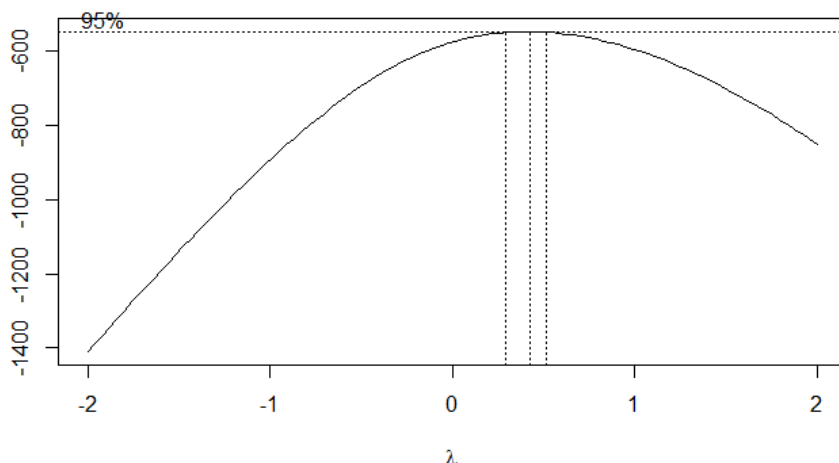


```
1 ---
2 title: "Proba5_transformaciones"
3 author: "Fernanda Pérez"
4 date: "2024-08-14"
5 output:
6   html_document:
7     df_print: paged
8   word_document: default
9   pdf_document: default
10 ---
11
12
13 Trabaja con el set de datos Mc Donalds menu Download Mc Donalds menu, que contiene diversas características del menú de alimentos de Mc Donalds.
14
15 Selecciona una variable, que no sea Calorías, y encuentra la mejor transformación de datos posible para que la variable seleccionada se comporte como una distribución Normal. Realiza:
16
17 ```{r}
18 install.packages("noritest", repos = "https://cloud.r-project.org")
19
20 ```
21
22 ```{r}
23 install.packages("noritest")
24 ```
25
26
27 ```{r}
28 install.packages("moments")
29 ```
30
31 ```{r}
32 install.packages("car")
33
34
35 ```{r}
36 library(noritest)
37 library(moments)
38 library(MASS)
39
40
41 ```{r}
42 library(car)
43
44
45 ```{r}
46 df <- read.csv("D:/Downloads/mc-donalds-menu.csv")
47
48
49 ```{r}
50 #Variable: Total.Fat
51 variable <- df$Total.Fat
52 variable <- variable[variable != 0]
53
54
55 2.Utiliza la transformación Box-Cox. Utiliza el modelo exacto y el aproximado de acuerdo con las sugerencias de Box y Cox para la transformación
56
57 ```{r}
58 #Ocupaba que todos los valores fueran positivos
59
60
61 ```{r}
62 #ocupaba que todos los valores fueran positivos
63 variable <- df$Total.Fat + 1
64
65
66 boxcox_exact <- boxcox(M2$Total.Fat ~ 1, lambda = seq(-2, 2, by = 0.1))
67 lambda_exact <- boxcox_exact$x[which.max(boxcox_exact$y)]
68 variable_transformed_exact <- (M2$Total.Fat^lambda_exact - 1) / lambda_exact
69
70
71 lambda_approx <- powerTransform(M2$Total.Fat)$lambda
72 variable_transformed_approx <- bcPower(M2$Total.Fat, lambda_approx)
73
74 ```
```



```

68 Escribe las ecuaciones de los modelos encontrados.
69 ```{r}
70 cat("Modelo exacto: Y' = (Y^", lambda_exact, " - 1) / ", lambda_exact, "\n")
71 cat("Modelo aproximado: Y' = bcPower(Y, ", lambda_approx, ")\n")
72
73 ```

```

```

Modelo exacto: Y' = (Y^ 0.4242424 - 1) / 0.4242424
Modelo aproximado: Y' = bcPower(Y, 0.40588 )

```

```

74 3. Analiza la normalidad de las transformaciones obtenidas con los datos originales. Utiliza como argumento de normalidad:
75 Compara las medidas: Mínimo, máximo, media, mediana, cuartil 1 y cuartil 3, sesgo y curtosis.
76 obten el histograma de los 2 modelos obtenidos (exacto y aproximado) y los datos originales.
77 Realiza la prueba de normalidad de Anderson-Darling o de Jarque Bera para los datos transformados y los originales
78
79 ```{r}

```

```

80 #Analizar la normalidad de las transformaciones obtenidas
81 original_stats <- summary(variable)
82 transformed_exact_stats <- summary(variable_transformed_exact)
83 transformed_approx_stats <- summary(variable_transformed_approx)
84
85 #Comparar medias, mediana, cuartiles, sesgo, curtosis
86 original_skewness <- skewness(variable)
87 original_kurtosis <- kurtosis(variable)
88 transformed_exact_skewness <- skewness(variable_transformed_exact)
89 transformed_exact_kurtosis <- kurtosis(variable_transformed_exact)
90 transformed_approx_skewness <- skewness(variable_transformed_approx)
91 transformed_approx_kurtosis <- kurtosis(variable_transformed_approx)
92

```

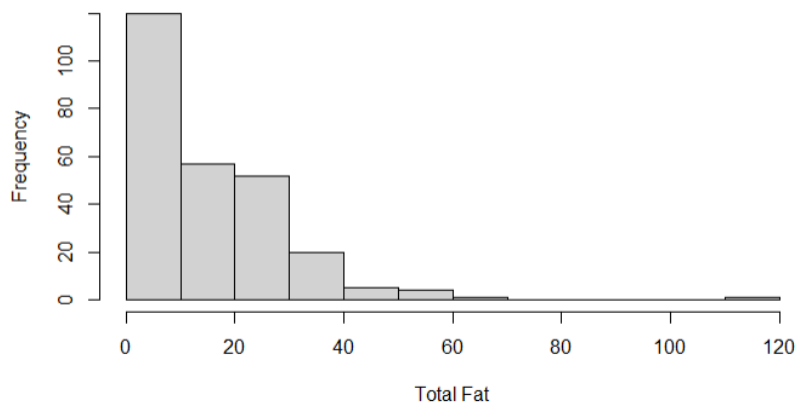
```

93 #Histogramas
94 hist(variable, main="Datos Originales", xlab="Total Fat")
95 hist(variable_transformed_exact, main="Transformación Box-Cox Exacta", xlab="Total Fat Transformado")
96 hist(variable_transformed_approx, main="Transformación Box-Cox Aproximada", xlab="Total Fat Transformado")
97
98 #Prueba de normalidad
99 library(nortest)
100 ad_original <- ad.test(variable)
101 ad_exact <- ad.test(variable_transformed_exact)
102 ad_approx <- ad.test(variable_transformed_approx)
103 ```

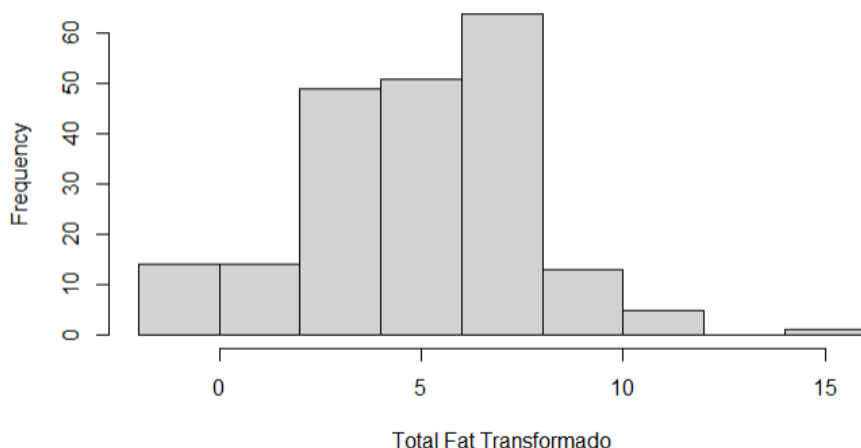
```



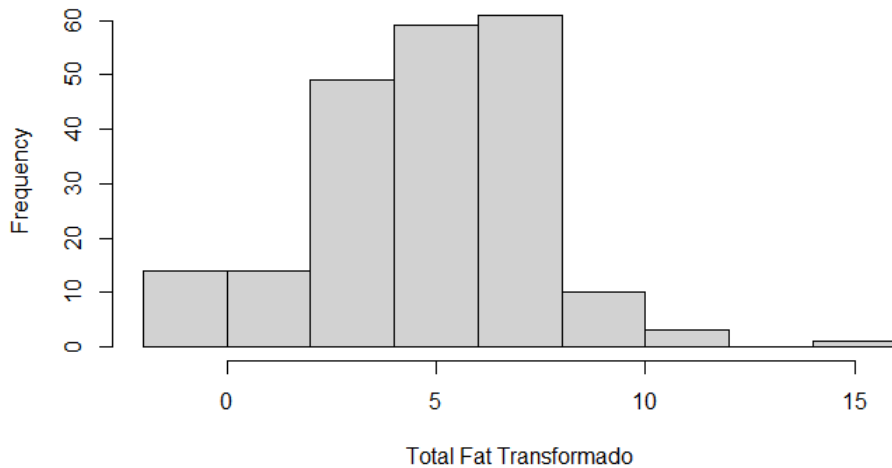
Datos Originales



Transformación Box-Cox Exacta



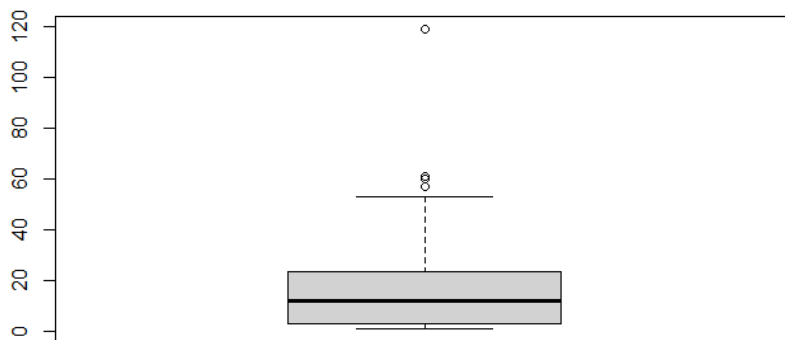
Transformación Box-Cox Aproximada



```

105 4.Detecta anomalías y corrige tu base de datos (datos atípicos, ceros anómalos, etc).
106 ```{r}
107 boxplot(variable)
108 outliers <- boxplot.stats(variable)$out
109 df_sin_outliers <- df[!variable %in% outliers, ]
110 ```

```



```

112 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).

```

```

113 ```{r}
114 library(bestNormalize)
115 yeo_johnson <- bestNormalize(variable, method = "yeojohnson")
116 lambda_yeo <- yeo_johnson$lambda
117 variable_yeo_transformed <- yeo_johnson$x.t
118 ```

```

```

120 6. Escribe la ecuación del modelo encontrado.

```

```

121 ```{r}
122 outliers <- boxplot.stats(variable_transformed_exact)$out
123 cat("outliers en el modelo exacto:", outliers, "\n")
124 
125 outliers_approx <- boxplot.stats(variable_transformed_approx)$out
126 cat("outliers en el modelo aproximado:", outliers_approx, "\n")
127 
128 variable_cleaned_exact <- variable_transformed_exact[!variable_transformed_exact %in% outliers]
129 variable_cleaned_approx <- variable_transformed_approx[!variable_transformed_approx %in% outliers_approx]
130 
131 ```

```

```

outliers en el modelo exacto: 15.48168
outliers en el modelo aproximado: 14.61818

```

```

132 5.
133 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).
134 ```{r}
135 library(car)
136
137 yeo_johnson <- powerTransform(variable, family="yjPower")
138 lambda_yeo_johnson <- yeo_johnson$lambda
139 variable_yeo_johnson <- yjPower(variable, lambda_yeo_johnson)
140
141 test_yeo_johnson <- ad.test(variable_yeo_johnson)
142
143 cat("Lambda para Yeo-Johnson:", lambda_yeo_johnson, "\n")
144 cat("valor p de la prueba de Anderson-Darling:", test_yeo_johnson$p.value, "\n")
145
146 ```

```

```

Lambda para Yeo-Johnson: 0.2344595
valor p de la prueba de Anderson-Darling: 2.495107e-14

```

```

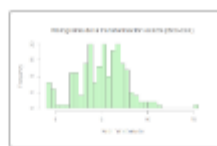
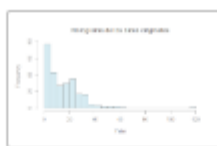
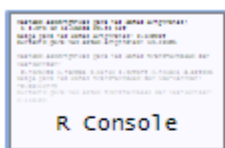
147 7. Analiza la normalidad de las transformaciones obtenidas con los datos originales. Utiliza como argumento de normalidad:
148 Compara las medidas: Mínimo, máximo, media, mediana, cuartil 1 y cuartil 3, sesgo y curtosis.
149 Obten el histograma de los 2 modelos obtenidos (exacto y aproximado) y los datos originales.
150 Realiza la prueba de normalidad de Anderson-Darling para los datos transformados y los originales
151 ```{r}
152 summary_original <- summary(variable)
153 summary_yeo_johnson <- summary(variable_yeo_johnson)
154
155 skewness_original <- skewness(variable)
156 skewness_yeo_johnson <- skewness(variable_yeo_johnson)
157
158 kurtosis_original <- kurtosis(variable)
159 kurtosis_yeo_johnson <- kurtosis(variable_yeo_johnson)
160
161 cat("Medidas descriptivas para los datos originales:\n", summary_original, "\n")
162 cat("Sesgo para los datos originales:", skewness_original, "\n")
163 cat("Curtosis para los datos originales:", kurtosis_original, "\n\n")
164
165 cat("Medidas descriptivas para los datos transformados con Yeo-Johnson:\n", summary_yeo_johnson, "\n")
166 cat("Sesgo para los datos transformados con Yeo-Johnson:", skewness_yeo_johnson, "\n")
167 cat("Curtosis para los datos transformados con Yeo-Johnson:", kurtosis_yeo_johnson, "\n")
168
169
170 # Histograma datos originales
171 hist(variable, main = "Histograma de los datos originales", xlab = "valor", col = "lightblue", breaks = 30)
172

```

```

170 # Histograma datos originales
171 hist(variable, main = "Histograma de los datos originales", xlab = "valor", col = "lightblue", breaks = 30)
172
173 # Histograma transformación exacta de Box-Cox
174 hist(variable_transformed_exact, main = "Histograma de la transformación exacta (Box-Cox)", xlab = "Valor Transformado", col =
    "lightgreen", breaks = 30)
175
176 # Histograma transformación aproximada de Box-Cox
177 hist(variable_transformed_approx, main = "Histograma de la transformación aproximada (Box-Cox)", xlab = "Valor Transformado", col =
    "lightcoral", breaks = 30)
178
179 ```

```



Medidas descriptivas para los datos originales:

1 3.375 12 15.16538 23.25 119

Sesgo para los datos originales: 2.140359

Curtosis para los datos originales: 13.45501

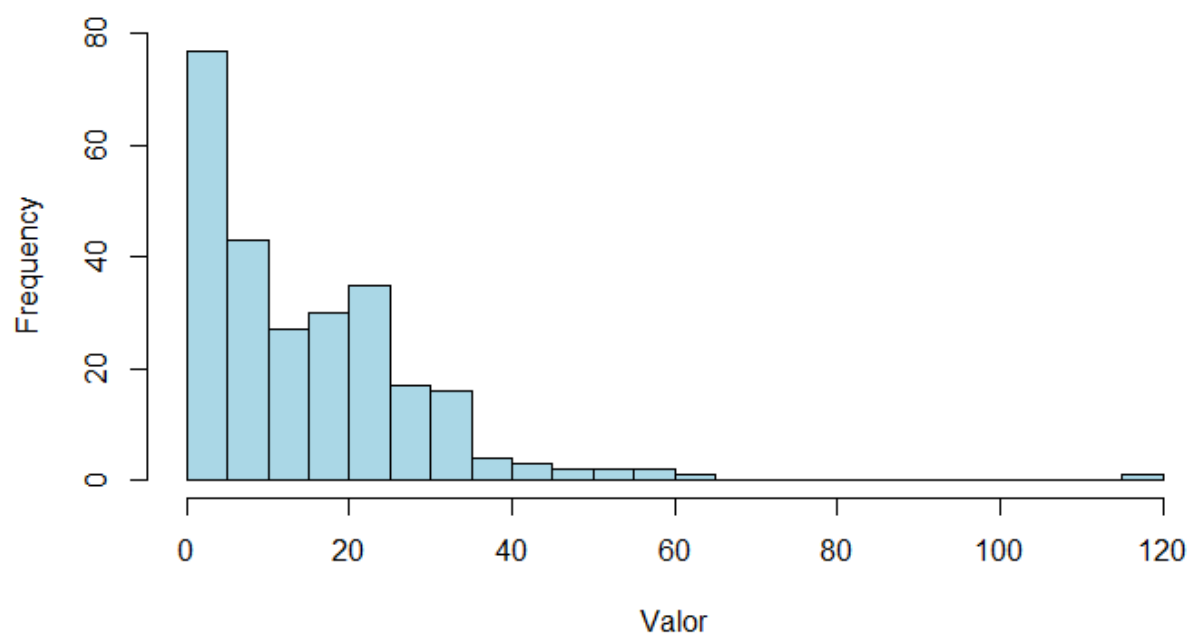
Medidas descriptivas para los datos transformados con Yeo-Johnson:

0.7526498 1.762086 3.51715 3.329679 4.741841 8.839236

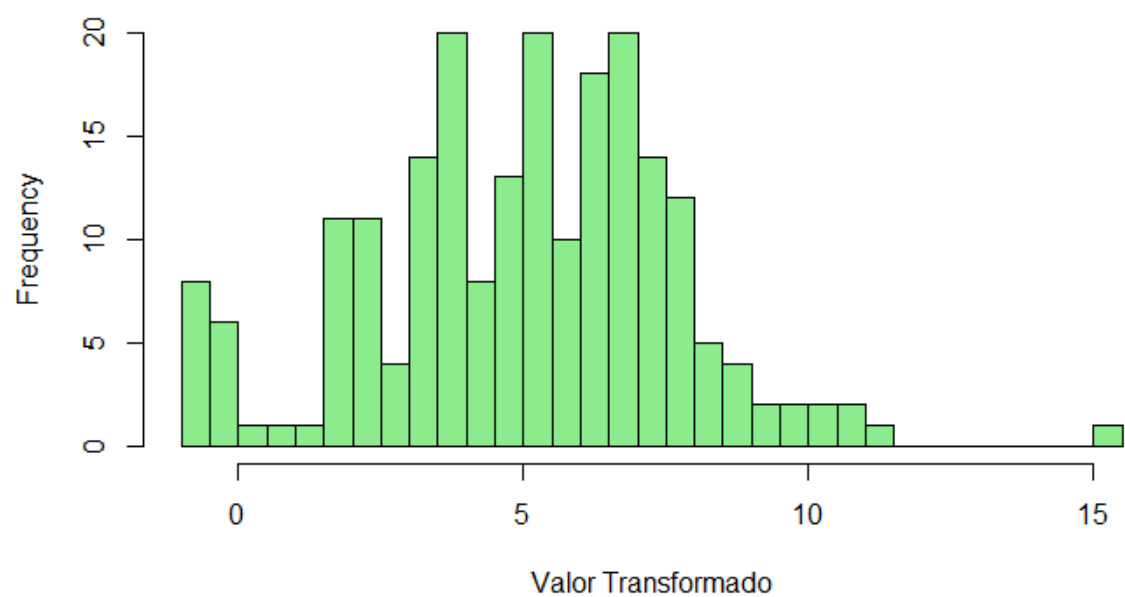
Sesgo para los datos transformados con Yeo-Johnson: -0.08413779

Curtosis para los datos transformados con Yeo-Johnson: 2.118102

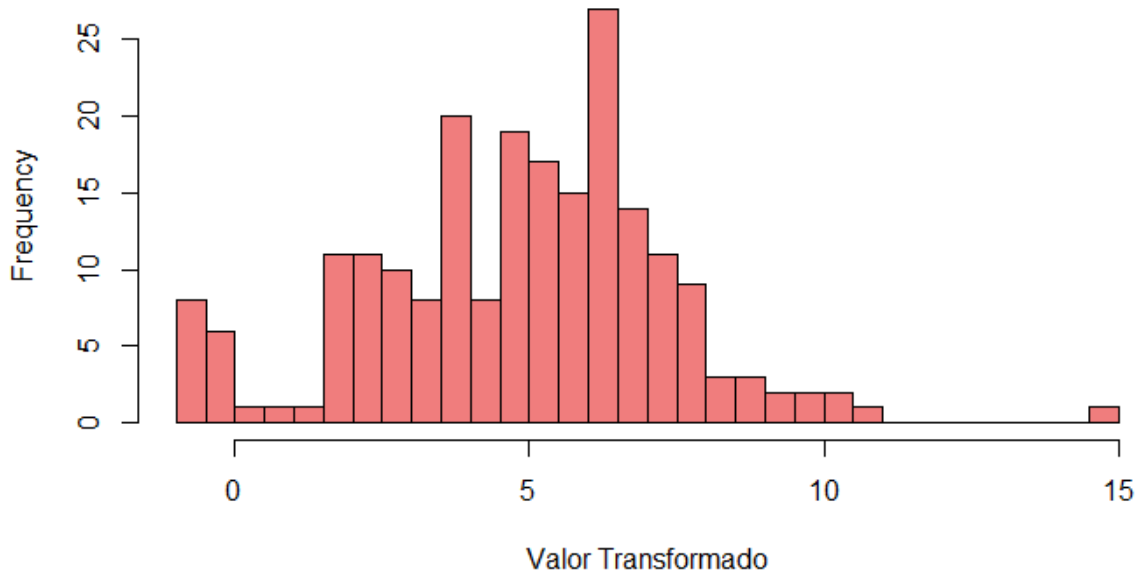
Histograma de los datos originales



Histograma de la transformación exacta (Box-Cox)



Histograma de la transformación aproximada (Box-Cox)



```
181 > test_ad_yeo_johnson <- ad.test(variable_yeo_johnson)
182
183
184 cat("Prueba de normalidad de Anderson-Darling para Yeo-Johnson:\n")
185 cat("Estadístico:", test_ad_yeo_johnson$statistic, "\n")
186 cat("Valor p:", test_ad_yeo_johnson$p.value, "\n")
187
188 >
```

```
Prueba de normalidad de Anderson-Darling para Yeo-Johnson:
Estadístico: 5.806554
Valor p: 2.495107e-14
```

```
189
190
191 > if (test_yeo_johnson$p.value > test_ad_yeo_johnson$p.value) {
192   cat("La transformación de Yeo-Johnson será mejor en este caso porque tiene un valor p mayor en la prueba de normalidad.\n")
193 } else {
194   cat("La transformación de Box-Cox parece será mejor en este caso porque tiene un mejor ajuste a la normalidad.\n")
195 }
196
197 0
198 >
```

```
La transformación de Box-Cox parece será mejor en este caso porque tiene un mejor ajuste a la normalidad.
[1] 0
```

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

Ventajas de Box-Cox:

-Eficacia en la normalización: Como vemos en los resultados, con Box-Cox se mejoró la normalidad de la distribución en comparación con los datos originales.

- Simetría en los datos transformados: El sesgo y la curtosis disminuyeron con la transformación.

Desventajas de Box-Cox:

-Limitación a datos positivos: Box-Cox solo se limita a solo aplicarse a datos que son estrictamente positivos.

Ventajas de Yeo-Johnson:

-Aplicable a datos negativos: Se puede aplicar a datos con valores negativos o cero, así que es más flexible.

Desventajas de Yeo-Johnson:

-Menor ajuste a la normalidad: En los resultados que vimos Yeo-Johnson no logró un ajuste tan bueno a la normalidad como Box-Cox.

```
228 10. Analiza las diferencias entre la transformación y el escalamiento de los datos:
229 Escribe al menos 3 diferencias entre lo que es la transformación y el escalamiento de los datos
230 Indica cuándo es necesario utilizar cada uno
231
232
233 La transformación se encarga de cambiar la distribución de los datos o sea normalizar, mientras que el escalamiento se encarga de
ajustar la escala de los datos sin alterar la distribución.
234
235 El impacto que tienen es que la transformación si modifica la forma de la distribución mientras que el escalamiento solamente ajusta
la magnitud de los datos.
236
237 La transformación solo cambia la forma de la distribución y el escalamiento solamente ajusta la magnitud de los datos.
238
239
240 Cuano es necesario usar cada uno:
241
242 Transformación: cuando queremos normalidad para análisis estadísticos.
243 Escalamiento: cuando preparamos datos para algoritmos sensibles a las magnitudes, como en el machine learning.
```