REsearch and methodology in Data Science

Cours 1 – Méthodologie du traitement de données

Olivier Schwander <olivier.schwander@lip6.fr>

Master DAC Data Science
UPMC - LIP6

2019-2020

Méthodologie

Les différentes étapes

Quelles sont les différentes étapes effectuées par un système de fouille de données ?

Conception d'un système

Les questions à se poser en premier

- Quel type de données ?
- Quel type de tâche ?
- Quelle quantité de données ?
- Quelle qualité des données ?
- Quels objectifs ?

Ensuite

- Quel prétraitement des données ?
- Quelles méthodes ?
- Comment choisir les paramètres ?
- Comment les évaluer ?
- Comment présenter les résultats ?
- Comment les interpréter ?

Données et tâches

Types de données

- Vectorielles
- Temporelles
- Graphes
- Texte

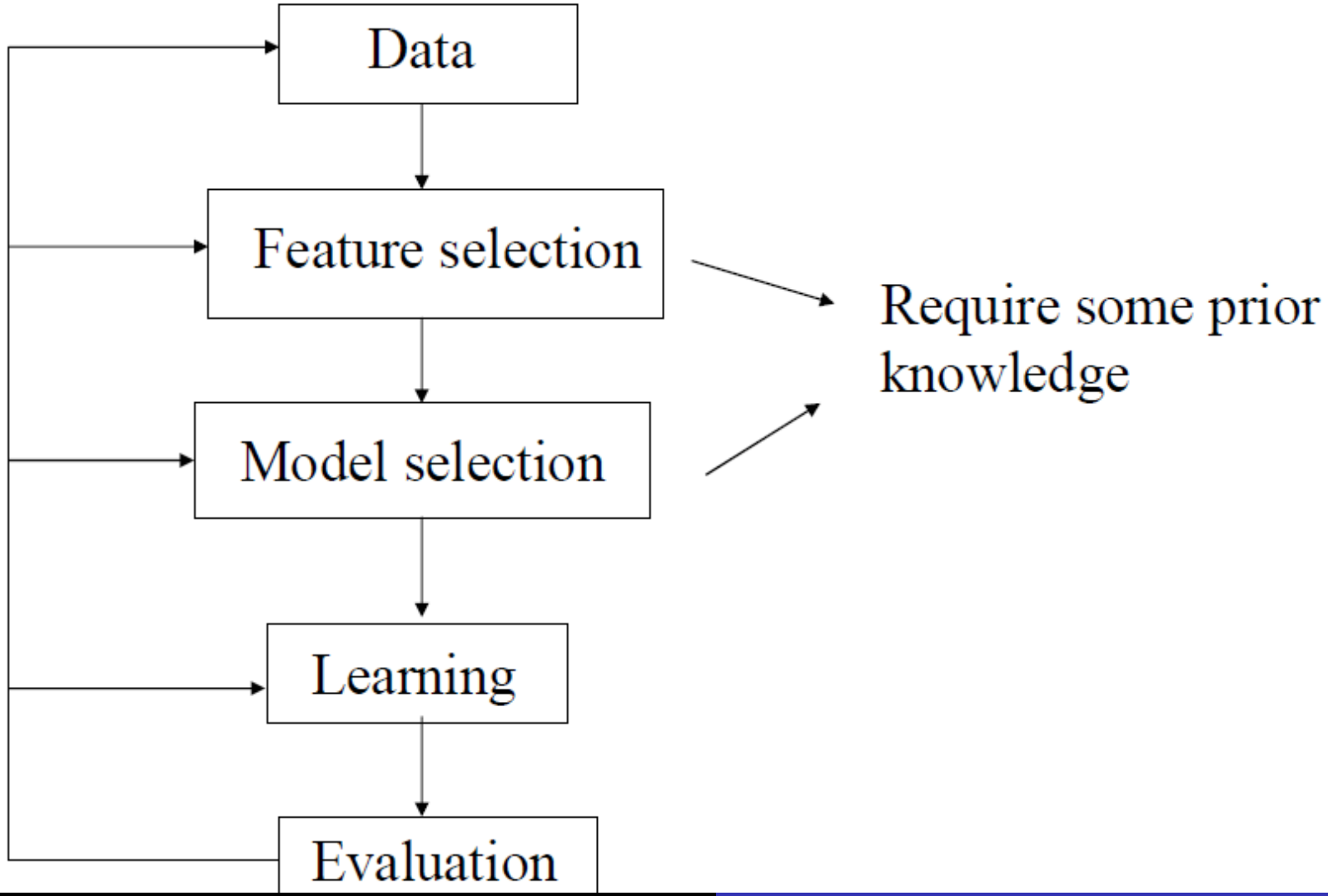
Différentes tâches

- Classification
- Régression
- Détection d'évènements
- Segmentation
- Recherche d'information
- Recommandation

Chaîne de traitement des données

1. **Données**
   - Charger
   - Analyser
   - Transformer
2. **Méthodes**
   - Choisir
   - Paramétrer
   - Apprendre
3. **Évaluation**
   - Mesurer
   - Présenter
   - Interpréter

Concevoir un modèle



Acquisition des données

Capteurs

- Données physiques (erreurs intrinsèques, position du capteur)
- Températures, humidité, pression, etc

Indicateurs

- Calculés d'une façon ou d'un autre
- Rentrés à la mainn

Extract / Transform / Load Voir cours Business Intelligence

Systèmes d'apprentissage

- Vision, Texte, Voix
- pour guider un autre système d'IA

Pré-traitement

- Renommage
- Normalisation
- Discrétisation
- Abstraction
- Aggrégation
- *Sélection d'attributs - Features sélection*
- Création d'attributs

Biais dans les données

- Comprendre la source des données
- Éviter des choix a priori basé sur l'intuition
- Connaissance experte souvent utile

Malédiction de la dimension

- Dimension du problème trop élevée
- Trop de variables
- Trop de paramètres

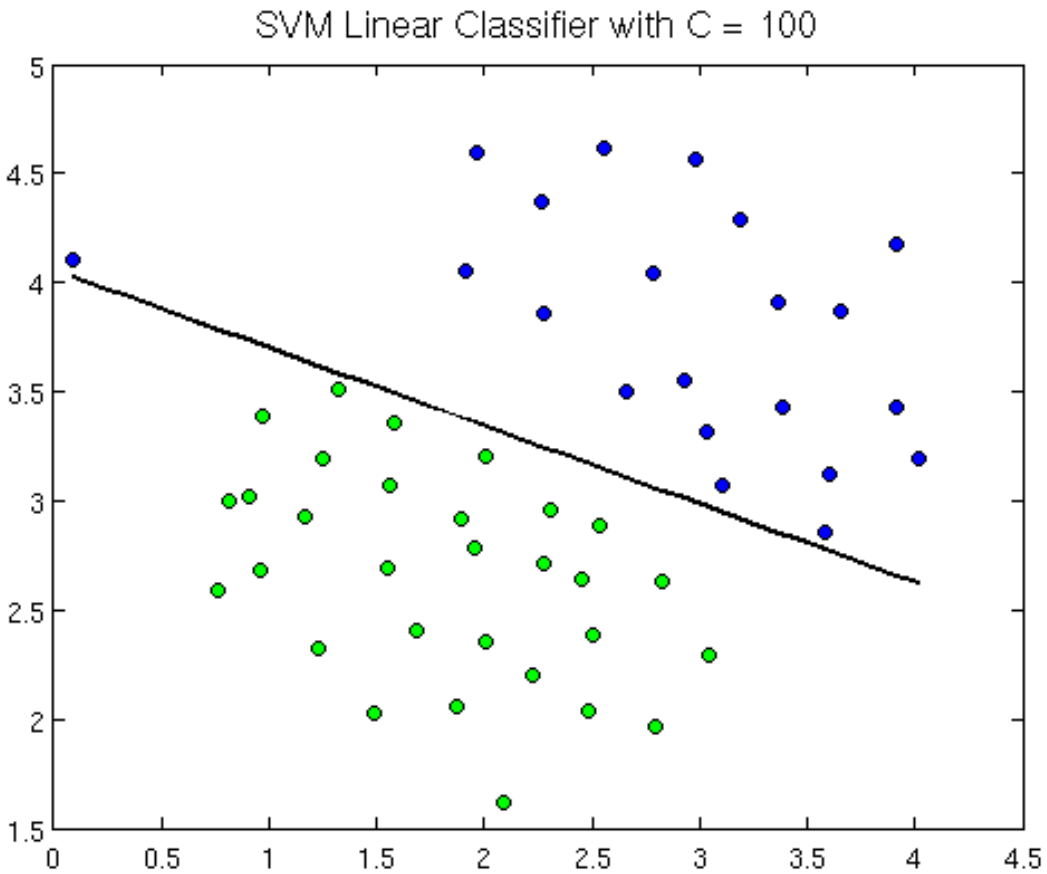
Intuition

- Plus la dimension est grande, plus les points sont isolés
- $\frac{\text{Volume hypersphere}}{\text{Volume hypercube}} \rightarrow 0$

Solutions

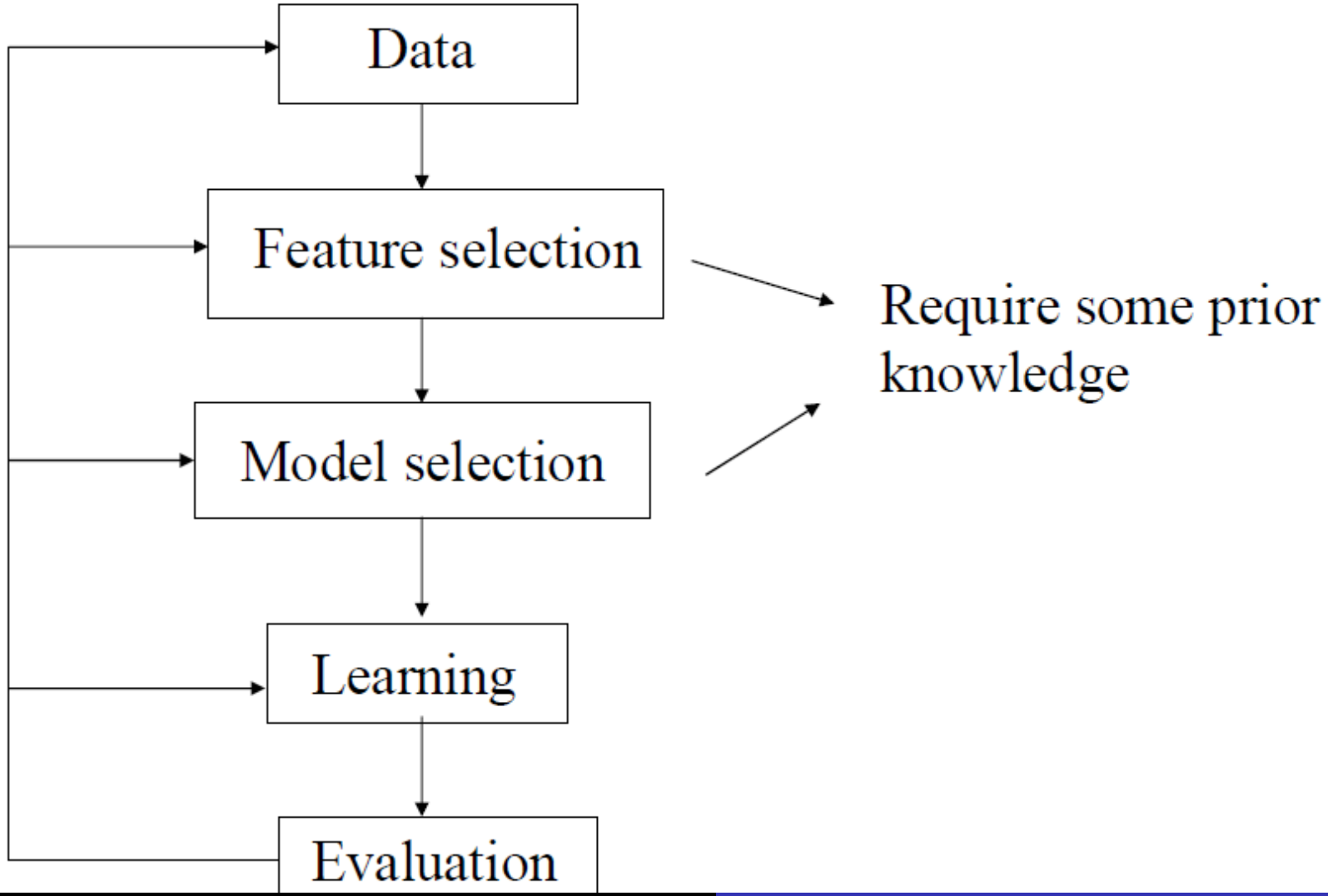
- Réduire la dimension
- Transformation manuelle (expert)
- Apprendre la transformation

Outliers



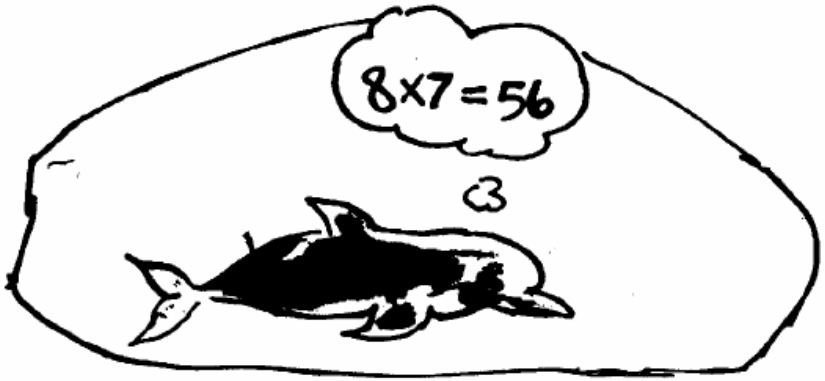
- Il faut supprimer les outliers...
- ...mais ça n'est pas simple

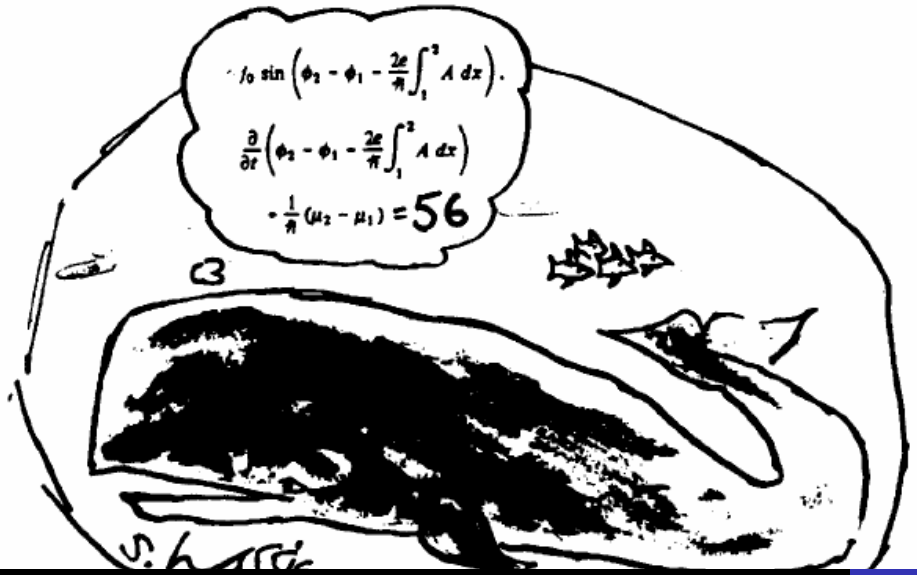
Concevoir un modèle



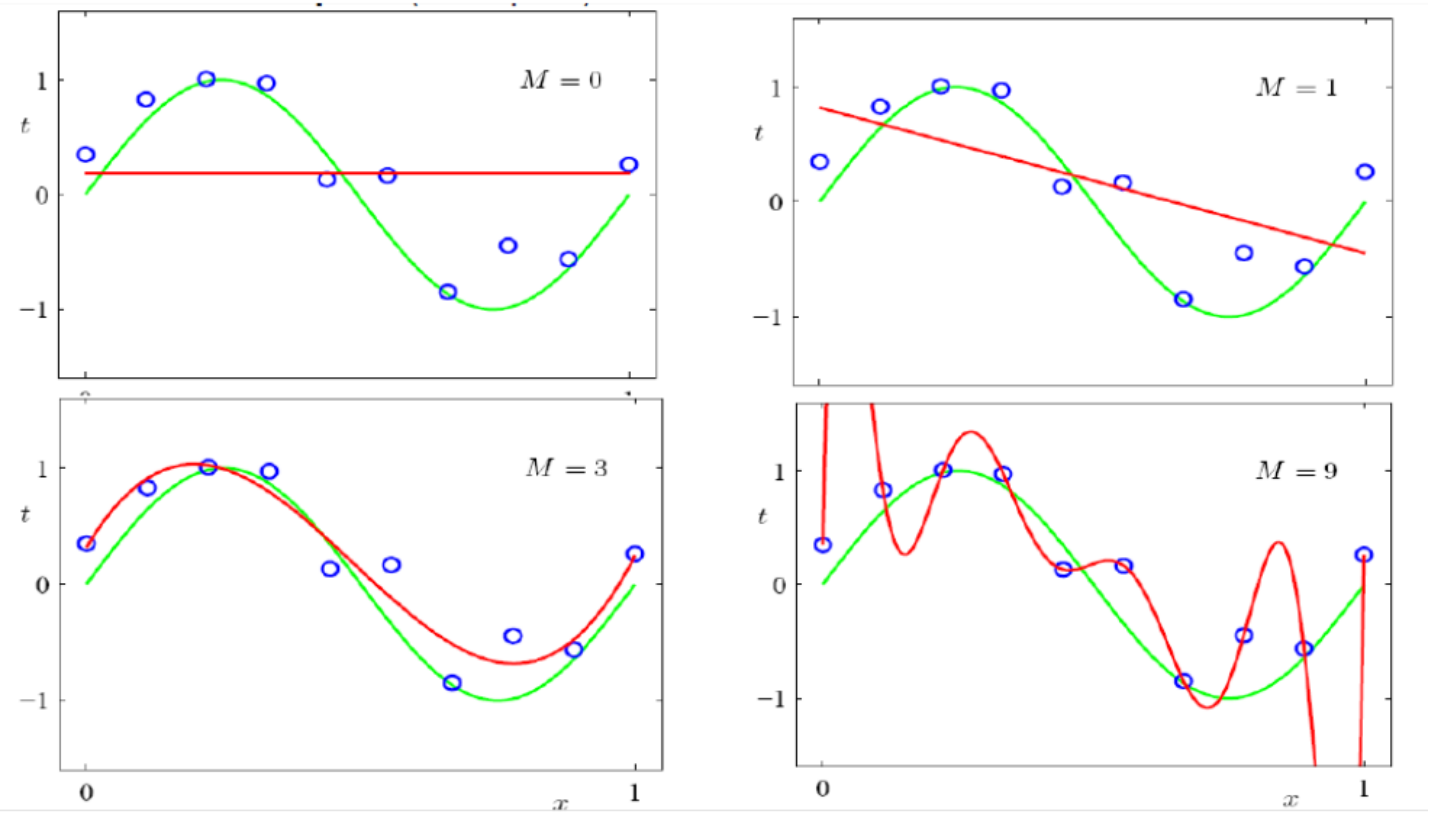
Sélection de modèle

Sélection de modèle



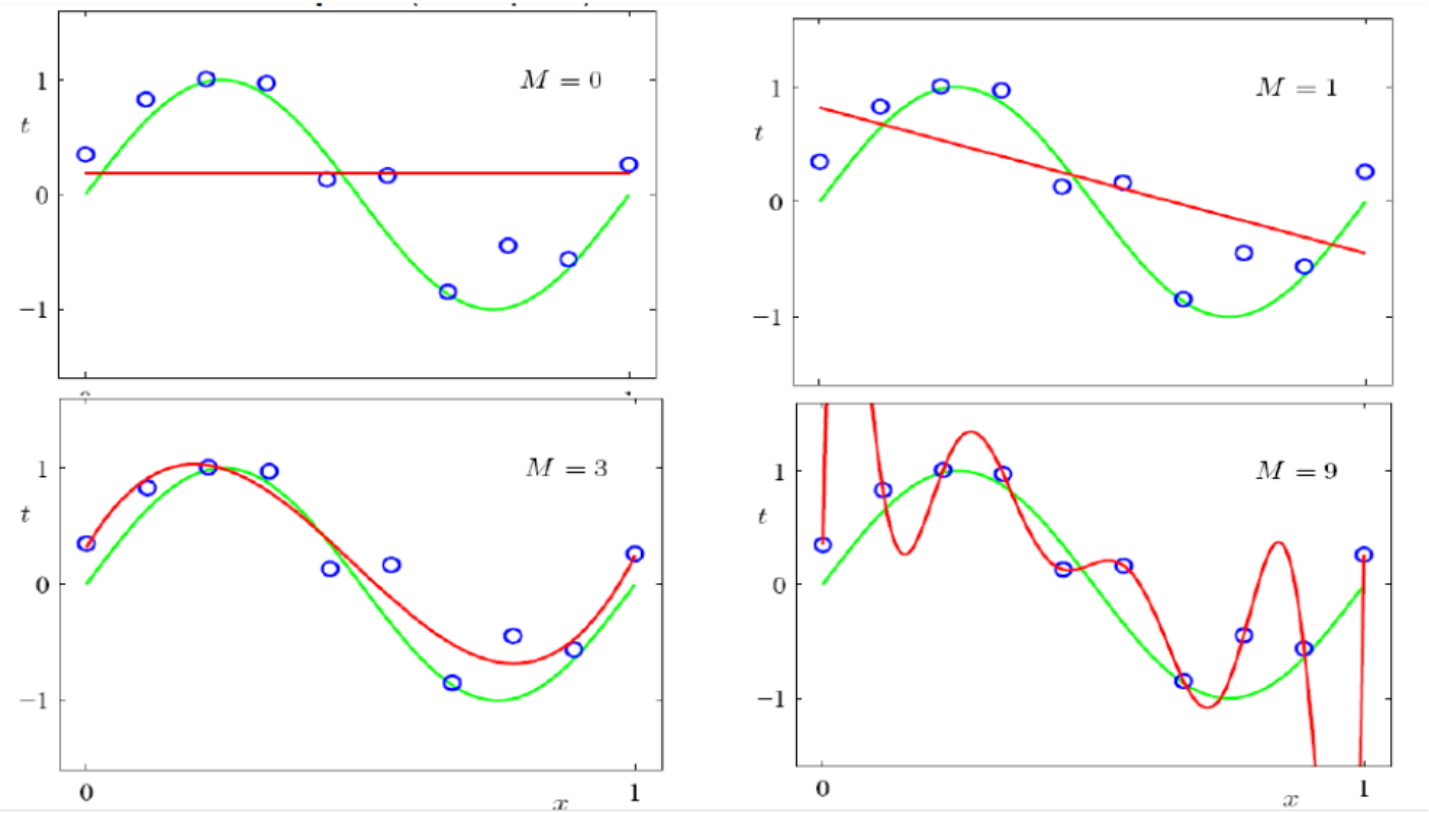


Sélection de modèle



Quel est le meilleur modèle ?

Sélection de modèle



Conclusion: On ne doit pas choisir le modèle qui correspond le mieux aux données, mais celui qui **généralise** le mieux

Sélection de modèle

On cherche des moyens de sélectionner le "meilleur" modèle parmi un ensemble de modèles possibles

Bruit et Régularités **Données** = **Bruit** + **Régularités**

- Bruit: Erreurs dans l'acquisition
- Régularités: Processus de génération sous jacent

Objectif: **Modèle final** = **Capture du bruit** + **Modèle des régularités**

Meilleur modèle:

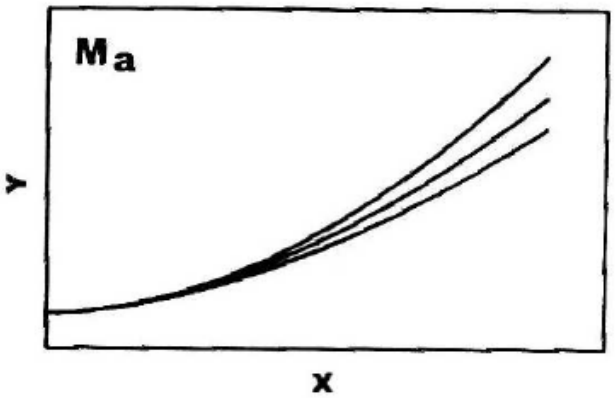
- Meilleur modèle des régularité
- Meilleure capture du bruit

Sur-apprentissage / Overfitting

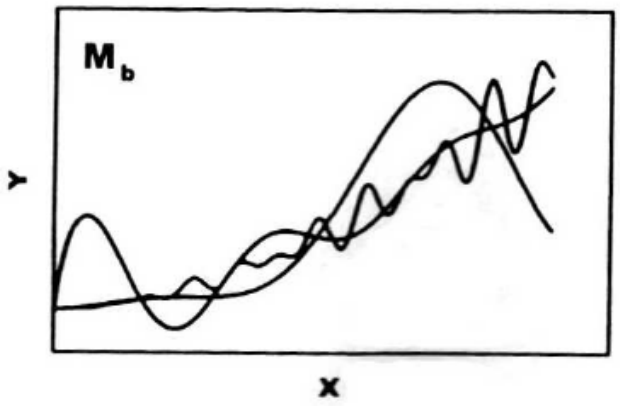
Sur-apprentissage

Quand est-ce qu'un modèle sur-apprend ?

Simple Model



Complex Model

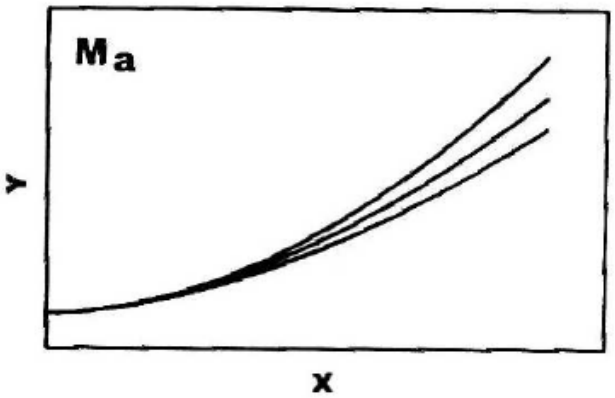


La complexité d'un modèle est liée au nombre de ses paramètres, et à la complexité sous-jacente de la classe de fonction choisie.

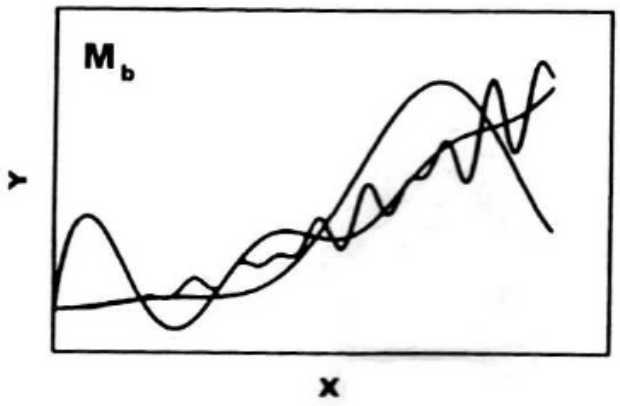
Sur-apprentissage

Quand est-ce qu'un modèle sur-apprend ?

Simple Model



Complex Model



La complexité d'un modèle est liée au nombre de ses paramètres, et à la complexité sous-jacente de la classe de fonction choisie.

Critère d'information d'Akaike - 1973

$$AIC = -2 \ln \hat{L} + 2k$$

- $\hat{L}$ est la vraisemblance du modèle sur les données $= P(x|\theta^*, f)$
- k est le nombre de paramètres du modèle

Méthodologie

- Entraîner plusieurs modèles
- Calculer leur AIC
- Prendre le modèle avec le meilleur AIC (le plus faible)

Critère d'information d'Akaike - 1973

Divergence de Kullback-Leibler (KL)

- On suppose que les données sont générées par un processus p
- Soit des modèles f_i
- $KL(p||f_i)$ mesure l'information perdue en approchant p par f_i
- Le meilleur modèle est celui qui minimise cette divergence
- **Problème:** on ne connait pas p

Estimateur **asymptotique**

- l'AIC permet de comparer des modèles

Variante pour petits jeux de données:

- $AICc = AIC + \frac{2k(k+1)}{n-k-1}$

Autres critères

- Critère d'information Bayésien - 1978: $BIC = -2 \ln \hat{L} + k \ln n$
- Minumum Description Length - 1978: *learning as data compression*

Principe général à retenir: rasoir d'Occam

- *Pluralitas non est ponenda sine necessitate*
- *Les multiples ne doivent pas être utilisés sans nécessité*
- Sélectionner le modèle le plus simple qui modélise les données *suffisamment* bien

Sélection de modèles par échantillonage

Deux grandes familles de méthodes pour se faire une idée de l'erreur de généralisation..

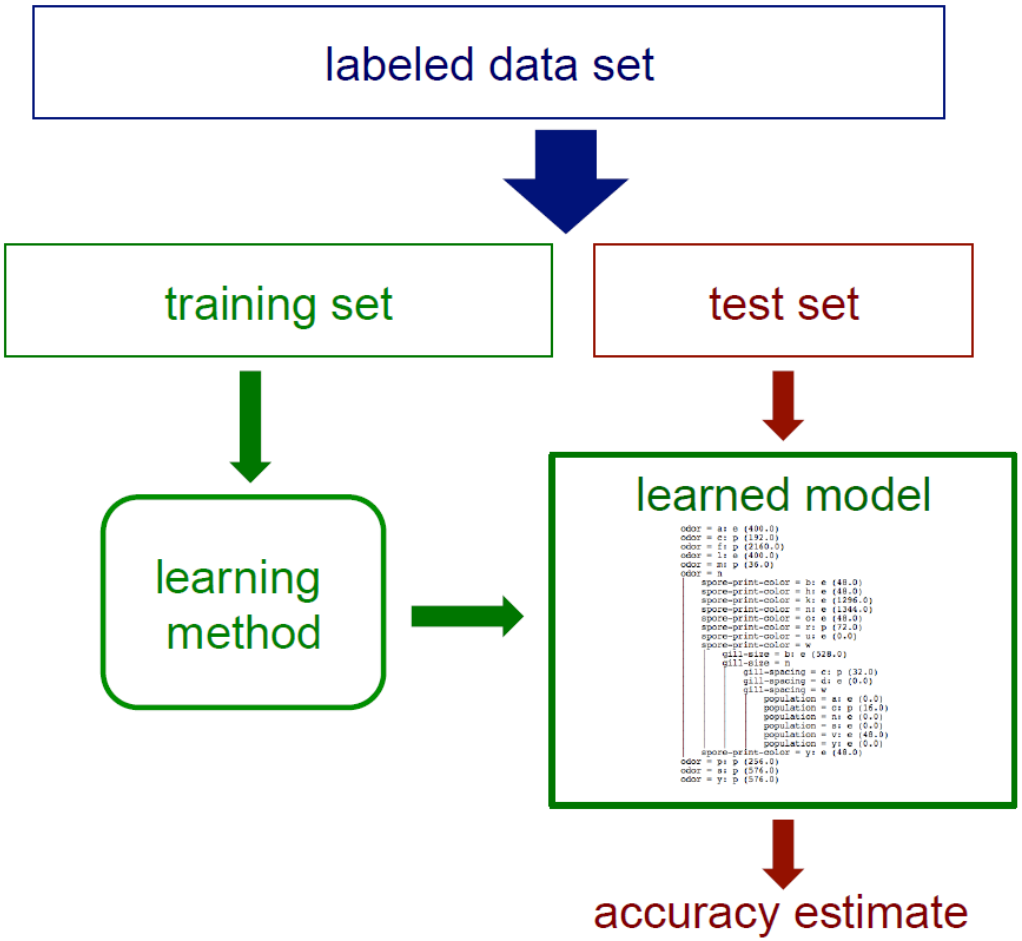
- La loi des grands nombres: l'utilisation de bornes statistiques permettant de borner la différence entre l'erreur empirique et l'erreur théorique (sous certaines hypothèses)

$$\forall f \in \mathcal{F}, \quad \mathcal{R}_P(f) \leqslant \widehat{\mathcal{R}}_n(f) + \frac{1}{\sqrt{2n}} \sqrt{\ln(2) \underbrace{|f|_\pi}_{\text{complexite}} + \ln \frac{1}{\delta}}.$$

- L'utilisation d'échantillons différents pour l'évaluation de l'erreur

Évaluation

Evaluation en Machine Learning



Sélection de modèles par échantillonage

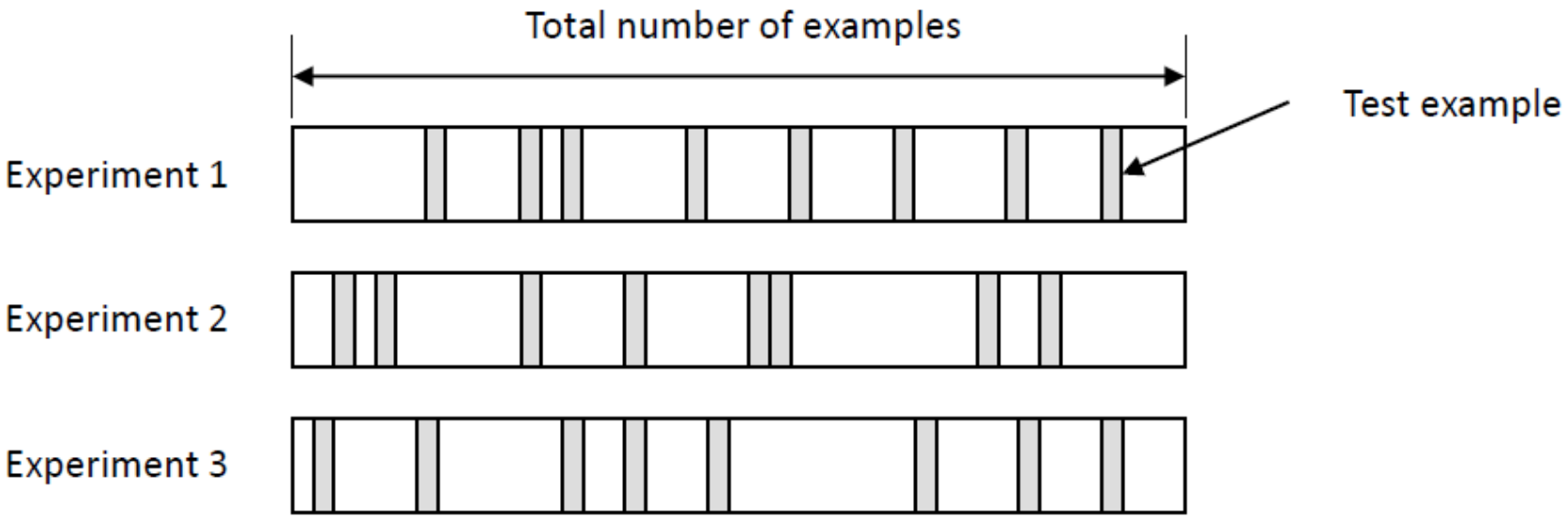
Problèmes

- A-t-on assez de données pour consituter ces différents ensembles ?
- L'utilisation d'un unique ensemble d'apprentissage ne nous permet pas de savoir si le modèle est sensibles aux données d'apprentissage

Plusieurs solutions:

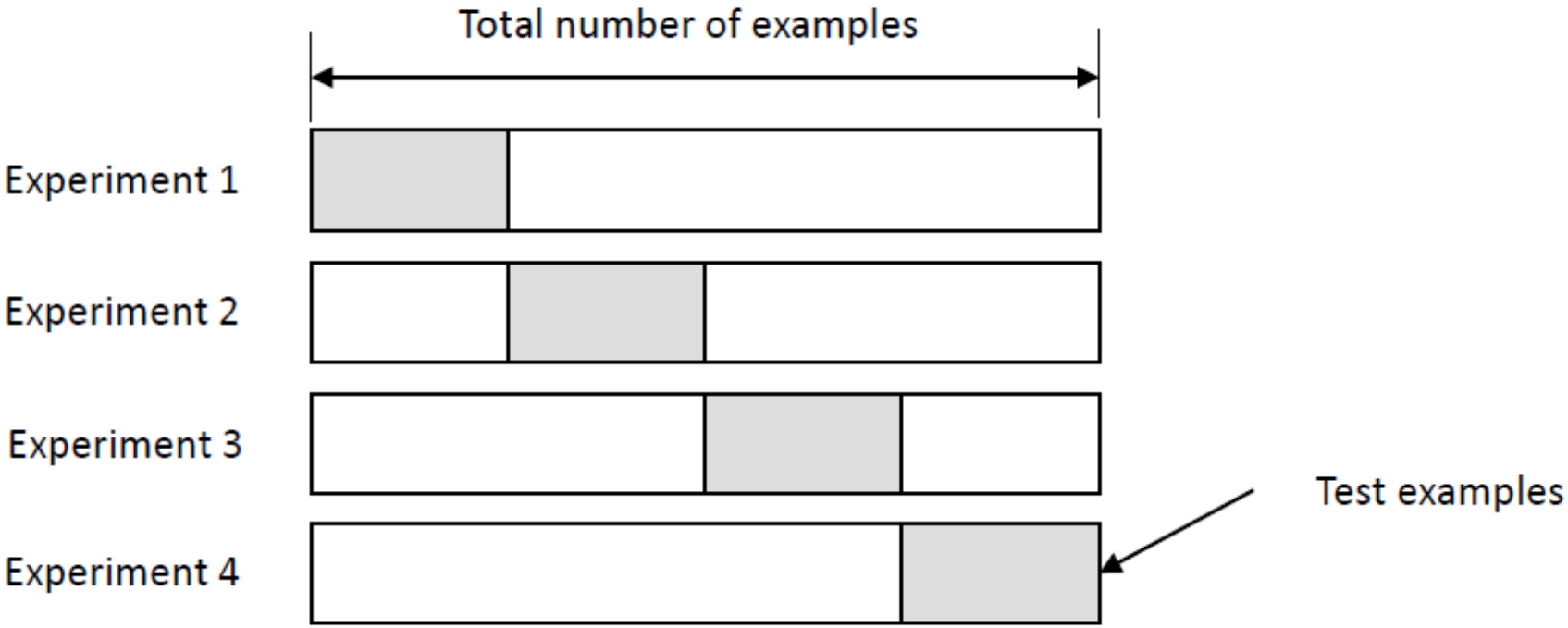
- Rééchantillonage aléatoire
- Cross-Validation

Rééchantillonage aléatoire



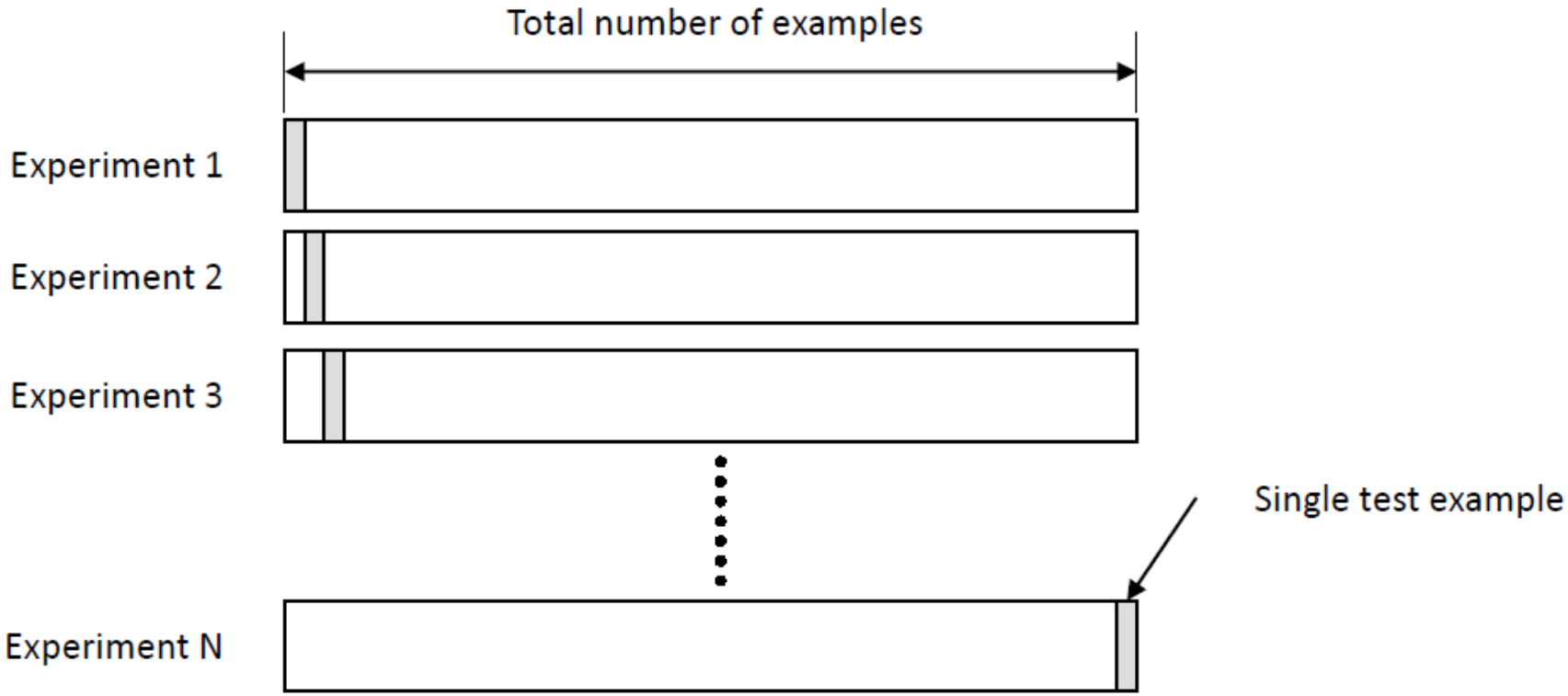
- L'estimation de l'erreur du modèle est obtenue en moyennant les erreurs obtenus sur les différentes expériences
- Cette estimation est significativement meilleure que celle obtenue précédemment, si le nombre d'expériences est suffisant

Cross-Validation



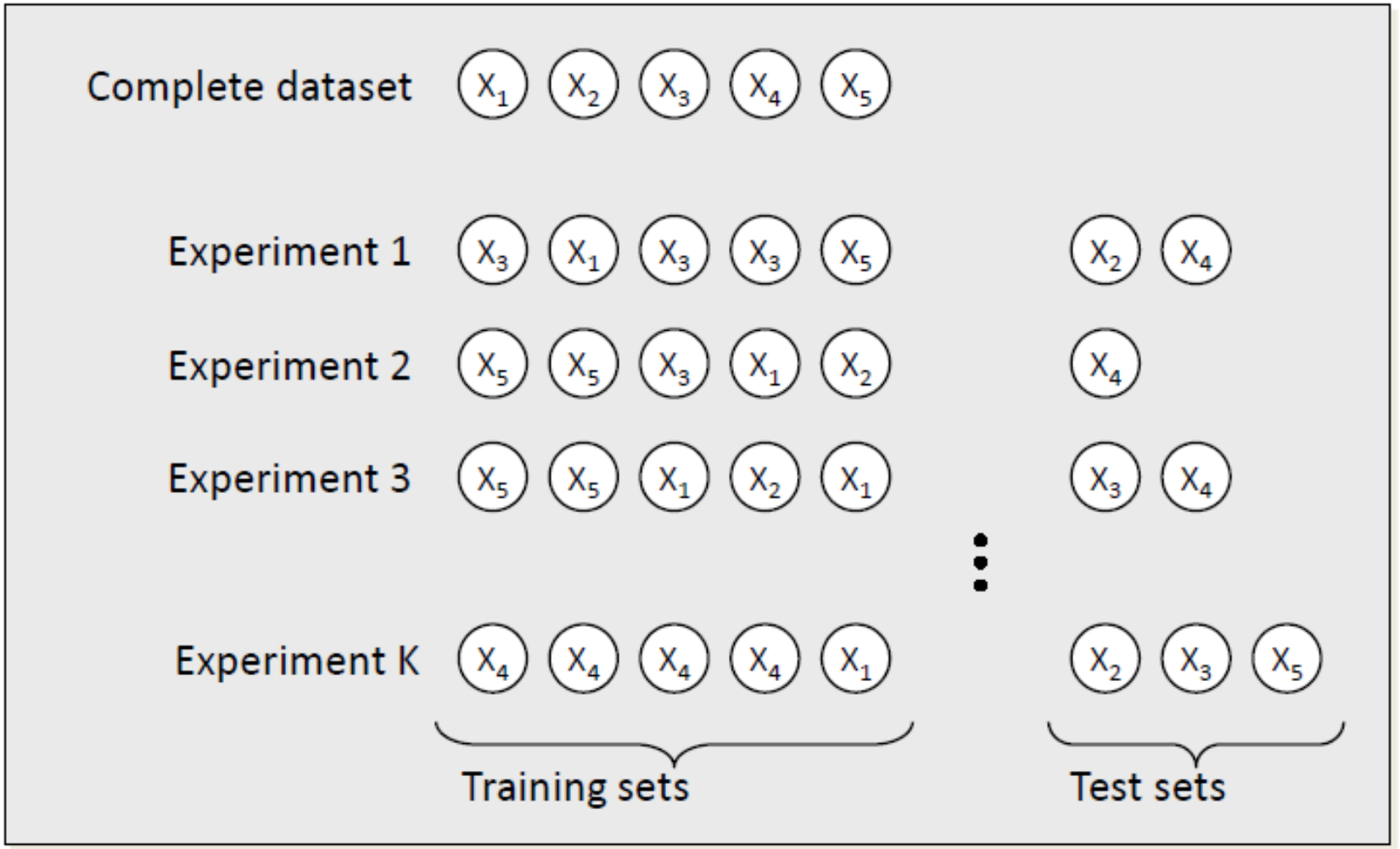
- L'estimation de l'erreur du modèle est obtenue en moyennant les erreurs obtenus sur les différentes expériences
- Tous les exemples sont utilisés pour apprendre au moins un modèle

Leave-one-out



- L'estimation de l'erreur du modèle est obtenue en moyennant les erreurs obtenus sur les différentes expériences
- Cas dégénéré de CV $\rightarrow$ plus robuste, meilleurs pour les petits jeux

Bootstrap


Complete dataset
X1 X2 X3 X4 X5
Experiment 1
X3 X1 X3 X3 X5
X2 X4
Experiment 2
X5 X5 X3 X1 X2
X4
Experiment 3
X5 X5 X1 X2 X1
X3 X4
Experiment K
X4 X4 X4 X4 X1
X2 X3 X5
Training sets
Test sets

Train/Test/Validation

On considère le cas particulier où l'on veut **à la fois** trouver le meilleur modèle **mais aussi** estimer sa performance.

Solution Il faut découper en trois:

- Train set
- Validation set : pour découvrir le meilleur modèle
- Test set : pour évaluer la performance

Courbes d'apprentissage

(dessin au tableau)

Séléction de caractéristique

Séléction de caractéristique

Sélection de caractéristiques sélectionner un sous-ensemble des caractéristiques existantes:

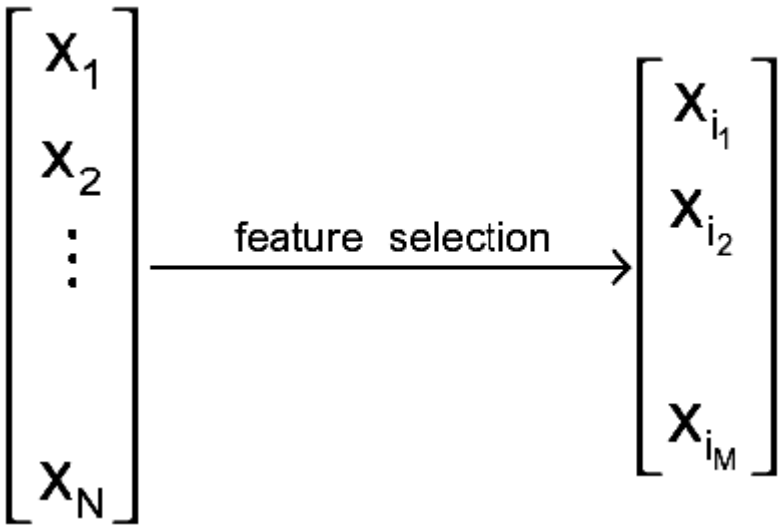
- Approches de type **Filtering**
- Approches de type **Wrappers**

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)**

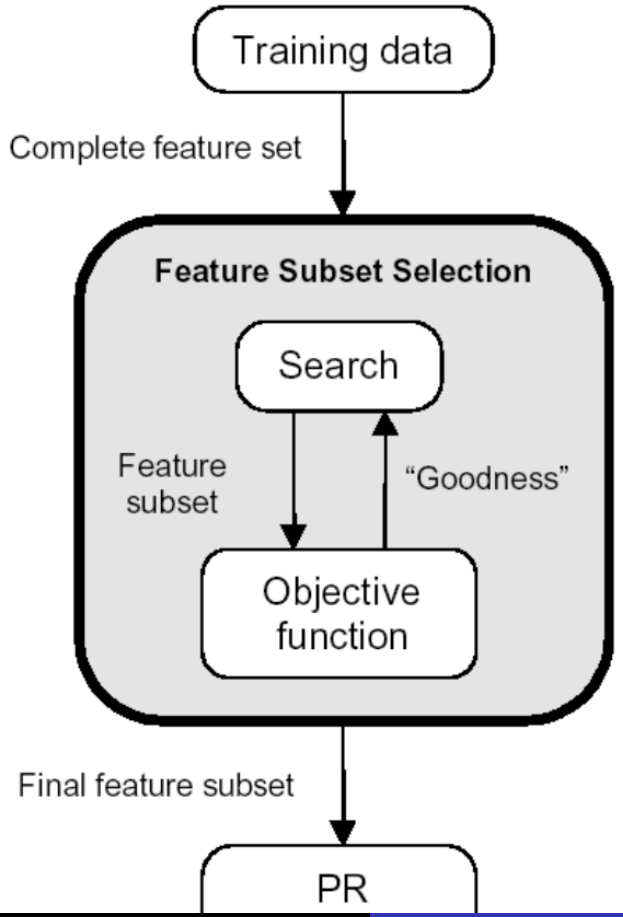
Séléction de caratéristiques

- 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 $x = (x_{i_1}, x_{i_2}, ..., x_{i_M})$



Sélection de caratéristiques

- Très grand espace de recherche
- Besoin de méthodes approchées



Deux approches

Méthodes de filtrage: sélection *a priori*

- Estimation du pouvoir prédictif de chaque caractéristique
- Étude mono-dimensionnelle de chaque caractéristique
- Sélection de celles avec le pouvoir prédictif le plus élevé

Méthodes de wrappers: sélection *a posteriori*

- Choix basé sur la qualité du modèle obtenu

Corrélation

Mesure de l'intensité de la liaison entre deux variables

Corrélation linéaire Soit la variable X_i (caractéristique) et la variable Y (étiquette):

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

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

Corrélation empirique

Comme d'habitude lois inconnues pour X_i et Y

Estimateur

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

- Dépendance linéaire
- Versions non-linéaires
- **Corrélation n'est pas causalité**

Méthodes de Filtrage

- Tri des variable par ordre de pertinence
- Conservation des caractéristiques les plus pertinentes

Avantage

- Chaque caractéristique est analysée **indépendamment des autres**.
- Rapide

Limite

- Chaque caractéristique est analysée **indépendamment des autres**.
- Une variable pourrait être utile en combinaison avec une autre

Méthodes de Wrappers

- Choisir un sous-ensemble de caractéristiques
- Entraîner un modèle et l'évaluer
- Choisir le sous-ensemble qui donne les meilleurs performances

Coûteux:

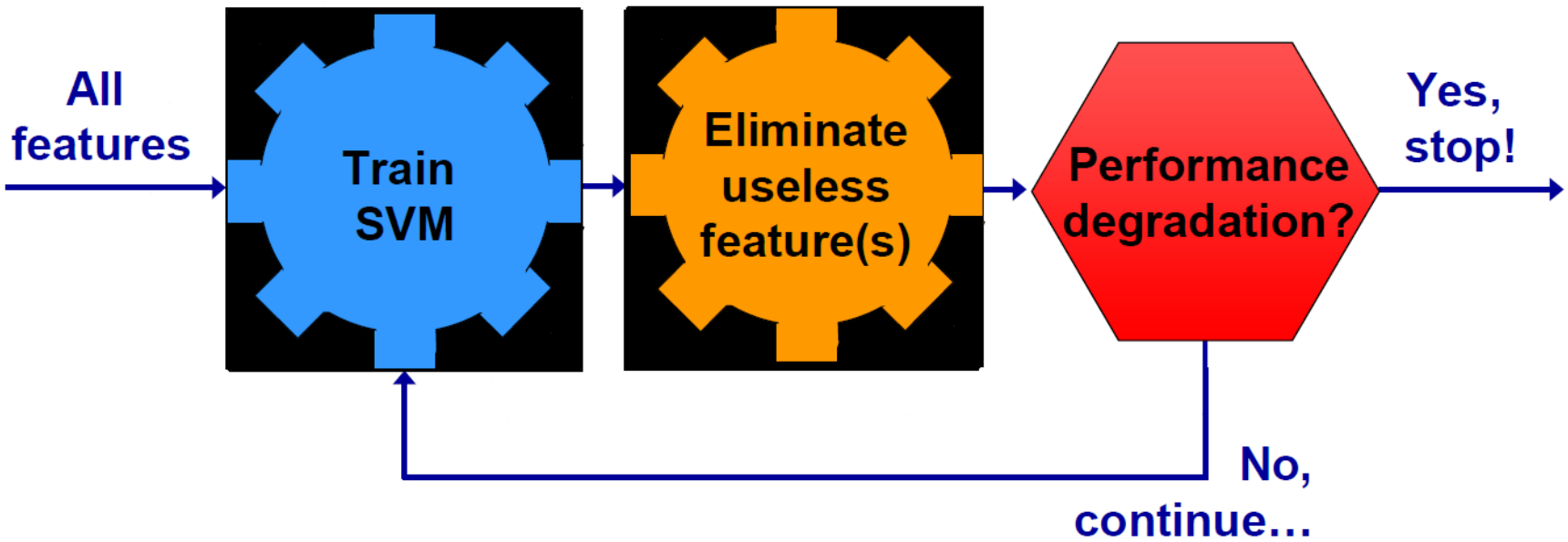
- Nombre exponentiel de sous-ensembles
- Entraînement des modèles

Recherche gloutonne: ajout graduel de caractéristiques basé sur un score à chaque pas de l'algorithme

Attention: le score doit refléter la performance du système (en généralisation)

Méthodes embarquées

De moins en moins de caractéristiques



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

Conclusion

Protocole expérimental classique:

- Étudier les données
- Diviser les données en trois ensembles
- Entrainer un modèle sur l'ensemble de *train*
- Évaluer le modèle sur l'ensemble de *validation*
- Recommencer jusqu'à obtenir le meilleur modèle et les meilleurs hyper-paramètres
- Évaluer la qualité finale du modèle sur l'ensemble de *test*