



PROJET
Analyse de Données / Calcul Scientifique :
A quoi tu ressembles sans ton masque ?

DAI Guohao
DANTON LALOY Calliopé
DE ROCKER Tom

Département Sciences du Numérique - Première année
2020-2021

1 Séance 2

Question 1

Comparer la fonction *eig* de matlab calculant l'ensemble des couples vecteurs propres - valeurs propres à la fonction basée sur la méthode la puissance itérée *test_v11.m* calculant les n premiers couples permettant d'obtenir un certain pourcentage de la trace.

Taille 200 × 200

Type 1	temps	meilleure vp	pire vp	meilleure cpe	pire cp
puissance itérée	1.87	0	10^{-14}	10^{-9}	10^{-8}
eig	1×10^{-2}	0	10^{-14}	10^{-16}	10^{-13}

Type 2	temps	meilleure vp	pire vp	meilleure cpe	pire cp
puissance itérée	1.2×10^{-1}	10^{-16}	10^{-15}	10^{-9}	10^{-8}
eig	1×10^{-2}	0	10^{-8}	10^{-16}	10^{-6}

Type 3	temps	meilleure vp	pire vp	meilleure cpe	pire cp
puissance itérée	2.3×10^{-1}	0	10^{-15}	10^{-9}	10^{-8}
eig	3×10^{-2}	0	10^{-11}	10^{-16}	10^{-11}

Type 4	temps	meilleure vp	pire vp	meilleure cpe	pire cp
puissance itérée	1.6	0	10^{-14}	10^{-9}	10^{-8}
eig	1×10^{-2}	0	10^{-14}	10^{-16}	10^{-14}

Taille 1000 × 1000

Type 2	temps	meilleure vp	pire vp	meilleure cpe	pire cp
puissance itérée	8.56	0	10^{-14}	10^{-9}	10^{-8}
eig	2.7×10^{-1}	0	10^{-7}	10^{-16}	10^{-6}

Question 2

Réarrangement de l'algorithme de la puissance itérée pour avoir un meilleur temps de calcul :

Algorithm 1 Méthode de la puissance itérée améliorée

$\beta = v^T \cdot A \cdot v$

repeat

$$v = \frac{z}{\|z\|_2}$$

$$z = AV$$

$$\beta = v^T \cdot z$$

$$norme = \frac{\|\beta \cdot v - z\|_2}{\|\beta\|_2}$$

until ($norme > eps$)

Question 3

Quel est selon vous le principal inconvénient de la méthode de la puissance itérée en termes de temps de calcul ?

La puissance itérée calcule de nombreux produits matrice-vecteur pour chaque valeur propre calculée. Cela pose de gros problèmes de temps de calcul, le produit matrice-vecteur étant coûteux ($O(n^2)$)

Question 4

On se demande vers quelle matrice converge V , une matrice de m colonnes orthogonales, lorsque'on lui applique l'algorithme de la puissance itérée (c'est-à-dire sans l'orthonormaliser à chaque itération).

V va converger vers une matrice de rang 1 et de dimension $n \times m$, dont chaque colonne correspond au vecteur propre de la plus grande valeur propre divisé par la racine carrée de m .

Question 5

Nous étudions des variantes de la méthode de la puissance itérée afin d'éviter de calculer toute la décomposition spectrale de la matrice A . Dans l'algorithme 2, on calcule la décomposition spectrale complète de la matrice H . On montre que cela n'est pas un problème.

La matrice H possède une dimension bien inférieure à la matrice de base, la décomposition spectrale complète de cette matrice ne pose donc pas de problèmes de complexité.

Question 7

Commentaires de l'algorithme 4 dans *subspace_iter_v1.m* :

Algorithm 2 Méthode d'itération subsapatale v1 avec projection de Raleigh-Ritz

Input : Symetric matrix $A \in R^{n \times n}$, tolerance ϵ , *MaxIter* (max nb of iterations)

and *PercentTrace* the target percentage of the trace of A .

Output : $n_e v$ dominant eigenvectors V_{out} and the corresponding eigenvalues Λ_{out} .

Generate an initial set of m orthonormal vectors $V \in R^{nm}; k = 0; PercentReached = 0$ (line 48-49)

repeat

$k = k + 1$ (line 54)

 Compute Y such that $Y = A \cdot V$ (line 56)

$V \leftarrow$ orthonormalisation of the columns of Y (line 58)

 Rayleigh-Ritz projection applied on matrix A and orthonormal vectors V (line 61)

 Convergence analysis step : save eigenpairs that have converged and update *PercentReached* (line 70-109)

until (*PercentReached* > *PercentTrace* or $n_e v = m$ or $k > MaxIter$)

Question 8

Quel est le coût en FLOPS du calcul de A^p et de $A^p \cdot V$? Comment réduire son coût ?

— Le coût de calcul de A^p est de : $n^3 \times p$

— Le coût de calcul de $A^p \cdot V$ est de : $n^3 \times p \times n^2$

On peut utiliser un algorithme d'exponentiation rapide afin d'effectuer moins de calcul matrice-matrice dans le calcul de A^p (ou autant dans une minorité de cas).

Question 10

Observer et expliquer le comportement de l'algorithme lorsque p augmente.

En utilisant une puissance p de A plus élevée, l'algorithme fait moins d'itérations avant de converger.

Question 11

Expliquer pourquoi la précision de certaines paires diffère pour la méthode `subspace_iter_v1.m`

Au fur et à mesure des calculs de couples, l'erreur va augmenter car chaque itération va accumuler l'erreur des itérations précédentes et la sienne. Ainsi, les couples associés à des vecteurs propres plus petits, calculés à la fin, auront une erreur plus importante.

Question 12

Essayer d'anticiper ce qu'il va se passer en terme de convergence pour la méthode `subspace_iter_v3.m`

Question 14

Différence entre les 4 types de matrice

- Type 1 : $D(i) = n - i + 1$
- Type 2 : aléatoire entre 0 et 1
- Type 3 : exponentielle entre 0 et 1
- Type 4 : presque régulier entre 0 et 1

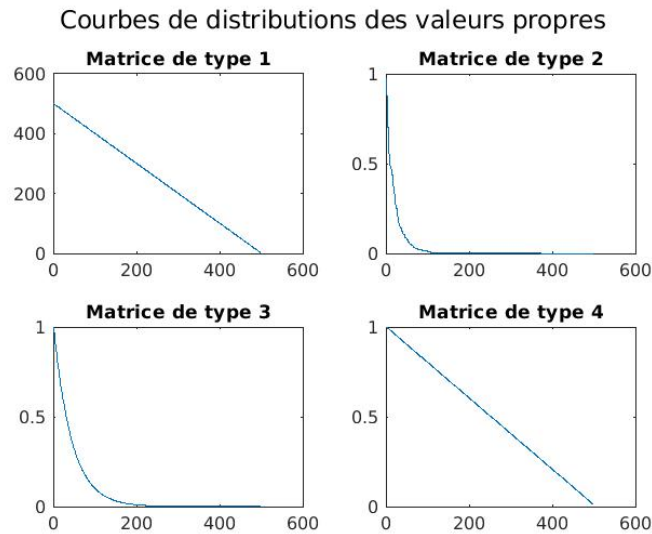


FIGURE 1 – Courbes représentant la distribution des valeurs propres selon le type de matrice

Question 15

Comparer les performances des différents algorithmes.

Type 1

taille 100×100	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	5.3×10^{-1}	10^{-16}	10^{-13}	10^{-8}	10^{-7}
v1	2×10^{-1}	10^{-15}	10^{-14}	10^{-13}	10^{-8}
v2	3×10^{-2}	10^{-16}	10^{-14}	10^{-13}	10^{-8}
v3	4×10^{-2}	10^{-16}	10^{-14}	10^{-9}	10^{-8}
puissance itérée	1×10^{-1}	0	10^{-15}	10^{-9}	10^{-8}
puissance itérée améliorée	7×10^{-2}	0	10^{-15}	10^{-9}	10^{-8}
eig	1×10^{-2}	0	10^{-15}	10^{-16}	10^{-14}

taille 300×300	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	4.2	10^{-16}	10^{-12}	10^{-10}	10^{-7}
v1	1.380	10^{-14}	10^{-13}	10^{-14}	10^{-7}
v2	1.2×10^{-1}	10^{-14}	10^{-13}	10^{-14}	10^{-8}
v3	1.4×10^{-1}	10^{-14}	10^{-13}	10^{-8}	10^{-7}
puissance itérée	6.810	0	10^{-14}	10^{-9}	10^{-8}
puissance itérée améliorée	3.58	0	10^{-14}	10^{-9}	10^{-8}
eig	2×10^{-2}	0	10^{-13}	10^{-16}	10^{-13}

taille 500×500	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	2.840×10^1	10^{-16}	10^{-12}	10^{-11}	10^{-7}
v1	1.720	10^{-13}	10^{-13}	10^{-13}	10^{-7}
v2	1.8×10^{-1}	10^{-13}	10^{-13}	10^{-13}	10^{-7}
v3	2×10^{-1}	10^{-13}	10^{-13}	10^{-8}	10^{-7}
puissance itérée	1.771×10^1	0	10^{-14}	10^{-9}	10^{-8}
puissance itérée améliorée	9.06	0	10^{-14}	10^{-9}	10^{-8}
eig	6×10^{-2}	0	10^{-14}	10^{-16}	10^{-13}

Type 2

taille 100×100	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	5×10^{-2}	0	10^{-12}	10^{-6}	10^{-3}
v1	4×10^{-2}	10^{-16}	10^{-15}	10^{-11}	10^{-10}
v2	5×10^{-2}	10^{-16}	10^{-15}	10^{-15}	10^{-9}
v3	4×10^{-2}	10^{-16}	10^{-15}	10^{-15}	10^{-9}
puissance itérée	1×10^{-2}	10^{-16}	10^{-15}	10^{-9}	10^{-8}
puissance itérée améliorée	1×10^{-2}	10^{-16}	10^{-15}	10^{-9}	10^{-8}
eig	0	0	10^{-7}	10^{-16}	10^{-7}

taille 300×300	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	1.09	0	10^{-12}	10^{-16}	10^{-7}
v1	7×10^{-2}	0	10^{-15}	10^{-12}	10^{-8}
v2	5×10^{-2}	0	10^{-15}	10^{-15}	10^{-8}
v3	5×10^{-2}	0	10^{-15}	10^{-15}	10^{-8}
puissance itérée	3.6×10^{-1}	10^{-16}	10^{-15}	10^{-9}	10^{-8}
puissance itérée améliorée	2×10^{-1}	10^{-16}	10^{-15}	10^{-9}	10^{-8}
eig	3×10^{-2}	0	10^{-8}	10^{-16}	10^{-6}

taille 500×500	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	6.9×10^{-1}	10^{-16}	10^{-14}	10^{-13}	10^{-5}
v1	1.6×10^{-1}	10^{-16}	10^{-15}	10^{-16}	10^{-8}
v2	1.1×10^{-1}	10^{-16}	10^{-15}	10^{-16}	10^{-8}
v3	1×10^{-1}	0	10^{-15}	10^{-14}	10^{-8}
puissance itérée	2.38	10^{-16}	10^{-14}	10^{-9}	10^{-8}
puissance itérée améliorée	1.76	10^{-16}	10^{-14}	10^{-9}	10^{-8}
eig	6×10^{-2}	0	10^{-7}	10^{-16}	10^{-6}

Type 3

taille 100×100	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	1.1×10^{-1}	0	10^{-13}	10^{-8}	10^{-7}
v1	8×10^{-2}	10^{-16}	10^{-15}	10^{-10}	10^{-8}
v2	4×10^{-2}	10^{-16}	10^{-15}	10^{-15}	10^{-9}
v3	5×10^{-2}	10^{-16}	10^{-15}	10^{-15}	10^{-9}
puissance itérée	2×10^{-2}	0	10^{-16}	10^{-9}	10^{-8}
puissance itérée améliorée	1×10^{-2}	0	10^{-16}	10^{-9}	10^{-8}
eig	0	0	10^{-12}	10^{-16}	10^{-11}

taille 300×300	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	5.4×10^{-1}	0	10^{-13}	10^{-9}	10^{-8}
v1	1.6×10^{-1}	10^{-16}	10^{-14}	10^{-15}	10^{-8}
v2	5×10^{-2}	0	10^{-15}	10^{-16}	10^{-8}
v3	8×10^{-2}	0	10^{-15}	10^{-15}	10^{-8}
puissance itérée	7×10^{-1}	0	10^{-15}	10^{-9}	10^{-8}
puissance itérée améliorée	4.9×10^{-1}	0	10^{-15}	10^{-9}	10^{-8}
eig	2×10^{-2}	0	10^{-12}	10^{-16}	10^{-11}

taille 500×500	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	3.22	0	10^{-13}	10^{-11}	10^{-8}
v1	2.1×10^{-1}	10^{-15}	10^{-14}	10^{-13}	10^{-8}
v2	8×10^{-2}	0	10^{-15}	10^{-13}	10^{-8}
v3	7×10^{-2}	0	10^{-14}	10^{-11}	10^{-8}
puissance itérée	1.46	0	10^{-15}	10^{-9}	10^{-8}
puissance itérée améliorée	9.7×10^{-1}	0	10^{-15}	10^{-9}	10^{-8}
eig	6×10^{-2}	0	10^{-12}	10^{-16}	10^{-11}

Type 4

taille 100×100	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	5.1×10^{-1}	0	10^{-13}	10^{-8}	10^{-7}
v1	1.7×10^{-1}	10^{-15}	10^{-14}	10^{-13}	10^{-8}
v2	4×10^{-2}	10^{-16}	10^{-14}	10^{-13}	10^{-8}
v3	5×10^{-2}	10^{-16}	10^{-14}	10^{-9}	10^{-8}
puissance itérée	9×10^{-2}	10^{-16}	10^{-15}	10^{-9}	10^{-8}
puissance itérée améliorée	7×10^{-2}	0	10^{-15}	10^{-9}	10^{-8}
eig	1×10^{-2}	0	10^{-15}	10^{-16}	10^{-14}

taille 300×300	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	3.86	0	10^{-12}	10^{-10}	10^{-7}
v1	1.4	10^{-14}	10^{-13}	10^{-14}	10^{-7}
v2	1.2×10^{-1}	10^{-14}	10^{-13}	10^{-14}	10^{-7}
v3	1.4×10^{-1}	10^{-14}	10^{-13}	10^{-8}	10^{-7}
puissance itérée	6.38	0	10^{-14}	10^{-9}	10^{-7}
puissance itérée améliorée	3.57	0	10^{-14}	10^{-9}	10^{-7}
eig	3×10^{-2}	0	10^{-14}	10^{-16}	10^{-14}

taille 500×500	temps	meilleure vp	pire vp	meilleure cpe	pire cp
v0	2.53×10^{-1}	0	10^{-12}	10^{-11}	10^{-7}
v1	2.05	10^{-13}	10^{-13}	10^{-13}	10^{-7}
v2	2.4×10^{-1}	10^{-14}	10^{-13}	10^{-14}	10^{-8}
v3	2.1×10^{-1}	10^{-14}	10^{-13}	10^{-8}	10^{-7}
puissance itérée	1.875×10^1	0	10^{-14}	10^{-9}	10^{-8}
puissance itérée améliorée	9.2	0	10^{-14}	10^{-9}	10^{-8}
eig	5×10^{-2}	0	10^{-14}	10^{-16}	10^{-14}