Sorbonne Université

UE 4IN910 - « Algorithmique numérique »
Master Informatique - M1 SFPN

Rapport de TME
TME n°1 - Calcul matriciel

Calle Viera Andersson

Enseignant
GRAILLAT Stef

Table des matières

Introduction

Dans ce T.M.E nous allons nous intéresser aux calculs matriciels et plus particuliérement à la décomposition S.V.D de matrices dans le cadre de l'imagerie.
Pour finir nous nous intéresserons à l'algorithme PageRank de Google.
Le code est réalisé en MatLab et fournit avec le rapport.
La fonction **CompressSVD.m** rélise la compression d'une image et l'affiche, **TauxCompress.m** calcul le taux de compression et **Ploting.m** utilise ces deux fonctions pour afficher la courbe d'évolution de la compression.
La fonction **Defloute.m** réalise le défloutage d'une image et l'affiche.
PowerM.m est l'implémentation de l'algorithme de la puissance itérée.

1 Exercice 6 :

Question 1 :

À l'Exercice 1.3 nous avons montrer que pour une matrice A $\in \mathbb{R}^{m\times n}$ la matrice la plus proche de rang k < n pouvait s'exprimer comme $A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T$

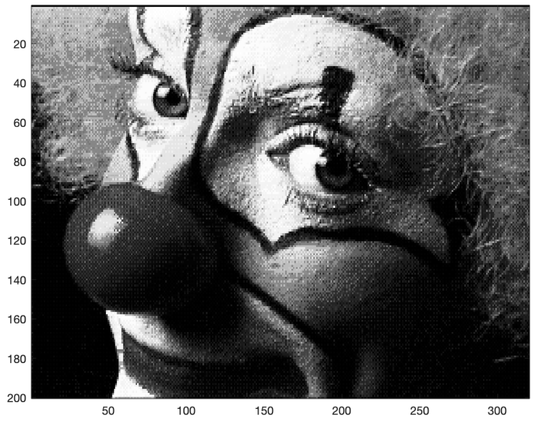
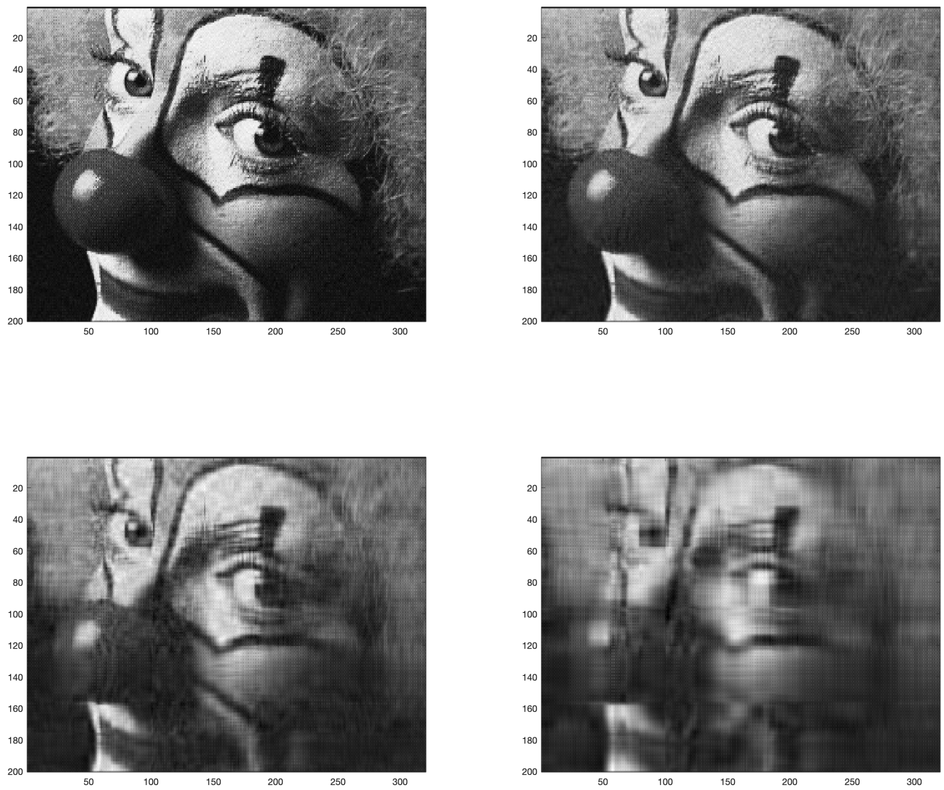


Image originale de taille 200 x 300

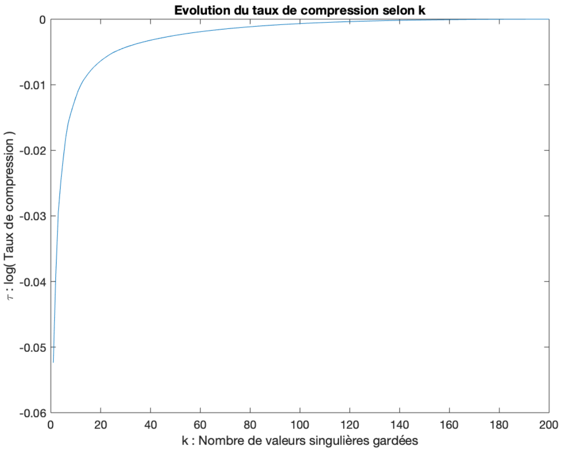


Différentes images compressées pour k $\in$ {100, 50, 20, 10} (de haut en bas et de gauche à droite)

Question 2 :

Avec une approche assez intuitive on peut définir le taux de compression comme $\tau = \frac{Volume final}{Volume Initial}$. On peut donc remarquer que plus ce taux est faible, plus la taille du fichier compressé sera faible.

Pour déterminer le volume d'une image I nous allons faire une décomposition S.V.D de I et utiliser ici la norme de Frobenius de sa matrice contenant ses valeurs singulières.
Notons $I = U_I \Sigma_I V_I^T$, l'image originale et $J = U_J \Sigma_J V_J^T$, on a donc que $\tau = \frac{\|\Sigma_J\|_F}{\|\Sigma_I\|_F}$



Taux de compression en échelle logarithmique

On remarque bien que le taux de compression augmente très rapidement jusqu'à 20 valeurs singulières, ensuite le gain de compression n'est plus significatif. Pour avoir un fichier le plus lèger possible sans toutefois perdre en qualité on devra donc choisir $k \approx 20$.

2 Exercice 7 :

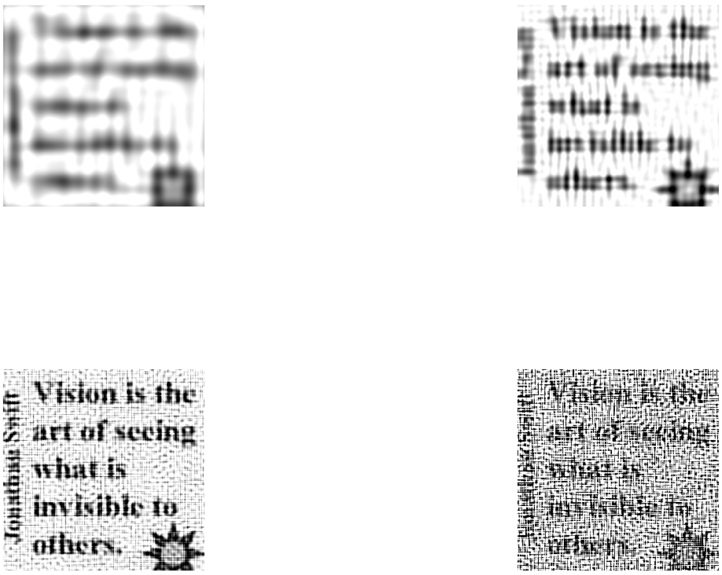
Soit G une image flouttée, F l'image originale, K le modèle de floutage et η le bruit.
On pose g = vec(G), f = vec(F) et on peut ainsi écrire que $= Kf + \eta$.
Ici on cherche à résoudre le problème : $\min\limits_f \|g - Kf\|_2^2$

Question 1 :

Écrivons la décomposition S.V.D de K tel que $K = U\Sigma V^T$, nous allons allons montrer que : $f^* = V\Sigma^{-1}U^T = \sum\limits_{i=1}^{n} \frac{u_i^T g}{\sigma_i} v_i$ est solution. Tout d'abord on peut remarquer que :
$\min\limits_f \|g - Kf\|_2^2 \iff \exists f,\ \|g - Kf\|_2^2 = 0 \iff \exists f,\ g - Kf = 0$ (par définition d'une norme).
Notons f^* un tel vecteur, on a donc que :
$g - Kf^* = 0 \implies Kf^* = g \implies K^{-1}Kf^* = K^{-1}g \implies f^* = K^{-1}g$.
Or à l'Exercice 1.1 nous avons montré que $K^{-1} = V\Sigma^{-1}V^T$, où $\Sigma^{-1} = diag(\sigma_1, \ldots, \sigma_n)$ donc :
$f^* = V\Sigma^{-1}V^T g = \sum\limits_{i=1}^{n} \frac{u_i^T g}{\sigma_i} v_i$ □

Question 2 :

Image floutée originale de taille 256 x 256



Différentes images déflouttées pour k ∈ {256, 1000, 5000, 8000} (de haut en bas et de gauche à droite)
L'optimum est atteint pour k = 5000 et on observe un dégradement de l'image pour $k \geq 5000$

3 Exercice 8 :

Soit $W = \{$ ensembles des pages atteignables depuis une page racine$\}$, notons $|W| = n$. On pose dans toute la suite $p = 0.85$ et $\delta = \frac{1-p}{n}$.
On définit la matrice de conncetion $G = (g_{ij})_{\substack{i\geq 1\\ j\geq 1}}$ de W comme :

$$g_{ij} = \begin{cases} 1 & \text{si il y a un lien de i vers j} \\ 0 & \text{sinon} \end{cases}$$

On note $c_j = \sum_{i=1}^{n} g_{ij}$ le degré sortant de la j-ième page. Et on définit alors la matrice $A \in \mathbb{R}^{n\times n}$ comme :

$$A_{ij} = \begin{cases} p\frac{g_{ij}}{c_j} + \delta & \text{si } c_j \neq 0, \\ \frac{1}{n} & \text{sinon} \end{cases}$$

Question 1 :

Montrons que A^T est stochastique $i.e$ $\forall j \geq 1 \sum_{i=1}^{n} A_{ij} = 1$.
Soit $j \geq 1$, on va raisonner par diffraction de cas :

- Supposons que $c_j = 0$ par définition :
  $\implies \sum_{i=1}^{n} A_{ij} = \sum_{i=1}^{n} \frac{1}{n} = n\frac{1}{n} = 1$
- Supposons que $c_j \neq 0$:
  $\implies \sum_{i=1}^{n} A_{ij} = \sum_{i=1}^{n} p\frac{g_{ij}}{c_j} + \delta = \frac{p}{c_j}\sum_{i=1}^{n} g_{ij} + \sum_{i=1}^{n}\delta = \frac{p}{\sum_{i=1}^{n} g_{ij}}\sum_{i=1}^{n} g_{ij} + n\delta = p + n\left(\frac{1-p}{n}\right) = 1$

Donc A^T est bien une matrice stochastique.
Comme montré à l'exercice 3.1 on en déduit que 1 est une valeur propre de A^T. Or A et A^T ont le même polynôme caractéristique donc les même valeurs propres et alors 1 est aussi valeur propre de A. Grâce à l'égalité de l'exercice 3.2 on peut aussi remarquer que $\rho(A) = \rho(A) = \max_{i\in[\![1,n]\!]} |\lambda_i| = 1$, où $(\lambda_1, \ldots, \lambda_n)$ sont les valeurs propres de A. □

Question 2 :

On peut remarquer que pour $m = 1$, $A^m = A$, or $\forall i, j \geq 1$ $A_{ij} > 0$.
En effet $n > 0$ donc $\frac{1}{n} > 0$, c_j est définit comme une somme de 0 ou de 1 donc $c_j >= 0$ de plus : $p = 0.85 > 0 \implies \frac{1-p}{n} = \delta > 0$.
Donc A est bien une matrice primitive.
D'après le Théorème de Perron-Frobenius A admet une valeur propre réelle strictement positive $r > 0$ telle que :

- Pour toute valeur propre s de A $|s| < r$
- $\exists x \in \mathbb{R}^n, \|x\| = 1$ et $\forall i \in [\![1,\ n]\!] x_i > 0$ te que $Ax = rx$

Ici, par défnfition de r, on a que $r = \rho(A) = 1$, donc on a que $Ax = x$. De plus on a que $\|x\| = 1$ donc en particulier on a que $\|x\|_1 = \sum_{i=1}^{n} |x_i| = 1$, or x est strictement positive donc $\sum_{i=1}^{n} |x_i| = \sum_{i=1}^{n} x_i = 1$. □

Question 3 :

Si nous exécutons[U,G] = surfer('http ://www.sorbonne-universite.fr/',15) et que nous effectuons notre méthode de la puissance sur le graphe G ainsi obtenu à un seuil $\epsilon = 10^{-18}$ nous obtenons que la plus grande valeur propre est $l^* = 6.7519$.