimo	Q1	Mediana	Media	Q3	Máximo	Curtosis	Sesgo	Valor p
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Original	21	39.000	51.000	54.407	63.500	128.000	0.984	1.045	0
Primer modelo	1	2.597	4.301	4.825	7.000	11.358	-1.014	0.279	0
Segundo Modelo	0	1.871	2.833	2.664	3.743	4.629	-0.609	-0.636	0

A data.frame: 2 × 9

	Minimo	Q1	Mediana	Media	Q3	Máximo	Curtosis	Sesgo	Valor p
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
Original	0.000	5.750	17.500	29.423	48.000	128.000	0.461	1.020	0.000
Yeo	2.997	3.555	3.798	3.783	3.996	4.629	-0.294	0.002	0.776

8. Define la mejor transformación de los datos de acuerdo a las características de los modelos que encontraste.

Consideraciones iniciales:

1. Para ambas transformaciones se eliminaron los datos menores a 0 al provocar varaciones importantes en la distribucion de la muestra como se puede observar en los histogramas.
2. Para la transformacion Box-Cox se llevaron a cabo 2 versiones del mismo, donde se vario el valor de Lambda(L) de manera que en una de ellas se utilizo un valor L = 1 y en la otra se utilizo la funcion de Box-Cox para encontrar la L ideal para lograr una distribucion normal.

Observaciones iniciales: Despues de llevar a cabo la transformacion Box-Cox y de Yeo Johnson se evaluó cada una de ellas deacuerdo a la prueba de normalidad Anderson-Darling donde se observó lo siguiente:

1. Al eliminar los valores iguales o menores a 0 del dataset inicial se mejoró considerablmente la distribución normal en ambas transformaciones, esto basado en la representacion grafica de las distribuciones y en la curtosis, el sesgo y el valor P resultante.
2. Respecto a las dos iteraciones hechas en la transformacion Box-Cox se encontro que pese a que una de ellas presenta un sesgo mas cercano a 0, su curtosis no da un resultado deseado por lo que ambas iteraciones presentan ventajas y desventajas acorde a estos valores ademas de que en su representacion mediante el histograma se puede ver que pese a que mejora respecto a los datos originales, no se logra una distribucion normal ideal.
3. En cuanto a la transformacion de Yeo Johnson, se logra observar una distribucion normal mas cercana a lo buscado en el histograma de la misma, conclusión que se ve apoyada por el valor del Sesgo donde este es mucho mas cercana en 0 en comparación a los valores obtenidos en la transformacion de Box-Cox junto con que el valor P es mucho mayor a comparacion de los demas, lo cual es lo esperado.

Conclusion: En conclusión se observa que *la transformación de Yeo-Johnson arroja mejores resultados en cuanto a la busqueda de normalidad de los datos.* Esta conclusión se ve apoyada por:

1. Un sesgo mucho mas cercano a 0 en comparacion a las otras transformaciones.
2. Un histograma balanceado donde a simple vista se puede observar una distribución normal aproximada.
3. Una curtosis mas acercada al valor deseado en comparacion a las otras transformaciones.
4. Al investigar sobre los beneficios de cada transformación, se encuentra que esta transformacion funciona mejor en muestras con valores 0 o cercanos a 0, comportamiento que presenta la muestra analizada.

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

Ventajas de transformacion Box-Cox:

1. Simplicidad: Solo requiere un parámetro lambda para ajustar la transformación.
2. Interpretación: Puede tener una interpretación más intuitiva en ciertos casos.
3. Resultados estables: La transformación Box-Cox tiende a funcionar bien y a producir resultados numéricamente estables en la mayoría de los casos.

Desventajas de la transformación de Box-Cox:

1. Requisitos de positividad: No puede usarse con datos que contengan valores negativos o cero.
2. Sensibilidad a valores atípicos: Puede ser sensible a valores atípicos, lo que puede influir en la elección del parámetro lambda y afectar la transformación.
3. Limitación en valores cercanos a cero: Si los datos contienen valores cercanos a cero, la transformación Box-Cox puede generar resultados inestables.

Ventajas de transformacion Yeo-Johonson

1. Flexibilidad: La transformación de Yeo-Johnson puede utilizarse con datos que contienen valores negativos y cero.
2. Menos sensibilidad a valores atípicos: Es menos sensible a valores atípicos en comparación con la transformación de Box-Cox, lo que puede resultar en transformaciones más robustas.
3. Mayor rango de aplicabilidad: La transformación de Yeo-Johnson puede manejar una gama más amplia de distribuciones de datos.

Desventajas de transformacion Yeo-Johonson:

1. Complejidad: La transformación de Yeo-Johnson involucra cálculos más complejos que la transformación de Box-Cox.
2. Interpretación menos intuitiva: A diferencia de la transformación de Box-Cox, la transformación de Yeo-Johnson puede carecer de una interpretación clara y directa en términos de potencia.

10. Analiza las diferencias entre la transformación y el escalamiento de los datos:

1. Escribe al menos 3 diferencias entre lo que es la transformación y el escalamiento de los datos

1. Objetivo: El objetivo principal de la transformación de datos es modificar la distribución o la estructura de los datos para hacerlos más adecuados para ciertos análisis o modelos, por otro lado, el objetivo del escalamiento de datos es ajustar las magnitudes de los valores presentes para que tengan un rango similar o estén en una escala específica.
2. Tipo de Operación: La transformación implica aplicar operaciones matemáticas a los valores originales de los datos, como logaritmos, raíces, potencia. Por otro lado, el escalamiento implica reescalar los valores para que se ajusten a una escala específica, como [0, 1] o una media de 0 y una desviación estándar de 1.
3. Efecto en la Distribución: La Transformación de Datos puede cambiar la forma y la estructura de la distribución original de los datos, por otro lado, el escalamiento de datos no cambia la forma o la estructura de la distribución de los datos, solo ajusta sus magnitudes manteniendo la relación relativa entre ellos.

2. Indica cuándo es necesario utilizar cada uno

1. Transformación de Datos:

- Cuando los datos no siguen una distribución normal.
- Cuando se desea reducir la influencia de valores atípicos.
- Cuando se busca estabilizar la varianza en series de tiempo u otros contextos donde la variabilidad cambia.

2. Escalamiento de Datos:

- Cuando se están utilizando algoritmos basados en distancias o magnitudes, como algoritmos de clustering (k-means) y métodos de reducción de dimensionalidad (PCA).
- Cuando se desea evitar que las valores con magnitudes más grandes dominen el proceso de aprendizaje en modelos de machine learning como regresiones y redes neuronales.

```
1
```

✓ 0 s se ejecutó 12:29