ENSAE ParisTech
École nationale de la statistique et de l'administration économique

Apprentissage Statistique Appliqué

3ème année

Arnak DALALYAN

Table des matières

Préface

Ce polycopié a pour objectif de présenter les notions basiques de l'Apprentissage Statistique et servir de support pour le cours «Apprentissage Statistique Appliqué» en troisième année de l'ENSAE.

Je tiens à remercier Jean-Yves Audibert pour les emprunts qu'il m'a permis de faire au polycopié de son cours de «Machine Learning» enseigné en deuxième année de l'École des Ponts ParisTech.

1 Introduction

L'objectif général de l'Apprentissage Statistique (Machine Learning) est d'élaborer des procédures automatiques qui permettent de mettre en évidence des règles générales à partir d'exemples. Il s'agit donc d'imiter le fonctionnement inductif du cerveau humain dans le but de développer des systèmes d'intelligence artificielle. C'est considéré aujourd'hui comme une branche de l'Informatique (ou, plus précisément, de Computer Science). Cependant, les liens avec la *Statistique* sont très étroits, notamment avec la théorie non-paramétrique. Pour schématiser, on peut dire que la différence principale entre la Statistique et l'Apprentissage est que le concept central en Statistique est le modèle, alors qu'en Apprentissage c'est l'algorithme.

Tout comme en statistique, le point de départ en apprentissage est l'échantillon $Z_1, \ldots, Z_n$, que l'on veut utiliser, par exemple, pour faire des prévisions. Il existe différentes branches de l'Apprentissage caractérisées par la nature de l'échantillon et l'objectif poursuivi.

Apprentissage supervisé : les observations $Z_i = (X_i, Y_i)$ sont composées d'une variable d'entrée $X_i \in \mathbb{R}^d$, souvent appelée prédicteur ou feature, et d'une variable sortie Y_i appelée étiquette ou label, appartenant soit à un ensemble fini soit à $\mathbb{R}$. L'objectif est de prévoir, pour un $x \in \mathbb{R}^d$ donné (choisi de façon déterministe ou aléatoire), la valeur de l'étiquette correspondante.

Apprentissage non-supervisé : les observations Z_i ne sont pas étiquetées. Le plus souvent, $Z_i \in \mathbb{R}^d$ pour un $d \in \mathbb{N}$, assez grand, et le but est de caractériser la loi de probabilité ayant engendré ces observations. Le clustering ou encore l'estimation de la densité sont les problèmes les plus étudiés en Apprentissage non-supervisé.

Apprentissage semi-supervisé : seule une faible proportion n_1 des observations est étiquetée. On a donc $Z_i = (X_i, Y_i)$ pour $i = 1, \ldots, n_1$ et $Z_i = X_i$ pour $i = n_1+1, \ldots, n$. Le but est le même qu'en apprentissage supervisé. Ce cadre est intéressant pour de nombreuses applications où le coût d'étiquetage est très élevé.

Apprentissage en ligne : les observations sont révélées une par une. On privilégie alors les procédures dont le coût computationnel de mise à jour est faible.

Apprentissage actif : les objectifs sont les mêmes qu'en apprentissage supervisé, mais

la différence est que l'utilisateur peut choisir le feature X_i en fonction des résultats observés précédemment : $Z_1, \ldots, Z_{i-1}$. Cela se rapproche de la planification des expériences.

2

Apprentissage supervisé : régression et classification binaire

Sommaire

2.1 Généralités

Nous observons une base de données composée de n couples $Z_i = (X_i, Y_i)$ que nous supposons être des réalisations indépendantes d'une même loi P inconnue. On écrira

$$Z_i = (X_i, Y_i) \overset{iid}{\sim} P.$$

Les X_i appartiennent à un espace $\mathcal{X}$ et s'appellent les entrées ou les features. Typiquement, $\mathcal{X} = \mathbb{R}^d$ pour un grand entier d. Les Y_i appartiennent à un espace $\mathcal{Y}$, et s'appellent les sorties ou les étiquettes. Typiquement, $\mathcal{Y}$ est fini ou $\mathcal{Y}$ est un sous-ensemble de $\mathbb{R}$.

But de l'apprentissage supervisé : prévoir l'étiquette Y associée à toute nouvelle entrée X, où il est sous-entendu que la paire (X, Y) est une nouvelle réalisation de la loi P, cette réalisation étant indépendante des réalisations précédemment observées.

Une fonction de prédiction est une fonction (mesurable) de $\mathcal{X}$ dans $\mathcal{Y}$. Dans ce qui suit, nous supposons que toutes les quantités que nous manipulons sont mesurables. L'ensemble de

toutes les fonctions de prédiction est noté $\mathcal{F}(\mathcal{X}, \mathcal{Y})$. La base de données $Z_1, \ldots, Z_n$ est appelée ensemble d'apprentissage.

Un algorithme d'apprentissage est une fonction qui à tout ensemble d'apprentissage renvoie une fonction de prédiction, c'est-à-dire une fonction de l'union

$$\bigcup_{n=1}^{\infty} \mathcal{Z}^n$$

dans l'ensemble $\mathcal{F}(\mathcal{X}, \mathcal{Y})$, où $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. C'est un estimateur de «la meilleure» fonction de prédiction, où le terme «meilleure» sera précisé ultérieurement.

Soit $\ell(y, y')$ la perte encourue lorsque la sortie réelle est y et la sortie prédite est y'. La fonction $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ est appelée fonction de perte. Les deux exemples les plus fréquemment utilisés sont :

Exemple de la classification : $\ell(y, y') = \mathbb{1}(y \neq y')$, c'est-à-dire $\ell(y, y') = 1$ si $y \neq y'$ et $\ell(y, y') = 0$ sinon. Un problème d'apprentissage pour lequel cette fonction de perte est utilisée est appelé problème de classification. L'ensemble Y considéré en classification est le plus souvent fini, voire même de cardinal deux en classification binaire.

Exemple de la régression L_p : $\mathcal{Y} = \mathbb{R}$ et $\ell(y, y') = |y - y'|^p$ où $p \geq 1$ est un réel fixe. Dans ce cas, on parle de régression L_p. La tâche d'apprentissage lorsque $p = 2$ est aussi appelée régression aux moindres carrés.

La qualité d'une fonction de prédiction $g : \mathcal{X} \to \mathcal{Y}$ est mesurée par son risque (ou erreur de généralisation) :

$$R_P(g) = \mathbf{E}_P[\ell(Y, g(X))]. \tag{2.1}$$

Le risque est donc l'espérance par rapport à loi P de la perte encourue sur la donnée (X, Y) par la fonction de prédiction g.

La «meilleure» fonction de prédiction est la (ou plus rigoureusement une) fonction de $\mathcal{F}(\mathcal{X}, \mathcal{Y})$ minimisant le risque R_P :

$$g_P^* \in \arg \min_{g \in \mathcal{F}(\mathcal{X}, \mathcal{Y})} R_P(g). \tag{2.2}$$

Une telle fonction g_P^* n'existe pas nécessairement mais existe pour les fonctions de pertes usuelles, notamment celles que nous considérerons par la suite. Cette «meilleure» fonction sera appelée fonction oracle ou prédicteur de Bayes. Elle dépend de la probabilité inconnue P et, par conséquent, est inconnue.

2.2 Deux exemples

Dans ce paragraphe, nous proposons des exemples illustrant la problématique précédente.

2.2.1 Reconnaissance des caractères

La reconnaissance de caractères manuscrits est un des problèmes sur lequel les méthodes d'apprentissage ont permis des avancées fulgurantes. Le contexte est le suivant : nous disposons

d'une image numérisée d'un caractère manuscrit. Cette image est essentiellement un tableau de nombres réels indiquant l'intensité lumineuse en chacun des pixels. Nous souhaitons trouver la fonction qui à ce tableau de réels renvoie le caractère présent dans l'image.

A l'heure actuelle, les meilleures méthodes pour trouver une telle fonction sont de nature statistique : elles reposent donc sur

(1) la constitution d'une base d'images de caractères où les images sont étiquetées par le caractère qu'elle contient. Un X_i correspond donc à une de ces images et un Y_i désigne le caractère que X_i contient.
(2) l'utilisation de cette base pour proposer une estimation non paramétrique de la fonction cible.

Le taux d'erreur sur la reconnaissance de chiffres manuscrits (problème intéressant notamment les centres de tri postaux) sont de l'ordre de 0.35% pour les meilleurs algorithmes.

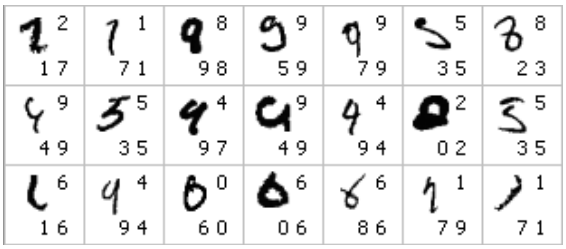


Figure 1 : *Reconnaissance de chiffres manuscrits. Les 21 erreurs sur les 10 000 caractères de la base de données MNIST d'un des meilleurs algorithmes*[a]*. Le nombre en haut à droite indique la vraie valeur. Les nombres en bas à droite sont les deux prédictions les plus probables.*

a. Dan Claudiu Ciresan, Ueli Meier, Luca Maria Gambardella, Juergen Schmidhuber (2010). Deep Big Simple Neural Nets Excel on Handwritten Digit Recognition, *arXiv :1003.0358*

2.2.2 Prédiction des marchés financiers

Pour faire face aux fluctuations incontrôlées du marché, les banques proposent aujourd'hui des produits dont les fluctuations sont indépendantes de la tendance (baissière ou haussière) du marché. Ces placements, dits de gestion alternative, reposent sur l'achat des actions dont le prix va croître le plus (ou du moins baisser le moins) et la vente pour le même montant des actions dont le prix va baisser le plus (ou croître le moins).

La difficulté pour mettre en place ce type de produit est d'identifier ces actions «sur-performantes» et «sous-performantes». Une méthode utilisée par les banques est :

- de recenser un ensemble de paramètres caractérisant l'action. Ces paramètres proviennent autant des analystes techniques (qui étudient les courbes à travers des paramètres tels que la moyenne mobile, les seuils de résistance, ...) que des analystes financiers (qui étudient les paramètres de la société : chiffre d'affaires, bénéfices, indices de rentabilité, ...) ;
- de constituer une base de données où une entrée X_i est un vecteur des paramètres décrits ci-dessus et la sortie Y_i associée évalue la sur/sous-performance de l'action sur la période suivant l'observation de ces paramètres (typiquement de l'ordre d'une semaine),

- l'apprentissage de la fonction oracle qui, à une date donnée, pour chaque action du marché, associe aux paramètres observés l'indice de sur/sous-performance de l'action.

2.3 Les fonctions oracles

Soit P_X la loi de X. Par définition de P_X, pour toute fonction intégrable $h : \mathcal{X} \to \mathbb{R}$,

$$\mathbf{E}_P[h(X)] = \int_{\mathcal{X}} h(x) dP_X(x) = \int_{\mathcal{Z}} h(x) dP(x, y).$$

Soit $P_{Y|X}(y|x)$ la loi conditionnelle de la variable aléatoire Y sachant $X = x$. Par définition, pour tout $x \in \mathcal{X}$, $P(\cdot|x)$ est une probabilité sur l'espace $\mathcal{Y}$ (en particulier, nous avons $P_{Y|X}(\mathcal{Y}|x) = \int_{\mathcal{Y}} dP_{Y|X}(y|x) = 1$) et pour toute fonction intégrable $f : \mathcal{Z} \to \mathbb{R}$,

$$\mathbf{E}_P[f(X, Y)] = \int_{\mathcal{X}} \Big\{ \int_{\mathcal{Y}} f(x, y) dP_{Y|X}(y|x) \Big\} dP_X(x) = \int_{\mathcal{X}} \mathbf{E}_P[f(x, Y) \,|\, X = x] \, dP_X(x). \quad (2.3)$$

Théorème 2.1.

1. *Supposons que pour tout $x \in \mathcal{X}$, l'infimum $\inf_{y \in \mathcal{Y}} \mathbf{E}[\ell(Y, y)|X = x]$ est atteint. Alors une fonction $g_P^* : \mathcal{X} \to \mathcal{Y}$ telle que pour tout $x \in \mathcal{X}$, $g_P^*(x)$ minimise $y \mapsto \mathbf{E}[\ell(Y, y)|X = x]$ est une fonction oracle. C'est-à-dire,*

$$\forall\, x \quad \in \mathcal{X} \; g_P^*(x) \in \arg\min_{y \in \mathcal{Y}} \mathbf{E}[\ell(Y, y)|X = x] \quad \Longrightarrow \quad g_P^* \in \arg\min_{g \in \mathcal{F}(\mathcal{X}, \mathcal{Y})} R_P(g). \quad (2.4)$$

2. *En régression aux moindres carrés, une fonction oracle est*

$$\eta_P^*(x) = \mathbf{E}_P(Y|X = x) = \int_{\mathcal{Y}} y dP_{Y|X}(y|x).$$

Par ailleurs, cette fonction satisfait :

$$\forall\, \eta : \mathcal{X} \to \mathbb{R} \qquad R_P(\eta) = R_P(\eta_P^*) + \mathbf{E}[\{\eta(X) - \eta_P^*(X)\}^2]. \quad (2.5)$$

3. *En classification, les fonctions oracles sont les fonctions g_P^* satisfaisant*

$$g_P^*(x) \in \arg\max_{y \in \mathcal{Y}} P(Y = y|X = x)$$

pour tout $x \in \mathcal{X}$. Par ailleurs, lorsque $\mathcal{Y} = \{0; 1\}$, la fonction

$$x \mapsto \mathbb{1}_{\{\eta_P^*(x) > 1/2\}} \quad (2.6)$$

est une fonction oracle pour la classification binaire. On l'appelle souvent classifieur de Bayes.

Démonstration. Démontrons dans l'ordre les trois assertions du théorème.

1. Soit $g \in \mathcal{F}(\mathcal{X}, \mathcal{Y})$ une fonction de prédiction quelconque et soit $\bar{g} \in \mathcal{F}(\mathcal{X}, \mathcal{Y})$ une fonction de prédiction satisfaisant

$$\bar{g}(x) \in \arg\min_{y \in \mathcal{Y}} \mathbf{E}_P \left[\ell \left(Y, y \right) | X = x \right], \qquad \forall x \in \mathcal{X}.$$

On a, d'après (2.3),

$$\begin{aligned} R_P[\, g \,] &= \mathbf{E}_P \left[\ell \left(Y, g \left(X \right) \right) \right] \\ &= \int \mathbf{E}_P \left[\ell \left(Y, g \left(X \right) \right) | X = x \right] P_X \left(\mathrm{d}x \right) \\ &= \int \mathbf{E}_P \left[\ell \left(Y, g(x) \right) | X = x \right] P_X \left(\mathrm{d}x \right) \\ &\geqslant \int \min_{y \in \mathcal{Y}} \mathbf{E}_P \left[\ell \left(Y, y \right) | X = x \right] P_X \left(\mathrm{d}x \right) \\ &= \int \mathbf{E}_P \left[\ell \left(Y, \bar{g}(x) \right) | X = x \right] P_X \left(\mathrm{d}x \right) \\ &= R_P \left[\bar{g} \right]. \end{aligned}$$

Cela implique que $\bar{g}$ a un risque inférieur ou égal à celui de n'importe quel autre fonction de prédiction. Par conséquent, $\bar{g}$ est un prédicteur de Bayes.

2. On introduit la notation

$$\varphi(y) = \mathbf{E}_P \left[\left(Y - y \right)^2 \mid X = x \right] \tag{2.7}$$

$$= \mathbf{E}_P \left[Y^2 \,|\, X = x \right] - 2y \, \mathbf{E}_P \left[Y \,|\, X = x \right] + y^2. \tag{2.8}$$

La première assertion du théorème implique que la fonction g^* définie par

$$g^*(x) \in \arg\min_{y \in \mathbb{R}} \mathbf{E}_P \left[\left(Y - y \right)^2 \mid X = x \right] = \arg\min_{y \in \mathbb{R}} \varphi \left(y \right)$$

est un classifieur de Bayes. La minimisation de cette fonction se fait facilement et conduit à

$$\arg\min_{y \in \mathbb{R}} \varphi \left(y \right) = \mathbf{E}_P \left[Y \,|\, X = x \right].$$

Cela montre que le prédicteur de Bayes est égal à la fonction de régression $\eta^*(x)$.
D'autre part, le risque d'un prédicteur quelconque η est donné par

$$\begin{aligned} R_P[\, \eta \,] &= \mathbf{E}_P \left[\left(Y - \eta \left(X \right) \right)^2 \right] \\ &= \mathbf{E}_P \left[\left(Y - \eta^* \left(X \right) \right)^2 \right] + 2 \mathbf{E}_P \left[\left(Y - \eta^* \left(X \right) \right) \left(\eta^* - \eta \right) \left(X \right) \right] + \mathbf{E}_P \left[\left(\eta^* - \eta \right)^2 \left(X \right) \right] \\ &= R_P \left[\eta^* \right] + 2 \mathbf{E}_P \Big(\mathbf{E}_P \left[\left(Y - \eta^* \left(X \right) \right) \left(\eta^* - \eta \right) \left(X \right) | X \right] \Big) + \mathbf{E}_P \left(\eta^* - \eta \right)^2 \left(X \right). \end{aligned}$$

Pour finir la preuve de (2.5), il suffit de remarque que le terme croisé dans l'expression ci-dessus est égal à zéro. En effet,

$$\mathbf{E}_P \left[\left(Y - \eta^* \left(X \right) \right) \left(\eta^* - \eta \right) \left(X \right) \,|\, X \right] = \left(\eta^* - \eta \right) \left(X \right) \mathbf{E}_P \left[Y - \eta^* \left(X \right) \mid X \right] = 0.$$

3. En vertu de la première assertion,

$$\begin{aligned} g^*(x) &\in \arg\min_{y\in\{0,1\}} \mathbf{E}_P\left[\mathbb{1}\left(Y \neq y\right) \mid X = x\right] \\ &= \arg\min_{y\in\{0,1\}} P\left(Y \neq y \mid X = x\right) \\ &= \arg\max_{y\in\{0,1\}} P\left(Y = y \mid X = x\right) \\ &= \arg\max_{y\in\{0,1\}} \Big\{\eta^*(x)\mathbb{1}(y=1) + (1-\eta^*(x))\mathbb{1}(y=0)\Big\}. \end{aligned}$$

Par conséquent,

$$g^*(x) = \begin{cases} 0, & \text{if } P\left(Y = 1 \mid X = x\right) \leqslant \frac{1}{2} \\ 1, & \text{otherwise.} \end{cases}$$

Cela complète la preuve du théorème. □

2.4 Lien entre classification binaire et régression

Dans cette section, nous considérons le problème de prédiction binaire, c'est-à-dire où la sortie ne peut prendre que deux valeurs. C'est en particulier la problématique des logiciels de lutte contre le spam. Sans perte de généralité, nous pouvons considérer : $\mathcal{Y} = \{0; 1\}$. Le théorème suivant précise le lien entre classification binaire et régression aux moindres carrés dans le contexte de la prédiction binaire.

Considérons $Y = \{0; 1\}$. Soit η_P^* la fonction oracle en régression aux moindres carrés définie par $\eta_P^*(x) = \mathbf{E}_P(Y|X = x) = P(Y = 1|X = x)$. Soit g_P^* la fonction oracle en classification binaire définie par

$$g_P^*(x) = \mathbb{1}_{\{\eta_P^*(x)>1/2\}}.$$

Pour toute fonction de régression $\eta : \mathcal{X} \to \mathbb{R}$, on définit le classifieur $g_\eta = \mathbb{1}_{\{\eta>1/2\}}$.

Théorème 2.2. *Soit $R_{cl,P}$ et $R_{reg,P}$ respectivement les risques en classification et en régression aux moindres carres : précisément $R_{cl,P}(g) = P[Y \neq g(X)]$ et $R_{reg,P}(\eta) = \mathbf{E}_P\big[\big(Y - \eta(X)\big)^2\big]$. Nous avons*

$$R_{cl,P}(g_\eta) - R_{cl,P}(g_P^*) \leq 2\sqrt{R_{reg,P}(\eta) - R_{reg,P}(\eta_P^*)}. \tag{2.9}$$

2.5 Consistance d'un algorithme

Rappelons-nous qu'un algorithme d'apprentissage g est une application

$$g : \bigcup_{n=1}^{\infty} \mathcal{Z}^n \to \mathcal{F}(\mathcal{X}, \mathcal{Y}).$$

Par conséquent, $g = \{g_n : n \in \mathbb{N}_*\}$ où chaque $g_n : \mathcal{D}_n \mapsto \widehat{g}_n \in \mathcal{F}(\mathcal{X}, \mathcal{Y})$ avec $\mathcal{D}_n = (Z_1, \ldots, Z_n)$. Comme $\mathcal{D}_n$ est aléatoire, il en est de même pour $\widehat{g}_n(x)$ quelle que soit la valeur $x \in \mathcal{X}$. Par conséquent, la quantité

$$R_P(\widehat{g}_n) = \int_{\mathcal{X}\times\mathcal{Y}} \ell(y, \widehat{g}_n(x))\, dP(x, y)$$

est aléatoire [1]. Nous pouvons donc considérer l'espérance de cette variable aléatoire par rapport à $\mathcal{D}_n \overset{iid}{\sim} P$, notée $\mathbf{E}_P[R_P(\widehat{g}_n)]$.

Définition 2.1. *Un algorithme d'apprentissage est dit consistant par rapport à la loi P si et seulement si*

$$\mathbf{E}_P[R_P(\widehat{g}_n)] \xrightarrow[n\to\infty]{} R_P(g_P^*).$$

Un algorithme d'apprentissage est dit consistant par rapport à une famille de lois $\mathcal{P}$ si et seulement si il est consistant par rapport à tout $P \in \mathcal{P}$.

Un algorithme d'apprentissage est dit universellement consistant si et seulement si il est consistant par rapport à toute probabilité P sur $\mathcal{Z}$.

Les résultats de consistance universelle ne disent pas le nombre de données nécessaires pour avoir une garantie du type $\mathbf{E}_P[R_P(\widehat{g}_n)] \leq R_P(g_P^*) + \epsilon$ pour $\epsilon > 0$ fixé. Pour que ce nombre existe, il faudrait avoir un résultat de consistance universelle uniforme, c'est-à-dire

$$\lim_{n\to+\infty} \sup_P \left\{\mathbf{E}_P[R_P(\widehat{g}_n)] - R_P(g_P^*)\right\} = 0,$$

la consistance universelle n'affirmant que

$$\sup_P \lim_{n\to+\infty} \left\{\mathbf{E}_P[R_P(\widehat{g}_n)] - R_P(g_P^*)\right\} = 0.$$

En général, ce nombre n'existe pas d'après le théorème suivant.

Théorème 2.3. *Si* $\mathrm{Card}(\mathcal{X}) = +\infty$*, il n'existe pas d'algorithme d'apprentissage uniformément universellement consistant ni en régression aux moindres carrés ni en classification binaire.*

L'absence d'algorithme universellement uniformément consistant nous amène à définir un «bon» algorithme d'apprentissage comme étant un algorithme universellement consistant et ayant une propriété de convergence uniforme sur une classe de probabilités paraissant pertinente pour le problème à traiter. Plus précisément, si $\mathcal{P}$ est un ensemble de probabilités sur $\mathcal{Z}$ dans laquelle nous pensons que P est, nous souhaitons que le bon algorithme satisfasse

$$\lim_{n\to+\infty} \sup_{P\in\mathcal{P}} \left\{\mathbf{E}_P[R_P(\widehat{g}_n)] - R_P(g_P^*)\right\} = 0$$

et également avoir une suite $\sup_{P\in\mathcal{P}} \left\{\mathbf{E}_P[R_P(\widehat{g}_n)] - R_P(g_P^*)\right\}$ décroissant le plus vite possible vers 0 pour que peu de données soient nécessaires à l'algorithme pour prédire efficacement dans le cas où $P \in \mathcal{P}$. L'ensemble $\mathcal{P}$ doit être pensé comme une modélisation de notre a priori, et il en résulte un a priori implicite sur la fonction cible. L'obtention d'algorithmes incorporant un a priori et étant efficace lorsque l'a priori est correct est au cœur de la recherche actuelle en apprentissage statistique.

2.6 Exercices

1. Considérons le problème de classification binaire avec $\mathcal{Y} = \{0; 1\}$.

1. Une manière plus complète d'écrire cette quantité est $R_P(\widehat{g}_n) = \int_{\mathcal{X}\times\mathcal{Y}} \ell(y, \widehat{g}_n(\mathcal{D}_n; x))\, dP(x, y)$.

(a) Montrer que pour tout classifieur g, on a

$$R_P(g) - R_P(g_P^*) = \mathbf{E}_P\big[\mathbb{1}\{g(X) \neq g_P^*(X)\} \cdot |2\eta_P^*(X) - 1|\big],$$

où $\eta_P^*(x) = \mathbf{E}_P(Y|X = x)$ et $g_P^*(x) = \mathbb{1}_{\{\eta_P^*(x)>1/2\}}$.

(b) En déduire que n'importe quel autre classifieur qui coïncide avec g_P^* sur le complémentaire de l'ensemble $\mathcal{X}_{0.5} = \{x \in \mathcal{X} : \eta_P^*(x) = 0.5\}$ est également un classifieur de Bayes.

(c) Montrer que si $\mathbb{1}_{\{\eta(x)>1/2\}} \neq \mathbb{1}_{\{\eta_P^*(x)>1/2\}}$, alors $1/2 \in [\eta(x); \eta_P^*(x)]$. En déduire que sur l'événement $g_\eta(X) \neq g_P^*(X)$, on a $|2\eta_P^*(X) - 1| \leq 2|\eta_P^*(X) - \eta(X)|$.

(d) Démontrer le Théorème 2.2.

(e) Est-il vrai que

$$R_P(g) - R_P(g_P^*) = \mathbf{E}_P\big[|g(X) - g_P^*(X)| \cdot |2\eta_P^*(X) - 1|\big]$$

pour tout $g : \mathcal{X} \to \{0; 1\}$?

2. On considère le problème de classification binaire avec $Y \sim \mathcal{B}(p)$, loi de Bernoulli de paramètre $p \in]0, 1[$, et

$$\mathscr{L}(X|Y = 0) = \mathcal{U}([0, 1/2]), \qquad \mathscr{L}(X|Y = 1) = \mathcal{U}([0, 1]).$$

(a) Quelle est la loi marginale de X (notée P_X) ? On déterminera la fonction de répartition de X en fonction du paramètre p.

(b) Déterminer la densité f_X de X par rapport à la mesure de Lebesgue.

(c) Pour tout $x \in [0, 1]$, calculer $\mathbf{E}_P[Y\mathbb{1}_{\{X\leq x\}}]$.

(d) On pose $\eta^*(x) = \mathbf{E}_p[Y|X = x]$. Montrer que pour tout $x \in]0, 1[$,

$$\mathbf{E}_P[Y\mathbb{1}_{\{X\leq x\}}] = \int_0^x \eta^*(u) f_X(u)\, du$$

et en déduire l'expression de $\eta^*(x)$.

(e) Déterminer la loi conditionnelle de Y sachant $X = x$ ainsi que la forme du classifieur de Bayes.

3. Soit $\mathcal{Y} = \{a_1, \ldots, a_K\}$ et soit $\mathcal{X}$ un ensemble mesurable de $\mathbb{R}^d$ pour un $d \in \mathbb{N}^*$. On considère le cadre d'apprentissage supervisé avec $(X_1, Y_1), \ldots, (X_n, Y_n)$ i.i.d. de loi commune P sur $\mathbb{R}^d \times \mathcal{Y}$. On note ν_d, la mesure de Lebesgue sur $\mathbb{R}^d$ et δ_{a_k} la mesure de Dirac sur $\mathcal{Y}$. On dit que la fonction $f : \mathbb{R}^d \times \mathcal{Y} \to \mathbb{R}_+$ est la densité de P par rapport à la mesure produit

$$\nu_d \otimes \underbrace{\big(\delta_{a_1} + \ldots + \delta_{a_K}\big)}_{\text{mesure de comptage}},$$

si pour toute fonction $h : \mathbb{R}^d \times \mathcal{Y} \to \mathbb{R}$ mesurable bornée on a

$$\mathbf{E}_P[h(X, Y)] = \sum_{k=1}^{K} \int_{\mathbb{R}^d} h(x, a_k)\, f(x, a_k)\, dx.$$

(a) Montrer que pour tout classifieur $g : \mathcal{X} \to \mathcal{Y}$, on a

$$P(g(X) \neq Y) = 1 - \int_{\mathbb{R}^d} f(x, g(x))\, dx. \tag{2.10}$$

(b) En déduire que si P est une loi à densité alors le classifieur oracle est donné par la formule

$$g_P^*(x) = \arg\max_{a \in \mathcal{Y}} f(x, a). \tag{2.11}$$

(c) Soit $K = 2$ avec $a_1 = 0$ et $a_2 = 1$. Montrer que le classifieur (2.11) coïncide avec le classifieur oracle (2.6) du Théorème 2.1.

(d) On suppose maintenant que $\mathcal{Y} = \{0, 1\}$ et que $f(x, k) = \varphi_k(x)\mathbf{P}(Y = k)$ où φ_k est la densité de la loi gaussienne de moyenne $\mu_k \in \mathbb{R}^d$ et de matrice de covariance Σ. Déterminer la forme du classifieur de Bayes et montrer qu'il coïncide avec la règle de classification linéaire de Fisher. Proposer un estimateur simple de g_P^* dans ce contexte basé sur un échantillon i.i.d. (X_i, Y_i), $i = 1, \ldots, n$.

3

Méthode de minimisation du risque empirique

Sommaire

3.1 Introduction

Rappelons tout d'abord que le risque d'une fonction de prédiction $g : \mathcal{X} \to \mathcal{Y}$ est défini par

$$R_P(g) = \mathbf{E}_P[\ell(Y, g(X))].$$

Le but d'un algorithme d'apprentissage est de trouver une fonction de prédiction dont le risque est aussi faible que possible (autrement dit aussi proche que possible du risque des prédicteurs oracles).

La distribution P générant les données étant inconnue, le risque R_P et les prédicteurs oracles sont inconnus. Néanmoins, le risque $R_P(g)$ peut être estimé par son équivalent empirique :

$$\widehat{R}_n(g) = \frac{1}{n}\sum_{i=1}^{n} \ell\big(Y_i, g(X_i)\big). \tag{3.1}$$

Si nous supposons $\mathbf{E}_P\big[\ell\big(Y, g(X)\big)^2\big] < +\infty$, alors la loi forte des grands nombres et le théorème de la limite centrale permettent d'affirmer

$$\widehat{R}_n(g) \xrightarrow[n\to\infty]{p.s.} R_P(g), \qquad \sqrt{n}\big\{\widehat{R}_n(g) - R_P(g)\big\} \xrightarrow[n\to\infty]{\mathscr{D}} \mathcal{N}\big(0, \mathrm{Var}[\ell(Y, g(X))]\big).$$

Pour toute fonction de prédiction g, la variable aléatoire $\widehat{R}_n(g)$ effectue donc des déviations en $O(1/\sqrt{n})$ autour de sa moyenne $R_P(g)$. Puisque nous cherchons une fonction qui minimise le risque R_P et puisque ce risque est approché par le risque empirique $\widehat{R}_n$, il est naturel de considérer l'algorithme d'apprentissage, dit de minimisation du risque empirique, défini par

$$\widehat{g}_{n,\mathcal{G}} = \arg\min_{g\in\mathcal{G}} \widehat{R}_n(g), \tag{3.2}$$

où $\mathcal{G}$ est un sous-ensemble de $\mathcal{F}(\mathcal{X},\mathcal{Y})$. Prendre $\mathcal{G} = \mathcal{F}(\mathcal{X},\mathcal{Y})$ n'est pas une bonne idée. Tout d'abord, cela entraîne un problème de choix puisqu'en général, pour tout ensemble d'apprentissage, il existe une infinité de fonctions de prédiction minimisant le risque empirique (voir Figure 2). Par ailleurs et surtout, si on prend l'algorithme du plus proche voisin comme minimiseur du risque empirique (en régression aux moindres carrés ou en classification), alors on peut montrer que cet algorithme est «loin» d'être universellement consistant.

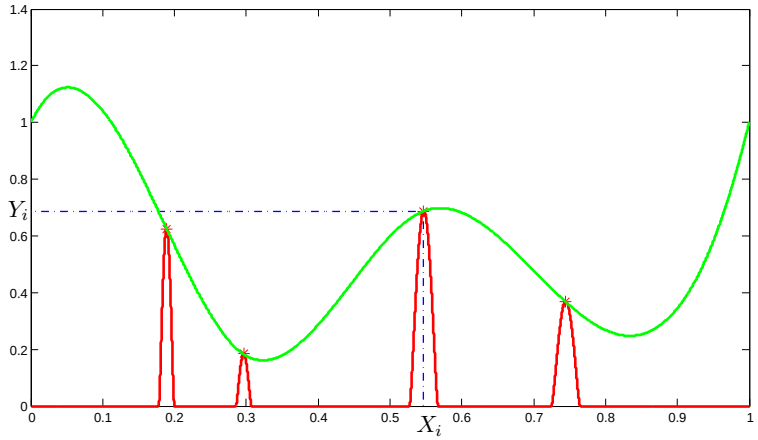


Figure 2 : *Surapprentissage en régression aux moindres carrés : les couples entrées-sorties sont représentés par des astérisques. Les deux courbes minimisent le risque empirique* $\frac{1}{n}\sum_{i=1}^{n}\big(Y_i - g(X_i)\big)^2$ *(puisqu'elles ont toutes les deux un risque empirique nulle). La courbe rouge semble apprendre par cœur la valeur des sorties associées aux entrées de l'ensemble d'apprentissage. On dit qu'elle «surapprend» (overfit).*

Prendre $\mathcal{G} = \mathcal{F}(\mathcal{X},\mathcal{Y})$ mène en général à un surapprentissage dans la mesure où l'algorithme résultant a un risque empirique qui peut être très inférieure à son risque réel (même lorsque la taille de l'ensemble d'apprentissage tend vers l'infini).

En pratique, il faut prendre $\mathcal{G}$ suffisamment grand pour pouvoir raisonnablement approcher toute fonction tout en ne le prenant pas trop grand pour éviter que l'algorithme «surapprenne». La «grandeur» de l'ensemble $\mathcal{G}$ est appelée capacité ou complexité. Un autre point de vue consiste à rajouter a $\widehat{R}_n(g)$ une pénalisation, quand par exemple, la fonction g est trop irrégulière. Ces deux approches sont en fait proches l'une de l'autre.

Soit $g^*_{P,\mathcal{G}}$ la fonction minimisant le risque sur $\mathcal{G}$:

$$g^*_{P,\mathcal{G}} = \arg\min_{g\in\mathcal{G}} R_P(g). \tag{3.3}$$

On suppose le minimum atteint pour simplifier l'exposé. La fonction $g^*_{P,\mathcal{G}}$ est appelée prédicteur oracle sur $\mathcal{G}$. D'après l'inégalité

$$R_P(\widehat{g}_{n,\mathcal{G}}) \geq R_P(g^*_{P,\mathcal{G}}) \geq R_P(g^*_P),$$

l'excès de risque de $\widehat{g}_{n,\mathcal{G}}$ se décompose en deux termes positifs, appelés erreur stochastique (ou erreur d'estimation) et erreur systématique (ou erreur d'approximation ou biais) :

$$R_P(\widehat{g}_{n,\mathcal{G}}) - R_P(g_P^*) = \underbrace{R_P(\widehat{g}_{n,\mathcal{G}}) - R_P(g_{P,\mathcal{G}}^*)}_{\text{erreur stochastique}} + \underbrace{R_P(g_{P,\mathcal{G}}^*) - R_P(g_P^*)}_{\text{erreur systématique}}.$$

Plus $\mathcal{G}$ est grand, plus l'erreur d'approximation est faible mais plus l'erreur d'estimation est en général grande. Il y a donc un compromis à trouver dans le choix de $\mathcal{G}$. Ce compromis est souvent appelé dilemme «biais-variance», où le terme variance provient du lien entre l'erreur stochastique et la variabilité de l'ensemble d'apprentissage que nous avons supposé dans notre formalisme être une réalisation de variables aléatoires i.i.d..

L'erreur stochastique peut être bornée par deux fois le supremum du processus empirique : $g \mapsto |R_P(g) - \widehat{R}_n(g)|$ défini sur $\mathcal{G}$. En effet, nous avons

$$\begin{aligned} R_P(\widehat{g}_{n,\mathcal{G}}) - R_P(g_{P,\mathcal{G}}^*) &= R_P(\widehat{g}_{n,\mathcal{G}}) - \widehat{R}_n(\widehat{g}_{n,\mathcal{G}}) + \underbrace{\widehat{R}_n(\widehat{g}_{n,\mathcal{G}}) - \widehat{R}_n(g_{P,\mathcal{G}}^*)}_{\leq 0} + \widehat{R}_n(g_{P,\mathcal{G}}^*) - R_P(g_{P,\mathcal{G}}^*) \\ &\leq \sup_{g\in\mathcal{G}} |R_P(g) - \widehat{R}_n(g)| + \sup_{g\in\mathcal{G}} |R_P(g) - \widehat{R}_n(g)| \\ &\leq 2 \sup_{g\in\mathcal{G}} |R_P(g) - \widehat{R}_n(g)|. \end{aligned} \tag{3.4}$$

On donnera plus tard dans ce chapitre des bornes plus élaborées concernant l'erreur de l'estimation ; toutes ces bornes reposeront sur l'inégalité (3.4).

3.2 Performance de la minimisation du risque empirique

Afin de quantifier la performance du minimiseur du risque empirique, nous allons étudier le comportement de l'erreur stochastique. Les résultats de consistance sont basés sur l'étude de $\mathbf{E}[R_P(\widehat{g}_{n,\mathcal{G}})]$, où l'espérance est prise par rapport à la loi de l'échantillon d'apprentissage $\mathcal{D}_n$. Pour mieux décrire la v.a. $R_P(\widehat{g}_{n,\mathcal{G}})$, nous allons établir différents types d'inégalités P.A.C. (probablement approximativement correct), c'est-à-dire des inégalités du type :

$$P\Big(R_P(\widehat{g}_{n,\mathcal{G}}) - R_P(g_{P,\mathcal{G}}^*) \leq \delta_n(\epsilon) \Big) \geq 1 - \epsilon, \qquad \forall \epsilon \in]0,1[, \tag{3.5}$$

pour une fonction $\delta_n :]0,1[\to \mathbb{R}_+$ bien choisie. Bien-entendu, plus cette fonction est proche de zéro plus le résultat (3.5) est précis.

Afin d'énoncer le premier résultat de ce type, nous avons besoin d'un résultat probabiliste auxiliaire, portant le nom d'inégalité de concentration, permettant d'évaluer la déviation d'un processus empirique par rapport à sa moyenne.

Proposition 3.1 (Inégalité de Hoeffding)**.** *Soit $V_1, \ldots, V_n$ des variables aléatoires réelles indépendantes telles que $a_i \leq V_i \leq b_i$ presque sûrement. On pose $\overline{V}_n = \frac{1}{n}\sum_{i=1}^n V_i$. On a alors*

$$P\big(\overline{V}_n - \mathbf{E}[\overline{V}_n] > t\big) \leq e^{-2n^2t^2/\sum_{i=1}^n (b_i-a_i)^2}, \qquad \forall t > 0. \tag{3.6}$$

Démonstration. Soit $W_i = V_i - 0.5(b_i + a_i)$ et $\varphi(s) = \ln \mathbf{E}[e^{sW_i}]$, la fonction génératrice des cumulants de W_i. Comme $|W_i|$ est bornée par $(b_i - a_i)/2$, l'espérance sous le logarithme est

finie. Il est claire que $\varphi(0) = 0$. Par conséquent, d'après la formule de Taylor, il existe $\zeta \in [0, s]$ tel que :

$$\varphi(s) = s\varphi'(0) + \frac{s^2}{2}\varphi''(\zeta) \leq s\varphi'(0) + \frac{s^2}{2}\|\varphi''\|_\infty. \tag{3.7}$$

On vérifie aisément que $\varphi'(s) = \mathbf{E}[W_i e^{sW_i}]/\mathbf{E}[e^{sW_i}]$ et que

$$\varphi''(s) = \frac{\mathbf{E}[W_i^2 e^{sW_i}]}{\mathbf{E}[e^{sW_i}]} - \left\{\frac{\mathbf{E}[W_i e^{sW_i}]}{\mathbf{E}[e^{sW_i}]}\right\}^2 \leq \frac{\mathbf{E}[W_i^2 e^{sW_i}]}{\mathbf{E}[e^{sW_i}]} \leq \frac{(b_i - a_i)^2}{4}.$$

Il en résulte que $\varphi(s) \leq s\mathbf{E}[W_i] + \frac{s^2(b_i-a_i)^2}{8}$, $\forall s \in \mathbb{R}$, ce qui équivaut à

$$\mathbf{E}[e^{s(W_i - \mathbf{E}[W_i])}] \leq e^{\frac{s^2(b_i-a_i)^2}{8}}, \qquad \forall s \in \mathbb{R}. \tag{3.8}$$

Pour conclure, on utilise l'argument de Chernoff (passer à l'exponentielle, puis inégalité de Markov, puis utilisation de l'indépendance). Plus précisément, d'après l'inégalité de Markov :

$$\begin{aligned} P\big(\overline{V}_n - \mathbf{E}[\overline{V}_n] > t\big) &= P\big(\overline{W}_n - \mathbf{E}[\overline{W}_n] > t\big) \\ &\leq e^{-st}\mathbf{E}[\exp\{s(\overline{W}_n - \mathbf{E}[\overline{W}_n])\}] \\ &= e^{-st}\mathbf{E}\Big[\exp\Big\{\frac{s}{n}\sum_{i=1}^n (W_i - \mathbf{E}[W_i])\Big\}\Big] \end{aligned}$$

En utilisant le fait que les W_i sont indépendantes, ainsi que l'inégalité (3.8), il vient

$$\begin{aligned} e^{-st}\mathbf{E}\Big[\exp\Big\{\frac{s}{n}\sum_{i=1}^n (W_i - \mathbf{E}[W_i])\Big\}\Big] &= e^{-st}\prod_{i=1}^n \mathbf{E}\Big[\exp\Big\{\frac{s}{n}(W_i - \mathbf{E}[W_i])\Big\}\Big] \\ &\leq \exp\Big\{-st + \sum_{i=1}^n \frac{s^2(b_i-a_i)^2}{8n^2}\Big\}. \end{aligned}$$

Cette inégalité étant valable pour tout $s \in \mathbb{R}$, elle l'est aussi pour $s = 4n^2t/\sum_i (b_i - a_i)^2$ et implique l'inégalité énoncée dans la proposition. □

Remarquons d'abord qu'en appliquant l'inégalité de Hoeffding à la suite des variables aléatoires $-V_i$ qui sont à valeurs dans $[-b_i, -a_i]$, on trouve

$$P\big(\overline{V}_n - \mathbf{E}[\overline{V}_n] < -t\big) \leq e^{-2n^2t^2/\sum_{i=1}^n (b_i-a_i)^2}, \qquad \forall t > 0.$$

Par conséquent, en utilisant l'inégalité $P(A \cup B) \leq P(A) + P(B)$, appelée borne d'union, il vient

$$P\big(|\overline{V}_n - \mathbf{E}[\overline{V}_n]| > t\big) \leq 2e^{-2n^2t^2/\sum_{i=1}^n (b_i-a_i)^2}, \qquad \forall t > 0.$$

Lorsque l'ensemble $\mathcal{G}$ de fonctions considérées est fini, le cardinal de $\mathcal{G}$ est une première mesure de complexité (ou capacité) de $\mathcal{G}$.

Théorème 3.1. *Supposons qu'il existe deux nombres réels $a < b$ tels que la perte ℓ vérifie $a \leq \ell(y, y') \leq b$ pour tout $y, y' \in \mathcal{Y}$. Alors pour tout $\varepsilon > 0$, avec probabilité au moins $1 - \varepsilon$,*

$$R_P(\widehat{g}_{n,\mathcal{G}}) - R_P(g^*_{P,\mathcal{G}}) \leq (b - a)\sqrt{\frac{2\ln(2|\mathcal{G}|\varepsilon^{-1})}{n}}.$$

Démonstration. D'après l'inégalité (3.4) :

$$R_P(\widehat{g}_{n,\mathcal{G}}) - R_P(g^*_{P,\mathcal{G}}) \leq 2 \sup_{g \in \mathcal{G}} |R_P(g) - \widehat{R}_n(g)|.$$

En vertu de l'inégalité de Hoeffding : pour tout $g \in \mathcal{G}$, pour tout $\varepsilon > 0$, avec probabilité au moins $1 - \varepsilon$:

$$|R_P(g) - \widehat{R}_n(g)| \leq (b-a)\sqrt{\frac{\ln(2\varepsilon^{-1})}{2n}}.$$

L'utilisation de la borne d'union complète la démonstration. □

3.3 Classification binaire et dimension de Vapnik-Chervonenkis

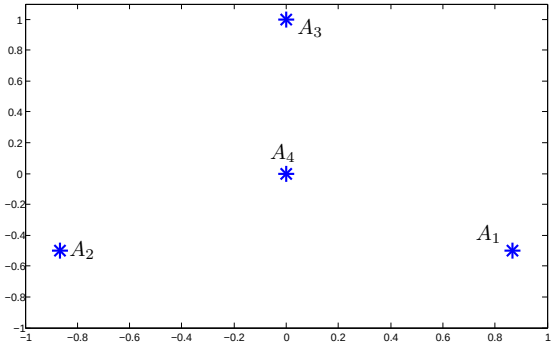
En utilisant des arguments plus ingénieux, il est possible d'obtenir un résultat plus fin, qui est valable même pour les classes $\mathcal{G}$ contenant un nombre infini de fonctions. L'idée est que la quantité pertinente pour mesurer la capacité d'une classe $\mathcal{G}$ n'est pas le cardinal de $\mathcal{G}$, mais le nombre de fonctions dans $\mathcal{G}$ qui conduisent à des prédictions différentes sur l'échantillon $\boldsymbol{X}_n = (X_1, \ldots, X_n)$. En effet, pour deux fonctions $g, g' \in \mathcal{G}$ qui coïncident sur l'échantillon, c'est-à-dire $g(X_i) = g'(X_i)$, $i = 1, \ldots, n$, les risques empiriques $\widehat{R}_n(g)$ et $\widehat{R}_n(g')$ sont égaux. On devrait donc être en mesure de remplacer $|\mathcal{G}|$ dans l'inégalité oracle du Théorème 3.1 par le cardinal d'une sous-classe de $\mathcal{G}$ qui contient <u>un</u> minimiseur du risque empirique, quel que soit l'échantillon $\mathcal{D}_n$.

Afin de concrétiser cette idée, on introduit quelques notions supplémentaires. Étant donné que nous nous intéressons au problème de classification binaire avec $\mathcal{Y} = \{0, 1\}$, tout prédicteur g peut être identifié à l'ensemble $G := G(g) = \{x \in \mathcal{X} : g(x) = 1\}$. On a alors $g(x) = \mathbb{1}_G(x)$. Pour tout ensemble $S = \{x_1, \ldots, x_k\} \subset \mathcal{X}$ on note 2^S l'ensemble des parties de S, et on définit la trace de $\mathcal{G}$ sur S par

$$T_{\mathcal{G}}(S) = \big\{S \cap G : g \in \mathcal{G}\big\} \subset 2^S.$$

Lorsque la dernière inclusion est une égalité, on dit que S est pulvérisé (shattered) par $\mathcal{G}$.

Exemple : Soit $\mathcal{X} = \mathbb{R}^2$ et $\mathcal{G} = \{\mathbb{1}_{B_R(a)} : R > 0, a \in \mathbb{R}^2\}$, où $B_R(a)$ désigne le disque de $\mathbb{R}^2$ de rayon R centré en a. Soit $A_1, \ldots, A_4$ quatre points de $\mathbb{R}^2$ positionnés comme suit :



Posons $S = \{A_1, A_2, A_3\}$ et $S' = \{A_1, A_2, A_3, A_4\}$ On vérifie aisément que S est pulvérisé par $\mathcal{G}$ mais pas S'. En effet, il est clair que la trace de $\mathcal{G}$ sur S' ne contient pas l'ensemble $\{A_1, A_2, A_3\}$. Par ailleurs, c'est le seul sous-ensemble de S' qui n'appartient pas à $T_{\mathcal{G}}(S')$.

Nous sommes maintenant en position pour définir une nouvelle notion de complexité de $\mathcal{G}$, appelé dimension de Vapnik-Chervonenkis, qui dépend de l'échantillon $\boldsymbol{X}_n$.

Définition 3.1. *La dimension de Vapnik-Chervonenkis de $\mathcal{G}$ est*

$$V_{\mathcal{G}} = \max\Big\{J \in \mathbb{N} : \exists S \subset \mathcal{X} \text{ t.q. } |S| = J \text{ et } S \text{ est pulvérisé par } \mathcal{G}\Big\}.$$

Si $V_{\mathcal{G}} < +\infty$, $\mathcal{G}$ est appelé classe de Vapnik-Chervonenkis. La dimension de Vapnik-Chervonenkis de $\mathcal{G}$ sur l'échantillon $\boldsymbol{X}_n = (X_1, \ldots, X_n)$ est définie par

$$V_{\mathcal{G}}(\boldsymbol{X}_n) = \max\Big\{J \in \mathbb{N} : \max_{S \subset \{X_1,\ldots,X_n\}:|S|=J} |T_{\mathcal{G}}(S)| = 2^J\Big\}.$$

Un ensemble fini des entrées S est donc pulvérisé par $\mathcal{G}$ si tout étiquetage de S peut être obtenu par une fonction de prédiction $g \in \mathcal{G}$. La dimension de Vapnik-Chervonenkis (la VC-dimension) de $\mathcal{G}$ est alors égale au cardinal maximal d'un ensemble vérifiant cette propriété.

La VC-dimension d'une classe $\mathcal{G}$ est par définition supérieure ou égale à 0. Il n'est pas difficile de vérifier que $V_{\mathcal{G}} = 0$ si et seulement si $\mathcal{G}$ ne contient qu'un seul élément. Il découle également de la définition que

$$V_{\mathcal{G}}(\boldsymbol{X}_n) \leq n, \qquad \text{et} \qquad V_{\mathcal{G}} = \sup_{n\in\mathbb{N}} \sup_{\boldsymbol{X}_n \in \mathcal{X}^n} V_{\mathcal{G}}(\boldsymbol{X}_n). \tag{3.9}$$

Lemme 3.1 (Lemme de Sauer (admis))**.** - *Pour tout $S = \{x_1, \ldots, x_N\}$ avec $\boldsymbol{x}_N = (x_1, \ldots, x_N) \in \mathcal{X}^N$,*

$$|T_{\mathcal{G}}(S)| \leq \sum_{k=0}^{V_{\mathcal{G}}(\boldsymbol{x}_N)} C_N^k \leq (N+1)^{V_{\mathcal{G}}(\boldsymbol{x}_N)} \leq (N+1)^{V_{\mathcal{G}}(N)},$$

où $V_{\mathcal{G}}(N) = \sup_{\boldsymbol{x}_N \in \mathcal{X}^N} V_{\mathcal{G}}(\boldsymbol{x}_N)$.

Théorème 3.2 (Vapnik-Chervonenkis)**.** *Soit $\mathcal{Y} = \{0, 1\}$ et soit $\ell : \mathcal{Y} \times \mathcal{Y} \to \{0, 1\}$ la perte de classification. Pour tout $\varepsilon > 0$, avec une probabilité au moins $1 - \varepsilon$, on a*

$$R_P(\widehat{g}_{n,\mathcal{G}}) - R_P(g^*_{P,\mathcal{G}}) \leq 2\sqrt{\frac{2V_{\mathcal{G}}(2n)\ln[4(2n+1)\varepsilon^{-1}]}{n}}.$$

Démonstration. Si l'on définit le réel positif t par $nt^2 = 2V_{\mathcal{G}}(2n)\ln[4(2n+1)\varepsilon^{-1}]$, l'inégalité que l'on cherche à prouver s'écrit

$$P\Big(R_P(\widehat{g}_{n,\mathcal{G}}) - R_P(g^*_{P,\mathcal{G}}) > 2t\Big) \leq \varepsilon.$$

Or, d'après (2.11), on a $P\big\{R_P(\widehat{g}_{n,\mathcal{G}}) - R_P(g^*_{P,\mathcal{G}}) > 2t\big\} \leq P(\sup_{g\in\mathcal{G}} |\widehat{R}_n(g) - R_P(g)| > t)$. En vertu du lemme de symétrisation, voir (3.13) ci-après, on a

$$P\Big(\sup_{g\in\mathcal{G}} |\widehat{R}_n(g) - R_P(g)| > t\Big) \leq 4P\Big(\sup_{g\in\mathcal{G}}\{\widehat{R}_n(g) - \widehat{R}'_n(g)\} > t\Big).$$

On considère l'ensemble $T_{\mathcal{G}}(\boldsymbol{X}_n, \boldsymbol{X}'_n)$. Ainsi, tout élément de $T_{\mathcal{G}}(\boldsymbol{X}_n, \boldsymbol{X}'_n)$ est un ensemble de la forme $\{x_{j_1}, \ldots, x_{j_K}\}$ pour un entier $K \leq 2n$. A chacun de ces éléments, on associe un prédicteur $g \in \mathcal{G}$ telle que $g(x_{j_k}) = 1$ pour tous ces éléments. De cette façon, on construit une bijection entre $T_{\mathcal{G}}(\boldsymbol{X}_n, \boldsymbol{X}'_n)$ et un sous-ensemble de $\mathcal{G}$ que l'on note $\widehat{\mathcal{G}}_n$. Remarquons

que $\sup_{g\in\mathcal{G}}\{\widehat{R}_n(g) - \widehat{R}'_n(g)\} = \sup_{g\in\widehat{\mathcal{G}}_n}\{\widehat{R}_n(g) - \widehat{R}'_n(g)\}$ et que $|\widehat{\mathcal{G}}_n| \leq (2n+1)^{V_{\mathcal{G}}(2n)}$. Par conséquent, en utilisant la borne d'union, il vient

$$\begin{aligned} P\Big(\sup_{g\in\widehat{\mathcal{G}}_n}\{\widehat{R}_n(g) - \widehat{R}'_n(g)\} > t\Big) &\leq \sum_{g\in\widehat{\mathcal{G}}_n} P\Big(\widehat{R}_n(g) - \widehat{R}'_n(g) > t\Big) \\ &\leq |\widehat{\mathcal{G}}_n| \max_{g\in\widehat{\mathcal{G}}_n} P\Big(\widehat{R}_n(g) - \widehat{R}'_n(g) > t\Big) \\ &\leq (2n+1)^{V_{\mathcal{G}}(2n)} \max_{g\in\widehat{\mathcal{G}}_n} P\Big(\widehat{R}_n(g) - \widehat{R}'_n(g) > t\Big). \end{aligned} \tag{3.10}$$

La suite découle immédiatement de l'inégalité de Hoeffding. □

Les lecteurs attentifs remarqueront que nous avons triché dans la démonstration du théorème de Vapnik-Chervonenkis. En effet, l'inégalité (3.10) n'est pas correcte car l'ensemble $\widehat{\mathcal{G}}_n$ ainsi que ses éléments sont aléatoires et, par conséquent, on ne peut pas employer la borne d'union. La démonstration correcte de ce théorème est un peu plus longue et est laissée comme exercice à la fin de ce chapitre.

3.4 Remarques

Complexité de $\mathcal{G}$ et consistance : Tous les résultats démontrés dans ce chapitre portent sur l'évaluation de l'erreur stochastique de l'algorithme de minimisation du risque empirique. Cependant, on sait que l'erreur globale se calcule comme la somme de l'erreur stochastique et de l'erreur systématique, cette dernière étant égale à

$$R_P(g^*_{P,\mathcal{G}}) - R_P(g^*_P).$$

Cela montre que pour garantir la consistance du minimiseur du risque empirique (MRE) $\widehat{g}_{n,\mathcal{G}}$, il est – dans la plupart des cas – indispensable de faire dépendre $\mathcal{G}$ de la taille de l'échantillon. En effet, si $\mathcal{G}$ est fixé quel que soit n et ne contient pas de prédicteur oracle, alors le risque $R_P(\widehat{g}_{n,\mathcal{G}})$ ne peut pas tendre vers le risque du prédicteur oracle, car la différence de ces deux risques reste toujours minorée par la constante $R_P(g^*_{P,\mathcal{G}}) - R_P(g^*_P) > 0$. En conclusion, pour espérer avoir la consistance du MRE il faut prendre $\mathcal{G} = \mathcal{G}_n$. De plus, plus $\mathcal{G}$ est gros, plus l'erreur systématique du MRE est faible. Par conséquent, la complexité de $\mathcal{G}_n$ est en général une fonction croissante de n.

Inégalités oracles : On suppose désormais que $\mathcal{G} = \mathcal{G}_n$ dépend de n. Les résultats obtenus dans ce chapitre peuvent être écrits sous la forme :

$$R_P(\widehat{g}_{n,\mathcal{G}_n}) \leq A \min_{g\in\mathcal{G}_n} R_P(g) + B\sqrt{\frac{\ln\{\mathscr{C}_n(\mathcal{G}_n)/\varepsilon\}}{n}}, \qquad \text{avec une probabilité} \geq 1-\varepsilon, \tag{3.11}$$

où A et B sont des constantes numériques et $\mathscr{C}_n(\mathcal{G}_n)$ désigne la complexité de $\mathcal{G}_n$ (ici, soit $\mathscr{C}_n(\mathcal{G}_n) = 2|\mathcal{G}_n|$ soit $\mathscr{C}(\mathcal{G}_n) = 4(2n+1)^{V_{\mathcal{G}_n}(2n)}$).

On dit que (3.11) est une inégalité oracle. On l'appelle inégalité oracle précise (sharp oracle inequality), si $A = 1$. On l'appelle optimale, si le second terme du membre de droite décroît vers zéro avec une vitesse optimale dans un certain sens. La vitesse $\sqrt{\ln\{\mathscr{C}_n(\mathcal{G}_n)\}/n}$ est une vitesse pessimiste, dite aussi vitesse lente, car elle est obtenue

sans aucune hypothèse sur la relation entre la classe $\mathcal{G}_n$ est les prédicteurs oracles. On verra plus loin que dans un scénario plus optimiste où la classe $\mathcal{G}_n$ contiendrait un prédicteur oracle, il est possible d'atteindre la vitesse $\ln\{\mathscr{C}_n(\mathcal{G}_n)\}/n$.

Evaluation de l'erreur : Les inégalités oracles présentées dans ce chapitre nous donnent une idée assez claire de la vitesse à laquelle le risque du MRE tend vers le risque du meilleur prédicteur de la classe, mais ne nous fournissent pas de borne supérieure, ne serait-ce que grossière, du risque $R_P(\widehat{g}_{n,\mathcal{G}})$ calculable à partir des données. Cependant, en examinant les démonstrations des Théorèmes 3.1 et 3.2, on peut remarquer que ce que l'on borne réellement, c'est la probabilité de l'événement $\sup_g |\widehat{R}_n(g) - R_P(g)| > t$ qui est forcément supérieure ou égale à $P\big(|\widehat{R}_n(\widehat{g}_{n,\mathcal{G}}) - R_P(\widehat{g}_{n,\mathcal{G}})| > t\big) \geq P\big(R_P(\widehat{g}_{n,\mathcal{G}}) > \widehat{R}_n(\widehat{g}_{n,\mathcal{G}}) + t\big)$. On peut donc déduire des démonstrations précédentes des inégalités du type

$$R_P(\widehat{g}_{n,\mathcal{G}}) \leq \widehat{R}_n(\widehat{g}_{n,\mathcal{G}}) + B\sqrt{\frac{\ln\{\mathscr{C}_n(\mathcal{G}_n)/\varepsilon\}}{n}}, \qquad \text{avec une probabilité} \geq 1-\varepsilon. \quad (3.12)$$

L'intérêt de cette inégalité est que le membre droite est calculable à partir des données et peut servir comme une évaluation de l'erreur de prédiction.

D'où vient le terme «dimension» pour $V_{\mathcal{G}}$? On peut vérifier que si les entrées X sont d-dimensionnels $\mathcal{X} = \mathbb{R}^d$ et

$$\mathcal{G} = \{\mathbb{1}_{]0,\infty[}(\langle \boldsymbol{v}, \boldsymbol{x}\rangle) : \boldsymbol{v} \in \mathbb{R}^d\}$$

est l'ensemble des classifieurs linéaires, alors $V_{\mathcal{G}}$ coïncide avec la dimension de $\mathcal{X}$, c'est-à-dire $V_{\mathcal{G}} = d$.

La nature de la classe $\mathcal{G}$: La démonstration du Théorème de Vapnik-Chervonenkis, dont les grandes lignes sont présentées dans l'Exercice 3 ci-dessous, reste valable lorsque la classe $\mathcal{G}$ dépend de l'échantillon d'apprentissage $\mathcal{D}_n$. Par conséquent, le Théorème 3.2 couvre certains classifieurs très utilisés en pratique que l'on va introduire dans les chapitres suivants. Il s'agit du classifieur par k plus proches voisins (k nearest neighbor, kNN), des arbres de décision, des réseaux de neurones, etc.

3.5 Exercices

1. On considère le modèle de régression L_1 : $\mathcal{Y} = \mathbb{R}$ et $\ell(y, y') = |y - y'|$. Supposons que la loi conditionnelle de Y sachant $X = x$ admet une densité strictement positive $f_{Y|X}(y|x)$ par rapport à la mesure de Lebesgue sur $\mathbb{R}$ quelle que soit la valeur de $x \in \mathcal{X}$.
   (a) Déterminer le prédicteur de Bayes g_P^*.
   (b) On cherche maintenant la meilleure prédiction appartenant à la classe des prédictions constantes $\mathcal{G} = \{g : \mathcal{X} \to \mathbb{R} : \exists c \in \mathbb{R} \text{ t.q. } g(x) \equiv c\}$. Déterminer le minimiseur du risque empirique $\widehat{g}_{n,\mathcal{G}}$.
2. La démonstration du Théorème 3.2 de Vapnik-Chervonenkis repose sur l'inégalité de symétrisation suivante. Soit ℓ une fonction de perte à valeurs dans $[0, 1]$ (ou tout autre intervalle de longueur 1) et soit $\mathcal{G}$ un sous-ensemble de $\mathcal{F}(\mathcal{X}, \mathcal{Y})$. Considérons un échantillon $\mathcal{D}'_n = (Z'_1, \ldots, Z'_n)$ de même loi que et indépendant de $\mathcal{D}_n$. Alors, pour tout

$t > \sqrt{2/n}$,

$$P\Big(\sup_{g\in\mathcal{G}}\big\{\widehat{R}_n(g) - R_P(g)\big\} > t\Big) \le 2P\Big(\sup_{g\in\mathcal{G}}\big\{\widehat{R}_n(g,\mathcal{D}_n) - \widehat{R}_n(g,\mathcal{D}'_n)\big\} > t/2\Big) \quad (3.13)$$

avec $\widehat{R}_n(g,\mathcal{D}'_n) = \frac{1}{n}\sum_{i=1}^n \ell\big(Y'_i, g(X'_i)\big) := \widehat{R}'_n(g)$. Le but de cet exercice est de démontrer l'inégalité (3.13).

(a) Soit g_n une fonction telle que le maximum de $\widehat{R}_n(g) - R_P(g)$ sur $\mathcal{G}$ est atteint en g_n. Justifier l'inégalité

$$\mathbb{1}\big(\widehat{R}_n(g) - R_P(g) > t\big)\mathbb{1}\big(\widehat{R}'_n(g) - R_P(g) < t/2\big) \le \mathbb{1}\big(\widehat{R}_n(g) - \widehat{R}'_n(g) > t/2\big)$$

et en déduire que

$$P\bigg(\widehat{R}_n(g_n) - \widehat{R}'_n(g_n) > \frac{t}{2}\,\Big|\,\mathcal{D}_n\bigg) \ge \mathbb{1}_{\{\widehat{R}_n(g_n) - R_P(g_n) > t\}} P\bigg(\widehat{R}'_n(g_n) - R_P(g_n) < \frac{t}{2}\,\Big|\,\mathcal{D}_n\bigg).$$

(b) En utilisant l'inégalité de Tchebychev, démontrer que pour toute fonction de perte ℓ à valeurs dans $[a,b]$, on a

$$P\bigg(\widehat{R}'_n(g_n) - R_P(g_n) < \frac{t}{2}\,\Big|\,\mathcal{D}_n\bigg) > 1 - \frac{(b-a)^2}{nt^2}.$$

(c) Conclure.

3. Soit $\boldsymbol{Z}_{2n} = (Z_1,\dots,Z_{2n})$ un échantillon i.i.d. de loi P sur $\mathcal{X}\times\mathcal{Y}$. On pose $Q_g(Z_i) = \ell(Y_i, g(X_i))$ et $Q_g(\boldsymbol{Z}_{2n}) = [Q_g(Z_1),\dots,Q_g(Z_{2n})]$. Soit $\boldsymbol{s}\in\{-1,+1\}^{2n}$ un vecteur signe quelconque vérifiant $\sum_{i=1}^{2n} s_i = 0$.

(a) Montrer que pour toute fonction de perte à valeurs dans $[0,1]$, ou tout autre intervalle de longueur 1,

$$P\Big(\sup_{g\in\mathcal{G}}\big\{\widehat{R}_n(g) - R_P(g)\big\} > t\Big) \le 2P\Big(\sup_{g\in\mathcal{G}}\langle \boldsymbol{s}, Q_g(\boldsymbol{Z}_{2n})\rangle > nt\Big)$$

pour tout $t > \sqrt{2/n}$.

(b) Soit $\boldsymbol{\xi} = (\xi_1,\dots,\xi_{2n})$ un vecteur aléatoire dont les composantes ξ_i sont i.i.d. telle que $P(\xi_i = 1) = P(\xi_i = -1) = 1/2$. Déduire de la question précédente que

$$P\Big(\sup_{g\in\mathcal{G}}\big\{\widehat{R}_n(g) - R_P(g)\big\} > t\Big) \le 2P\Big(\sup_{g\in\mathcal{G}}\langle \boldsymbol{\xi}, Q_g(\boldsymbol{Z}_{2n})\rangle > nt \,\Big|\, \sum_{i=1}^{2n}\xi_i = 0\Big)$$

pour tout $t > \sqrt{2/n}$.

(c) Vérifier que $\mathbf{E}\big[\xi_i Q_g(Z_i)\big|\sum_{i=1}^{2n}\xi_i = 0\big] = 0$ pour tout $i\in\{1,\dots,2n\}$.

(d) On suppose dans toute la suite de cet exercice que $Q_g(z)\in\{0,1\}$ pour tout $g\in\mathcal{G}$ et pour tout $z\in\mathcal{Z} = \mathcal{X}\times\mathcal{Y}$. Soit $\boldsymbol{Q}\in\{0,1\}^{2n}$ un vecteur vérifiant $\sum_i Q_i = m$. Montrer que

$$P\Big(\langle\boldsymbol{\xi},\boldsymbol{Q}\rangle > nt \,\Big|\, \sum_{i=1}^{2n}\xi_i = 0\Big) = \sum_{k=\lceil (m+nt)/2\rceil}^{m} P\Big(\sum_{i=1}^{m}\xi_i = 2k-m \,\Big|\, \sum_{i=1}^{2n}\xi_i = 0\Big),$$

où $\lceil a \rceil$ désigne le plafond de a, c'est-à-dire la plus petite valeur entière strictement plus grande que a. En déduire que

$$P\Big(\langle \boldsymbol{\xi}, \boldsymbol{Q} \rangle > nt \Big| \sum_{i=1}^{2n} \xi_i = 0\Big) = \sum_{k=\lceil (m+nt)/2 \rceil}^{m} \frac{C_m^k C_{2n-m}^{n-k}}{C_{2n}^n}.$$

(e) On admet que

$$\sum_{k=\lceil (m+nt)/2 \rceil}^{m} \frac{C_m^k C_{2n-m}^{n-k}}{C_{2n}^n} \leq e^{-t^2 n^2/(n+1)} \leq e^{-t^2 n/2}. \tag{3.14}$$

Démontrer le théorème de Vapnik-Chervonenkis.

4. Soit $\mathcal{X} = \mathbb{R}^d$. Calculer $V_{\mathcal{G}}$ lorsque
   (a) $d = 1$ et $\mathcal{G}_1 = \{g = \mathbb{1}_{]a,b[} : \text{ avec } a, b \in [-\infty, +\infty]\}$ est l'ensemble de tous les intervalles ouverts,
   (b) $d = 1$ et $\mathcal{G}_2 = \{g : g = g_1 + g_2 - g_1 g_2 \text{ avec } g_1, g_2 \in \mathcal{G}_1\}$ est l'ensemble de toutes les réunions d'une paire d'intervalles.
   (c) $d = 2$ et $\mathcal{G}_3 = \{g : g = \mathbb{1}_{[a,b]^2} \text{ avec } a, b \in \mathbb{R}\}$,
   (d) $d = 2$ et $\mathcal{G}_4 = \{g : g(x, y) = \mathbb{1}_{ax+by>0} \text{ avec } a, b \in \mathbb{R}\}$,
   (e) $d = 2$ et $\mathcal{G}_5 = \{g : g(x, y) = \mathbb{1}_{ax+by+c>0} \text{ avec } a, b, c \in \mathbb{R}\}$,
   (f) $d = 2$ et $\mathcal{G}_6 = \{g = 1_C : C \text{ est un sous-ensemble convexe de } \mathbb{R}^2\}$.
5. Le but de cet exercice est de démontrer le lemme de Sauer, énoncé juste avant le théorème de Vapnik-Chervonenkis. Soit $S = \{x_1, \ldots, x_N\}$ un sous-ensemble fini de $\mathcal{X}$ et $\mathcal{G} \subset \mathcal{F}(\mathcal{X}, \mathcal{Y})$ une classe de Vapnik-Chervonenkis. On cherche d'abord à démontrer l'inégalité

$$|T_{\mathcal{G}}(S)| \leq \sum_{k=0}^{V_{\mathcal{G}}(S)} C_N^k \tag{3.15}$$

par récurrence sur $|S|$.
   (a) Prouver que pour $N = 1$ et $V_{\mathcal{G}} = 0$, on a $|T_{\mathcal{G}}(S)| \leq 1$.
   (b) Prouver que pour $N = 1$ et $V_{\mathcal{G}} = 1$, on a $|T_{\mathcal{G}}(S)| \leq 2$. En déduire que (3.15) est vraie pour $N = 1$.
   (c) Soit N un entier strictement supérieur à 1. Supposons maintenant que l'inégalité (3.15) est vraie pour tout ensemble de cardinal $\leq N - 1$ et cherchons à démontrer sa validité pour un ensemble S de cardinal N. Fixons un $x_0 \in S$ et posons

$$S_0 = S \setminus \{x_0\}, \qquad g_0(x) = \begin{cases} 1, & \text{si } x = x_0 \\ 0, & \text{sinon} \end{cases}.$$

Soit

$$\mathcal{G}_1 = \Big\{ f \in \mathcal{F}(\mathcal{X}, \mathcal{Y}) : \exists\, g \in \mathcal{G} \text{ t. q. } f = g - g_0 + g g_0 \Big\}, \tag{3.16}$$

$$\mathcal{G}_2 = \Big\{ f \in \mathcal{G} : \ g(x_0) = 0 \text{ et } g + g_0 \in \mathcal{G} \Big\}. \tag{3.17}$$

Montrer que $|T_{\mathcal{G}}(S)| = |T_{\mathcal{G}_1}(S_0)| + |T_{\mathcal{G}_2}(S_0)|$.

(d) Vérifier que $V_{\mathcal{G}_1}(S_0) \leq V_{\mathcal{G}}(S)$.

(e) Vérifier que $V_{\mathcal{G}_2}(S_0) \leq V_{\mathcal{G}}(S) - 1$.

(f) En utilisant l'identité $C_N^k = C_{N-1}^{k-1} + C_{N-1}^k$, en déduire l'inégalité (3.15).

(g) Vérifier que pour tout $d, n \in \mathbb{N}$ et pour tout $j \in \{0, \ldots, d\}$,

$$n(d - j + 1) \geq n - j + 1.$$

En déduire que $C_d^k n^k \geq C_n^k$ pour tout $k \in \{0, \ldots, d\}$.

(h) Déduire des questions précédentes l'assertion du lemme 3.1.

4

Plus proches voisins et arbres de décision

Sommaire

4.1 Méthodes à base de partition

Nous avons vu au chapitre précédent qu'un prédicteur avec une erreur stochastique faible peut être construit en minimisant le risque empirique sur un ensemble de prédicteurs $\mathcal{G}$ bien choisi. Le choix de $\mathcal{G}$ est un sujet de recherche inépuisable, mais il existe quelques exemples devenus classiques en apprentissage statistique. Les plus populaires parmi ces exemples sont l'algorithme de k plus proches voisins (k nearest neighbor ou kNN) et les arbres de décision et de régression (classification and regression trees ou CART) tout les deux utilisant des classes $\mathcal{G}$ qui sont constituées des fonctions $g : \mathcal{X} \to \mathcal{Y}$ constantes par morceaux sur une partition de $\mathcal{X}$. Ce qui différencie ces deux algorithmes est la façon dont la partition en question est choisie.

Commençons tout d'abord par remarquer que dans les deux problèmes d'apprentissage supervisé – la classification binaire et la régression aux moindres carrés – on peut déterminer de façon explicite le prédicteur minimisant le risque empirique parmi tous les prédicteurs constants par morceaux sur une partition donnée de $\mathcal{X}$. En effet, soit $A_1, \dots, A_M$ des parties deux-à-deux disjointes de $\mathcal{X}$ telles que

$$\mathcal{X} = \bigsqcup_{m=1}^{M} A_m.$$

On note $\mathcal{A} = \{A_1, \dots, A_M\}$ et

$$\mathcal{G}_{\mathcal{A}} = \Big\{ g : \mathcal{X} \to \mathcal{Y} : \forall\, m = 1, \dots, M \quad g \text{ est constante sur } A_m \Big\}. \tag{4.1}$$

Il est important de souligner que la partition $\mathcal{A}$ peut être aléatoire ; elle peut notamment être choisie en fonction de l'échantillon d'apprentissage $\mathcal{D}_n$.

Théorème 4.1. *Soit $\mathcal{D}_n = \{Z_i = (X_i, Y_i); i = 1, \dots, n\} \subset (\mathcal{X} \times \mathcal{Y})^n$ un échantillon d'apprentissage et soit $\mathcal{A} = \{A_m; m = 1, \dots, M\}$ une partition de $\mathcal{X}$. Soit $\widehat{g}_{n,\mathcal{A}}$ le minimiseur du risque empirique sur $\mathcal{G}_{\mathcal{A}}$:*

$$\widehat{g}_{n,\mathcal{A}} \in \arg\min_{g \in \mathcal{G}_{\mathcal{A}}} \widehat{R}_n(g) = \arg\min_{g \in \mathcal{G}_{\mathcal{A}}} \frac{1}{n} \sum_{i=1}^{n} \ell\big(Y_i, g(X_i)\big), \tag{4.2}$$

où $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ est une fonction de perte donnée. Soit $N_m = \sum_{i=1}^n \mathbb{1}_{A_m}(X_i)$ le nombre d'exemples appartenant à la partie A_m et $\overline{Y}_{A_m} = \frac{1}{N_m} \sum_{i=1}^n Y_i \mathbb{1}_{A_m}(X_i)$ la moyenne des étiquettes observées dans la partie A_m.

1. *Dans le problème de classification binaire avec $\mathcal{Y} = \{0, 1\}$ et la perte $\ell(y, y') = \mathbb{1}(y \neq y')$, tout minimiseur de risque empirique est de la forme*

$$\forall\, m = 1, \dots, M;\ \forall\, x \in A_m \qquad \widehat{g}_{n,\mathcal{A}}(x) = \begin{cases} 1, & si \quad \overline{Y}_{A_m} > 1/2, \\ a_m, & si \quad \overline{Y}_{A_m} = 1/2, \\ 0, & si \quad \overline{Y}_{A_m} < 1/2, \end{cases} \tag{4.3}$$

   avec $a_m \in \{0, 1\}$. Inversement, quelle que soit la suite $(a_1, \dots, a_M) \in \{0, 1\}^M$, le prédicteur (4.3) minimise le risque empirique.

2. *Dans le problème de régression aux moindres carrés avec $\mathcal{Y} = \mathbb{R}$ et la perte $\ell(y, y') = (y - y')^2$, il existe un unique minimiseur de risque empirique donné par*

$$\widehat{g}_{n,\mathcal{A}}(x) = \sum_{m=1}^{M} \overline{Y}_{A_m} \mathbb{1}_{A_m}(x) = \sum_{m=1}^{M} \left\{ \frac{1}{N_m} \sum_{i=1}^{n} Y_i \mathbb{1}_{A_m}(X_i) \right\} \mathbb{1}_{A_m}(x). \tag{4.4}$$

Démonstration. **A compléter (faite en cours).** □

4.2 Algorithme kNN

Le cadre considéré est toujours celui d'apprentissage supervisé. Nous disposons de n copies indépendantes de couples entrée-sortie $Z_i = (X_i, Y_i) \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ avec $\mathcal{Y} = \{0, 1\}$ pour la classification binaire et $\mathcal{Y} \subset \mathbb{R}$ pour la régression. Pour l'algorithme kNN il est indispensable d'équiper l'espace des entrées $\mathcal{X}$ d'une métrique. Pour simplifier, on supposera tout au long de cette section que $\mathcal{X}$ est un sous-ensemble de $\mathbb{R}^d$ muni de la distance euclidienne.

Soit $k \geq 1$ un entier plus petit que la taille de l'échantillon : $k \leq n$. Pour chaque $x \in \mathbb{R}^d$ et pour chaque $i = 1, \dots, n$, on note $d_i(x)$ la distance entre x et X_i : $d_i(x) = \|X_i - x\|$. On définit la première statistique de rang $r_1(x)$ comme l'indice du plus proche voisin de x parmi $X_1, \dots, X_n$, c'est-à-dire

$$r_1(x) = i^* \qquad \text{si et seulement si} \qquad d_{i^*}(x) = \min_{1 \leq i \leq n} d_i(x) \quad \text{et} \quad d_{i^*}(x) < \min_{1 \leq i < i^*} d_i(x).$$

La seconde inégalité est nécessaire pour que le rang soit défini de façon unique. Lorsque $i^* = 1$, le minimum de la seconde inégalité porte sur un ensemble vide ; on considère alors que ce minimum est égal à $+\infty$. On définit le rang $r_k(x)$ par récurrence :

$$r_k(x) = i^* \quad \text{si et seulement si} \quad d_{i^*}(x) = \min_{\substack{1\le i\le n \\ i\neq r_1,\dots,r_{k-1}}} d_i(x) \quad \text{et} \quad d_{i^*}(x) < \min_{\substack{1\le i< i^* \\ i\neq r_1,\dots,r_{k-1}}} d_i(x). \tag{4.5}$$

Pour un entier $k \in [1, n]$ fixé, les statistiques de rang $r_1, \dots, r_n$ définies ci-dessus conduisent vers la partition $\mathcal{A}_k$ telle que

$$A_m = \Big\{ x \in \mathbb{R}^d : \big\{ r_1(x), \dots, r_k(x) \big\} = \varphi^m \Big\}, \qquad m = 1, \dots, \mathrm{C}_n^k$$

où C_n^k désigne le nombre de combinaisons de k éléments parmi n et $\{\varphi^1, \dots, \varphi^m\}$ sont ces combinaisons. En d'autres termes, les ensembles A_m sont les parties de $\mathbb{R}^d$ sur lesquelles l'application

$$x \mapsto \big\{ r_1(x), \dots, r_k(x) \big\}$$

est constante. Expliqué avec des mots simples, la partition $\mathcal{A}_k$ est caractérisée par les propriétés suivantes :

— pour deux $x, x' \in A_m$ les k plus proches voisins de x et de x' parmi $\{X_1, \dots, X_n\}$ sont les mêmes,

— alors que pour $x \in A_m$ et $x' \in A_{m'}$ (avec $m \neq m'$) les k plus proches voisins de x parmi $\{X_1, \dots, X_n\}$ sont différents des k plus proches voisins de x' parmi $\{X_1, \dots, X_n\}$.

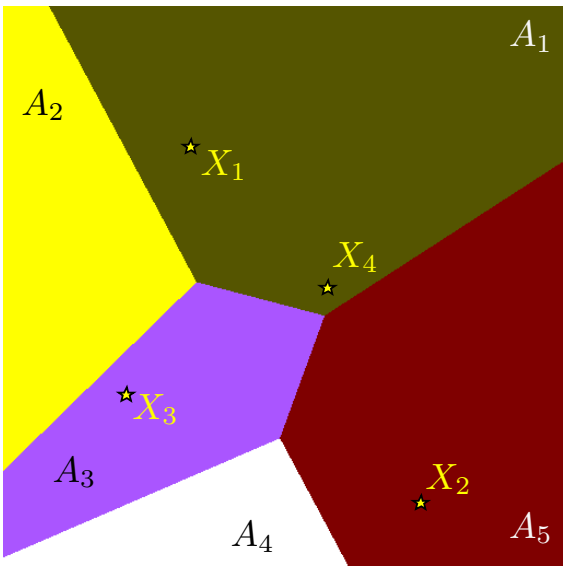


Figure 3 : *Un exemple de partition $\mathcal{A}_k$ de $\mathcal{X} = [0,1]^2$, avec $k = 2$, déterminée par un échantillon de taille $n = 4$. L'ensemble A_1 contient les points x pour lesquels les deux plus proches voisins sont X_1 et X_4, A_2 contient les x dont les deux plus proches voisins sont X_1 et X_3, …, A_5 contient les x dont les deux plus proches voisins sont X_2 et X_4, et finalement, A_6 est vide car il n'y a aucun point x dont les deux plus proches voisins sont X_1 et X_2.*

Définition 4.1. *Dans le problème de la régression aux moindres carrés, on dit que $\widehat{\eta}_k$ est un prédicteur fourni par l'algorithme de k plus proches voisins ou, pour abréger, le prédicteur*

kNN, si

$$\widehat{\eta}_{n,k}(x) = \sum_{m=1}^{\mathrm{C}_n^k} \mathbb{1}_{A_m}(x)\overline{Y}_{A_m},$$

où $\{A_1, \ldots, A_{\mathrm{C}_n^k}\}$ *est la partition définie ci-dessus.*
De même, on dit que $\widehat{g}_{n,k}(x)$ *est un classifieur fourni par l'algorithme de k plus proches voisins ou, pour abréger, le classifieur kNN, si*

$$\widehat{g}_{n,k}(x) = \mathbb{1}(\widehat{\eta}_{n,k}(x) > 1/2) = \sum_{m=1}^{\mathrm{C}_n^k} \mathbb{1}_{A_m}(x)\mathbb{1}_{]1/2,1]}(\overline{Y}_{A_m}).$$

Pour que le prédicteur kNN soit consistant, il faut choisir $k = k_n$ comme une fonction croissante de la taille de l'échantillon. Plus on a des données, plus la valeur du paramètre $k = k_n$ peut être élevée. Les petites valeurs de k induisent une grande volatilité du résultat et, par conséquent, favorisent le surapprentissage. A l'inverse, si k est très grand, la partition $\mathcal{A}_k$ est très grossière et le prédicteur qui en résulte est peu flexible. Il faut donc trouver la valeur de k garantissant une flexibilité suffisante du prédicteur tout en évitant le surapprentissage. Le résultat ci-dessous donne des conditions théoriques permettant d'atteindre cet objectif mais, en pratique, il est conseillé de choisir k par validation croisée.

Théorème 4.2 (Admis)**.** *Si* $k = k_n$ *tend vers* $+\infty$ *lorsque* $n \to +\infty$ *moins vite que n, c'est-à-dire* $k/n \to 0$*, alors le prédicteur kNN est consistant. Il en est de même pour le classifieur kNN.*

4.3 Arbres de décision et algorithme CART

Les arbres de décision, très utilisés en pratique, représentent un autre cas particulier des méthodes à partition. Le principe philosophique est toujours le même : *diviser pour conquérir*. La différence très importante par rapport à l'algorithme des k-plus proches voisins est que la partition générée par un arbre de décision est basée non seulement sur les variables explicatives X_i, mais aussi sur les étiquettes observées Y_i.

Il y a plusieurs façons de construire un arbre de décision à partir d'un échantillon d'apprentissage. Les deux algorithmes les plus répandus sont C4.5 et CART.

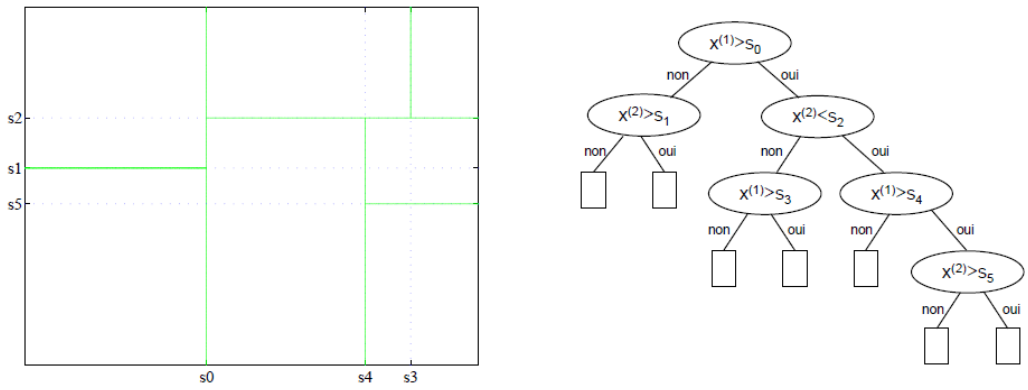


Figure 4 : *A droite : un arbre de décision correspondant à un problème de classification à deux variables explicatives quantitatives, c'est-à-dire* $X_i = (x_i^{(1)}, x_i^{(2)}) \in [0,1]^2$. *A gauche : la partition engendrée par l'arbre de décision..*

Dans un arbre de décision, chaque noeud correspond à un sous-ensemble A de $\mathcal{X}$ et un test T (également appelé critère de segmentation) auquel on soumet les variables explicatives $x \in \mathcal{X}$. Si le test peut donner lieu à K résultats différents, c'est-à-dire $T : A \to \{1, \ldots, K\}$, alors le noeud correspondant à (A, T) donne naissance à k noeuds-fils, tel que l'ensemble $A^{(k)}$ associé au $k^{\text{ème}}$ fils est $A^{(k)} = \{x \in A : T(x) = k\}$. La construction de l'arbre est donc un processus récursif, initialisé par le noeud racine qui correspond à $A = \mathcal{X}$. De façon schématique, ce processus peut être décrit de la façon suivante :

```
Initialisation : ARBRE = le noeud racine
Expansion : pour chaque noeud  N de l'arbre ARBRE
            si N ne vérifie pas la condition d'arrêt
                choisir un critère de segmentation T
                créer les noeuds-fils
                mettre à jour ARBRE = ARBRE + {noeuds-fils}
            fin si
Élagage : pour chaque noeud N de l'arbre ARBRE
            si N vérifie la condition d'élagage
               ARBRE = ARBRE - {le noeud N et ses descendants}
            fin si
```

Par conséquent, les différents algorithmes de construction d'un arbre de décision ont pour différences principales

(a) la manière dont les critères de segmentation sont choisis,

(b) le critère d'arrêt,

(c) le critère d'élagage.

Typiquement, le critère d'arrêt consiste à vérifier que l'une des conditions suivantes est vérifiée :

- la profondeur de l'arbre dépasse un seuil prescrit,
- le nombre de feuilles dans l'arbre dépasse un seuil prescrit,
- l'effectif du noeud est inférieur à un seuil prescrit.

Sans rentrer dans les détails techniques, mentionnons juste que l'étape d'élagage procède à la suppression d'une branche lorsque cette opération ne détériore pas significativement le risque estimé. Dans la plupart des cas, l'erreur estimée est calculé à l'aide d'un échantillon de test, différent de celui qui a servi dans l'étape d'expansion.

Une fois que l'arbre est construit, la règle de classification qui s'ensuit est simple :

- Pour un $x \in \mathcal{X}$, on détermine la feuille (le neoud terminal) qui le contient en parcourant l'arbre de haut en bas.
- Dans le problème de la classification, on affecte à x l'étiquette y correspondant à la classe majoritairement représentée parmi les exemples X_i qui «tombent» dans cette feuille.
- Dans le problème de la régression, on affecte à x la moyenne des étiquettes Y_i correspondant aux exemples X_i qui «tombent» dans cette feuille.

Dans ce qui suit, on présentera plus en détail le choix des tests de l'algorithme CART. Ce dernier, dont l'acronyme signifie « Classification And Regression Trees », s'attelle à un arbre de décision binaire (c'est-à-dire chaque noeud non-terminal a deux fils) utilisant comme critère de segmentation l'indice de Gini dans le cas de la classification et la variance dans le cas de la régression.

L'indice de Gini d'un neoud associé à un ensemble $A \subset \mathcal{X}$ est défini par :

$$G(A) = 1 - \overline{Y}_A{}^2 - (1 - \overline{Y}_A)^2 \qquad [0 \leq G(A) \leq 0.5, \ \forall A].$$

Notons qu'un noeud est bon pour la prédiction, si à quelques exceptions près les étiquettes des exemples associés à ce noeud sont les mêmes. L'intérêt de l'indice de Gini réside dans le fait qu'un noeud est bon si $G(A) \approx 0$, alors que pour un mauvais noeud $G(A) \approx 0.5$. En effet, d'une part, $G(A) = 0$ si et seulement si $\overline{Y}_A = 0$ ou $\overline{Y}_A = 1$. D'autre part, $G(A) = 0.5$ si et seulement si $\overline{Y}_A = 0.5$, ce qui implique que le noeud correspondant à A ne facilite pas la tache de prédiction.

On évalue la qualité d'un critère qui segmente l'ensemble A en deux parties disjointes A_1 et A_2 par le gain d'homogénéité :

$$I_G(A_1, A_2) = G(A) - qG(A_1) - (1 - q)G(A_2)$$

où $q = N_{A_1}/N_A$, la proportion des $X_i \in A$ qui se dirige vers A_1. Parmi toutes les partitions $\{A_1, A_2\}$ candidates de A, l'algorithme CART choisit celle qui maximise le gain d'homogénéité.

Dans le cas de la régression, le principe est le même à une seule différence près : l'indice de Gini est remplacé par la variance. C'est à dire, la qualité d'une segmentation est mesurée par

$$I_E(A_1, A_2) = E(A) - qE(A_1) - (1 - q)E(A_2)$$

où

$$E(A) = \frac{1}{N_A} \sum_{i:X_i \in A} (Y_i - \overline{Y}_A)^2.$$

4.4 Exercices

1. Le but de cet exercice est de montrer que l'algorithme kNN employé avec $k = 1$ n'est pas consistent. Pour cela, nous considérons le problème de classification binaire avec

$\mathcal{X} = [0,1]$ et $\mathcal{Y} = \{0;1\}$. On note P_X la loi marginale des X_i et suppose que

$$\eta^*(x) = P(Y_1 = 1 | X_1 = x) \equiv \frac{3}{4}, \qquad \forall x \in \mathcal{X}.$$

L'objectif des questions suivantes est de calculer le risque du classifieur oracle g_P^* ainsi que celui du classifieur kNN $\widehat{g}_{n,k}$ avec $k = 1$. On verra que ce dernier ne dépend pas de la taille de l'échantillon et est strictement plus grand que le risque de l'oracle.

(a) Montrer que pour tout application déterministe $g : \mathcal{X} \to \{0;1\}$, on a

$$R_P(g) = \mathbf{E}_{P_X}[\eta^*(X)] + \mathbf{E}_{P_X}[g(X)(1 - 2\eta^*(X))].$$

(b) En déduire que si $\eta^* \equiv 3/4$, alors le classifieur oracle (appelé aussi classifieur de Bayes) est donné par $g_P^* \equiv 1$ et son risque vaut $R_P(g_P^*) = 1/4$.

(c) Montrer que pour tout $g : \mathcal{X} \to \{0;1\}$, on a

$$R_P(g) = \frac{3}{4} - \frac{1}{2} \int_{\mathcal{X}} g(x)\, P_X(dx).$$

(d) Soit $\mathcal{D}_n = \{(X_i, Y_i); i = 1, \ldots, n\}$ et $\widehat{g}_{n,1}(x) = \widehat{g}_{\text{PPV}}(x, \mathcal{D}_n)$ le classifieur du plus proche voisin (PPV). Fixons $x \in \mathcal{X}$ et cherchons à calculer $\mathbf{E}_P[\widehat{g}_{\text{PPV}}(x, \mathcal{D}_n)]$, où l'espérance est par rapport à l'échantillon $\mathcal{D}_n$. Pour tout $i = 1, \ldots, n$, posons

$$Z_i = \begin{cases} 1, & \text{si } X_i \text{ est le PPV de x} \\ 0, & \text{sinon.} \end{cases}$$

Montrer que

$$\mathbf{E}_P[\widehat{g}_{\text{PPV}}(x, \mathcal{D}_n)] = \sum_{i=1}^{n} \mathbf{E}_P[Y_i Z_i]. \tag{4.6}$$

(e) Vérifier que pour tout i, Y_i est indépendant de $(X_1, \ldots, X_n)$. En déduire que Y_i et Z_i sont indépendantes.

(f) En utilisant la question précédente et la relation évidente $\sum_{i=1}^{n} Z_i = 1$ montrer que

$$\mathbf{E}_P[R_P(\widehat{g}_{\text{PPV}})] = \frac{3}{8}.$$

Conclure.

(g) Considérer le cas des 3 plus proches voisins $\widehat{g}_{3-\text{PPV}}$. Montrer que son risque moyen $\mathbf{E}_P[R_P(\widehat{g}_{3-\text{PPV}})]$ est égal à $21/64$.

(h) Passons maintenant au cas général d'un prédicteur kNN $\widehat{g}_{k-\text{PPV}}$. Soient $V_1, \ldots, V_k$ des variables aléatoires i.i.d. de loi de Bernoulli de paramètre $3/4$. Montrer que

$$\mathbf{E}_P[\widehat{g}_{k-\text{PPV}}(x, \mathcal{D}_n)] = \mathbf{P}(\overline{V}_k > 1/2).$$

En déduire que cette espérance tend vers 1 lorsque $k \to \infty$ et, par conséquent, le risque espéré $\mathbf{E}_P[R_P(\widehat{g}_{k-\text{PPV}})]$ tend vers le risque de l'oracle, c'est à dire vers $1/4$.

5

Méthode de convexification en classification binaire

Sommaire

5.1 Convexification de l'ensemble des classifieurs et de la perte

Le problème de minimisation du risque empirique :

$$\widehat{g}_{n,\mathcal{G}} \in \arg\min_{\mathcal{G}} \widehat{R}_n[g], \tag{5.1}$$

considéré dans les chapitres précédents, est souvent difficile à mettre en oeuvre à cause de la non convexité de l'ensemble $\mathcal{G}$ et la non convexité de l'application $\widehat{R}_n[g]$.

Avant de rentrer dans le vif du sujet, il convient de rappeler quelques notions basiques de l'analyse et de programmation convexes. Soit V un espace vectoriel et $C \subset V$. On dit que C est un ensemble convexe si pour tout $v_1, v_2 \in C$, le segment $[v_1, v_2] = \{v_1 + \alpha(v_2 - v_1) : \alpha \in [0,1]\}$ est inclus dans C. En d'autres termes, C est convexe si et seulement si

$$v_1, v_2 \in C,\ \alpha \in [0,1] \qquad \Longrightarrow \qquad \alpha v_1 + (1-\alpha) v_2 \in C.$$

On montre par récurrence que

$$v_1, \ldots, v_N \in C;\ \alpha_1, \ldots, \alpha_N \in \mathbb{R}_+ \qquad \Longrightarrow \qquad \frac{\alpha_1 v_1 + \ldots \alpha_N v_N}{\alpha_1 + \ldots + \alpha_N} \in C.$$

Soit C un sous-ensemble convexe de V. Une fonction $f : C \to \mathbb{R}$ est dite convexe si elle vérifie

$$f\big(\alpha v_1 + (1-\alpha) v_2\big) \leq \alpha f(v_1) + (1-\alpha) f(v_2), \qquad \forall v_1, v_2 \in C;\ \forall \alpha \in [0,1]. \tag{5.2}$$

On montre par récurrence que si $f : C \to \mathbb{R}$ est convexe, alors

$$v_1, \ldots, v_N \in C;\ \alpha_1, \ldots, \alpha_N \in \mathbb{R}_+ \quad \Longrightarrow \quad f\left(\frac{\alpha_1 v_1 + \ldots \alpha_N v_N}{\alpha_1 + \ldots + \alpha_N}\right) \leq \frac{\alpha_1 f(v_1) + \ldots \alpha_N f(v_N)}{\alpha_1 + \ldots + \alpha_N}.$$

Supposons maintenant que l'espace vectoriel V est muni d'une métrique : on peut mesurer la distance entre deux éléments de V. On montre que toute fonction convexe f définie sur un ensemble convexe fermé C atteint son minimum. Si f et C vérifient ces conditions, le problème qui consiste à déterminer $(f^*, v^*) \in \mathbb{R} \times C$ tels que

$$f^* = f(v^*) = \min_{v \in C} f(v)$$

est appelé problème de programmation convexe. On dit que C est l'espace de recherche et f est la fonction de coût. Contrairement au cas où la fonction de coût ou l'espace de recherche ne sont pas convexes, un problème de programmation convexe peut généralement être résolu en un temps polynomial. Les 3 exemples les plus courants de problème de programmation convexe sont la programmation linéaire, la programmation conique de second ordre et la programmation semi-définie positive.

Revenons au problème (5.1) de minimisation du risque empirique. Ce problème peut être résolu en un temps raisonnable si $\mathcal{G}$ est fini et son cardinal n'est pas trop grand, ou encore si le problème a une solution explicite simple, comme c'est le cas des méthodes à partition. Mais dès qu'on considère des ensembles $\mathcal{G}$ un tout petit peu plus complexes, le problème de minimisation (5.1) devient très difficile à résoudre en pratique.

Pour pallier ce défaut, on procède à la convexification du problème. Cela passe par la convexification de l'ensemble sur lequel porte la minimisation et par la convexification de la fonction de coût. Tout au long de ce chapitre, on considère le problème de classification binaire. Pour rendre les formules plus simple, on suppose que $\mathcal{Y} = \{-1; +1\}$ (au lieu de $\mathcal{Y} = \{0; 1\}$ considéré dans les chapitres précédents).

5.1.1 Convexification de l'espace de recherche

Nous étendons la définition de la fonction de prédiction à des fonctions quelconques à valeurs réelles de la manière suivante. Pour une fonction $h : \mathcal{X} \to \mathbb{R}$, on affecte à x l'étiquette 0 si $h(x) \leq 0$ et l'étiquette 1 si $h(x) > 0$. Cela équivaut à considérer le prédicteur binaire $g(x) = \operatorname{sgn}\big(h(x)\big)$, où la fonction signe est définie par

$$\operatorname{sgn}(u) = \begin{cases} +1, & u > 0, \\ -1, & u \leq 0 \end{cases}, \qquad u \in \mathbb{R}.$$

Ainsi, tout $h \in \mathcal{F}(\mathcal{X}, \mathbb{R})$ peut être considéré comme un prédicteur. L'avantage de l'ensemble $\mathcal{F}(\mathcal{X}, \mathbb{R})$ par rapport à $\mathcal{F}(\mathcal{X}, \{-1; +1\})$ est que le premier est convexe, alors que le second ne l'est pas. En effet, si $g_1, g_2 \in \mathcal{F}(\mathcal{X}, \{-1; +1\})$ et $\alpha \in]0, 1[$, alors $\alpha g_1 + (1 - \alpha) g_2 \mathcal{X} \to \{-1; (2\alpha - 1); (1 - 2\alpha); +1\}$ et, par conséquent, la combinaison convexe $\alpha g_1 + (1 - \alpha) g_2$ n'appartient pas nécessairement à $\mathcal{F}(\mathcal{X}, \{-1; +1\})$.

La perte de prédiction de la valeur y par $h(x)$ est

$$\begin{aligned}\ell\big(y,h(x)\big) &= \mathbb{1}\big(y \neq \operatorname{sgn}(h(x))\big)\\ &= \mathbb{1}(y=1)\mathbb{1}\big(1 \neq \operatorname{sgn}(h(x))\big) + \mathbb{1}(y=-1)\mathbb{1}\big(-1 \neq \operatorname{sgn}(h(x))\big)\\ &= \mathbb{1}(y=1)\mathbb{1}\big(h(x)\leq 0\big) + \mathbb{1}(y=-1)\mathbb{1}\big(h(x)>0\big)\\ &= \mathbb{1}(y=1)\mathbb{1}\big(yh(x)\leq 0\big) + \mathbb{1}(y=-1)\mathbb{1}\big(yh(x)<0\big).\end{aligned}$$

Il est clair que l'indicatrice de l'ensemble $\{(y;x) : yh(x) < 0\}$ est majorée par celle de l'ensemble $\{(y;x) : yh(x) \leq 0\}$. Par conséquent,

$$\begin{aligned}\ell\big(y,h(x)\big) &\leq \mathbb{1}(y=1)\mathbb{1}\big(yh(x)\leq 0\big) + \mathbb{1}(y=-1)\mathbb{1}\big(yh(x)\leq 0\big)\\ &\leq \mathbb{1}\big(yh(x)\leq 0\big).\end{aligned} \tag{5.3}$$

De la même manière, on vérifie que $\ell\big(y,h(x)\big) \geq \mathbb{1}\big(yh(x)<0\big)$. De plus, il n'est pas difficile de prouver que si $\mathcal{H} \subset \mathcal{F}(\mathcal{X},\mathbb{R})$ est tel que $P\big(h(X)=0\big)=0$, $\forall h \in \mathcal{H}$, alors le risque de la fonction de prédiction $\operatorname{sgn}(h)$ est donné par :

$$R_P\big(\operatorname{sgn}(h)\big) = \mathbf{E}_P\big[\phi_0\big(-Yh(X)\big)\big],$$

où on a utilisé la notation $\phi_0(x) = \mathbb{1}_{[0,+\infty)}(x)$.

5.1.2 Convexification de la perte

Pour pouvoir résoudre le problème de minimisation du risque empirique, il ne reste plus qu'à procéder à la convexification de la fonction de perte. Plus précisément, si $\mathcal{H}$ est un sous-ensemble convexe de $\mathcal{F}(\mathcal{X},\mathbb{R})$, on peut définir le minimiseur du risque empirique par

$$\widehat{h}_{n,\mathcal{H}} \in \arg\min_{h\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n} \phi_0\big(-Y_i h(X_i)\big).$$

Ceci est un problème d'optimisation dont l'espace de recherche est convexe mais la fonction de coût ne l'est pas. Pour y remédier, on remplace la fonction indicatrice ϕ_0 par une fonction convexe $\phi : \mathbb{R} \to \mathbb{R}$.

Définition 5.1. *Soit $\phi : \mathbb{R} \to \mathbb{R}$ une fonction convexe. On appelle ϕ-risque d'un classifieur $h : \mathcal{X} \to \mathbb{R}$ la quantité suivante :*

$$A_P(h) = \mathbf{E}_P\big[\phi\big(-Yh(X)\big)\big].$$

On appelle ϕ-classifieur de Bayes le prédicteur minimisant le ϕ-risque :

$$h_P^* \in \arg\min_{h\in\mathcal{F}(\mathcal{X},\mathbb{R})} A_P(h).$$

Soit $\mathcal{H} \subset \mathcal{F}(\mathcal{X},\mathbb{R})$. On appelle minimiseur (sur $\mathcal{H}$) du ϕ-risque empirique le prédicteur :

$$\widehat{h}_{n,\mathcal{H}} = \arg\min_{h\in\mathcal{H}} \underbrace{\frac{1}{n}\sum_{i=1}^{n} \phi\big(-Y_i h(X_i)\big)}_{\phi-\text{risque empirique } \widehat{A}_n(h)}. \tag{5.4}$$

L'idée principale est de choisir comme ϕ un majorant convexe de ϕ_0, de telle sorte qu'un classifieur ayant un ϕ-risque faible ait également un fable risque de classification. Les fonctions ϕ les plus utilisées en apprentissage sont :

$\phi(x) = (1+x)_+$	$\phi(x) = e^x$	$\phi(x) = \log_2(1+e^x)$	$\phi(x) = (1+x)^2$ ou $(1+x)_+^2$
perte charnière	perte de boosting	perte logistique	perte quadratique

L'intérêt de ces fonctions ϕ sera expliqué dans les sections suivantes par le biais des arguments théoriques liés à la consistance des algorithmes.

5.2 Consistance du minimiseur du ϕ-risque

Le choix de la fonction ϕ est cruciale pour la performance du minimiseur du ϕ-risque empirique. Lorsque ϕ est convexe et l'ensemble $\mathcal{H}$ l'est également, le problème (5.4) est un problème de programmation convexe ; il existe une multitude d'algorithmes pour résoudre un tel problème. Le calcul du minimiseur du ϕ-risque empirique est donc certainement plus facile à mettre en oeuvre que celui du MRE, c'est-à-dire, le temps de calcul pour $\widehat{h}_{n,\mathcal{H}}$ est généralement beaucoup plus réduit que celui de $\widehat{g}_{n,\mathcal{G}}$. Il est légitime alors de se demander si cette réduction du temps de calcul n'a pas été acquis au détriment de la qualité de prédiction.

Dans cette section, nous allons nous focaliser sur la qualité de classification du minimiseur du ϕ-risque empirique, la qualité étant mesurée par l'excès du risque. Le résultat suivant exhibe des conditions sur la fonction ϕ qui garantissent que le minimiseur du ϕ-risque empirique a un risque de classification proche de celui du classifieur de Bayes g_P^*.

Théorème 5.1. *Soit $\phi : \mathbb{R} \to \mathbb{R}$ une fonction convexe telle que*

$$u\big(\phi(u) - \phi(-u)\big) \geq 0, \qquad \forall u \in \mathbb{R}. \tag{5.5}$$

Définissons la fonction $\psi : [0,1] \to \mathbb{R}$ par $\psi(p) = \inf_{u\in\mathbb{R}} \big\{p\phi(-u) + (1-p)\phi(u)\big\}$. Si il existe $\gamma \in [0,1]$ et $c > 0$ tels que

$$|1-2p| \leq c\big[\phi(0) - \psi(p)\big]^\gamma, \qquad \forall p \in [0,1] \tag{5.6}$$

alors, quelle que soit la fonction de prédiction h,

$$R_P[\mathrm{sgn}(h)] - R_P[g_P^*] \leq c\big\{A_P(h) - A_P(h_P^*)\big\}^\gamma. \tag{5.7}$$

Démonstration du Théorème 5.1. Il découle de l'exercice 1 du premier chapitre, que

$$R_P[\mathrm{sgn}(h)] - R_P[g_P^*] = \mathbf{E}_P\Big[\mathbb{1}\big(\mathrm{sgn}(h(X)) \neq g_P^*(X)\big) \cdot \big|2\eta^*(X) - 1\big|\Big], \tag{5.8}$$

où η^* désigne la fonction de régression $\eta^*(x) = \mathbf{P}(Y = +1 | X = x)$. Or, on a déjà vu que $\mathbb{1}\big(\mathrm{sgn}(h(X)) \neq g_P^*(X)\big) \leq \phi_0\big(-h(X)g_P^*(X)\big)$ et, d'après la condition (5.6) du théorème,

$$\big|2\eta^*(X) - 1\big| \leq c\big[\phi(0) - \psi\big(\eta^*(X)\big)\big]^\gamma.$$

Par conséquent,

$$\begin{aligned} R_P[\mathrm{sgn}(h)] - R_P[g_P^*] &\leq c\,\mathbf{E}_P\Big[\phi_0\big(-h(X)g_P^*(X)\big)\cdot\big\{\phi(0)-\psi\big(\eta^*(X)\big)\big\}^\gamma\Big] \\ &= c\,\mathbf{E}_P\Big[\Big(\phi_0\big(-h(X)g_P^*(X)\big)\cdot\big\{\phi(0)-\psi\big(\eta^*(X)\big)\big\}\Big)^\gamma\Big] \\ &\leq c\,\Big(\mathbf{E}_P\Big[\phi_0\big(-h(X)g_P^*(X)\big)\cdot\big\{\phi(0)-\psi\big(\eta^*(X)\big)\big\}\Big]\Big)^\gamma, \end{aligned}$$

où la dernière inégalité résulte de l'inégalité de Jensen. Rappelons-nous que $g_P^*(x) = \mathbb{1}(2\eta^*(x)-1>0)$ et que $\phi_0(u) = \mathbb{1}(u \geq 0)$. Par conséquent,

$$\phi_0\big(-h(X)g_P^*(X)\big) = \mathbb{1}\big(h(X)(2\eta^*(X)-1)\leq 0\big).$$

Cela nous conduit vers :

$$R_P[\mathrm{sgn}(h)] - R_P[g_P^*] \leq c\,\Big(\mathbf{E}_P\Big[\mathbb{1}\big(h(X)(2\eta^*(X)-1)\leq 0\big)\cdot\big\{\phi(0)-\psi\big(\eta^*(X)\big)\big\}\Big]\Big)^\gamma. \quad (5.9)$$

Afin de poursuivre la démonstration, nous avons besoin de deux lemmes auxiliaires.

Lemme 5.1. *Soient $h \in \mathbb{R}$ et $\eta \in [0,1]$ deux nombres réels tels que $h(2\eta-1)\leq 0$. Alors*

$$\phi(0) \leq \eta\phi(-h) + (1-\eta)\phi(h).$$

Démonstration. Remarquons d'abord que pour $h=0$ l'assertion du lemme est évidente. Considérons maintenant le cas $h \neq 0$. Par convexité, on a

$$\begin{aligned} \phi(0) &\leq \frac{1}{2}\phi(h) + \frac{1}{2}\phi(-h) \\ &= \eta\phi(-h) + (1-\eta)\phi(h) + \frac{1}{2}\big(2\eta-1\big)\big(\phi(h)-\phi(-h)\big) \\ &= \eta\phi(-h) + (1-\eta)\phi(h) + \frac{1}{2h^2}\underbrace{h\big(2\eta-1\big)}_{\leq 0}\underbrace{h\big(\phi(h)-\phi(-h)\big)}_{\geq 0} \qquad (5.10) \\ &\leq \eta\phi(-h) + (1-\eta)\phi(h), \end{aligned}$$

où dans l'inégalité (5.10) on a utilisé l'hypothèse du lemme et la première condition du théorème. □

Avant d'énoncer et de prouver le deuxième lemme, remarquons que le ϕ-risque de toute fonction de prédiction h se calcule par la formule :

$$\begin{aligned} A_P(h) &= \mathbf{E}_P\Big[\phi(-h(X))\mathbb{1}(Y=1)\Big] + \mathbf{E}_P\Big[\phi(h(X))\mathbb{1}(Y=-1)\Big] \\ &= \mathbf{E}_P\Big(\mathbf{E}_P\Big[\phi(-h(X))\mathbb{1}(Y=1)\,\Big|\,X\Big]\Big) + \mathbf{E}_P\Big(\mathbf{E}_P\Big[\phi(h(X))\mathbb{1}(Y=-1)\,\Big|\,X\Big]\Big) \\ &= \mathbf{E}_P\Big(\phi(-h(X))\mathbf{E}_P\big[\mathbb{1}(Y=1)\,|\,X\big]\Big) + \mathbf{E}_P\Big(\phi(h(X))\mathbf{E}_P\big[\mathbb{1}(Y=-1)\,|\,X\big]\Big) \\ &= \mathbf{E}_P\Big(\phi(-h(X))\eta^*(X)\Big) + \mathbf{E}_P\Big(\phi(h(X))\big(1-\eta^*(X)\big)\Big). \qquad (5.11) \end{aligned}$$

Lemme 5.2. *La fonction ψ définie dans l'énoncé du théorème et le prédicteur de Bayes h_P^* sont reliés par la relation suivante :*

$$\mathbf{E}_P\Big[\psi\big(\eta^*(X)\big)\Big] = A_P(h^*).$$

Démonstration. Soit

$$h^0(x) \in \arg\min_{u\in\mathbb{R}} \Big\{\eta^*(x)\phi(-u) + (1-\eta^*(x))\phi(u)\Big\}. \tag{5.12}$$

Il découle de la définition même de la fonction ψ que

$$\psi\big(\eta^*(X)\big) = \eta^*(X)\phi\big(-h^0(X)\big) + \big(1-\eta^*(X)\big)\phi\big(h^0(X)\big).$$

En prenant l'espérance et en utilisant (5.11), on en déduit que $\mathbf{E}_P\big[\psi\big(\eta^*(X)\big)\big] = A_P(h^0)$.

D'autre part, toujours d'après la définition de ψ, pour tout prédicteur h, on a

$$\psi\big(\eta^*(X)\big) \le \eta^*(X)\phi\big(-h(X)\big) + \big(1-\eta^*(X)\big)\phi\big(h(X)\big).$$

En prenant l'espérance et en utilisant (5.11), il vient :

$$\mathbf{E}_P\big[\psi\big(\eta^*(X)\big)\big] \le \mathbf{E}_P\Big[\eta^*(X)\phi\big(-h(X)\big) + \big(1-\eta^*(X)\big)\phi\big(h(X)\big)\Big] = A_P(h), \qquad \forall h.$$

Cela implique que $A_P(h^0) = A_P(h_P^*)$. C'est exactement ce qu'il fallait démontrer. □

Retournons maintenant à la démonstration du théorème. En combinant (5.9) et le résultat du Lemme 5.1, il vient :

$$R_P[\mathrm{sgn}(h)] - R_P[g_P^*] \le c\left(\mathbf{E}_P\Big[\eta^*(X)\phi(-h(X)) + (1-\eta^*(X))\phi(h(X)) - \psi\big(\eta^*(X)\big)\Big]\right)^\gamma. \tag{5.13}$$

Pour conclure, il suffit de noter que d'après (5.11) et le Lemme 5.2, on a :

$$\mathbf{E}_P\Big[\eta^*(X)\phi(-h(X)) + (1-\eta^*(X))\phi(h(X)) - \psi\big(\eta^*(X)\big)\Big] = A_P(h) - A_P(h_P^*).$$ □

Notons d'abord que les conditions imposées sur ϕ dans le théorème précédent ne sont pas trop restrictives. Les quatres fonctions ϕ mentionnées à la fin de la section précédente vérifient ces conditions ; elles sont convexes, vérifient l'inégalité (5.5) et l'inégalité (5.6) avec les constantes c et γ suivantes :

perte	charnière	boosting	logistique	quadratique
$\phi(u)$	$(1+u)_+$	e^u	$\log_2(1+e^u)$	$(1+u)^2$ ou $(1+u)_+^2$
$\psi(p)$	$2\min(p,1-p)$	$2\sqrt{p(1-p)}$	$-p\log_2 p - (1-p)\log_2(1-p)$	$4p(1-p)$
c	1	$\sqrt{2}$	$\sqrt{2\ln(2)}$	1
γ	1	1/2	1/2	1/2

La démonstration de cette assertion est laissée en exercice (voir Exercice 1 à la fin de ce chapitre).

Soit $\widehat{h}_{n,\mathcal{H}}$ le minimiseur du ϕ-risque empirique pour une fonction ϕ satisfaisant les conditions du Théorème 5.1. Il en découle que

$$\underbrace{R_P[\mathrm{sgn}(\widehat{h}_{n,\mathcal{H}})] - R_P[g_P^*]}_{\text{excès de risque}} \le c\big(\underbrace{A_P[\widehat{h}_{n,\mathcal{H}}] - A_P[h_P^*]}_{\text{excès de }\phi\text{-risque}}\big)^\gamma. \tag{5.14}$$

Cette inégalité, combinée avec l'inégalité de Jensen, implique que si $\widehat{h}_{n,\mathcal{H}}$ est consistant par rapport au ϕ-risque, alors $\mathrm{sgn}(\widehat{h}_{n,\mathcal{H}})$ est consistant par rapport au risque de classification. Dans ces cas-là, la perte de qualité causée par la convexification du risque est très faible, surtout pour les grands échantillons.

5.3 Machines à Vecteurs Supports (SVM)

Les machines à vecteurs supports, plus connues sous le sigle SVM provenant de l'anglais "Support Vector Machines", est un cas particulier du minimiseur du risque empirique convexifié. La fonction ϕ la plus couramment utilisée pour les SVM est la perte charnière $\phi(x) = (1+x)_+$, mais toute autre ϕ parmi les exemples présentés plus tôt dans ce chapitre peut également être utilisée avec succès. La caractéristique principale des SVM porte non pas sur le choix de ϕ, mais sur le choix de l'ensemble $\mathcal{H}$ sur lequel se fait la minimisation du ϕ-risque. Dans le cas des SVM, $\mathcal{H}$ a la particularité d'être une boule dans un espace de Hilbert.

Commençons par un petit rappel des espaces de Hilbert. Un espace vectoriel $\mathcal{H}$ s'appelle un espace préhilbertien s'il est muni d'un produit scalaire $\langle \cdot, \cdot \rangle$. On appelle produit scalaire toute application de $\mathcal{H} \times \mathcal{H} \to \mathbb{R}$ qui est bilinéaire, symétrique et définie-positive. Un espace préhilbertien et complet s'appelle espace de Hilbert. Il est naturellement muni de la norme induit par le produit scalaire :

$$\|h\| = \sqrt{\langle h, h \rangle}.$$

Supposons maintenant que $\mathcal{H}_0 \subset \mathcal{F}(\mathcal{X}, \mathbb{R})$ est un espace de Hilbert. On peut définir le minimiseur du ϕ-risque empirique sur la boule de rayon $t > 0$ dans l'espace $\mathcal{H}_0$. Pour tout $t > 0$ fixé, on définit alors le prédicteur

$$\widehat{h}_t \in \arg \min_{h \in \mathcal{H}_0 : \|h\| \leq t} \frac{1}{n} \sum_{i=1}^{n} \phi\big(- Y_i h(X_i) \big). \tag{5.15}$$

Ce prédicteur $\widehat{h}_t$ peut être calculé en pratique en utilisant des algorithmes d'optimisation convexe, pour peu que la fonction ϕ soit convexe. La formule (5.15) s'applique uniquement dans le cas de la classification binaire, mais a son analogue dans le cas de régression aux moindres carrées ; il suffit simplement de remplacer $\phi\big(- Y_i h(X_i) \big)$ par $\phi\big(Y_i - h(X_i) \big)$. Dans le cas de la régression, la fonction ϕ la plus utilisée est la fonction quadratique $\phi(x) = x^2$ qui a l'avantage d'être convexe et continûment différentiable.

Le paramètre t figurant dans la formule (5.15) est un paramètre de lissage qui permet d'équilibrer l'erreur stochastique et l'erreur d'approximation. En effet, lorsque t est trop grand, le prédicteur $\widehat{h}_t$ est susceptible d'avoir une erreur stochastique trop grande et, donc, de sur-apprendre. A l'inverse, lorsque t est très proche de zéro, le prédicteur $\widehat{h}_t$ est choisi dans une famille trop restreinte. Cela implique, en général, que l'erreur d'approximation est trop grande et que le prédicteur sous-apprend. Pour trouver une bonne valeur de t, il n'y a pas de règle générale. On la choisit la plus souvent par validation croisée.

Lorsque $\mathcal{H}_0$ est un espace de Hilbert engendré par un noyau K [1], on appelle $\widehat{h}_t$ un prédicteur à vecteurs supports. On définira un peu plus tard les notions de noyau et d'un espace de Hilbert engendré par un noyau. Notons ici qu'il y a une équivalence entre la formulation contrainte (5.15) et la minimisation du ϕ-risque pénalisé :

$$\widetilde{h}_\lambda \in \arg \min_{h \in \mathcal{H}_0} \underbrace{\frac{1}{n} \sum_{i=1}^{n} \phi\big(- Y_i h(X_i) \big)}_{\widehat{A}_n(h)} + \lambda \|h\|^2. \tag{5.16}$$

1. On dit alors que $\mathcal{H}_0$ est un *reproducing kernel Hilbert space (RKHS.*

Intuitivement, l'équivalence de ces deux problèmes découle du fait que (5.16) est obtenu à partir de (5.15) par la méthode de multiplicateur de Lagrange. Le sens précis de l'équivalence est fourni par le résultat suivant.

Proposition 5.1. *Pour tout $t > 0$, il existe un $\lambda > 0$, tel que la solution $\widetilde{h}_\lambda$ est une solution de* (5.15)*. Inversement, pour tout $\lambda > 0$, il existe $t > 0$, tel que $\widehat{h}_t$ est une solution de* (5.16)*.*

Démonstration. On ne démontrera que la deuxième assertion de la proposition. Soit $\lambda > 0$ un réel fixé. Soit $\widetilde{h}_\lambda$ une solution quelconque du problème (5.16). Posons $t = \|\widetilde{h}_\lambda\|$ et montrons que $\widehat{h}_t$ est une solution de ((5.15)).

Nous avons, d'une part,

$$\|\widehat{h}_t\| \leq t = \|\widetilde{h}_\lambda\|. \tag{5.17}$$

D'autre part, comme $\widetilde{h}_\lambda$ appartient à la boule de $\mathcal{H}_0$ de rayon t centré en 0 et $\widehat{h}_t$ est le minimiseur du ϕ-risque sur cette boule, on a

$$\widehat{A}_n(\widehat{h}_t) \leq \widehat{A}_n(\widetilde{h}_\lambda). \tag{5.18}$$

En combinant (5.17) et (5.18), on obtient

$$\widehat{A}_n(\widehat{h}_t) + \lambda\|\widehat{h}_t\| \leq \widehat{A}_n(\widetilde{h}_\lambda) + \lambda\|\widetilde{h}_\lambda\|. \tag{5.19}$$

Comme $\widetilde{h}_\lambda$ est une solution de (5.16), cette dernière inégalité implique que $\widehat{h}_t$ est également une solution de (5.16). C'est exactement ce qu'on voulait démontrer. □

5.3.1 Espace à noyau reproduisant

Soit $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ une fonction symétrique et définie positive, c'est-à-dire

[symétrique]	$\forall x, x' \in \mathcal{X}$	$K(x, x') = K(x', x)$,
[déf.-pos.]	$\forall x_1, \ldots, x_m \in \mathcal{X}$	la matrice $[K(x_j, x_{j'}]_{j,j'=1}^m$ est définie-positive.

La fonction K est censée mesurer la similarité de deux features x et x' ; plus $K(x, x')$ est grand, plus x et x' se ressemblent. En utilisant cette fonction K, appelée noyau, on définit l'espace vectoriel

$$\mathcal{H}_0 = \Big\{ h \in \mathcal{F}(\mathcal{X}, \mathbb{R}) : \ \exists \alpha_1, \ldots, \alpha_m \in \mathbb{R},\ x_1, \ldots, x_m \in \mathcal{X} \text{ t.q. } h = \sum_{j=1}^m \alpha_j K(\cdot, x_j) \Big\}. \tag{5.20}$$

On vérifie facilement que c'est l'espace vectoriel engendré par les fonctions $x \mapsto K(x, a)$, pour tout $a \in \mathcal{X}$. L'espace $\mathcal{H}_0$ est un espace préhilbertien muni du produit scalaire

$$\langle h, h' \rangle = \sum_{j=1}^m \sum_{j'=1}^{m'} \alpha_j \alpha'_{j'} K(x_j, x'_{j'}), \qquad \text{si } h = \sum_{j=1}^m \alpha_j K(\cdot, x_j),\ h' = \sum_{j'=1}^{m'} \alpha'_{j'} K(\cdot, x'_{j'}). \tag{5.21}$$

On peut compléter $\mathcal{H}_0$ de manière à créer un espace de Hilbert. Par un abus de notation sans conséquence, ce dernier sera noté $\mathcal{H}_0$. On l'appelle espace de Hilbert à noyau reproduisant.

Définition 5.2. *On dit que le prédicteur $\widetilde{h}^{\mathrm{SVM}}$ est un SVM si pour un $\lambda > 0$ et pour un espace de Hilbert à noyau reproduisant $\mathcal{H}_0$, on a*

$$\widetilde{h}^{\mathrm{SVM}} \in \arg\min_{h\in\mathcal{H}_0} \Big(\widehat{A}_n(h) + \lambda\|h\|^2\Big) = \arg\min_{h\in\mathcal{H}_0} \Big(\frac{1}{n}\sum_{i=1}^{n} \phi\big(-Y_i h(X_i)\big) + \lambda\|h\|^2\Big). \tag{5.22}$$

On peut donner différents exemples d'espace de Hilbert à noyau reproduisant. L'exemple le plus simple correspond au cas où $\mathcal{X}$ est l'espace Euclidien $\mathbb{R}^d$ et le noyau K est le produit scalaire usuel $K(x,\widetilde{x}) = \langle x,\widetilde{x}\rangle = x^\top\widetilde{x}$. Dans ce cas, on vérifie aisément que $\mathcal{H}_0$ est l'ensemble de fonctions linéaires de $\mathcal{X} = \mathbb{R}^d$ dans $\mathbb{R}$. C'est-à-dire, à chaque h dans $\mathcal{H}_0$ correspond un $x_h \in \mathbb{R}^d$ tel que $h(x) = x_h^\top x$, pour tout $x \in \mathbb{R}^d$. De plus, on a $\|h\|_{\mathcal{H}_0}^2 = \|x_h\|_2^2$ où $\|\cdot\|_2$ désigne la norme Euclidienne.

La propriété très intéressante des espaces à noyau reproduisant, qui les rends attractifs pour les applications en apprentissage statistique, est que tout minimiseur du problème (5.22) est une combinaison linéaire des n fonctions $K(\cdot,X_1),\dots,K(\cdot,X_n)$ où $X_1,\dots,X_n$ est l'échantillon observé. Ce théorème, connu sous le nom «representer theorem», s'énonce de la manière suivante.

Théorème 5.2. *Soit $K : \mathcal{X}\times\mathcal{X} \to \mathbb{R}$ un noyau défini positif. Alors, pour tout $\lambda > 0$, il existe un vecteur $\boldsymbol{\alpha}^{\mathrm{SVM}} = (\alpha_1^{\mathrm{SVM}},\dots,\alpha_n^{\mathrm{SVM}})^\top \in \mathbb{R}^n$ tel que la solution $\widetilde{h}^{\mathrm{SVM}}$ de (5.22) s'écrit sous la forme*

$$\widetilde{h}^{\mathrm{SVM}}(\cdot) = \sum_{j=1}^{n} \alpha_j^{\mathrm{SVM}} K(\cdot, X_j). \tag{5.23}$$

Il découle de ce théorème que la résolution du problème d'optimisation (5.22), qui porte sur un espace potentiellement de dimension infinie, se ramène à la résolution d'un problème d'optimisation dans un espace de dimension n, où n est la taille de l'échantillon. En effet, on peut calculer $\boldsymbol{\alpha}^{\mathrm{SVM}}$ en résolvant le problème d'optimisation convexe

$$\boldsymbol{\alpha}^{\mathrm{SVM}} \in \arg\min_{\boldsymbol{\alpha}\in\mathbb{R}^n} \left\{ \frac{1}{n}\sum_{i=1}^{n} \phi(-Y_i\,\boldsymbol{\alpha}^\top \mathbf{K}\boldsymbol{e}_i) + \lambda\,\boldsymbol{\alpha}^\top\mathbf{K}\boldsymbol{\alpha} \right\}, \tag{5.24}$$

où $\mathbf{K}$ désigne la matrice de dimension $n\times n$ d'élément $\mathbf{K}_{i,j} = K(X_i,X_j)$ et $\boldsymbol{e}_i$ est le i-ème vecteur de la base canonique (c'est-à-dire $(\boldsymbol{e}_i)_j = \mathbb{1}(i=j)$).

Remarque 5.1.

1. *Lorsque ϕ est la fonction charnière ou la fonction quadratique, la résolution du problème (5.24) équivaut à celle d'un programme conique du second ordre (second-order cone programming). La plupart des solveurs ont des fonctions dédiées pour résoudre ce type de problèmes.*
2. *La solution du problème (5.24) n'est pas nécessairement unique, alors que celle de (5.22) l'est. Par conséquent, si $\widetilde{\boldsymbol{\alpha}}$ et $\widetilde{\boldsymbol{\alpha}}'$ sont deux vecteurs qui minimisent la fonction de coût de (5.24), alors on a*

$$\sum_{i=1}^{n} \widetilde{\alpha}_i K(x,X_i) = \sum_{i=1}^{n} \widetilde{\alpha}'_i K(x,X_i), \qquad \forall x\in\mathcal{X}.$$

5.4 Boosting

Nous allons maintenant présenter l'algorithme de boosting, qui fait partie de ce qu'on appelle des méthode d'«ensemble learning». Cela veut dire que le boosting est une méthode qui permet d'agréger une famille de méthodes de prédiction simples. Soit $\mathcal{G} \subset \mathcal{F}(\mathcal{X}, \{\pm 1\})$ la famille qu'on souhaite agréger. Par exemple, pour $\mathcal{X} = \mathbb{R}^d$, $\mathcal{G}$ peut être l'une des familles suivantes :

$$\mathcal{G}_1 = \Big\{ g(x) = \mathbb{1}(\boldsymbol{a}^\top \boldsymbol{x} > b) : \boldsymbol{a} \in \mathbb{R}^d, b \in \mathbb{R} \Big\}, \tag{5.25}$$

$$\mathcal{G}_2 = \Big\{ g : \mathcal{X} \to \{\pm 1\} : g \text{ est constant par morceaux sur une partition donnée } \mathcal{A} \Big\}. \tag{5.26}$$

Typiquement, la famille $\mathcal{G}$ considérée n'est pas un sous-ensemble convexe de l'espace vectoriel $\mathcal{F}(\mathcal{X}, \mathbb{R})$. Pour procéder à sa convexification, on considère la λ-enveloppe convexe de $\mathcal{G}$ définie par :

$$\mathcal{H}_\lambda = \left\{ \sum_{j=1}^m \alpha_j g_j : m \in \mathbb{N}, \substack{\boldsymbol{\alpha} \in \mathbb{R}_+^m, \\ (g_1, \dots, g_m) \in \mathcal{G}^m,} \sum_{j=1}^m \alpha_j \le \lambda \right\}. \tag{5.27}$$

Exercice 5.1. *Montrer que $\mathcal{H}_\lambda$ est un ensemble convexe quel que soit $\lambda > 0$.*

Le boosting, au sens large, est la méthode définie comme la solution du problème d'optimisation :

$$\widehat{h}_\lambda^{\text{boost}} \in \arg\min_{h \in \mathcal{H}_\lambda} \widehat{A}_n(h) = \arg\min_{h \in \mathcal{H}_\lambda} \frac{1}{n} \sum_{i=1}^n \phi\big(- Y_i h(X_i) \big). \tag{5.28}$$

Le paramètre $\lambda > 0$ est considéré comme un paramètre d'ajustement. En pratique, il est choisi par la validation croisée.

La consistance universelle de cette méthode a été démontrée par Lugosi et Vayatis (Annals of Statistics, 2004). Nous allons énoncer le théorème correspondant sans donner sa démonstration.

Théorème 5.3. *Si $\lambda = \lambda_n \to \infty$ est tel que $\lambda_n \phi'(\lambda_n) \sqrt{\ln n / n} \to \infty$ et $\mathcal{G}$ est tel que*
a) $\lim_{\lambda \to \infty} \inf_{h \in \mathcal{H}_\lambda} A(h) = \inf_{h \in \mathcal{F}(\mathcal{X}, \mathbb{R})} A(h)$,
b) $\mathcal{G}$ a une VC-dimension finie,
alors $\widehat{h}_\lambda^{\text{boost}}$ est universellement consistant.

Il convient de noter que même si le problème (5.28) est convexe, sa résolution demeure un problème difficile en raison du caractère infini-dimensionnel de l'espace des contraintes $\mathcal{H}_\lambda$. Pour pallier ce défaut, un algorithme itératif, AdaBoost, a été introduit. Il est fréquent dans la littérature d'utiliser le terme «boosting» pour l'algorithme itératif susmentionné plutôt que pour la solution du problème (5.28). Notons cependant qu'il n'y a pas de garantie formelle assurant que AdaBoost converge vers une solution de (5.28).

Remarque 5.2. *A l'étape d) de l'algorithme AdaBoost, on augmente les poids des observations mal classées et on diminue ceux des observations bien classées. Ainsi, le prédicteur de l'étape m essaye de se spécialiser sur les observations sur lesquelles ses prédécesseurs ont échoués.*

Input : données $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n) \in \mathbb{R}^p \times \{\pm 1\}$ et prédicteurs simples $\mathcal{G}$
Parameter : nombre d'itérations M,
Output : prédicteur $\widehat{g}$

1: Initialiser les poids des observations:

$$w_i = \frac{1}{n}, \qquad i = 1, \ldots, n.$$

2: Pour $m = 1, \ldots, M$:

a) entraîner un classifieur MRE avec les poids w_i:

$$\widehat{g}_m \in \arg\min_{g \in \mathcal{G}} \sum_{i=1}^{n} w_i \mathbb{1}\big(y_i \neq g(\boldsymbol{x}_i)\big).$$

b) Calculer l'erreur normalisée

$$\widehat{e}_m = \frac{\sum_{i=1}^{n} w_i \mathbb{1}\big(y_i \neq \widehat{g}_m(\boldsymbol{x}_i)\big)}{\sum_i w_i}.$$

c) Calculer le coefficient de $\widehat{g}_m$:

$$\widehat{\alpha}_m = \frac{1}{2} \ln\Big(\frac{1 - \widehat{e}_m}{\widehat{e}_m}\Big).$$

d) Mettre à jour les poids:

$$w_i \leftarrow w_i \exp\{\widehat{\alpha}_m \mathbb{1}\big(y_i \neq \widehat{g}_m(\boldsymbol{x}_i)\big)\}, \qquad i = 1, \ldots, n.$$

3: Retourner $\widehat{g}(\boldsymbol{x}) = \operatorname{sgn}\Big(\sum_{m=1}^{M} \widehat{\alpha}_m \widehat{g}_m(\boldsymbol{x})\Big)$.

FIGURE 5.1 – L'algorithme AdaBoost

Remarque 5.3. *Lorsque $\mathcal{G}$ est la classe des prédicteurs constants par morceaux sur les parties d'une partition, l'étape a) de l'algorithme AdaBoost peut être effectue de manière explicite.*

Remarque 5.4. *L'algorithme AdaBoost ne contient plus le paramètre d'ajustement λ présent dans la formulation variationnelle* (5.28)*. Ce paramètre est remplacé par M, le nombre de composantes à inclure dans la combinaison linéaire finale.*

Nous n'allons pas expliquer en détail les origines de cet algorithme, mais justifierons simplement les étapes c) et d) de l'algorithme. Ces étapes découlent de la proposition suivante.

Proposition 5.2. *Soit $h^\circ \in \mathcal{F}(\mathcal{X}, \mathbb{R})$ une fonction de prédiction donnée et soit $\phi(u) = e^u$ la perte exponentielle (ou de boosting). La solution du problème*

$$(\widehat{\alpha}, \widehat{g}) \in \arg\min_{\alpha \geq 0, g \in \mathcal{G}} \widehat{A}_n(h^\circ + \alpha g)$$

est donnée par

$$\widehat{g} \in \arg\min_{g\in\mathcal{G}} \sum_{i=1}^{n} e^{-Y_i h^\circ(X_i)} \mathbb{1}(Y_i \neq g(X_i)),$$
$$\widehat{\alpha} = \frac{1}{2} \ln\Big(\frac{1-\widehat{e}}{\widehat{e}}\Big), \quad \textit{où} \quad \widehat{e} = \frac{\sum_i e^{-Y_i h^\circ(X_i)} \mathbb{1}(Y_i \neq \widehat{g}(X_i))}{\sum_i e^{-Y_i h^\circ(X_i)}}.$$

Démonstration. Pour alléger les formules, posons

$$w_i = \frac{e^{-Y_i h^\circ(X_i)}}{\sum_\ell e^{-Y_\ell h^\circ(X_\ell)}}, \quad i = 1, \dots, n.$$

En utilisant la définition du ϕ-risque empirique, on a

$$\begin{aligned}(\widehat{\alpha}, \widehat{g}) &\in \arg\min_{\alpha\geq 0, g\in\mathcal{G}} \frac{1}{n} \sum_{i=1}^{n} \exp\Big\{ - Y_i h^\circ(X_i) - \alpha Y_i g(X_i)\Big\} \\ &= \arg\min_{\alpha\geq 0, g\in\mathcal{G}} \sum_{i=1}^{n} w_i e^{-\alpha Y_i g(X_i)} \\ &= \arg\min_{\alpha\geq 0, g\in\mathcal{G}} \sum_{i=1}^{n} w_i e^{-\alpha Y_i g(X_i)} \Big(\mathbb{1}(Y_i = g(X_i)) + \mathbb{1}(Y_i \neq g(X_i))\Big).\end{aligned}$$

Comme $Y_i \in \{\pm 1\}$ et $g(X_i) \in \{\pm 1\}$ pour tout i, il est clair que

$$\begin{aligned} Y_i = g(X_i) \quad &\Longleftrightarrow \quad Y_i g(X_i) = 1, \\ Y_i \neq g(X_i) \quad &\Longleftrightarrow \quad Y_i g(X_i) = -1.\end{aligned}$$

Cela implique que

$$(\widehat{\alpha}, \widehat{g}) \in \arg\min_{\alpha\geq 0, g\in\mathcal{G}} \Big\{ e^{-\alpha} \sum_{i=1}^{n} w_i \mathbb{1}(Y_i = g(X_i)) + e^{\alpha} \sum_{i=1}^{n} w_i \mathbb{1}(Y_i \neq g(X_i))\Big\} \tag{5.29}$$
$$= \arg\min_{\alpha\geq 0, g\in\mathcal{G}} \Big\{ e^{-\alpha} \sum_{i=1}^{n} w_i \big(1 - \mathbb{1}(Y_i \neq g(X_i))\big) + e^{\alpha} \sum_{i=1}^{n} w_i \mathbb{1}(Y_i \neq g(X_i))\Big\} \tag{5.30}$$
$$= \arg\min_{\alpha\geq 0, g\in\mathcal{G}} \Big\{ \big(e^{\alpha} - e^{-\alpha}\big) \sum_{i=1}^{n} w_i \mathbb{1}(Y_i \neq g(X_i)) + e^{-\alpha}\Big\}. \tag{5.31}$$

Nous commençons par minimiser l'expression obtenue par rapport à $g \in \mathcal{G}$. Comme $\alpha > 0$, on a $e^{\alpha} - e^{-\alpha} > 0$ et, par conséquent, le point de minimum de l'expression (5.31) peut être obtenu en résolvant le problème

$$\widehat{g} \in \arg\min_{\alpha\geq 0, g\in\mathcal{G}} \sum_{i=1}^{n} w_i \mathbb{1}(Y_i \neq g(X_i)). \tag{5.32}$$

En insérant ce prédicteur $\widehat{g}$ dans (5.31) et en utilisant le fait que

$$\sum_{i=1}^{n} w_i \mathbb{1}(Y_i \neq \widehat{g}(X_i)) = \widehat{e},$$

on voit que le point de minimum $\widehat{\alpha}$ est solution de

$$\widehat{\alpha} \in \arg\min_{\alpha>0} \Big\{ \underbrace{(e^{\alpha} - e^{-\alpha})\widehat{e} + e^{-\alpha}}_{G(\alpha)} \Big\}.$$

Il est clair que $G'(\alpha) = (e^{\alpha}+e^{-\alpha})\widehat{e}-e^{-\alpha}$ et que $G''(\alpha) = (e^{\alpha}-e^{-\alpha})\widehat{e}+e^{-\alpha} = e^{\alpha}\widehat{e}+e^{-\alpha}(1-\widehat{e}) \geq 0$. Il en résulte que G est une fonction convexe et que son minimum est atteint au point α qui vérifie $G'(\alpha) = 0$. La solution de cette équation est $\alpha = 1/2 \ln (1-\widehat{e})/\widehat{e}$. Cela entraîne le résultat désiré. □

5.5 Exercices

1. Soit ϕ une fonction convexe et soit

$$\psi(p) = \min_{u\in\mathbb{R}} \Big(p\phi(u) + (1-p)\phi(-u) \Big), \qquad \forall p \in [0,1].$$

   (a) Montrer que $\phi(0) = \psi(1/2)$.

   (b) Vérifier l'inégalité $\psi(p) \leq \phi(0)$ et en déduire que la condition (5.6) équivaut à

$$\psi(1/2) - \psi(p) \geq \Big(\frac{2}{c} \big| 1/2 - p \big| \Big)^{1/\gamma}. \tag{5.33}$$

   (c) Montrer que $\psi(p) = \psi(1-p)$. En déduire que si l'inégalité (5.33) est satisfaite pour tout $p \in [0, 1/2]$, alors elle l'est pour tout $p \in [0, 1]$.

   (d) Soit ψ une fonction continûment différentiable sur $[0, 1/2]$ telle que pour une constante $a > 0$, $\psi'(p) \geq a$ pour tout $p \in [0, 1/2]$. Montrer que la condition (5.6) est satisfaite avec $\gamma = 1$ et $c = 2/a$. En déduire que la fonction charnière vérifie la condition précitée avec $\gamma = 1$ et $c = 1$.

   (e) Montrer que ψ est une fonction concave et que si $\psi'(1/2)$ existe alors $\psi'(1/2) = 0$.

   (f) Soit ψ une fonction deux fois continûment différentiable sur $[0, 1/2]$ avec

$$\sup_{u\in]0,1/2[} \psi''(u) = -a < 0.$$

   Montrer que la condition (5.6) est remplie avec $\gamma = 1/2$ et $c = \sqrt{8/a}$.

   (g) Vérifier qu'on a le tableau suivant

perte	boosting	logistique	quadratique
$\phi(u)$	e^u	$\log_2(1+e^u)$	$(1+u)^2$
$\psi(p)$	$2\sqrt{p(1-p)}$	$-p\log_2 p - (1-p)\log_2(1-p)$	$4p(1-p)$
$\psi''(p)$	$-0.5[p(1-p)]^{-3/2}$	$-[p(1-p)\ln 2]^{-1}$	-8
$\sup \psi''(p)$	-4	$-4/\ln 2$	-8

   et en déduire que les trois fonctions de pertes présentes dans ce tableau vérifient la condition (5.6) avec $\gamma = 1/2$ et une constante c que l'on déterminera.