

TP1 – Rappels Python

Objectif

L'objectif de ce TP est de passer en revue quelques bibliothèques utiles (Numpy, Pandas, Matplotlib, .etc.) pour implémenter un modèle du machine Learning.

Consignes

Utiliser jupyter notebook de Anaconda pour implémenter les codes.

Chaque binôme doit présenter son travail sous forme d'un fichier notebook (TP1.ipynb) et envoyer la version correcte corrigée aux adresses : houda.elbouhissi@univ-bejaia.dz et samia.chibani@univ-bejaia.dz en fournissant les explications nécessaires.

La bibliothèque Numpy

La bibliothèque NumPy est un module servant à manipuler des matrices ou des tableaux à plusieurs dimensions. NumPy dispose aussi de fonctions permettant de générer des tableaux.

Numpy est installé par défaut sur Anaconda, et si vous utilisez un autre éditeur vous pouvez l'installer avec : **pip install numpy**

L'utilisation de fonctionnalités de numpy commence par l'importation de cette bibliothèque :

```
import numpy as np
```

L'emploi de raccourcis (ici np, que vous pouvez modifier, plutôt que numpy) permet de faciliter l'écriture des appels des fonctions de la bibliothèque.

Atelier 1

$$\text{Si } A = \begin{pmatrix} -1 & -1 & -3 \\ 1 & 2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \text{ et } B = \begin{pmatrix} -2 & -1 & -6 \\ 1 & 1 & 3 \\ 0 & 0 & 1 \end{pmatrix}. \text{ Calculer } A \times B$$

A est-elle inversible ? Si c'est le cas, calculer A^{-1} (on arrondira les éléments de A à 10^{-3}).

Atelier 2

Que donne le code suivant ?

```
import numpy
v = numpy.array([1, 2, 3, 4]) * numpy.pi / 4 # numpy.pi = π
w = numpy.sin(v)
print(w)
```

Que font les fonctions de NumPy suivantes : zeros([num]), ones([num]), linspace(debut, fin, num), random.random([num]) ?

Atelier 3

Pour cet atelier, la réponse à chaque question ne doit contenir qu'une seule ligne de code.

- Créer un tableau T1 qui ne contient que les nombres pairs qui se trouvent entre 0 et 50.
- Créer un tableau T2 qui contiennent les cosinus des carrés des éléments de T1.
- Quel est le minimum du tableau T2 ?

- Quel est le nombre de fois que le maximum du tableau T2 est présent dans T2.

La librairie Matplotlib

Matplotlib est une bibliothèque Python qui permet de faire toutes sortes de tracés.

Matplotlib est installé par défaut sur Anaconda, et si vous utilisez un autre éditeur vous pouvez l'installer avec : **pip install Matplotlib**

L'utilisation de fonctionnalités de Matplotlib commence par l'importation de cette librairie :

```
import Matplotlib.pyplot as plt
```

L'emploi de raccourcis (ici plt, que vous pouvez modifier) permet de faciliter l'écriture des appels des fonctions de la librairie.

```
x = [0, 1, 2] # liste des abscisses  
y = [1, -1, 0] # liste des ordonnées  
plt.plot(x, y) # trace y en fonction de x plt.show()  
# affiche la fenêtre du tracé
```

Atelier 1

En utilisant les fonctions linspace et cos de NumPy, tracer la fonction $y = \cos(x)$ sur $[0, 10\pi]$. Quelle est l'influence du nombre de points passé à la fonction linspace ? Grâce à un second appel à la fonction plt.plot avant l'appel à plt.show, superposer le graphe de la fonction $y = \exp(-x/10) \cos(x)$. Toujours avant l'appel à plt.show, ajouter un titre avec plt.title('Le titre de votre choix') et des noms aux axes avec plt.xlabel('x') et plt.ylabel('y=f(x)').

La librairie Pandas

La librairie pandas est une librairie importante pour la manipulation de données sous Python. Pandas implémente la classe DataFrame qui est une structure de table (ligne, colonne), chaque colonne porte un nom et contient un seul type de données.

Pandas est installé par défaut sur Anaconda, et si vous utilisez un autre éditeur vous pouvez l'installer avec : **pip install pandas**

L'utilisation de fonctionnalités de Pandas commence par l'importation de cette librairie :

```
import pandas as pd
```

L'emploi de raccourcis (ici pd, que vous pouvez modifier, plutôt que pandas) permet de faciliter l'écriture des appels des fonctions de la librairie.

Atelier 1

Ce premier atelier consiste à manipuler les données du covid à l'aide de la librairie Pandas. Les données à récupérer « covid-hospit-incid-reg-2023-03-31-18h01.csv » sont disponibles à l'adresse <https://www.data.gouv.fr/fr/datasets/donnees-hospitalieres-relatives-a-lepidemie-de-covid-19/>

1. En utilisant la fonction tail (ou head), visualiser la structure du tableau. Quelles sont les différentes données et leur type (utiliser dtypes).
2. Les dates sont considérées comme des chaînes de caractères. Il est plus simple pour réaliser des opérations de convertir la colonne sous forme de dates (utiliser to_datetime).
3. On supprime les colonnes relatives aux départements et au sexe puis on agrège par jour.
4. Maintenant, tracer les données (utiliser l'option logy pour une échelle logarithmique).
5. Refaire le même graphique pour votre sexe.

Atelier 2

1. Du même lien de l'atelier 1, récupérez le fichier `temperature.csv`.
2. Utiliser la fonction `describe` sur le dataframe. Que fait cette fonction ?
3. Créer un nouveau dataframe ne contenant que les mois de mars, juin, septembre et décembre et en supprimant les villes de la région 'East'.
4. Récupérer les données sous numpy et calculer la moyenne des températures pour chaque mois. Déterminer aussi la matrice de corrélation entre les 4 mois de l'année.

La librairie Seaborn

Seaborn permet de réaliser un graphique similaire avec Seaborn, une bibliothèque un peu plus élaborée pour Python. Seaborn s'appuie sur Matplotlib et simplifie certains types de graphiques liés à des statistiques. Nous pouvons utiliser la fonction **distplot** pour tracer un histogramme avec une estimation de densité de noyau. Une estimation de densité de noyau est une courbe – qui est en réalité une version lissée de l'histogramme plus facile à analyser.

Seaborn est installé par défaut sur Anaconda, et si vous utilisez un autre éditeur vous pouvez l'installer avec : **pip install seaborn**

L'utilisation de fonctionnalités de Seaborn commence par l'importation de cette librairie :

```
import seaborn as sn
```

L'emploi de raccourcis (ici `sn`, que vous pouvez modifier, plutôt que `seaborn`) permet de faciliter l'écriture des appels des fonctions de la librairie.

Pour les deux ateliers suivants, nous utiliserons le jeu de données : <https://www.data.gouv.fr/fr/datasets/impot-de-solidarite-sur-la-fortune-impot-sur-la-fortune-immobiliere-par-collectivite-territoriale/>, choisir l'ISF 2017, puis accéder au contenu avec la flèche.

Atelier 1

A partir de la table `isf`, tracé le diagramme à barres verticales représentant les nombre de communes parmi les villes de plus de 20 000 habitants ayant plus de 50 redevables à l'ISF.

Atelier 2

A partir de la table `isf`, représenter l'impôt moyen selon le patrimoine moyen (pour les villes de plus de 20 000 habitants ayant plus de 50 redevables à l'ISF).