Fondements de l'apprentissage machine

Automne 2014

Roland Memisevic

Leçon 1

IFT 3395/6390, Automne 2014

- Local :
  - cours théorique :
    Mercredi 15 : 30 - 17 : 30 S-144 Pav. Roger-Gaudry
    Jeudi 9 : 30 - 10 : 30 Z-210 Pav. Claire-McNicoll
  - démo :
    Jeudi 10 : 30 - 12 : 30 1340 Pav. André-Aisenstadt
- Professeur :

Roland Memisevic,
3349 Pav. Andre-Aisenstadt
roland.memisevic@umontreal.ca

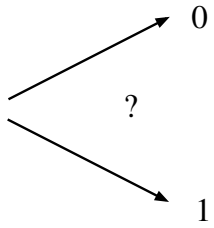
- Page Web du cours :

http ://www.iro.umontreal.ca/~memisevr/teaching/ift3395_2014/index.html

Qu'est-ce que l'apprentissage machine ("Machine Learning") ?

- Certaines tâches sont très difficiles ou fastidieuses pour programmer.
- L'apprentissage automatique remplace la programmation par l'apprentissage à partir d'exemples.

Que caractèrise un visage ?



Que caractèrise le chiffre 2 ?

- Quelles sont les règles qui définissent le ”2”.

Que caractèrise la langue Française ?

“Wikipedia.fr est un site de l’association Wikimedia France. Les resultats du moteur de recherche proviennent de Wikiwix”

- Comment peut-on reconnaître la langue, étant donné un flux de caractères ?

D’autres exemples

- Détection de pourriel
- Reconnaissance de l’identité faciale (eg., Facebook)
- Reconnaissance de sourire (par un appareil photo)
- Détection des anomalies des réseaux
- Reconnaissance de la musique (par téléphone mobile)
- Prédiction des cours boursiers
- Traduction automatique
- Système de recommandation
- Reconnaissance de parole
- Reconnaissance d’activités (console de jeu video)
- Surveillance automatique
- Construction de robots autonomes
- Des voiture qui peuvent conduire par elles même
- “Divination par télépathie”
- etc.

L’analyse des données

- Dans beaucoup de domaines scientifiques on est confrontés à une rapide augmentation des données.
- L’apprentissage automatique fournit un moyen d’utiliser ces données.

```
01111110010000010011111110100111110000010000101
01101010111101010001001000111000001010010100100
00100000000000101111100111110011010010000100100
11001010011001000010100100101100010100001110100
00011101000011001010100011110000010010000001011
10010100001011111001000000111011110010011011110
10100111111101110110011010100011000010100110000
```

Relation aux statistiques

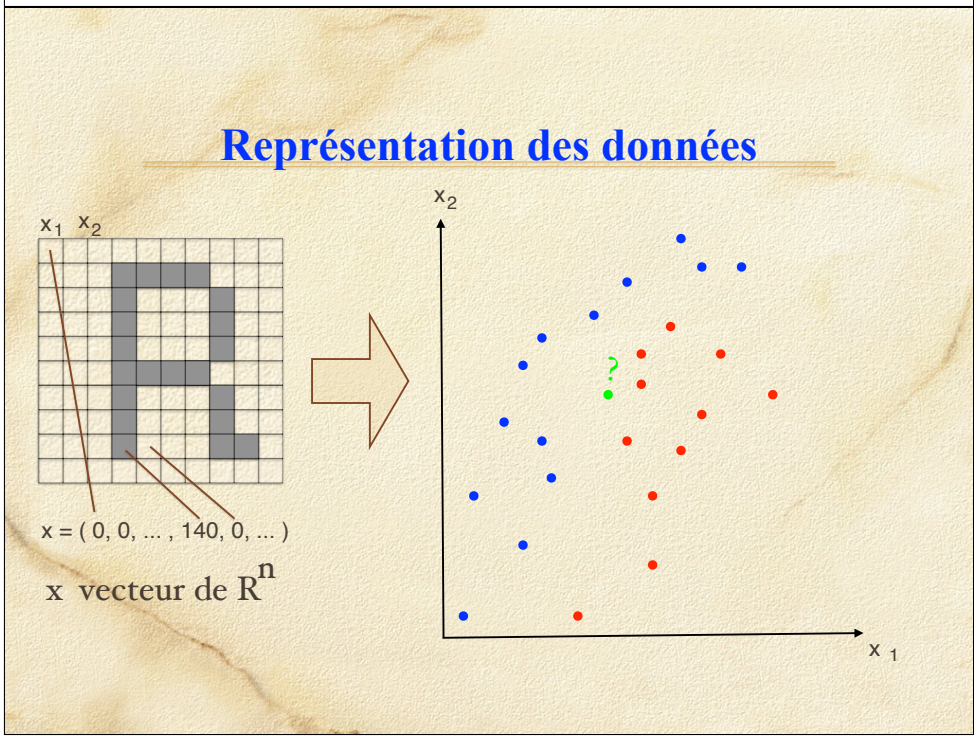
- La statistique est le science classique pour l'analyse de données.
- Une tentative de distinguer les deux :
- L'accent dans la statistique est sur les outils qui permettent aux humains d'analyser les données. L'accent dans l'apprentissage machine est sur les systèmes qui comprennent des données par eux-mêmes.

Relation aux neuro-sciences

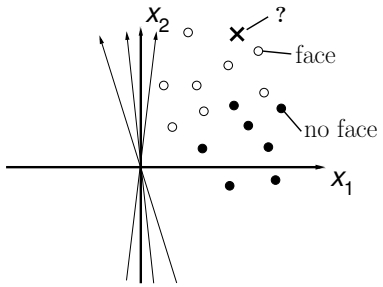
- Le meilleur système du monde pour l'analyse des données est le cerveau.
- Plusieurs techniques d'apprentissage automatique sont inspirées par ce que nous savons sur le traitement de l'information dans le cerveau.
- En retour, l'étude de la théorie de l'apprentissage peut fournir des conseils sur le traitement de l'information dans le cerveau.
- Exemple : "Le cervau Bayésien" (The Bayesian Brain)

Notation

- $\boldsymbol{x}$: entrée, $\boldsymbol{t}$: sortie désirée, $\boldsymbol{y}(\boldsymbol{x})$: sortie de modèle
- $\boldsymbol{x}$ et $\boldsymbol{t}$ peuvent être
  - des scalaire ou des vecteurs
  - pour des scalaire nous utiliserons aussi x and t (des caractères non-gras)
- $\boldsymbol{x}$ et $\boldsymbol{t}$ peuvent être
  - continu : par exemple $x = 1.73457$, or $\boldsymbol{x} \in \mathbb{R}^D$
  - discrète : par exemple $x \in \{0, 1\}$, or $\boldsymbol{x} \in \{\text{'}a\text{'}, \text{'}b\text{'}, \text{'}c\text{'}, \ldots, \text{'}z\text{'}\}^D$
- L'apprentissage revient souvant à ajuster des paramètres d'un modèle. Nous utilisons souvent le vecteur $\boldsymbol{w}$ pour désigner tous les paramètres d'un modèle. Donc $\boldsymbol{y}(\boldsymbol{x}) = \boldsymbol{y}(\boldsymbol{x}; \boldsymbol{w})$



Travailler avec des données de grande dimension



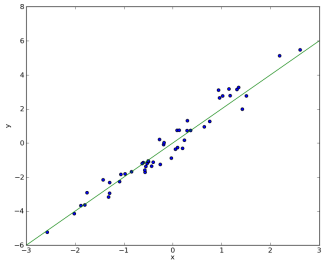
- Les exemples, $\boldsymbol{x}$, (surtout les entrées mais parfois aussi les sorties) sont souvant de grande dimension.
- Nous ne pouvons pas imaginez que trois dimensions.
- Notres intuitions tridimensionnelles fonctionne souvent, mais parfois ils échouent !
- Des espaces de grande dimension ont des propriétés particulières (”malédiction de la dimension”)

Classification des tâches d'apprentissage

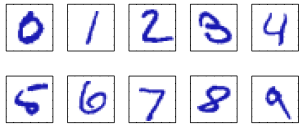
- **Apprentissage supervisé :** On a des exemples d'entraînement ($\boldsymbol{x}$, $\boldsymbol{t}$), et la tâche est d'apprendre une fonction $\boldsymbol{y}(\boldsymbol{x})$ qui peut *predire* les sorties, $\boldsymbol{t}$, pour des nouvelles entrées $\boldsymbol{x}$.
- **Apprentissage non supervisé :** On a des exemples d'entraînement non marqué $\boldsymbol{x}$, et la tâche est d'apprendre une meilleur representation pour les données.
- (**Apprentissage par renforcement :** La tâche est de prendre une série de mesures qui maximisent récompense (par exemple, dans les jeux. Ne sera pas vu en détail dans ce cours.)

Apprentissage supervisé

- **Regression :** Prévoir des sorties continues.

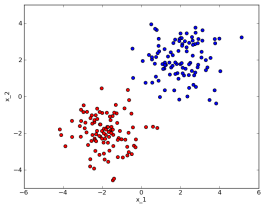


- **Classification :** Prévoir des sorties discrètes.



Apprentissage non supervisé

- **Regroupement (”clustering”) :** Trouver des groupes. C'est comme mettre les données dans une nouvelle représentation (discrète).



- **Réduction de la dimension :** Trouver une nouvelle représentation continue.
- L'apprentissage non supervisé est toujours une forme de *compression avec perte* : Les données sont représentées en utilisant moins de bits.

Sujets (prévu)

- Introduction
- Régression linéaire
- Classification linéaire
- Les réseaux de neurones
- Méthodes à noyaux
- Raisonnement Bayésien
- Clustering, modèles de mélange, l'algorithme EM
- Séquences et modèles de Markov cachés (HMM)
- Les modèles graphiques, inférence approximative, l'échantillonnage
- Prédictions structurées
- L'apprentissage de traits, l'apprentissage profond

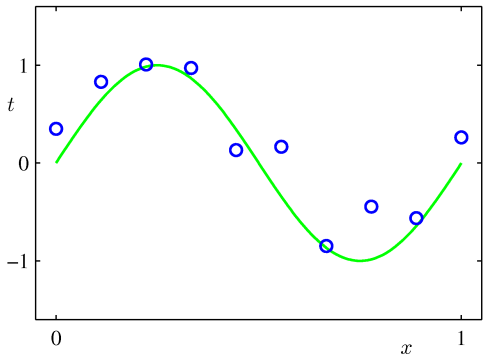
Généralisation et sur-apprentissage ("overfitting")

- Formellement, l'apprentissage est d'adapter des paramètres du modèle sur la base de données d'entraînement.
- Une question centrale est de savoir si le modèle va fonctionner sur des données futures : "Généralisation".
- Ceci est une fonction (a) de comment bien le modèle fait sur les données d'entraînement, et (b) de la taille de l'espace d'hypothèses.

Exemple :

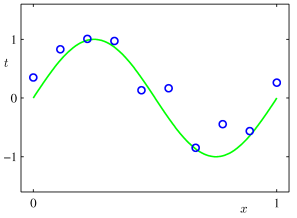
Il est très facile de *memoriser* chaque exemple d'entraînement. Mais cela sera inutile sur de nouvelles données !

Exemple : Estimation d'une courbe (1-D)



- La vraie relation sous-jacente entre x et t est en vert.
- Les observations sont bruyantes (en bleue).

Exemple : Estimation d'une courbe (1-D)



- Modèle polynomial :

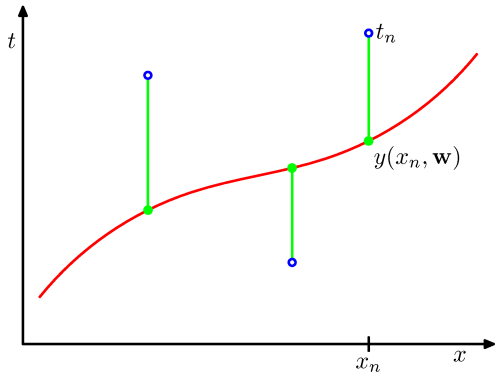
$$y(x, \boldsymbol{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M$$

avec des paramètres

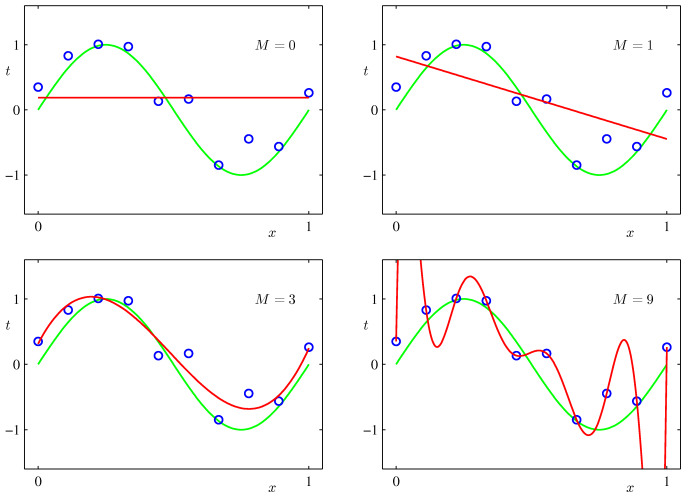
$$\boldsymbol{w} = (w_0, w_1, \ldots, w_M)^{\mathrm{T}}$$

- Pour estimer les paramètres, on peut minimiser l'erreur quadratique : $E(\boldsymbol{w}) = \frac{1}{2} \sum_{n=1}^{N} \left(y(x_n, \boldsymbol{w}) - t_n\right)^2$

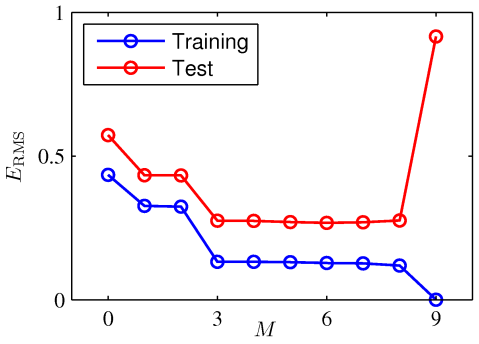
Exemple : Estimation d'une courbe (1-D)



Exemple : Estimation d'une courbe (1-D)

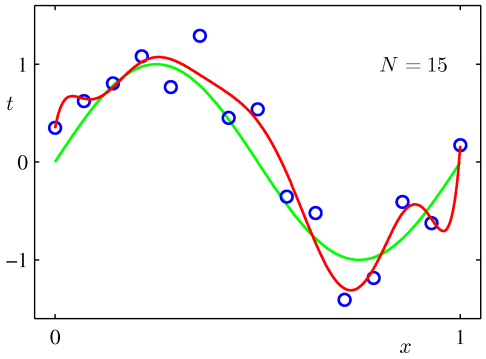


Sur-apprentissage ("overfitting")



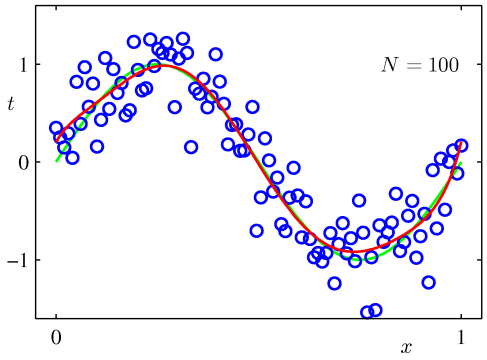
- La dimination de l'erreur sur les données d'entraînement ne signifie pas que celle-ci diminu sur des données test.
- La reduction de la capacité (du *M*) permet de généraliser.

Sur-apprentissage



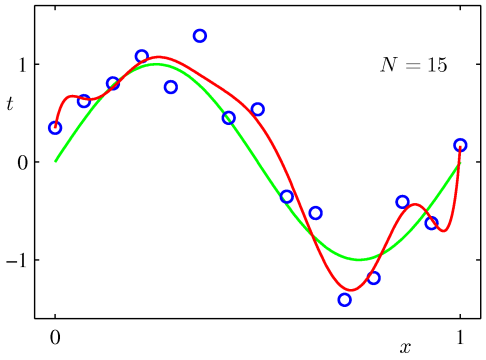
- L'augmentation des données permet aussi de généraliser.

Sur-apprentissage



- L'augmentation des données permet aussi de généraliser.

Exemple : Estimation d'une courbe (1-D)



- Que faire si la quantité de données d'entraînement est limitée ?

Prévenir sur-apprentissage

- **Selection de modèles** : Choisissez la complexité correcte.
- **Régularization :** Pénaliser les grands coefficients :

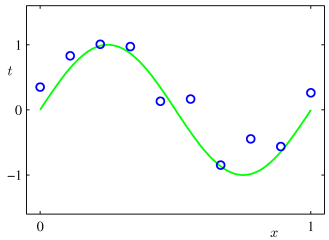
$$\tilde{E}(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(y(x_n, \boldsymbol{w}) - t_n\right)^2 + \frac{\lambda}{2}\|\boldsymbol{w}\|^2$$

- **Modelisation Bayesienne :** Ne pas essayer d'adapter un modèle à tous ! Déduire une distribution de probabilité conditionnelle $p(\text{modele}|\text{data})$. Cette approche est beaucoup plus difficile à faire dans la pratique, mais il est un moyen naturel de prévenir la sur-apprentissage, et cela fonctionne très bien.
- Toutes ces approches sont liées.

Validation croisée

- Pour effectuer la sélection du modèle, il est courant de diviser les données d'entraînement dans un sous-ensemble de données **d'entraînement** approprié et un sous-ensemble de données de **validation**.
- Ensuite, nous estimons plusieurs modèles différents, par example en choisissant différents valeurs pour M ôu λ, et choisissons le modèle qui fait le mieux sur les données de validation.
- On peut échanger les rôles de formation et de validation des sous-ensembles pour obtenir une estimation plus stable. C'est s'appelle validation croisée (**cross-validation**).
- Cas extrême : Prendre touts les sous-ensembles avec $N - 1$ de cas d'entraînement comme ensemble d'apprentissage. **"Leave-one-out" cross validation.**

No free lunch



- Si tout ce que nous avons sont des données d'entraînement, il est *rien* de façon de généraliser.
- Pour avoir en mesure d'apprendre, nous *doit* faire des supposition. Biais inductif **Inductive bias**.
- Une supposition très commun : La douceur de la fonction sous-jacente.

Beaucoup de questions de recherche

- Amélioration de la vitesse, précision, généralité des méthodes.
- Trouvez le biais inductif correct pour la tâche à accomplir.
- L'apprentissage de bout en bout de modèles complexes.
- Des applications.

Conférences et revues principales

- NIPS : Neural Information Processing Systems, ICML, UAI, AISTATS
- PAMI : Pattern Analysis and Machine Intelligence, Journal of Machine Learning Research, Journal Machine Learning, Neural Computation