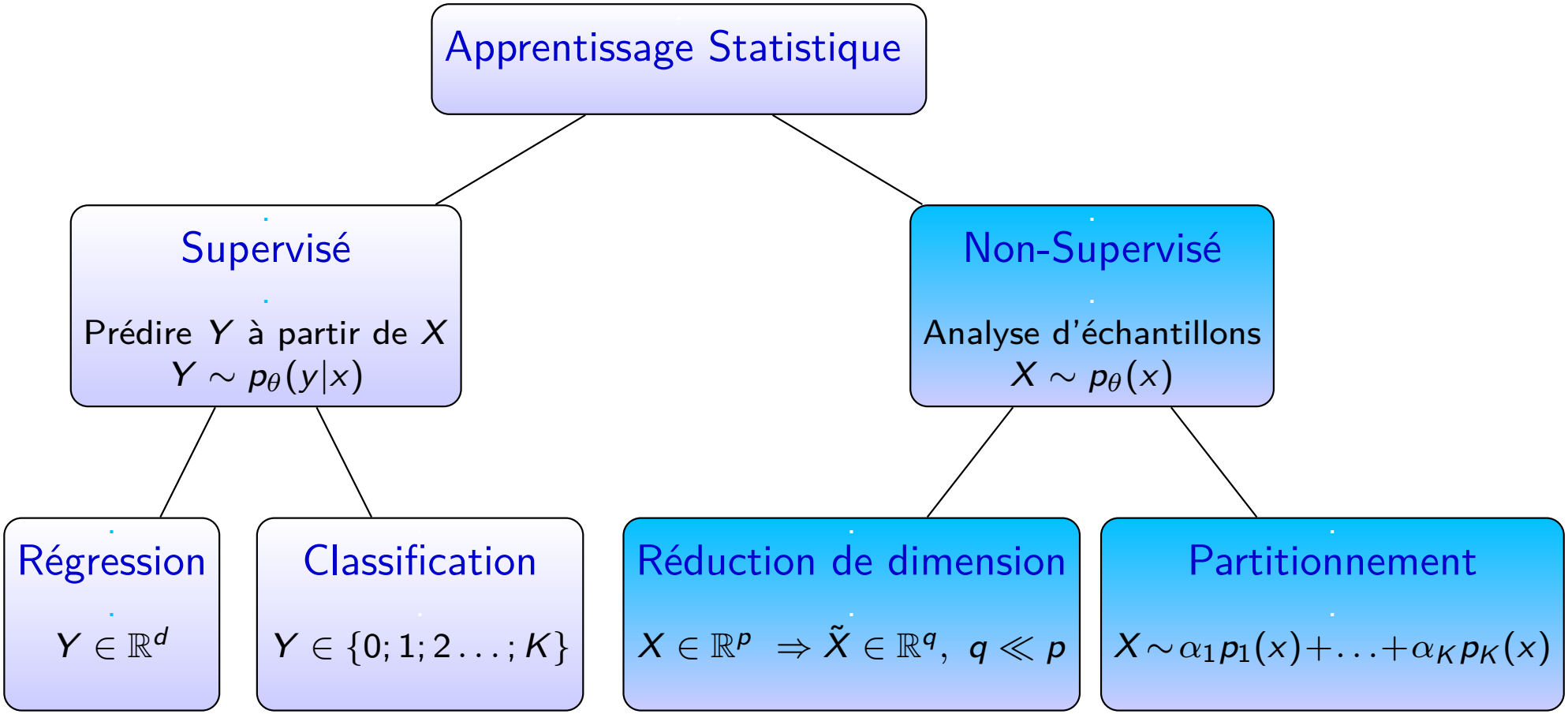
Statistique & Apprentissage

Paul-Henry Cournède

Amphi 9

Introduction à l'Apprentissage Statistique



V - Apprentissage Non-Supervisé

Soit X à valeurs dans $\mathbb{R}^p$.

V.1 - Apprentissage de représentations - Réduction de Dimension

Objectif : Déterminer une transformation $\psi : \mathbb{R}^p \to \mathbb{R}^q$ telle que $\psi(X)$ soit une variable mieux adaptée à la résolution d'un problème (de régression, de classification, de partitionnement...) : $\psi(X)$ sera appelée une représentation de X.

Un critère important peut être celui de la dimension : $q \ll p$.

Cette transformation ψ sera déterminée (apprise) à partir de données $(x_1, \ldots, x_N)$.

V.1.a - Analyse en Composantes Principales

Définition : On appelle variété affine de dimension q dans $\mathbb{R}^p$ l'ensemble $\mathcal{A}_q$:

$$\mathcal{A}_q = \{y \in \mathbb{R}^p : y = \mu + A_q\lambda,\ \lambda \in \mathbb{R}^q\}, \quad \text{où :.}$$

- $\mu \in \mathbb{R}^p$ est un facteur de localisation.
- $A_q \in \mathcal{M}_{p,q}$ est une matrice de q vecteurs orthonormés ($A_q^T A_q = I_q$)

Problème d'approximation d'une variable aléatoire sur une variété affine :

Soit q fixé, soit X notre variable aléatoire, on cherche donc ψ telle que :

- $\psi(X)$ soit à valeurs dans une variété affine d'ordre q
- $\mathbb{E}(\|\psi(X) - X\|^2)$ soit minimale.

Problème d'approximation d'une variable aléatoire sur une variété affine : Soit q fixé, soit X notre variable aléatoire, on cherche donc ψ telle que :

- $\psi(X)$ soit à valeurs dans une variété affine d'ordre q
  $\Longrightarrow \psi(X) = \mu + A_q\lambda(X)$, avec $\lambda(X)$ à valeurs dans $\mathbb{R}^q$
- $\mathbb{E}(\|\psi(X) - X\|^2)$ soit minimale.

Pour $(x_1, \ldots, x_N)$, on prend l'espérance empirique de $\|\psi(X) - X\|^2 = \|\mu + A_q\lambda(X) - X\|^2$ et on cherche donc à minimiser :

$$\mathcal{E}(\mu, A_q, \{\lambda_i\}_{1\leq i\ N}) = \frac{1}{N}\sum_{i=1}^{N}\|x_i - \mu - A_q\lambda_i\|^2.$$

Pour A_q fixé, CNS pour que μ et $\{\lambda_i\}_{1\leq i\leq N}$ minimisent $\mathcal{E}$:

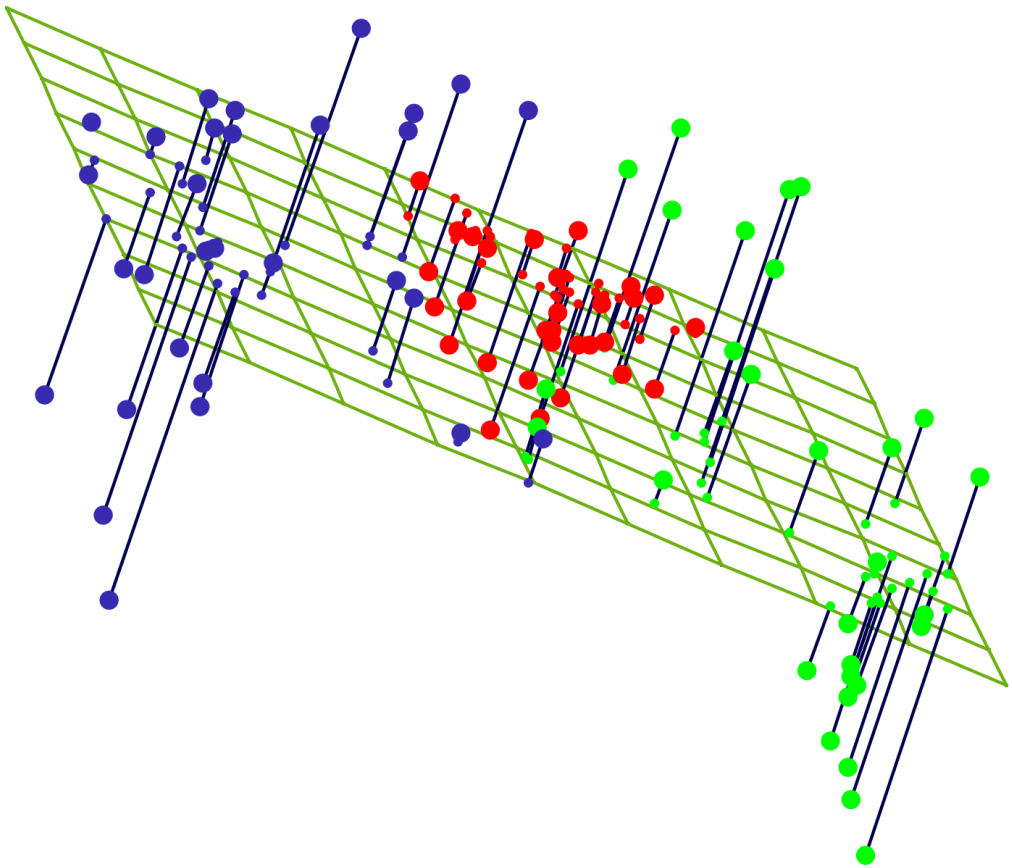
$$\begin{cases} \dfrac{\partial\mathcal{E}}{\partial\mu} = 2\displaystyle\sum_{i=1}^{N}(x_i - \mu - A_q\lambda_i) = 0 \\ \dfrac{\partial\mathcal{E}}{\partial\lambda_i} = -2\displaystyle\sum_{i=1}^{N} A_q^T\,(x_i - \mu - A_q\lambda_i) = 0, \quad \forall\ 1 \leq i \leq N. \end{cases}$$

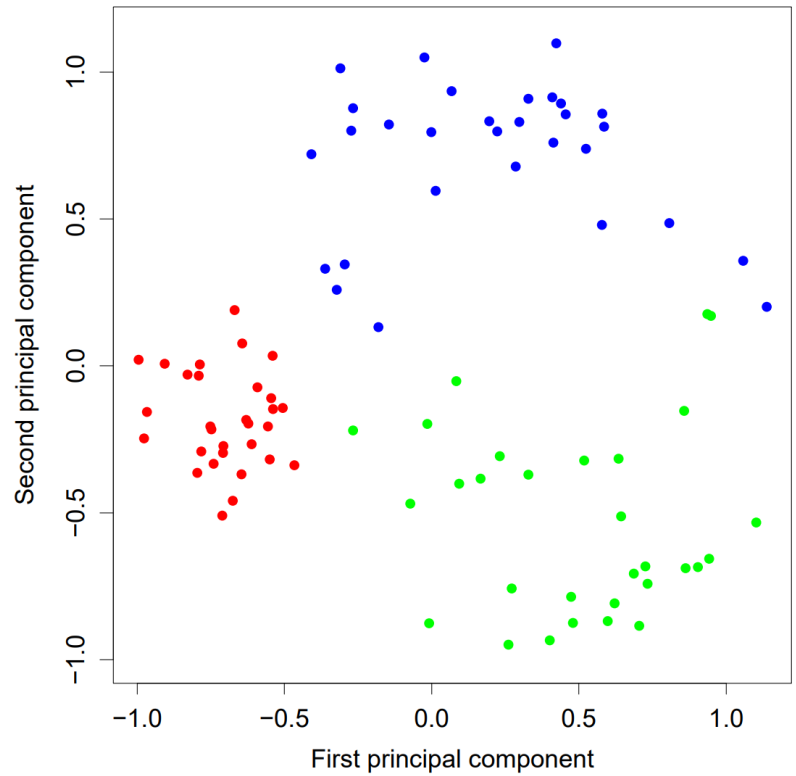
$\Longrightarrow$ réalisée pour $\mu = \overline{x}$ et $\lambda_i = A_q^T(x_i - \overline{x})$

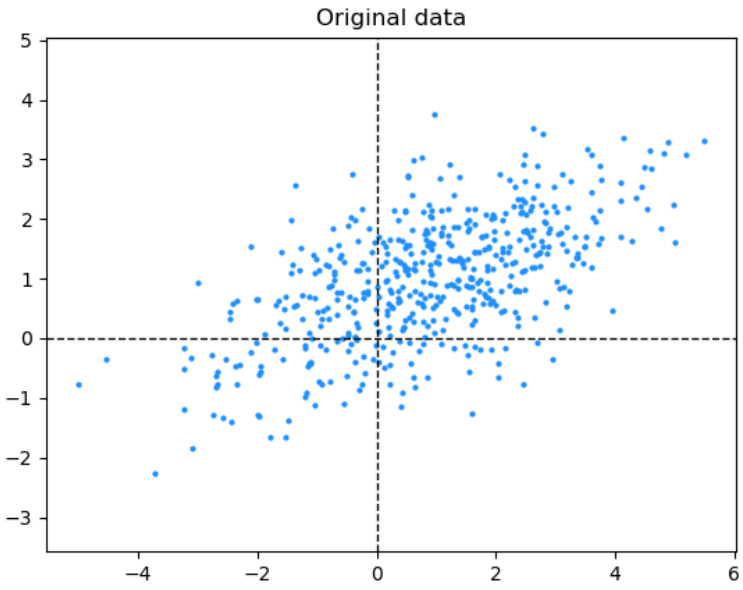
$\Longrightarrow$ minimisation sur l'ensemble des matrices orthogonales dans $\mathcal{M}_{p,q}$ de

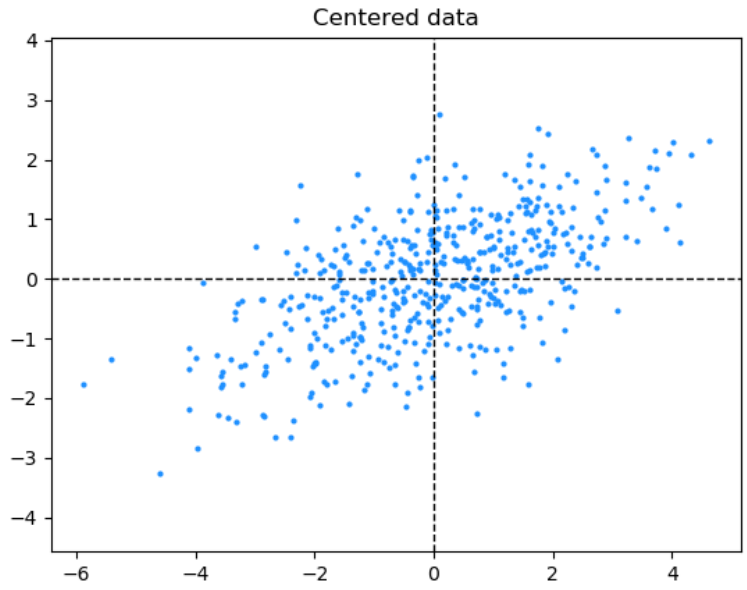
$$\frac{1}{N}\mathcal{E}\,(A_q) = \sum_{i=1}^{N}\|x_i - \overline{x} - A_qA_q^T(x_i - \overline{x})\|^2$$

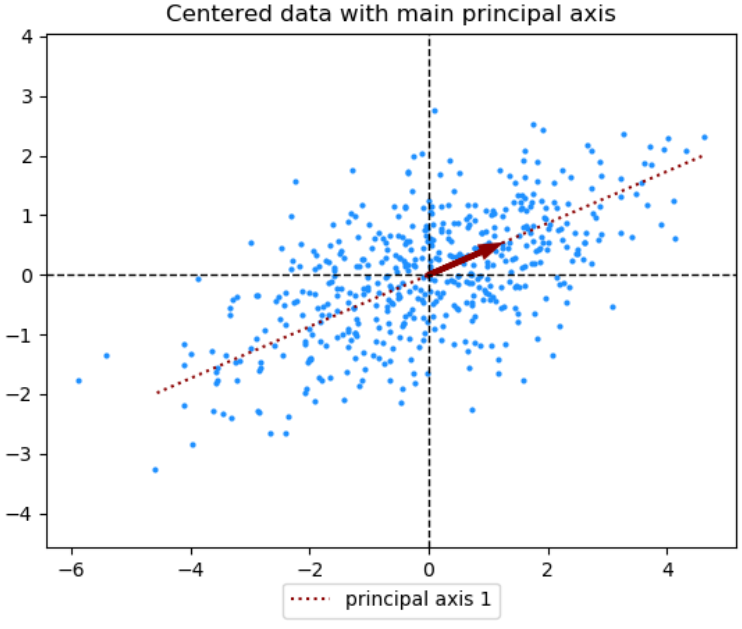
$\Longrightarrow$ Projection orthogonale sur le sous-espace engendré par les vecteurs colonnes de A_q

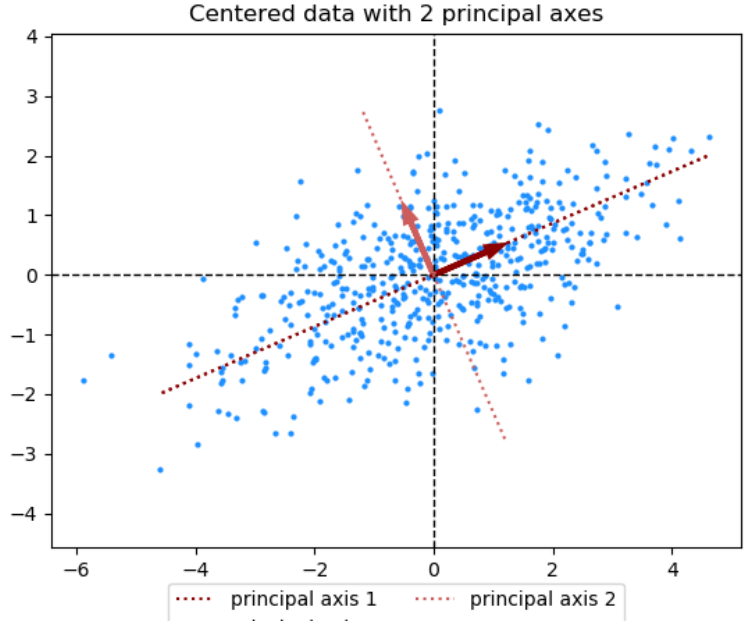



Second principal component
First principal component
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0


Original data


Centered data


Centered data with main principal axis
principal axis 1


Centered data with 2 principal axes
principal axis 1
principal axis 2

Théorème de décomposition en valeurs singulières :

Soit $X \in \mathcal{M}_{N,p}(\mathbb{R})$. Alors il existe :

- $U \in \mathcal{M}_{N,N}(\mathbb{R})$ orthogonale ($U^T U = I_p$),
- $D \in \mathcal{M}_{N,p}(\mathbb{R})$, matrice diagonale rectangle de rang r, $D = \text{diag}(d_1, \ldots, d_r, 0, \ldots, 0)$, avec $d_1 \geq d_2 \geq \cdots \geq d_r > 0$
- $V \in \mathcal{M}_{p,p}$ orthogonale

telles que : $X = U D V^T$.

- colonnes de $U \equiv$ vecteurs singuliers à gauches, colonnes de $V \equiv$ vecteurs singuliers à droite.
- éléments diagonaux non nuls de $D \equiv$ valeurs singulières de X, elles sont uniques.

Théorème d'approximation sur une variété affine par composantes principales :

Soit $\tilde{X} \in \mathcal{M}_{N,p}$, $\tilde{X} = \begin{pmatrix} x_{11} - \overline{x}_1 & \ldots & x_{1p} - \overline{x}_p \\ \vdots & & \vdots \\ x_{N1} - \overline{x}_1 & \ldots & x_{Np} - \overline{x}_p \end{pmatrix}$: matrice de design centrée, de rang r.

Soit une décomposition en valeurs singulières pour $\tilde{X}$: $\tilde{X} = U D V^T$.

Soit $q \leq r$ et soit V_q, la matrice des q premiers vecteurs singuliers à droite.

Alors V_q est solution du problème de minimisation de $\mathcal{E}(A_q) = \sum_{i=1}^{N} \|x_i - \overline{x} - A_q A_q^T (x_i - \overline{x})\|^2$ sur l'ensemble des matrices orthogonales de $\mathcal{M}_{p,q}$.

C'est à dire : $x_i \approx \overline{x} + V_q V_q^T (x_i - \overline{x})$.

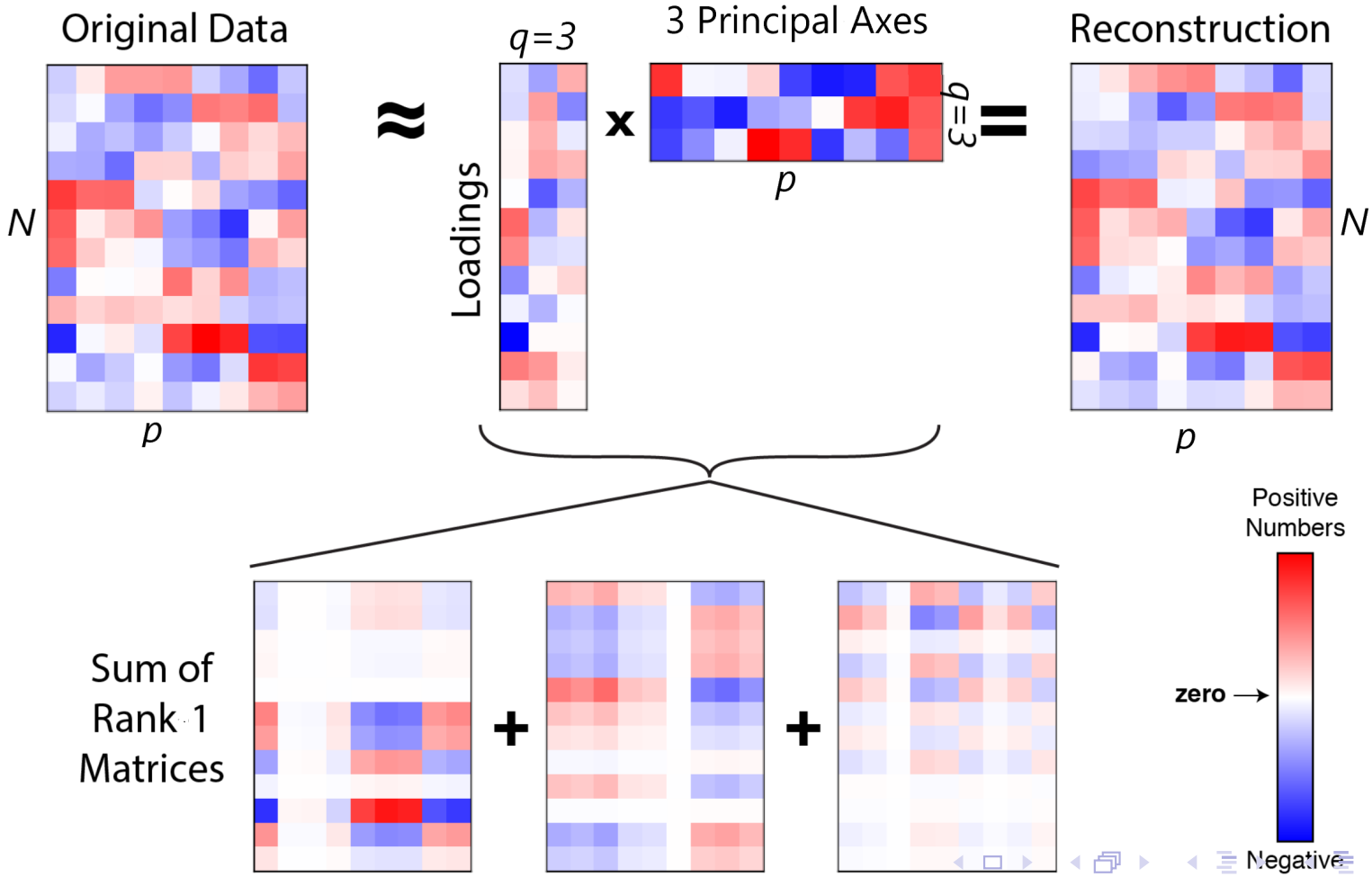
- Les vecteurs colonnes de V_q sont dits les directions principales ou axes principaux.
- $V_q^T (x_i - \overline{x}) = (U_{i1} d_1, U_{i2} d_2, \ldots U_{iq} d_q)^T$ donne les q coordonnées de $x_i - \overline{x}$ selon les axes principaux : appelées loadings.
- $d_k u_k \in \mathbb{R}^N$ est la k-ème composante principale, projection des $x_i - \overline{x}$ sur le k-ème axe pr.

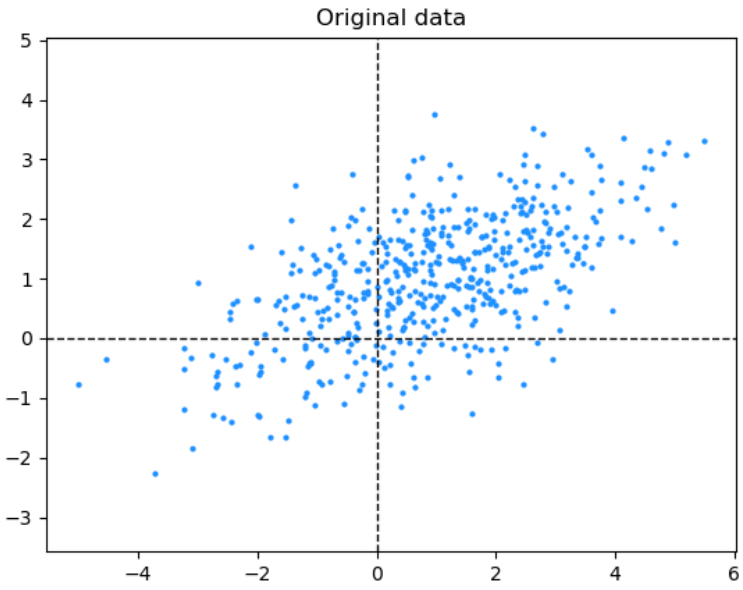
Bilan : Soient $(x_1, \ldots, x_N)$, $x_i \in \mathbb{R}^p$, et $\tilde{X} \in \mathcal{M}_{N,p}$ la matrice de design centrée associée.

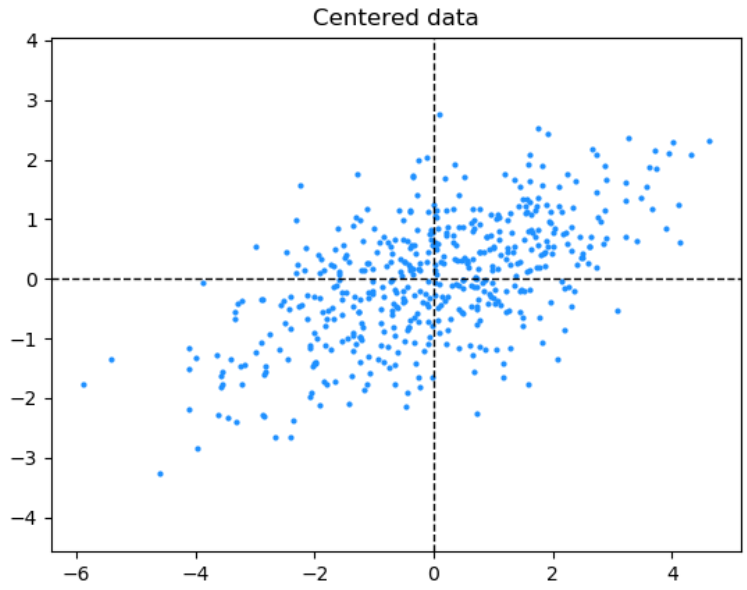
L'Analyse en Composantes Principales repose sur une SVD : $\tilde{X} = UDV^T$ qui donne :

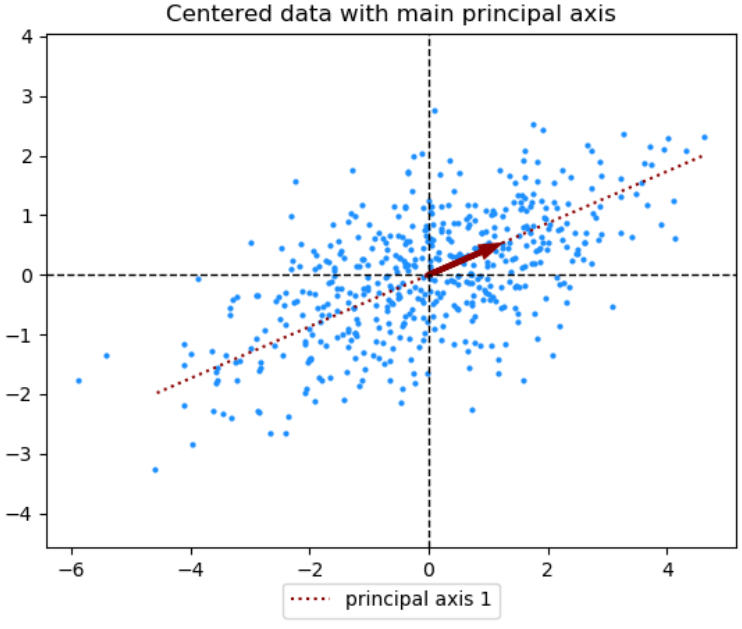
- les valeurs singulières ordonnées $d_1 \geq d_2 \geq \cdots \geq d_r$
- des axes principaux ordonnés $v_1, \ldots, v_r$ correspondant aux colonnes de V
- la matrice $UD = \tilde{X}V$, dont les colonnes sont les composantes principales et les lignes les loadings.

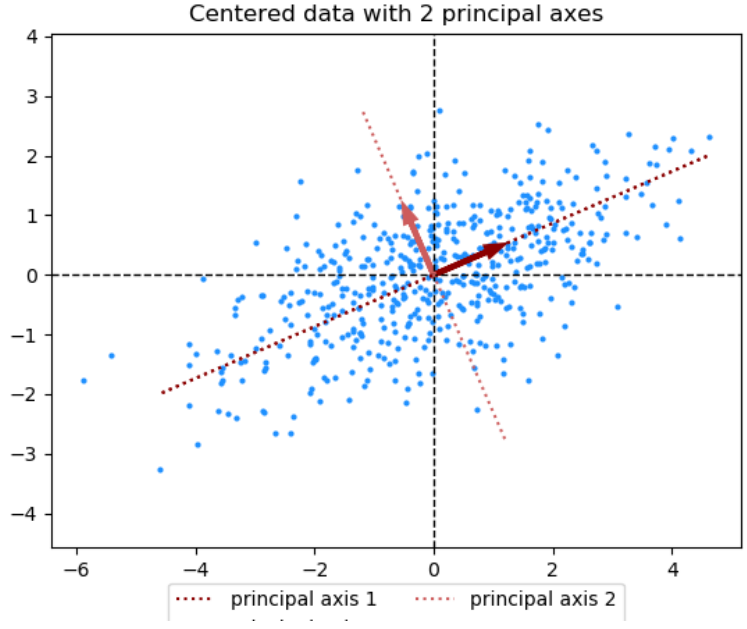
NB : Choix du nombre de composantes principales q a priori ou après analyse des résultats




Original data


Centered data


Centered data with main principal axis
principal axis 1


Centered data with 2 principal axes
principal axis 1
principal axis 2

Soit $(x_1, \ldots, x_N)$ un échantillon d'observations centrées, $\overline{x} = 0$, et soit $X \in \mathcal{M}_{N,p}$ la matrice de design associée de rang r, et la décomposition en valeurs singulières : $X = UDV^T$.

Définition : On appelle variance totale de l'échantillon :

$$VT(x_1, \ldots, x_N) = \frac{1}{N} \sum_{i}^{N} \|x_i\|^2 = \frac{1}{N} \operatorname{tr}(X^T X) = \frac{1}{N} \sum_{i=1}^{p} d_i^2$$

En effet : $\operatorname{tr}(X^T X) = \operatorname{tr}(V D^T U^T U D V^T) = \operatorname{tr}(V D^T D V^T) = \operatorname{tr}(D^T D) = \sum^{p} d_i^2$.

Proposition : Soit $q \leq r$, V_q les q premiers vecteurs colonnes de V.

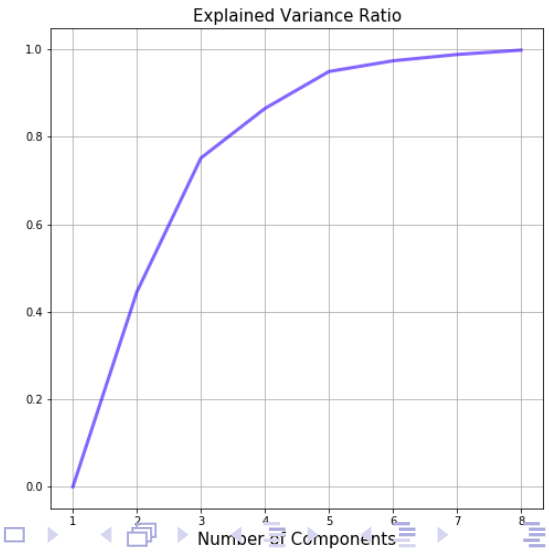
Soit $X_q = XV_q$: matrice de design $\in \mathcal{M}_{N,q}$ de l'échantillon projeté orthogonalement sur V_q

Alors X_q est l'échantillon projeté de variance totale maximale sur un sous-espace de dimension q.

La variance totale de l'échantillon associé à X_q est $VT_q = \frac{1}{N} \sum^{q} d_i^2$.

On appelle proportion de variance expliquée :

$$\frac{VT_q}{VT(x_1, \ldots, x_N)} = \frac{\sum_{i=1}^{q} d_i^2}{\sum_{i=1}^{p} d_i^2} .$$



V.1.b - Auto-encodeurs

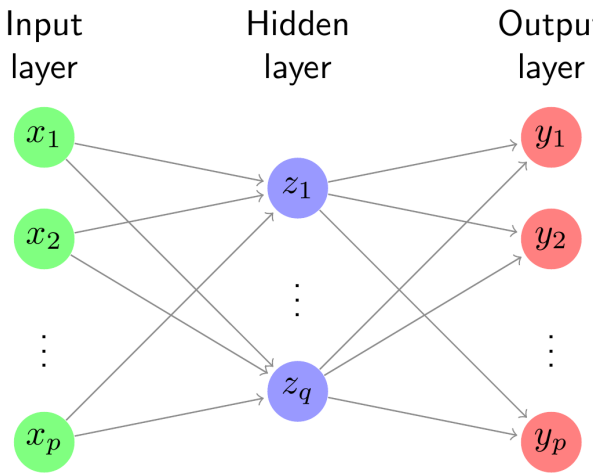
ACP = approximation sur une variété affine de dimension q : $X \approx \overline{x} + V_q V_q^T (X - \overline{x})$

$\Longrightarrow$ Perceptron à 3 couches $\psi : \mathbb{R}^p \to \mathbb{R}^p$, avec une couche cachée à q neurones :

$$\psi(x) = h_2 \circ g_2 \circ h_1 \circ g_1(x)$$

$$g_1 : \begin{array}{l} \mathbb{R}^p \to \mathbb{R}^q \\ x \mapsto V_q^T x - V_q^T \overline{x} \end{array} \qquad h_1 : \begin{array}{l} \mathbb{R}^q \to \mathbb{R}^q \\ z \mapsto z \end{array}$$

$$g_2 : \begin{array}{l} \mathbb{R}^q \to \mathbb{R}^p \\ z \mapsto V_q z + \overline{x} \end{array} \qquad h_2 : \begin{array}{l} \mathbb{R}^p \to \mathbb{R}^p \\ y \mapsto y \end{array}$$



- Le réseau ainsi défini minimise $J(\theta) = \sum_{i=1}^{N} \|x_i - \psi(x_i)\|^2$ parmi tous les réseaux de même type (p, q, p) et fonctions d'activation identité
- L'étape $x \mapsto g_1(x) = z$ est appelée encodage.
- L'étape $z \mapsto g_2(z) = y \approx x$ est appelée décodage.
- Si $\overline{x} = 0$: $g_1(x) = V_q^T x$ et $g_2(z) = V_q z$.

$\Longrightarrow$ Généralisation à des architectures plus complexes et des fonctions d'activation non-linéaires.

Un autoencodeur est un réseau de neurones $\psi : \mathbb{R}^p \to \mathbb{R}^p$ qui se décompose en deux sous réseaux : $\psi = \psi^{dec} \circ \psi^{enc}$, où $\psi^{enc} : \mathbb{R}^p \to \mathbb{R}^q$ est dit réseau d'encodage, et $\psi^{dec} : \mathbb{R}^q \to \mathbb{R}^p$ est dit réseau de décodage.

$q < p$ et pour $x \in \mathbb{R}^p$, $\psi^{enc}(x)$ est appelée représentation de x.

Le réseau est entraîné de façon non supervisée pour la fonction de perte $\|x - \psi(x)\|^2$.

V.2 - Clustering

Soit X variable aléatoire dans $\mathcal{X} \subset \mathbb{R}^p$, et soit $E = (x_1, \ldots, x_N)$ un échantillon i.i.d. pour X.

Objectif du clustering : regrouper les points les plus similaires dans des groupes appelés clusters : un cluster doit être le plus homogène possible, et des clusters différents les plus séparés possible

NB : La similarité dépend fortement de la métrique choisie.

Définitions :

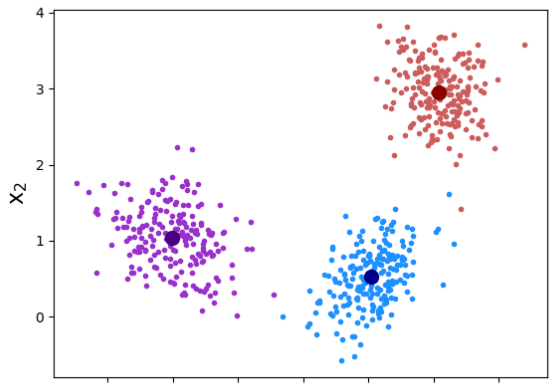
Un cluster $E^{(k)}$ est un sous ensemble de E. On identifiera le cluster avec l'ensemble des indices des points qu'il contient : $\pi^{(k)} = \{i : x_i \in E^{(k)}\}$.

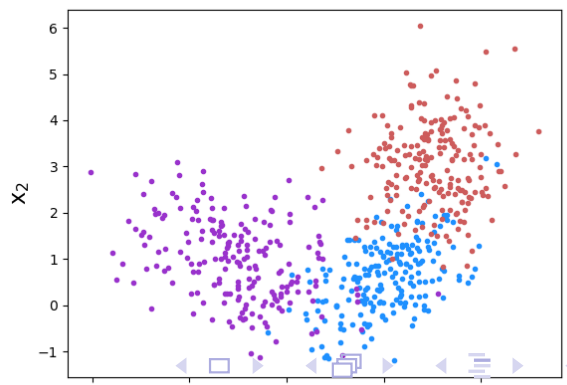
Le centroïde de $E^{(k)}$ est le barycentre du cluster : $\overline{x}^{(k)} = \dfrac{1}{|\pi^{(k)}|} \displaystyle\sum_{j \in \pi^{(k)}} x_j$,

Un partitionnement (clustering) de taille K est une partition de E en K clusters non vides : $\Pi = \{\pi^{(1)}, \cdots, \pi^{(K)}\}$

Remarque : Le clustering peut également être vu comme une réduction de dimension : $\forall i \in \pi^{(k)}$, $x_i \approx \overline{x}^{(k)}$, les N points sont résumés par les K centroïdes, avec $K \ll N$.

$\Longrightarrow$ À K fixé, maximiser la cohérence globale de la partition Π parmi les partitions de taille K.





V.2.a - Algorithme de clustering pour une taille K fixée

Définition : Nous appellons inertie d'un échantillon $(X_1, \ldots, X_N)$ la statistique

$$T(X_1, \ldots, X_N) = \sum_{i=1}^{N} \|X_i - \overline{X}\|^2$$

Propriété : Soit une réalisation $(x_1, \ldots, x_N)$, et soit un clustering associé à ce nuage de points : $\Pi = \{\pi^{(1)}, \cdots, \pi^{(K)}\}$. Soit l'inertie $T := T(x_1, \ldots, x_N)$, nous avons :

$$T = W(\Pi) + B(\Pi), \quad \text{avec :}$$

$$\text{Inertie intra-cluster :} \quad W(\Pi) = \sum_{k=1}^{K} \sum_{i \in \pi^{(k)}} \|x_i - \overline{x}^{(k)}\|^2$$

$$\text{Inertie inter-cluster :} \quad B(\Pi) = \sum_{k=1}^{K} |\pi^{(k)}| \, \|\overline{x}^{(k)} - \overline{x}\|^2.$$

En effet : $T = \sum_{k=1}^{K} \sum_{i \in \pi^{(k)}} \|x_i - \overline{x}\|^2 = \sum_{k=1}^{K} \sum_{i \in \pi^{(k)}} \left(\|x_i - \overline{x}^{(k)}\|^2 + \|\overline{x}^{(k)} - \overline{x}\|^2 \right)$, en utilisant la décomposition de König-Huygens.

Soit finalement : $T = \sum_{k=1}^{K} \sum_{i \in \pi^{(k)}} \|x_i - \overline{x}^{(k)}\|^2 + \sum_{k=1}^{K} |\pi^{(k)}| \, \|\overline{x}^{(k)} - \overline{x}\|^2$

Problème d'optimisation : On souhaite : $W(\Pi) \searrow$ et $B(\Pi) \nearrow$

Comme T est constant pour un échantillon donné, il suffit de minimiser $W(\Pi) = \sum_{k=1}^{K} \sum_{i \in \pi^{(k)}} \|x_i - \overline{x}^{(k)}\|^2$ sur l'ensemble des partitions possibles.

On cherche : $\Pi^* = \underset{\Pi \in \{\text{partitions de taille } k\}}{\arg\min} W(\Pi) = \sum_{k=1}^{K} \sum_{i \in \pi^{(k)}} \|x_i - \overline{x}^{(k)}\|^2$.

Une procédure heuristique d'optimisation : le K-means

Soit K, taille du clustering donnée.

Initialisation : Soit une première partition obtenue aléatoirement $\Pi_0 = \{\pi_0^{(1)}, \cdots, \pi_0^{(K)}\}$ et $\overline{x}_0^{(1)}, \ldots, \overline{x}_0^{(K)}$ les centroïdes correspondants.

```
Do
      t+=1
      For i = 1 : N
            trouver le centroïde x̄_{t-1}^{(p)} le plus proche de x_i et classer x_i dans le
            cluster p.
      end
      Π_t = la nouvelle partition obtenue.
      Calcul de x̄_t^{(1)}, ..., x̄_t^{(K)} les nouveaux centroïdes.
While |W(Π_t) − W(Π_{t−1})| > 0
```

Proposition : Soit $(\Pi_t)_{t>t_0}$ la séquence de partitions construites pendant l'algorithme K-means. Alors, $\exists T \geq 0$ tel que $\forall t < T : W(\Pi_t) < W(\Pi_{t-1})$ et $W(\Pi_T) = W(\Pi_{T-1})$.
Pendant les itérations du K-means, $W(\Pi_t)$ est strictement décroissante puis se stabilise.

⚠ Convergence vers un minimum local $\Longrightarrow$ répétition de l'algorithmes pour différentes initialisations aléatoires, puis choix de la meilleure configuration finale.

NB : Parfois, plus intéressant de réaliser le clustering sur $(\psi(x_i), \ldots, \psi(x_N))$ où $\psi(X)$ est une représentation mieux adaptée de la variable X.

V.2.b - Choix du nombre de clusters optimal

- Equivalent à la sélection de modèle $\Longrightarrow$ définition de critères adaptés

Définitions : Soit $\Pi = (\pi^{(1)}, \ldots, \pi^{(K)})$ un clustering de l'échantillon $(x_1 \ldots, x_N)$.

• L'homogénéité d'un cluster $\pi^{(k)}$ est définie par : $H^{(k)} = \dfrac{1}{|\pi^{(k)}|} \displaystyle\sum_{i \in \pi^{(k)}} \|x_i - \overline{x}^{(k)}\|$.

L'homogénéité du clustering sera alors : $H(\Pi) = \dfrac{1}{K} \displaystyle\sum_{k=1}^{K} H^{(k)}$.

• La séparabilité entre deux clusters $\pi^{(k)}$ et $\pi^{(l)}$ est simplement donnée par la distance entre les deux centroïdes $\|\overline{x}^{(k)} - \overline{x}^{(l)}\|$, et la séparabilité du clustering est la moyenne de la séparabilité de tous les clusters pris deux à deux :

$$S(\Pi) = \frac{2}{K(K-1)} \sum_{k=1}^{K} \sum_{l=k+1}^{K} \|\overline{x}^{(k)} - \overline{x}^{(l)}\| \ .$$

• L'indice de Davies-Bouldin du cluster $\pi^{(k)}$ est donné par : $DB^{(k)} = \max\limits_{l:\, l \neq k} \dfrac{H^{(k)} + H^{(l)}}{\|\overline{x}^{(k)} - \overline{x}^{(l)}\|}$.

Et l'indice de Davies-Bouldin du clustering Π est $DB(\Pi) = \dfrac{1}{K} \displaystyle\sum_{k=1}^{K} DB^{(k)}$.

NB : Plus les clusters sont homogènes et bien séparés, plus $DB(\Pi)$ sera petit.

Définition : Soit $\Pi = (\pi^{(1)}, \ldots, \pi^{(K)})$ un clustering. Soit $i \in \pi^{(k)}$.
Pour le point x_i, on considère la moyenne des distances aux autres points du même cluster :

$$a(x_i) = \frac{1}{|\pi^{(k)}|} \sum_{j \in \pi^{(k)}, j \neq i} \|x_j - x_i\|$$

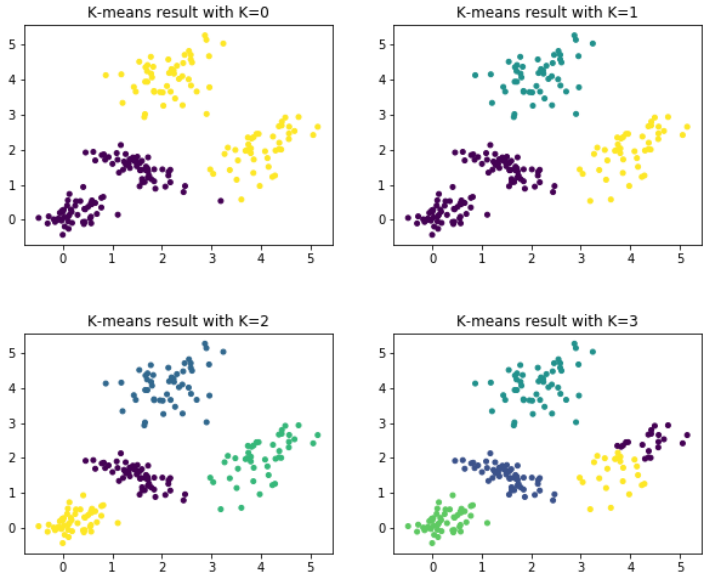
et la valeur minimale que pourrait prendre cette grandeur si x_i appartenait à un autre cluster :

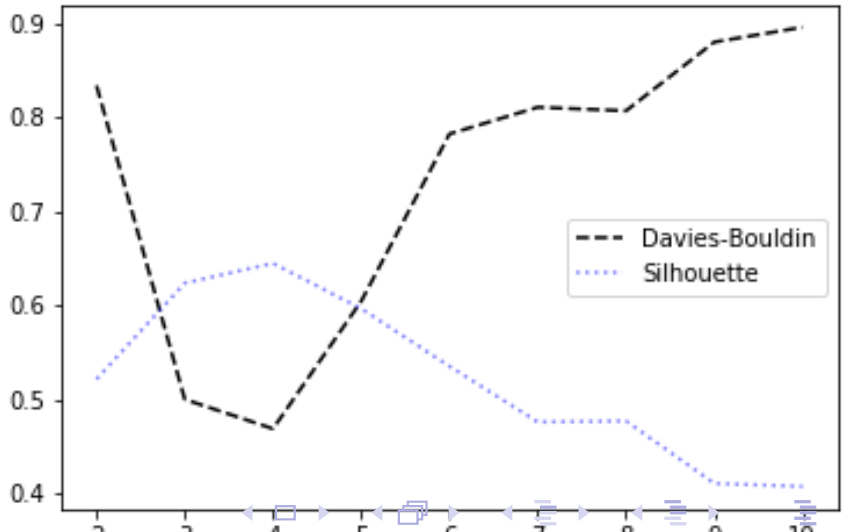
$$b(x_i) = \min_{l:\, l \neq k} \frac{1}{|\pi^{(l)}|} \sum_{j \in \pi^{(l)}} \|x_j - x_i\|$$

Le coefficient de silhouette du point x_i est alors donné par : $s(x_i) = \frac{b(x_i) - a(x_i)}{\max\left(a(x_i), b(x_i)\right)}$,

et le coefficient de silhouette du clustering par : $s(\Pi) = \frac{1}{N} \sum_{i=1}^{N} s(x_i)$.

$s(\Pi) \leq 1$ et plus $s(\Pi)$ sera proche de 1, meilleur le clustering sera considéré.





Le mot de la fin...

- Apprentissage statistique, Data Science, Intelligence artificielle... thématiques les plus porteuses dans la recherche et dans l'industrie actuellement

  $\Longrightarrow$ des besoins dans tous les domaines (biologie, marketing, sciences sociales, finance, médias, industrie...)

- Domaine facilement accessible : ressources web, logiciels, librairies disponibles facilement...
- Artisanat, ingénierie, et art...
- Ne pas considérer les algorithmes comme des recettes !
- Ne pas oublier de bien réfléchir au problème en amont : modélisation, prétraitement des données, feature engineering, ...
- Vos forces différenciantes :
  - votre capacité à comprendre pourquoi et comment les algorithmes peuvent donner des résulats ;
  - votre capacité à transposer ces algorithmes dans des applications très spécifiques, des nouveaux domaines.