

## ▼ Introducción a NumPy

### Facultad de Economía, UNAM

**Profesor: Cesar Hernández**

**NumPy (Numerical Python)** es una librería de Python que permite realizar arreglos con vectores y matrices además de diversas funciones matemáticas.

```
# Para importar la librería
import numpy as np
```

## ▼ Listas

Primero aprenderemos a trabajar con objetos de clase *list*.

- Para crear una lista:

```
lista = [12, 39, 41, 7]
lista

[12, 39, 41, 7]
```

- Para conocer qué tipo de objeto es *lista*:

```
print(type(lista))

<class 'list'>
```

- Para unir el contenido de dos listas (no se pueden hacer operaciones matemáticas con listas):

```
print(lista + lista)

[12, 39, 41, 7, 12, 39, 41, 7]
```

## ▼ Vectores

Podemos convertir los objetos tipo *list* a objetos tipo *array* con la ayuda de **NumPy** y la función *np.array*.

- Para transformar nuestro objeto *lista* a un *vector*:

```
vector = np.array(lista)
vector

array([12, 39, 41, 7])
```

- Para saber qué tipo de objeto es *vector*:

```
print(type(vector))

<class 'numpy.ndarray'>
```

- Para saber el número de elementos del *vector*:

```
vector.size

4
```

- Para obtener el menor valor dentro del *vector*:

```
vector.min()

7
```

- Para obtener el mayor valor dentro del *vector*:

```
vector.max()

41
```

- Para ordenar los elementos del *vector* de menor a mayor

```
vector.sort()  
vector  
  
array([ 7, 12, 39, 41])
```

- Para sumar los elementos de un *vector*:

```
vector.sum()  
  
99
```

## ▼ Operaciones con vectores

Podemos hacer algunas operaciones matemáticas con objetos tipo *np.array*.

- Primero crearemos el *vector\_a* y el *vector\_b*:

```
vector_a = np.array([-3, -1, 2, 4])  
vector_b = np.array([-2, 0, 1, 3])  
print(vector_a, vector_b)  
  
[-3 -1  2  4] [-2  0  1  3]
```

- Para sumar dos vectores:

```
print(vector_a + vector_b)  
  
[-5 -1  3  7]
```

- Para restar dos vectores:

```
print(vector_b - vector_a)  
  
[ 1  1 -1 -1]
```

- Para multiplicar un vector por un escalar:

```
print(vector_b * 5)
```

```
[-10  0  5 15]
```

- Para multiplicar dos vectores:

```
print(vector_a * vector_b)  
[ 6  0  2 12]
```

- Para dividir dos vectores:

```
print(vector_b / vector_a)  
[ 0.66666667 -0.          0.5          0.75          ]
```

- Para redondear los valores de un vector:

```
division = vector_b / vector_a  
division.round()  
array([ 1., -0.,  0.,  1.])
```

## ▼ Matrices y operaciones con matrices

Podemos hacer matrices si integramos varias listas a objetos de tipo *np.array*.

- Primero crearemos los objetos *vector\_c*, *vector\_d* y *vector\_e* y después los integraremos en un objeto tipo *np.array*:

```
vector_c = [ 15, 21, 45]  
vector_d = [ 63, 30, 29]  
vector_e = [ 36, 54, 71]  
matriz = np.array([vector_c, vector_d, vector_e])  
matriz  
  
array([[15, 21, 45],  
       [63, 30, 29],  
       [36, 54, 71]])
```

- Para sumar dos matrices:

```
print(matriz + matriz)
```

```
[[ 30  42  90]
 [126  60  58]
 [ 72 108 142]]
```

- Para multiplicar dos matrices:

```
print(matriz * matriz)
```

```
[[ 225  441 2025]
 [3969  900  841]
 [1296 2916 5041]]
```

- Para dividir dos matrices:

```
print(matriz / matriz)
```

```
[[1.  1.  1.]
 [1.  1.  1.]
 [1.  1.  1.]]
```

- Para obtener el determinante de una matriz:

```
np.linalg.det(matriz)
```

```
40940.99999999999
```

- Para obtener la matriz inversa:

```
np.linalg.inv(matriz)
```

```
array([[ 0.01377592,  0.02293544, -0.01809922],
       [-0.08375467, -0.01355609,  0.05862094],
       [ 0.05671576, -0.00131897, -0.02132337]])
```

## Material adicional

**NumPy** es una librería de código abierto que cuenta con una gran comunidad por detrás. Podrás encontrar diversos recursos en línea que te ayudarán a continuar con tu aprendizaje. A continuación te dejamos algunas sugerencias de recursos en línea con los que podrás reforzar tu aprendizaje:

- [Learn NumPy in 5 minutes](#)
- [Introduction to NumPy](#)
- [Learn NumPy](#)

## Créditos

Esta obra fue generada mediante Colaboratory el 27 de enero de 2020 y forma parte de las actividades realizadas en las materias de [Matemáticas I y Taller III](#), [Facultad de Economía, UNAM](#).

Esta obra está bajo una [licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional](#). Creative Commons (CC).

