TD – Algèbre linéaire

Introduction et objectifs

Dans ce TD consacré à l'algèbre linéaire, nous revenons sur quelques thèmes classiques autour du calcul matriciel.

On utilisera les matrices via la bibliothèque `numpy` et, plus particulièrement, le module `linalg` pour les éventuelles « manipulations » comme l'obtention des valeurs propres d'une matrice carrée ou les calculs de déterminants.

Factorisation de Cholesky

Présentation

Une matrice symétrique définie positive (SDP) *M* peut être écrite comme produit d'une matrice triangulaire inférieure (*L*, cette lettre correspondant à l'anglais « lower ») et de sa transposée (${}^tL$, qui est donc triangulaire supérieure) : $M = L \times {}^tL$. C'est la factorisation de Cholesky.

En imposant aux termes diagonaux d'être strictement positifs (ils sont tous non nuls puisque le déterminant de *M*, qui est strictement positif (*M* étant SDP), est le carré du produit des coefficients diagonaux de *L* (ou de ${}^tL$ puisqu'ils sont identiques)), cette factorisation est unique. La matrice *L* (ou sa transposée) alors obtenue peut être considérée comme une racine carrée de la matrice *M*.

Pour montrer que $M \in \mathscr{M}_n(\mathbb{R})$ est SDP, on peut :

- Calculer les valeurs propres de *M* et montrer qu'elles sont toutes strictement positives. On prendra garde aux situations où toutes ou partie de ces valeurs propres sont proches de 0, les approximations de calcul pouvant conclure à la nullité erronée d'une ou plusieurs valeurs propres non nulles.
- Utiliser le critère de Sylvester :

$$M \text{ est SDP} \Leftrightarrow \forall p \in [\![1\,;n]\!], \det M_p > 0 \text{ où } M_p = \left(m_{ij}\right)_{(i,j)\in[\![1\,;\,p]\!]^2}$$

  Cette méthode offre l'avantage de pouvoir être mise en œuvre facilement.

Exercice N°1 – Mise en œuvre

Ecrire une fonction `Cholesky` qui reçoit en argument une matrice `M` de $\mathcal{M}_n(\mathbb{R})$ et qui :

- Vérifie que la matrice `M` est bien SDP.
- Renvoie la matrice triangulaire inférieure $L = (l_{ij})_{(i,j)\in[\![1\,;n]\!]^2}$ telle que :

$$M = L \times {}^tL \text{ et } \forall k \in [\![1\,;n]\!], l_{kk} > 0$$

On calcule les coefficients de L <u>colonne par colonne</u> et on vérifiera que l'on a, pour toute matrice $M \in \mathcal{M}_n(\mathbb{R})$ SDP d'ordre supérieur ou égal à 2 :

- $l_{11} = \sqrt{m_{11}}$ et $\forall i \in [\![2\,;n]\!], l_{i1} = \dfrac{m_{i1}}{l_{11}}$.
- Puis, pour tout entier $k \in [\![2\,;n]\!]$:

$$l_{kk} = \sqrt{m_{kk} - l_{k1}^2 - l_{k2}^2 - \ldots - l_{kk-1}^2} = \sqrt{m_{kk} - \sum_{j=1}^{k-1} l_{kj}^2}$$

$$\forall i \in [\![k+1\,;n]\!], l_{ik} = \frac{1}{l_{kk}}\left(m_{ik} - l_{k1}l_{i1} - l_{k2}l_{i2} - \ldots - l_{kk-1}l_{ik-1}\right) = \frac{1}{l_{kk}}\left(m_{ik} - \sum_{j=1}^{k-1} l_{kj}l_{ij}\right)$$

On testera la fonction avec les matrices suivantes :

$$\begin{pmatrix} 2 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{pmatrix} \quad \begin{pmatrix} 4 & 1 & 1 \\ 1 & 4 & 1 \\ 1 & 1 & 4 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 \\ 1 & 2 & 3 & 3 \\ 1 & 2 & 3 & 4 \end{pmatrix}$$

Retour sur le produit matriciel

Généralités

Considérons le produit matriciel des deux matrices A et B carrées d'ordre 2 :

$$\mathrm{A} \times \mathrm{B} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \times \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} = \begin{pmatrix} a\alpha + b\gamma & a\beta + b\delta \\ c\alpha + d\gamma & c\beta + d\delta \end{pmatrix}$$

On doit donc effectuer 8 multiplications et 4 additions.
En ne retenant que les multiplications pour mesurer la complexité du calcul et en notant celle-ci $c(2)$, on a donc : $c(2) = 8$.

De façon générale, le produit de deux matrices carrées d'ordre *n* requiert un total de n^3 multiplications puisque le calcul de chacun des n^2 coefficients requiert *n* multiplications.

La méthode de Volker STRASSEN

Revenons au cas initial des matrices carrées d'ordre 2.
Le mathématicien allemand Volker STRASSEN (1936-) a proposé en 1969 d'effectuer la multiplication des matrices A et B de la façon suivante :

- Calculer : $m_1 = (a+d)\times(\alpha+\delta)$, $m_1 = (c+d)\times\alpha$, $m_3 = a\times(\beta-\delta)$, $m_4 = d\times(\gamma-\alpha)$, $m_5 = (a+b)\times\delta$, $m_6 = (c-a)\times(\alpha+\beta)$ et $m_7 = (b-d)\times(\gamma+\delta)$.
- Calculer alors :
$$\mathrm{A}\times\mathrm{B} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}\times\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} = \begin{pmatrix} m_1+m_4-m_5+m_7 & m_3+m_5 \\ m_2+m_4 & m_1-m_2+m_3+m_6 \end{pmatrix}$$

On ne doit plus effectuer que 7 multiplications…

Si nous travaillons maintenant avec des matrices A et B carrées d'ordre 4, nous pouvons nous ramener à la situation précédente en effectuant une multiplication par blocs :

$$\mathrm{A}\times\mathrm{B} = \begin{pmatrix} \mathrm{A}_{11} & \mathrm{A}_{12} \\ \mathrm{A}_{21} & \mathrm{A}_{22} \end{pmatrix}\times\begin{pmatrix} \mathrm{B}_{11} & \mathrm{B}_{12} \\ \mathrm{B}_{21} & \mathrm{B}_{22} \end{pmatrix} = \begin{pmatrix} \mathrm{M}_1+\mathrm{M}_4-\mathrm{M}_5+\mathrm{M}_7 & \mathrm{M}_3+\mathrm{M}_5 \\ \mathrm{M}_2+\mathrm{M}_4 & \mathrm{M}_1-\mathrm{M}_2+\mathrm{M}_3+\mathrm{M}_6 \end{pmatrix}$$

où les A_{ij} et les B_{ij} sont des matrices carres d'ordre 2.

On doit donc effectuer 7 produits matriciels de matrices carrées d'ordre 2. Le nombre total de multiplications vaudra donc cette fois : $c(4) = 7\times c(2) = 7\times 7 = 49$.

A chaque fois que l'on doublera l'ordre de la matrice, la complexité sera multipliée par 7. En d'autres termes : $c(2n) = 7\times c(n)$.

Pour une matrice d'ordre $n = 2^p$ avec $p \geq 1$, on a classiquement :

$$c(n) = c(2^p) = 7^{p-1}\times c(2) = 7^{p-1}\times 7 = 7^p = 7^{\log_2(n)} = 7^{\frac{\ln n}{\ln 2}} = e^{\ln 7\times\frac{\ln n}{\ln 2}} = n^{\frac{\ln 7}{\ln 2}} \simeq n^{2{,}807}$$

Le gain peut ne pas sembler spectaculaire mais il est déterminant pour de grandes valeurs de *n* (il n'est pas rare que les calculs techniques mettent en œuvre des matrices carrées dont l'ordre de grandeur de *n* est 10 000, 100 000 voire plus).

Exercice N°2 – Mise en œuvre

Ecrire une fonction récursive `MulStrassen` permettant d'effectuer, selon la méthode décrite ci-dessus, la multiplication de deux matrices carrées d'ordre $n = 2^p$ avec $p \geq 1$.
La fonction `MulStrassen` s'assurera que les ordres des matrices passées en argument sont tous deux égaux à une même puissance entière de 2.