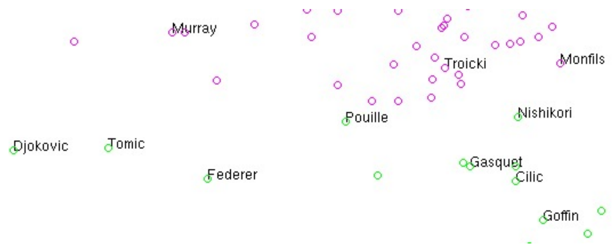
MAP 572 - Mise en oeuvre de méthodes numériques.

Raphaël Forien, Lucas Gerin, Ludovic Sacchelli, Nicole Spillane

---

Exploration de données : visualisation et clustering

Pour $n = 100$ joueurs de tennis on dispose de $p = 6$ statistiques : pourcentage de 1er services réussis, de balles de break sauvées, etc. Le but est d'analyser ces données de façon automatique :
- les représenter graphiquement de façon pertinente,
- déterminer des corrélations entre variables,
- classer les joueurs en différentes catégories.



---

1 Données

Question 1. Télécharger sur le `moodle` du cours le fichier `TennisChiffresTop100_2016.xls` et l'importer dans `matlab` avec la commande

```
[DonneesBrutes, NomsJoueurs, tab] = xlsread('TennisChiffresTop100.xls')
```

2 Visualisation : Analyse en Composantes Principales

2.1 Principe théorique

On note $X_{i,j}$ la j-ème statistique du joueur i. Les données sont donc formées de n points dans $\mathbb{R}^p$, on souhaite les représenter dans $\mathbb{R}^2$ de la façon la plus pertinente possible.

On définit la matrice des données centrées réduites $(\tilde{X}_{i,j})_{i\leq n, j\leq p}$ par :

$$\tilde{X}_{i,j} = \frac{X_{i,j} - \mathrm{mean}(X_{\bullet j})}{\mathrm{std}(X_{\bullet j})}$$

où $\mathrm{mean}(X_{\bullet j})$ est la moyenne du vecteur colonne $X_{\bullet j} = \begin{pmatrix} X_{1,j} \\ \dots \\ X_{n,j} \end{pmatrix}$, et $\mathrm{std}(X_{\bullet j})$ en est l'écart-type. La matrice $\frac{1}{n}\tilde{X}^T\tilde{X}$ est la matrice $p \times p$ des *corrélations*, elle est symétrique positive, donc diagonalisable de valeurs propres réelles positives $\lambda_1 \geq \cdots \geq \lambda_p \geq 0$. On peut par ailleurs la diagonaliser dans une base orthogonale $\mathbf{u}_1, \dots, \mathbf{u}_p$.

Question 2. Calculer la matrice de corrélation (vous pouvez utiliser `mean(X),std(X)`). Quelles variables sont les plus corrélées ?

Le point de vue de l'ACP est de chercher le vecteur $\mathbf{u} = (u_1, \dots, u_p) \in \mathbb{R}^p$ tel que la projection du nuage de points sur $\mathbf{u}$ ait une inertie maximale. Le résultat théorique est le suivant (voir par exemple le Théorème 1.1 dans [1]) :

Théorème 1 *Soit $X = (X_1, \dots, X_p)$ une variable aléatoire à valeurs dans $\mathbb{R}^p$, le vecteur propre $\mathbf{u}^1 = (u_1^1, \dots, u_p^1)$ associé à λ_1 est le vecteur qui maximise*

$$\mathbf{u} \mapsto \mathrm{Var}(< X, \mathbf{u} >).$$

De même, le vecteur propre $\mathbf{u}^2$ associé à λ_2 est celui qui maximise $\mathrm{Var}(< X, \mathbf{u}^2 >)$ parmi les vecteurs orthogonaux à $\mathbf{u}^1$.

2.2 Mise en oeuvre

Question 3. Compte tenu du résultat théorique, projeter les données suivant les deux composantes principales. Autrement dit :

– Calculer $\mathbf{u}^1, \mathbf{u}^2$,

– Tracer les n points de coordonnées $(a_i, b_i) := (< X_{i\bullet}, \mathbf{u}^1 >, < X_{i\bullet}, \mathbf{u}^2 >)$.

Vous aurez besoin des commandes suivantes : `eig(A)`,`plot`,... Pour faire apparaître le nom des joueurs, la commande `text(a,b,'texte')` écrit la chaîne de caractères `texte` à la position `(a,b)`.

2.3 Disque des corrélations

Pour un paramètre $i \leq p$, on pose

$$r_{i,1} = u_i^1 \sqrt{\lambda_1}, \qquad r_{i,2} = u_i^2 \sqrt{\lambda_2}.$$

Question 4. Tracer les p points de coordonnées $(r_{i,1}, r_{i,2})$ ainsi que le cercle unité.

Les points $A_i = (r_{i,1}, r_{i,2})$ se trouvent à l'intérieur du cercle unité. Le dessin obtenu s'interprète de la façon suivante : les variables significatives se trouvent au bord du cercle. Pour celles-ci,

$$\begin{array}{lll} \widehat{A_i O A_j} \approx 0° & \Rightarrow & \text{variables } i, j \text{ positivement corrélées} \\ \widehat{A_i O A_j} \approx 180° & \Rightarrow & \text{variables } i, j \text{ négativement corrélées} \\ \widehat{A_i O A_j} \approx 90° & \Rightarrow & \text{variables } i, j \text{ décorrélées} \end{array}$$

(pour une justification théorique, voir [1], Section 1.6).

3 Clustering : l'algorithme k-*means*

Au vu des résultats de l'ACP, il apparaît que les données des n joueurs ne sont pas homogènes. On souhaite partitionner l'ensemble des n points en k sous-ensembles (ou *cluster*), selon la proximité des projections (a_i, b_i).

Question 5. Vu le nuage de points obtenu après ACP, quel k choisir ?

Question 6. Implémenter l'algorithme k-means :

Paramètres :
k : nombre de cluster
$(a_1, b_1), \ldots, (a_n, b_n)$: points à partitionner
$(x_1, y_1), \ldots, (x_k, y_k)$: centres initiaux des cluster, choisis arbitrairement parmi les n points

Itérer jusqu'à stabilisation :
 Pour chaque $1 \leq i \leq n$
  Placer i dans le cluster dont le centre est le plus proche de (a_i, b_i) :

$$i \rightarrow \text{cluster } r \text{ tel que } ||(x_r, y_r) - (a_i, b_i)|| \text{ est minimal.}$$

  Mettre à jour le centre (x_r, y_r).

Question 7. Sur le graphique de l'ACP, représenter les points en différentes couleurs selon le cluster.

Question 8. **[Théorique]** Justifier que l'algorithme k-means termine. Trouver un exemple qui montre que la partition donnée par k-means peut dépendre des positions initiales $(x_1, y_1), \ldots, (x_k, y_k)$.

Références

[1] A.Tsybakov. *Apprentissage statistique.* Cours de l'École Polytechnique (2014).