

# Introduction au Machine Learning

## Préambule

Lucas Gnecco Heredia

LAMSADE - Université Paris-Dauphine  
lucas.gnecco-heredia@dauphine.psl.eu - C602

06 septembre 2022

\* *Remerciement spécial à Florian Yger*

# Table des matières

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
  - Brève présentation du ML
  - Apprentissage supervisé
  - Apprentissage non-supervisé
- 4 Apprentissage supervisé (IML Ch 2)
  - Formalisation du problème
  - Espace des hypothèses
  - Fonctions de coût
  - Généralisation et sur-apprentissage
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
  - Modèles linéaires et moindres carrés
  - Plus proches voisins
- 6 TD / TP

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

# Quelques mots sur la recherche en ML

Un domaine vaste

# Quelques mots sur la recherche en ML

## Un domaine vaste

- à la croisée de nombreux domaines (maths appliquées, statistique, informatique...) et avec de nombreuses applications (vision, signal, robotique)

# Quelques mots sur la recherche en ML

## Un domaine vaste

- à la croisée de nombreux domaines (maths appliquées, statistique, informatique...) et avec de nombreuses applications (vision, signal, robotique)
- avec des considérations théoriques (convergences des méthodes, causalité,...) autant que pratiques (systèmes de recommandation, reconnaissance de visages).

# Quelques mots sur la recherche en ML

## Un domaine vaste

- à la croisée de nombreux domaines (maths appliquées, statistique, informatique...) et avec de nombreuses applications (vision, signal, robotique)
- avec des considérations théoriques (convergences des méthodes, causalité,...) autant que pratiques (systèmes de recommandation, reconnaissance de visages).

## Un domaine en forte demande

- demande en entreprise
- quantité de publications

## Quelques pointeurs

Publications principalement dans des conférences et articles disponibles en ligne.

### Conférences majeures

- 1 NeurIPS, ICML, ICLR, COLT, AAAI, IJCAI, AISTat, ...
- 2 ECML-PKDD, ACML, ECAI, ...

### Journaux majeurs

- 1 JMLR, Journal of Machine Learning, ...

### Quelques chercheurs influents (et actifs sur Twitter)

Francis Bach (INRIA), Gabriel Peyré (ENS/CNRS), Yann Lecun (NYU/FAIR), ...

## Intérêts du moment

- Trustworthy Machine Learning - interprétabilité, causalité, respect de la vie privée, robustesses aux exemples adversariaux, justice/égalité (fairness)
- Machine learning pour données structurés - graphes, problèmes d'optimisation combinatoires

# Un domaine qui ne se résume pas qu'au Deep Learning

## Constat

Le Deep Learning a permis de réelles avancées ces 15 dernières années **mais** il ne faut pas résumer le ML au Deep Learning.

## Limites

- Interprétabilité des résultats (boîte noire) - même si des solutions existent
- Inapplicable (en l'état) sur de petits jeux de données

## Pour ce cours

On se concentre sur les méthodes de base en ML (pas de deep learning au programme).

# Types d'apprentissages

Traditionnellement, on distingue :

- Apprentissage supervisé
- Apprentissage non-supervisé
- Apprentissage par renforcement

# Types d'apprentissages

Traditionnellement, on distingue :

- Apprentissage supervisé
- Apprentissage non-supervisé
- Apprentissage par renforcement

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

# Ressources

Livres guide (Disponibles en ligne et à la BU)

- Hastie, et al. **The elements of statistical learning : data mining, inference, and prediction.** (ESL)
- Azencott. **Introduction au Machine Learning** (IML)

# Ressources

Livres guide (Disponibles en ligne et à la BU)

- Hastie, et al. **The elements of statistical learning : data mining, inference, and prediction.** (ESL)
- Azencott. **Introduction au Machine Learning** (IML)

Autres références utiles

- Murphy. **Machine learning : a probabilistic perspective**
- Bishop. **Pattern recognition and machine learning**

# Plan (envisagé) de ce cours

- Introduction
  - C0 - Rappels / Introduction
- Partie I - Apprentissage supervisé
  - C1 - Premiers modèles pour la classification
  - C2 - Méthodes linéaires pour la régression
  - C3 - Méthodes linéaires pour la classification (LDA, Régression logistique)
- Partie II - Méthodologie
  - C4 - Méthodologie en apprentissage artificiel
- Partie III - Apprentissage non-supervisé
  - C5 - Clustering
  - C6 - Réduction de dimension

# Méthodologie

## Avant le cours

Lecture des sections des livres guide données pour chaque session

## Pendant le cours

- Cours sur la thématique du jour
- Exercices et travail pratique

# Pr -requis

Dans ce cours, on part du principe que vous savez :

- repr senter des donn es (nuages de points en 2d, histogrammes, ...)
- calculer des r sum s des donn es (moyennes, m dianes, variance, covariance, ...)
- pr -traiter des donn es (centrer, r duire, normaliser)
- appliquer des mod les simples (r gression lin aire, ACP)

# Évaluation

- 30% - Projet Python + rapport en binôme, **possibilité de soutenance si nécessaire**
- 70% - Examen

# Quelques outils à maîtriser

## Code

- Python + Scikit-learn/Numpy/Pandas (pour les projets)
- Latex & Bibtex (pour les rapports)

## Outils collaboratifs

- Google Colab - notebooks en ligne
- Overleaf - éditeur latex en ligne
- GitHub - gestionnaire de versions

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
  - Brève présentation du ML
  - Apprentissage supervisé
  - Apprentissage non-supervisé
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

# But d'aujourd'hui

- Comprendre les ingrédients et les concepts principales d'un algorithme de ML
- Différencier entre algorithme "*classique*" et *d'apprentissage*
- Avoir une première session de pratique avec les librairies que l'on va utiliser

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
  - Brève présentation du ML
  - Apprentissage supervisé
  - Apprentissage non-supervisé
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

# Pourquoi et quand utiliser le machine learning

On parle d'apprentissage automatique quand un programme a la capacité de modifier son "comportement" à partir des données et sans que ce soit explicitement programmée

## Intuition simplifiée

Un programme classique contient les règles, prend les entrées et produit les sorties.

Un programme de ML prend les entrées et les sorties et "trouve" les règles.

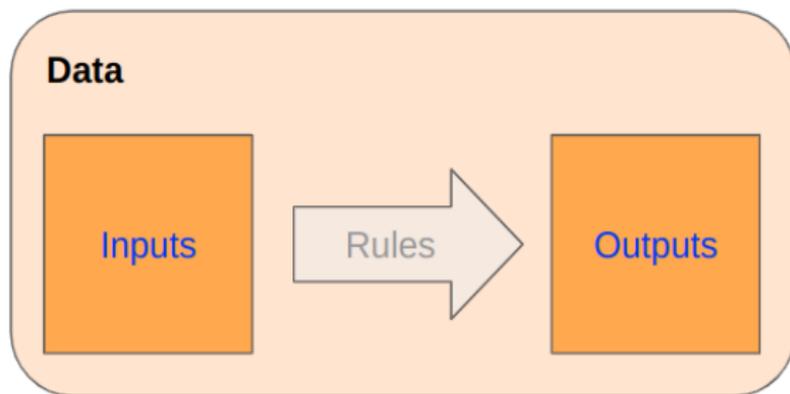
*Exemple avec un modèle  $f(x) = x^T \beta + \epsilon$*

# Ingrédients du ML

Il y a deux éléments fondamentaux :

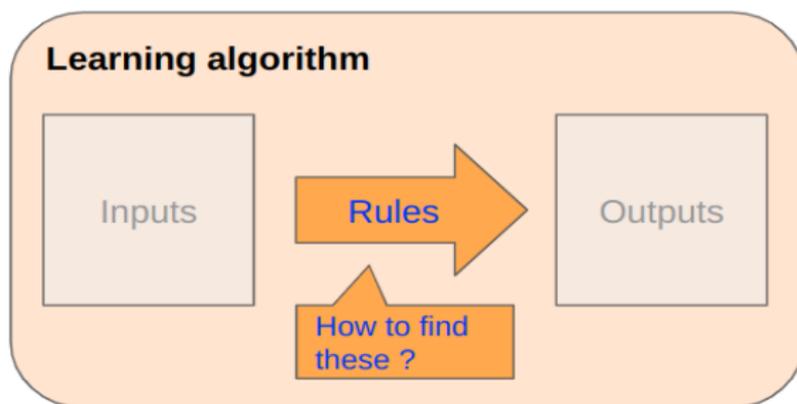
# Ingrédients du ML

Il y a deux éléments fondamentaux :



# Ingrédients du ML

Il y a deux éléments fondamentaux :



- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
  - Brève présentation du ML
  - Apprentissage supervisé
  - Apprentissage non-supervisé
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

# Apprentissage supervisé

On observe des couples  $\{(x_i, y_i)\}_{i=1}^N \subseteq \mathcal{X} \times \mathcal{Y}$  sans connaître le processus  $f : \mathcal{X} \mapsto \mathcal{Y}$  qui les a générés. On va chercher à approximer  $f$  et à trouver une fonction  $\hat{f}$  telle que  $\hat{f}(x) \approx y$  pour les observations passées (*mais aussi futures!*).

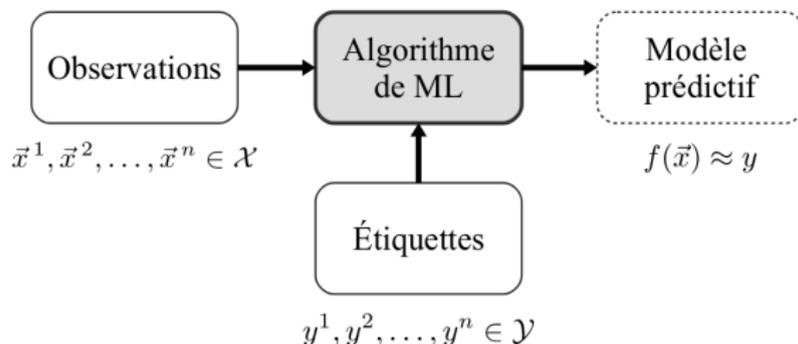


Figure – [Aze19]

# Apprentissage supervisé (exemples)

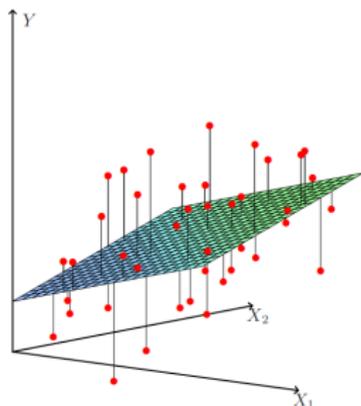


Figure – Exemple de régression

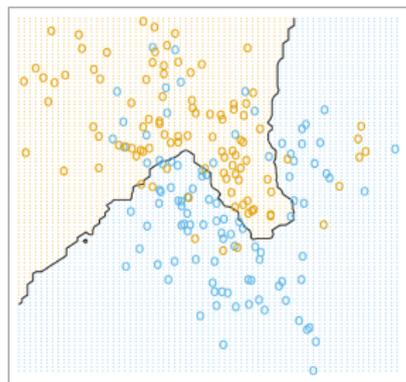


Figure – Exemple de classification

Figure – [Has+09]

# Régression et classification

## Régression

Un problème d'apprentissage supervisé où les étiquettes  $\mathcal{Y}$  son *continues* (en général  $\mathcal{Y} \subseteq \mathbb{R}$ ).

**Ex** : Prédire le nombre de clics sur un lien

## Classification

Un problème d'apprentissage supervisé où les étiquettes  $\mathcal{Y}$  son *discrètes*.

**Ex** : Identifier si un email est un spam, detecter des objets dans une photo

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
  - Brève présentation du ML
  - Apprentissage supervisé
  - Apprentissage non-supervisé
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

# Apprentissage non-supervisé

On observe des données *non étiquetées*  $\{x_i\}_{i=1}^N \subseteq \mathcal{X}$ . On cherche à mieux comprendre les données et leur structure.



Figure – [Aze19]

# Apprentissage non-supervisé (exemples)

## Clustering

- Trouver les types (ou profils) d'acheteurs d'un magasin
- Identifier des groupes de documents ayant un sujet similaire à partir de leur contenu et meta-information (pas d'étiquettes)

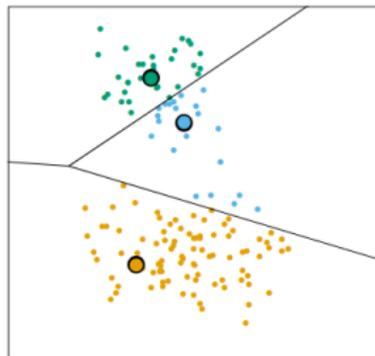


Figure – [Has+09]

# Apprentissage non-supervisé (exemples)

## Réduction de dimension

- Trouver une représentation plus pertinente des données (peut être supervisé aussi)

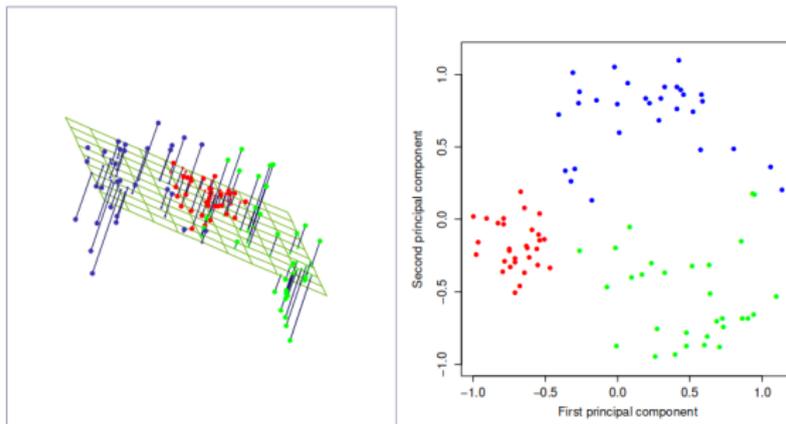


Figure – [Has+09]

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
  - Formalisation du problème
  - Espace des hypothèses
  - Fonctions de coût
  - Généralisation et sur-apprentissage
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
  - Formalisation du problème
    - Espace des hypothèses
    - Fonctions de coût
    - Généralisation et sur-apprentissage
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

# Problème d'apprentissage supervisé

- *Jeu d'apprentissage*  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N \subseteq \mathcal{X} \times \mathcal{Y}$ .

# Problème d'apprentissage supervisé

- *Jeu d'apprentissage*  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N \subseteq \mathcal{X} \times \mathcal{Y}$ .
- En générale on a  $\mathcal{X} = \mathbb{R}^d$ . Chaque dimension correspond à un attribut (*feature* en anglais). Les étiquettes  $y_i$  sont aussi appelées *targets* ou *labels* en anglais.

# Problème d'apprentissage supervisé

- *Jeu d'apprentissage*  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N \subseteq \mathcal{X} \times \mathcal{Y}$ .
- En générale on a  $\mathcal{X} = \mathbb{R}^d$ . Chaque dimension correspond à un attribut (*feature* en anglais). Les étiquettes  $y_i$  sont aussi appelées *targets* ou *labels* en anglais.
- On suppose qu'il existe une fonction  $\phi$  qu'a produit les vraies étiquettes  $y_i$ . On cherchera à l'approximer.

## IMC des élèves de la classe

On pourrait mesurer la taille, le poids et l'âge pour chaque élève de la classe pour essayer de prédire le IMC.

Taille	Poids	Âge	IMC
173	68.2	24.5	20.1
162	56.0	23.1	19.5
169	72.1	22.9	24.3

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
  - Formalisation du problème
  - Espace des hypothèses
  - Fonctions de coût
  - Généralisation et sur-apprentissage
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

# Fonctions de modélisation

Il nous faut définir la famille de fonctions  $\mathcal{F}$  (dites aussi *hypothèses*) à considérer pour lier  $\mathcal{X}$  et  $\mathcal{Y}$  dans notre problème

# Fonctions de modélisation

Il nous faut définir la famille de fonctions  $\mathcal{F}$  (dites aussi *hypothèses*) à considérer pour lier  $\mathcal{X}$  et  $\mathcal{Y}$  dans notre problème

## Exemples de familles d'hypothèses

# Fonctions de modélisation

Il nous faut définir la famille de fonctions  $\mathcal{F}$  (dites aussi *hypothèses*) à considérer pour lier  $\mathcal{X}$  et  $\mathcal{Y}$  dans notre problème

## Exemples de familles d'hypothèses

- Fonctions linéaires

# Fonctions de modélisation

Il nous faut définir la famille de fonctions  $\mathcal{F}$  (dites aussi *hypothèses*) à considérer pour lier  $\mathcal{X}$  et  $\mathcal{Y}$  dans notre problème

## Exemples de familles d'hypothèses

- Fonctions linéaires
- Polynômes de degré au plus  $m$

# Fonctions de modélisation

Il nous faut définir la famille de fonctions  $\mathcal{F}$  (dites aussi *hypothèses*) à considérer pour lier  $\mathcal{X}$  et  $\mathcal{Y}$  dans notre problème

## Exemples de familles d'hypothèses

- Fonctions linéaires
- Polynômes de degré au plus  $m$
- Arbres de décision de taille au plus  $m$

# Trouver la meilleur fonction

Étant donné le jeu de données  $\mathcal{D}$  et la famille d'hypothèses  $\mathcal{F}$ , notre tâche consiste à trouver  $f \in \mathcal{F}$  qui approche au mieux  $\phi$ . On a besoin de :

# Trouver la meilleur fonction

Étant donné le jeu de données  $\mathcal{D}$  et la famille d'hypothèses  $\mathcal{F}$ , notre tâche consiste à trouver  $f \in \mathcal{F}$  qui approche au mieux  $\phi$ . On a besoin de :

- Une façon de mesurer la qualité d'une hypothèse.

# Trouver la meilleur fonction

Étant donné le jeu de données  $\mathcal{D}$  et la famille d'hypothèses  $\mathcal{F}$ , notre tâche consiste à trouver  $f \in \mathcal{F}$  qui approche au mieux  $\phi$ . On a besoin de :

- Une façon de mesurer la qualité d'une hypothèse.
- Une façon de chercher une hypothèse dans  $\mathcal{F}$ .

# Trouver la meilleur fonction

Étant donné le jeu de données  $\mathcal{D}$  et la famille d'hypothèses  $\mathcal{F}$ , notre tâche consiste à trouver  $f \in \mathcal{F}$  qui approche au mieux  $\phi$ . On a besoin de :

- Une façon de mesurer la qualité d'une hypothèse.
- Une façon de chercher une hypothèse dans  $\mathcal{F}$ .

## Attention

Nous allons voir l'importance du choix de l'espace des hypothèses. Un espace trop petit ou trop grand va générer des problèmes de sur ou sous apprentissage

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
  - Formalisation du problème
  - Espace des hypothèses
  - Fonctions de coût
  - Généralisation et sur-apprentissage
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

Une fonction de coût (ou *loss*)  $L : \mathcal{Y} \times f(\mathcal{X}) \mapsto \mathbb{R}$  prend en entrée les **vraies étiquettes**  $y$  et les **étiquettes prédites**  $f(x)$ , et nous donne une mesure d'erreur.

On voudrait donc que nos prédictions **minimisent** la loss.

## Exemples de loss

### ■ Loss 0/1 (classification) :

$$L(y, f(x)) = \begin{cases} 1 & f(x) \neq y \\ 0 & f(x) = y \end{cases}$$

Une fonction de coût (ou *loss*)  $L : \mathcal{Y} \times f(\mathcal{X}) \mapsto \mathbb{R}$  prend en entrée les **vraies étiquettes**  $y$  et les **étiquettes prédites**  $f(x)$ , et nous donne une mesure d'erreur.

On voudrait donc que nos prédictions **minimisent** la loss.

## Exemples de loss

- Loss 0/1 (classification) :

$$L(y, f(x)) = \begin{cases} 1 & f(x) \neq y \\ 0 & f(x) = y \end{cases}$$

- Loss quadratique (régression) :

$$L(y, f(x)) = (f(x) - y)^2$$



# Autres fonctions de coût importantes

## Classification binaire

- **Loss logistique** ( $\mathcal{Y} = \{-1, 1\}$ ) :

$$L(y, f(x)) = \log(1 + \exp(-y \cdot f(x)))$$

# Autres fonctions de coût importantes

## Classification binaire

- **Loss logistique** ( $\mathcal{Y} = \{-1, 1\}$ ) :

$$L(y, f(x)) = \log(1 + \exp(-y \cdot f(x)))$$

- **Cross-entropy** ( $\mathcal{Y} = \{0, 1\}$ ) :

$$L(y, f(x)) = -y \log f(x) - (1 - y) \log(1 - f(x))$$

# Autres fonctions de coût importantes

## Classification binaire

- **Loss logistique** ( $\mathcal{Y} = \{-1, 1\}$ ) :

$$L(y, f(x)) = \log(1 + \exp(-y \cdot f(x)))$$

- **Cross-entropy** ( $\mathcal{Y} = \{0, 1\}$ ) :

$$L(y, f(x)) = -y \log f(x) - (1 - y) \log(1 - f(x))$$

## Domaine de la fonction de coût

Attention aux valeurs reçues par la fonction de coût ! Certaines reçoivent des étiquettes dans  $\{0, 1\}$ , autres dans  $\{-1, 1\}$

## Pourquoi toutes ces fonctions de coût ?

Dans le cas de classification la **Loss 0/1** est la vraie mesure que l'on veut optimiser. Alors, pourquoi utiliser une autre fonction de coût ?

# Pourquoi toutes ces fonctions de coût ?

Dans le cas de classification la **Loss 0/1** est la vraie mesure que l'on veut optimiser. Alors, pourquoi utiliser une autre fonction de coût ?

La **Loss 0/1** est très difficile à optimiser !

- Optimiser une fonction pas lisse et constant partout est très difficile !
- Une fonction de coût lisse et convexe sera beaucoup plus simple à optimiser

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
  - Formalisation du problème
  - Espace des hypothèses
  - Fonctions de coût
  - Généralisation et sur-apprentissage
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

# Généralisation

On veut que notre algorithme d'apprentissage fasse des prédictions correctes pour des nouvelles données pas encore vues lors de la phase d'apprentissage.

# Généralisation

On veut que notre algorithme d'apprentissage fasse des prédictions correctes pour des nouvelles données pas encore vues lors de la phase d'apprentissage.

On appelle cette capacité ***généralisation***

# Généralisation

On veut que notre algorithme d'apprentissage fasse des prédictions correctes pour des nouvelles données pas encore vues lors de la phase d'apprentissage.

On appelle cette capacité ***généralisation***

Deux situations opposées de mauvaise généralisation peuvent se présenter

## Sur-apprentissage (*overfitting* en anglais)

Le modèle a appris “par coeur” les données d’entraînement, y inclus le bruit en général présent dans nos observations.

## Sur-apprentissage (*overfitting* en anglais)

Le modèle a appris “par coeur” les données d’entraînement, y inclus le bruit en général présent dans nos observations.

- **Très bonne performance** sur l’ensemble d’entraînement

## Sur-apprentissage (*overfitting* en anglais)

Le modèle a appris “par coeur” les données d’entraînement, y inclus le bruit en général présent dans nos observations.

- **Très bonne performance** sur l’ensemble d’entraînement
- **Mauvaise performance** sur des nouvelles données

## Sur-apprentissage (*overfitting* en anglais)

Le modèle a appris “par coeur” les données d’entraînement, y inclus le bruit en général présent dans nos observations.

- **Très bonne performance** sur l’ensemble d’entraînement
- **Mauvaise performance** sur des nouvelles données
- Associé à un **excès de complexité** ou **flexibilité** du modèle (ex : polynômes de degré très grand)

## Sur-apprentissage (*overfitting* en anglais)

Le modèle a appris “par coeur” les données d’entraînement, y inclus le bruit en général présent dans nos observations.

- **Très bonne performance** sur l’ensemble d’entraînement
- **Mauvaise performance** sur des nouvelles données
- Associé à un **excès** de complexité ou flexibilité du modèle (ex : polynômes de degré très grand)

## Sous-apprentissage (*underfitting* en anglais)

Le modèle est trop simple et n’est pas capable de capturer les vraies relation dans les données.

## Sur-apprentissage (*overfitting* en anglais)

Le modèle a appris “par coeur” les données d’entraînement, y inclus le bruit en général présent dans nos observations.

- **Très bonne performance** sur l’ensemble d’entraînement
- **Mauvaise performance** sur des nouvelles données
- Associé à un **excès** de complexité ou flexibilité du modèle (ex : polynômes de degré très grand)

## Sous-apprentissage (*underfitting* en anglais)

Le modèle est trop simple et n’est pas capable de capturer les vraies relation dans les données.

- **Mauvaise performance** sur l’ensemble d’entraînement  
nouvelles données

## Sur-apprentissage (*overfitting* en anglais)

Le modèle a appris “par coeur” les données d’entraînement, y inclus le bruit en général présent dans nos observations.

- **Très bonne performance** sur l’ensemble d’entraînement
- **Mauvaise performance** sur des nouvelles données
- Associé à un **excès** de complexité ou flexibilité du modèle (ex : polynômes de degré très grand)

## Sous-apprentissage (*underfitting* en anglais)

Le modèle est trop simple et n’est pas capable de capturer les vraies relation dans les données.

- **Mauvaise performance** sur l’ensemble d’entraînement  
nouvelles données
- Associé à un **manque** de complexité ou flexibilité du modèle

## Exemple de sur et sous apprentissage

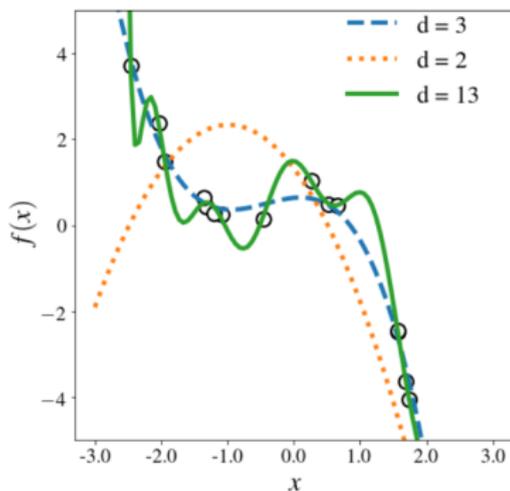


Figure – [Has+09]

	Underfitting	Just right	Overfitting
Symptoms	<ul style="list-style-type: none"> <li>- High training error</li> <li>- Training error close to test error</li> <li>- High bias</li> </ul>	<ul style="list-style-type: none"> <li>- Training error slightly lower than test error</li> </ul>	<ul style="list-style-type: none"> <li>- Low training error</li> <li>- Training error much lower than test error</li> <li>- High variance</li> </ul>
Regression			
Classification			
Deep learning			
Remedies	<ul style="list-style-type: none"> <li>- Complexity model</li> <li>- Add more features</li> <li>- Train longer</li> </ul>		<ul style="list-style-type: none"> <li>- Regularize</li> <li>- Get more data</li> </ul>

Figure – Taken from here

# Remarques

A prendre en compte...

# Remarques

## A prendre en compte...

- Il faut toujours évaluer un modèle avec des nouvelles données !  
(jeu d'entraînement / jeu de test)

# Remarques

## A prendre en compte...

- Il faut toujours évaluer un modèle avec des nouvelles données ! (jeu d'entraînement / jeu de test)
- **Attention surtout au sur-apprentissage.** Les bonnes performances sur l'ensemble d'entraînement mènent parfois à des fausses conclusions

# Remarques

## A prendre en compte...

- Il faut toujours évaluer un modèle avec des nouvelles données ! (jeu d'entraînement / jeu de test)
- **Attention surtout au sur-apprentissage.** Les bonnes performances sur l'ensemble d'entraînement mènent parfois à des fausses conclusions
- Il faudra parfois limiter la complexité d'un modèle pour éviter le sur-apprentissage (**régularisation**)

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
  - Modèles linéaires et moindres carrés
  - Plus proches voisins
- 6 TD / TP

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
  - Modèles linéaires et moindres carrés
  - Plus proches voisins
- 6 TD / TP

# Modèle linéaire

On parle de modèle linéaire quand on modélise la relation entre  $\mathcal{X}$  et  $\mathcal{Y}$  avec une fonction linéaire en  $\mathcal{X}$  :

$$y \approx \beta_0 + \sum_{k=1}^p x^k \beta^k = \beta_0 + \mathbf{x}^T \boldsymbol{\beta}$$

# Modèle linéaire

On parle de modèle linéaire quand on modélise la relation entre  $\mathcal{X}$  et  $\mathcal{Y}$  avec une fonction linéaire en  $\mathcal{X}$  :

$$y \approx \beta_0 + \sum_{k=1}^p x^k \beta^k = \beta_0 + \mathbf{x}^T \boldsymbol{\beta}$$

Souvent on va augmenter  $\mathbf{x}$  avec une composante constante de 1 et inclure aussi l'intercepte  $\beta_0$  dans le vecteur  $\boldsymbol{\beta}$ . Ceci donne une expression plus compacte pour le même modèle

$$y \approx \mathbf{x}^T \boldsymbol{\beta}$$

# Modèle linéaire

On parle de modèle linéaire quand on modélise la relation entre  $\mathcal{X}$  et  $\mathcal{Y}$  avec une fonction linéaire en  $\mathcal{X}$  :

$$y \approx \beta_0 + \sum_{k=1}^p x^k \beta^k = \beta_0 + \mathbf{x}^T \boldsymbol{\beta}$$

Souvent on va augmenter  $x$  avec une composante constante de 1 et inclure aussi l'intercepte  $\beta_0$  dans le vecteur  $\boldsymbol{\beta}$ . Ceci donne une expression plus compacte pour le même modèle

$$y \approx \mathbf{x}^T \boldsymbol{\beta}$$

Apprendre un modèle linéaire revient à trouver le  $\boldsymbol{\beta}$ .

# Apprendre un modèle linéaire

Étant donné un jeu de données  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N \subseteq \mathbb{R}^d \times \mathbb{R}$ , il existent plusieurs méthodes pour apprendre un modèle linéaire, dont le plus populaire est les moindres carrés

# Apprendre un modèle linéaire

Étant donné un jeu de données  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N \subseteq \mathbb{R}^d \times \mathbb{R}$ , il existent plusieurs méthodes pour apprendre un modèle linéaire, dont le plus populaire est les moindres carrés

On cherche  $\beta^{LS}$  qui minimise la fonction de coût quadratique sur l'ensemble de données

$$\beta^{LS} = \operatorname{argmin}_{\beta \in \mathbb{R}^{d+1}} \sum_{i=1}^N \left( y_i - x_i^T \beta \right)^2$$

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
  - Modèles linéaires et moindres carrés
  - Plus proches voisins
- 6 TD / TP

Le modèle de plus proches voisins se base sur une hypothèse de localité. Soit  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$  notre jeu d'entraînement

Le modèle de plus proches voisins se base sur une hypothèse de localité. Soit  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$  notre jeu d'entraînement

### k-plus proches voisins

On choisit  $k$  le nombre de voisins à considérer. Pour chaque nouvelle donnée  $x$  on trouve  $N^k(x)$  les  $k$  plus proches éléments à  $x$  dans  $\mathcal{D}$

- **Régression** :  $f(x) = \frac{1}{k} \sum_{x_i \in N^k(x)} y_i$  (moyenne dans  $N^k(x)$ )
- **Classification** :  $f(x) =$  Classe majoritaire dans  $N^k(x)$

- 1 Généralités sur le ML
- 2 Déroulé du cours
- 3 Introduction (IML Ch 1), (ESL Sec 2.1 - 2.2)
- 4 Apprentissage supervisé (IML Ch 2)
- 5 Exemples : Moindres carrés et plus proches voisins (ESL Sec 2.3)
- 6 TD / TP

Dans *ce notebook* on pourra pratiquer les thématiques abordées, en particulier un exemple comparant un modèle linéaire et un modèle de k-plus proches voisins.

On verra aussi en pratique le sur apprentissage en essayant d'approximer une fonction polynomiale

## Références

- [Aze19] Chloé-Agathe Azencott. *Introduction au machine learning*. Dunod, 2019.
- [Has+09] Trevor Hastie et al. *The elements of statistical learning : data mining, inference, and prediction*. T. 2. Springer, 2009.