CentraleSupélec

Big Data : Informatique pour les données et calculs massifs

12 – Objectifs et principes du *Machine Learning*

Stéphane Vialle

CentraleSupélec université PARIS-SACLAY ÉCOLE DOCTORALE Sciences et technologies de l'information et de la communication (STIC) RISEGrid Grand Est ALSACE CHAMPAGNE-ARDENNE LORRAINE

Stephane.Vialle@centralesupelec.fr
http://www.metz.supelec.fr/~vialle

CentraleSupélec

Objectifs et principes du *Machine Learning*

1. **Définitions et Objectifs du ML**
2. Taxonomie des algorithmes
3. Entrainement, validation et tests
4. Problèmes liés à la « grande dimension »

Définitions et objectifs du ML

Définition :

Le *Machine Learning* (ML) est un ensemble :

- d'outils statistiques,
- d'algorithmes informatiques
- et d'outils informatiques

qui permettent :
d'automatiser la construction d'une fonction de prédiction *f* à partir d'un ensemble d'observations (l'ensemble d'apprentissage)

Le **ML** est une **discipline hybride** à cheval sur :

- les mathématiques statistiques (essentiellement)
- l'informatique (algorithmique séquentielle et parallèle, codage performant et traitements large échelle)

3

Définitions et objectifs du ML

Définition :

Du point de vue informatique, un **modèle** de *Machine Learning* est :

- un algorithme
- qui permet de construire une fonction de prédiction *f* à partir d'un jeu de données d'apprentissage

- La construction de *f* constitue l'apprentissage ou l'entrainement du modèle
- Une prédiction correspond à l'évaluation de *f* sur les variables prédictives d'une observation

$$x \longrightarrow f(x)$$

4

Définitions et objectifs du ML

Objectif du *Machine Learning* :

- L'objectif du ML n'est **pas** de trouver des causes en examinant une chronologie (comme dans une démarche scientifique),
- mais **de trouver des corrélations** pertinentes entre les variables prédictives des observations et les variables cibles

→ Rien n'impose qu'une variable prédictive soit la cause d'un phénomène décrit par une variable cible

Ex : *il y a une corrélation entre :*

- *la couleur des dents d'un fumeur (variable prédictive, observation)*
- *et le taux de goudron dans ses poumons (variable cible)*

Mais la couleur de ses dents n'est pas la cause de l'état de ses poumons!

5

Définitions et objectifs du ML

Objectif du *Machine Learning* :

- L'objectif du ML n'est **pas** de trouver des causes en examinant une chronologie (comme dans une démarche scientifique),
- mais **de trouver des corrélations** pertinentes entre les variables prédictives des observations et les variables cibles

→ Avec la plupart des modèles de ML il faut se contenter de détections de corrélations sans explications…

…car le ML ne cherche pas des relations de causes à effets (il ne raisonne pas!)

- Certains modèles de ML peuvent toutefois expliquer leur processus / leur « raisonnement » (ex : arbres de décisions)
- Cette capacité d'explication/de détail du processus est de plus en plus réclamée dans les applications industrielles

6

Définitions et objectifs du ML

Qualités d'un bon algorithme de ML en environnement industriel

Proposition de Ted Dunning, MapR, congrès Big Data, Paris 2014

- **Déployabilité** : passer à l'échelle sur un environnement distribué

  Un algorithme complexe qui ne passe pas à l'échelle ne sera probablement pas utile !

- **Robustesse** : supporter des données du monde réel, incohérentes et incomplètes

  Les algorithmes très pointus mais très sensibles aux données « sales » ne seront pas applicables facilement.
  Ne pas être trop sensible aux données aberrantes est un atout si on ne peut pas bien préparer les données.

- Transparence
- Adéquation aux compétences disponibles
- Proportionnalité

7

Définitions et objectifs du ML

Qualités d'un bon algorithme de ML en environnement industriel

Proposition de Ted Dunning, MapR, congrès Big Data, Paris 2014

- Déployabilité
- Robustesse
- **Transparence :** détecter automatiquement une dégradation des perfs de l'appli quand le processus d'apprentissage progresse

  Reboucler avec une évaluation globale de l'application … pas simple.
  Très utile pour les algorithmes d'apprentissage continu « online »

- **Adéquation aux compétences disponibles :** ne pas exiger d'expertise trop poussée pour l'implantation et l'optimisation

  Les statisticiens/*data scientists* ne sont pas des informaticiens, et réciproquement ! Si un algorithme nécessite des compétences pointues en Math et en Informatique, il va coûter cher… **Même pb en HPC**

- Proportionnalité

8

Définitions et objectifs du ML

Qualités d'un bon algorithme de ML en environnement industriel

Proposition de Ted Dunning, MapR, congrès Big Data, Paris 2014

- Déployabilité
- Robustesse
- Transparence
- Adéquation aux compétences disponibles
- **Proportionnalité :** le temps et l'argent investis dans un algo de ML ou dans son optim, doivent être proportionnels au gain obtenu

  Souci classique, mais révélateur de mauvaises expériences !

Ces « qualités » en environnement industriel révèlent les *difficultés / mauvaises surprises* déjà rencontrées.

9

Définitions et objectifs du ML

Compétences et rôle du *data scientist* :

1 - Etre capable de **choisir le bon algorithme de ML**
→ avoir une double connaissance
- du problème métier que l'on veut modéliser
- des algo de ML et de leurs hypothèses présupposées

2 - Etre capable de **guider le processus d'apprentissage**
→ explorer et préparer les données
→ choisir les variables prédictives les plus significatives

Rmq : La visualisation des données est souvent primordiale pour être capable d'améliorer le processus d'apprentissage

10

CentraleSupélec

Objectifs et principes du *Machine Learning*

1. Objectifs du « Machine Learning »
2. **Taxonomie des algorithmes**
3. Entrainement, validation et tests
4. Problèmes liés à la « grande dimension »

11

CentraleSupélec

Taxonomie des algorithmes

Deux axes de classement des algorithmes de ML

- **Le mode d'apprentissage**
  → algorithmes supervisés et non-supervisés
- **Le type de problème traité pour les algorithmes supervisés**
  → algorithmes de régression et de classification

Algorithme	Mode d'apprentissage	Type de problème
Régressions linéaires, polynomiales et régularisés	Supervisé	Régression
Naïve Bayes	Supervisé	Classification
Arbre de décision	Supervisé	Régression ou classification
Clustering hiérarchique	Non-supervisé	*(Classification)*
….		

12

Taxonomie des algorithmes

Mode d'apprentissage supervisé :

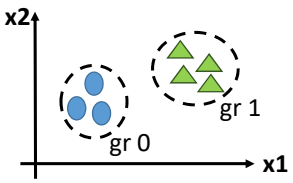
- Les données sont des ensembles de couples entrée-sortie
- Les sorties peuvent être
  - des mesures observées
    (ex. sorties de capteurs)
  - des indications d'experts
    (ex. insuffisant/superficiel/maitrisé/expert)
- Ces algorithmes cherchent à mettre au point/à apprendre une fonction de prédiction qui associe les entrées aux sorties

13

Taxonomie des algorithmes

Mode d'apprentissage supervisé :

Ex. : des *individus* de coordonnées 2D (x,y) forment 2 groupes bien identifiables et disjoints



1. On introduit une variable de plus : l'Id du groupe de sortie, et on étiquette les entrées (gr_0 ou gr_1)
2. L'algorithme « apprend » des couples : ((x1,x2), gr_Id)
3. L'algorithme met au point une fonction de prédiction
   $f : (x1,x2) \rightarrow gr\_Id$

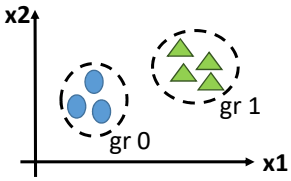
   qui tente de répondre correctement pour tous les points sur lesquels on l'a entraîné, mais aussi sur d'autres points

14

Taxonomie des algorithmes

Mode d'apprentissage supervisé :

Ex. : des *individus* de coordonnées 2D (x,y) forment 2 groupes bien identifiables et disjoints



1. On introduit une variable de plus : l'Id du groupe de sortie, et on étiquette les entrées (gr_0 ou gr_1)
2. L'algorithme « apprend » des couples : ((x1,x2), gr_Id)
3. L'

Rmq : ici l'apprentissage se fait à partir d'indications fournies par un expert

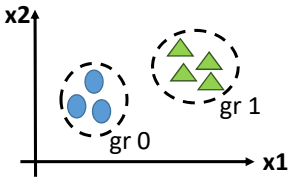
qui tente de répondre correctement pour tous les points sur lesquels on l'a entraîné, mais aussi sur d'autres points

15

Taxonomie des algorithmes

Mode d'apprentissage non-supervisé :

Ex. : des *individus* de coordonnées 2D (x,y) forment 2 groupes bien identifiables et disjoints



1. On n'introduit pas de variable supplémentaire, et l'algo construit **tout seul** des groupes de pts d'entrées à partir des coordonnées des pts
2. Certains algo peuvent être contraints pour construire *k* groupes ou des groupes d'un *rayon maximal*
3. L'algorithme met toujours au point une fonction de prédiction
   $f : (x1,x2) \rightarrow gr\_Id$

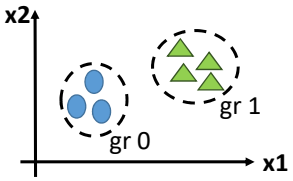
   qui tente de répondre correctement pour tous les points sur lesquels on l'a entraîné, mais aussi sur d'autres points

16

Taxonomie des algorithmes

Mode d'apprentissage non-supervisé :

Ex. : des *individus* de coordonnées 2D (x,y) forment 2 groupes bien identifiables et disjoints



1. On n'introduit **pas** de variable supplémentaire, et l'algo construit **tout seul** des groupes de pts d'entrées à partir des coordonnées des pts
2. Certains algo peuvent être contraints pour construire *k* groupes ou des
3. L' diction

Rmq : ici l'apprentissage ne se fait plus à partir d'indications fournies par un expert, mais à partir de fluctuations dans les valeurs d'entrée

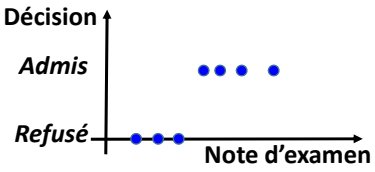
qui tente de répondre correctement pour tous les points sur lesquels on l'a entraîné, mais aussi sur d'autres points

17

Taxonomie des algorithmes

Algorithmes supervisés

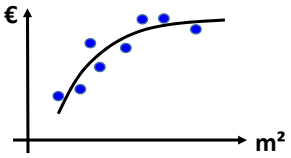
- Si la sortie peut prendre un ensemble fini de valeurs (1,…,k), → algorithmes supervisés de **classification** (selon les *étiquettes* des valeurs d'entrée)



Décision = f(note d'examen)

→ Problème de **classification** pour décider l'admission ou le refus

- Si la sortie peut prendre une infinité de valeurs (réelles), → algorithmes supervisés de **régression**



Prix = f(taille en m²)

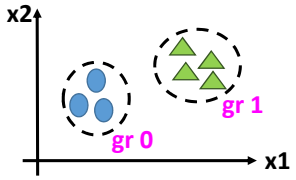
→ Problème de **régression** pour estimer le prix

18

Taxonomie des algorithmes

Algorithmes non-supervisés

- L'algorithme identifie (seul)un ensemble fini de groupes de données d'entrées (1,…,k),
  → un algo non-supervisés réalise **toujours** une classification



Groupe = f(x1,x2)

→ Problème de **classification** en deux groupes émergeants

19

Objectifs et principes du *Machine Learning*

1. Objectifs du « Machine Learning »
2. Taxonomie des algorithmes
3. **Entrainement, validation et tests**
4. Problèmes liés à la « grande dimension »

20

Entrainement, validation et tests

Approche de base :

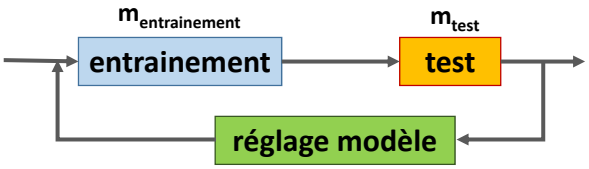
Soit m l'ensemble des observations disponibles

→ On le coupe en **deux** :

$m_{entrainement}$: 70% de m + m_{test} : 30% de m

→ On entraine le modèle sur $m_{entrainement}$, et on le teste sur m_{test}

Mais en fait … on entre dans une boucle d'optimisation :



on optimise le modèle en affinant ses paramètres

→ On risque d'optimiser le modèle pour les observations des tests (m_{test}) !!

21

Entrainement, validation et tests

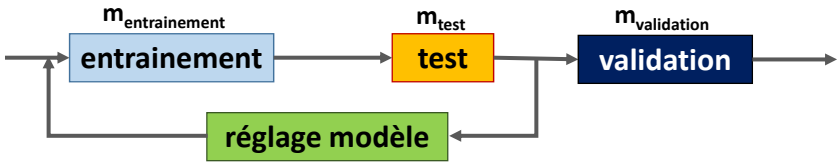
Approche améliorée :

Soit m l'ensemble des observations disponibles

→ On le coupe en **trois** :

- $m_{entrainement}$: 60% de m
- m_{test} : 20% de m
- $m_{validation}$: 20% de m

} Pour la boucle d'optimisation



→ Une fois le modèle optimisé et entrainé, on valide (ou non) sa généricité sur un jeu de données encore jamais utilisé

22

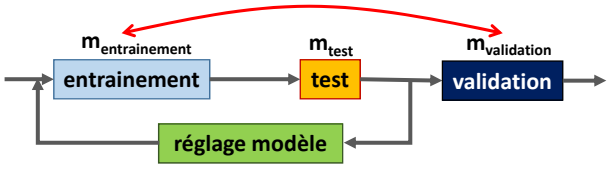
Entrainement, validation et tests

Approche améliorée par validation croisée :

→ On coupe toujours l'ensemble des observations en **trois** :

- m_{test} : 20% de m : Que pour les tests
- $m_{entrainement}$: 60% de m
- $m_{validation}$: 20% de m

} Globalement 80% pour entrainement et validation



→ On utilise 80% des données pour participer *tantôt* à $m_{entrainement}$ et *tantôt* à $m_{validation}$

2 variantes

23

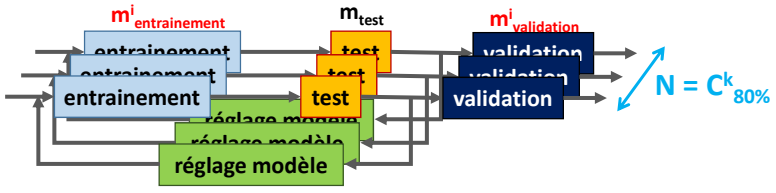
Entrainement, validation et tests

Approche par validation croisée : « leave-k-out cross-validation »

- m_{test} : 20% de m
- $m_{entrainement}$: 80% de m – *k* observations
- $m_{validation}$: *k* observations

Et on réalise toutes les combinaisons possibles de cross-validation

→ $N = C^k_{80\%}$ expérimentations



Puis les N erreurs calculées sont utilisées pour évaluer la perf globale du modèle

→ Approche méthodique mais longue !

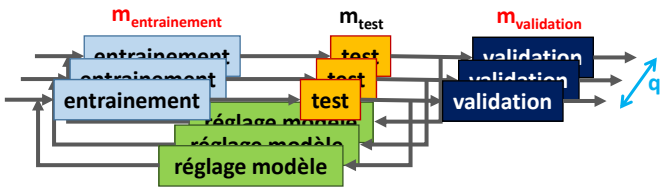
24

Entrainement, validation et tests

Approche par validation croisée : «k-fold cross-validation »

- m_{test} : 20% de m
- 80% de m découpés aléatoirement en *q* paquets de même taille
  - $m_{entrainement}$: *q-1* paquets
  - $m_{validation}$: *1* paquet

Et on teste les *q* combinaisons possibles de paquets



Puis les *q* erreurs calculées sont utilisées pour évaluer la perf globale du modèle

→ Approche moins systématique mais plus rapide !

25

Objectifs et principes du *Machine Learning*

1. Objectifs du « Machine Learning »
2. Taxonomie des algorithmes
3. Entrainement, validation et tests
4. **Problèmes liés à la « grande dimension »**

26

Problèmes de la grande dimension

1 - La malédiction de la dimension

"Curse of dimensionality"
Richard Bellman, 1961.

Le volume englobant les données augmente exponentiellement avec la dimension de l'espace des données

- Souvent les données deviennent **éparses** dans un espace en grande dimension
- Les modèles de régression deviennent moins pertinents :
  - On peut expliquer un nuage de point épars par de nombreux modèles !
  - Une fluctuation dans une variable peut entraîner de gros changements dans le modèle qui devient instable
- Les modèles de classification sont également perturbés par des données éparses (plus complexes)

27

Problèmes de la grande dimension

2 - Des volumes de données énormes

Si les données ne sont **pas éparses**, alors leur volume devient vraiment énorme !

→ Problèmes de stockage, de chargement en RAM, et d'interrogation en temps limité

→ Et de plus en plus de difficultés à représenter les données et à les visualiser…

28

Problèmes de la grande dimension

3 - De plus en plus de caractéristiques encodées dans chaque donnée

Chaque dimension correspond à une caractéristique du système
(ex. une donnée de marché ↔ une dimension du système)

→ Chaque donnée encode de plus en plus de caractéristiques

- Problème de corrélation des caractéristiques (partielle ou forte)
- Problème des caractéristiques non pertinentes pour l'analyse

→ Un grand nombre de dimensions **complique** l'utilisation d'un modèle

29

Objectifs et principes du *Machine Learning*

30