# Fouille de Données et Media Sociaux
# Cours 2
# Master DAC Data Science
# UPMC - LIP6

Ludovic Denoyer

21 septembre 2015

# Contexte

## Observation

La plupart des bonnes solutions à un problème prédictif/analytique provient de la qualité des caractéristiques fournies au systèmes.

## Problèmes

**Inutilité** : Certaines caractéristiques sont inutiles

**Malédiction des grandes dimensions** : Le nombre d'exemples nécessaires pour antrainer un système croit exponnetiellement ebn fonction de la taille de l'espace d'entrée

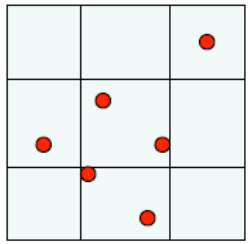
**Autres problèmes** : coût des caractéristiques, type de caractéristiques, ...

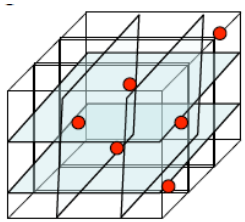
**Problématique :** Comment découvrir de bonnes caractéristiques

# Malédiction des grandes dimensions

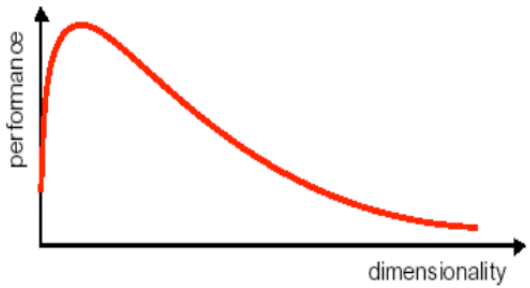
## Intuition

Un même nombre d'exemples prend **moins de place** quand la dimensionnalité du problème est **petite**.





**Concrètement :** Au dessus d'une certaine dimension, la performance (en généralisation) de notre classifier va avoir tendance à décroitre fortement
$\Rightarrow$ perte de la capacité à généraliser



# Plan du Cours

**Problématique :** Comment trouver des caractéristiques pertinentes ?

## Sélection de caractéristiques :

Sélectionner un sous-ensemble des caractéristiques existantes :

- Approches de type **Filtering**
- Approches de type **Wrappers**

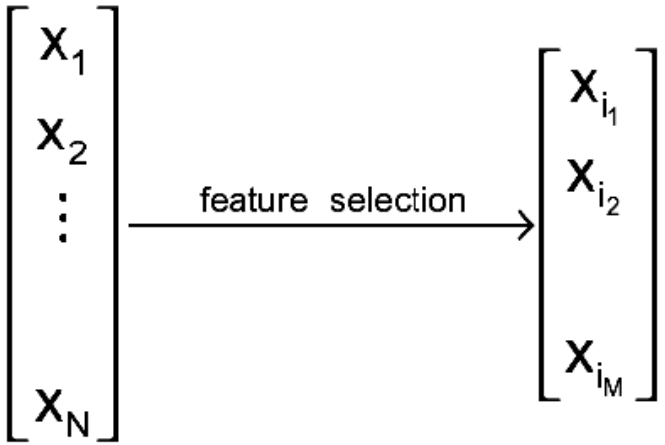
## Extraction de caractéristiques :

Combiner des caractéristiques existantes pour obtenir un (petit nombre) de caractéristiques pertinentes :

- Approches de type **PCA**
- Approches de type **Auto-Encodage**
- Approches de type **Representation Learning (Deep Learning)**

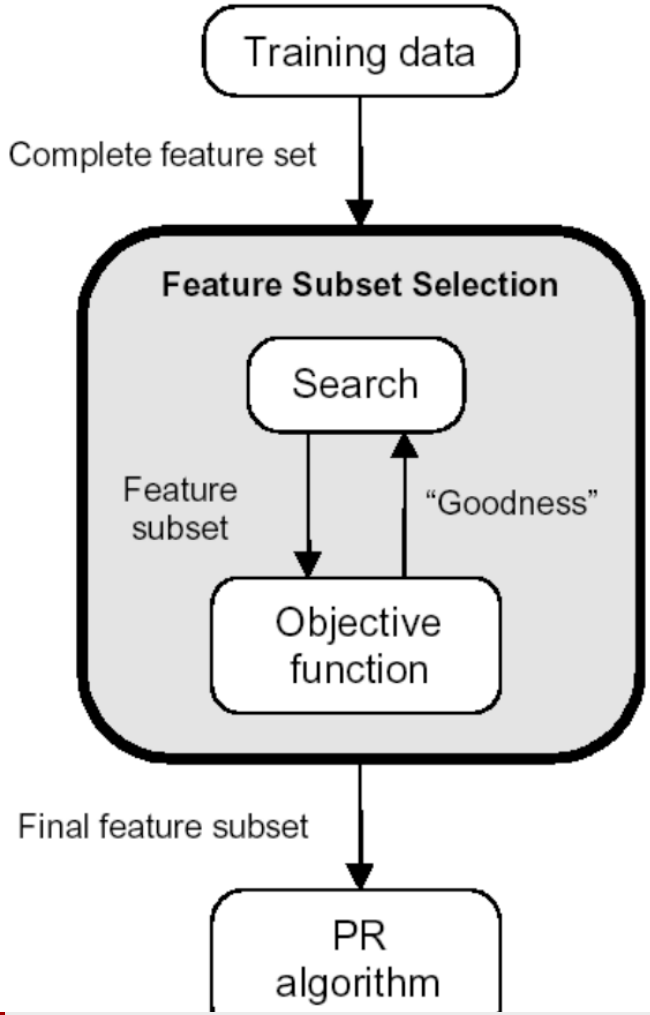
# Features Selection

- Soit un ensemble d'entrée $\mathcal{X} = \mathbb{R}^n$ tel que $x = (x_1, x_2, ..., x_n)$
- On cherche à trouver un sous-ensemble de dimensions caractérisé par un ensemble $\mathcal{I}$ d'index dans $[1; n]$
- Etant donné $\mathcal{I} = (i_1, ...i_M)$, le nouvel espace d'entrée sera caractérisé par $x = (x_{i_1}, x_{i_2}, ..., x_{i_M})$



# Problème

La sélection de caractéristique est un **problème de recherche** discret :
très grand espace de recherche $\Rightarrow$ développement de stratégies approchées



# Deux stratégies

## Méthodes de filtrage

Les méthodes de filtrage considèrent que la sélection des caractéristiques s'effectue *a priori* de l'apprentissage

- On ne garde que les caractéristiques dont le "pouvoir prédictif" est élevé
- Ce "pouvoir prédictif" est estimé par l'étude mono-dimensionnelle de la caractéristique

## Méthodes de wrappers

Les méthodes **Wrappers** sélectionnent les caractéristiques directement sur la qualité du modèle obtenu

# Méthodes de Filtrage

On considère l'espace de sortie $\mathcal{Y}$.

## Principe

Pour chaque caractéristique $i \in [1; n]$, on va calculer un score $R(i)$ et ne conserver que les caractéristiques dont le score est le plus élevé. **Quel score utiliser ?**

- Intuitivement, $R(i)$ doit nous donner une idée de la capacité de la caractéristique $i$ a aider la prédiction (i.e $\in \mathcal{Y}$)

Plusieurs critères :

- Théorie de l'information
- Corrélation
- Test statistiques

# Corrélation

## Corrélation

En probabilités et en statistiques, étudier la corrélation entre deux ou plusieurs variables aléatoires ou statistiques numériques, c'est étudier l'intensité de la liaison qui peut exister entre ces variables.source : Wikipedia, comme d'habitude

Différents types de liaisons peuvent exister entre variables. La plus simple est la liaison **linéaire** $\Rightarrow$ corrélation linéaire

## Corrélation linéaire

Soit la variable $X_i$ (caractéristique) et la variable $Y$ (étiquette) :

$$Corr(X, Y) = \frac{Cov(X_i, Y)}{\sqrt{Var(X_i)Var(Y)}} \quad (1)$$

avec $Cov(X_i, Y) = E[X_i Y] - E[X_i]E[Y] = E\left[(X_i - E[X_i])(Y - E[Y])\right]$,
$Cov(X_i, Y) = 0$ ssi $X_i$ et $Y$ sont indépendantes

# Corrélation

## Estimation

Les lois de distributions de $X_i$ et $Y$ sont inconnues, il faut dont les **estimer** à partir de l'ensemble d'apprentissage :

$$R(i) = \frac{\sum_{k=1}^{N}(x_i^k - \bar{x_i})(y^k - \bar{y})}{\sqrt{\sum_{k=1}^{N}(x_i^k - \bar{x_i})^2 \sum_{k=1}^{N}(y^k - \bar{y})^2}} \quad (2)$$

**Conclusion :** Mesure la dépendance linéaire entre la caractéristique et la sortie. D'autres mesures "non-linéaire" peuvent être utilisées....

# Information Mutuelle

## Information Mutuelle

Dans la théorie des probabilités et la théorie de l'information, l'information mutuelle de deux variables aléatoires est une quantité mesurant la dépendance statistique de ces variables. Elle se mesure souvent en bit.source : Wikipedia

$$I(i) = \int_{x_i} \int_{y} p(x_i, y) \log \frac{p(x_i, y)}{p(x_i)p(y)} dx_i dy \tag{3}$$

$I(i)$ mesure ici la dépendance entre la distribution de $X_i$ et la ditrribution de la sortie $Y$

## Cas des variables discrètes

$$I(i) = \sum_{x_i} \sum_{y} P(x_i, y) \log \frac{P(x_i, y)}{P(x_i)P(y)} \tag{4}$$

- $P(y)$ est la probabilité *a priori* de la classe
- $P(x_i)$ est la probabilité *a priori* de la modalité

# Méthodes de filtrage

| Method | | X | | | Y | | | Comments |
|---|---|---|---|---|---|---|---|---|
| Name | Formula | B | M | C | B | M | C | |
| Bayesian accuracy | Eq. 3.1 | + | s | | + | s | | Theoretically the golden standard, rescaled Bayesian relevance Eq. 3.2. |
| Balanced accuracy | Eq. 3.4 | + | s | | + | s | | Average of sensitivity and specificity; used for unbalanced dataset, same as AUC for binary targets. |
| Bi-normal separation | Eq. 3.5 | + | s | | + | s | | Used in information retrieval. |
| F-measure | Eq. 3.7 | + | s | | + | s | | Harmonic of recall and precision, popular in information retrieval. |
| Odds ratio | Eq. 3.6 | + | s | | + | s | | Popular in information retrieval. |
| Means separation | Eq. 3.10 | + | i | + | + | | | Based on two class means, related to Fisher's criterion. |
| T-statistics | Eq. 3.11 | + | i | + | + | | | Based also on the means separation. |
| Pearson correlation | Eq. 3.9 | + | i | + | + | i | + | Linear correlation, significance test Eq. 3.12, or a permutation test. |
| Group correlation | Eq. 3.13 | + | i | + | + | i | + | Pearson's coefficient for subset of features. |
| $\chi^2$ | Eq. 3.8 | + | s | | + | s | | Results depend on the number of samples $m$. |
| Relief | Eq. 3.15 | + | s | + | + | s | + | Family of methods, the formula is for a simplified version ReliefX, captures local correlations and feature interactions. |
| Separability Split Value | Eq. 3.41 | + | s | + | + | s | | Decision tree index. |
| Kolmogorov distance | Eq. 3.16 | + | s | + | + | s | + | Difference between joint and product probabilities. |
| Bayesian measure | Eq. 3.16 | + | s | + | + | s | + | Same as Vajda entropy Eq. 3.23 and Gini Eq. 3.39. |
| Kullback-Leibler divergence | Eq. 3.20 | + | s | + | + | s | + | Equivalent to mutual information. |
| Jeffreys-Matusita distance | Eq. 3.22 | + | s | + | + | s | + | Rarely used but worth trying. |
| Value Difference Metric | Eq. 3.22 | + | s | | + | s | | Used for symbolic data in similarity-based methods, and symbolic feature-feature correlations. |
| Mutual Information | Eq. 3.29 | + | s | + | + | s | + | Equivalent to information gain Eq. 3.30. |
| Information Gain Ratio | Eq. 3.32 | + | s | + | + | s | + | Information gain divided by feature entropy, stable evaluation. |
| Symmetrical Uncertainty | Eq. 3.35 | + | s | + | + | s | + | Low bias for multivalued features. |
| J-measure | Eq. 3.36 | + | s | + | + | s | + | Measures information provided by a logical rule. |
| Weight of evidence | Eq. 3.37 | + | s | + | + | s | + | So far rarely used. |
| MDL | Eq. 3.38 | + | s | | + | s | | Low bias for multivalued features. |

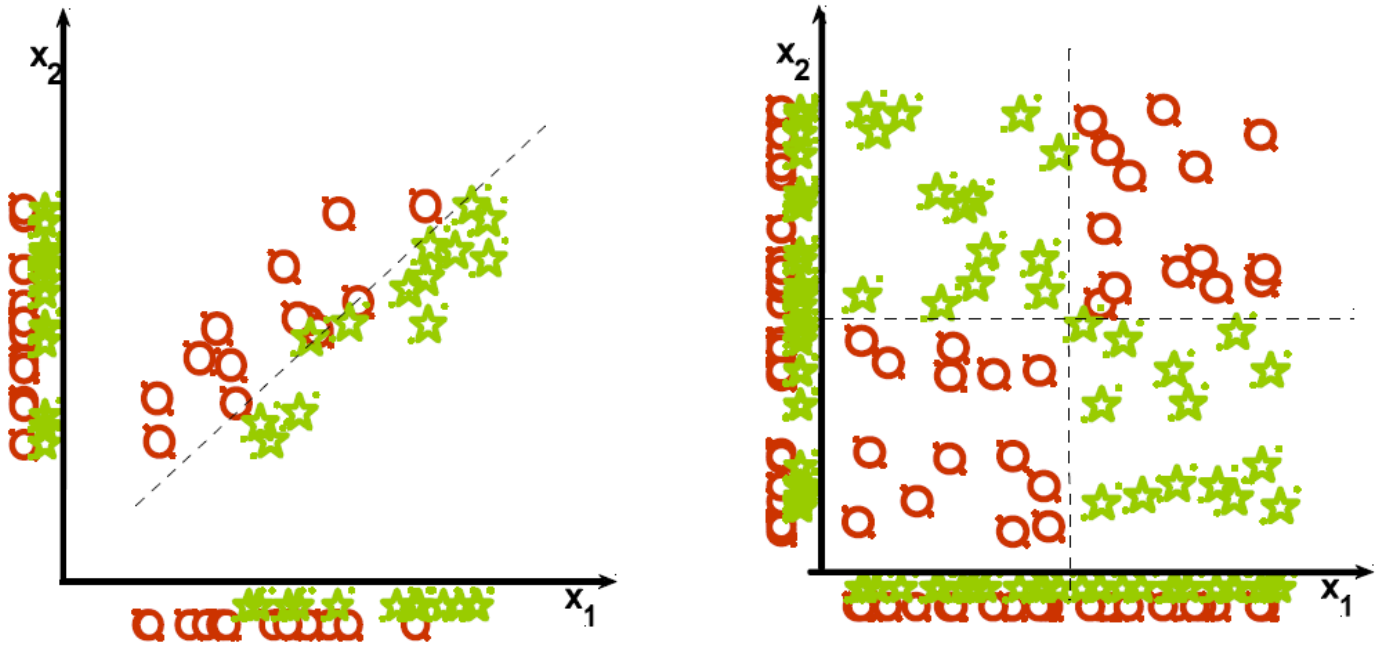
Voir livre de I. Guyon - Chapitre 3

# Méthodes de Filtrage

- Les méthodes de filtrage visent à ordonner les variables par ordre de pertinence, et à ne conserver que les meilleures.
- Elles traitent les caractéristiques **indépendamment les unes des autres**.
- Ce sont des méthodes rapides de sélection de variables.

Quelles en sont les limites ?

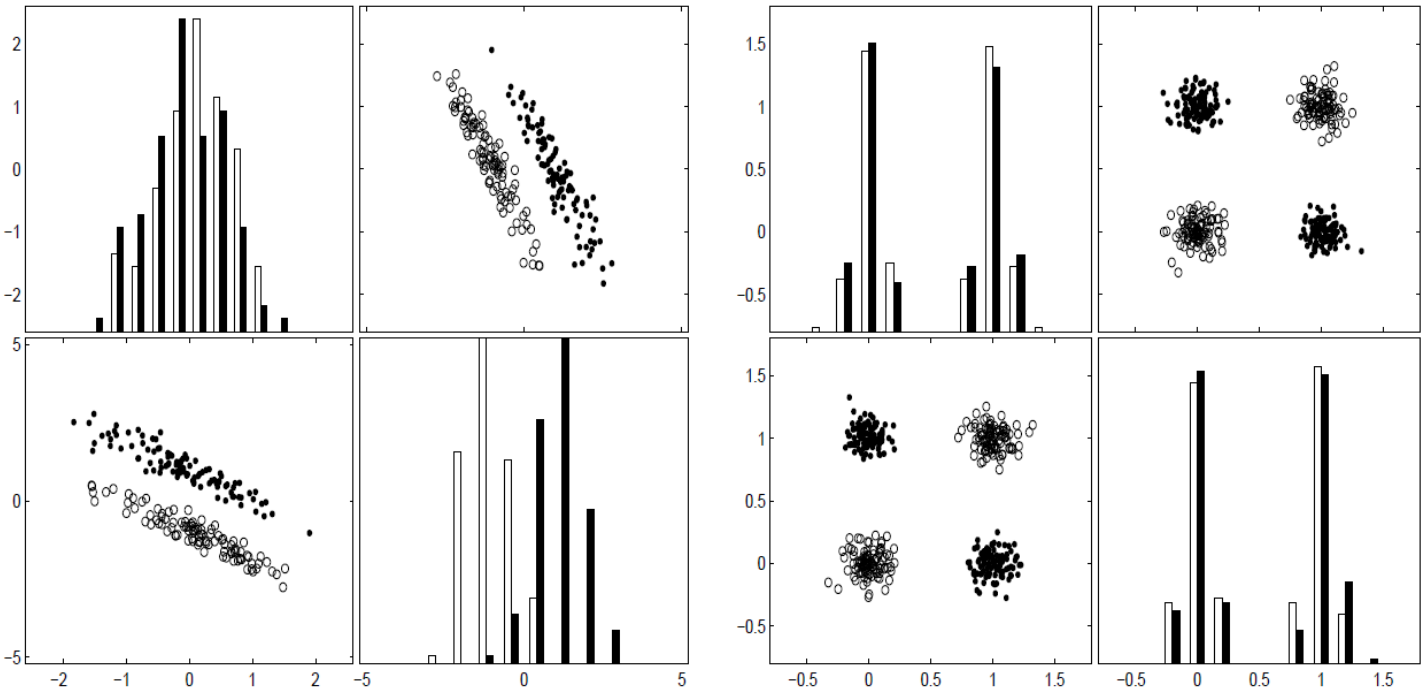
# Limites




*Guyon-Elisseeff, JMLR 2004; Springer 2006*

# Limites

**Est-ce qu'une variable qui n'est pas utile "toute seule" peut être utile quand même ?**



# Méthodes de Wrappers

## Principe

La sélection des variables est directement faites en fonction de la performance du prédicteur utilisé

## Solution Naive

On peut tester tous les sous-ensembles de features.

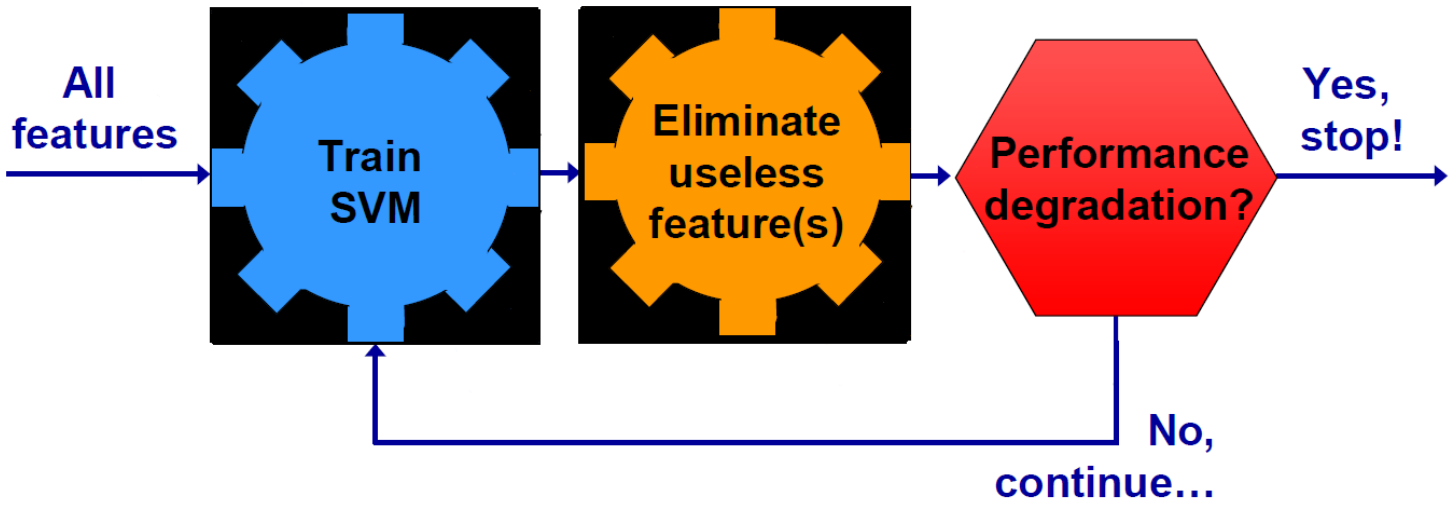
- Problème NP-complet ! !

Différents algorithmes de recherche : glouton, beam-search, ...

## Exemple : Recherche gloutonne

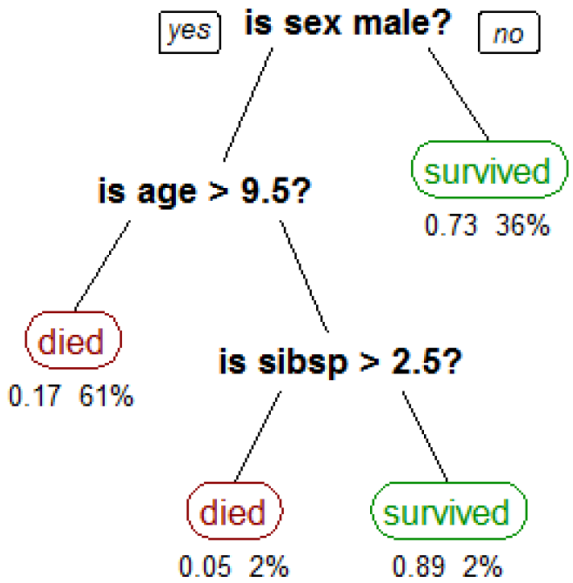
- Ajout graduel de caractéristiques basé sur un score à chaque pas de l'algorithme
- Ce score doit refléter la performance du système (en généralisation ! ) $\Rightarrow$ cross-validation

# Méthodes embarquées




Recursive Feature Elimination (RFE) SVM. *Guyon-Weston, 2000. US patent 7,117,188*

# Méthodes d'arbres (MORACOI / BI)

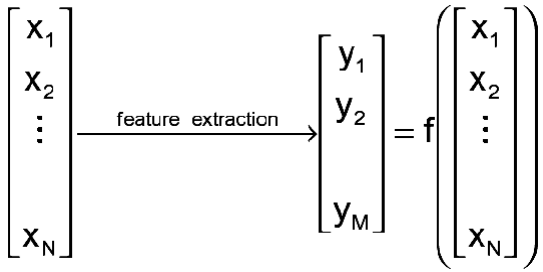



$$H_S(C|A) = -\sum_i P(v_i) \sum_k P(c_k|v_i) \log(P(c_k|v_i))$$

# Construction de features et réduction de dimensionnalité

## Features Extraction

Le but de l'extraction de features est de construire un nouveau sous-ensemble de caractéristiques de taille plus faible que l'ensemble original, les nouvelles cractéristiques permettant d'obtenir une performance meilleure



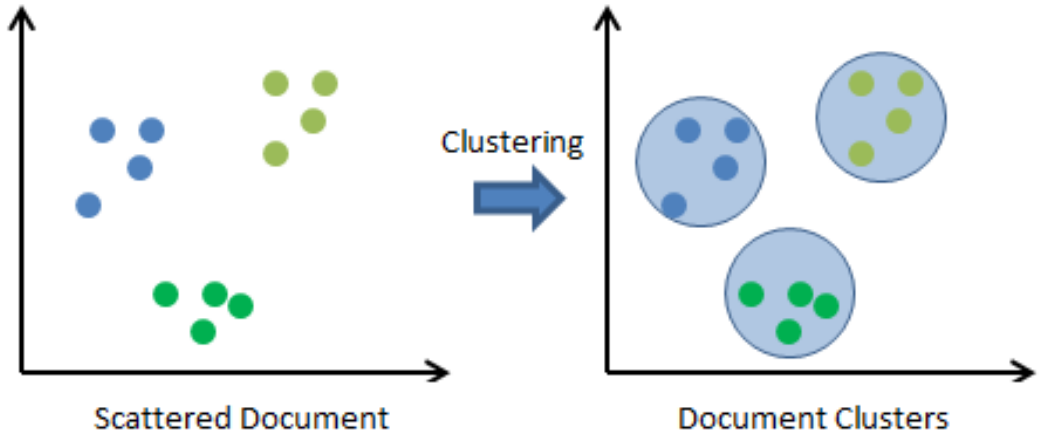

Différentes méthodes :

- Non-supervisé - Clustering, PCA, SVD, ...
- Supervisé - Neural Networks

# Clustering

## Clustering

Une méthode de clustering est une méthode permettant de regrouper des données par groupes de données "similaires".




Dans le cadre de la sélection de caractéristiques :

- Le clustering est effectué sur les caractéristiques
- Le cluster remplace dans l'espace de représentation l'ensemble des caractéristiques qu'il contient

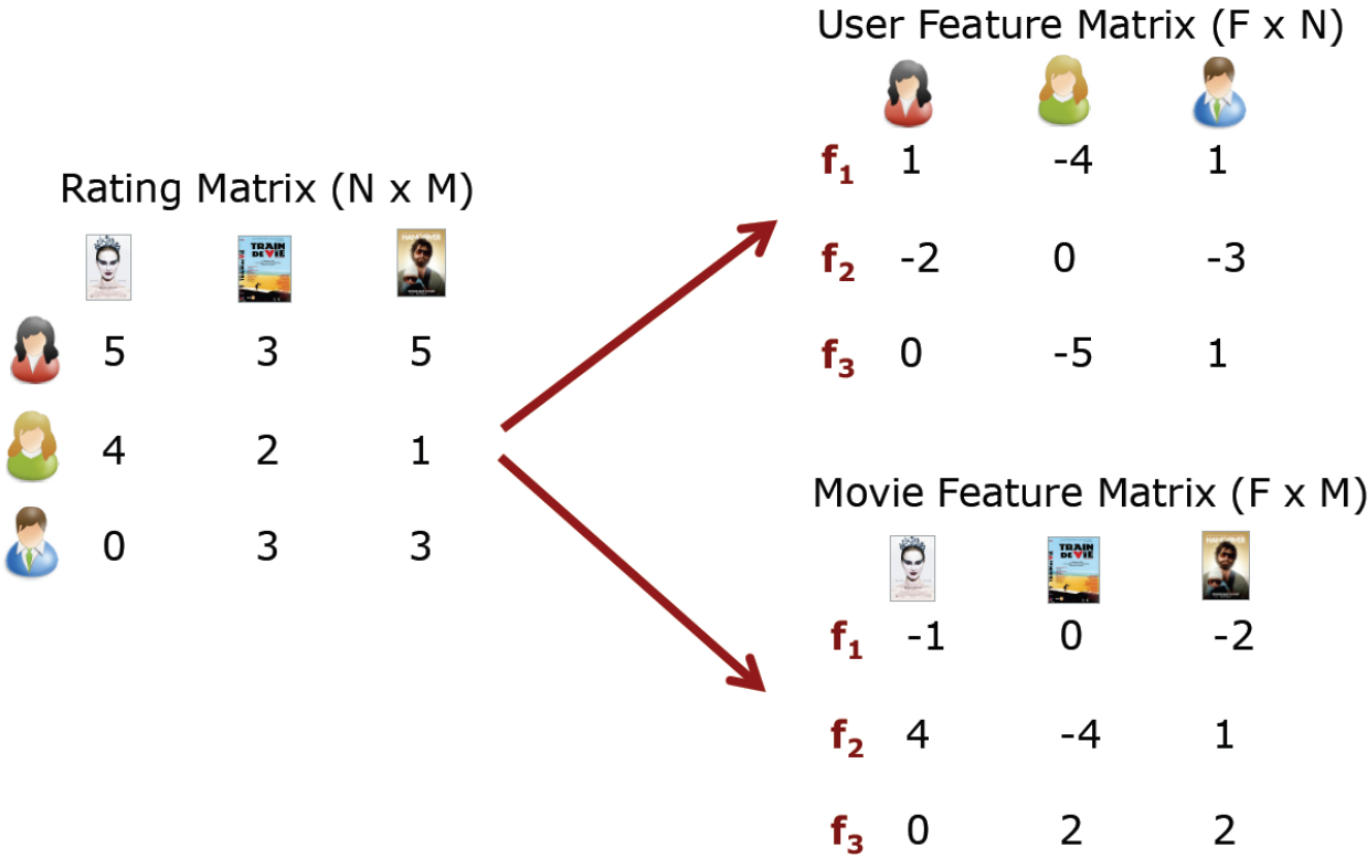
# SVD et Factorisation Matricielle

## Singular Value Decomposition

Le but de la SVD est de former un ensemble de features par combinaison linéaire de features existantes. Le critère de création de ces features est un critère de de reconstruction de données (au sens des moindres carrés)

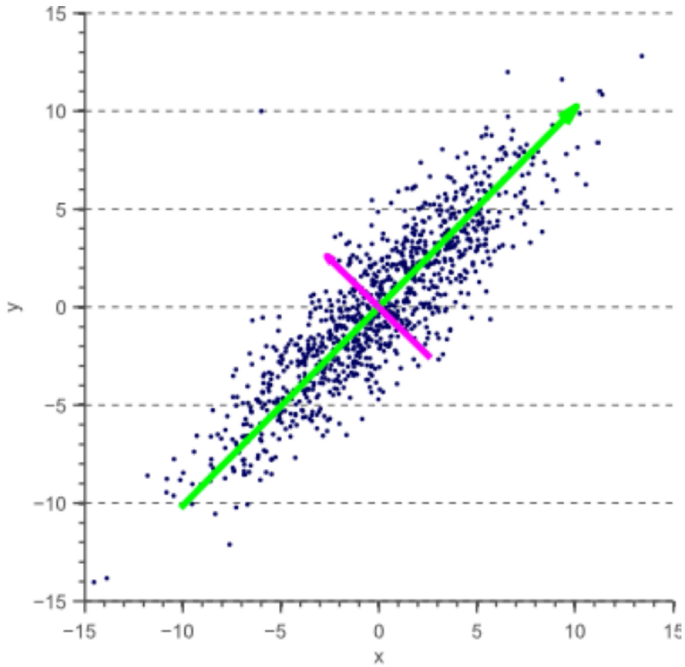
$$X = P \times Q^T \Rightarrow min||X - P \times Q^T||^2 \tag{5}$$

# SVD et Factorisation Matricielle



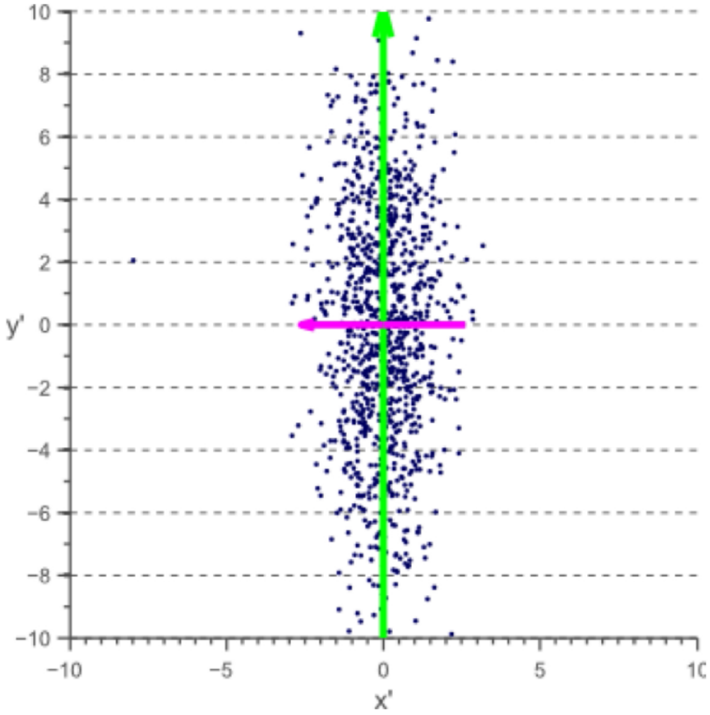
# PCA

- La PCA est une méthode permettant de trouver des variables décorrélées dans nos données




$$Cov(X_1, X_2) = \begin{pmatrix} 16.87 & 14.94 \\ 14.94 & 17.27 \end{pmatrix} \tag{6}$$

# PCA

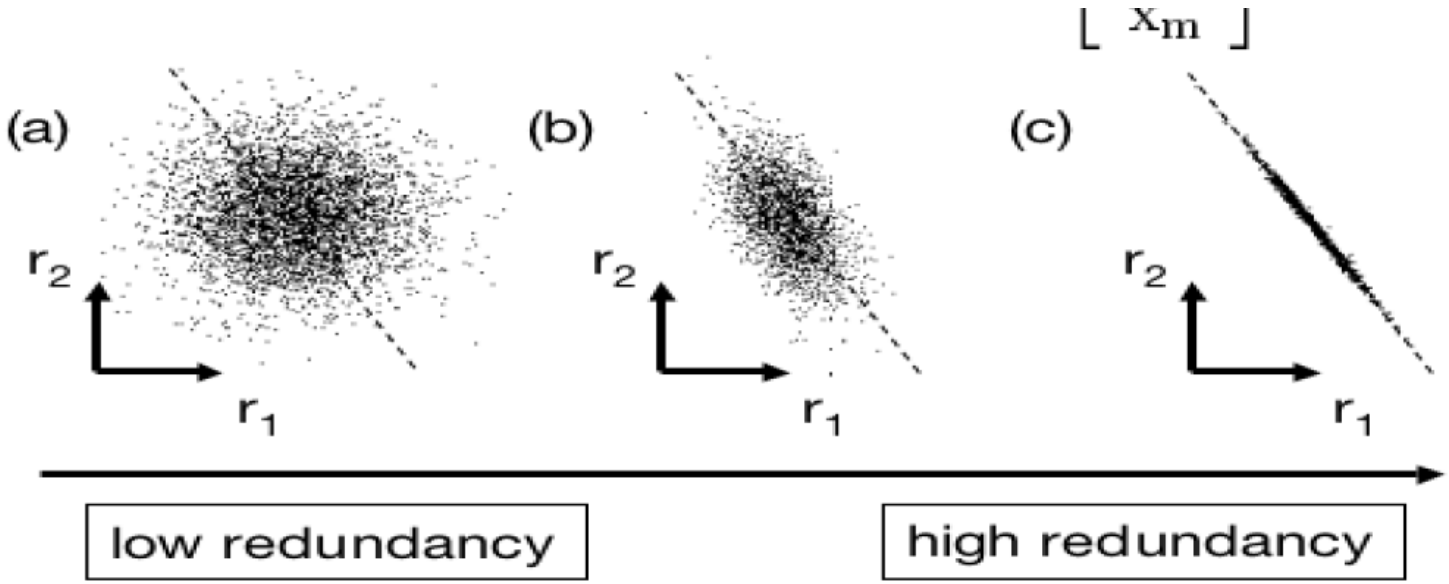



$$Cov(X'_1, X'_2) = \begin{pmatrix} 1.06 & 0 \\ 0 & 16.0 \end{pmatrix} \quad (7)$$

La transformation est obtenue par un changement de base : $X' = PX$

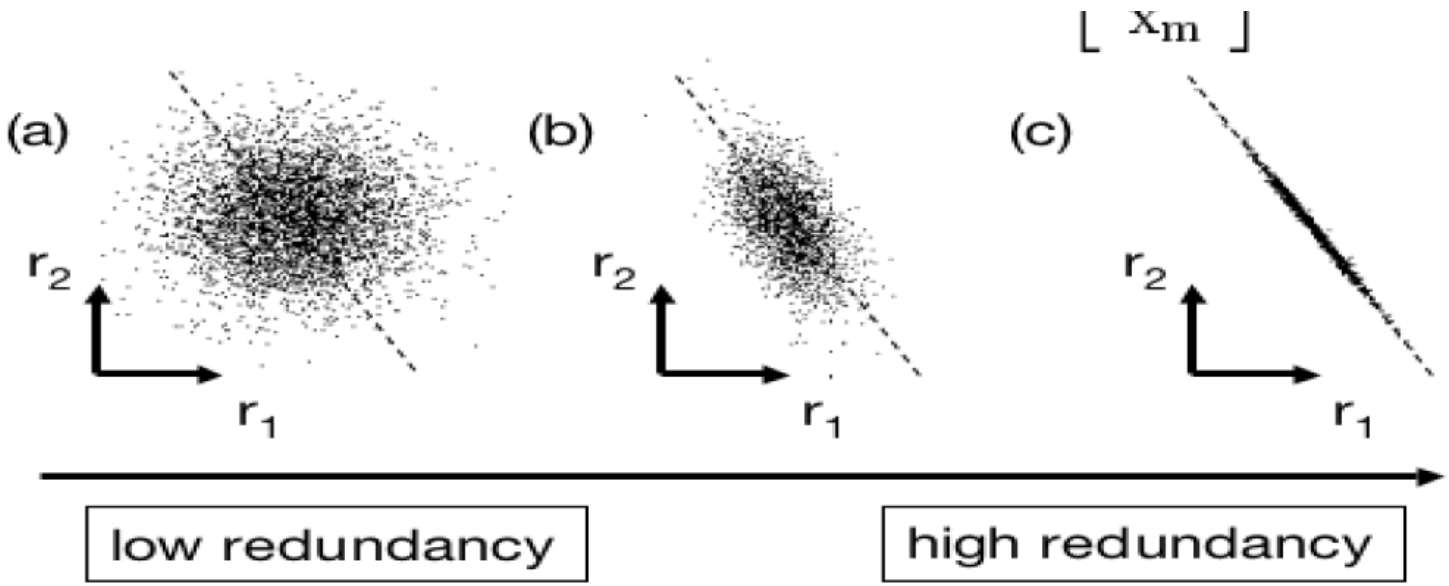
- Les lignes de $P$ sont les nouveaux vecteurs bases des données
- Sur quel critère trouver $P$ et de quelle manière ?

# PCA




Quelles sont les critères permettant de caractériser la redondance ?

# PCA




Considérons que $X$ est une matrice $m \times n$ ($m$ variables, $n$ individus) dont la variance des variables est 0 (données centrées). La matrice de variance/covariance s'écrit :

$$S_X = \frac{1}{n-1} X X^T \tag{8}$$

- $S_X$ est une matrice carrée $m \times m$

# PCA

Le but de l'ACP est de trouver la transformation $P$ telle que :

- Minimise la redondance (les termes hors diagonals doivent être proches de 0)
- Maximise le signal (les termes diagonals doivent être élevés)

$\Rightarrow$ obtention d'une matrice diagonale $S_Y = S_{PX}$.

- L'ACP considère que les vecteurs base de la matrice $P$ sont orthogonaux ($\langle p_i . p_j \rangle = 0$)
- De plus, les directions avec la variance max sont les directions les plus importantes (principales)

# Algorithme PCA

1. Sélectionner la direction dans laquelle la variance est maximisée
2. Trouver la second direction sous contrainte qu'elle soit orthogonale à la première
3. Continuer jusqu'à avoir sélectionné $m$ directions
4. Les variances associées aux directions résument l'importance de chaque direction

Il existe une solution analytique à cette algorithme.

# PCA

$$
\begin{aligned}
S_Y &= \frac{1}{n-1} YY^T \\
&= \frac{1}{n-1} (PX)(PX)^T \\
&= \frac{1}{n-1} (PX)X^T P^T \\
&= \frac{1}{n-1} P(XX^T)P^T \\
&= \frac{1}{n-1} PAP^T
\end{aligned}
\tag{9}
$$

avec $A$ qui est une matrice symétrique et peut être écrite en fonction de ses vecteurs propres : $A = EDE^T$ avec $D$ diagonale.

# PCA

Si l'on sélectionne $P = E^T$ alors :

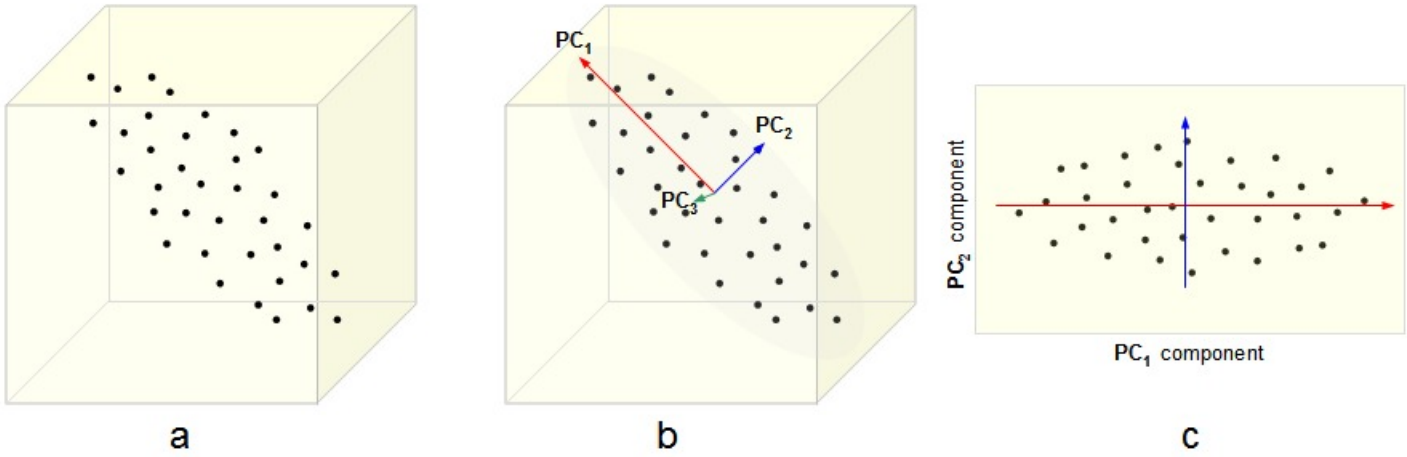
$$\begin{aligned} S_Y &= \frac{1}{n-1} PAP^T \\ &= \frac{1}{n-1} P(P^T DP)P^T \\ &= \frac{1}{n-1} (PP^T)D(PP^T) \\ &= \frac{1}{n-1} (PP^{-1})D(PP^{-1}) \text{ car P est orthonormale et } P^T = P^{-1} \\ &= \frac{1}{n-1} D : \text{CQFD} \end{aligned} \tag{10}$$

Il faut donc trouver les vecteurs propres de $XX^T$

# PCA : Workflow

- Centrer les données en soustrayant la moyenne de chaque variable
- Calculer la matrice de covariance
- Calculer les vecteurs propres de cette matrice
- Projeter les données dans le nouvel espace
- Ne conserver que les nouvelle caractéristique dont les valeurs propres sont les plus importantes

# PCA : Exemple



# Autres méthodes

- Risque régularisé $L_1$
- Réseaux de neurones
- Représentation Learning

# Conclusion - from Guyon 2003

1. **Do you have domain knowledge?** If yes, construct a better set of "ad hoc" features.
2. **Are your features commensurate?** If no, consider normalizing them.
3. **Do you suspect interdependence of features?** If yes, expand your feature set by constructing conjunctive features or products of features, as much as your computer resources allow you (see example of use in Section 4.4).
4. **Do you need to prune the input variables** (e.g. for cost, speed or data understanding reasons)? If no, construct disjunctive features or weighted sums of features (e.g. by clustering or matrix factorization, see Section 5).
5. **Do you need to assess features individually** (e.g. to understand their influence on the system or because their number is so large that you need to do a first filtering)? If yes, use a variable ranking method (Section 2 and Section 7.2); else, do it anyway to get baseline results.
6. **Do you need a predictor?** If no, stop.

# Conclusion - from Guyon 2003

7. **Do you suspect your data is "dirty"** (has a few meaningless input patterns and/or noisy outputs or wrong class labels)? If yes, detect the outlier examples using the top ranking variables obtained in step 5 as representation; check and/or discard them.

8. **Do you know what to try first?** If no, use a linear predictor.[3] Use a forward selection method (Section 4.2) with the "probe" method as a stopping criterion (Section 6) or use the $\ell_0$-norm embedded method (Section 4.3). For comparison, following the ranking of step 5, construct a sequence of predictors of same nature using increasing subsets of features. Can you match or improve performance with a smaller subset? If yes, try a non-linear predictor with that subset.

9. **Do you have new ideas, time, computational resources, and enough examples?** If yes, compare several feature selection methods, including your new idea, correlation coefficients, backward selection and embedded methods (Section 4). Use linear and non-linear predictors. Select the best approach with model selection (Section 6).

10. **Do you want a stable solution** (to improve performance and/or understanding)? If yes, sub-sample your data and redo your analysis for several "bootstraps" (Section 7.1).

# Conclusion

- La sélection de caractéristiques est un point clef de la data science
- Un gros catalogue de méthodes existent
- En TP : une autre famille de méthodes $\rightarrow$ les méthodes $L_1$
- En AS : les réseaux de neurones profonds