Apprentissage non supervisé : Fouille de Données

M.-J. Huguet

INSA TOULOUSE

https://homepages.laas.fr/huguet

2018-2019

LAAS CNRS

Plan

1. Contexte : l'Intelligence Artificielle
2. Contexte : l'apprentissage automatique
3. Problème de clustering
4. Premières méthodes
5. Méthodes basées voisinage (densité) et basées graphes
6. Boite à outils
7. **Fouille de données**
8. **Réduction de dimensions** (Analyse en Composantes principales)

Sources

- **Introduction to Data Mining**
  - Livre et supports :
  - https://www-users.cs.umn.edu/~kumar001/dmbook/index.php
- **Mining on Massive Data Set :**
  - livre et MOOC associé
  - http://www.mmds.org/
- **Cours CNAM**
  - Ingénierie de la fouille et de la visualisation de données massives
  - http://cedric.cnam.fr/vertigo/Cours/RCP216/
- **Data Mining and Constraint programming**
  - https://link.springer.com/book/10.1007%2F978-3-319-50137-6

Plan – section 7

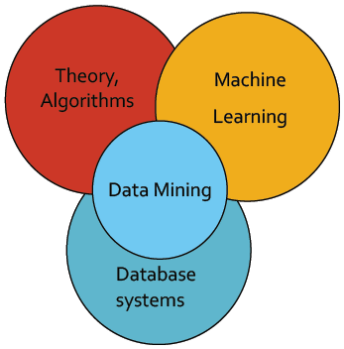
7. **Fouille de données**
   1. Contexte
   2. Extraction de motifs
   3. Fouille de graphes / réseaux sociaux
8. **Réduction de dimension**

Contexte : Fouille de données (1)

- **Data Mining**
  - Processus de découverte de connaissances dans des collections de données
    - Correctes, non triviales, intéressantes, exploitables, …
  - Collection de grande dimension
    - Problématique de représentation, stockage, accès aux données
  - Problématiques autour des données
    - Stockage (systèmes)
    - Gestion (représentation, accès, bases/entrepôt de données, flux, …)
    - Analyse
      - Prédiction
      - Découverte de connaissances

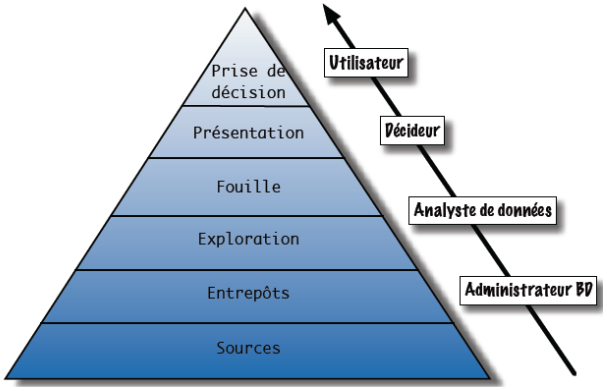
Contexte : Fouille de données (2)

- **Méthodes d'analyse**
  - problématiques d'apprentissage automatique
    - Analyse Descriptive / Prédictive
      - Comprendre les données
      - Prédire des valeurs futures
  - **Passage à l'échelle**
    - Réduction de dimensions
    - Algorithmique
      - Structures de données
      - Complexité
      - Optimisation de code
    - Architectures de calcul



Contexte : Applications - Fouille de données (1)

- **Informatique décisionnelle**
  - KDD : Knowledge Discovery Databases
  - Processus complet
    - Pre-processing des données :
      - Préparation des données (nettoyage, normalisation, ….)
    - Fouille de données
    - Post-processing des résultats :
      - Validité et pertinence des connaissances obtenues / besoins initiaux,
      - Visualisation



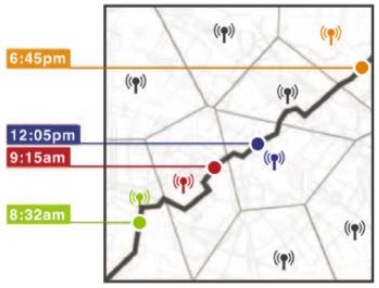
Contexte : Applications - Fouille de données (2)

- **Découverte de motifs**
  - Pattern similaires, pattern fréquents, association de pattern
  - Analyse de paniers de consommation, de documents, ….
- **Réseaux sociaux**
  - Extraction de communautés, identification de rôles, diffusion d'information, recherche d'information, …
- **Systèmes de recommandation**
  - Informations ciblées (filtrage) : centrée sur les objets, les utilisateurs ou l'environnement (réseau) social
- **Détection de changements / d'anomalies**
  - Phénomènes imprévus mais ayant du sens
    - Sécurité (bancaire, systèmes informatiques, réseaux), Scientifique (monitoring de systèmes, de personnes, ….)

Contexte : Vie Privée (1)

- **Vie privée**
  - Données personnelles identifiantes
    - identité, préférences, habitudes, âge, orientations politiques, religieuses, . . .
  - Liée du contexte

  - **Divulgation** : Unique in the Crowd: The privacy bounds of human mobility. de Montjoie et. al, 2013
    - *"In fact, in a dataset where the location of an individual is specified hourly, and with a spatial resolution equal to that given by the carrier's antennas, four spatio-temporal points are enough to uniquely identify 95% of the individuals."*
    - *Collecte : 15 mois d'appels / SMS*
    - *Données spatio-temporelles*
    - *1,5 millions d'abonnées*



Contexte : Vie Privée (2)

- **Protection de la vie privée**
  - Sécurité des systèmes informatiques et des processus métiers

  - **Cadre légal** (Europe)
    - GDPR (RGPD en français):
      - Application mai 2018
      - Sanctions financières importantes

General Data Protection Regulation

CNIL.

  - **Privacy by design**
    - Protéger la vie privée ➔ réduire les traces laissées
      - Proactif et non réactif
      - Protection de la vie privée par défaut

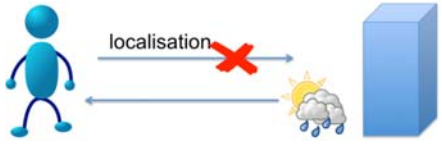
Contexte : Vie Privée (3)

- **Techniques de protection de la vie privée**
  - **Anonymisation**
    - Modifier les données pour rendre l'identification impossible (ou trop couteuse)
    - Difficulté : risque de recoupement sur données massives
  - **Pseudo-anonymisation**
    - Données identifiantes ➔ pseudonyme
    - Ex : fonction de hachage à partir d'une clé
  - **Chiffrement**
    - Crypter des données : seuls les systèmes ayant la clé peuvent les déchiffrer
  - **Calcul multi-partie sécurisé**
    - Mise en œuvre sécurisée de méthodes distribuées
    - Branche de la cryptographie

Contexte : Vie Privée (4)

- **Privacy by design**
  - Architecture centralisée avec serveur tiers de confiance
  - Architecture décentralisée et algorithmes distribués (protocoles)

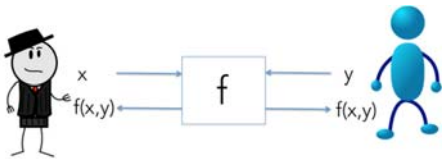
  - **Ex : Services géo-localisés**
    - Applications utilisant la géo-localisation pour fournir un service
      - Environnement de confiance versus de non confiance
        - **Confiance** : le fournisseur de services doit respecter la politique de confidentialité
        - **Non confiance** : le fournisseur de services est considéré comme un adversaire



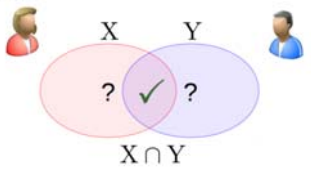
- **Types d'adversaires**
  - Semi-honnête :
    - tente d'inférer des informations à partir des échanges
  - Malveillant :
    - toute stratégie pour découvrir des informations

Contexte : Vie Privée (5)

- **Calcul multi-partie sécurisé**



  - Les participants n'apprennent rien de plus que leurs entrées et la sortie de la fonction f
  - Exemple : problème du millionnaires (Yao 1982) :
    - Max : savoir qui est le plus riche sans révéler ce que l'on possède
    - Intersection sécurisée



  - **Chiffrement « homomorphe »**
    - Déchiffrement d' une opération sur données chiffrée = opération sur données en clair
    - Pas de connaissance des valeurs ni du résultat

Plan – section 7

7. **Fouille de données**
   1. Contexte
   2. Extraction de motifs
   3. Fouille de graphes / réseaux sociaux
8. **Réduction de dimension**

Découverte de patterns (1)

- **Ensemble d'items : $I = \{i_1, \dots, i_n\}$ d'une base de données D**
  - ItemSet : $X \subseteq I$
  - Frequence de X : $freq(X)$: nombre de transactions de D contenant X
  - Support de X : $supp(X)$: fraction des transactions de D contenant X
    - $supp(X) = \frac{freq(X)}{|D|}$

ID	Items					
1	A	B				
2	A		C	D	E	
3		B	C	D		F
4	A	B	C	D		
5	A	B	C			F

$I=\{A, B, C, D, E, F\}$
$X=\{A,B,C\}$

$freq(X)=2$
$supp(X)=2/5$

- **Sous ensemble fréquent (FrequentItemSet)**
  - Pour une base D et un seuil $\bar{s}$, déterminer tous les ItemSet X tel que $supp(X) \geq \bar{s}$

Découverte de patterns (2)

- **Règle d'association (Association Rule)**
  - Exprimer une relation entre deux itemsets : $X \rightarrow Y$ (co-occurrence de X et Y)
    - avec X et Y deux itemsets tels que $X \cap Y = \emptyset$
  - Support : $supp(X \rightarrow Y) = supp(X \cup Y)$
  - Confiance : $conf(X \rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)} = \frac{freq(X \cup Y)}{freq(X)}$

ID	Items					
1	A	B				
2	A		C	D	E	
3		B	C	D		F
4	A	B	C	D		
5	A	B	C			F

Exemple :
$\{A,B\} \rightarrow \{E,F\}$
$\{B,C\} \rightarrow \{C\}$

$supp(\{B,C\} \rightarrow \{D\}) = supp(\{B,C,D\}) = 2/5$

$conf(\{B,C\} \rightarrow \{D\}) =$
$freq(\{B,C,D\})/freq(\{B,C\}) = 2/3$

- **Problème** : Obtenir toutes les règles d'association $X \rightarrow Y$ telles que
  - $supp(X \cup Y) \geq \bar{s}$ et $conf(X \rightarrow Y) \geq \bar{c}$

Découverte de patterns (3)

- **Applications**
  - Etudes de paniers de consommation
    - 1 ensemble de produits
    - Chaque ligne : 1 panier de produits achetés
      - Frequent Itemset : déterminer les sous-ensembles d'items qui apparaissent souvent simultanément dans un panier
      - Suggestion d'achats : si un panier contient un ensemble de produits X, il est susceptible de contenir les produits Y
  - **Analyse de documents**
    - Déterminer des co-occurrences de termes
    - Détection de plagiats
  - **Bio-informatique**
    - Corrélation entre marqueurs génétiques et maladies

Découverte de patterns (4)

- **Combinatoire**
  - Découverte de motifs fréquents
  - Découverte de règles d'associations
    - Si n items :
      - 2^n sous ensembles possibles
        - $2^n - 1$ en enlevant l'ensemble vide
      - $\sum_{k=1}^{n-1}\binom{n}{k} = \sum_{k=1}^{n-1}\frac{n!}{k!(n-k)!}$ règles d'association possibles
        - $2^n - 2$ en enlevant $\emptyset \rightarrow X$ et $X \rightarrow \emptyset$
  - L'énumération exhaustive est prohibitive

Découverte de patterns (5)

- **Découverte de règles d'association**
  - Méthode Brute-Force
    - Énumérer toutes les règles d'association
    - Calculer les valeurs de support et de confiance
    - Sélectionner celles respectant les seuils définis
  - Impossible de passer à l'échelle
  - **Observation** :
    - Les règles d'association provenant d'un **même** itemset ont les mêmes valeurs de support mais de valeurs de confiance différentes
      - $X \rightarrow Y, supp(X \rightarrow Y) = supp(X \cup Y)$
    - Découpler le problème en 2 étapes

Découverte de patterns (6)

ID	Items					
1	A	B				
2	A		C	D	E	
3		B	C	D		F
4	A	B	C	D		
5	A	B	C			F

- Mêmes valeurs de support
- Valeurs de confiance différentes

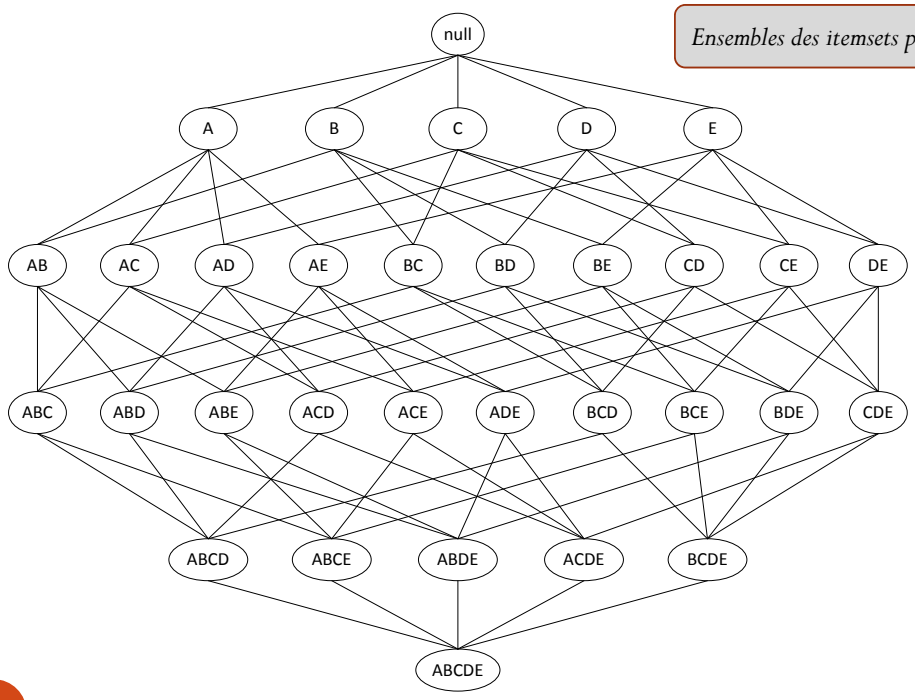
Règles d'association avec les 3 items B, C et D

	Support *freq(BCD)/5*	Confiance *freq(BCD)/freq(ant)*
{B,C}→{D}	0,4 (2/5)	0,67 (2/3)
{B,D}→{C}	0,4	1,00 (2/2)
{C,D }→{B}	0,4	0,67
{D}→{B,C}	0,4	0,67
{C}→{B,D}	0,4	0,50
{B}→{C,D}	0,4	0,50

- **Déterminer les règles d'association**
  1. Générer les itemsets fréquents (ayant un support supérieur à un seuil)
  2. Générer les règles d'association à partir de chaque itemset fréquent
     - Enumérer les associations possibles pour les itemsets fréquents
  - Toujours problème de passage à l'échelle pour générer les itemsets fréquents …

Découverte de patterns (7)

Ensembles des itemsets pour $I = \{A, B, C, D, E\}$



Méthode « Apriori » (1)

- But : réduire le nombre de sommets candidats à explorer
- **Principe** :
  - Si un itemset X est fréquent alors ses sous-ensembles sont également fréquents
  - $\forall X, Y : (X \subseteq Y) \Rightarrow supp(X) \geq supp(Y)$
    - Le support d'un itemset ne peut pas dépasser celui de ses sous-ensembles
    - Propriété d'anti-monotonie

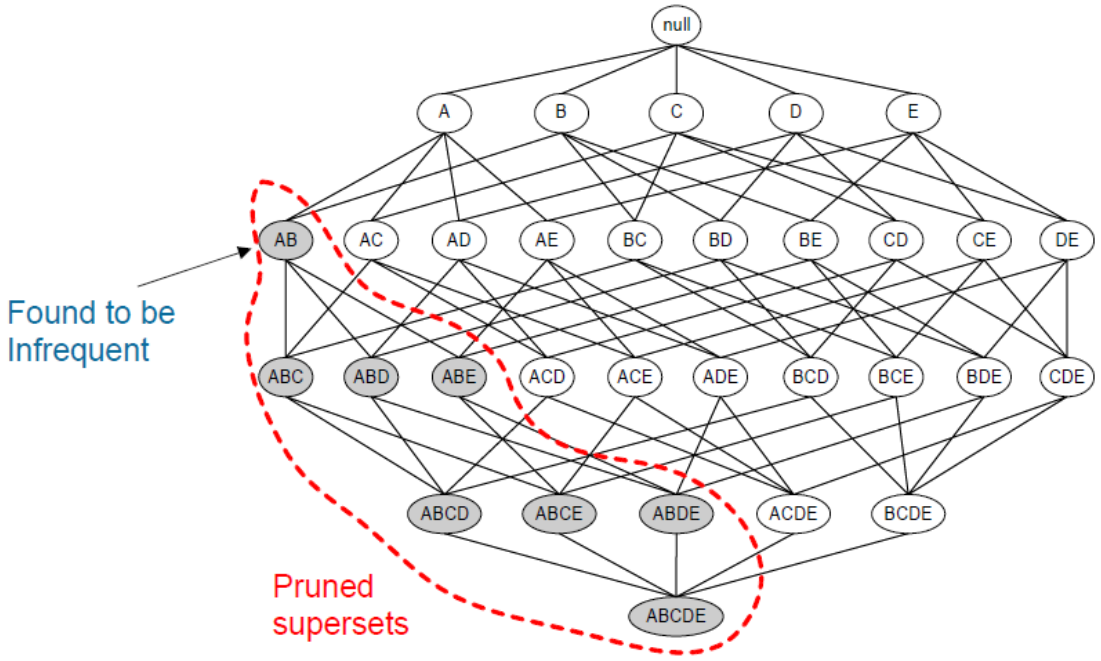
ID	Items					
1	A	B				
2	A		C	D	E	
3		B	C	D		F
4	A	B	C	D		
5	A	B	C			F

1-Item	Freq	Supp
A	4	0,8
B	4	0,8
C	4	0,8
D	3	0,6
E	1	0,2
F	2	0,4

2-Items	Freq	Supp
A,B	3	0,6
A,C	3	0,6
A,D	2	0,4
A,E	1	0,2
A,F	1	0,2
B,C	3	0,6
B,D	2	0,4
B,E	0	0
B,F	2	0,4
C,D	3	0,6
C,E	1	0,2
C,F	2	0,4
D,E	1	0,2
D,F	1	0,2
E,F	0	0

Méthode « Apriori » (2)

- **Elagage de l'arbre**



Méthode « Apriori » (3)

- **Application**

ID	Items					
1	A	B				
2	A		C	D	E	
3		B	C	D		F
4	A	B	C	D		
5	A	B	C			F

Seuil Fréquence = 3
Seuil Support = 3/5=0,6

1-Item	Freq	Supp
A	4	0,8
B	4	0,8
C	4	0,8
D	3	0,6
E	1	0,2
F	2	0,4

2-Items	Freq	Supp
{A,B}	3	0,6
{A,C}	3	0,6
{A,D}	2	0,4
{B,C}	3	0,6
{B,D}	2	0,4
{C,D}	3	0,6

3-Items	Freq	Supp
{A,B,C}	2	0,4

Méthode « Apriori » (4)

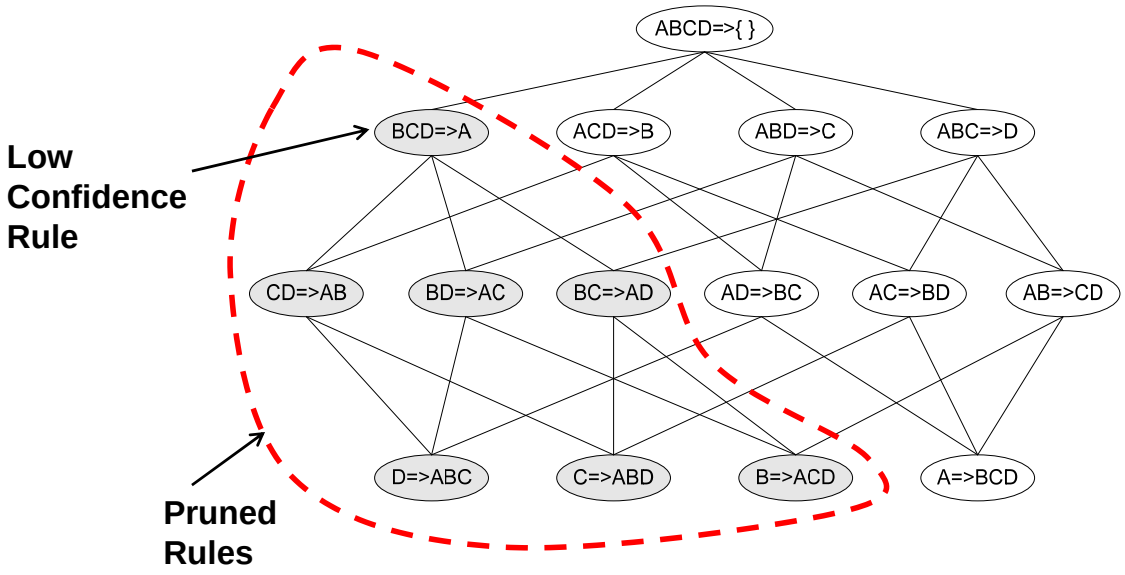
- **Déterminer les règles d'association**
  - Partir d'un itemset fréquent X et déterminer tous les sous-ensembles L tels que $L \rightarrow X - L$ respectant le seuil de confiance $conf(L \rightarrow X - L) \geq \bar{c}$
    - Ex : itemset fréquent $X = \{A, B, C, D\}$, l'ensemble des règles d'association est :

{A}→{BCD}	{B}→{ACD}	{C}→{ABD}	{D}→{ABC}		
{AB}→{CD}	{AC}→{BD}	{AD}→{BC}	{BC}→{AD}	{BD}→{AC}	{CD}→{AB}
{ABC}→{D}	{ABD}→{C}	{ACD}→{B}	{BCD}→{A}		

  - Propriétés : soit $X = \{A, B, C, D\}$; [$conf(X \rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)} = \frac{freq(X \cup Y)}{freq(X)}$]
    - Pour un même itemset, la confiance d'une règle est anti-monotone par rapport aux nombres de termes du second membre :
      - $conf(\{A, B, C\} \rightarrow \{D\}) \geq conf(\{A, B\} \rightarrow \{C, D\}) \geq conf(\{A\} \rightarrow \{B, C, D\})$

Méthode « Apriori » (5)

- **Elagage de l'arbre**



Méthode « Apriori » (6)

- **Application**

Seuil Fréquence = 3
Seuil Support = 3/5=0,6

ID	Items					
1	A	B				
2	A		C	D	E	
3		B	C	D		F
4	A	B	C	D		
5	A	B	C			F

1-Item	Freq	Supp
A	4	0,8
B	4	0,8
C	4	0,8
D	3	0,6
E	1	0,2
F	2	0,4

2-Items	Freq	Supp
{A,B}	3	0,6
{A,C}	3	0,6
{A,D}	2	0,4
{B,C}	3	0,6
{B,D}	2	0,4
{C,D}	3	0,6

3-Items	Freq	Supp
{A,B,C}	2	0,4

Seuil Confiance=0,8

	Supp	Conf
{A}→{B}	0,6	0,75
{B}→{A}	0,6	0,75
{A}→{C}	0,6	0,75
{C}→{A}	0,6	0,75
{B}→{C}	0,6	0,75
{C}→{B}	0,6	0,75
{C}→{D}	0,6	0,75
{D}→{C}	0,6	1

Méthode « Apriori » (7)

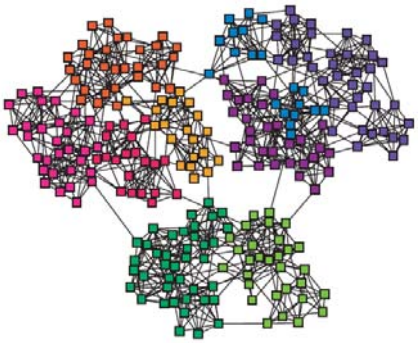
- **Nombreuses optimisations de la méthodes**
  - Structures de données,
  - Génération de règles redondantes : élagage
- Mesures complémentaires : intérêt des connaissances produites
  - Pondérer la confiance dans une règle par la fréquence d'apparition du second membre
    - $X \rightarrow Y$ peut être jugée pertinent seulement parce que Y apparaît souvent indépendamment de X
    - Intérêt $(X \rightarrow Y) = conf(X \rightarrow Y) - supp(Y)$
- Extensions : recherche de séquences de patterns
  - Transactions successives sur un horizon temporel

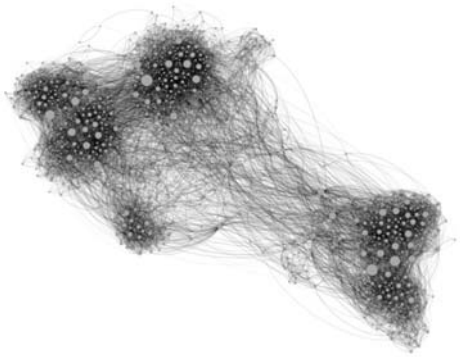
Plan – section 7

7. **Fouille de données**
   1. Contexte
   2. Recherche de motifs dans des bases
   3. Fouille de graphes
8. **Réduction de dimension**

Réseaux sociaux (1)

- **Réseau social**
  - Ensemble d'entités en interaction
    - Travaux en sociologie, histoire, sciences humaines et sociales ….
    - Réseaux informatique (internet, web, d'échanges, de capteurs, …)
    - Transport (réseau aérien, routier, ferroviaire, d'énergie, …)
  - Données du réseau social : données des interactions, des entités





Réseaux sociaux (2)

- **Une expérience : « phénomène du petit monde »** (Milgram, 1967)
  - **But** :
    - montrer que chaque individu peut être relié à n'importe quel autre par une chaine de taille réduite
  - **Expérience** :
    - faire transiter des lettres du Nebraska au Massachussetts de proche en proche, chaque transition est supposée rapprocher de la destination
      - Peu de lettres sont arrivées à destination
      - Chaînes courtes (5 à 6 intermédiaires)

  - **Conclusion**
    - Il existe des chaines courtes
    - Les entités intermédiaires ont pu les déterminer sans connaissance globale
    - Si une chaine n'est pas trouvée : cela ne veut pas dire qu'elle n'existe pas
    - Si une chaine de longueur x est trouvée, cela ne veut pas dire qu'il n'en existe pas une plus courte

Réseaux sociaux et graphes (1)

- **Graphes**
  - Modèles naturels pour les réseaux sociaux

- **Graphe** : $G = (X, E)$
  - X : ensemble des **sommets**
  - E : ensemble de **relations** binaires
    - non orientées (**arêtes**) : $x_i - x_j$
    - orientées (**arcs**) : $x_i \rightarrow x_j$
  - **Ordre d'un graphe :**
    - nombre de sommets
  - **Densité d'un graphe**
    - Nombre de relations existantes / nombre de relations possibles

- **Voisinage d'un sommet x : $N(x)$**
  - Ensemble des sommets y tels que $(x, y) \in E$
  - Il existe une arête (ou arc) entre x et y
- **Degré d'un sommet x : d_x**
  - nombre de voisins

- **Graphe partiel :**
  - obtenu en retirant certaines relations
- **Sous Graphe**
  - Graphe obtenu en supprimant certains sommets et les relations associées à ses sommets

Réseaux sociaux et graphes (2)

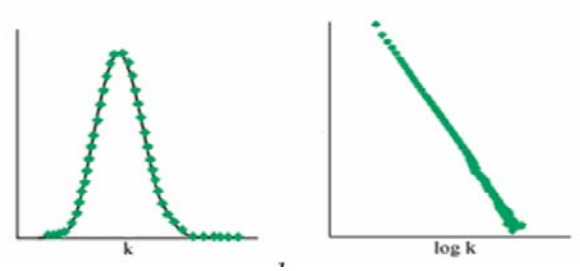
- **Graphes**
  - Compréhension des réseaux sociaux
    - Structure et analyse des propriétés d'un réseau social
    - Evolution de la structure d'un réseau social
  - Exemple : Expérience « petit monde » reproduite sur d'autres réseaux sociaux
    - Facebook : Résultats similaires
- **Spécificité**
  - Sommets : objets du réseau
  - Arêtes : relations du réseau
  - Ensembles de grande taille, existence d'attributs, données hétérogènes, …

Réseaux sociaux et graphes (3)

- **Problématiques diverses**
  - **Modélisation** :
    - Quelle est la structure d'un réseau social ?
    - Peut-on obtenir un réseau artificiel ayant la même structure ?
  - **Analyse** :
    - Quelles sont les propriétés d'un réseau social ?
    - Quelles sont les évolutions temporelles ?
  - **Algorithmique** :
    - Comment calculer sur des grands graphes ?
  - **Métrologie** :
    - Comment mesurer des réseaux réels ?
  - **Identification** :
    - De communautés, de leader, ….

Caractéristiques d'un réseau social (1)

- **Loi de distribution des degrés**
  - Graphe aléatoire classique : loi gaussienne
  - Réseau social : loi de puissance
  - **Conséquence**
    - 20% des sommets concentrent 80% des liens
    - Identification de
      - Sommets avec beaucoup de liens sortant
      - Sommets avec beaucoup de liens entrants
    - Sensibilité aux attaques sur les sommets ayant beaucoup de liens



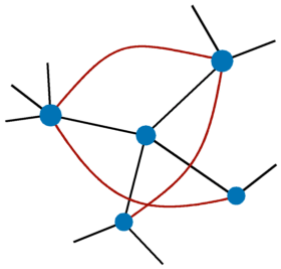
Caractéristiques d'un réseau social (2)

- **Diamètre**
  - **Plus court chemin entre 2 sommets** = valeur minimisant de la somme des valuations des arêtes à parcourir
    - Algorithme de Dijkstra
    - Graphe non valué : nombre d'arêtes ➔ Algorithme de parcours en largeur
  - **Distance moyenne** = moyenne des plus courts chemins
  - **Diamètre** = excentricité maximale = valeur maximale des distances entre toutes les paires de sommets
    - Calculer tous les plus courts chemins 2 à 2; conserver le max
  - **Rayon** = excentricité minimale = Plus petite distance à laquelle peut se trouver un sommet de tous les autres

- En pratique : diamètre en $O(\log n)$ où n = nombre de sommets
  - Graphes aléatoires et réseaux sociaux

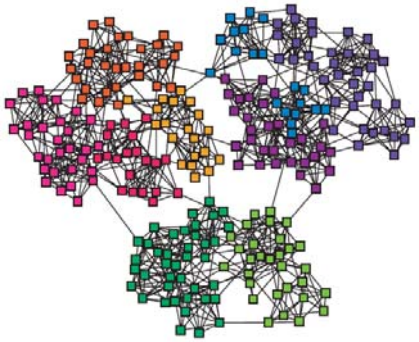
Caractéristiques d'un réseau social (3)

- **Coefficient (Taux) de clustering / agglomération / connexion**
  - Pour un sommet x de degré d_x:
    - Nombre de relations entre les voisins de x : $|e_{yz} \in E \,/\, y, z \in N(x)|$
    - Par rapport au nombre d'arêtes total (si clique) : $\frac{d_x \times (d_x - 1)}{2}$
    - Coefficient : $c(x) = 2 * \frac{|e_{yz} \in E, y, z \in N(x)|}{d_x \times (d_x - 1)}$
    - Pour un graphe : $c(G) = \frac{\sum_{i=1}^{n} c(i)}{n}$



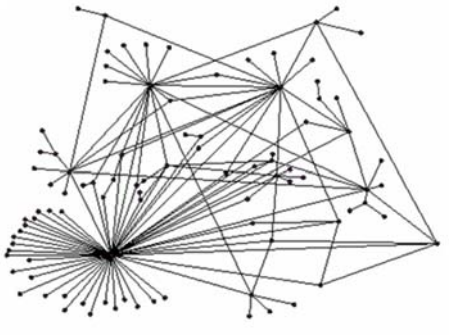
Caractéristiques d'un réseau social (4)

- **Composantes connexes**
  - Ensemble maximal de sommets tels qu'il existe un chemin entre toute paire de l'ensemble
  - Graphe connexe : tous les sommets sont dans la même composante connexe
- **Communautés**
  - Ensemble de sommets avec une forte densité de relation et peu de relations en dehors de l'ensemble

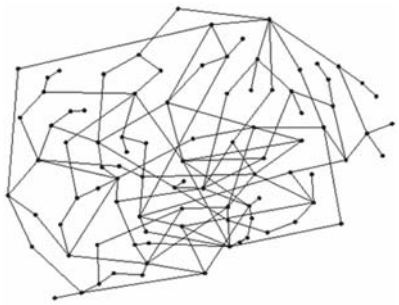


Caractéristiques d'un réseau social (5)

- **En résumé**
  - Faible densité
  - Fort taux d'agglomération
  - Distribution de degrés très hétérogènes
  - Composante connexe de grande taille
  - Présence de communautés
  - Distance moyenne faible
  - Deviennent de plus en plus denses et diamètre diminue avec le temps



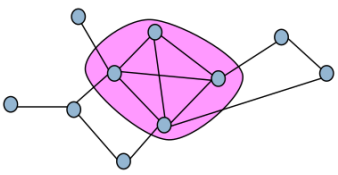
  - **Propriétés différentes des graphes aléatoires**

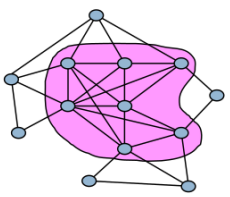


Caractéristiques d'un réseau social (6)

- **Nombreuses autres mesures**
  - Centralité
    - Pour un sommet x, plusieurs mesures de centralité
      - estimer à quel point ce sommet est "central" dans le réseau
    - Centralité de degré = d_x
    - Centralité d'intermédiarité (betweenness) = nombre de PCC passant par le sommet
  - ….

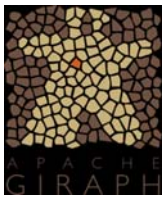
- **Structures particulières**
  - Clique
    - Sous graphe complet
  - Composante k-connexe
    - Ensemble de sommets tels qu'il existe k-chemins disjoints entre toute paire
  - ….





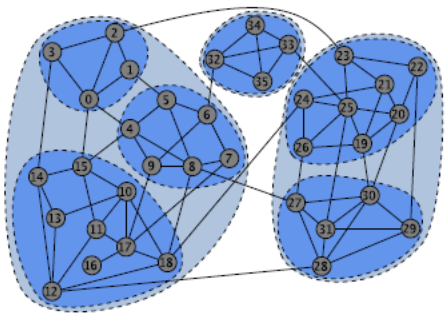
Caractéristiques d'un réseau social (7)

- **Taille des données**
  - Internet = Millions de sommets (routeurs)
  - Facebook = plus de 800 millions d'utilisateurs actifs
  - Web = Google connaît plus de 1 000 milliards d'URL distinctes
- **Besoin d'algorithmes efficaces**
  - Ex : calcul de diamètre en $O(n \times m)$
    - Problème polynomial mais …
    - Approximation en $O(m)$
  - Recherche d'approximation linéaire
    - et de preuve de ces approximations
- **Différents outils**



Partitionnement de graphes (1)

- **Détection de communautés**

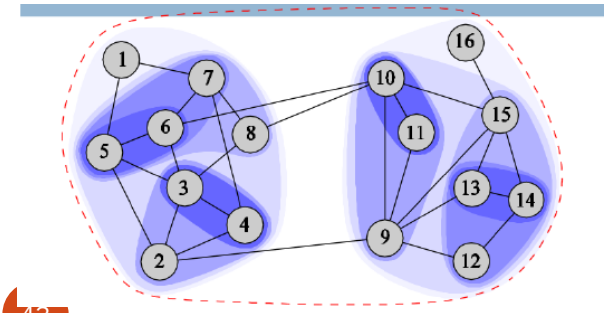


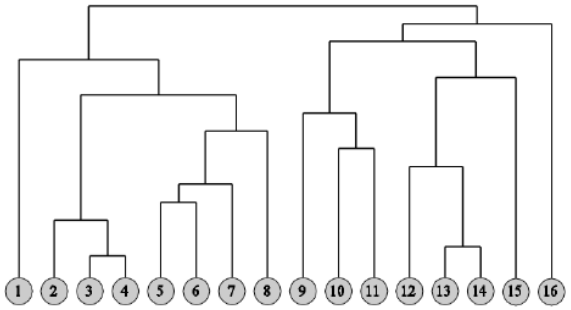
  - Partitionner un graphe :
    - Chaque sommet appartient à un et un seul cluster
    - Note : communautés avec recouvrement …

Partitionnement de Graphes (2)

- **Adaptation de méthodes standards**
  - **Méthode hiérarchique ascendante**
    1. Associer une communauté (un cluster) à chaque sommet
    2. Calculer une distance entre chaque paire de communautés
    3. Fusionner les deux communautés les plus proches
    - Retour étape 2

    - Distance entre communautés : min, max, moyenne entre paires de sommets





Partitionnement de Graphes (3)

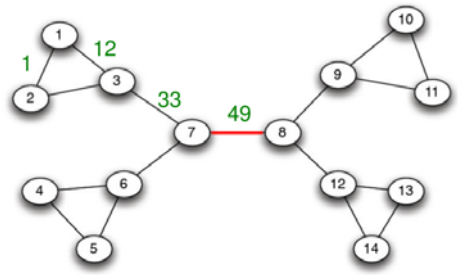
- **Adaptation de méthodes standards**
  - **Méthode hiérarchique descendante**
    1. Déterminer la centralité de chaque arête (betweenness)
       - Nombre de plus courts chemins passant par l'arête
    2. Retirer l'arête de plus forte centralité et regrouper les sommets
    3. Mettre à jour les centralités des arêtes affectées par la suppression
    - Retour étape 2 jusqu'à plus d'arêtes

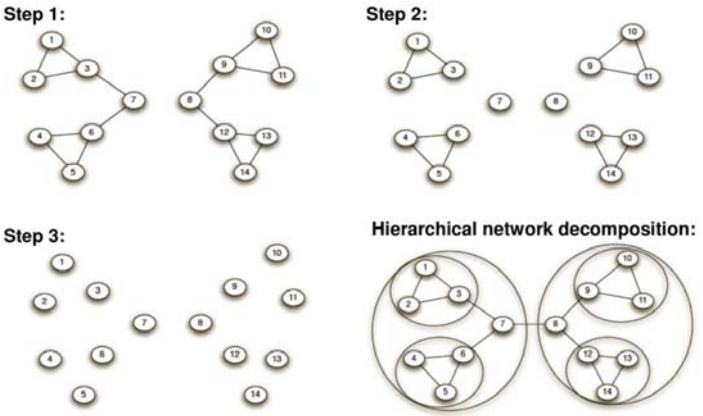
    - A chaque étape, les composantes connexes sont les communautés

    - Algorithme de Girvan et Newman (2002)

Partitionnement de Graphes (4)

- **Illustration algorithme**

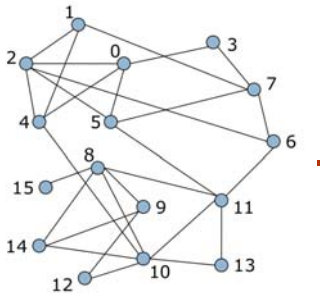




Partitionnement de Graphes (5)

- **Variantes**
  - Définir la notion de cluster de manière plus formelle : la modularité
  - **Modularité** Q (module == cluster)
    - Différence entre
      - le nombre de liens présents dans un module et
      - le nombre de liens attendus dans ce module pour un graphe aléatoire
    - Pour toute partition S en communautés
      - $Q = \sum_{s \in S}(nb\ liens \in s\ - nb\ liens\ attendus \in s)$
      - $Q = \sum_{s \in S}\left[\frac{l_s}{m} - \left(\frac{d_s}{2m}\right)^2\right]$ (avec l_s nombre de lien de s et d_s « degré » de s
    - Intervalle [-1;1]

Partitionnement de Graphes (6)

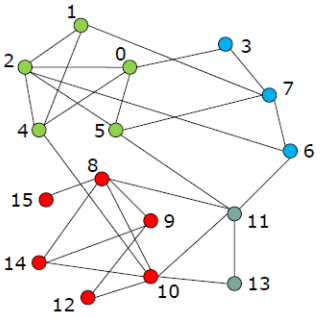


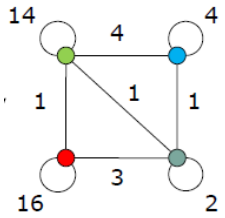
- **Algorithme utilisant la modularité**
  - Objectif : maximiser la modularité
  - Principe :
    - algorithme glouton
    - Évaluation incrémentale de la modularité
  - Méthode de Louvain (2008)
    1. Associer une communauté (un cluster) à chaque sommet

    Répéter
    1. Supprimer le sommet x de sa communauté
    2. Insérer x dans la communauté voisine qui maximise ΔQ

    Graphe « agrégé

    Arrêt : maximum local atteint ou gain de modularité faible
  - Complexité limitée





Partitionnement de Graphes (7)

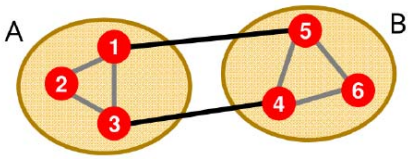
- **Performances algorithme**
  - Valeur modularité / Temps de calcul
    - Blondel, V. D., Guillaume, J.-L., Lambiotte, R., Lefebvre, E. *Fast unfolding of communities in large networks*. Journal of Statistical Mechanics: Theory and Experiment 2008 (10), P1000. https://arxiv.org/pdf/0803.0476v2.pdf

	Karate	Arxiv	Internet	Web nd.edu	Phone	Web uk-2005	Web WebBase 2001
Nodes/links	34/77	9k/24k	70k/351k	325k/1M	2.6M/6.3M	39M/783M	118M/1B
CNM	.38/0s	.772/3.6s	.692/799s	.927/5034s	-/-	-/-	-/-
PL	.42/0s	.757/3.3s	.729/575s	.895/6666s	-/-	-/-	-/-
WT	.42/0s	.761/0.7s	.667/62s	.898/248s	.56/464s	-/-	-/-
Our algorithm	.42/0s	.813/0s	.781/1s	.935/3s	.769/134s	.979/738s	.984/152mn

Table 1. Summary of numerical results. This table gives the performances of the algorithm of Clauset, Newman and Moore [8], of Pons and Latapy [7], of Wakita and Tsurumi [16] and of our algorithm for community detection in networks of various sizes. For each method/network, the table displays the modularity that is achieved and the computation time. Empty cells correspond to a computation time over 24 hours. Our method clearly performs better in terms of computer time and modularity. It is also interesting to note the small value of Q found by WT for the mobile phone network. This bad modularity result may originate from their heuristic which creates balanced communities, while our approach gives unbalanced communities in this specific network.

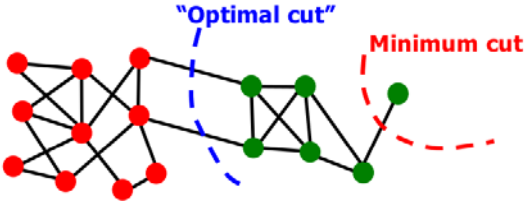
Partitionnement de Graphes (8)

- **Minimisation de la coupe (Min Cut)**
  - Minimiser le nombre de liens inter-communautés



Cut(A,B)=2

  - Difficultés

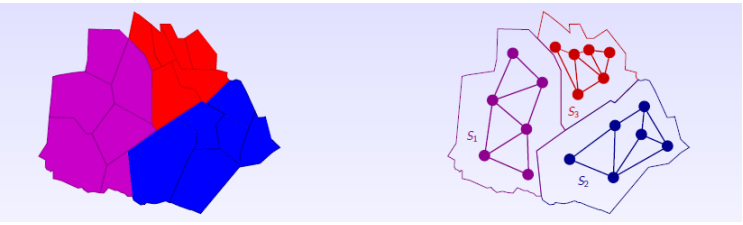


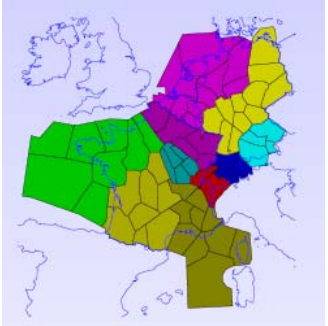
Min Cut=1

  - Variante pour évaluer un « bon » partitionnement : coupe normalisée
    - $nCut(A,b) = \frac{cut(A,B)}{nb\_ext(A)} + \frac{cut(A,B)}{nb\_ext(B)}$
    - Avec $nb\_ext(X)$ le nombre de liens ayant une extrémités dans X

Partitionnement de Graphes (9)

- **Résolution du problème Min cut**
  - Méthode d'optimisation combinatoire
    - Exacte ou approchée
    - Variantes avec des arêtes pondérés
      - Exemple : partitionnement espace aérien (thèse Bichot 2012)



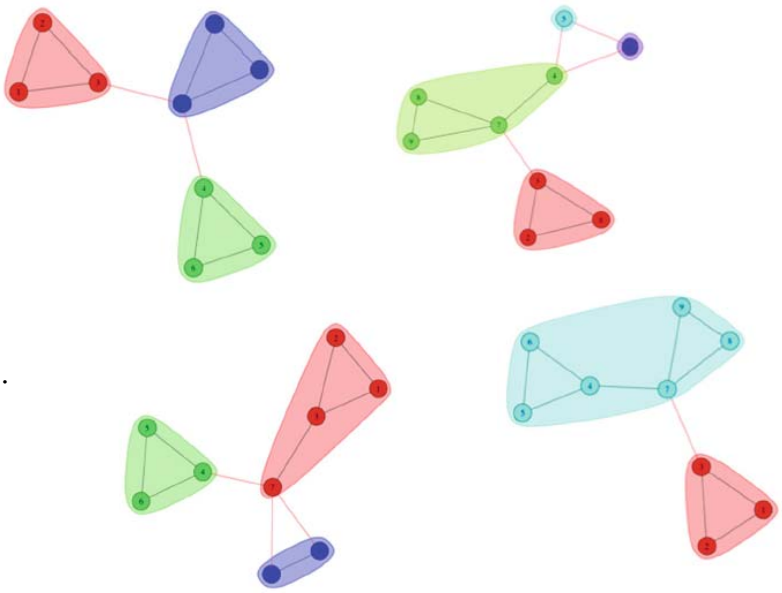


Partitionnement de Graphes (10)

- **Propagation de labels (**Raghavan et al., 2007)
  - Initialisation
    - Affection d'un label différent à chaque sommet (un label différent par sommet)
  - Itérations
    - Trier les sommets dans un ordre aléatoire
    - Pour chaque sommet x :
      - déterminer le label l maximal pris par ses voisins (random si égalité)
    - Arrêt : stabilité sur les labels ou nombre maximal d'itérations
  - Intérêt : méthode rapide (nombre d'arêtes)
  - Limite :
    - résultats non stables
      - sensibilité liées à numérotation aléatoire et égalité
    - Communautés de grande taille

Partitionnement de Graphes (11)

- **Illustration**

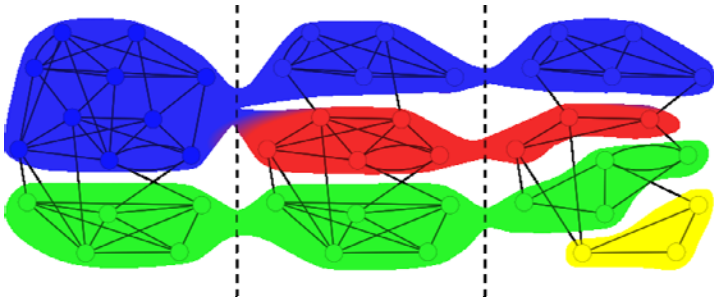


  - Résultats différents …

- Raghavan U. N., Albert R., Kumara S., « Near linear time algorithm to detect community structures in large-scale networks », Physical Review E, vol. 76, no 3, p. 036106, 2007

Partitionnement de Graphes (12)

- **Pour aller plus loin**
  - Fortunato S., « Community detection in graphs », Physics Reports, vol. 486, no 3, p. 75-174, 2010
  - Communautés dans des réseaux dynamiques



  - Communautés avec recouvrement

Plan – section 8

7. Fouille de données
   1. Contexte
   2. Extraction de motifs
   3. Fouille de graphes / réseaux sociaux
8. **Réduction de dimension**

Contexte

- **Passage à l'échelle (scalability)**
  - Capacité à faire face à une forte hausse du volume de données
  - **Approche : réduction du volume**
    - Échantillonnage
    - Réduction du nombre de caractéristiques des données
    - Limites en cas de faible densité en information
      - Difficulté à détecter des régularités
  - **Approche : réduction des calculs**
    - Méthode de calcul locale / voisinage
    - Méthodes approchées et approximation
    - Structure de données, Hachage, Table d'index, …
  - **Approche basée sur les infrastructures**
    - Répartir les traitements et les données (cf cours Map-Reduce)

Impact de la taille des données (1)

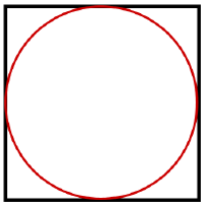
- **Malédiction / Fléau de la dimensionnalité**
  - Un ensemble $X = \{x_i\}$ de n données individus / données
  - Dimensions des données : d attributs / features
    - Exemple : une image avec d pixels

$$x_i = \begin{pmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,d} \end{pmatrix}$$

    - Note :
      - les attributs seront appelés « variables » par conformité avec la littérature
  - **Difficultés** :
    - Visualisation des données : Plus facile en dimension 2 ou 3 ….
    - Cout algorithmique des traitements : Mémoire, calculs, acquisition
    - Apprentissage : plus efficace sur modèle de taille réduite

Impact de la taille des données (2)

- **Malédiction / Fléau de la dimensionnalité**
  - Pour un même nombre de données : la densité diminue avec la dimension
    - Besoin d'augmenter le volume de données avec la dimension
    - Difficulté d'analyses statistique
  - Impact de la distribution des données
    - Les données uniformément distribuées dans des volumes en dimension *d* sont proches des hyper-cubes externes MAIS
      - Ratio volume hyper-sphere et volume hyper cube diminue avec dimension
      - La plupart des données de l'hypercube ne sont pas dans l'hypersphère
      - Difficultés : méthodes basées distances / densité



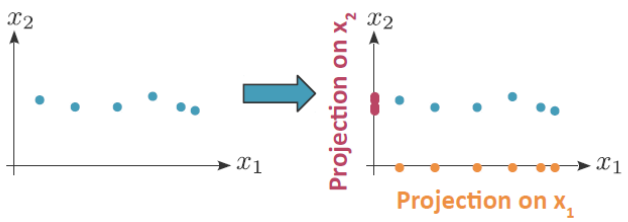
Dimension	Vol. sphère / vol. cube englobant
1	1
2	0,78732
4	0,329707
6	0,141367
8	0,0196735
10	0,00399038

Réduction du volume par réduction de dimension

- **Déterminer de nouvelles « variables »**
  - À partir des « variables initiales »

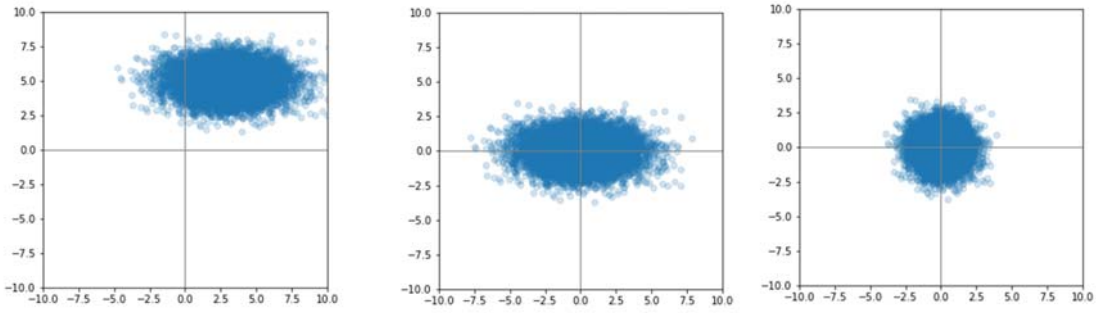
- Méthode « linéaire »
  - Trouver un sous-espace linéaire de dimension $k < d$
  - Les nouvelles variables sont des combinaisons linéaires des variables initiales

  - Projection des données sur ce nouvel espace



Réduction du volume par réduction de dimension

- **Approche : Analyse en Composantes principales (ACP – PCA)**
  - **Objectifs**
    - Réduire le nombre de dimensions pour représenter les données tout en minimisant l'information perdue
  - **Principe**
    - Maximiser la **variance** (inertie) lors de la projection dans le nouvel espace de représentation
    - Standardisation des données: centrer et normaliser



Standardisation des données (1)

- **Variance**
  - Soit x une variable statistique :

Valeur de x	v1	v2	…		vp
Effectif	n1	n2			np

    - Taille population n =somme des n_i, avec n_i : nb d'occurrences de v_i
  - **Variance = carré de l'écart type** (où $\hat{x}$ moyenne de x)
    - $V(x) = \frac{1}{n}\left(n_1(v_1 - \hat{x})^2 + n_2(v_2 - \hat{x})^2 + \ldots n_p\left(v_p - \hat{x}\right)^2\right) = {\sigma_x}^2$
    - Mesure de dispersion autour de la moyenne

      - Ex : [10; 20; 30; 40; 50] (effectif = 1 pour chaque valeur)
        - Variance = 200 (Ecart type = 14,14)
      - Ex : [0,1; 0,2; 0,3; 0,4; 0,5] (effectif = 1 pour chaque valeur)
        - Variance = 0,02 (Ecart type = 0,1414)

Standardisation des données (2)

- **Centrer et normer chaque valeur :**
  - Centrer : la moyenne est égale à 0
  - Normaliser : la variance est égale à 1
  - Ajuster chaque valeur v_i : $v_i \leftarrow \frac{v_i - \hat{x}}{\sigma_x}$
    - Ex [10; 20; 30; 40; 50] ➔ [-1,41; -0,70; 0; 0,70; 1,41]
      - Moyenne $= 0$, Variance $= 1$
    - Ex : [0,1; 0,2; 0,3; 0,4; 0,5] ➔ [-1,41; -0,70; 0; 0,70; 1,41]
      - Moyenne $= 0$, Variance $= 1$

Analyse en Composantes Principales (1)

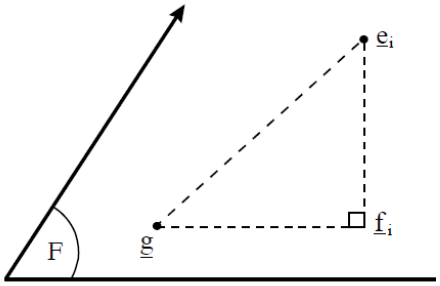
- **Données initiales :**

		attributs					
		1	2		j		d
échantillons	1	x_1,1	x_1,2		x_1,j		x_1,d
	2	x_2,1	x_2,2		x_2,j		x_2,d
	i	x_i,1	x_i,2		x_i,j		x_i,d
	n	x_n,1	x_n,2		x_n,j		x_n,d

- **But ACP :**
  - Déterminer k nouvelles colonnes combinaison linéaire des d colonnes initiales de telle sorte que la perte d'information soit minimale
    - Nouvelles colonnes ➔ composantes principales
    - Nouveaux axes ➔ axes principaux
    - Combinaisons linaires ➔ facteurs principaux

Analyse en Composantes Principales (2)

- Projection des points dans un sous espace



Soit F un sous-ensemble de $\mathbf{R}^p$

$\underline{f}_i$ la projection orthogonale de $\underline{e}_i$ sur F

$$\|\underline{e}_i - \underline{g}\|^2 = \|\underline{e}_i - \underline{f}_i\|^2 + \|\underline{f}_i - \underline{g}\|^2 \qquad \forall i = 1 \ldots n$$

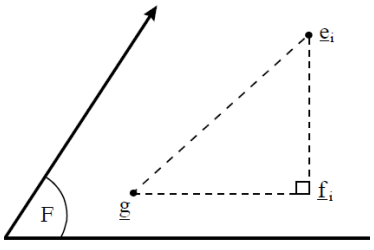
Analyse en Composantes Principales (3)

- Chercher le nouvel espace F tel que

$$\sum_{i=1}^{n} p_i \|\underline{e}_i - \underline{f}_i\|^2 \text{ soit minimal}$$

- Ce qui revient à maximiser

$$\sum_{i=1}^{n} p_i \|\underline{f}_i - \underline{g}\|^2$$



- C'est à dire la variance (inertie)

$$\underbrace{\sum_{i=1}^{n} p_i \|\underline{e}_i - \underline{g}\|^2}_{\text{Inertie totale}} - \underbrace{\sum_{i=1}^{n} p_i \|\underline{e}_i - \underline{f}_i\|^2}_{\text{minimiser cette quantité (carrés des distances entre points individus et leurs projections)}} = \underbrace{\sum_{i=1}^{n} p_i \|\underline{f}_i - \underline{g}\|^2}_{\text{maximiser l'inertie du nuage projeté}}$$

$\Leftrightarrow$

Analyse en Composantes Principales (4)

- **Projection :**
  - La composante j fournit les coordonnées des données dans le nouveau repère sur le $j$ème axe

$$\underline{c}^j = \begin{pmatrix} c_1^j \\ c_2^j \\ \vdots \\ c_n^j \end{pmatrix}$$

  - Chaque composante fournit une valeur de la variance (inertie)

1ère composante	$\mathbf{c^1}$	variance : λ_1
2ème composante	$\mathbf{c^2}$	variance : λ_2
3ème composante	$\mathbf{c^3}$	variance : λ_3

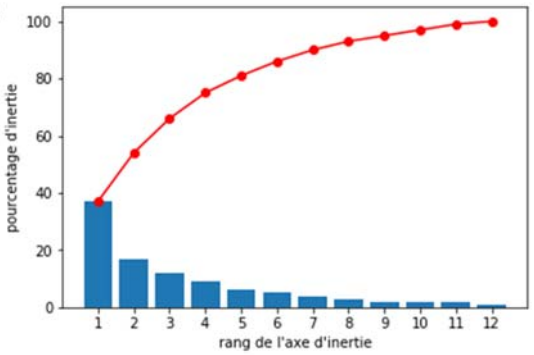
  - Les axes d'inertie sont orthogonaux :
    - Les composantes principales ne sont pas deux à deux corrélées

Analyse en Composantes Principales (5)

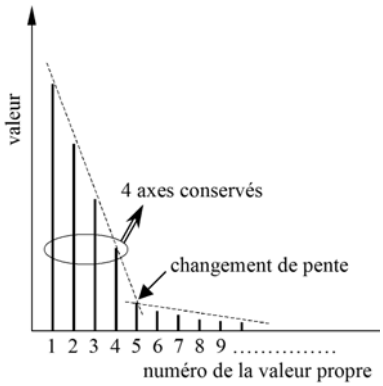
- **Nombre d'axes**
  - Pour représenter des données de dimension d :
    - d axes orthogonaux au maximum
  - Pour représenter un ensemble de n données :
    - Espace à $n-1$ dimensions pour passer par tous les points
  - Bilan : $\min(d, n-1)$
    - Réduire le nombre d'axes intéressants

Analyse en Composantes Principales (6)

- **Choix du nombre de composantes**
  - Variance totale expliquée
    - $\lambda_1 + \lambda_2 + \ldots + \lambda_p$

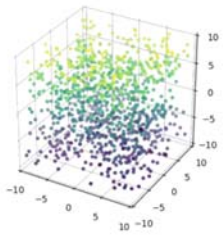


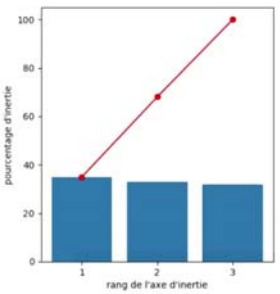
    - Ratio obtenu pour chaque composante :
      - $\frac{\lambda_i}{\sum_{k=1}^{p} \lambda_k}$
    - Ratio cumulé sur 2 composante :
      - $\frac{\lambda_1+\lambda_2}{\sum_{k=1}^{p} \lambda_k}$



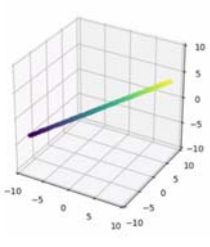
Analyse en Composantes Principales (7)

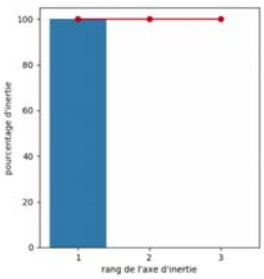
- **Cas limites**
  - Pas de structure dans les données
    - Les variables sont totalement non corrélées
    - Inertie répartie sur chaque axe





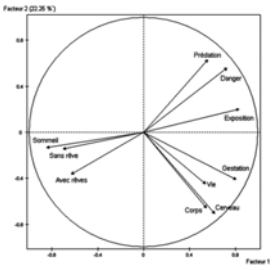
  - Données liées
    - Les variables sont 2 à 2 corrélées
    - Tous les points sont alignés ➔ axe





Réduction de dimensions (1)

- **ACP**
  - Algorithme calculant toutes les composantes
  - Algorithme incrémental
- **Interprétation des résultats**
  - Analyse des liens entre données : ➔ projection du nuage des individus
  - Analyse des liens entre les variables : ➔ projection du nuages des variables
  - Etudier les axes d'inertie des données revient à étudier les axes d'inertie des variables



- **Nombreuses autres méthodes**
  - Linéaires et non linéaires
  - Voir cours d'analyse de données

Réduction de dimensions (2)

- **Pour aller plus loin**
  - Algèbre linéaire /Statistiques / Traitement du signal
- Résultats sur challenge Netflix
  - Prédire des notes pour des films
  - Vulgarisation : science4all – série IA – épisode 20 (réduction de dimensions)
    - https://www.youtube.com/watch?v=Z2kqh--pItQ&index=20&list=PLtzmb84AoqRTl0m1b82gVLcGU38miqdrC&ab_channel=Science4All

Conclusion (1)

- **Un aperçu :**
  - Fouille de données
  - Découvertes de motifs
  - Réseaux sociaux et extraction de communautés
- **Beaucoup d'autres problèmes**
  - **Découverte de motifs** : text mining, sequential frequent itemset, ….
  - **Réseaux sociaux** : extraction de communautés, identification de rôles, diffusion d'information, recherche d'information, réseaux dynamiques
  - **Systèmes de recommandation** : Informations ciblées (filtrage) : centrée sur les objets, les utilisateurs ou l'environnement (réseau) social
  - **Détection de changements / d'anomalies** : phénomènes imprévus mais ayant du sens

Conclusion (2)

- **Et beaucoup d'autres axes de travail**
  - **Réduction des couts de calcul**
    - Complexité, algorithmique, structures de données, optimisation de code, …
    - Algorithmiques parallèles, algorithmes distribués, …
  - **Architectures de calcul et de stockage**
    - Calcul parallèle, calcul distribué, ….
  - **Couplage fouille de données et optimisation combinatoire**
    - Nombreux problèmes d'optimisation combinatoire en fouille de données
    - Méthodes exactes / méthodes approchées
    - Approximation
  - **Couplage fouille de données et protection de la vie privée**
    - Méthodes de cryptographie, méthodes de calcul distribué, …