01-05
Conseils et considérations pratiques

Sommaire

Sommaire

Petites décisions, grandes conséquences

- Les algorithmes de partitionnement représentent de puissantes techniques d'apprentissage non supervisé, cependant des **choix** doivent être faits !
  - Doit-on mettre les données à l'échelle ?
  - En partitionnement K-moyennes
    - Quelle valeur de K choisir ?
  - En partitionnement hiérarchique
    - Quel type de dissimilarité utiliser ?
    - Quelle méthode de lien choisir ?
    - À quelle hauteur couper le dendrogramme ?
  - Avec DBSCAN
    - Quelle densité utiliser ?

En fonction des choix réalisés, les résultats peuvent être complètement différents !

Petites décisions, grandes conséquences

- Les algorithmes de partitionnement représentent de puissantes techniques d'apprentissage non supervisé, cependant des **choix** doivent être faits !
  - Doit-on mettre les données à l'échelle ?
  - En partitionnement K-moyennes
    - Quelle valeur de K choisir ?
  - En partitionnement hiérarchique
    - Quel type de dissimilarité utiliser ?
    - Quelle méthode de lien choisir ?
    - À quelle hauteur couper le dendrogramme ?
  - Avec DBSCAN
    - Quelle densité utiliser ?

Il n'existe pas de réponse unique et meilleure que les autres à ces questions. Chaque solution permettant de mettre en lumière un aspect intéressant des données doit être considérée

résultats peuvent être complètement différents !

Sommaire

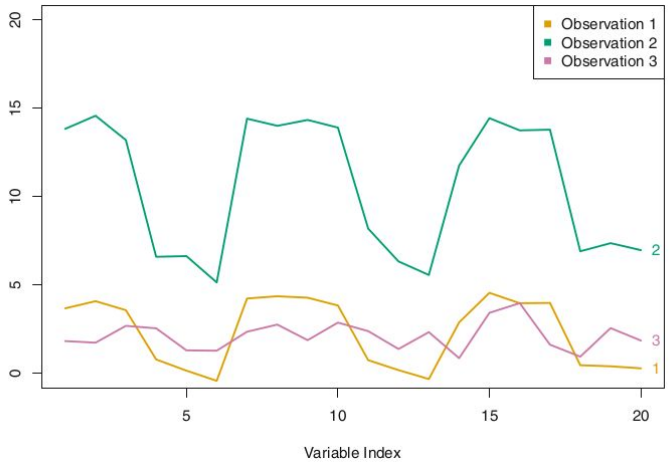
Mesures de distance

- Comme nous l'avons vu précédemment, il y a un lien fort entre **distance** et **dissimilarité** (ou inverse de la similarité)
- Le choix d'une mesure de distance est crucial doit être réalisé en connaissance des données. L'analyse exploratoire est donc une étape indispensable avec toute tentative de modélisation par partitionnement
- Parmi les distances les plus couramment utilisées, on trouve entre-autre
  - La distance euclidienne
  - La distance de Manhattan (city-block)
  - La similarité cosinus
  - **La distance basée sur la corrélation**
  - **Les distances basées sur la distribution des clusters (ex. Mahalanobis)**

Distance basée sur la corrélation

- Une première alternative possible à la distance euclidienne est la **distance basée sur la corrélation**

  → **Deux observations sont similaires si leurs variables explicatives sont corrélées**



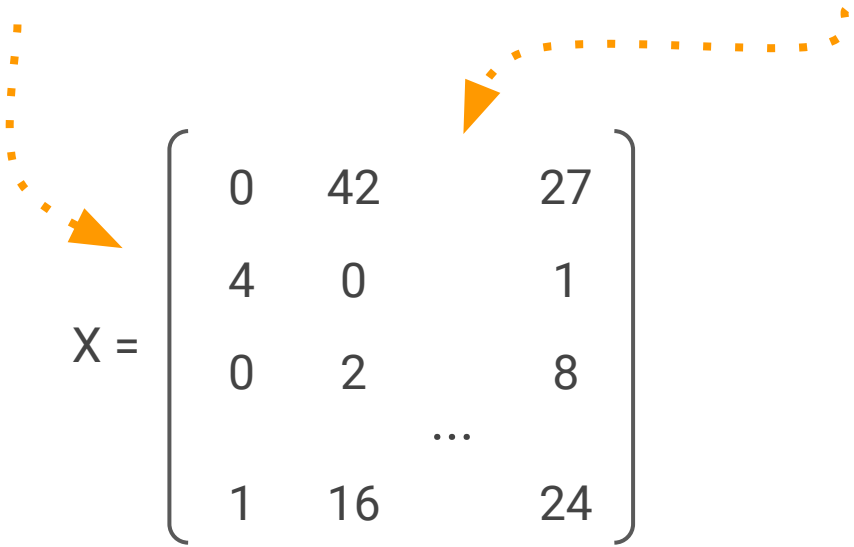
Extrait de [1] - Page 398

Exemple #1

- Prenons l'exemple d'une boutique en ligne souhaitant effectuer un partitionnement des consommateurs basé sur leurs achats passés
- L'objectif est d'identifier des **sous groupes de consommateurs similaires**, de manière à leur proposer des articles et publicités d'intérêt
- Remarque: nous allons voir ce type d'application beaucoup plus en détails lors de la partie 3 du cours dédiée aux Systèmes de recommandation

Exemple #1

- Supposons que les données soient regroupées dans une matrice **X** dont les lignes correspondent aux **consommateurs** et les colonnes aux **articles disponibles** à l'achat.



Exemple #1

- Supposons que les données soient regroupées dans une matrice **X** dont les lignes correspondent aux **consommateurs** et les colonnes aux **articles disponibles** à l'achat.

Les éléments de la matrice indiquent le nombre de fois qu'un acheteur donné à acheté un article donné

$$X = \begin{pmatrix} 0 & 42 & \cdots & 27 \\ 4 & 0 & & 1 \\ 0 & 2 & & 8 \\ \vdots & & \ddots & \\ 1 & 16 & & 24 \end{pmatrix}$$

Exemple #1

- Supposons que les données soient regroupées dans une matrice **X** dont les lignes correspondent aux **consommateurs** et les colonnes aux **articles disponibles** à l'achat.

Les éléments de la matrice indiquent le nombre de fois qu'un acheteur donné a acheté un article donné

$$X = \begin{bmatrix} 0 & 42 & \cdots & 27 \\ 4 & 0 & & 1 \\ 0 & 2 & & 8 \\ & & \cdots & \\ 1 & 16 & & 24 \end{bmatrix}$$

Quel est l'impact du choix de mesure de dissimilarité et laquelle utiliser pour regrouper les consommateurs en sous groupes ?

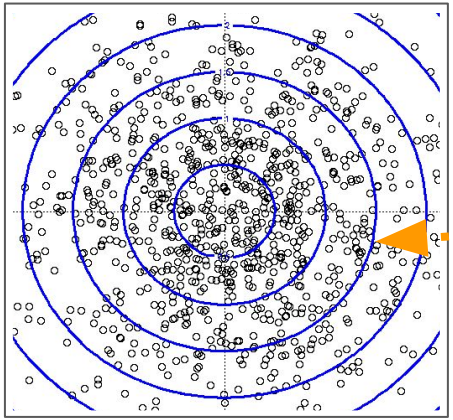
Exemple #1

- **Distance euclidienne**
  Les consommateurs ayant acheté **peu d'articles** seront regroupés (consommateurs occasionnels). Ceci n'a pas grande utilité commerciale

- **Distance basée sur la corrélation**
  Les consommateurs ayant acheté des **articles similaires** seront regroupés indépendamment des quantités achetées

  → Ainsi, la distance basée sur la corrélation est un meilleur choix pour cette application

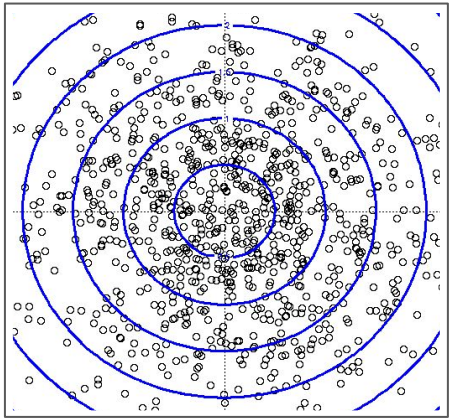
Example #2



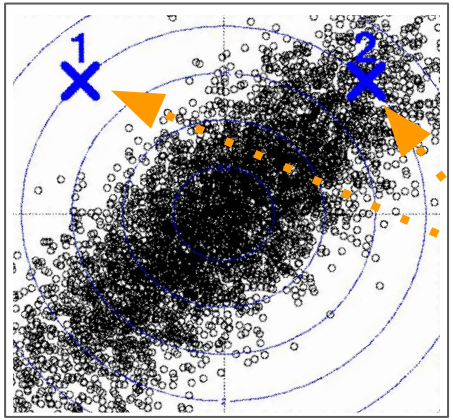
Point **uniformément** distribués
Distance euclidienne

Les lignes de contour représentent les points **équidistant** de l'origine

Example #2



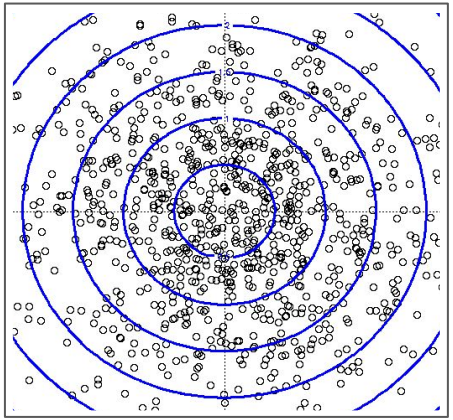
Point **uniformément** distribués
Distance euclidienne



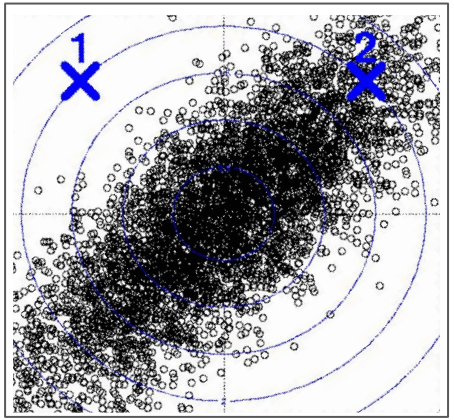
Point **normalement** distribués
Distance euclidienne

Ici les points 1 et 2 sont à une même distance euclidienne de l'origine

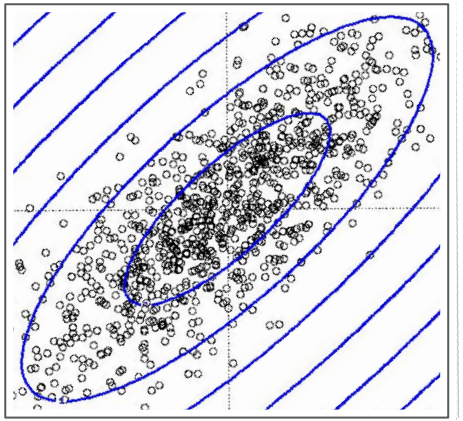
Example #2



Point **uniformément** distribués
Distance euclidienne



Point **normalement** distribués
Distance euclidienne



Point **normalement** distribués
Distance de **Mahalanobis**

Distance de Mahalanobis

- La distance de Mahalanobis est la **distance au centroïde normalisée**
- Considérons un cluster **C** de centroïde (c_1, ..., c_d) et d'écart-type (σ_1, ..., σ_d) ainsi qu'un point **P** (x_1, ..., x_d)
- Alors la **distance normalisée** entre **P** et **C** selon la dimension **i** s'exprime sous la forme:

$$y_i = \frac{(x_i - c_i)}{\sigma_i}$$

- La distance de Mahalanobis du point P au cluster C est:

$$\mathrm{MD} = \sqrt{\sum_{i=1}^{d} y_i^2}$$

Sommaire

Fléau de la dimension (1/2)

- Comme vu lors du cours *420-A52-SF, Algorithmes d'apprentissage supervisé*, les espaces (euclidiens ou non) à haute dimension montrent des propriétés étonnantes…
- Ces propriétés étonnantes, ou contre-intuitives, sont désignées par le **Fléau de la dimension** (curse of dimensionality)
- Notamment
  - La distance entre chaque points tend à être équivalente
  - Les vecteurs tendent à être orthogonaux deux-à-deux

Fléau de la dimension (2/2)

- Ceci peut naturellement poser problème lors de l'application du clustering (calcul de distance, …). Il convient dans ce cas de **réduire le nombre de dimensions**
  - Par Analyse en Composantes Principales (ACP)
  - Par sélection d'un sous-espace des variables
  - En utilisant d'autres types d'algorithmes
    - Correlation clustering
    - Projected clustering (PreDeCon, ...)
    - Approches hybrides (FIRES, …)

Sommaire

Partitionnement K-moyennes

- Rappel: la solution optimale est un problème **NP-hard**
- Complexité de la solution approchée ~ $O(kn)$
- Complexité linéaire pour la solution approchée …
- … mais l'algorithme des K moyennes est de nature itérative et **peut être lent à converger** !

Partitionnement hiérarchique

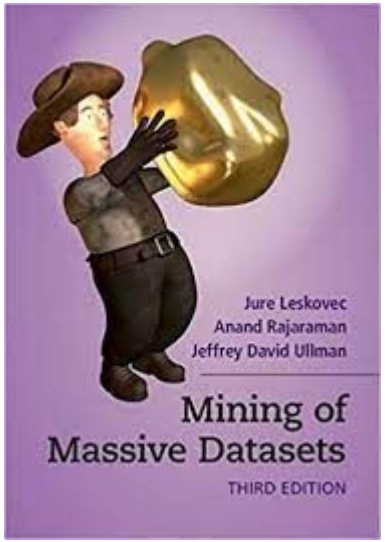
- Chaque étape requiert le **calcul des distances entre chaque paires de clusters**
- $O(n^2)$, $O((n - 1)^2)$, $O((n - 2)^2)$, ….
- $O(n^3)$! L'algorithme est **cubique** !
- Certaines optimisations permettent d'obtenir $O(n^2 \log n)$
- **Cela reste tout de même une forte limitation de l'algorithme**

Partitionnement DBSCAN

- Au pire, $O(n^2)$
- Selon la valeur de ε et si un index est utilisé, il est possible d'obtenir $O(n \log n)$

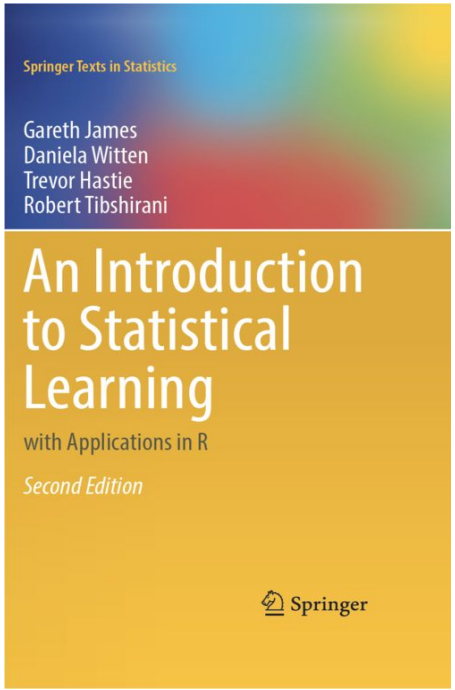
Complexité des algorithmes

- Les complexités précédentes montrent que les algorithmes de partitionnement K-moyennes, hiérarchique et DBSCAN **ne sont pas adaptés aux données volumineuses / massives**
- Il existe des algorithmes adaptés à ce type de données
  - **Algorithme BFR**
  - **Algorithme CURE**
  - **Implémentation MapReduce de l'algorithme de K-moyennes**
- Consulter la référence [3] pour en apprendre plus



Sommaire

Lectures recommandées



- Introduction to Statistical Learning with Applications in R Second edition (2021)

  → 12.4 Clustering Methods

Références

[1] CS229: Machine Learning - Stanford University

[2] Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani, “Introduction to Statistical Learning with Applications in R - Second edition”

[3] Jure Leskovec, Anand Rajaraman, and Jeffrey D. Ullman, “Mining of Massive Datasets”