

Code de partage avec Capytale : ded2-1732010

## Préambule - manipuler des matrices

Voici quelques commandes essentielles pour manipuler des matrices, à tester :

```
import numpy as np
A=np.array([[0,1],[1,0]]); B=np.array([[1,2],[3,4]]) ; C=np.array
([[1,2,8],[3,4,9]])
A+B; A+C; np.dot(A,B); np.dot(A,C); np.dot(C,A);
A[1,:];C[:,2]
np.sum(C[1,:])
import numpy.linalg as al
al.matrix_power(A,3);al.matrix_power(A,2)
```

**Exercice 1** - identités remarquables et puissances

On considère les matrices  $A = \begin{pmatrix} 0 & 0,5 & 0 \\ 1 & 0 & 1 \\ 0 & 0,5 & 0 \end{pmatrix}$  et  $B = \begin{pmatrix} 0 & 1 & 5 \\ -1 & -2 & 3 \\ 1 & 5 & 3 \end{pmatrix}$

1. Définir ces matrices avec Python
2. (a) Avec Python, calculer ensuite  $(A+B)^2$  puis  $A^2 + 2AB + B^2$  et commenter le résultat.  
(b) Les matrices  $A$  et  $B$  vérifient-elles d'autres identités remarquables ?
3. Avec Python, calculer  $A^2, A^3$  et  $A^4$  puis émettre une conjecture sur  $A^n$

## Quelques compléments avec linalg

**Exercice 2** - échauffement

On pose  $A = \begin{pmatrix} 1 & 2 \\ 0 & -1 \end{pmatrix}$  et  $B = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$

1. Définir les matrices  $A$  et  $B$  avec Python.
2. Tester la commande `al.inv(A)` et commenter le résultat renvoyé.
3. Tester la commande `al.solve(A,B)` et commenter le résultat renvoyé.

**Exercice 3** - Déterminer une équation de parabole

- On considère une parabole d'équation  $y = ax^2 + bx + c$  où  $a, b$  et  $c$  sont trois réels avec  $a \neq 0$
- On suppose que la parabole passe par les point  $A(1; 4)$ ,  $B(-2; -5)$  et  $C(3; 0)$
- On cherche à déterminer les trois réels  $a, b$  et  $c$

1. Justifier que  $a, b$  et  $c$  sont solutions du système : 
$$\begin{cases} a + b + c &= 4 \\ 4a - 2b + c &= -5 \\ 9a + 3b + c &= 0 \end{cases}$$
2. Avec Python, définir la matrice  $A$  associée au système et la matrice  $B$  second membre du système.
3. Avec Python, montrer que  $A$  est inversible et trouver son inverse.
4. Avec Python, résoudre le système en utilisant  $A^{-1}$
5. Avec Python, résoudre le système en utilisant la fonction `solve`