M1_A5

Sofia Cantu

2024-08-14

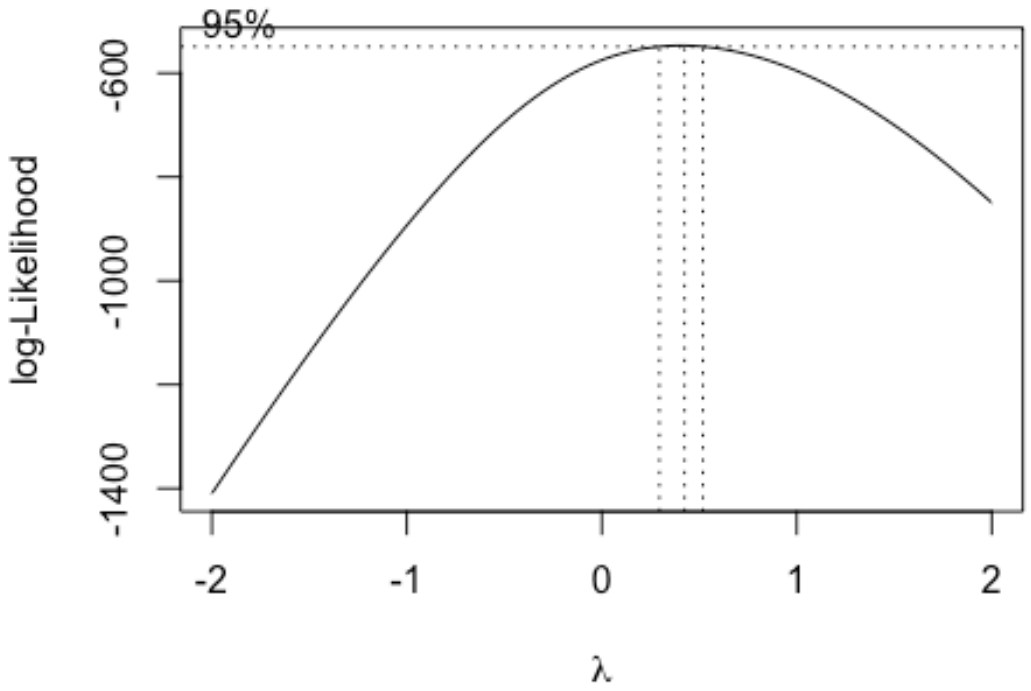
```r
# M de Menu
M = read.csv("~/Downloads/ArchivosCodigos/mc-donalds-menu.csv")

# Selección de variable, que no sea Calorías (escogi Total Fat)
selected_var <- M$Total.Fat
selected_var <- selected_var[is.finite(selected_var) & !is.na(selected_var)]

#Al ser solo un 0, lo elimine.
selected_var <- selected_var[selected_var > 0]
summary(selected_var)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.50    8.00   16.00   17.45   23.50  118.00
```

```r
any(selected_var <= 0)
```

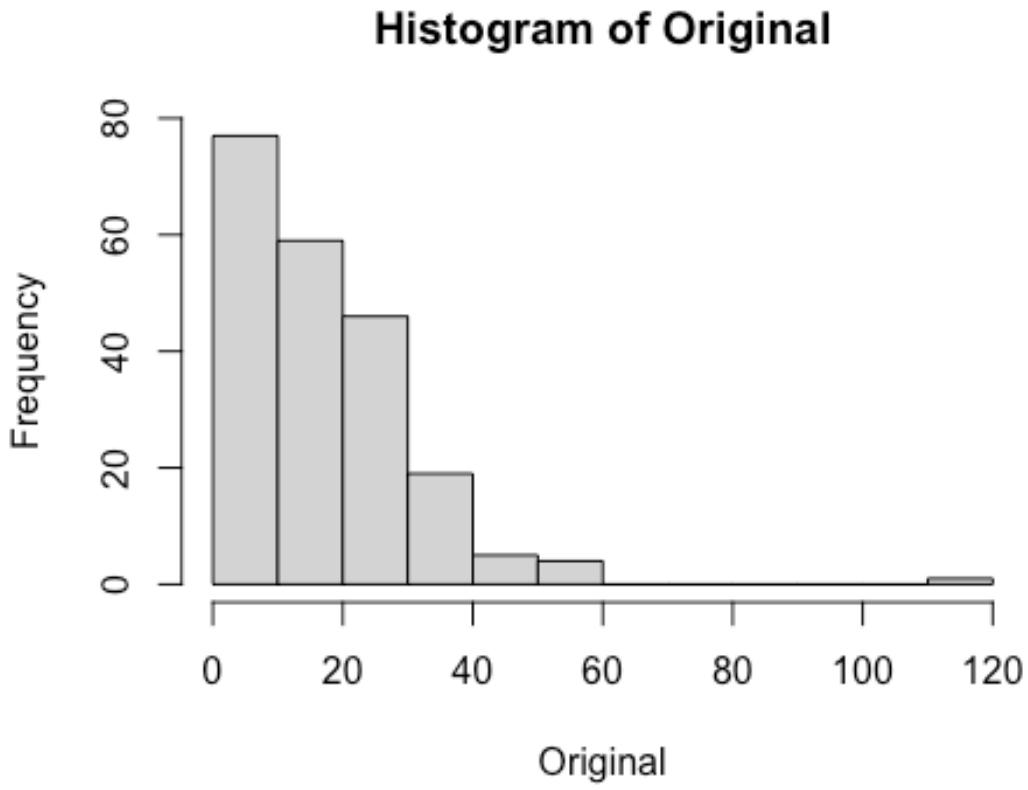
```
## [1] FALSE
```

```r
# 1. Box-Cox
bc <- boxcox(selected_var ~ 1)
```

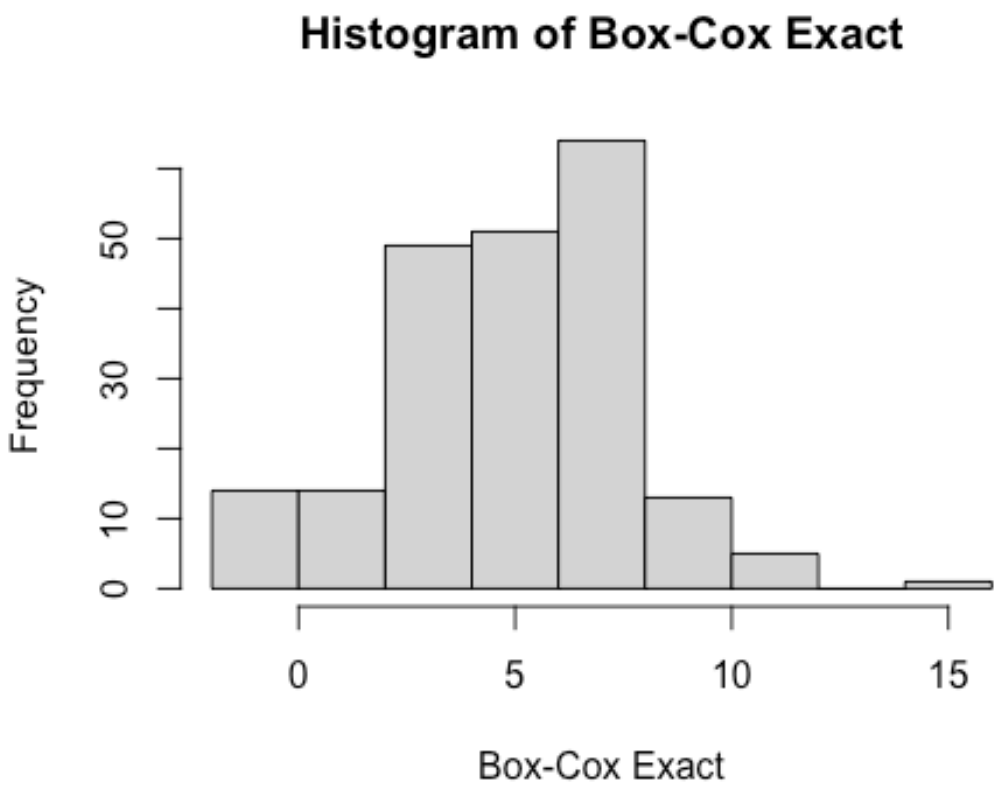



```r
lambda_exact <- bc$x[which.max(bc$y)]
lambda_approx <- round(lambda_exact * 2) / 2

bc_exact <- (selected_var^lambda_exact - 1) / lambda_exact
bc_approx <- (selected_var^lambda_approx - 1) / lambda_approx

# 2. Equaciones
cat("Exact model: y = (x^", lambda_exact, " - 1) /", lambda_exact, "\n")

## Exact model: y = (x^ 0.4242424  - 1) / 0.4242424

cat("Approximate model: y = (x^", lambda_approx, " - 1) /", lambda_approx,
"\n")

## Approximate model: y = (x^ 0.5  - 1) / 0.5

# 3. Analizar la normalidad
analyze_normality <- function(M, name) {
  summary_stats <- c(
    min = min(M),
    q1 = quantile(M, 0.25),
    median = median(M),
    mean = mean(M),
    q3 = quantile(M, 0.75),
```

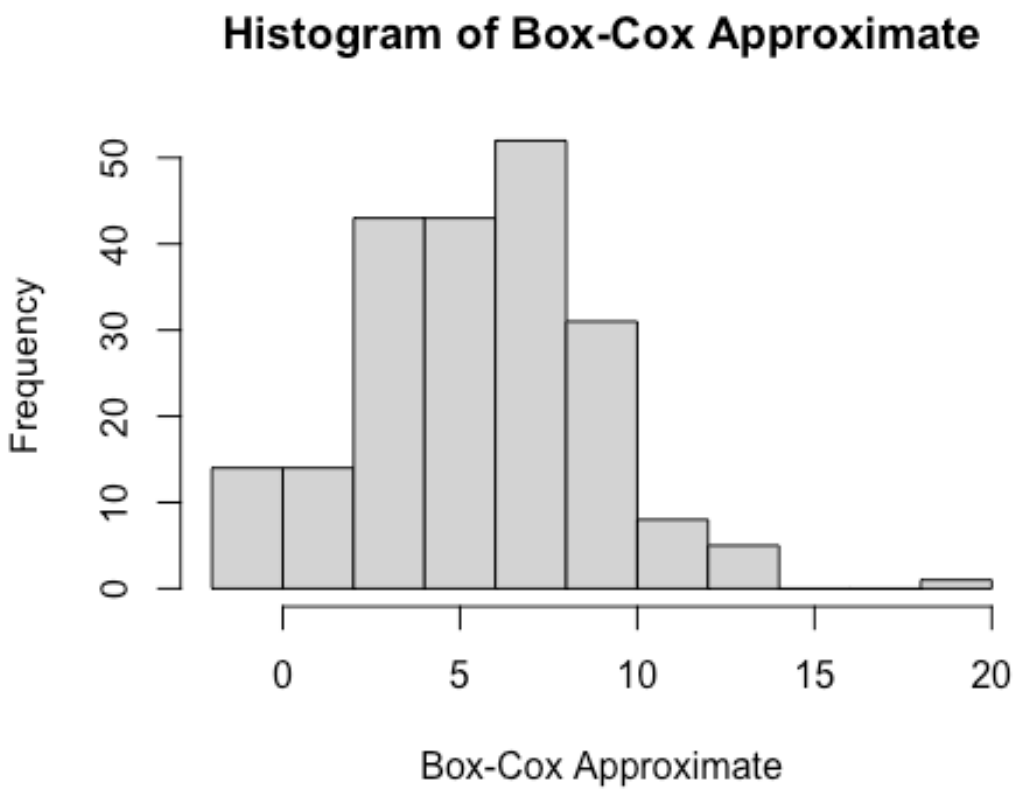
```
    max = max(M),
    skewness = skewness(M),
    kurtosis = kurtosis(M)
  )

  ad_test <- ad.test(M)

  hist(M, main = paste("Histogram of", name), xlab = name)

  return(list(summary = summary_stats, ad_test = ad_test))
}

original_analysis <- analyze_normality(selected_var, "Original")
```



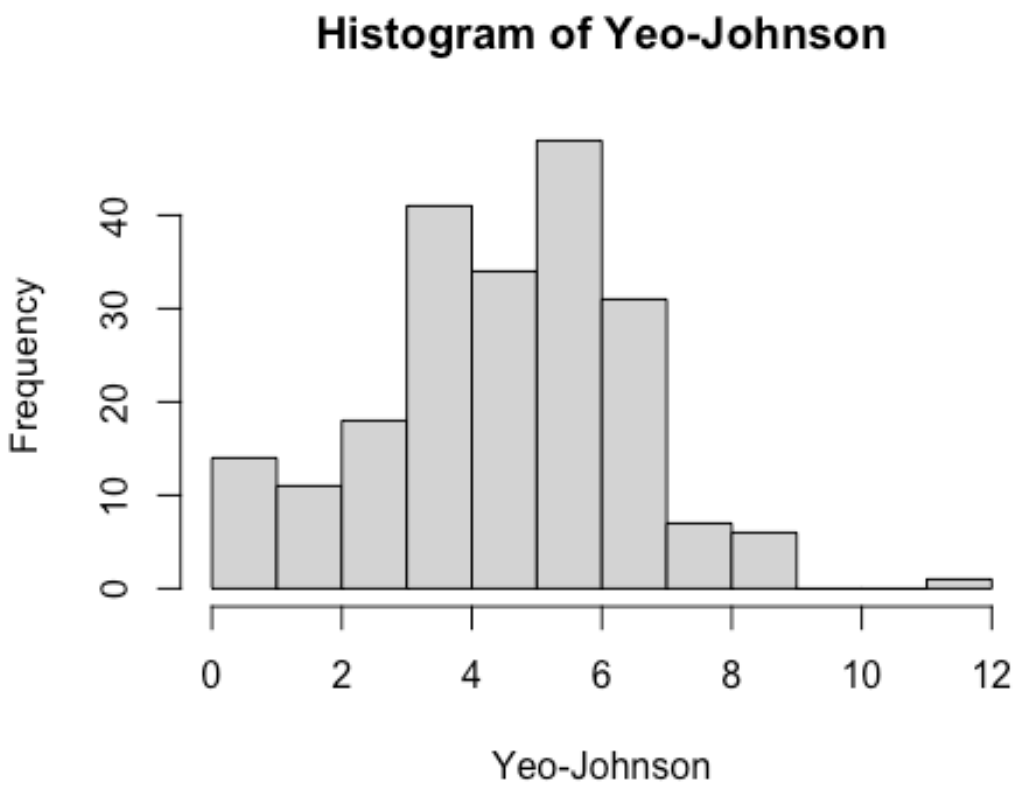

```
exact_analysis <- analyze_normality(bc_exact, "Box-Cox Exact")
```




```
approx_analysis <- analyze_normality(bc_approx, "Box-Cox Approximate")
```




```r
# 4. Detect and correct anomalies
# NA

# 5. Transformación Yeo-Johnson
yj <- car::powerTransform(selected_var, family = "yjPower")
lambda_yj <- yj$lambda

# 6. Equación de Yeo-Johnson
cat("Yeo-Johnson model: lambda =", lambda_yj, "\n")

## Yeo-Johnson model: lambda = 0.3355056

# 7. Analizar la normalidad de la transformación de Yeo-Johnson
yj_transformed <- car::yjPower(selected_var, lambda_yj)
yj_analysis <- analyze_normality(yj_transformed, "Yeo-Johnson")
```




```
# Print results
print(original_analysis)

## $summary
##        min     q1.25%     median       mean     q3.75%        max
skewness
##   0.500000   8.000000  16.000000  17.454976  23.500000 118.000000
2.369224
##   kurtosis
##  12.476340
##
## $ad_test
##
##  Anderson-Darling normality test
##
## data:  M
## A = 4.2288, p-value = 1.53e-10

print(exact_analysis)

## $summary
##         min      q1.25%      median        mean      q3.75%         max
## -0.60052939  3.33813595  5.28516775  4.96964542  6.63839307 15.48167823
##    skewness    kurtosis
```

```
##  0.05162364  0.52340285
##
## $ad_test
##
##  Anderson-Darling normality test
##
## data:  M
## A = 0.89202, p-value = 0.02232
```

```
print(approx_analysis)
```

```
## $summary
##        min     q1.25%     median       mean     q3.75%        max
skewness
## -0.5857864  3.6568542  6.0000000  5.7213543  7.6948110 19.7255610
0.2977618
##   kurtosis
##  0.9458052
##
## $ad_test
##
##  Anderson-Darling normality test
##
## data:  M
## A = 0.66959, p-value = 0.07931
```

```
print(yj_analysis)
```

```
## $summary
##          min       q1.25%       median         mean       q3.75%
max
##  0.434338279  3.248934288  4.730637027  4.474469285  5.736142646
11.832928183
##     skewness     kurtosis
## -0.005188732  0.241110082
##
## $ad_test
##
##  Anderson-Darling normality test
##
## data:  M
## A = 0.97832, p-value = 0.01365
```

8. Definir la mejor transformación

Para definir la mejor transformación, se deben comparar las pruebas de normalidad. La Box-Cox Aproximada con lamda (λ) de 0.5 y normalidad (p) de 0.07931 (la mejor normalidad) fue la mejor al compararla con la Box-Cox Exacta $\lambda \approx 0.424$ con $p = 0.02235$ y la peor Yeo-Johnson con $\lambda \approx 0.336$ y $p = 0.01365$ (no se ajusta tan bien a la normalidad).

La economía del modelo también respalda la elección de Box-Cox, por la simplicidad. Ya que no hay valores negativos o ceros en los datos (había uno pero no tan relevamnte), la flexibilidad adicional ofrecida por Yeo-Johnson no es necesaria.

9. Conclusión sobre Box-Cox y Yeo-Johnson

Las dos transformaciones se utilizan comúnmente para normalizar datos, pero la que funciona mejor depende de el caso que estemos trabajando. La transformación Box-Cox es menos complejo y es más efectivo con la normalización, mientras que Yeo-Johnson puede manejar datos que incluyen ceros y valores negativos. En pocas palabras, si puedes usa Box-Cox y ,si los datos no te lo permiten, usa Yeo-Johnson por la flexibilidad..

Para este se ha demostrado que Box-Clox es más efectiva.

10. Analizar las diferencias entre transformación y escalado

Dif1: Impacto en la Distribución

#Transformación: Cambia la distribución de los datos, lo que puede afectar la asimetría, curtosis, y otras propiedades estadísticas. Un ejemplo es lo que hicimos en esta actividad. #Escalamiento: No afecta la distribución subyacente; los datos conservan su forma original, pero los valores se reescalan.

Dif2: Propósito:

#Transformación: Modificar la distribucion de los datos para acercarnos a una distribucion normal y cumplir con los supuestos estadísticos. #Escalamiento: Cambia la magnitud de los datos para que esten en un rango común y no se desvalance el modelo (en especial cuando se trabaja con la medición de distancias entre puntos de datos). El escalamiento no altera la forma de la distribución, solo la escala.

Dif3: Aplicación

#Transformación: Es util cuando se requiere ajustar la distribución de los datos para cumplir con los supuestos de normalidad en modelos estadísticos como regresión. #Escalamiento: Se utiliza principalmente en algoritmos de machine learning que son sensibles a la magnitud de los datos.