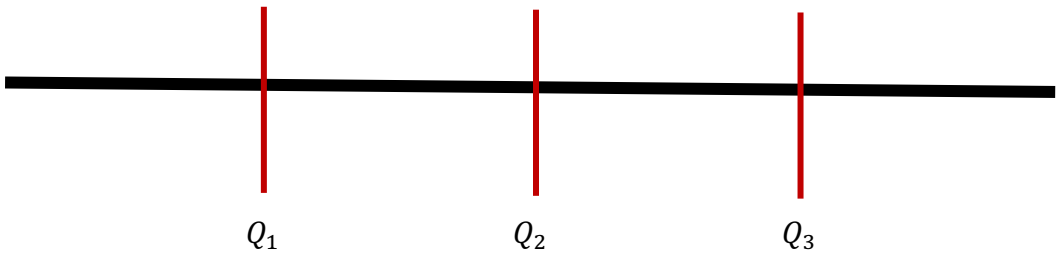
IDENTIFICACIÓN DE VALORES ATIPICOS A TRAVÉS DEL USO DE CUARTILES

INTRODUCCIÓN

Los cuartiles son los valores que dividen a una muestra de datos en cuatro partes iguales. Utilizando cuartiles puede evaluarse rápidamente la dispersión y la tendencia central de un conjunto de datos[1].



CUARTIL	DESCRIPCIÓN
1er cuartil (Q_1)	25% de los datos es menor que o igual a este valor.
2do cuartil (Q_2)	La mediana. 50% de los datos es menor que o igual a este valor.
3er cuartil (Q_3)	75% de los datos es menor que o igual a este valor.
Rango intercuartílico (IQR)	La distancia entre el primer 1er cuartil y el 3er cuartil $(Q_3 - Q_1)$; de esta manera, abarca el 50% central de los datos.

Conviene señalar que Q_2 coincide con la mediana, que es un dato estadístico que divide el conjunto de valores en dos partes iguales o simétricas[2].

[1] https://support.minitab.com/es-mx/minitab/18/help-and-how-to/graphs/how-to/boxplot/interpret-the-results/quartiles/

[2] https://economipedia.com/definiciones/cuartil.html

Documento elaborado para la materia de Diseño Electrónico Basado en Sistemas Embebidos impartida en la Facultad de Ingeniería Tampico de la Universidad Autónoma de Tamaulipas.
Elaborado por: Dr. Alejandro H. García Ruiz en abril de 2022 y actualizado en marzo de 2024

Fórmula para encontrar la posición de un cuartil mediante el uso de constantes de corrección[3]

$$Posición_{Q_i} = \frac{i \cdot (N - alfa - beta + 1)}{4} + alfa \qquad (1)$$

Donde:

$i = cuartil$
$N =$ Total de datos del conjunto

El valor de las constantes de corrección (alfa y beta) depende del método de estimación utilizado para el cálculo del cuartil.

A continuación, se presenta a los métodos de estimación utilizados por el módulo Numpy en Python. Dichos métodos fueron presentados originalmente en el trabajo de R. J. Hyndman and Y. Fan (1996), denominado "Sample Quantiles in Statistical Packages".

- **Inverted_cdf**

  Este método da resultados discontinuos

  $if\ g > 0 \quad then\ take\ j$
  $if\ g = 0 \quad then\ take\ i$

- **Averaged_inverted_cdf**

  Este método da resultados discontinuos

  $if\ g > 0 \quad then\ take\ j$
  $if\ g = 0 \quad then\ average\ between\ bounds$

- **Closest_observation**

  Este método da resultados discontinuos

  $if\ g > 0 \quad then\ take\ j$
  $if\ g = 0\ and\ index\ is\ odd \quad then\ j$
  $if\ g = 0 and\ index\ is\ even \quad then\ take\ i$

[3] https://numpy.org/doc/stable/reference/generated/numpy.percentile.html

- **Interpolated_inverted_cdf**

  Este método da resultados continuos

  $$alfa = 0$$
  $$beta = 1$$

- **Hazen**

  Este método da resultados continuos

  $$alfa = 1/2$$
  $$beta = 1/2$$

- **Weibull**

  Este método da resultados continuos

  $$alfa = 0$$
  $$beta = 0$$

- **Linear**

  Este método da resultados continuos.
  MÉTODO USADO POR DEFECTO EN EL MÓDULO NUMPY

  $$alfa = 1$$
  $$beta = 1$$

- **Median_unbiased**

  Este método da resultados continuos
  Es uno de los mejores métodos si la distribución de la muestra es desconocida

  $$alfa = 1/3$$
  $$beta = 1/3$$

- **Normal_unbiased**

  Este método da resultados continuos
  Es uno de los mejores métodos si la distribución de la muestra es normal

  $$alfa = 3/8$$
  $$beta = 3/8$$

Cálculo por interpolación lineal (Versión con corrección por: Método Weibull):

1. Ordenar al conjunto de datos V de menor a mayor. Puede ordenarse al revés, sin embargo, en dicho caso deberá tenerse cuidado al aplicar las fórmulas para obtener los datos correctos

2. Buscar la posición del cuartil deseado (Q_1, Q_2, Q_3) con la expresión (2)

   En el método de estimación Weibull, la expresión (1) es simplificada a la expresión (2) debido a que ambas constantes de corrección poseen el valor 0.

$$Posición_{Q_i} = \frac{i \cdot (N+1)}{4} \qquad (2)$$

   Donde:

$$i = \ cuartil\ buscado$$
$$N = \text{Total de datos del conjunto}$$

3. Si $Posición_{Q_i}$ es un número entero, entonces Q_i toma el valor que tenga el número asociado a la posición indicada en $Posición_{Q_i}$

$$Q_i = V\left[Posición_{Q_i}\right]$$

4. Si $Posición_{Q_i}$ no es un número entero, entonces Q_i toma el valor que tenga el número asociado a la posición de la parte entera de $Posición_{Q_i}$ sumado a la multiplicación de la parte decimal de $Posición_{Q_i}$ por la diferencia de los números asociados a la parte entera de $Posición_{Q_i} + 1$ y $Posición_{Q_i}$

$$Q_i = V\left[P.entera\_Posición_{Q_i}\right] + P.decimal\_Posición_{Q_i} \cdot \left(V\left[P.entera\_Posición_{Q_i} + 1\right] - V\left[P.entera\_Posición_{Q_i}\right]\right)$$

Ejemplo 1

$$datos = \{13, 24, 31, 32, 46, 51, 56, 74, 78, 91, 93, 141\}$$

Para Q_1:

$$datos = \{13, 24, \mathbf{31}, 32, 46, 51, 56, 74, 78, 91, 93, 141\}$$

$Posición_{Q_1} = \frac{1\cdot(12+1)}{4} = \mathbf{3.25}$
$Q_1 = 31 + 0.25 \cdot (32 - 31) = \mathbf{31.25}$

Para Q_2:

$$datos = \{13, 24, 31, 32, 46, \mathbf{51}, 56, 74, 78, 91, 93, 141\}$$

$Posición_{Q_2} = \frac{2\cdot(12+1)}{4} = \mathbf{6.5}$
$Q_2 = 51 + 0.5 \cdot (56 - 51) = \mathbf{53.5}$

Para Q_3:

$$datos = \{13, 24, 31, 32, 46, 51, 56, 74, \mathbf{78}, 91, 93, 141\}$$

$Posición_{Q_3} = \frac{3\cdot(12+1)}{4} = \mathbf{9.75}$
$Q_3 = 78 + 0.75 \cdot (91 - 78) = \mathbf{87.75}$

Para rango intercuartílico:

$IQR = Q_3 - Q_1 = \mathbf{87.75 - 31.25 = 56.5}$

Ejemplo 2

$$datos = \{7, 9, 16, 36, 39, 45, 45, 46, 48, 51\}$$

Para Q_1:

$$datos = \{7, \mathbf{9}, 16, 36, 39, 45, 45, 46, 48, 51\}$$

$Posición_{Q_1} = \frac{1 \cdot (10+1)}{4} = \mathbf{2.75}$
$Q_1 = 9 + 0.75 \cdot (16 - 9) = \mathbf{14.25}$

Para Q_2:

$$datos = \{7, 9, 16, 36, \mathbf{39}, 45, 45, 46, 48, 51\}$$

$Posición_{Q_2} = \frac{2 \cdot (10+1)}{4} = \mathbf{5.5}$
$Q_2 = 39 + 0.5 \cdot (45 - 39) = \mathbf{42.0}$

Para Q_3:

$$datos = \{7, 9, 16, 36, 39, 45, 45, \mathbf{46}, 48, 51\}$$

$Posición_{Q_3} = \frac{3 \cdot (10+1)}{4} = \mathbf{8.25}$
$Q_3 = 46 + 0.25 \cdot (48 - 46) = \mathbf{46.5}$

Para rango intercuartílico:

$IQR = Q_3 - Q_1 = \mathbf{46.5 - 14.25 = 32.25}$

VALORES ATÍPICOS

En un conjunto de datos, un valor atípico es un valor que es mucho mayor o menor que la mediana[4]. Los valores atípicos son muy comunes en la ciencia de datos.

Por ejemplo, en el cálculo de la temperatura media de 10 objetos en una habitación, si la mayoría tienen entre 20 y 25 °C, pero hay un horno a 350 °C, la mediana de los datos puede ser 23 °C, pero la temperatura media será 55 °C. En este caso, la mediana refleja mejor la temperatura de la muestra al azar de un objeto que la media[5].

Los valores atípicos pueden ser indicativos de datos que pertenecen a una población diferente del resto de las muestras establecidas. No obstante, en algunas ocasiones también pueden deberse a otros factores.

La recopilación de datos puede ser compleja y es común observar puntos de datos generados por error. Por ejemplo, un viejo dispositivo de monitoreo puede leer mediciones sin sentido antes de fallar por completo. El error humano también es una fuente de valores atípicos, en particular cuando la entrada de datos se realiza manualmente. Un individuo, por ejemplo, puede ingresar erróneamente su altura en centímetros en lugar de pulgadas o colocar el decimal en el lugar equivocado[6].

¿Cómo se puede distinguir a un valor atípico en un conjunto de datos?

Existen diferentes métodos que se pueden aplicar para identificar a valores atípicos, sin embargo, este documento se hará uso de la prueba de Tukey para identificar a los valores atípicos.

[4] https://www.nagwa.com/es/videos/265196163426/#:~:text=Un%20valor%20at%C3%ADpico%20es%20un,intercuart%C3%ADlico%2C%20se%20llaman%20valores%20at%C3%ADpicos.

[5] https://es.wikipedia.org/wiki/Valor_at%C3%ADpico

[6] https://rafalab.github.io/dslibro/robust-summaries.html

PRUEBA DE TUKEY

Los valores atípicos son en ocasiones una cuestión subjetiva, y existen numerosos métodos para clasificarlos. El método más común por su sencillez y resultados es la prueba de Tukey, que toma como referencia la diferencia entre el tercer cuartil Q_3 y el primer cuartil Q_1, o rango intercuartílico[7].

En un diagrama de caja (BoxPlot) se considera un valor atípico el que se encuentra 1.5 veces esa distancia de uno de esos cuartiles (atípico leve) o a 3 veces esa distancia (atípico extremo).

Valor atípico leve

Siendo Q_1 y Q_3 el primer y tercer cuartil, e IQR el rango intercuartil $Q_3 - Q_1$, un valor atípico leve será aquel que:

$$q < Q_1 - 1.5 \cdot IQR$$

o

$$q > Q_3 + 1.5 \cdot IQR$$

Valor atípico extremo

Siendo Q_1 y Q_3 el primer y tercer cuartil, e IQR el rango intercuartil $Q_3 - Q_1$, un valor atípico extremo será aquel que:

$$q < Q_1 - 3.0 \cdot IQR$$

o

$$q > Q_3 + 3.0 \cdot IQR$$

Representación Gráfica de la Prueba de Tukey

El Diagrama de Caja y Bigotes (Box and Whisker Plot en inglés, también conocido como BoxPlot) es un tipo de gráfico que muestra un resumen de una gran cantidad de datos en cinco medidas descriptivas, además de intuir su morfología y simetría[8].

Este tipo de gráficos nos permite identificar valores atípicos y comparar distribuciones. Además de conocer de una forma cómoda y rápida como el 50% de los valores centrales se distribuyen.

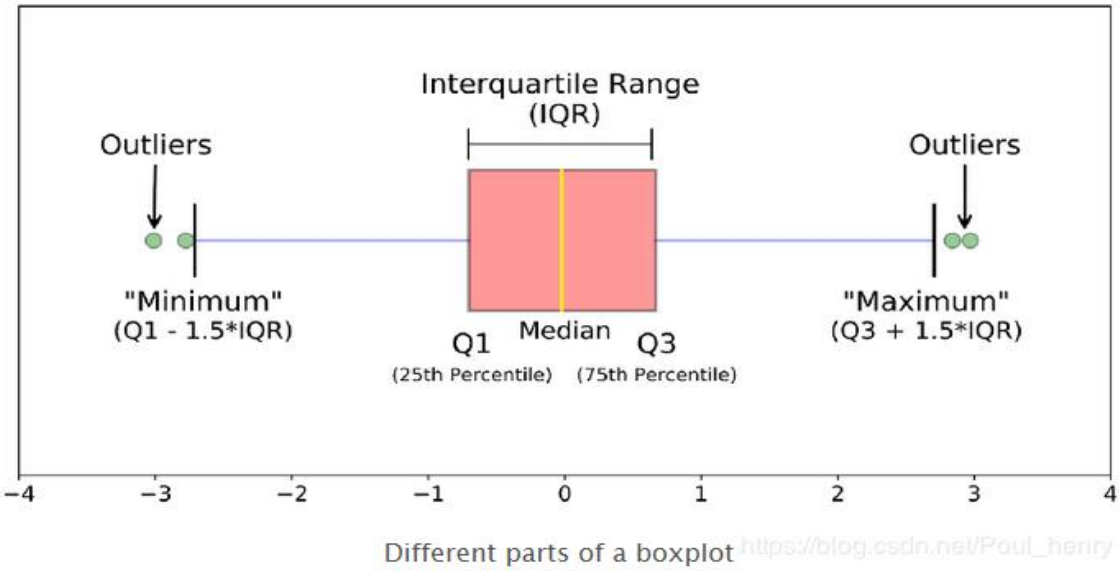
[7] https://es.wikipedia.org/wiki/Valor_at%C3%ADpico

[8] https://www.pgconocimiento.com/diagrama-boxplot/

Medidas descriptivas[9]:

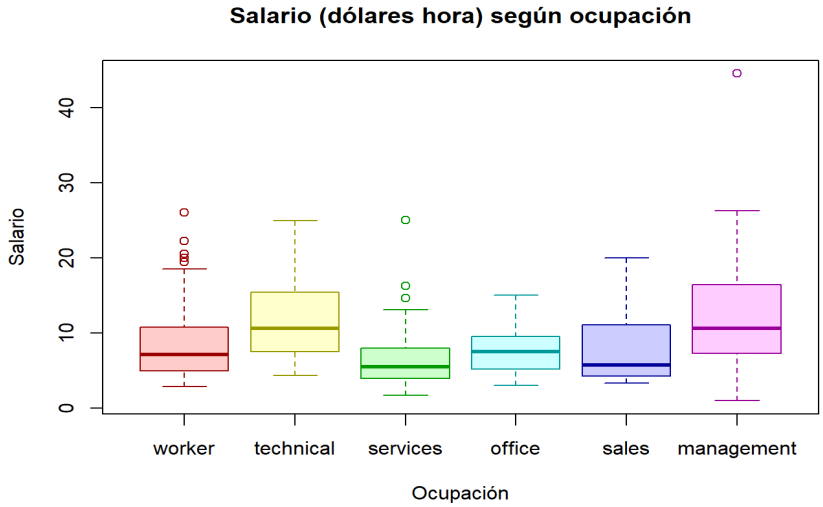
1. Primer cuartil
2. Tercer cuartil
3. Mediana
4. Límite Inferior (Mínimo)
5. Límite Superior (Máximo)

Componentes de un gráfico BoxPlot



Fuente: https://programmerclick.com/article/2315971450/

Ejemplo:



Fuente: https://www.uv.es/vcoll/graficos.html

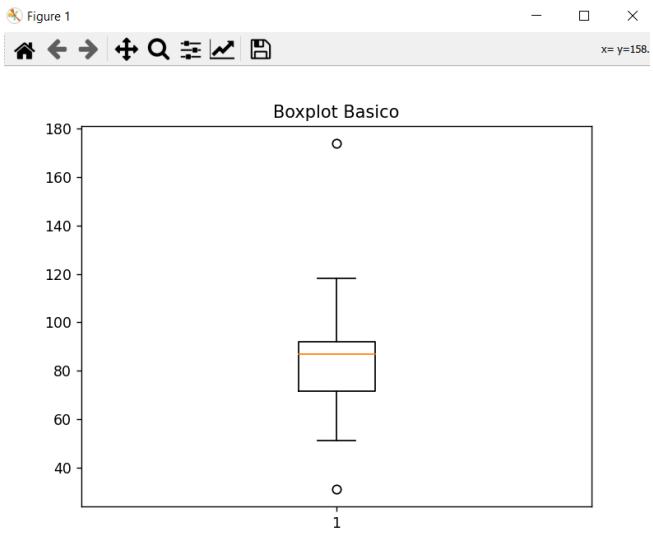
[9] https://kramirez.net/ProbaEstad/Material/Presentaciones/Capitulo8.pdf

Graficación de Boxplot en Python mediante el módulo Matplotlib

Ejemplo: Básico

```
import matplotlib.pyplot as plt

datos=[108, 31, 75, 87, 79, 88, 89, 118, 51, 89, 174, 95, 51, 70, 73]

plt.boxplot(datos)
plt.title("Boxplot Basico")
plt.show()
```

Resultado

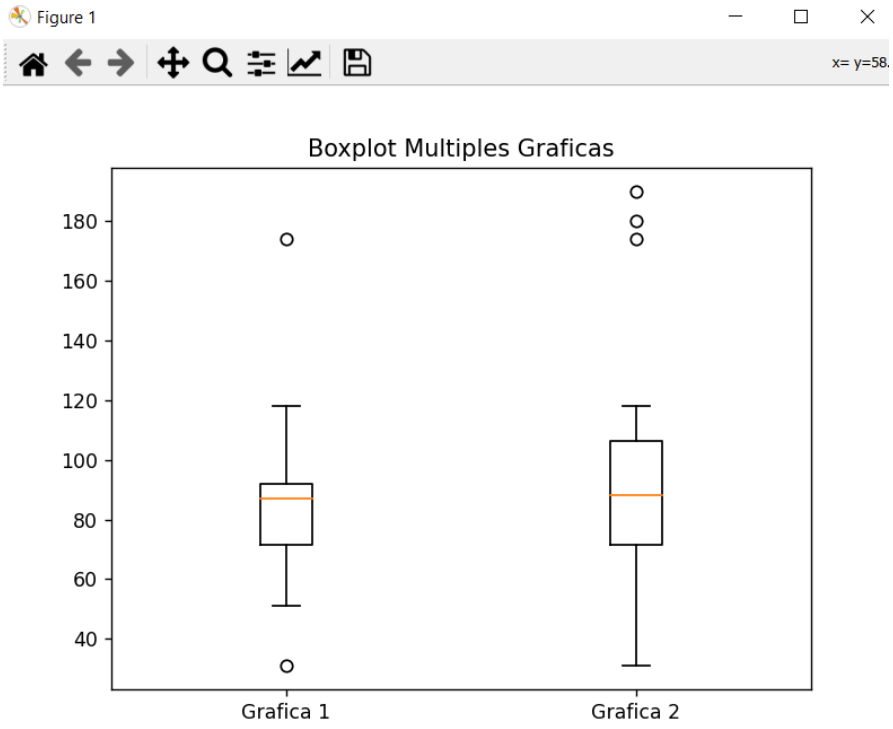


Ejemplo: Múltiples Gráficas

```
import matplotlib.pyplot as plt

datos1=[108, 31, 75, 87, 79, 88, 89, 118, 51, 89, 174, 95, 51, 70, 73]
datos2=[180, 31, 75, 190, 79, 88, 89, 118, 51, 89, 174, 95, 51, 70, 73]

datos =[datos1, datos2]

plt.boxplot(datos, labels=["Grafica 1", "Grafica 2"])

plt.title("Boxplot Multiples Graficas")
plt.show()
```

Documento elaborado para la materia de Diseño Electrónico Basado en Sistemas Embebidos impartida en la Facultad de Ingeniería Tampico de la Universidad Autónoma de Tamaulipas.
Elaborado por: Dr. Alejandro H. García Ruiz en abril de 2022 y actualizado en marzo de 2024

Resultado



Ejemplo: Avanzado

```
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec

datos1=[108, 31, 75, 87, 79, 88, 89, 118, 51, 89, 174, 95, 51, 70, 73]
datos2=[180, 31, 75, 190, 79, 88, 89, 118, 51, 89, 174, 95, 51, 70, 73]

datos =[datos1, datos2]


# Create 2x2 sub plots
gs = gridspec.GridSpec(2, 2)
figure = plt.figure(figsize=(12, 7))


ax = figure.add_subplot(gs[0, :])
bp = ax.boxplot(datos,
                patch_artist = True, # fill with color
                vert = False, # vertical box alignment
                labels=["Grafica 1", "Grafica 2"]) # will be used to label ticks
ax.set_title("Boxplot DEFAULT")

colors = ['lime', 'blue']
for patch, color in zip(bp['boxes'], colors):
    patch.set_facecolor(color)
```

```
ax = figure.add_subplot(gs[1, 0])
bp = ax.boxplot(datos, whis= 1.5,
                patch_artist = True, # fill with color
                vert = False, # vertical box alignment
                labels=["Grafica 1", "Grafica 2"]) # will be used to label ticks
ax.set_title("Boxplot Whis de 1.5")

# changing color and linewidth of
# whiskers
for whisker in bp['whiskers']:
    whisker.set(color ='#8B008B',
                linewidth = 1.5,
                linestyle =":")

# changing color and linewidth of
# caps
for cap in bp['caps']:
    cap.set(color ='red',
            linewidth = 2)

ax = figure.add_subplot(gs[1, 1])
bp = ax.boxplot(datos, whis= 3.0,
                patch_artist = True, # fill with color
                vert = False, # vertical box alignment
                labels=["Grafica 1", "Grafica 2"]) # will be used to label ticks
ax.set_title("Boxplot Whis de 3.0")

# changing color and linewidth of
# medians
for median in bp['medians']:
    median.set(color ='aqua',
               linewidth = 3)

# changing style of fliers
for flier in bp['fliers']:
    flier.set(marker ='D',
              color ='#e7298a',
              alpha = 0.5)

# x-axis labels
ax.set_yticklabels(['datos 1', 'datos 2'])

plt.show()
```

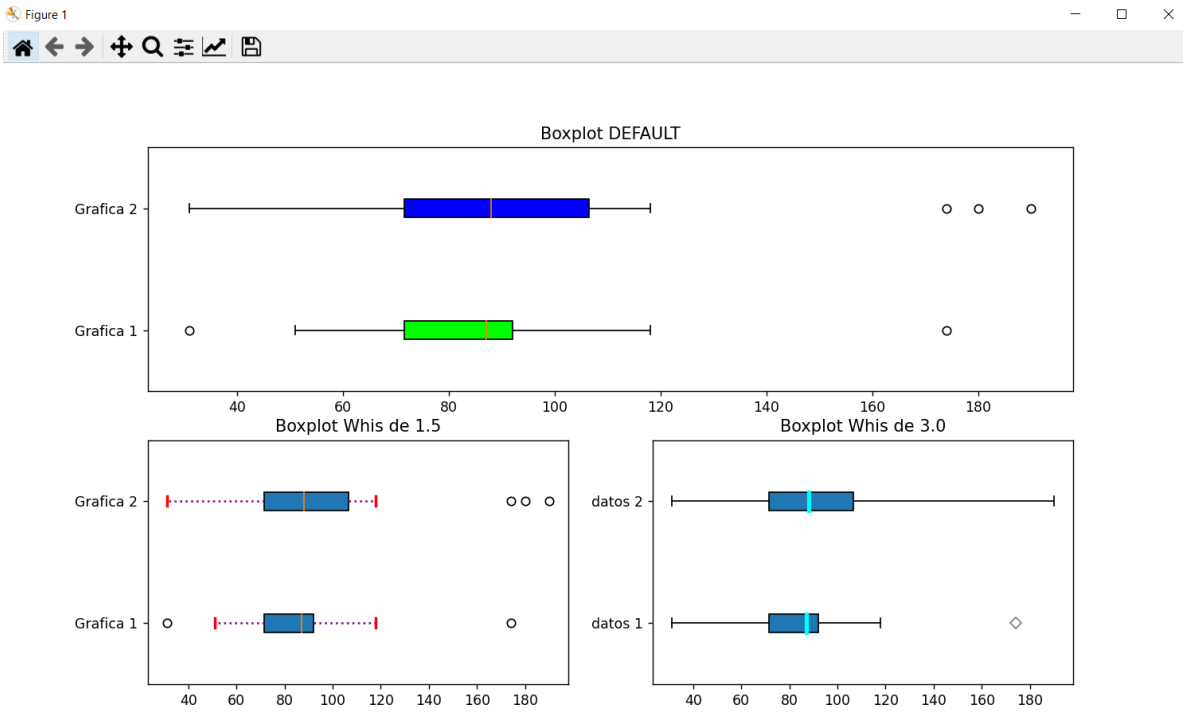


Figure 1
Boxplot DEFAULT
Grafica 2
Grafica 1
40
60
80
100
120
140
160
180
Boxplot Whis de 1.5
Grafica 2
Grafica 1
40
60
80
100
120
140
160
180
Boxplot Whis de 3.0
datos 2
datos 1
40
60
80
100
120
140
160
180

Anexos

A continuación, se presenta una lista de colores que puede ser utilizada en las gráficas con matplotlib. Esta información fue consultada en abril de 2022 de la página: https://matplotlib.org/3.5.0/gallery/color/named_colors.html

CSS Colors

black	bisque	forestgreen	slategrey
dimgray	darkorange	limegreen	lightsteelblue
dimgrey	burlywood	darkgreen	cornflowerblue
gray	antiquewhite	green	royalblue
grey	tan	lime	ghostwhite
darkgray	navajowhite	seagreen	lavender
darkgrey	blanchedalmond	mediumseagreen	midnightblue
silver	papayawhip	springgreen	navy
lightgray	moccasin	mintcream	darkblue
lightgrey	orange	mediumspringgreen	mediumblue
gainsboro	wheat	mediumaquamarine	blue
whitesmoke	oldlace	aquamarine	slateblue
white	floralwhite	turquoise	darkslateblue
snow	darkgoldenrod	lightseagreen	mediumslateblue
rosybrown	goldenrod	mediumturquoise	mediumpurple
lightcoral	cornsilk	azure	rebeccapurple
indianred	gold	lightcyan	blueviolet
brown	lemonchiffon	paleturquoise	indigo
firebrick	khaki	darkslategray	darkorchid
maroon	palegoldenrod	darkslategrey	darkviolet
darkred	darkkhaki	teal	mediumorchid
red	ivory	darkcyan	thistle
mistyrose	beige	aqua	plum
salmon	lightyellow	cyan	violet
tomato	lightgoldenrodyellow	darkturquoise	purple
darksalmon	olive	cadetblue	darkmagenta
coral	yellow	powderblue	fuchsia
orangered	olivedrab	lightblue	magenta
lightsalmon	yellowgreen	deepskyblue	orchid
sienna	darkolivegreen	skyblue	mediumvioletred
seashell	greenyellow	lightskyblue	deeppink
chocolate	chartreuse	steelblue	hotpink
saddlebrown	lawngreen	aliceblue	lavenderblush
sandybrown	honeydew	dodgerblue	palevioletred
peachpuff	darkseagreen	lightslategray	crimson
peru	palegreen	lightslategrey	pink
linen	lightgreen	slategray	lightpink

Documento elaborado para la materia de Diseño Electrónico Basado en Sistemas Embebidos impartida en la Facultad de Ingeniería Tampico de la Universidad Autónoma de Tamaulipas.
Elaborado por: Dr. Alejandro H. García Ruiz en abril de 2022 y actualizado en marzo de 2024