

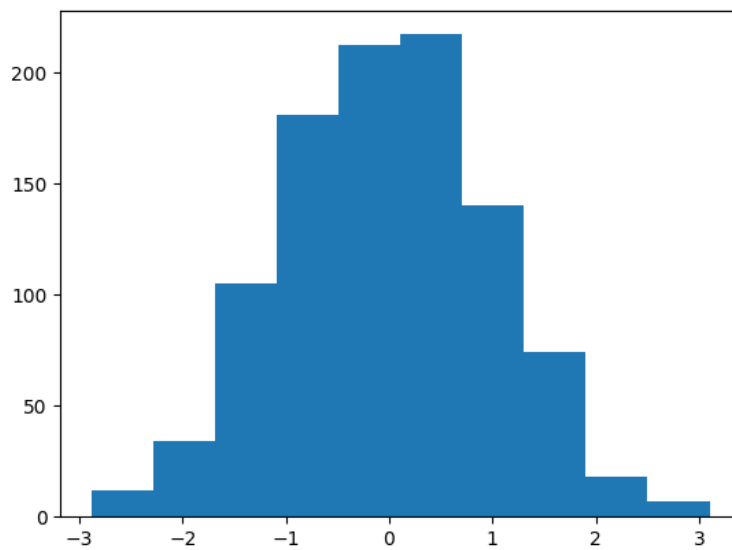
Visualización con Matplotlib: Histogramas y diagramas de caja

Vamos a ver de forma sencilla como hacer los gráficos de los histogramas y los boxplots:

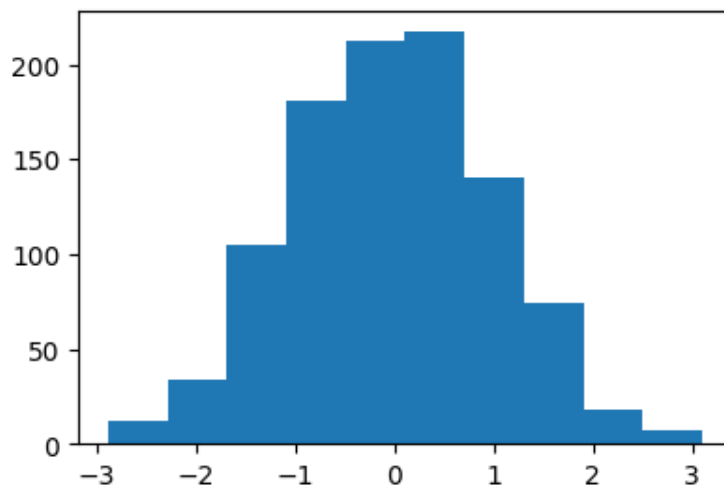
```
import matplotlib.pyplot as plt
import numpy as np
data=np.random.randn(1000)
```

Histogramas

```
plt.hist(data);
```

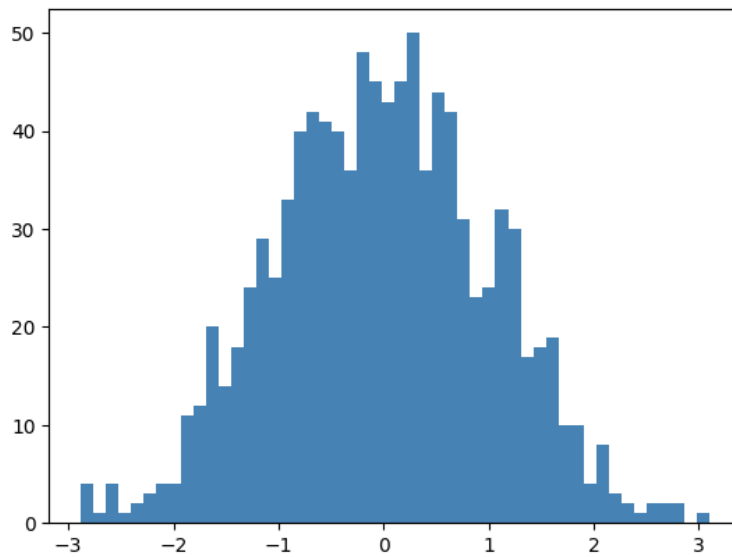


```
fig=plt.figure(figsize=(4.5,3))
ax=plt.axes()
ax.hist(data);
```



La función `hist()` tiene muchas opciones para ajustar tanto el cálculo como la visualización; aquí tienes un ejemplo de un histograma más personalizado (en el que tocamos el número de columnas bins, la transparencia y el color):

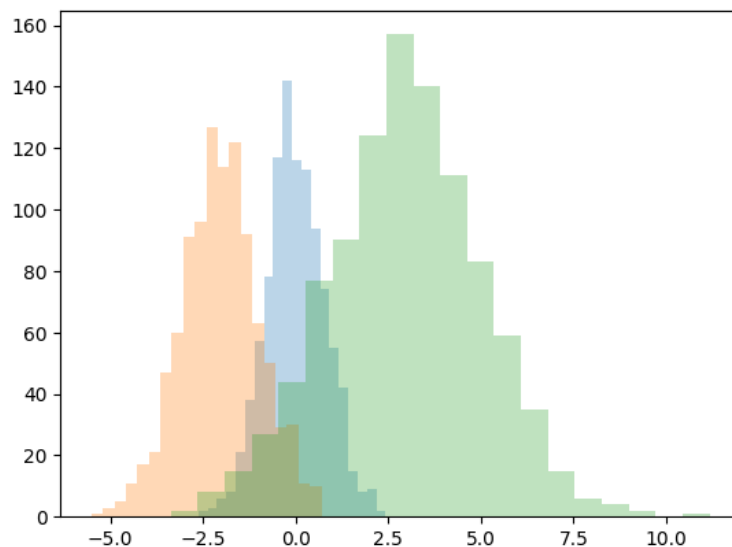
```
plt.hist(data, bins=50, alpha=1, color="steelblue");
```



La transparencia es particularmente interesante para comparar histogramas o mostrar dicha comparación (y aprovecho para enseñarte un truco para cuando tengas argumentos repetidos que poner en tus llamadas a función):

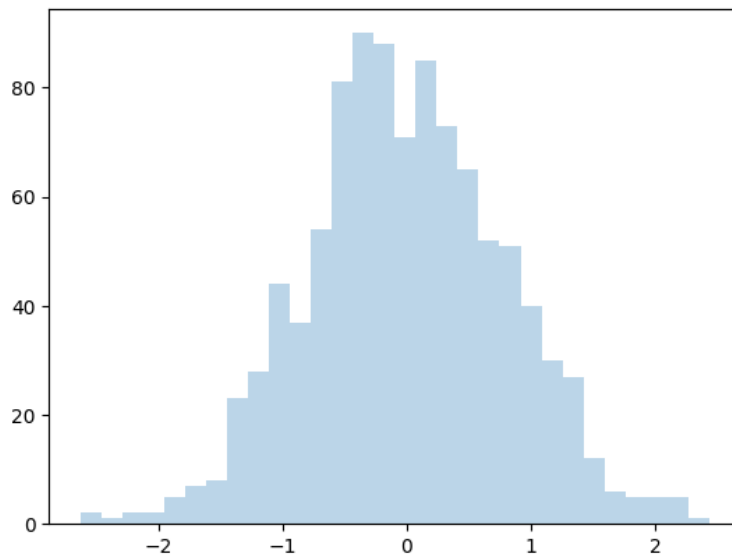
```
x1 = np.random.normal(0, 0.8, 1000)
x2 = np.random.normal(-2, 1, 1000)
x3 = np.random.normal(3, 2, 1000)
```

```
kwargs=dict(alpha=0.3,bins=20)
plt.hist(x1,**kwargs)
plt.hist(x2,**kwargs)
plt.hist(x3,**kwargs);
```



Es también el momento de enseñarte a capturar los límites de esos bins:

```
fig=plt.figure()
ax=plt.axes()
limites=ax.hist(x1,alpha=0.3,bins=30)
```



Limites es una tupla en la que el primer valor son los conteos y el segundo los límites de las barras:

```
limites[0]
```

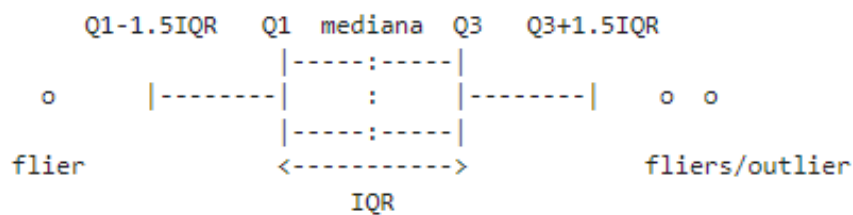
```
array([ 2.,  1.,  2.,  2.,  5.,  7.,  8., 23., 28., 44., 37., 54., 81., 90., 88., 71., 85., 73., 65., 52., 51., 40., 30., 27., 12.,  6.,  5.,  5.,  5.,  1.])
```

```
limites[1]
```

```
array([-2.62411746, -2.4555986 , -2.28707975, -2.11856089, -1.95004204, -1.78152318, -1.61300433, -1.44448547, -1.27596662, -1.10744776, -0.93892891, -0.77041005, -0.6018912 , -0.43337234, -0.26485349, -0.09633463,  0.07218422,  0.24070308,  0.40922193,  0.57774079,  0.74625964,  0.9147785 ,  1.08329735,  1.25181621,  1.42033506,  1.58885392,  1.75737277,  1.92589163,  2.09441048,  2.26292934,  2.43144819])
```

Boxplots

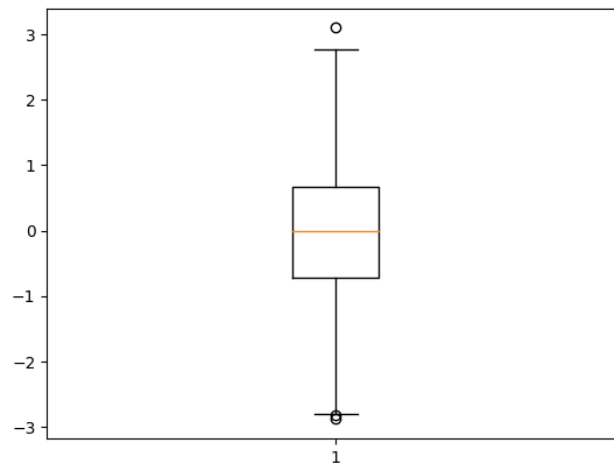
Los diagramas de caja eran esos diagramas en los que resumíamos las medidas de posición y añadíamos una posible forma de obtener valores anómalos, extremos o outliers.



Donde IQR es el rango intercuartilico (la diferencia entre el valor de Q3 o tercer cuartil o percentil-75 y el valor de Q1 o primer cuartil, o percentil-25)

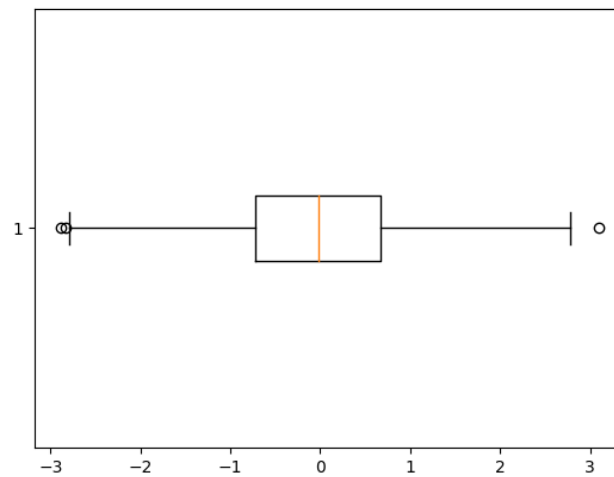
En matplotlib:

```
plt.boxplot(data,whis=1.5);
```



En matplotlib empleamos por defecto diagramas de caja verticales. Pero podemos cambiarlo:

```
plt.boxplot(data,vert=False);
```



Y podemos mostrar varios juntos:

```
plt.boxplot([x1,x2,x3]);
```

