

Enseignant : Jaafar Chaaouri


Email: Jaafar.chaaouri@fsm.rnu.tn

▼ TP1 : NumPy/Matplotlib

```
%matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
```

▼ 1. Manipulations simples avec Numpy

```
A = np.arange(15).reshape(3, 5)
print(A)
```



```
[[ 0  1  2  3  4]
 [ 5  6  7  8  9]
 [10 11 12 13 14]]
```

▼ 1.1 Retournez la sous-matrice `[[6, 7], [11, 12]]`

votre réponse

Étudiez la fonction [np.sum](#). On va s'intéresser à l'argument `axis=` qui joue un rôle important pour la manipulation des tableaux.

▼ 1.2 Afficher la somme des lignes de A

(un vecteur de 5 éléments)

votre réponse

▼ 1.3 Afficher la somme des colonnes de A

(un vecteur de 3 éléments)

votre réponse

▼ 1.4 Afficher la somme de tous les éléments de A

(un nombre)

votre réponse

▼ 1.5 Extraire tous les nombre impairs de A

Résultat recherché : `array([1, 3, 5, 7, 9, 11, 13])`

votre réponse

▼ 1.6 Remplacer tous les éléments impairs de A par -1 sans modifier A :

sortie attendue :

```
[[ 0 -1  2 -1  4]
 [-1  6 -1  8 -1]
 [10 -1 12 -1 14]]
```

votre réponse

▼ 2. Numpy : quelques manipulations avec des jeux de données

▼ 2.1 Un jeu de données artificielles

Dans le répertoire `data/` se trouvant à la racine de ce dépôt il y a le fichier `kms-dataset2d-X.csv`

votre réponse

▼ 2.1.1 À l'aide de la fonction [loadtxt](#), charger le fichier dans la variable `data` de type tableau de NumPy.

votre réponse

▼ 2.1.2 Afficher sa forme (nombre de lignes, nombre de colonnes), puis la somme des éléments de la première colonne.

votre réponse

▼ 2.2 Le jeu de données **iris**

Le jeu de données iris (https://en.wikipedia.org/wiki/Iris_flower_data_set) décrit un échantillon de 150 fleurs de différentes espèces d'iris par 4 attributs mesurés en cm : petal length, petal width, sepal length, sepal width, la dernière colonne étant le nom de l'espèce.

Le fichier `iris.csv` se trouve dans le répertoire `data` de ce dépôt.

▼ 2.2.1 Charger les données du fichier `iris.csv` dans un tableau NumPy et l'affecter à la variable `iris`.

Que se passe-t-il si l'on ne précise pas l'argument `delimiter=` ? Comprendre ce qui se passe. Indiquer le bon délimiteur. Essayer de résoudre les problèmes de lectures de ce fichier.

votre réponse

▼ 2.2.2 Afficher les noms des différentes espèces d'Iris dans ce jeu de données (dernière colonne)

votre réponse

▼ 2.2.3 Trouver un moyen de ne garder que les valeurs numériques concernant l'Iris Setosa :

Obtenir un tableau avec des 4 premières dimensions des lignes correspondant à l'Iris Setosa seulement.

votre réponse

▼ 2.2.4 Calculer la moyenne, la médiane et l'écart-type de chacune des dimensions.

votre réponse

Nous verrons comment Pandas permet de manipuler des tableaux avec des données de types différents.

▼ 3. Matplotlib

- ▼ 3.1 Afficher le nuage de points correspondant aux deux premières dimensions du jeu de données Iris (petal length et petal width) pour l'iris Setosa

votre réponse

- ▼ 3.2 Afficher sur le même graphique les nuages de points correspondant aux deux autres espèces, pour les mêmes dimensions.

Ajouter les légendes des axes et des couleurs pour les trois espèces.

votre réponse


▼ 3.2 Demo avec pandas et seaborn

(ce n'est pas un exercice)

```
import pandas as pd
import seaborn as sns

iris2 = sns.load_dataset('iris')
```

```
iris2.head()
```



	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

```
sns.pairplot(data=iris2 ,kind='scatter', hue='species')
```



<seaborn.axisgrid.PairGrid at 0x1a1793e320>

