Ingénieur en instrumentation

« Apprentissage non supervisé »

Anissa MOKRAOUI

Laboratoire de Traitement et Transport de l'Information (L2TI, UR 3043)
Bâtiment E, bureau 211
E-mail : anissa.mokraoui@univ-paris13.fr
Tel : 01 49 40 40 60

1

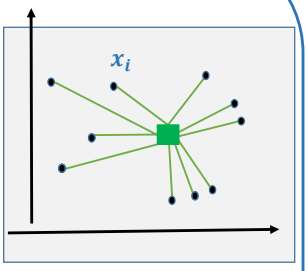
Apprentissage non supervisé

- **L'apprentissage non supervisé analyse la structure du jeu de données pour apprendre de lui-même à réaliser des tâches particulières par exemple :**
  - **Le clustering**
  - **La détection d'anomalie**
  - **La réduction de dimensionnalité**

TD-TP3 : Clustering, détection d'anomalie, réduction de dimensionnalité

2

Algorithme K-means clustering



- Le clustering permet de classer des documents, des photos, des tweets, de segmenter la clientèle d'une société.
- Le clustering apprend à classer des données selon leurs ressemblances.
- L'algorithme K-means cherche la position des centres qui minimise la distance entre les points x_i d'un cluster et le centre de ce dernier μ_j :

Fonction côut Inertia : $\sum_{i=0,\mu_j \in C_j}^{n} \min\left(\|x_i - \mu_j\|\right)^2$ **avec**

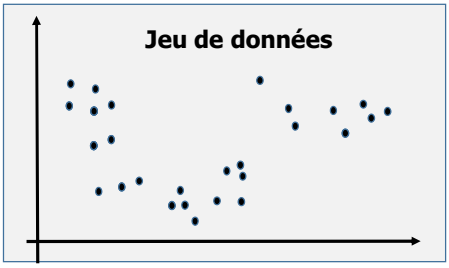
x_i : point d'un cluster

μ_j : centre du cluster j

n : nombre de points dans le cluster

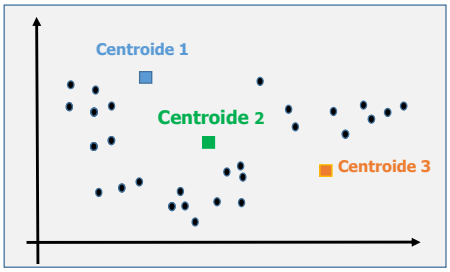
- ➢ Algorithme cherche à minimiser la variance des clusters
- ➢ Algorithme itératif

3

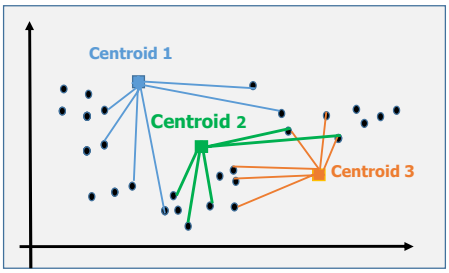


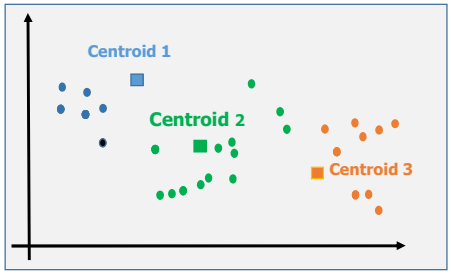
1. K centroides sont choisis aléatoirement parmi le jeu de données (barycentres des futures clusters) :

- ➢ Selon la position aléatoire de départ des centroides, le K-means peut donner de mauvais clusters.
- ➢ Pour pallier à ce problème, le K-means est exécuté avec différentes positions de départ, la solution retenue est celle qui minimise la somme des distances entre les points du cluster et son centre.

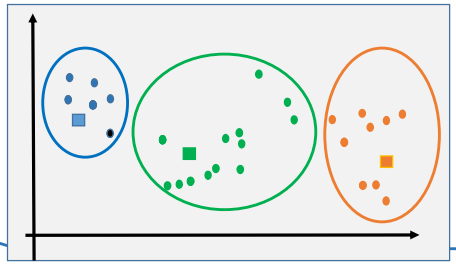


2. **Chaque point du jeu de données est affecté au cluster du centroid le plus proche :**



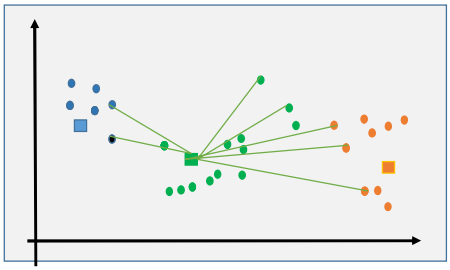


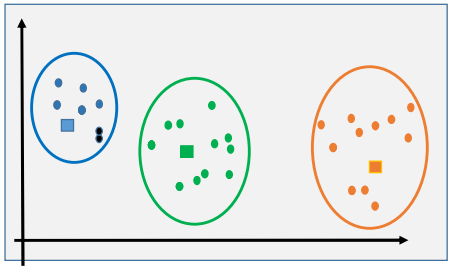
3. **Le centroide est déplacé au centre du cluster**



5

4. **Le processus est itéré jusqu'à ce que les centroids convergent vers une position d'équilibre :**





TD-TP 3 : Exercice 1

6

Détection d'anomalie

- La détection d'anomalie est recherchée dans plusieurs domaines d'applications :
  - Le développement des systèmes de sécurité ;
  - La détection de fraude bancaire ;
  - La défaillance technique dans une usine ;
  - La détection de pathologie sur des images multimodales ;
  - La détection d'anomalie dans une vidéo.
- L'algorithme de machine learning s'appuie sur la structure du jeu de données.
- L'algorithme analyse les données dont les caractéristiques sont très éloignées de celles des autres données.

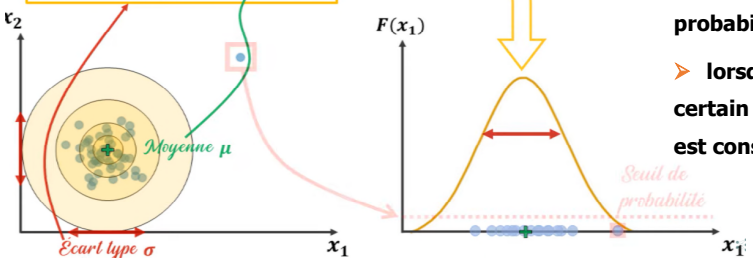
7

Détection d'anomalie : densité de probabilité

- On calcule la moyenne et l'écart type des données pour déterminer une fonction de densité de probabilité :

$$F(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Densité de probabilité (loi Normale)

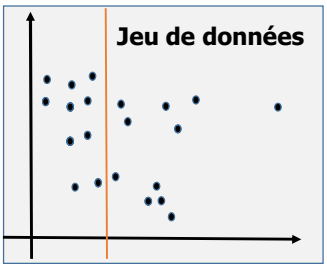


- On se sert de cette fonction pour calculer la probabilité d'existence d'un échantillon donné,
- lorsque la probabilité est inférieure à un certain seuil on considère que cet échantillon est considéré comme anormal

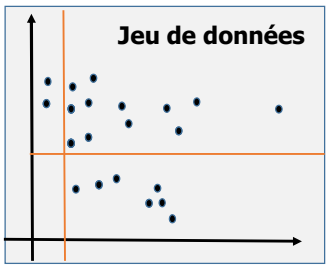
8

Détection d'anomalie : Isolation Forest algorithm

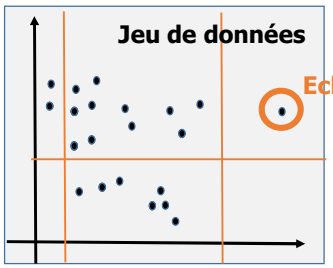
- Une série de découpes aléatoires



Nombre de découpe = 1
Aucun échantillon isolé



Nombre de découpe = 2
Aucun échantillon isolé

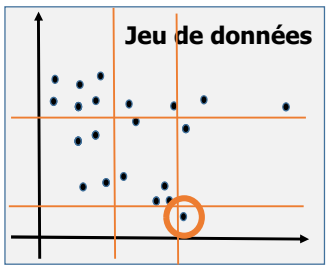


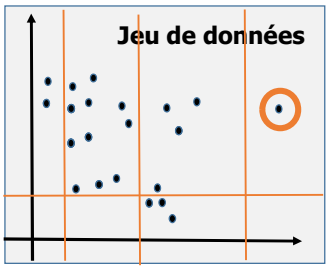
Nombre de découpe = 3
Echantillon isolé

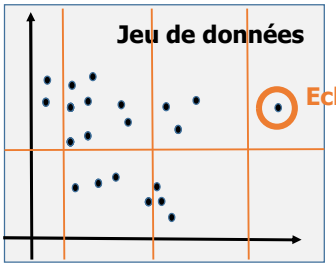
- On compte le nombre de découpe à réaliser pour isoler les échantillons
- Plus le nombre de découpe est faible, plus la probabilité d'anomalie est forte.

9

- Dans le cas où la probabilité est faible, il est possible d'isoler un échantillon perdu dans la masse de données avant d'isoler une véritable anomalie (très peu probable, mais possible)
- Solution basée sur une technique d'ensemble : Générerez plusieurs estimateurs







- Chaque estimateur génère des découpes aléatoires,
- Analysez l'ensemble des résultats,
- Disqualifiez les erreurs commises par certains estimateurs (parce que la majorité l'emporte)

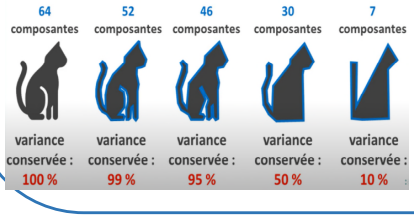
TD-TP 3 : Exercices 2 et 3

10

Réduction de la dimensionnalité

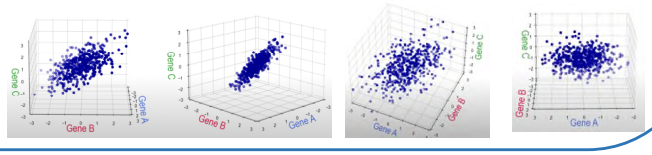
- **Réduction de la complexité d'un jeu de données en projetant ses données dans un espace de plus petite dimension (moins de variables) dans le but de :**
  - **accélérer l'apprentissage**
  - **lutter contre le fléau de la dimension (risque d'overfitting lié au superlus de dimension)**
  - **visualiser des données multi-dimensionnelles en 2 ou 3 dimensions**
  - **décorréler les variables en générant de nouvelles variables non corrélées**
  - **débruiter les données**

- **Exemples :**



Représentation des données de face et de profil

Représentation des données sous quatre angles de vues



Analyse en Composantes Principales (ACP)

- **Quel type de données ?**
  - **Des individus en lignes**
  - **Des variables quantitatives en colonnes**

Exemples : **Analyse sensorielle : note du descripteur k pour le produit i)**
Ecologie : concentration du polluant *k* dans la rivière *i*
Economie : valeur de l'indicateur *k* pour l'année *i*
Biologie : mesure *k* pour l'animal *i*
Marketing : valeur d'indice de satisfaction *k* pour la marque *i*
Génétique : expression gène *k* pour le patient *i*

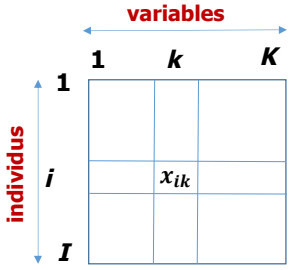


Tableau de données en ACP

Données de température :

- **15 individus : villes de France**
- **14 variables :**
  - **12 températures mensuelles moyennes (sur 30 ans) ;**
  - **2 variables géographiques (latitude et longitudes)**

	Janv	Févr	Mars	Avri	Mai	Juin	jul	Août	Sept	Octo	Nove	Déce	Lati	Long
Bordeaux	5.6	6.6	10.3	12.8	15.8	19.3	20.9	21	18.6	13.8	9.1	6.2	44.5	-0.34
Brest	6.1	5.8	7.8	9.2	11.6	14.4	15.6	16	14.7	12	9	7	48.24	-4.29
Clermont	2.6	3.7	7.5	10.3	13.8	17.3	19.4	19.1	16.2	11.2	6.6	3.6	45.47	3.05
Grenoble	1.5	3.2	7.7	10.6	14.5	17.8	20.1	19.5	16.7	11.4	6.5	2.3	45.1	5.43
Lille	2.4	2.9	6	8.9	12.4	15.3	17.1	17.1	14.7	10.4	6.1	3.5	50.38	3.04
Lyon	2.1	3.3	7.7	10.9	14.9	18.5	20.7	20.1	16.9	11.4	6.7	3.1	45.46	4.51
Marseille	5.5	6.6	10	13	16.8	20.8	23.3	22.8	19.9	15	10.2	6.9	43.18	5.24
Montpellier	5.6	6.7	9.9	12.8	16.2	20.1	22.7	22.3	19.3	14.6	10	6.5	43.36	3.53
Nantes	5	5.3	8.4	10.8	13.9	17.2	18.8	18.6	16.4	12.2	8.2	5.5	47.13	-1.33
Nice	7.5	8.5	10.8	13.3	16.7	20.1	22.7	22.5	20.3	16	11.5	8.2	43.42	7.15
Paris	3.4	4.1	7.6	10.7	14.3	17.5	19.1	18.7	16	11.4	7.1	4.3	48.52	2.2
Rennes	4.8	5.3	7.9	10.1	13.1	16.2	17.9	17.8	15.7	11.6	7.8	5.4	48.05	-1.41
Strasbourg	0.4	1.5	5.6	9.8	14	17.2	19	18.3	15.1	9.5	4.9	1.3	48.35	7.45
Toulouse	4.7	5.6	9.2	11.6	14.9	18.7	20.9	20.9	18.3	13.3	8.6	5.5	43.36	1.26
Vichy	2.4	3.4	7.1	9.9	13.6	17.1	19.3	18.8	16	11	6.6	3.4	46.08	3.26

Centrage et réduction des données

- **Centrer les données ne modifie pas la forme du nuage :**
  - **Moyenne de la variable *k* :** $\bar{x}_k = \frac{1}{I}\sum_{i=1}^{I} x_{ik}$
- **Normaliser les données est indispensable :**

$$x_{ik} \longrightarrow \frac{x_{ik} - \bar{x}_k}{s_k}$$

avec s_k l'écart type de la variable *k* :

$$s_k = \sqrt{\frac{1}{l}\sum_{i=1}^{l}(x_{ik} - \bar{x}_k)^2}$$

dans le cas où les unités de mesure sont différentes d'une variable à l'autre.

Données de température (centrées + normalisées)

	Janv	Févr	Mars	Avri	Mai	Juin	juil	Août	Sept	Octo	Nove	Déce
Bordeaux	0.84	0.98	1.40	1.33	0.94	0.85	0.52	0.74	0.90	0.84	0.67	0.72
Brest	1.10	0.54	-0.29	-1.30	-1.95	-1.98	-2.06	-1.83	-1.28	-0.18	0.62	1.14
Clermont	-0.71	-0.63	-0.50	-0.50	-0.44	-0.31	-0.21	-0.24	-0.44	-0.63	-0.76	-0.66
Grenoble	-1.28	-0.90	-0.36	-0.28	0.05	-0.02	0.13	-0.03	-0.16	-0.52	-0.82	-1.35
Lille	-0.81	-1.07	-1.51	-1.52	-1.40	-1.46	-1.33	-1.27	-1.28	-1.09	-1.05	-0.71
Lyon	-0.97	-0.85	-0.36	-0.06	0.32	0.38	0.42	0.27	-0.05	-0.52	-0.70	-0.92
Marseille	0.79	0.98	1.20	1.48	1.63	1.71	1.69	1.66	1.63	1.52	1.30	1.09
Montpellier	0.84	1.03	1.13	1.33	1.22	1.31	1.39	1.41	1.30	1.29	1.19	0.87
Nantes	0.53	0.26	0.11	-0.13	-0.37	-0.37	-0.50	-0.50	-0.33	-0.07	0.16	0.35
Nice	1.82	2.03	1.74	1.70	1.56	1.31	1.39	1.51	1.86	2.08	2.05	1.77
Paris	-0.30	-0.41	-0.43	-0.20	-0.09	-0.19	-0.36	-0.45	-0.55	-0.52	-0.47	-0.29
Rennes	0.43	0.26	-0.23	-0.64	-0.92	-0.94	-0.94	-0.91	-0.72	-0.41	-0.07	0.29
Strasbourg	-1.84	-1.85	-1.78	-0.86	-0.30	-0.37	-0.41	-0.65	-1.06	-1.60	-1.74	-1.87
Toulouse	0.37	0.42	0.65	0.45	0.32	0.50	0.52	0.69	0.74	0.55	0.39	0.35
Vichy	-0.81	-0.79	-0.77	-0.79	-0.57	-0.42	-0.26	-0.39	-0.55	-0.75	-0.76	-0.76

13

Problèmes et objectifs

Le tableau peut-être vu comme un ensemble de lignes ou un ensemble de colonnes :

- **Etude des individus :**
  - **Quand 2 individus se ressemblent du point de vue de l'ensemble des variables ?**
  - **S'il y a beaucoup d'individus, peut-on faire un bilan des ressemblances ?**

    Construction de groupes d'individus, partition des individus
- **Etude des variables :**
  - **Recherche des ressemblances (liaisons linéaires : coefficient de corrélation) entre les variables**

    Visualisation de la matrice des corrélations

    Recherche d'indicateurs

Objectifs de l'ACP :

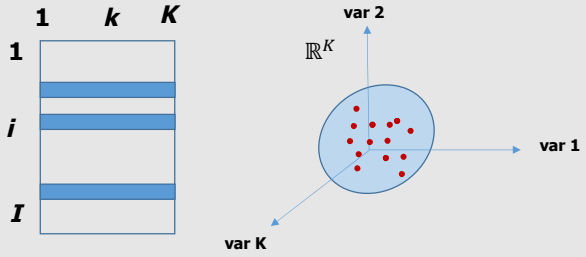
- **Descriptif statistique exploratoire : Visualisation de données par graphiques simples**
- **Synthèse -- résumé de grands tableaux individus par variables**

14

Nuages de points

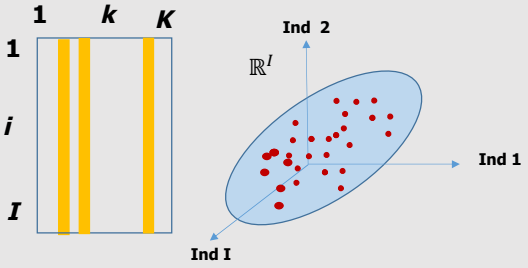
Etude des individus

TD-TP 3 : Exercice 4A



- On étudie les individus
- Chaque individu est représenté par un point
- On considère un nuage de point qui évolue dans espace à K dimension

Etude des variables



- On étudie les variables
- Chaque variable est représentée par un point
- On considère un nuage de point qui évolue dans espace à I dimension

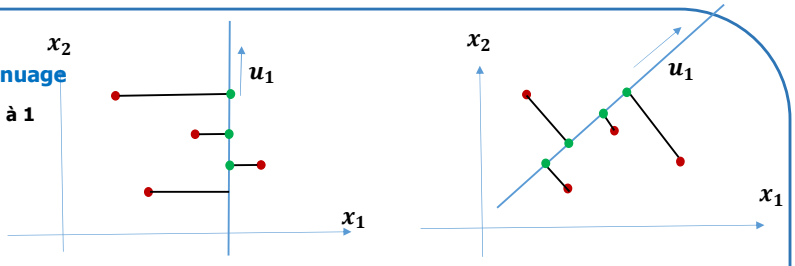
15

■ **Exemples : Ajustement des points du nuage**

Soit $K = 2$, on cherche à réduire la dimension à 1 ($M = 1$)

- : points du nuage
- : points projetés

N : nombre de points



■ **L'ACP cherche la droite dont la direction est donnée par le vecteur u_1 (la variance des points projetés (en $M = 1$) est la plus grande possible, la direction de variance principale) :**

$$s_k = \frac{1}{N}\sum_{i=1}^{N}\left(u_1^T x_i - u_1^T \bar{x}\right)^2 = \frac{1}{N}\sum_{i=1}^{N}\left(u_1^T (x_i - \bar{x})\right)^2 = \frac{1}{N}\sum_{i=1}^{N} u_1^T (x_i - \bar{x})(x_i - \bar{x})^T u_1$$

$$= u_1^T \left(\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})(x_i - \bar{x})^T\right) u_1 = u_1^T S u_1$$ avec S la matrice de covariance empirique

- On contraint u_1 et on maximise : $u_1^T S u_1 + \lambda_1 \left(1 - u_1^T u_1\right)$
- La solution doit satisfaire : $S u_1 = \lambda_1 u_1$ C'est la définition des vecteurs propres de la matrice S

16

- Si u_1 est un vecteur propre, la variance est égale à : $u_1^T S u_1 = u_1^T \lambda_1 u_1 = \lambda_1$

  Avec λ_1 (scalaire) est la valeur propre du vecteur propre u_1

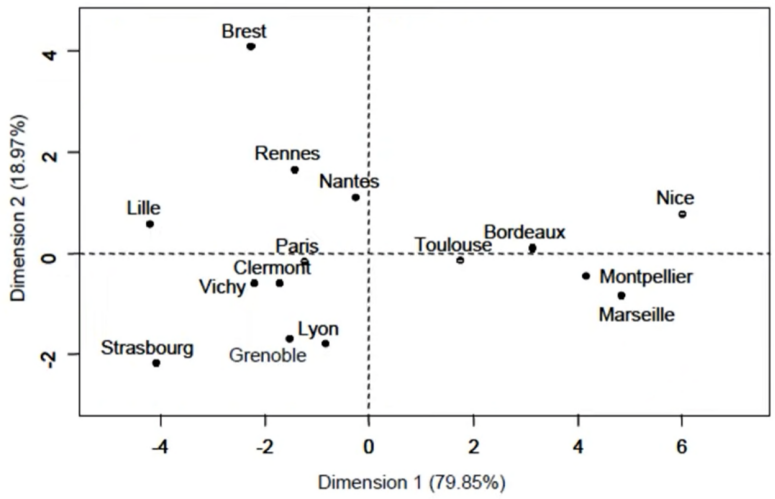
- Pour maximiser x, on prend le vecteur propre u_1 dont la valeur propre λ_1 est la plus grande
- La fonction pour l'ACP avec une dimension égale à 1 est donnée par : $y(\mathbf{x}) = u_1^T \mathbf{x}$

- L'ACP pour $M > 1$, on itère les calculs des vecteurs propres :
  - On cherche une autre projection qui maximise la variance (orthogonale aux projections précédentes)
  - On garde les M vecteurs propres $u_1, u_2, \ldots, u_M$ de S ayant les plus grandes valeurs propres $\lambda_1, \lambda_2, \ldots, \lambda_M$
- Résumé de l'algorithme ACP :
  - On calcule la matrice U des vecteurs propres de S
  - On ordonne les valeurs propres (ordre décroissant)
  - On déduit : $y(\mathbf{x}) = \left(\mathbf{U}_{:,1:M}\right)^T \mathbf{x}$

17

Ajustement des points : graphe individus

Exemple données de température :



18