

SY09 Printemps 2020

TP 04 — Analyse en composantes principales

L'objectif de cet exercice est de se familiariser avec la bibliothèque `scikit-learn` à travers le calcul de l'ACP.

Introduction

La bibliothèque `scikit-learn` permet facilement de calculer une ACP. Pour cela, il faut importer la classe suivante :

```
from sklearn.decomposition import PCA
```

Il faut ensuite instancier cette classe. Le seul paramètre qui nous intéresse est `n_components` qui fixe le nombre de composantes principales à retenir. On peut également fixer ce nombre de composantes principales en spécifiant une proportion d'inertie expliquée minimale sous la forme d'un nombre flottant entre 0 et 1.

Par exemple, pour calculer les deux premières composantes principales d'un tableau individu-variable `X`, on construit l'objet

```
cls = PCA(n_components=2)
```

Pour calculer la nouvelle représentation et renvoyer les composantes principales, il faut utiliser la méthode `fit_transform` en fournissant le jeu de données `X`. La méthode `fit_transform` apprend les deux premiers axes principaux et calcule la transformation correspondante du jeu de données `X`.

```
pcs = cls.fit_transform(X)
```

La variable `pcs` contient alors les `n_components` composantes principales. De plus, les attributs suivants sont disponibles depuis l'objet `cls`

- `components_` : les axes principaux u_i rangés par lignes.
- `explained_variance_` : la variance expliquée par chacun des axes principaux
- `explained_variance_ratio_` : le pourcentage de variance expliquée par chacun des axes principaux

Pour appliquer la représentation apprise avec `X` sur un autre tableau individu-variable `Y`, on utilise

```
pcs_Y = cls.transform(Y)
```

- ① Charger le jeu de données du TD précédent avec l'instruction

```
notes = pd.read_csv("data/notes.txt", sep="\s+")
```

Retrouver approximativement avec les axes principaux, la base B_3 du TD précédent ainsi que les variances expliquées correspondantes.

- ② Visualiser les individus dans le premier plan principal. On pourra utiliser la fonction `add_labels` du TP03 pour ajouter les étiquettes.

- ③ Deux nouveaux étudiants ont les notes suivantes :

Étudiant	math	scie	fran	lati	d-m
Alice	8.0	6.0	10.0	9	14
Steve	10	11	4.5	8.0	6

Placez-les dans le premier plan principal.

1 ACP sur les données « Crabs »

On utilisera les données **Crabs** présentes dans le fichier `data/crabs.csv`. Ce jeu de données est constitué de 200 crabs décrits par huit variables (trois variables qualitatives, et cinq quantitatives).

- ④ Charger le jeu de données et sélectionner les variables quantitatives en utilisant le code Python suivant :

```
crabs = pd.read_csv("data/crabs.csv", sep="\s+")
crabsquant = crabs.iloc[:, 3:8]
```

1.1 Analyse

- ⑤ Effectuer dans un premier temps une analyse descriptive des données. On s'interrogera notamment sur les différences de caractéristiques morphologiques, en particulier selon l'espèce ou le sexe : semble-t-il possible d'identifier l'une ou l'autre à partir d'une ou plusieurs caractéristiques morphologiques ?
- ⑥ Dans un second temps, on étudiera la corrélation entre les différentes variables. Quelle en est vraisemblablement la cause ? Quel traitement est-il possible d'appliquer aux données pour s'affranchir de ce phénomène ?

1.2 ACP

Cette étude vise à utiliser l'ACP pour trouver une représentation des crabs qui permettent de distinguer visuellement différents groupes, liés à l'espèce et au sexe.

1. Tester tout d'abord l'ACP sur `crabsquant` sans traitement préalable. Que constatez-vous ? Comment pouvez-vous expliquer ce phénomène à la lumière de l'analyse exploratoire de ces données menée préalablement ?
2. Trouver une solution pour améliorer la qualité de votre représentation en termes de visualisation des différents groupes.

2 ACP « à la main »

On associera dans cet exercice les mêmes pondérations à tous les individus, et on munira \mathbb{R}^p de la métrique euclidienne.

- ⑦ Charger le jeu de données « notes » et définir les matrices M et D_p .
- ⑧ Centrer le jeu de donnée.

⑨ Calculer la matrice V telle que définie dans le cours et calculer les valeurs et vecteurs propres de la matrice VM en les ordonnant par ordre décroissant. On pourra utiliser la fonction `eigh` du module `linalg` défini par

```
import scipy.linalg as linalg
```

⑩ En déduire les axes factoriels de l'ACP du nuage de points défini par le jeu de données `notes`. Quels sont les pourcentages d'inertie expliquée par chacun de ces axes ?

⑪ Calculer les composantes principales à l'aide des vecteurs propres calculés précédemment.

⑫ Retrouver les composantes principales à l'aide de la matrice W .

⑬ Soit $A \in \mathbb{R}^{n \times m}$ une matrice. Montrer que AA^T et $A^T A$ ont les mêmes valeurs propres non nulles.