Sorbonne Université
3M235 - Calcul matriciel numérique
Travaux pratiques - feuille 2

On rappelle (cfr. cours du 15/02/2019 sur le problème de Procruste) qu'étant donné deux matrices rectangulaires de même taille $A, B \in \mathbb{R}^{m\times n}$, le problème de minimisation de l'écart quadratique moyen à transformation orthogonale près consiste à trouver une matrice orthogonale $\Omega \in \mathbb{R}^{m\times m}$ qui réalise le minimum

$$\min\{\|\Omega A - B\|_F \mid \Omega \in O(m)\}. \tag{1}$$

Une approche qui permet sa résolution est basée sur la décomposition en valeurs singulières : on procède en une SDV de la matrice carrée $BA^T \in \mathbb{R}^{m\times m}$:

$$BA^T = U\Sigma V^T,$$

et une solution au problème de minimisation ci-dessus est alors obtenue en choisissant

$$\Omega = UV^T.$$

Si au lieu de toutes les transformations orthogonales, on ne s'autorise que les matrices de rotation ($SO(n) = \{\Omega \in O(n) \mid det(\Omega) = 1\}$), alors un minimum est atteint en choisissant

$$\Omega = URV^T,$$

où $R \in \mathbb{R}^{m\times m}$ est la matrice diagonale dont tous les éléments diagonaux sont égaux à 1 sauf éventuellement le dernier dont la valeur est égale au signe de $det(UV^T)$ (ce qui revient à changer le signe du dernier vecteur singulier à droite dans la SVD, donc celui associé à la plus petite valeur singulière, et assure que $det(\Omega) = 1$).

Application à l'identification et la classification des protéines.

On commencera par télécharger à l'adresse :

`http://www.ljll.math.upmc.fr/smets/3M235_TP2_Immunoglobuline.pdb`

le fichier (au format dit PDB pour Protein Data Bank) qui décrit la molécule d'immunoglobuline. Cette molécule est un des anti-corps qui forment notre système immunitaire, il s'agit d'une chaîne complexe comprenant 3085 atomes de carbone, oxygène et azote, en plus d'autres d'atomes d'hydrogènes (non repris dans le fichier, leurs positions dans la chaîne pouvant se déduire de celles des autres composants, car l'hydrogène n'a qu'un électron de valence).

Ci-dessous un extrait de ce fichier :

```
ATOM   45  C   PRO L   7  -30.386  10.736  -3.027  1.00 22.35    7FAB 145
ATOM   46  O   PRO L   7  -30.111  11.750  -2.389  1.00 22.47    7FAB 146
ATOM   47  CB  PRO L   7  -29.877  11.300  -5.464  1.00 20.97    7FAB 147
ATOM   48  CG  PRO L   7  -30.004  12.804  -5.531  1.00 23.40    7FAB 148
ATOM   49  CD  PRO L   7  -31.490  12.993  -5.431  1.00 19.63    7FAB 149
ATOM   50  N   PRO L   8  -30.247   9.570  -2.426  1.00 22.59    7FAB 150
```

Les colonnes qui nous importeront sont la deuxième, qui décrit l'indice de l'atome en question, et les colonnes 7, 8 et 9 qui décrivent la position de l'atome dans un référentiel donné. Ainsi, l'atome numéro 45 (du carbone C) se trouve en $x = -30.386$, $y = 10.736$ et $z = -3.027$ (le repère et les unités sont données plus haut dans le fichier mais nous n'en aurons pas utilité ici).

Etape 1. A l'aide de Python, lire ce fichier ligne à ligne, ne retenir que les lignes commençant par la chaîne ATOM, et former au passage une matrice numpy Immuno Ide taille 3×3085 dont chaque colonne contiendra les coordonnées (x, y, z) d'un des atomes de la molécule d'immunoglobuline.

On suppose maintenant recevoir pour analyse une chaîne de 12 atomes, avec suspicion que cette chaîne fasse partie d'une molécule d'immunoglobuline. On trouvera les données correspondant à cet extrait à l'adresse :

`http://www.ljll.math.upmc.fr/smets/3M235_TP2_Chaine_Mystere.txt`

Etape 2. Charger de même ces données dans une matrice Myst de taille 3×12. (Notez le format de fichier passablement simplifié par rapport au précédent).

Le problème qui se pose à nous est alors de retrouver à l'intérieur des données de l'immunoglobuline, une sous-chaîne de longueur 12 qui corresponde au mieux aux données mystère, modulo une possible translation et rotation ! (puisque bien sûr, rien ne nous assure que les données mystères soient décrites dans le même référentiel que les données de l'immunoglobuline).

Etape 3. Commencer par recentrer les données mystère, en retranchant à chaque colonne de Myst un même vecteur $u \in \mathbb{R}^3$ afin que le barycentre de toutes les colonnes de la matrice B ainsi obtenue soit égal au vecteur nul de $\mathbb{R}^3$.

Etape 4. Pour chaque k entre 0 et 3073, on considère la sous-matrice $C := \text{Immuno}[:, k : k+12]$ de taille 3×12 de Immuno correspondant à la sous-chaîne de longueur 12 commençant à l'atome numéro k. Comme à l'étape 3, on commence par recentrer C et on note A la matrice 3×12 ainsi obtenue. On résout ensuite le problème de procruste (variante $SO(3)$) pour ce couple de matrices (A, B) (lire la documentation de `numpy.linalg.svd`), et on calcule la valeur du minimum.

Etape 5. On détermine ensuite l'indice k (ou les indices ?) pour le(s)quel(s) ce minimum du problème de procruste est le plus petit, et si cette valeur est suffisamment proche de zéro (en comparaison à l'échelle typique des données) on pourra déduire avoir trouvé la position de la chaîne. On ira ensuite regarder dans le fichier .pdb original à cette position de quel acide aminé il s'agit (le nom est abrégé en colonne 4), et sauf erreur il devrait s'agir de la Tyrosine.

Etape 6. On tracera enfin à l'aide de matplotlib (on pourra par exemple utiliser `mplot3d`) la molécule d'immunoglobuline complète ainsi que la chaîne mystère (après avoir appliqué à cette dernière une translation et rotation adéquate, telles qu'obtenues lors de la résolution du problème de procruste) pour s'assurer visuellement de la concordance des données.
[Pour les amateurs/trices, il serait possible de modifier légèrement notre moteur de rendu 3D de la feuille de TP1 pour lui faire visualiser nos données ici, en lieu et place de `mplot3d`]