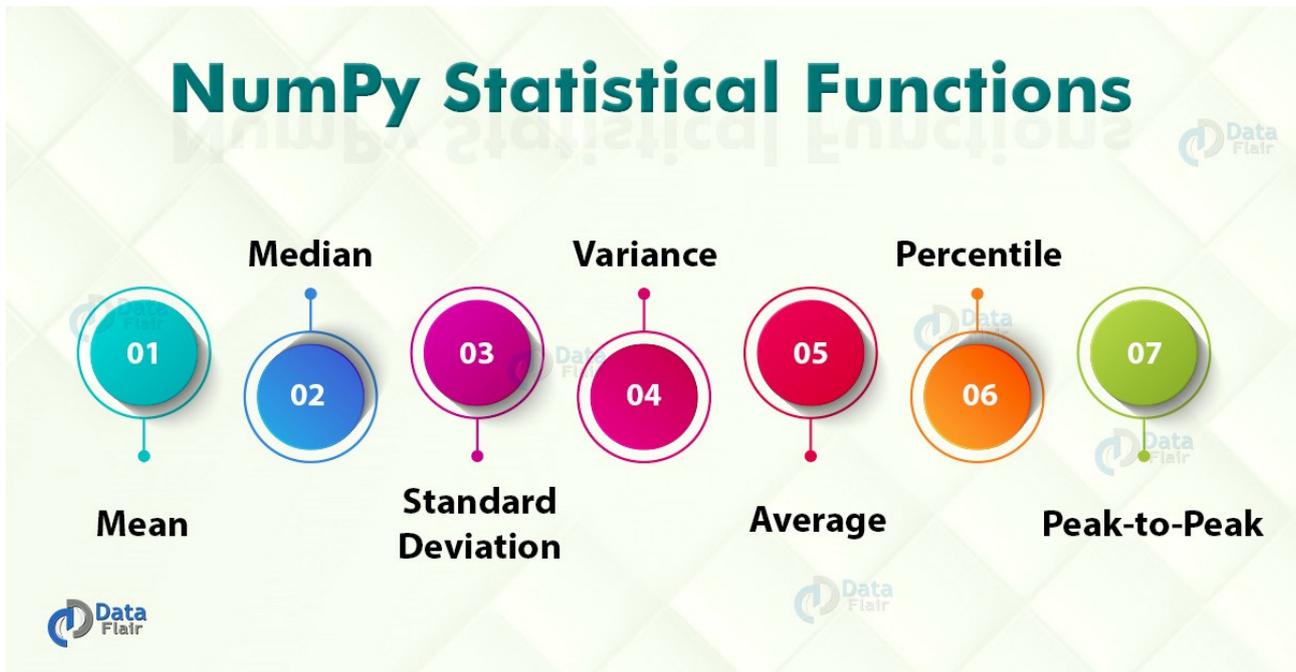


# Funciones estadísticas NumPy



La estadística se ocupa de recopilar y luego analizar esos datos. Incluye métodos para recolectar las muestras, describir los datos y luego concluir esos datos. NumPy es el paquete fundamental para los cálculos científicos y, por lo tanto, va de la mano para las funciones estadísticas de NumPy.

NumPy contiene varias funciones estadísticas que se utilizan para realizar análisis de datos estadísticos. Estas funciones estadísticas son útiles para encontrar un máximo o un mínimo de elementos. También se utiliza para encontrar conceptos estadísticos básicos como desviación estándar, varianza, etc.

## Funciones estadísticas NumPy

NumPy está equipado con las siguientes funciones estadísticas:

1. **np.amin ()**: esta función determina el valor mínimo del elemento a lo largo de un eje especificado.
2. **np.amax ()**: esta función determina el valor máximo del elemento a lo largo de un eje especificado.
3. **np.mean ()**: determina el valor medio del conjunto de datos.
4. **np.median ()** - Determina el valor mediano del conjunto de datos.
5. **np.std ()** - Determina la desviación estándar
6. **np.var** - Determina la varianza.
7. **np.ptp ()**: devuelve un rango de valores a lo largo de un eje.
8. **np.average ()** - Determina el promedio ponderado
9. **np.percentile ()** - Determina el percentil n de datos a lo largo del eje especificado.

# Encontrar el máximo y el mínimo de matriz en NumPy

Las **funciones** NumPy **np.amin ()** y **np.amax ()** son útiles para determinar el valor mínimo y máximo de los elementos de la matriz a lo largo de un eje especificado.

```
import numpy as np
arr= np.array([[1,23,78],[98,60,75],[79,25,48]])
print(arr)
#Minimum Function
print(np.amin(arr))
#Maximum Function
print(np.amax(arr))
```

## Salida

```
[[1 23 78]
 [98 60 75]
 [79 25 48]]
1
98
```

# Hallar la media, la mediana, la desviación estándar y la varianza en NumPy

## Media

La media es la suma de los elementos dividida por su suma y dada por la siguiente fórmula:

$$\bar{x} = \frac{1}{n}(x_1 + x_2 + \dots + x_n)$$

Calcula la media sumando todos los elementos de las matrices y luego la divide por el número de elementos. También podemos mencionar el eje a lo largo del cual se puede calcular la media.

```
import numpy as np
a = np.array([5,6,7])
print(a)
print(np.mean(a))
```

## Salida

```
[5 6 7]
6,0
```

# Mediana

La mediana es el elemento central de la matriz. La fórmula difiere para conjuntos pares e impares.

<u>Odd</u> $\frac{n+1}{2}$	<u>Even</u> $\frac{n}{2}, \frac{n}{2} + 1$
-------------------------------	---

Puede calcular la mediana para matrices unidimensionales y multidimensionales. La mediana separa el rango superior e inferior de valores de datos.

```
import numpy as np
a = np.array([5,6,7])
print(a)
print(np.median(a))
```

## Salida

```
[5 6 7]
6,0
```

# Desviación Estándar

La desviación estándar es la raíz cuadrada del promedio de las desviaciones cuadradas de la media. La fórmula para la desviación estándar es:

$$SD = \sqrt{\frac{\sum |x - \bar{x}|^2}{n}}$$

```
import numpy as np
a = np.array([5,6,7])
print(a)
print(np.std(a))
```

## Salida

```
[5 6 7]
0.816496580927726
```

# Varianza

La varianza es el promedio de las desviaciones cuadradas. La siguiente es la fórmula para lo mismo:

$$\sigma^2 = \frac{\sum (x_r - \bar{x})^2}{n}$$

```
import numpy as np
a = np.array([5,6,7])
print(a)
print(np.var(a))
```

## Salida

```
[5 6 7]
0.6666666666666666
```

# Función de promedio NumPy

La función NumPy `np.average()` determina el promedio ponderado junto con las matrices multidimensionales. El promedio ponderado se calcula multiplicando el componente por su peso, los pesos se especifican por separado. Si no se especifican los pesos, produce el mismo resultado que la media.

```
import numpy as np
a = np.array([5,6,7])
print(a)
#without weight same as mean
print(np.average(a))
#with weight gives weighted average
wt = np.array([8,2,3])
print(np.average(a, weights=wt))
```

## Salida

```
[5 6 7]
6.0
5.615384615384615
```

# Función de percentil numérico

Tiene la siguiente sintaxis:

**numpy.percentile (input, q, axis)**

Los parámetros aceptados son:

- **input:** es la matriz de entrada.
- **q:** es el percentil que calcula de los elementos del arreglo entre 0-100.
- **axis:** especifica el eje a lo largo del cual se realiza el cálculo.

```
a = np.array([2,10,20])  
print(a)  
print(np.percentile(a,10,0))
```

**Salida**

```
[2 10 20]  
3,6
```

## Resumen

Estas funciones son útiles para realizar cálculos estadísticos en los elementos de la matriz. Las funciones estadísticas de NumPy aumentan aún más el alcance del uso de esta biblioteca. El objetivo de las funciones estadísticas es eliminar la necesidad de recordar fórmulas extensas. Hace que el procesamiento sea más fácil de usar.