

Intro to ML Python

Tricks Jupyter Notebook

Shortcuts

- Ctrl + Enter : exécuter
- Shift + Enter : exécuter et suivant
- Shift + Tab : affiche la doc d'une fonction

Code

- `function?` --> affiche la doc de `function`

NumPy

Créer un tableau :

```
x_2d = np.array([(1, 1, 1), [2, 2, 2], (3, 3, 3)])
```

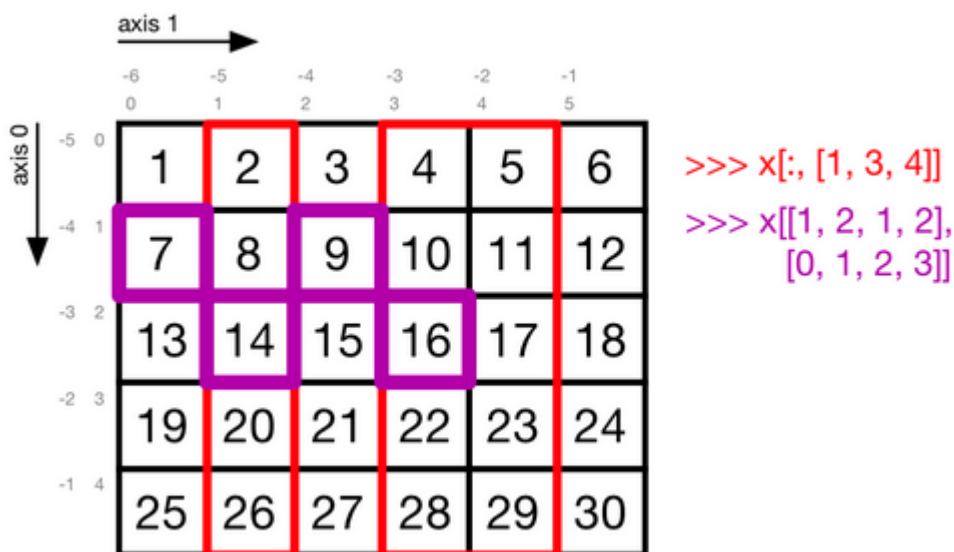
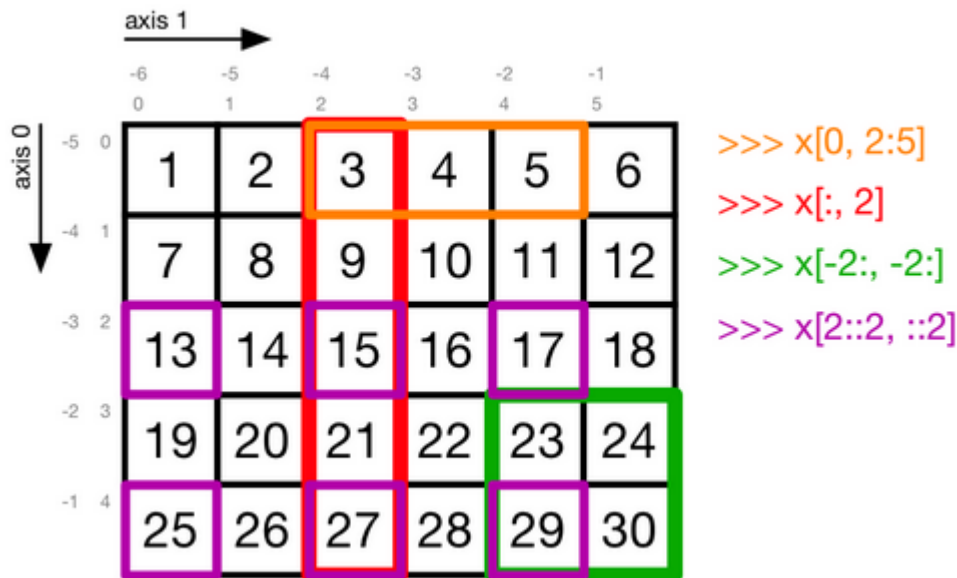
Un tableau a :

- un type `arr.dtype` (les types sont dispo dans la lib `np`)
- un nombre de dimensions `arr.ndim`
- un nombre d'éléments `arr.size`
- une forme `arr.shape` (par exemple `(2, 3, 4)` c'est un tableau avec 2 matrices, qui elles-même contiennent 3 matrices qui elles-même contiennent 4 éléments)

--> On peut changer la forme d'un tableau après sa création, la forme c'est juste la façon dont on interprète le tableau.

Indexing/Slicing

Comme en Python, on précise `debut:fin:pas` pour chaque dimension.



Concatenation/Stacking

```
# Concatenate a couple of matrices vertically.
m1 = np.array([[1, 2, 3], [4, 5, 6]])
m2 = np.array([[10, 20, 30], [40, 50, 60], [70, 80, 90]])
m3 = np.array([[100, 200, 300]])
m_cat = np.concatenate([m1, m2, m3], axis=0)
print(m_cat)

#[[ 1  2  3]
# [ 4  5  6]
# [10 20 30]
# [40 50 60]
# [70 80 90]
# [100 200 300]]
```

