

## TEMA 2. Numpy

Numpy es un paquete de módulos de computación numérica eficiente, da la posibilidad de usar vectores multidimensionales en un tipo eficiente denominado ndarray.

Ndarray es un vector n-dimensional rectangular de datos, estos datos pueden ser de cualquier tipo simple para los cuales incluye algoritmos eficientes.

Las funciones operan elemento a elemento y además de esto se tiene una programación rápida de bucles para datos fundamentales.  $X+y$  o  $\sin(x)$  lo haría en toda la matriz sin usar for.

Se declara con

```
A=numpy.array([[1,2,3],[4,5,6]],type)
```

Características:

- Todas las filas con la misma longitud
- Todos los elementos del mismo tipo
- Si no se especifica tipo coge uno por defecto.
- Con `zeros(n)` y `ones(n)` se pueden crear arrays de tamaño n iniciados a 0 o a 1.
- Tenemos `Linspace(a,b,n)` que genera n valores entre a y b (incluido) espaciados en la misma proporción.
- Parecido al anterior `arange(a,b,x,tipo)`, genera un array con valores entre a y b (sin b) de x en x valores.
- Podemos usar creación a partir de una función `a=fromfunction(función,(3,6))` se indica la función y las filas y las columnas.
- Se puede hacer uso de generadores aleatorios para crear arrays:
  - o `Numpy.random.uniform(0,1,9)`: 9 valores entre 0 y 1
  - o `Numpy.random.uniform(0,1,(5,9))`: 5 filas de 9 columnas con valores entre 0 y 1
- Cambios automáticos de tipo de datos en asignaciones.
- La selección de datos se puede hacer como listas normales, o usando mascara como un array de tipos booleanos y a continuación:
  - o `x=a[mask]`
  - o `x=compress(mask,a)`
- Los recorridos pueden hacerse sobre todos los elementos (`a.flat`) o sobre todos los índices y valores (`ndenumerate(a)`)

Dispone además de funciones para conocer datos sobre el array:

- `Shape`: indica formato filas,columnas
- `Reshape`: permite cambiar el formato filas/columnas
- `Dtype`: tipo de dato
- `Size`: numero de elementos.
- `Ndim`: numero de dimensiones
- Conversión a lista con `a.tolist()` o con `list(a)`, mejor `tolist`, mas eficiente.
- Conversión de objeto a array con `asarray()`

NOTA: Las funciones universales `ufuncs` son objetos que evalúan rápidamente una función elemento a elemento, en los arrays 1D es parecido a la comprensión de listas pero mas rápido.

## TEMA 2. Numpy

### NUMEROS ALEATORIOS

Se pueden generar números aleatorios escalares con import random:

- Random.seed(x)
- Random.random(): uniforme 0,1
- Random.uniform(-1,1): uniforme entre -1, 1
- Random.gauss(0,1): aleatorio entre 0,1
- Random.randint (1,1): aleatorio entero entre 1,10

Con numpy podemos generar números aleatorios vectorizados

- Numpy.random.seed(x)
- Numpy.random(n): n valores
- Numpy.uniform(-1,1,n): n uniforme entre -1,1
- Numpy.normal(m,s,n): n valores de la normal  $N(m,s)$

### MATRIX

Numpy dispone de un tipo matrix que se puede usar en arrays 1D y 2D

```
X1=array([1,2,3],float)
```

```
X2=matrix(x1)
```

El operador ^corresponde a la multiplicación entre matrices.