Interactions entre l'apprentissage statistique et la théorie des modèles

Séminaire d'Algèbre et Logique

Damien Galant

UMONS

18 Décembre 2020

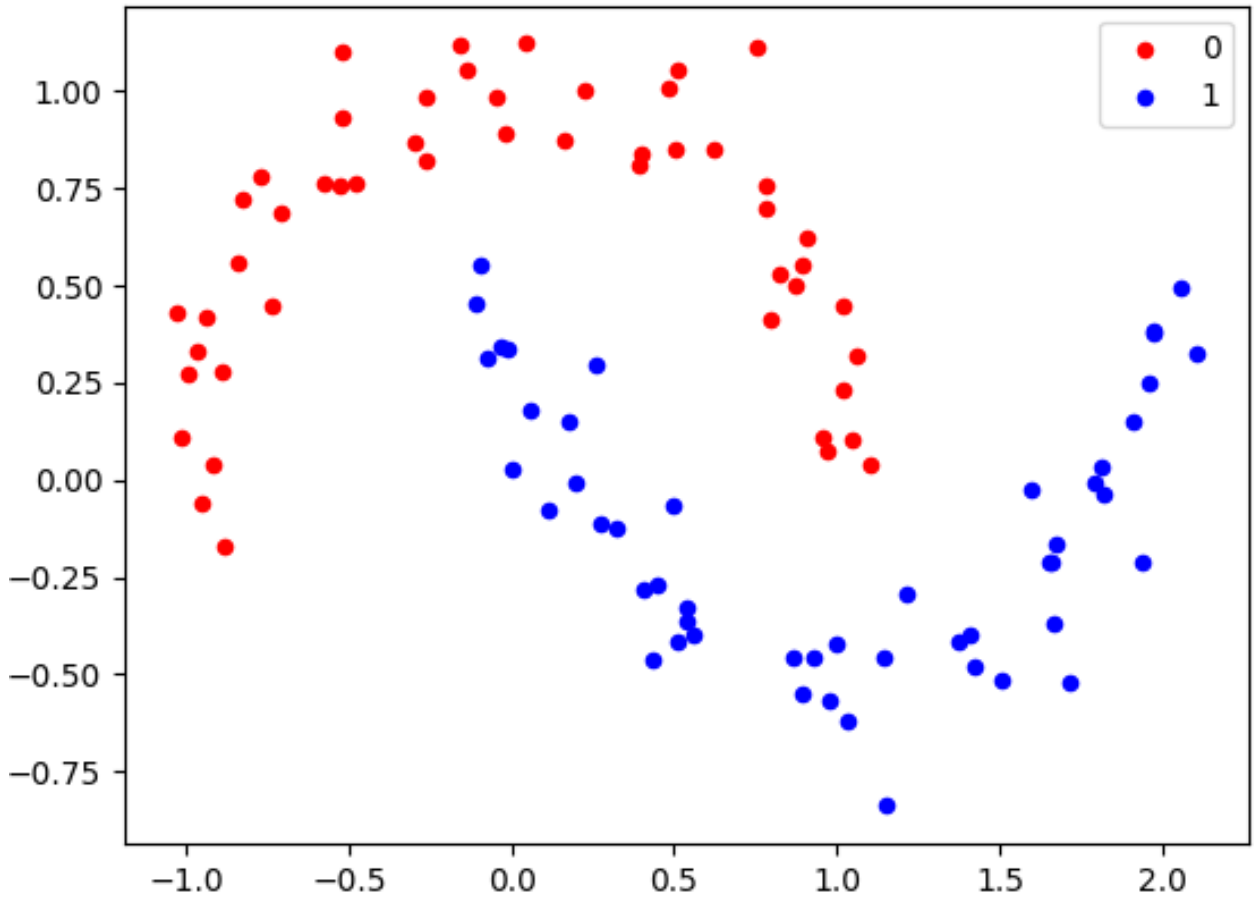
UMONS
University of Mons

Apprentissage statistique : contexte

On s'intéresse au problème de **classification binaire** :

- On se donne un espace $\mathcal{X}$ (encodant des images, par exemple) et un ensemble $\mathcal{Y}$ de deux étiquettes (labels), par exemple $\{0, 1\}$, et une fonction $f : \mathcal{X} \to \mathcal{Y}$.
- On va recevoir un « training set » $(x_1, f(x_1)), \cdots (x_k, f(x_k))$ où $x_1, \cdots, x_k \in \mathcal{X}$. On aimerait approximer la fonction f au mieux possible.
- Le point de vue de l'apprentissage statistique est de considérer que les données X sont issues d'un tirage selon une loi de probabilité $\mathbb{P}$.
- Une **fonction de prédiction** (ou « concept ») est une application mesurable $h : \mathcal{X} \to \mathcal{Y}$.

Un exemple de training set



Construit à partir de https://machinelearningmastery.com/generate-test-datasets-python-scikit-learn/

Méthodes d'apprentissage

- Soit $\mathcal{H}$ une classe de fonctions de prédiction, parfois appelé « classe d'hypothèses ».
- À un tuple d'exemples $(x_1, f(x_1)), \cdots, (x_k, f(x_k)) \in \mathcal{X} \times \{0, 1\}$ correspond un unique élément de

$$\mathcal{H}_{fin} := \{f_{|\mathcal{Z}} \mid \mathcal{Z} \subseteq \mathcal{X}, \mathcal{Z} \text{ fini}, f \in \mathcal{H}\}$$

- Une **méthode d'apprentissage** va associer à un ensemble fini de données d'entraînement une fonction de prédiction. Il s'agit donc d'une application

$$G : \mathcal{H}_{fin} \to \mathcal{F}(\mathcal{X}, \mathcal{Y})$$

où $\mathcal{F}(\mathcal{X}, \mathcal{Y})$ désigne l'ensemble des fonctions de $\mathcal{X}$ dans $\mathcal{Y}$.

Fonctions de prédiction ε-approximativement correctes

- Étant donnée une fonction de prédiction h, sa **probabilité de mauvaise classification** est donnée par

$$L_{\mathbb{P}}(h) := \mathbb{P}[f(X) \neq h(X)]$$

où $X \in \mathcal{X}$ provient d'un tirage selon la loi $\mathbb{P}$.

- On a beau avoir autant de données qu'on veut, il est en général impossible d'être certain d'avoir déterminé f. Il faut donc se permettre une marge d'erreur, ce qui mène à la définition suivante

Definition (Fonction de prédiction ε-approximativement correcte)

Une fonction de prédiction h est dite ε-approximativement correcte si

$$L_{\mathbb{P}}(h) \leq \varepsilon$$

Erreur de généralisation

Définition (Erreur de généralisation)

Étant donné un training set $(a_1, f(a_1)), \cdots, (a_n, f(a_n))$ *correspondant au tuple* $\bar{a} = (a_1, \cdots, a_n) \in \mathcal{X}^n$, ***l'erreur de généralisation*** *associée est définie par*

$$L_{\mathbb{P}}(G(f_{|\bar{a}}))$$

Autrement dit, il s'agit de la probabilité de mauvaise classification de la fonction de prédiction obtenue à partir des données en appliquant la méthode G.

PAC learning

La notion de « probably approximately correct learning » (PAC learning) a été introduite par Leslie Valiant en 1984 dans « A Theory of the Learnable » [Val84].

Definition (PAC-learnability)

On dit que $\mathcal{H}$ est PAC-learnable si il existe une méthode d'apprentissage $G : \mathcal{H}_{fin} \to \mathcal{F}(\mathcal{X}, \mathcal{Y})$ telle que pour tous $\varepsilon > 0$ et tous $\delta > 0$, il existe $N_{\varepsilon,\delta} \in \mathbb{N}$ tels que pour tout $f \in \mathcal{H}$ et toute distribution de probabilité $\mathbb{P}$ sur $\mathcal{X}$ rendant toutes les fonctions de $\mathcal{H}$ mesurables, on a

$$\mathbb{P}\left(\{\bar{a} \in \mathcal{X}^{N_{\varepsilon,\delta}} \mid \text{l'erreur de généralisation de } \bar{a} \text{ est supérieure à } \varepsilon\}\right) < \delta$$

Quelques commentaires

- Nous avons fait quelques hypothèses simplificatrices. En particulier, nous avons supposé qu'il existait une vraie fonction cible $f : \mathcal{X} \to \mathcal{Y}$. Ce n'est en général pas le cas car il peut y avoir un peu de bruit dans les données, ou de l'ambiguïté (quand est-ce qu'une image représente « un paysage » ?)
- Dans la notion de « PAC-learnability », on suppose que $f \in \mathcal{H}$. En fait la plupart des considérations font sens dans le cas où on approxime des fonctions quelconques par une classe de fonctions $\mathcal{H}$ souvent décrite par un certain nombre de paramètres.
  Cependant, cela ne change pas fortement pas l'analyse, voir par exemple [Gir13].
- Un avantage considérable de la notion de PAC-learnability est d'être « distribution free » (par définition). C'est utile car en pratique il peut être difficile de mettre des hypothèses a priori sur la distribution des données. On verra une caractérisation combinatoire de la PAC-learnability dans la suite de l'exposé.

Minimisation du risque empirique

Source principale : Christophe Giraud. *Fondements mathématiques de l'apprentissage statistique*. 2013. URL : http://www.math.polytechnique.fr/xups/xups13-02.pdf.

- On n'a pas accès à la probabilité de mauvaise classification $L(h)$ car la loi $\mathbb{P}$ est inconnue.
- On peut par contre calculer la probabilité **empirique** de mauvaise classification

$$\hat{L}_n(h) := \frac{1}{n} \sum_{1 \leq i \leq n} \mathbb{1}_{f(X_i) \neq h(X_i)} = \hat{\mathbb{P}}_n(f(X) \neq h(X))$$

où $\hat{\mathbb{P}}_n := \frac{1}{n} \sum_{1 \leq i \leq n} \delta_{X_i}$.

- À une classe d'hypothèses $\mathcal{H}$ (*ne contenant pas nécessairement* f), on peut associer un **classificateur de minimisation du risque empirique**

$$\hat{h}_{\mathcal{H}} \in \arg\min_{h \in \mathcal{H}} \hat{L}_n(h)$$

Coefficient d'éclatement

- Ce n'est pas tellement la taille de la classe $\mathcal{H}$ qui compte, mais plutôt sa flexibilité en termes de classification.
- Cette flexibilité de classification est quantifiée par les **coefficients d'éclatement**

$$\mathbb{S}_n(\mathcal{H}) := \max_{(x_1,\cdots,x_n)\in\mathcal{X}^n} \operatorname{card}\{(h(x_1),\cdots,h(x_n)) \mid h \in \mathcal{H}\}$$

qui donnent le nombre maximum d'étiquetages différents de n points qui peuvent être produits par les classifieurs dans $\mathcal{H}$.

Contrôle de l'erreur stochastique

Voici un exemple de résultat fournissant une borne supérieure de l'erreur stochastique et un intervalle de confiance pour la probabilité de mauvaise classification $L(\hat{h}_{\mathcal{H}})$ en terme du coefficient d'éclatement.

Théorème (Contrôle de l'erreur stochastique)

Pour tout $t > 0$, on a avec probabilité au moins $1 - e^{-t}$

$$L(\hat{h}_{\mathcal{H}}) - \min_{h \in \mathcal{H}} L(h) \leq 4\sqrt{\frac{2\log(2\mathbb{S}_{\mathcal{H}}(n))}{n}} + \sqrt{\frac{2t}{n}}$$

La preuve est trop longue que pour être présentée ici. Elle fait notamment intervenir une inégalité de concentration due à McDiarmid ainsi qu'un joli argument de symétrisation.
On notera que la théorie de l'apprentissage statistique requiert l'utilisation d'inégalités de concentration **non asymptotiques**.

Définition

La définition suivante est due à Vapnik et Chervonenkis en 1971.

Définition (VC-dimension)

On appelle ***VC-dimension*** *de $\mathcal{H}$ l'entier défini par*

$$\mathrm{VCdim}(\mathcal{H}) := \sup\left\{d \in \mathbb{N} \mid \mathbb{S}_d(\mathcal{H}) = 2^d\right\} \in \mathbb{N} \cup \{+\infty\}$$

en prenant la convention $\mathbb{S}_0(\mathcal{H}) = 1$.

La VC-dimension est égale au nombre maximum de points de $\mathcal{X}$ qui peuvent être classifiés de toutes les façons possibles en utilisant les classifieurs de $\mathcal{H}$.

Exemple : classificateurs affins dans le plan

Avec la tablette graphique !

Exemple : rectangles parallèles aux axes dans le plan

Avec la tablette graphique !

Lemme de Vapnik-Chervonenkis-Perles-Sauer-Shelah

La VC-dimension contrôle les coefficients d'éclatement grâce au résultat combinatoire remarquable suivant :

Théorème (Vapnik-Chervonenkis (1971) ; Perles-Shelah ; Sauer (1972))

Soit $\mathcal{H}$ une classe d'hypothèses telle que $d := \mathrm{VCdim}(\mathcal{H}) < \infty$.
Pour tout $n \in \mathbb{N}$, on a

$$\mathbb{S}_n(\mathcal{H}) \leq \sum_{0 \leq i \leq d} \binom{n}{i} \leq (n+1)^d$$

avec la convention $\binom{n}{i} = 0$ si $n < i$.

Remarque : la deuxième inégalité est élémentaire et sert à obtenir une majoration plus maniable. La force du lemme réside en la première inégalité.

Contexte historique

Ce lemme est au confluent de l'apprentissage statistique, de la combinatoire et de la théorie des modèles :

- Vladimir Vapnik et Alexey Chervonenkis sont des pionniers de la théorie de l'apprentissage statistique. Ils ont introduit la VC-dimension pour obtenir des bornes de déviation uniformes sur des classes de fonctions : « Vladimir Vapnik et Alexey Chervonenkis. « The uniform convergence of frequencies of the appearance of events to their probabilities. (Russian) ». In : *Teor. Veroyatnost. i Primenen.* 16 (1971), p. 264-279 »
- Les motivations de Saharon Shelah provenaient de la théorie des modèles : « Saharon Shelah. « A combinatorial problem ; stability and order for models and theories in infinitary languages. ». In : *Pacific J. Math.* 41.1 (1972), p. 247-261 »
- La motivation de Norbert Sauer était combinatoire et répond à une question que Paul Erdös lui avait communiqué, voir : « N Sauer. « On the density of families of sets ». In : *Journal of Combinatorial Theory, Series A* 13.1 (1972), p. 145-147. ISSN : 0097-3165 »

Retour au PAC-learning

Théorème (Équivalence entre PAC-learnability et VC-dimension finie)

Soit $\mathcal{H}$ une classe d'hypothèses sur $\mathcal{X}$. Alors,

$\mathcal{H}$ a une VC-dimension finie si et seulement si $\mathcal{H}$ est PAC-learnable

Si c'est le cas, on a une borne explicite sur $N_{\varepsilon,\delta}$ donnée par

$$N_{\varepsilon,\delta} \leq \max\left\{\frac{4}{\varepsilon}\log_2\left(\frac{2}{\delta}\right), \frac{8\mathrm{VCdim}(\mathcal{H})}{\varepsilon}\log_2\left(\frac{13}{\varepsilon}\right)\right\}$$

Lien avec la théorie des modèles

- On se donne une structure $\mathcal{A}$ (de domaine A) sur un langage $\mathcal{L}$ et une formule « partitionnée » du premier ordre (où les r premières variables joueront un rôle différent des s autres)

$$\phi(\bar{x}, \bar{y}) := \phi(x_1, \cdots, x_r, y_1, \cdots, y_s)$$

- Étant donné $\bar{b} \in A^s$, on utilise la notation

$$\phi(A^r, \bar{b}) := \{\bar{c} \in A^r \mid \mathcal{A} \models \phi(\bar{c}, \bar{b})\}$$

pour désigner un ensemble **définissable avec paramètres**.

- On définit la classe d'ensembles définissables avec s paramètres associée à ϕ par

$$\mathcal{H}_\phi := \{\phi(A^r, \bar{b}) \mid \bar{b} \in A^s\}$$

Lien avec la théorie des modèles : théories IP et NIP

- La propriété d'indépendance (« independance property ; **IP** ») a été introduite par Shelah en 1971. Elle peut se définir au niveau des formules. Si ϕ ne satisfait pas IP, on dit que ϕ est NIP.

Théorème (Michael C. Laskowski 1990, voir [Las92])

ϕ est NIP si et seulement si $\mathcal{H}_\phi$ a une VC-dimension finie

- Il s'avère que la propriété « posséder une formule IP » est **structurelle**. Une structure dont toutes les formules sont NIP est dite NIP.
- **Une structure est NIP si et seulement si toutes ses classes d'ensembles définissables avec paramètres sont de VC-dimension finie.**
- Les structures NIP ont de nombreuses propriétés, voir par exemple [Adl08].

Une classe de VC-dimension infinie

- Étant donné $\alpha \in \mathbb{R}$, on définit $h_\alpha : \mathbb{R} \to \{0, 1\}$ par

$$h_\alpha(x) := \begin{cases} 0 & \text{si } \sin(\alpha x) \geq 0 \\ 1 & \text{sinon} \end{cases}$$

- La classe

$$\mathcal{H}_{\sin} := \{h_\alpha \mid \alpha \in \mathbb{R}\}$$

  a une VC-dimension infinie (ici $\mathcal{X} = \mathbb{R}$, $\mathcal{Y} = \{0, 1\}$). Pourtant, elle est paramétrisée par un seul paramètre réel !
- Pour aller plus loin avec cet exemple, voir [HRN15].

Petite digression sur le nombre de paramètres

- Dans beaucoup de situations (par exemple physiques), il paraît pertinent de compter le « nombre minimal de paramètres » pour quantifier la complexité d'un objet.
- L'exemple des classificateurs en « $\sin(\alpha x)$ » montre que dans le problème de classification le nombre de paramètres réels de la famille rend mal compte de la complexité : ici il n'y a qu'*un* paramètre, α.
- Une situation bien connue où le nombre minimal de paramètres est bien défini et donne lieu à une notion intéressante est bien sûr l'algèbre linéaire. En effet, si E est un $\mathbb{K}$-espace vectoriel, alors

$$\dim E = \min\left\{n \in \mathbb{N} \mid \exists v_1, \cdots, v_n \in E, E = \operatorname{span}(v_1, \cdots, v_n)\right\}$$

où

$$\operatorname{span}(v_1, \cdots, v_n) := \left\{ \sum_{1 \leq i \leq n} \lambda_i v_i \mid \lambda_1, \cdots, \lambda_n \in \mathbb{K} \right\}$$

(avec la convention $\operatorname{span}() = \{0\}$).

Petite digression sur le nombre de paramètres (suite)

- Même dans des contextes non-linéaires, l'intuition du « nombre de paramètres » correspond souvent à la notion rigoureuse de dimension d'un espace vectoriel.
- Par exemple, on retrouve la dimension d'une variété différentielle comme dimension de son espace tangent.
- Ainsi, la sphère

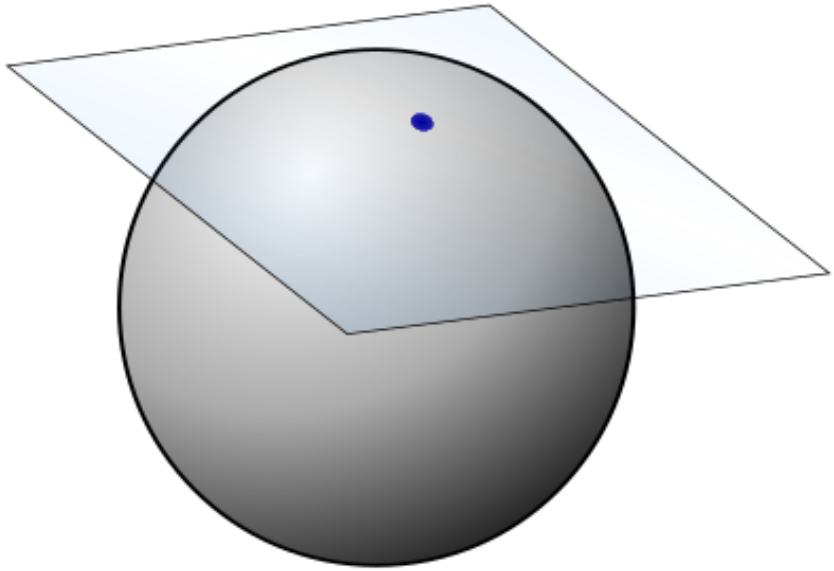
$$\mathbb{S}_2 := \left\{(x_1, x_2, x_3) \in \mathbb{R}^3 \mid x_1^2 + x_2^2 + x_3^2 = 1\right\}$$

  est de dimension 3 car son espace tangent en tout point est un espace vectoriel de dimension 2.
- Remarque : malgré tout, il est possible de donner un sens à la dimension d'une variété topologique sans passer par la dimension des espaces vectoriels. C'est cependant nettement plus compliqué, et requiert des résultats plus profonds comme le théorème d'invariance du domaine de Brouwer, voir [Tao11].

Illustration dans le cas d'une sphère

Deux façons de penser à la dimension 2 :

- On peut paramétriser (localement) la sphère avec deux coordonnées, disons la latitude et la longitude ;
- L'espace tangent en la sphère en tout point est de dimension 2, au sens des espaces vectoriels.



Source : https://fr.m.wikipedia.org/wiki/Fichier:Image_Tangent-plane.svg

Apprentissage online (online learning)

Source principale : le chapitre 21 de Shai Shalev-Shwartz et Shai Ben-David. *Understanding Machine Learning : From Theory to Algorithms*. Cambridge University Press, 2014 [SB14] (version en ligne).

- On se donne une classe d'hypothèses $\mathcal{H}$ sur un ensemble $\mathcal{X}$.
- On va devoir faire des prédictions pendant T tours de suite. L'horizon T est fixé et connu à l'avance. Pendant ces T tours numérotés de 1 à T, on recevra une entrée x_t et on devra faire une prédiction $\hat{y}_t$ basée sur les couples d'exemples reçus précédemment.
- Après avoir fourni sa prédiction $\hat{y}_t$, on reçoit en retour le vrai label $y_t = f(x_t)$.
- On suppose pour simplifier que f appartient à $\mathcal{H}$. Il n'y a pas d'hypothèses particulières sur la façon dont les données x_t apparaissent. Étant donné un algorithme $\mathcal{A}$ générant les $\hat{y}_t$, on cherche à borner

$$M(\mathcal{A}) := \max_{f \in \mathcal{H}} \max_{\bar{x}=(x_1,\cdots,x_T)} \sum_{1 \leq t \leq T} |\hat{y}_t - f(x_t)|$$

Comparaison avec l'apprentissage PAC

- Attention : les objectifs en online learning sont différents !
- Ici, on veut fournir de bonnes prédictions **au fur et à mesure qu'om reçoit des données**, alors qu'en PAC-learning on veut construire un prédicteur pour les prochains inputs à partir du training set.

Un premier algorithme simple

On supposera ici pour simplifier que $\mathcal{H}$ est **fini**.
Un premier algorithme consiste à considérer l'ensemble V_t des algorithmes consistants avec toutes les observations avant le temps t.

Algorithme 1 « Consistant » [SB14, p289]

Entrée: Une classe d'hypothèses finie $\mathcal{H}$.

1: On initialise $V_1 := \mathcal{H}$.
2: **pour** $t \leftarrow 1$ à T **faire**
3: On reçoit l'input x_t ;
4: On choisit n'importe quelle fonction de prédiction $h \in V_t$;
5: On utilise cette fonction de prédiction et on renvoie $\hat{y}_t = h(x_t)$;
6: On reçoit en retour le vrai label $y_t = f(x_t)$;
7: On met à jour l'ensemble des fonctions de prédiction possibles par $V_{t+1} := \{h \in V_t \mid h(X_t) = y_t\}$
8: **fin pour**

Une astuce cruciale

- Étant donné l'ensemble des fonctions de prédiction possibles au temps t, on définit

$$V_t^{(r)} := \{h \in V_t \mid h(x_t) = r\}$$

pour $r \in \{0, 1\}$.

- Au lieu de choisir n'importe quelle fonction de prédiction $h \in V_t$ (voir la ligne 4 de l'algorithme précédent), il vaut mieux prédire en utilisant

$$\hat{y}_t \in \arg\max_{r \in \{0,1\}} |V_t^{(r)}|$$

- On observe ainsi une différence cruciale entre le PAC-learning et le online learning : en online learning toutes les hypothèses qui minimisent les erreurs sur les données de test ne se valent pas.

Un meilleur algorithme

Algorithme 2 « Halving » [SB14, p289]

Entrée: Une classe d'hypothèses finie $\mathcal{H}$.

1: On initialise $V_1 := \mathcal{H}$.
2: **pour** $t \leftarrow 1$ à T **faire**
3: On reçoit l'input x_t ;
4: On renvoie $\hat{y}_t \in \arg\max_{r \in \{0,1\}} |V_t^{(r)}|$;
5: On reçoit en retour le vrai label $y_t = f(x_t)$;
6: On met à jour l'ensemble des fonctions de prédiction possibles par $V_{t+1} := \{h \in V_t \mid h(X_t) = y_t\}$
7: **fin pour**

Environnement antagoniste

Explication en direct avec la tablette graphique !

Arbres « pulvérisés »

Source principale : Shai Ben-David, D. Pál et S. Shalev-Shwartz. « Agnostic Online Learning ». In : *COLT*. 2009 [BPS09].

- On considère des arbres binaires parfaits de hauteur d, avec une notion de « fils gauche » et de « fils droit », et dont les nœuds internes (ceux qui ne sont pas des feuilles) sont étiquetés par des éléments de $\mathcal{X}$.
- Tout chemin de la racine à une feuille peut être décrit par une suite $(x_1, y_1), \cdots, (x_d, y_d)$ où les x_i sont les étiquettes des éléments de l'arbre et où les y_i indiquent si on emprunte le fils gauche ou le fils droit.

Définition (Arbre pulvérisé par une classe d'hypothèses)

Un arbre étiqueté par des éléments de $\mathcal{X}$ est dit pulvérisé par une classe $\mathcal{H}$ si pour chaque chemin $(x_1, y_1), \cdots, (x_d, y_d)$, il existe $h \in \mathcal{H}$ tel que pour tout $i \leq d$, $h(x_i) = y_i$ pour tout i.

Dimension de Littlestone

La dimension de Littlestone a été introduite en 1988 dans [Lit88].

Définition (Dimension de Littlestone)

On définit la définition de Littlestone $\mathrm{Ldim}(\mathcal{H})$ *d'une classe d'hypothèses* $\mathcal{H}$ *non-vide comme étant le plus grand naturel* d *tel qu'il existe un arbre binaire complet de profondeur* d *pulvérisé par* $\mathcal{H}$*.*

Exemple de dimension de Littlestone

On commentera un exemple de la section « Background : Littlestone's Dimension and the Realizable Case » de [BPS09] avec la tablette graphique.

Comparaison avec la VC-dimension

Proposition

Soit $\mathcal{H}$ une classe d'hypothèses non-vide. Alors

$$\mathrm{VCdim}(\mathcal{H}) \leq \mathrm{Ldim}(\mathcal{H}) \leq \log_2(|\mathcal{H}|)$$

Démonstration.

La première inégalité vient du fait que la pulvérisation de tout d-uple au sens de la VC-dimension implique l'existence d'un arbre de profondeur d. La second inégalité vient du fait qu'un arbre binaire de profondeur d a 2^d branches. □

Dimension de Littlestone et online-learnability

On a le résultat suivant, analogue pour la « online-learnability » de la correspondance « VC-dimension finie / PAC-learnability » :

Théorème (Littlestone 1988)

Pour tout algorithme déterministe $\mathcal{A}$ et toute classe de fonctions $\mathcal{H}$, on a $M(\mathcal{A}) \geq \mathrm{Ldim}(\mathcal{H})$. De plus, il existe un algorithme pour lequel il y a égalité pour toute classe de fonctions $\mathcal{H}$.

Standard Optimal Algorithm

On obtient l'algorithme suivant, tel que pour toute classe $\mathcal{H}$ de VC-dimension finie, on a $\mathcal{M}(A) = \mathrm{Ldim}(\mathcal{H})$:

Algorithme 3 Standard Optimal Algorithm [SB14, p292]

Entrée: Une classe d'hypothèses finie $\mathcal{H}$.

1: On initialise $V_1 := \mathcal{H}$.
2: **pour** $t \leftarrow 1$ à T **faire**
3: On reçoit l'input x_t ;
4: On renvoie comme prédiction $\hat{y}_t \in \arg\max_{r \in \{0,1\}} \mathrm{Ldim}(V_t^{(r)})$;
5: On reçoit en retour le vrai label $y_t = f(x_t)$;
6: On met à jour l'ensemble des fonctions de prédiction possibles.
7: **fin pour**

Lien avec la théorie des modèles : stabilité

Définition (Stabilité d'une formule dans un modèle)

Soit $\mathcal{M}$ une $\mathcal{L}$-structure, $\phi(x, y)$ une $\mathcal{L}$-formule partitionnée avec $\bar{x}$ d'arité r et $\bar{y}$ d'arité s.
*On dit que ϕ est **instable** si il existe deux suites $\bar{a}_i \subseteq M^r$ et $\bar{b}_i \subseteq M^s$ (où $i \in \mathbb{N}$) tels que pour tous $i, j \in \mathbb{N}$, on ait*

$$\mathcal{M} \models \phi(\bar{a}_i, \bar{b}_j) \Leftrightarrow i \leq j$$

Définition (Stabilité d'une théorie)

Une théorie complète T est dite instable si il existe un modèle de T admettant une $\mathcal{L}$-formule partitionnée instable.
*Dans le cas contraire, la théorie est dite **stable**.*

Lien entre la dimension de Littlestone et la stabilité

Théorème

Soit $\mathcal{M}$ une $\mathcal{L}$-structure, $\phi(\bar{x}, \bar{y})$ une $\mathcal{L}$-formule partitionnée avec $\bar{x}$ d'arité r et $\bar{y}$ d'arité s et $\mathcal{C}_\phi := \{\{\bar{a} \in M^r \mid \phi(\bar{a}, \bar{b})\} \mid \bar{b} \in M^s\}$ Alors la dimension de Littlestone de $\mathcal{C}_\phi$ est précisément le « Shelah 2-rank » de $\phi(\bar{x}, \bar{y})$ qui est fini si et seulement si $\phi(\bar{x}, \bar{y})$ est stable.

Application de la théorie des modèles au problème d'apprentissage online

Le lien entre stabilité et apprentissage online est le contenu principal de l'article Hunter Chase et James Freitag. « Model Theory and Machine Learning ». In : *The Bulletin of Symbolic Logic* 25.3 (2019), p. 319-332

- La construction d'exemples de classes infinies ayant une dimension de Littlestone finie était un problème du coté « online learning ».
- La théorie des modèles offre une solution à ce problème car **à chaque théorie stable correspond des classes $\mathcal{C}_\phi$ intéressantes.**
- Par exemple, ACF est une théorie stable, et une famille $\mathcal{C}_\phi$ associée est décrite par un système d'équations et d'inégalités polynomiales.
- Il y a sans doute beaucoup de choses à faire en utilisant les résultats de théorie de la stabilité afin d'obtenir des classes de dimension de Littlestone finies.
- Voir les articles récents : [Mar19], [Bha20], [CF20].

Références I

Référence générale sur le machine learning :

Shai Shalev-Shwartz et Shai Ben-David. *Understanding Machine Learning : From Theory to Algorithms*. Cambridge University Press, 2014.

Une version est fournie par les auteurs en ligne et disponible à
https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/copy.html

PAC-learning et VC-dimension :

Hans Adler. *Introduction to theories without the independence property*. 2008. URL : http://www.logic.univie.ac.at/~adler/docs/nip.pdf.

Christophe Giraud. *Fondements mathématiques de l'apprentissage statistique*. 2013. URL : http://www.math.polytechnique.fr/xups/xups13-02.pdf.

Références II


Lê Nguyên Hoang. *Vidéo « PAC-Learning »*. URL : https://www.youtube.com/watch?v=uB2X2OuD4Rg.


N Sauer. « On the density of families of sets ». In : *Journal of Combinatorial Theory, Series A* 13.1 (1972), p. 145-147. ISSN : 0097-3165.


Saharon Shelah. « A combinatorial problem ; stability and order for models and theories in infinitary languages. ». In : *Pacific J. Math.* 41.1 (1972), p. 247-261.


L. G. Valiant. « A Theory of the Learnable ». In : *Commun. ACM* 27.11 (nov. 1984), p. 1134-1142. ISSN : 0001-0782.


Vladimir Vapnik et Alexey Chervonenkis. « The uniform convergence of frequencies of the appearance of events to their probabilities. (Russian) ». In : *Teor. Veroyatnost. i Primenen.* 16 (1971), p. 264-279.

Références III


James Worrell. *Exposé « Computational learning theory »*. 2018. URL : http://fopss18.mimuw.edu.pl/programme.html.

Online learning et dimension de Littlestone :


Shai Ben-David, D. Pál et S. Shalev-Shwartz. « Agnostic Online Learning ». In : *COLT*. 2009.


Christophe Giraud. *Mathématiques pour l'intelligence Artificielle II*. 2020. URL : https://www.imo.universite-paris-saclay.fr/~giraud/Orsay/MIA.pdf.


Nick Littlestone. « Learning Quickly When Irrelevant Attributes Abound : A New Linear-Threshold Algorithm ». In : *Machine Learning* 2 (1988), pages285-318.

Lien avec la théorie des modèles :


Siddharth Bhaskar. *Thicket Density*. 2020. arXiv : 1702.03956 [math.LO].

Références IV

Hunter Chase et James Freitag. « Model Theory and Combinatorics of Banned Sequences ». In : *The Journal of Symbolic Logic* (2020), p. 1-19.

Hunter Chase et James Freitag. « Model Theory and Machine Learning ». In : *The Bulletin of Symbolic Logic* 25.3 (2019), p. 319-332.

Artem Chernikov. *Model theory and combinatorics*. URL : https://www.math.ucla.edu/~chernikov/ModelTheoryAndCombinatoricsBook.html.

Michael C. Laskowski. « Vapnik-Chervonenkis Classes of Definable Sets ». In : *Journal of the London Mathematical Society* s2-45.2 (avr. 1992), p. 377-384. ISSN : 0024-6107. eprint : https://academic.oup.com/jlms/article-pdf/s2-45/2/377/6296871/s2-45-2-377.pdf.

Références V


David Marker. *Model Theory and Mathematical Logic, In Honor of Chris Laskowski's 60th Birthday*. 2019. URL : http://homepages.math.uic.edu/~marker/machine_learning.pdf.

Divers :


Gilbert H. Harman, Sanjeev R. Kulkarni et Hariharan Narayanan. « sin(ωx) Can Approximate Almost Every Finite Set of Samples ». In : *Constructive Approximation* (2015).


Terence Tao. *Brouwer's fixed point and invariance of domain theorems, and Hilbert's fifth problem*. 2011. URL : https://terrytao.wordpress.com/2011/06/13/brouwers-fixed-point-and-invariance-of-domain-theorems-and-hilberts-fifth-problem/.