

DM 1. Simulation Numérique

Modules de calcul scientifique : Numpy et Matplotlib

0.1 Calcul Matriciels

Exercice 1 : Addition de matrices creuses

Un enjeu scientifique et technologique actuel est de savoir traiter des problèmes mettant en jeu un nombre très important de données. Un point crucial est souvent de pouvoir manipuler des matrices de très grandes dimensions, ce qui est a priori très coûteux, en temps de calcul et en mémoire. On peut cependant souvent considérer que les matrices manipulées ne contiennent que " peu " d'éléments non nuls : c'est ce que l'on appelle les matrices *creuses*.

Nous nous intéressons ici à l'implémentation de deux algorithmes d'addition de matrices : l'un pour une représentation classique des matrices, l'autre pour une représentation des matrices creuses.

Soit $n \in \mathbb{N}^*$, on note $M_n(\mathbb{R})$ l'ensemble des matrices carrées d'ordre n , à coefficients dans $\mathbb{R}$.

On représentera classiquement une matrice $M \in M_n(\mathbb{R})$ par un tableau à double entrées. En Python, cela sera un tableau de longueur n , chaque élément de ce tableau représentant une ligne de M . Chaque élément de ce tableau est donc un tableau de longueur n , dont tous les éléments sont des nombres (types `int` ou `float`).

Cette même matrice M sera représentée de manière creuse en ne décrivant que ses cases non vides par un tableau de triplets (i, j, x) , où x est l'élément de M situé sur la i^e ligne et la j^e colonne. On pourra supposer que les éléments non nuls de M sont ainsi décrits ligne par ligne.

Exemple : La matrice $\begin{pmatrix} 1 & 0 & 5 \\ 0 & -2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ sera représentée classiquement par le tableau `[[1,0,5],[0,-2,0],[0,0,0]]` et de manière creuse par le tableau `[(0,0,1),(0,2,5),(1,1,-2)]`.

- Q1 Écrire une fonction `add(M,N)` prenant en argument deux représentations classiques M et N de deux matrices M et N (carrées, de même ordre) et renvoyant la représentation classique de $M + N$. On prendra soin d'écrire une fonction " optimale " en terme de complexité, spatiale et temporelle.
- Q2 Écrire une fonction `add_creuse(M,N)` prenant en argument deux représentations creuses M et N de deux matrices M et N (carrées, de même ordre) et renvoyant la représentation creuse de $M + N$. On prendra soin d'écrire une fonction " optimale " en terme de complexité, spatiale et temporelle.
Pour simplifier, on ne justifiera pas que les éléments obtenus sont disposés dans le bon ordre.
- Q3 On suppose que M et N sont carrées, d'ordre n , représentées classiquement par M et N . Évaluer asymptotiquement la complexité temporelle de la fonction `add(M,N)`.
- Q4 On suppose que M et N sont carrées et contiennent chacune au plus p éléments non nuls, représentées de manière creuse par M et N . Évaluer asymptotiquement la complexité temporelle de la fonction `add_creuse(M,N)`.
- Q5 Discuter du choix de la représentation pertinente à utiliser pour additionner deux matrices.