TP 4 : Applications de la décomposition en valeurs singulières

Exercice 1. Dans cet exercice, nous allons manipuler des images (au format .png). Pour ce faire, vous allez devoir installer le module **IPCV** à Scilab, en suivant ces instructions :

— Lancez Scilab, et dans la console Scilab tapez la commande `atomsInstall("IPCV")`
— Une fenêtre de téléchargement s'affiche. Cela peut prendre un peu de temps avant que cela se termine (environ 1 minute).
— Lorsque la fenêtre disparaît, cela veut dire que le téléchargement est fini.
— **Redémarrez** Scilab.
— Au démarrage, vous devriez voir quelque chose comme `Start IPCV 4.1.2 for Scilab 6.0` etc.
— Tapez la commande `imshow(rand(500,500))`. Si vous voyez s'afficher une bouillie de pixels en noir et blanc, vous avez réussi.

Vous êtes presque prêts à commencer l'exercice, qui nécessite de manipuler une certaine image. Pour cela :

— Rendez-vous sur le Moodle du cours, et téléchargez l'un des fichiers d'images qui vous sont proposés. Enregistrez-le dans votre dossier personnel (ou dans le dossier de votre choix).
— Dans le *Navigateur de Fichiers* de Scilab, placez-vous dans ce même dossier. Cette étape est nécessaire pour que Scilab "voie" le fichier.
— Importez l'image à l'aide de la commande

```
X = im2double(imread('nom-du-fichier.png'));
```

S'il y a une erreur, vérifiez que vous êtes bien dans le bon dossier à l'intérieur de Scilab.
— Affichez l'image dans Scilab avec la commande `imshow(X);`
— Si vous voyez l'image s'afficher, vous avez réussi, et pouvez commencer le TP.

1. Pour toute matrice $X \in \mathcal{M}_{m,n}(\mathbb{R})$, la commande `[U,S,V]=svd(X);` permet d'obtenir la décomposition en valeurs singulières de X, sous la forme
$$X = USV^*, \tag{1}$$
où $S \in \mathcal{M}_{m,n}(\mathbb{R})$ est une matrice diagonale contenant les valeurs singulières de X, $U \in \mathcal{M}_m(\mathbb{R})$ et $V \in \mathcal{M}_n(\mathbb{R})$ sont deux matrices orthogonales.
— Effectuer la décomposition en valeurs singulières de X, la matrice définie lors de l'importation de l'image.
— Vérifier que (1) a bien lieu.
— Soit `D=diag(S)` le vecteur des valeurs singulières de l'image X. Tracer ces valeurs avec `plot(D)`. Que constatez-vous ?
2. Écrire une fonction `[M]=filtre(M,s)` qui prend en entrée une matrice M quelconque et un seuil de tolérance $s \geq 0$, et qui met à zéro tous les coefficients de M tels que $M_{ij} < s$.

3. Filtrer les valeurs singulières de l'image plus petites que 1 (c'est-à-dire construire $X_r = US_rV^*$ où $S_r = \texttt{filtre}(S, 1)$). Visualiser l'image ainsi obtenue avec `imshow(X_r)`. Vous pouvez essayer de faire varier la valeur du seuil du filtre et comparer les résultats.
4. Construire une image bruitée

```
X_b = X + 0.1*rand(size(X,1),size(X,2),'normal');
```

   On va essayer de construire une image régularisée `X_r` qui s'approche plus de X.
   — Visualiser cette image bruitée avec `imshow`.
   — Mesurer l'écart $\|X_b - X\|$ induit par le bruit.
   — Calculer la décomposition SVD de X_b, et filtrer ses valeurs singulières plus petites que 5. Comment vous apparaît l'image X_r ainsi obtenue ? Mesurer $\|X_r - X\|$. Est-ce satisfaisant ?
5. Il est interdit de s'amuser à refaire cet exercice avec une image de votre choix.

Exercice 2. On étudie dans cet exercice un problème de régression linéaire. On observe n couples (x_i, y_i) que l'on suppose liés par une relation affine approchée

$$y_i = ax_i + b + \varepsilon_i \tag{2}$$

où a, b sont des constantes à déterminer et ε_i est une perturbation (supposée petite devant y_i).

On définit a, b comme les solutions du problème de minimisation

$$\min_{a,b} \sum_{i=1}^{n} |ax_i + b - y_i|^2 \,. \tag{3}$$

Le but de l'exercice est de créer une fonction qui prend entrée le vecteurs $x = (x_i)_{1\leq i\leq n}$ et $y = (y_i)_{1\leq i\leq n}$, et fournit la solution (a, b) du problème (3).

Les lignes ci-dessous créent deux vecteurs x et y liés par la relation (2) avec $a = 1$ et $b = 3$:

```
n = 1000;
x = 10*rand(n,1,'uniform');
e = rand(n,1,'normal');
y = 3*x+1+e;
```

On peut ensuite afficher le nuage de points correspondant grâce à l'une ou l'autre des commandes suivantes :

```
plot(x,y,'+');
plot2d(x,y,style=-1);
```

Si l'on veut remplacer une partie des observations par des données aberrantes (c'est-à-dire des couples (x_i, y_i) qui ne sont pas liés par la relation (2), provenant par exemple d'erreurs de mesure), on peut ajouter les lignes suivantes :

```
p = 1/3; // proportion de données aberrantes
y(n-floor(n*p):n) = 30*rand(floor(n*p)+1,1,'uniform');
```

La proportion de données aberrantes est contrôlée par le nombre p qui doit être choisi entre 0 et 1.

1. Expliquer ce que font précisément les différentes lignes de code ci-dessus.
2. En utilisant l'équation normale vue en cours, ainsi qu'une décomposition en valeurs singulières de la matrice A associée, écrire une fonction `[a,b]=moindrescarres(x,y)` qui prend en entrée deux vecteurs colonnes de même taille x et y, et fournit la solution du problème (3).
3. Utiliser la fonction `moindrescarres` sur les données crées en début d'exercice (sans les données aberrantes). Tracer sur un même graphique les points (x_i, y_i) et la droite de régression $v = at + b$ obtenue. On pourra utiliser les commandes :

```
t = linspace(min(x),max(x),200);
v = a*t + b;
plot(t,v,'g'); // trace la droite en vert
```

4. Reprendre la question précédente en utilisant cette fois des observations comportant des données aberrantes. Que constatez-vous ?