

TD5 : Les tableaux `numpy`

Xavier André & Romain Tavenard

1 Préambule

Comme pour tous les TD ce semestre, votre code pour ce TD devra se trouver dans un seul fichier, cette fois nommé `TD5_numpy.py` et stocké dans le même répertoire `M1_S1_Python`. La structure de ce fichier devra être la suivante :

```
1 # Section 1 : Imports de module
2
3 # Section 2 : Définition de fonctions
4
5 # Section 3 : Tests de fonctions définies et manipulations en mode "
    script"
```

Il est à noter qu'une partie des questions proposées ici est inspirée de l'excellent "100 `numpy` exercises" par Nicolas Rougier (accessible [ici](#)). N'hésitez pas à aller vous mesurer aux défis qui y sont proposés de temps en temps, cela ne pourra vous faire que du bien :).

2 Définition de structures `numpy`

1. Définissez un vecteur `v` (de type `numpy.array`) d'entiers de taille 4 qui ne contienne que des 0.
2. Définissez une matrice `I_4` qui soit la matrice identité de $\mathcal{M}_{4 \times 4}$.
3. Définissez (sans entrer chaque valeur à la main) une matrice `M` ayant le contenu suivant :

$$M = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 \\ 8 & 9 & 10 & 11 \end{pmatrix} \quad (1)$$

4. Définissez (sans entrer chaque valeur à la main) une matrice `M_T` ayant le contenu suivant :

$$M = \begin{pmatrix} 0 & 4 & 8 \\ 1 & 5 & 9 \\ 2 & 6 & 10 \\ 3 & 7 & 11 \end{pmatrix} \quad (2)$$

5. Définissez une matrice `M2` d'entiers de taille 10×10 ayant des 1 sur tout le tour et des 0 à l'intérieur (BONUS si vous parvenez à définir `M2` en seulement 2 lignes de code).
6. Générez une matrice aléatoire de taille 10×10 en tirant selon une loi normale centrée réduite et standardisez-la pour que ses valeurs soient toutes comprises entre 0 et 1.

3 Calculs

- Effectuez le produit matriciel $M \times I_4$, puis le calcul $M \times I_4 + v$.
- Calculez les sommes ligne par ligne de la matrice `M` (vous devriez obtenir un vecteur de longueur 3). Vérifiez que ce vecteur est égal au vecteur des sommes en colonne de la matrice `M_T`.
- Soient les tableaux `x` et `y` définis comme :

```
1 x = np.array([0.1, 0.2, 0.4])
2 y = np.array([1., 2., 8.])
```

Calculez la matrice de Cauchy `C` définie comme

$$C_{ij} = \frac{1}{x_i - y_j} .$$

- Générez une matrice aléatoire de taille 10×3 et standardisez-la pour que la moyenne de chacune de ses lignes soit égale à 0.

4 Exercice de synthèse

Cet exercice est issu du polycopié associé au cours.

Supposons qu'on ait stocké dans le tableau suivant les notes reçues par 2 étudiants à 3 examens :

```
1 notes = np.array(
2     [[10, 12],
3      [15, 16],
4      [18, 12]]
5 )
```

1. Calculez la moyenne de chacun des deux étudiants.
2. Calculez le nombre de notes supérieures à 12 contenues dans ce tableau