Matrices et introduction au traitement d'images — TP12

1 Retour sur les matrices

Les deux exercices ci-dessous sont issus du TP précédent. Si vous les avez déjà faits passez directement à la partie 2 de ce TP.

Exercice 1 — Agréable inflation

Pour évaluer l'inflation, on sélectionne une liste de produits dans un magasin et on note leurs prix de vente une première fois en janvier 2023 et une deuxième fois en janvier 2024. On stocke les résultats dans un tableau `prix` où chaque colonne correspond à un produit et où la première ligne indique les prix en janvier 2023 et la deuxième les prix en janvier 2024.

Q1 Écrire une fonction `inflation` prenant en argument un prix x et un prix y et renvoyant le pourcentage d'augmentation entre x et y. Par exemple, `inflation(10,12)` doit renvoyer 20 car passer de 10 euros à 12 euros correspond à une augmentation de 20%.

Q2 Écrire une fonction `list_inflation` prenant en argument le tableau contenant les prix relevés en 2023 et en 2024 et renvoyant la liste des pourcentages d'augmentation des prix des articles sélectionnés. On vérifiera par exemple que si `prix` $= \begin{pmatrix} 10 & 5 & 2 & 4 \\ 12 & 7,5 & 4 & 3 \end{pmatrix}$ alors `list_inflation(prix)` renvoie `[20,50,100,-25]`.

Pour estimer l'inflation globale, on peut envisager deux méthodes de calcul.

Q3 La première méthode consiste à faire la moyenne des pourcentages d'augmentation de chacun des articles sélectionnés. Écrire une fonction `inflation_1` prenant en argument le tableau `prix` et renvoyant l'inflation calculée selon cette méthode.

Q4 La deuxième méthode consiste à calculer le pourcentage d'augmentation du total du panier, c'est-à-dire de la somme des prix des articles sélectionnés. Écrire une fonction `inflation_2` prenant en argument le tableau `prix` et renvoyant l'inflation claculée selon cette méthode.

Les deux méthodes mènent-elles au même résultat ? Laquelle vous semble la plus réaliste ?

Exercice 2 — Avec des booléens

Q1 Dans la console, définissez une matrice A de votre choix. Que renvoie la commande `A == A` ?

Q2 Dans la console, définissez une matrice B de taille différente de A. Que renvoie la commande `A == B` ?

Q3 Écrire une fonction `egal` prenant en arguments deux matrices A et B et renvoyant `True` si $A = B$ et `False` sinon.

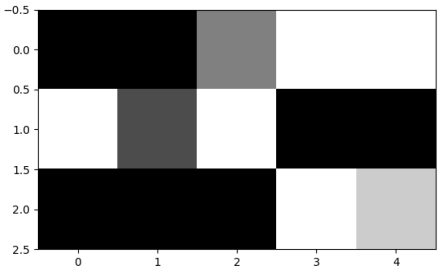
2 Images

Les matrices sont un moyen de coder des images en Python. Chaque coefficient de la matrice correspond à un pixel et indique la couleur du pixel. Dans ce TP on se limite à des images en noir et blanc, on utilise pour cela des coefficients compris entre 0 et 1 avec la convention : 0 = noir, 1 = blanc (et toute valeur comprise entre 0 et 1 correspond à une nuance de gris). D'autres façons de coder les images existent cependant [1].

1. On peut par exemple manipuler des images dont les pixels peuvent prendre des valeurs entières entre 0 et 255 (0 pour noir, 255 pour blanc). On peut aussi manipuler des images en couleurs grâce au code "RGB" (rouge, vert, bleu), on utilise alors

Voici un exemple de correspondance entre matrice et image :

$$\begin{pmatrix} 0 & 0 & 0,5 & 1 & 1 \\ 1 & 0,3 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0,8 \end{pmatrix} \text{ correspond à }$$



Pour afficher l'image correspondant à une matrice `matrice` on utilisera les commandes suivantes faisant appel à la bibliothèque `matplotlib.pyplot` :

```
plt.imshow(matrice,cmap='gray') # cmap pour color map
plt.show()
```

Exercice 3 Construction d'une image à la main

Q1 Écrire une fonction `nuancier` prenant en arguments deux entiers n et p et renvoyant la matrice M de taille $n \times p$ dont les coefficients sont données par :

$$\forall i \in [\![0, n-1]\!],\ \forall j \in [\![0, p-1]\!],\ M_{i,j} = \frac{i+j}{n+p-2}.$$

Q2 Afficher à l'écran l'image correspondant à la matrice `nuancier(10,5)`. Afficher ensuite celle correspondant à la matrice `nuancier(100,50)`. Commenter.

Exercice 4 Préparation pour le traitement

Une image test vous est fournie sur cahier de prépa, il s'agit du fichier `panda.jpg`, téléchargez-le et enregistrez-le dans un dossier personnel.

Pour pouvoir utiliser ce fichier dans Python, il faut l'importer en précisant son emplacement, c'est-à-dire la suite de dossiers à ouvrir pour accéder à ce fichier. On utilise pour cela la bibliothèque `matplotlib.image` :

```
import matplotlib.image as mpimg # Pour ouvrir et manipuler des images
```

La commande `imread` permet alors d'importer l'image dans Python sous forme d'un tableau de réels. Cette commande prend en argument une chaîne de caractères, correspondant à l'emplacement du fichier à ouvrir, et renvoie le tableau des valeurs des pixels de l'image. Par exemple la commande

```
mon_image_RGB = mpimg.imread(r"\\SRV-DC\ccaillaud$\Documents\panda.jpg")
```

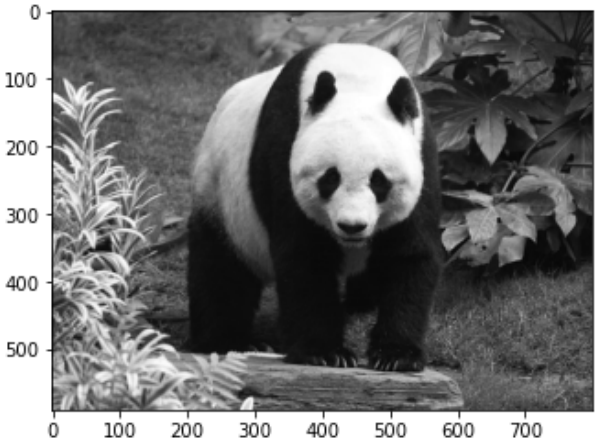
permet d'ouvrir le fichier `panda.jpg` si l'utilisateur `ccaillaud` l'a rangé dans son dossier `Documents`. Pour récupérer l'emplacement de votre fichier, utilisez l'explorateur de fichier et faites "clic droit" sur le fichier en question. Copiez alors son emplacement pour créer votre chaîne de caractères. Attention, il faut également faire précéder cette chaîne d'un `r`.

La fonction `imread` crée naturellement un tableau permettant de stocker une image en couleur, comportant donc plus de dimensions que nécessaire pour notre TP. On utilise donc la commande `mon_image = mon_image_RGB[:,:,0]` pour récupérer simplement un tableau de taille "$n \times p$" comme manipulé dans l'exercice précédent.

3 images en parallèle (codant les niveau de rouge, vert, et bleu) qu'on associe pour recomposer la couleur de chaque pixel.

Pour afficher une image, on utilise la commande `imshow`, avec l'argument `cmap='gray'` (pour préciser qu'il s'agit d'une image en niveaux de gris) puis `show`.

```
mon_image = mon_image_RGB[:,:,0] # pour avoir un simple tableau n x p
plt.imshow(mon_image, cmap='gray') # dessiner l'image
plt.show() # afficher l'image
```

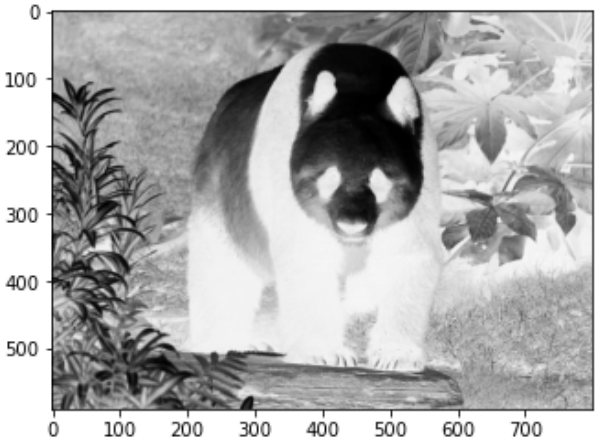


Vous devez obtenir le résultat ci-contre.

Q1 Souhaitez-vous vraiment continuer à appeler cette image `mon_image` pour tout le reste du TP ? Choisissez un nom parlant et simple.

Q2 Quelle est la taille de cette image ? Combien compte-t-elle de pixels en tout ?

Exercice 5 Négatif



On souhaite inverser les couleurs de l'image afin d'obtenir son négatif.

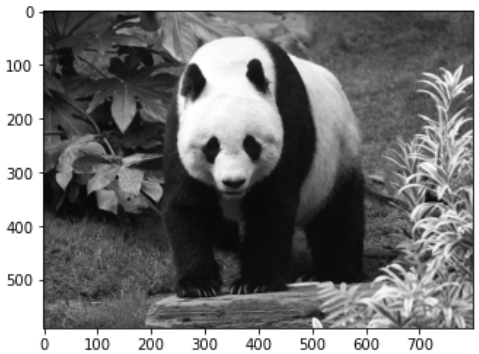
Q1 Écrire une fonction `pixel_negatif` prenant en argument la valeur d'un pixel correspondant à un niveau de gris et renvoyant la valeur correspondant à sa couleur négative.

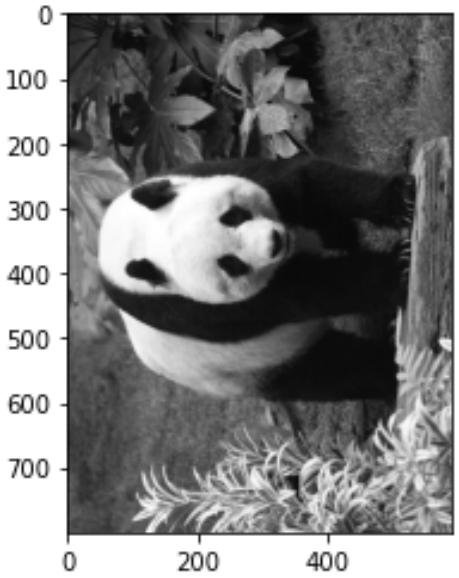
Q2 Écrire et tester une fonction `negatif` prenant en argument une image quelconque et renvoyant son négatif. Pour notre panda, on obtient l'image ci-contre.

Exercice 6 Miroir et rotation

Q1 Écrire une fonction `miroir` permettant de retourner une image horizontalement. Vous devez obtenir le résultat de gauche ci-dessous.

Q2 Écrire une fonction `rotation90gauche` renvoyant une image ayant subit une rotation de 90° dans le sens anti-horaire (et sans effet miroir !). Vous devez obtenir le résultat de droite ci-contre.





Exercice 7 Un peu de dessin

Dans ce dernier exercice, on propose de dessiner des figures géométriques dans des images. On produira des images carrées de taille $N \times N$ qui représenteront des figures comprises dans le carré $[-2, 2] \times [-2, 2]$ de $\mathbb{R}^2$.

Les figures seront dessinées sur une variable image globale, et la taille N sera aussi une variable globale, c'est-à-dire qu'on écrira en début de script :

```
N = 1000
dessin = np.zeros((N,N))
```

Afin d'utiliser les équations algébriques usuelles des figures géométriques telles que le disque, il faut être en mesure de transformer les indices d'un tableau en abscisses et ordonnées. Par exemple, le pixel d'indices $[N/2, N/2]$ au centre de l'image correspond à l'origine du repère $(0, 0)$. On admet qu'un indice `k` $\in [\![0, N-1]\!]$ correspond à une abscisse ou une ordonnée `abs_ord(k)` où `abs_ord` est la fonction suivante, que l'on recopiera :

```
def abs_ord(k):
    return (k-N/2)*(4/N)
```

Q1 Écrire une fonction `disque` permettant de dessiner un disque de centre (x_A, y_A) et de rayon r. Par exemple, avec $x_A = y_A = 0$ et $r = 1$, on doit obtenir l'image ci-dessous à gauche.

Q2 Reproduire le dessin de rosace ci-dessous à droite.