5. Transformaciones

Ozner Leyva

2024-08-14

```
M=read.csv("C:/Users/ozner/Downloads/mc-donalds-menu.csv") #Leer la base de datos
carb=M$Carbohydrates

library(readr)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(MASS)

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

library(nortest)
library(moments)
library(ggplot2)
library(car)

## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

Función para calcular estadísticas descriptivas

```
calcular_estadisticas <- function(x) {
  c(
    Mínimo = min(x, na.rm = TRUE),
    Máximo = max(x, na.rm = TRUE),
    Media = mean(x, na.rm = TRUE),
    Mediana = median(x, na.rm = TRUE),
    Cuartil_1 = quantile(x, 0.25, na.rm = TRUE),
    Cuartil_3 = quantile(x, 0.75, na.rm = TRUE),
    Sesgo = skewness(x, na.rm = TRUE),
    Curtosis = kurtosis(x, na.rm = TRUE)
  )
}
```

Calcular estadísticas para los datos originales

```
stats_originales <- calcular_estadisticas(carb)
print("Estadísticas de los datos originales:")
```

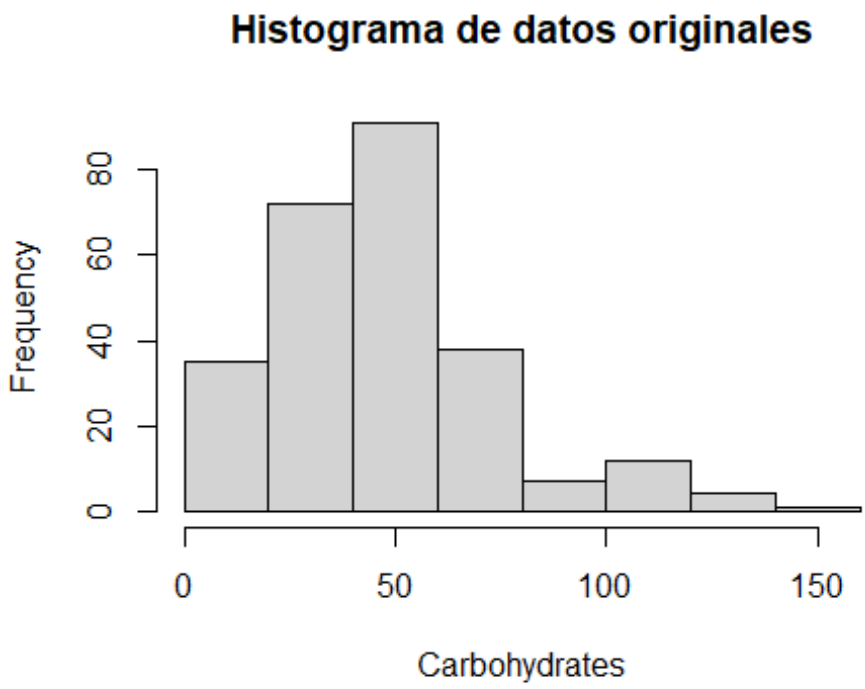
```
## [1] "Estadísticas de los datos originales:"
```

```
print(stats_originales)
```

```
##        Mínimo        Máximo         Media       Mediana Cuartil_1.25%
##     0.0000000   141.0000000    47.3461538    44.0000000    30.0000000
## Cuartil_3.75%         Sesgo      Curtosis
##    60.0000000     0.9074253     4.3575379
```

Histograma de los datos originales

```
hist(carb, main = "Histograma de datos originales", xlab =
"Carbohydrates")
```



Prueba de normalidad Anderson-Darling para los datos originales

```
ad_test_original <- ad.test(carb)
print("Prueba de normalidad Anderson-Darling para los datos originales:")
```

```
## [1] "Prueba de normalidad Anderson-Darling para los datos originales:"
```

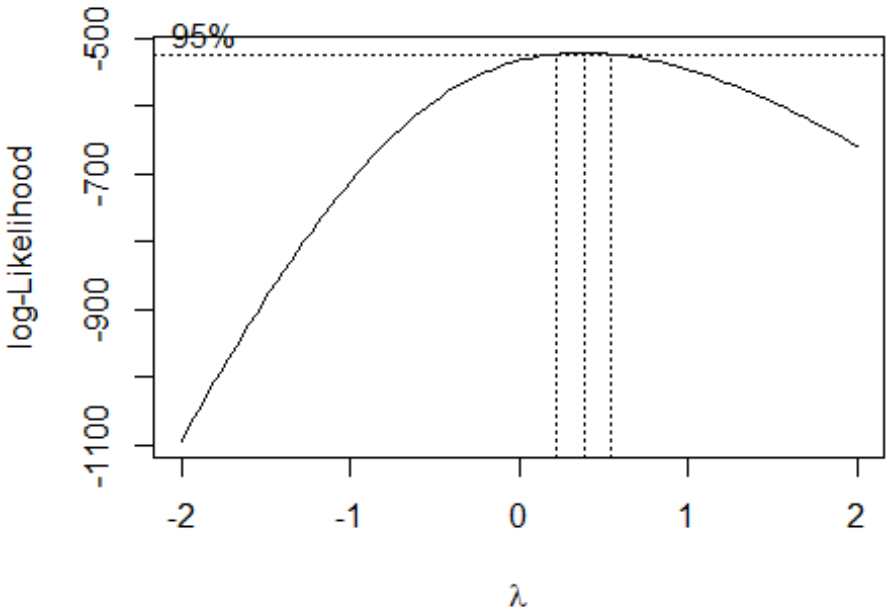
```
print(ad_test_original)
```

```
##
##  Anderson-Darling normality test
##
## data:  carb
## A = 4.1402, p-value = 2.547e-10
```

Transformación Box-Cox

```
# Asegurarse de que todos los valores sean positivos
# Vamos a quitar todos los os ya que son puras sodas de dieta y té
variable_seleccionada <- carb[carb != 0]

bc <- boxcox(variable_seleccionada ~ 1)
```




```
lambda_exacto <- bc$x[which.max(bc$y)]
lambda_aprox <- round(lambda_exacto * 4) / 4

# Aplicar transformaciones
bc_exacto <- if (lambda_exacto != 0) (variable_seleccionada^lambda_exacto
- 1) / lambda_exacto else log(variable_seleccionada)
bc_aprox <- if (lambda_aprox != 0) (variable_seleccionada^lambda_aprox -
1) / lambda_aprox else log(variable_seleccionada)

# Ecuaciones de los modelos
cat("Ecuación del modelo exacto: Y =", ifelse(lambda_exacto != 0,
paste("(X^", lambda_exacto, " - 1) /", lambda_exacto), "log(X)"), "\n")

## Ecuación del modelo exacto: Y = (X^ 0.383838383838384  - 1) /
0.383838383838384

cat("Ecuación del modelo aproximado: Y =", ifelse(lambda_aprox != 0,
paste("(X^", lambda_aprox, " - 1) /", lambda_aprox), "log(X)"), "\n")

## Ecuación del modelo aproximado: Y = (X^ 0.5  - 1) / 0.5
```

Calcular estadísticas para las transformaciones

```
stats_bc_exacto <- calcular_estadisticas(bc_exacto)
stats_bc_aprox <- calcular_estadisticas(bc_aprox)

print("Estadísticas de la transformación Box-Cox exacta:")
```

```
## [1] "Estadísticas de la transformación Box-Cox exacta:"
```

```
print(stats_bc_exacto)
```

```
##        Mínimo        Máximo         Media       Mediana Cuartil_1.25%
##    1.83026491   14.80496204    8.76261415    8.72087557    7.48011043
## Cuartil_3.75%         Sesgo      Curtosis
##   10.01678162    0.06961439    3.66808455
```

```
print("Estadísticas de la transformación Box-Cox aproximada:")
```

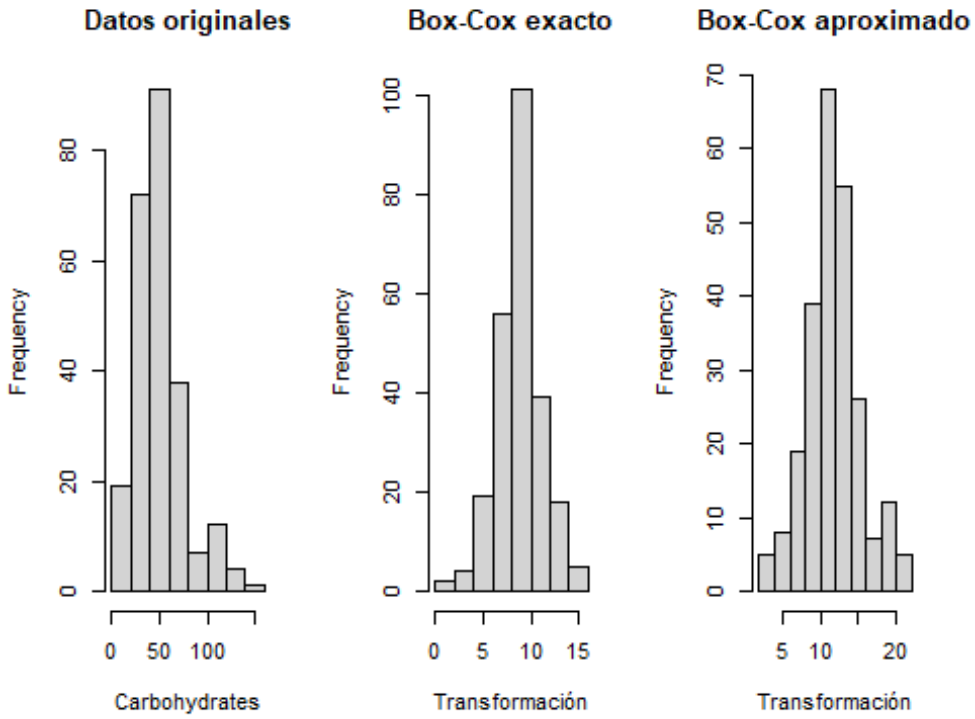
```
## [1] "Estadísticas de la transformación Box-Cox aproximada:"
```

```
print(stats_bc_aprox)
```

```
##        Mínimo        Máximo         Media       Mediana Cuartil_1.25%
##     2.0000000    21.7486842    11.7421600    11.5646600     9.6619038
## Cuartil_3.75%         Sesgo      Curtosis
##    13.6204994     0.3203003     3.6321662
```

Histogramas

```
par(mfrow = c(1, 3))
hist(variable_seleccionada, main = "Datos originales", xlab =
"Carbohydrates")
hist(bc_exacto, main = "Box-Cox exacto", xlab = "Transformación")
hist(bc_aprox, main = "Box-Cox aproximado", xlab = "Transformación")
```



Pruebas de normalidad

```
ad_test_bc_exacto <- ad.test(bc_exacto)
ad_test_bc_aprox <- ad.test(bc_aprox)
ad_test_original <- ad.test(variable_seleccionada)

print("Prueba de normalidad Anderson-Darling para Box-Cox exacto:")
```

```
## [1] "Prueba de normalidad Anderson-Darling para Box-Cox exacto:"
```

```
print(ad_test_bc_exacto)
```

```
##
##  Anderson-Darling normality test
##
## data:  bc_exacto
## A = 1.4743, p-value = 0.000818
```

```
print("Prueba de normalidad Anderson-Darling para Box-Cox aproximado:")
```

```
## [1] "Prueba de normalidad Anderson-Darling para Box-Cox aproximado:"
```

```
print(ad_test_bc_aprox)
```

```
##
##  Anderson-Darling normality test
##
## data:  bc_aprox
## A = 1.7716, p-value = 0.0001518
```

```
print("Prueba de normalidad Anderson-Darling para Datos Originales:")
```

```
## [1] "Prueba de normalidad Anderson-Darling para Datos Originales:"
```

```
print(ad_test_original)
```

```
##
##  Anderson-Darling normality test
##
## data:  variable_seleccionada
## A = 5.9462, p-value = 1.149e-14
```

Detección de anomalías

```
Q1 <- quantile(variable_seleccionada, 0.25)
Q3 <- quantile(variable_seleccionada, 0.75)
IQR <- Q3 - Q1
lower_bound <- Q1 - 1.5 * IQR
upper_bound <- Q3 + 1.5 * IQR

# Corregir anomalías
variable_corregida <- ifelse(variable_seleccionada < lower_bound,
```

```
lower_bound,
                                ifelse(variable_seleccionada > upper_bound,
upper_bound,
                                       variable_seleccionada))

# Actualizar la variable seleccionada
variable_seleccionada <- variable_corregida

# Función para calcular el valor p de Anderson-Darling
ad_p_value <- function(lambda, x) {
  transformed <- if (lambda != 0) (x^lambda - 1) / lambda else log(x)
  return(ad.test(transformed)$p.value)
}

# Encontrar el mejor lambda para Yeo-Johnson
yj_lambda <- optimize(ad_p_value, c(-2, 2), x = variable_seleccionada,
maximum = TRUE)$maximum

# Aplicar transformación Yeo-Johnson
yj_transform <- if (yj_lambda != 0) (variable_seleccionada^yj_lambda - 1)
/ yj_lambda else log(variable_seleccionada)

# Ecuación del modelo Yeo-Johnson
cat("Ecuación del modelo Yeo-Johnson: Y =", ifelse(yj_lambda != 0,
paste("(X^", yj_lambda, " - 1) /", yj_lambda), "log(X)"), "\n")
```

```
## Ecuación del modelo Yeo-Johnson: Y = (X^ 0.450819220791466  - 1) /
0.450819220791466
```

Calcular estadísticas para Yeo-Johnson

```
stats_yj <- calcular_estadisticas(yj_transform)
print("Estadísticas de la transformación Yeo-Johnson:")
```

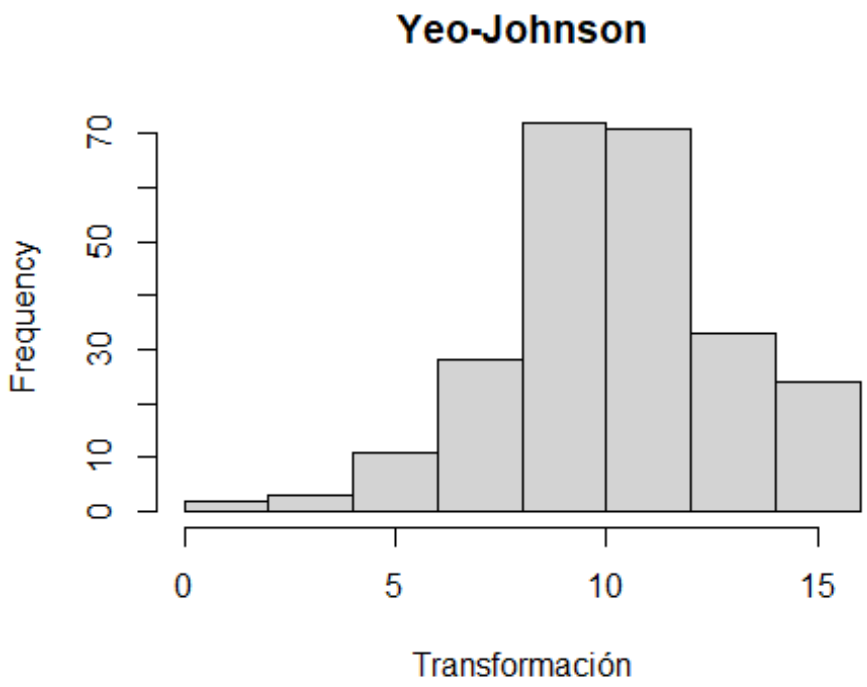
```
## [1] "Estadísticas de la transformación Yeo-Johnson:"
```

```
print(stats_yj)
```

```
##        Mínimo        Máximo         Media       Mediana Cuartil_1.25%
##     1.9257973    15.5871872    10.2572618    10.2441895     8.6565266
## Cuartil_3.75%         Sesgo      Curtosis
##    11.9351443    -0.1459197     3.2157502
```

Histograma

```
hist(yj_transform, main = "Yeo-Johnson", xlab = "Transformación")
```



Prueba de normalidad

```
ad_test_yj <- ad.test(yj_transform)
print("Prueba de normalidad Anderson-Darling para Yeo-Johnson:")

## [1] "Prueba de normalidad Anderson-Darling para Yeo-Johnson:"

print(ad_test_yj)

##
##  Anderson-Darling normality test
##
## data:  yj_transform
## A = 0.99151, p-value = 0.0127
```

Ventajas y desventajas de Box-Cox y Yeo-Johnson:

Box-Cox: Ventajas: Bien establecido y ampliamente utilizado. Eficaz para muchos tipos de datos. Desventajas: Solo funciona con datos positivos. Puede ser sensible a valores atípicos.

Yeo-Johnson: Ventajas: Puede manejar datos negativos y cero. Desventajas: Puede ser computacionalmente más intensivo.

Diferencias entre transformación y escalamiento de datos:

1. Tres diferencias principales:

La transformación cambia la forma de la distribución, mientras que el escalamiento solo cambia la escala.

La transformación puede afectar las relaciones no lineales entre variables, el escalamiento no.

La transformación puede hacer que los datos sean más interpretables en términos de la distribución normal, el escalamiento no afecta la normalidad.

2. Cuándo usar cada uno:

Transformación: Cuando se busca normalizar los datos, estabilizar la varianza, o linealizar relaciones.

Escalamiento: Cuando se necesita comparar o combinar variables con diferentes unidades o rangos.