

# Le Cadre PAC (Probably Approximately Correct) en Apprentissage Automatique

Bassem Ben Hamed

Control, Optimization and Model Reduction in ML - CIMPA 2025

18 février 2025

# Qu'est-ce que l'Apprentissage PAC ?

- Le cadre PAC formalise les questions fondamentales de l'apprentissage.
- Objectifs : Quelles fonctions peuvent être apprises efficacement ? Combien d'exemples sont nécessaires ?
- Hypothèses : Les exemples sont tirés indépendamment selon une distribution  $D$  inconnue mais fixe.
- But : Trouver une hypothèse  $h_S$  telle que  $R(h_S) \leq \epsilon$  avec une probabilité au moins  $1 - \delta$ .

## Définitions et Notations

- $X$  : ensemble des exemples ou instances (espace d'entrée).
- $Y$  : ensemble des étiquettes possibles. Ici,  $Y = \{0, 1\}$  (classification binaire).
- Un **concept**  $c : X \rightarrow Y$  est une fonction qui associe chaque exemple à une étiquette.
- Un **concept classe**  $C$  est un ensemble de concepts à apprendre.
- Exemples : un concept peut représenter un triangle dans un plan.

# Formulation du Problème d'Apprentissage

- Les exemples  $(x_1, \dots, x_m)$  sont tirés i.i.d. selon une distribution inconnue  $D$ .
- Le modèle d'apprentissage considère un ensemble de concepts possibles  $H$ , appelé **ensemble d'hypothèses**.
- L'objectif est de choisir une hypothèse  $h_S \in H$  qui minimise l'erreur de généralisation.
- L'**erreur de généralisation**  $R(h)$  d'une hypothèse  $h \in H$  est définie comme le risque de mal classifier un nouvel exemple tiré selon  $D$ .

## Definition

Soient une hypothèse  $h \in H$ , un concept cible  $c \in C$  et une distribution sous-jacente  $D$ . L'**erreur de généralisation** ou **risque** de  $h$  est définie par :

$$R(h) = P_{x \sim D}[h(x) \neq c(x)] = \mathbb{E}_{x \sim D} [\mathbf{1}_{h(x) \neq c(x)}] \quad (1)$$

où  $\mathbf{1}_{h(x) \neq c(x)}$  est la fonction indicatrice de l'événement  $h(x) \neq c(x)$ .

- L'erreur de généralisation est inaccessible au modèle d'apprentissage car  $D$  et  $c$  sont inconnus.
- Cependant, l'erreur empirique sur un échantillon étiqueté  $S$  peut être mesurée.

## Definition

Soient une hypothèse  $h \in H$ , un concept cible  $c \in C$  et un échantillon  $S = (x_1, \dots, x_m)$ . L'**erreur empirique** ou **risque empirique** de  $h$  est définie par :

$$R_S(h) = \frac{1}{m} \sum_{i=1}^m \mathbf{1}_{h(x_i) \neq c(x_i)} \quad (2)$$

- L'erreur empirique de  $h \in H$  est la moyenne des erreurs sur l'échantillon  $S$ .
- Contrairement à l'erreur de généralisation, elle est observable directement.

# Lien entre Erreur Empirique et Erreur de Généralisation

- Sous l'hypothèse que  $S$  est tiré i.i.d. selon  $D$ , l'espérance de l'erreur empirique est égale à l'erreur de généralisation :

$$\mathbb{E}_{S \sim D^m}[R_S(h)] = R(h) \quad (3)$$

- Cette relation découle de la linéarité de l'espérance et du fait que  $S$  est un échantillon i.i.d. En effet,

$$\begin{aligned} \mathbb{E}_{S \sim D^m}[\hat{R}_S(h)] &= \frac{1}{m} \sum_{i=1}^m \mathbb{E}_{S \sim D^m}[\mathbf{1}_{h(x_i) \neq c(x_i)}] \\ &= \frac{1}{m} \sum_{i=1}^m \mathbb{E}_{S \sim D^m}[\mathbf{1}_{h(x) \neq c(x)}] \end{aligned}$$

pour tout  $x$  dans l'échantillon  $S$ .

- D'où

$$\mathbb{E}_{S \sim D^m}[\hat{R}_S(h)] = \mathbb{E}_{S \sim D^m}[1_{h(x) \neq c(x)}] = \mathbb{E}_{x \sim D}[1_{h(x) \neq c(x)}] = R(h)$$

- Cela justifie l'utilisation de l'erreur empirique comme une estimation de l'erreur de généralisation.

- Le cadre d'apprentissage **Probably Approximately Correct (PAC)** vise à quantifier la performance d'un algorithme d'apprentissage.
- Soit  $n$  un entier représentant le coût de la représentation d'un élément  $x \in X$ , tel que ce coût est au plus  $O(n)$ .
- On note  $\text{size}(c)$  le coût maximal de la représentation computationnelle d'un concept  $c \in C$ .
- Par exemple, si  $x$  est un vecteur de  $\mathbb{R}^n$ , son coût de représentation est en  $O(n)$ .
- Soit  $h_S$  l'hypothèse retournée par un algorithme  $A$  après réception d'un échantillon étiqueté  $S$ .
- Pour simplifier la notation, la dépendance de  $h_S$  à  $A$  n'est pas explicitement indiquée.

## Definition

Une classe de concepts  $C$  est dite **PAC-apprenable** s'il existe un algorithme  $A$  et une fonction polynomiale  $\text{poly}(\cdot, \cdot, \cdot, \cdot)$  tels que :

- Pour tout  $\varepsilon > 0$  et  $\delta > 0$ ,
- Pour toute distribution  $D$  sur  $X$ ,
- Pour tout concept cible  $c \in C$ ,
- Si la taille de l'échantillon  $m \geq \text{poly}(1/\varepsilon, 1/\delta, n, \text{size}(c))$ , alors

$$P_{S \sim D^m}[R(h_S) \leq \varepsilon] \geq 1 - \delta \quad (4)$$

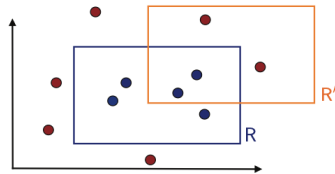
Si de plus,  $A$  s'exécute en temps polynomial en  $1/\varepsilon, 1/\delta, n, \text{size}(c)$ , alors  $C$  est dite **PAC-apprenable efficacement** et  $A$  est un **algorithme d'apprentissage PAC** pour  $C$ .

- Une classe de concepts  $C$  est **PAC-apprenable** si l'hypothèse retournée par l'algorithme après avoir observé un nombre de points polynomial en  $1/\varepsilon$  et  $1/\delta$  est **approximativement correcte**.
- Cela signifie que l'erreur est au plus  $\varepsilon$  avec une probabilité d'au moins  $1 - \delta$ .
- Le paramètre  $\delta > 0$  définit la **confiance**  $1 - \delta$ , et  $\varepsilon > 0$  définit la **précision**  $1 - \varepsilon$ .
- Si l'algorithme s'exécute en temps polynomial en  $1/\varepsilon$  et  $1/\delta$ , alors la taille de l'échantillon  $m$  doit aussi être polynomial.

- Le cadre PAC est **indépendant de la distribution** : aucune hypothèse spécifique sur  $D$ .
- L'échantillon d'entraînement et les exemples de test proviennent de la **même distribution**  $D$ .
- Le cadre PAC traite la **capacité d'apprentissage d'une classe de concepts**  $C$ , et non d'un concept particulier.
- La classe de concepts  $C$  est connue de l'algorithme, mais le concept cible  $c \in C$  est inconnu.
- Dans certains cas, on omet la dépendance polynomiale à  $n$  et  $\text{size}(c)$  pour se concentrer sur la complexité en échantillons.

# Exemple : Apprentissage des rectangles alignés sur les axes

- **Espace des instances** :  $X = \mathbb{R}^2$   
(points dans le plan)
- **Classe de concepts**  $C$  : ensemble des rectangles alignés sur les axes dans  $\mathbb{R}^2$
- **Objectif** : déterminer avec une faible erreur un rectangle cible en utilisant un échantillon d'apprentissage étiqueté
- **Résultat** : Nous montrerons que cette classe de concepts est PAC-apprenable



**Figure:** Concept cible  $R$  et hypothèse candidate  $R'$ . Les cercles représentent les exemples d'apprentissage. Un cercle bleu indique un point étiqueté positivement (1), car il est situé à l'intérieur du rectangle  $R$ . Les autres points sont représentés en rouge et étiquetés négativement (0).

## Analyse des régions d'erreur :

- **Faux négatifs** : points dans  $R$  mais hors de  $R'$  (étiquetés 0 par  $R'$ , mais vraiment 1)
- **Faux positifs** : points dans  $R'$  mais hors de  $R$  (étiquetés 1 par  $R'$ , mais vraiment 0)

## Algorithme PAC-learning :

- Construire le plus petit rectangle  $R' = R_S$  contenant tous les points positifs
- Propriété :  $R_S$  n'a pas de faux positifs
- La région d'erreur de  $R_S$  est incluse dans  $R$

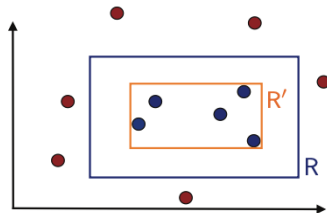


Figure: Illustration de l'hypothèse  $R' = R_S$  retournée par l'algorithme

# Analyse probabiliste de l'apprentissage des rectangles

## Hypothèses et définitions :

- Soit  $R \in \mathcal{C}$  un concept cible et  $\varepsilon > 0$
- $P[R]$  : probabilité qu'un point tiré selon  $D$  tombe dans  $R$
- On suppose  $P[R] > \varepsilon$  (sinon l'erreur est déjà  $\leq \varepsilon$ )

## Construction des régions :

- Définition de 4 régions rectangulaires :  $r_1, r_2, r_3, r_4$
- Chaque région a une probabilité  $\geq \varepsilon/4$
- Obtenues en réduisant  $R$  tout en maintenant  $P[r_j] \geq \varepsilon/4$

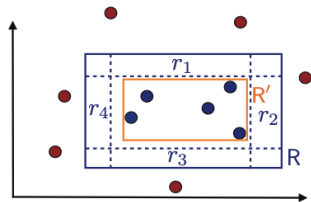


Figure: Illustration des régions  $r_1, r_2, r_3, r_4$

## Définition formelle des régions :

- $R = [l, r] \times [b, t]$
- $r_4 = [l, s_4] \times [b, t]$  où  $s_4 = \inf s : P[[l, s][b, t]] \geq \varepsilon/4$
- $\bar{r}_4 = [l, s_4[ \times [b, t]$  obtenu à partir de  $r_4$  en excluant le côté droit, on a  $P[\bar{r}_4] \leq \varepsilon/4$

## Propriété clé :

- Si  $R_S$  intersecte tous les  $r_i$  :
  - Un côté dans chaque région
  - Erreur  $\leq \varepsilon$  (union des  $\bar{r}_i$ )
- Par contraposition : si  $R(R_S) > \varepsilon$ , alors  $R_S$  manque au moins une région  $r_i$

Ainsi,

$$\begin{aligned} P_{S \sim D^m} [R(R_S) > \epsilon] &\leq P_{S \sim D^m} \left[ \bigcup_{i=1}^d \{R_S \cap r_i = \emptyset\} \right] \\ &\leq \sum_{i=1}^4 P_{S \sim D^m} [\{R_S \cap r_i = \emptyset\}] \\ &\leq 4(1 - \epsilon)^m \quad \text{car } (P[\tilde{r}_4] \geq \epsilon/4) \\ &< 4 \exp\left(-\frac{m\epsilon}{4}\right) \end{aligned} \tag{5}$$

où, on a utilisé l'inégalité générale  $1 - x \leq \exp(-x)$  valable pour tout  $x \in \mathbb{R}$ . Pour tout  $\delta > 0$ , afin d'assurer que  $P_{S \sim D^m} [R(R_S) > \epsilon] \leq \delta$ , on peut imposer

$$4 \exp\left(-\frac{m\epsilon}{4}\right) \leq \delta \Leftrightarrow m \geq \frac{4}{\epsilon} \log \frac{4}{\delta} \tag{6}$$

- Donc, pour tout  $\epsilon > 0$  et  $\delta > 0$ , si la taille de l'échantillon  $m$  est supérieure à  $\frac{4}{\epsilon} \log \frac{4}{\delta}$ , alors  $P_{S \sim D^m}[R(R_S) > \epsilon] \leq \delta$ .
- En outre, le coût computationnel de la représentation des points dans  $\mathbb{R}^2$  et des rectangles alignés sur les axes, qui peut être défini par leurs quatre coins, est constant.
- Cela prouve que la classe de concepts des rectangles alignés sur les axes est PAC-apprenante et que la complexité d'échantillonnage de l'apprentissage PAC des rectangles alignés sur les axes est de l'ordre  $O\left(\frac{1}{\epsilon} \log \frac{1}{\delta}\right)$ .

- Une manière équivalente de présenter des résultats de complexité d'échantillonnage comme (6), est de donner une borne de généralisation. Une borne de généralisation indique qu'avec une probabilité d'au moins  $1 - \delta$ ,  $R(R_S)$  est limité par une certaine quantité qui dépend de la taille de l'échantillon  $m$  et de  $\delta$ .
- Pour obtenir cela, il suffit de fixer  $\delta$  égal à la borne supérieure dérivée dans (5), où  $\delta = 4 \exp(-m/4)$ , et de résoudre pour  $\varepsilon$ . Cela donne avec une probabilité d'au moins  $1 - \delta$  que l'erreur de l'algorithme est bornée comme suit :

$$R(R_S) \leq \frac{4}{m} \log \frac{4}{\delta}. \quad (7)$$

- Les ensembles d'hypothèses en apprentissage automatique sont souvent infinis.
- Les bornes classiques de complexité d'échantillonnage sont inadéquates pour les ensembles infinis.
- La PAC-apprenabilité de certaines classes infinies (ex : rectangles alignés sur les axes) suggère que la généralisation est possible.
- Approche : Réduire les cas infinis à des cas finis en utilisant des mesures de complexité.

- **Complexité de Rademacher** : Mesure la capacité d'un ensemble d'hypothèses via l'inégalité de McDiarmid.
- La complexité de Rademacher empirique est NP-difficile à calculer pour certains ensembles.
- Mesures combinatoires alternatives :
  - **Fonction de croissance** : Borne la complexité de Rademacher.
  - **Dimension VC (Vapnik-Chervonenkis)** : Plus facile à calculer, mène à des bornes de généralisation.
- Applications : Bornes de généralisation pour les cas réalisables et non réalisables.