Introduction au Machine Learning

Chloé-Agathe Azencott & Fabien Moutarde

Introduction

Chloé-Agathe Azencott

Center for Computational Biology (CBIO)
Mines Paris – Institut Curie – INSERM U1331
PSL Research University & PR[AI]RIE, Paris, France

Sep 14, 2026

http://cazencott.info chloe-agathe.azencott@minesparis.psl.eu @cazencott@lipn.info

Contexte

- **Objectifs :** Être capable de **choisir** et **mettre en œuvre** :
  - des algorithmes d'**apprentissage supervisé** sur des données **tabulées/structurées** ;
  - des algorithmes de **clustering** sur des données **tabulées/structurées** ;
  - des algorithmes d'**apprentissage supervisé profond** sur
    - des **images** ;
    - des **données séquentielles.**

$\vec{x} \in \mathbb{R}^p$

Contexte

- **Objectifs :** Être capable de **choisir** et **mettre en œuvre** :
  - des algorithmes d'**apprentissage supervisé** sur des données **tabulées/structurées** ;
  - des algorithmes de **clustering** sur des données **tabulées/structurées** ;
  - des algorithmes d'**apprentissage supervisé profond** sur
    - des **images** ;
    - des **données séquentielles.**
- **Prérequis :** Cours de science des données et d'optimisation (1A)

Déroulé

Lu 14/09	09h00-12h15	**Introduction et généralités** **Chloé-Agathe Azencott** (CBIO) + Katia Antonenko (CBIO) + Mihaela Dimitrova (STIM)
	13h45-17h00	**Perceptrons multi-couche** **Hsiu-Wen Chang-Joly** (CAOR) + Mihaela Dimitrova (STIM) + Wassil Lebsaira (CAOR)
Ma 15/09	09h00-12h15	**Apprentissage profond, réseaux convolutionnels** **Hsiu-Wen Chang-Joly** (CAOR) + Mihaela Dimitrova (STIM) + Wassil Lebsaira (CAOR)
	13h45-17h00	**Méthodes à noyaux et SVMs** **Chloé-Agathe Azencott** (CBIO) + Mihaela Dimitrova (CMM) + Victor Gertner (CBIO)
Me 16/09	09h00-12h15	**Arbres de décision et méthodes ensemblistes** **Fabien Moutarde** (CAOR) + Stevan Le Stanc (PERSEE) + Alexandre Olech (CAOR)
	13h45-17h00	**Réseaux de neurones récurrents** **Hsiu-Wen Chang-Joly** (CAOR) + Stevan Le Stanc (PERSEE) + Alexandre Olech (CAOR)
Je 17/09	09h00-12h15	**Attention et transformers** **Hsiu-Wen Chang-Joly** (CAOR) + Alexandre Olech (CAOR)
	13h45-17h00	**Clustering** **Chloé-Agathe Azencott** (CBIO) + Katia Antonenko (CMM) + Victor Gertner (CBIO)
Ve 18/09	**09h30**-12h15	**Examen sur table**
	13h45-17h00	**Mini-symposium (présence obligatoire)** **Thibault Faney** (IFPEN) + Chargé-es de TP

Déroulé

- Informations pratiques et ressources sur Moodle (ID 42582)

Déroulé

- Informations pratiques et ressources sur Moodle (ID 42582)
- **TPs :**
  - Notebooks Jupyter/Google Colab
  - La **solution** vous sera fournie après la fin de la séance

Déroulé

- Informations pratiques et ressources sur Moodle (ID 42582)
- **TPs :**
  - Notebooks Jupyter/Google Colab
  - La **solution** vous sera fournie après la fin de la séance
- **Examen** :
  - Vendredi matin
  - Documents **papier** autorisés
  - Annales sur Moodle
- **Mini-symposium** : questions d'actualité en machine learning
  - **Présence obligatoire** pour valider
  - **Thibault Faney (IFPEN)**
  - Katia Antonenko, Mihaela Dimitrova, Victor Gertner, Steven Le Stanc, Alexandre Olech.

Apprentissage supervisé

1. Formalisme

$\mathcal{D} = \{(x_i, y_i)\}_{i=1..n}$ $\quad x_i \in \mathcal{X}$ $\quad y_i$: label of x_i $\quad y_i \in \mathcal{Y}$

- $\mathcal{X} = \mathbb{R}^p$ tabular data
  or $\mathcal{X}$: space of images / text / graphs / molecules ...
- $\mathcal{Y} = \mathbb{R}$ regression
  or $\mathcal{Y} = \{0, 1\}$ or $\{-1, 1\}$ binary classification
  or $\mathcal{Y} = \{1, .., C\}$ multiclass classification

Goal: learn a model $f: \mathcal{X} \to \mathcal{Y}$ $\quad$ f captures the relationship between x_i and y_i

In classification: use a decision function $f: \mathcal{X} \to \mathbb{R}$ then threshold f to obtain a binary label $\Rightarrow$ can be used to predict the label of a new x.

Multiclass classif: learn C binary classifiers / decision functions

2. Minimisation du risque empirique — Empirical Risk Minimization (ERM)

$\mathcal{D} = \{(x_i, y_i)\}_{i=1..n}$ $\quad x_i \in \mathcal{X}$ $\quad y_i \in \mathcal{Y}$

$$\hat{f} \in \underset{f \in \mathcal{H}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i))$$

$\hat{f}$: learned model

① $\mathcal{H}$: hypothesis space

Parametric models : $f_{\vec{\beta}}$ where $\vec{\beta} \in \mathbb{R}^d$ d parameters $\quad \mathcal{X} = \mathbb{R}^p$

$\mathcal{H} : \{f_{\vec{\beta}} ; \vec{\beta} \in \mathbb{R}^d\}$

Linear model : $f_{\vec{\beta}} : \sum_{j=1}^{p} \beta_j x_j + \beta_0$

Polynomial model : $f_{\vec{\beta}} : \beta_0 + \sum_{j=1}^{p} \beta_j x_j + \sum_{j,k} \beta_{jk} x_j x_k + \cdots$

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1..n} \qquad x_i \in \mathcal{X} \qquad y_i \in \mathcal{Y}$$

$$\hat{f} \in \underset{f \in \mathcal{H}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i))$$

learned model

② Loss function (risk) / perte, coût

$$L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$$

$y, f(x) \mapsto$ quantifies the error made when predicting $f(x)$ instead of y.

Regression — Quadratic loss $L(y, f(x)) = (y - f(x))^2$

Classification

- 0/1 loss $\begin{cases} 0 \text{ if } y = f(x) \\ 1 \text{ otherwise} \end{cases}$

binary classif: logistic loss

$$L : \{-1, 1\} \times \mathbb{R} \to \mathbb{R}$$

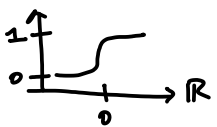
$$y, f(x) \mapsto \log\left(1 + \exp(-y f(x))\right)$$

multi class C decision functions f_k models the probability of belonging to class k

$$f_k : \mathcal{X} \to]0,1[$$

$f : \mathcal{X} \to \mathbb{R} \rightsquigarrow f : \mathcal{X} \to]0,1[$

sigmoid / logistic function $\sigma : u \mapsto \frac{1}{1+e^{-u}}$



Cross-entropy (entropie croisée)

$$L : \{0,1\}^C \times]0,1[^C \longrightarrow \mathbb{R}$$

$$(y_1, y_2, \dots, y_C) \quad (f_1(x), \dots f_C(x)) \mapsto -\sum_{k=1}^{C} y_k \log f_k(x)$$

$y_k = 1$ means label $= k$

③ optimisation the combination of $\mathcal{H}$ and L defines the optimisation problem

3. Généralisation et surapprentissage

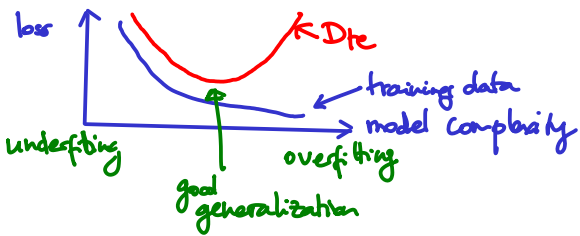
ability to make good predictions on data for which the labels are unknown.

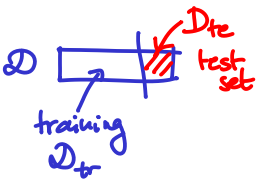
overfitting when learning something that is specific to the training data

more likely for complex models

for parametric models, complexity ≡ number of parameters / magnitude

ex) more parameters than training samples





Transform ERM to avoid overfitting

$\mathcal{D} = \{(x_i, y_i)\}_{i=1..n}$ $\quad x_i \in \mathcal{X}$ $\quad y_i \in \mathcal{Y}$

$$\hat{f} \in \underset{f \in \mathcal{H}}{\operatorname{argmin}} \left[\frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \lambda \Omega(f) \right]$$

learned model

regularisation parameter $\lambda > 0$

regularizer: quantifies the complexity of f

λ close to 0 → close to the ERM
what matters is the error, might overfit

λ large → what matters is the model complexity, might underfit

Parametric models $f_{\vec{\beta}}$:
- ridge regularization $\Omega(\vec{\beta}) = \|\vec{\beta}\|_2^2$ controls the magnitude of model parameters
- lasso $\Omega(\vec{\beta}) = \|\vec{\beta}\|_1$ → sparse model

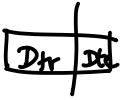
5. Sélection de modèle

$$\hat{f} \in \underset{f \in \mathcal{H}}{\operatorname{argmin}} \left[\frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \lambda \Omega(f) \right]$$

hyperparameter = param of the ML algo and not of the model f

How do I set the value of λ

Ideally



$$\hat{\lambda} \in \underset{\lambda > 0}{\operatorname{argmin}} \frac{1}{n_{te}} \sum_{(x_i, y_i) \in D_{te}} Err(y_i, \hat{f}_{\lambda}(x_i))$$

$\hat{f}_{\lambda}$: solution of the regularized ERM problem on the D_{tr}

How to solve this ?

Grid search : try out several values for λ and pick the one with the smallest error on the test set.

can be $\neq$ from L

Évaluation d'un modèle (classification binaire)

		Classe réelle		
		0	1	
Classe prédite	0	Vrais Négatifs (TN)	Faux Négatifs (FN)	TN+FN
	1	Faux Positifs (FP)	Vrais Positifs (TP)	TP+FP
		TN + FP	TP + FN	

- Sensibilité (sensitivity) = rappel (recall) = taux de vrais positifs = $\frac{TP}{TP + FN}$
- Spécificité (specificity) = taux de vrais négatifs (TN rate) = $\frac{TN}{TN + FP}$
- Précision (precision) = $\frac{TP}{TP + FP}$
- Précision (accuracy) = $\frac{TP + TN}{TP + FP + TN + FN}$
- Score F (F-score) = moyenne harmonique (précision, rappel) = $\frac{2\,TP}{TP + FP + TP + FN}$

Évaluation d'un modèle (classification binaire)

Receiver-Operator Characteristic curve

