Combinaison optimale de classifieurs binaires : solution logique sans algorithme et minimisation de risques convexifiés

Olivier LAFITTE[1,3], Jean-Marc BROSSIER[2,3]

[1]Université Sorbonne Paris Nord, LAGA, UMR 7539. F-93430 Villetaneuse, France

[2]Univ. Grenoble Alpes, CNRS, Grenoble INP*, GIPSA-Lab, 38000 Grenoble, France.
*Institute of Engineering Univ. Grenoble Alpes

[3]CNRS-CRM. Université de Montréal. Canada

lafitte@crm.umontreal.ca, jean-marc.brossier@gipsa-lab.grenoble-inp.fr

Résumé – Une idée classique en apprentissage supervisé consiste à calculer un classifieur fort en combinant des classifieurs faibles, ADABOOST en est une implémentation. Nous étudions ce problème dans le cas particulier de trois classifieurs et de deux classes en introduisant une représentation originale du coût basée sur une table de vérité. Cette approche permet d'identifier directement le classifieur résultant (et de calculer sa performance) sans avoir besoin d'exécuter un algorithme numérique. D'autre part, les substituts convexes étant très largement utilisés, quelques uns ont été comparés avec cette approche, et les différences entre les classifieurs résultants, la robustesse de ces résultats ainsi que ceux de la méthode logique sont explorés.

Abstract – A classical idea in supervised learning is to compute a strong classifier by combining weak classifiers, ADABOOST is an implementation of this. We study this problem in the particular case of three classifiers and two classes by introducing an original representation of the cost based on a truth table. This approach allows to directly identify the resulting classifier (and to compute its performance) without the need to to execute a numerical algorithm. On the other side, since usually convex substitutes are widely used, some have been compared with this approach, and the robustness between the resulting classifiers and the stability of these results are explored.

1 Introduction

Nous traitons d'apprentissage supervisé avec $\mathcal{X} = \mathbb{R}^d$ pour espace des caractéristiques et $\mathcal{Y} = \{-1, +1\}$ comme ensemble d'étiquettes attachées à deux classes. Le but est de classer un objet (décider si son étiquette est ± 1) à partir de $x \in \mathcal{X}$. Un algorithme supervisé utilise $\mathcal{S} = \{(x_i, y_i) \subset \mathcal{X} \times \mathcal{Y}, i = 1 \cdots n\}$ pour estimer un classifieur, *i.e.* une fonction $h : \mathcal{X} \to \mathcal{Y}$.

La base d'exemples $\mathcal{S}$ est supposée être obtenue par échantillonnage i.i.d. selon une loi inconnue $\mathbb{P}$ sur $\mathcal{X} \times \mathcal{Y}$. Une mesure habituelle de la fiabilité est le risque de fausse décision : $L_{\mathbb{P}}(h) = \mathbb{E}_{\mathbb{P}} 1_{Yh(X)<0}$ où 1_A vaut 1 si A est vrai, 0 sinon. En estimant $\mathbb{P}$ à partir de $\mathcal{S}$ par $\hat{\mathbb{P}}_n = n^{-1} \sum_{(x_i,y_i)\in\mathcal{S}} \delta_{(x_i,y_i)}$ le risque empirique s'écrit

$$L_{\hat{\mathbb{P}}_n}(h) = \mathbb{E}_{\hat{\mathbb{P}}_n} 1_{Yh(X)<0} = n^{-1} \sum_{i=1}^{n} 1_{y_i h(x_i)<0.}$$

On considère ici des classifieurs faibles, c'est-à-dire qui renvoient une décision correcte dans un peu plus de la moitié des cas. Cette notion a été introduite par [5] qui propose d'obtenir une classification forte en combinant des classifieurs faibles.

En combinant 3 classifieurs $(G_1, G_2, G_3) = \mathbf{G}$, on construit le classifieur $h = \text{sign}(\boldsymbol{\beta}^T \mathbf{G})$, avec $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3) \in \mathbb{R}^3$, dont le risque s'écrit :

$$\mathcal{R}_{1_{\bullet<0}}(\boldsymbol{\beta}, \mathcal{S}) := L_{\hat{\mathbb{P}}_n}(\text{sign}(\boldsymbol{\beta}^T \mathbf{G})).$$

Nous proposons une analyse fondée sur les accords et contradictions entre les G_i pour déterminer des classifieurs de risque empirique minimum. Cette approche conduit à un calcul de la solution qui ne requiert pas d'algorithme numérique.

Pour comparer cette approche aux approches habituelles, nous étudierons dans les sections 4 et 5 des solutions obtenues par minimisation de risques convexifiés suffisamment réguliers.

2 Table de vérité et risque empirique

Un rôle important [2] est joué par le fait d'avoir simultanément au moins deux points où, pour un point x_p de $\mathcal{X}$ dans les exemples $\mathcal{S}$, les classifieurs G_j, $j \in J_1$ sont vrais, et tous les autres sont faux, et pour un autre point x_q de $\mathcal{X}$ dans l'ensemble d'apprentissage les classifieurs G_j, $j \in J_1$ sont faux et tous les autres sont vrais. Cette situation peut être un souci pour la qualité de l'apprentissage et conduit à introduire une table de vérité.

Puisque $y_i \in \pm 1$ et que chaque $G_j(x_i) \in \pm 1$, $y_i G_j(x_i)$ vaut $+1$ si G_j est vrai et -1 si G_j est faux. Nous utilisons donc la notation $+1$ pour $G_j(x_i)$ vrai et -1 pour $G_j(x_i)$ faux.

Table de vérité. Si chaque exemple de $\mathcal{S}$ est classé par p classifieurs binaires, $\mathcal{S}$ peut être décomposé en une partition à 2^p éléments qui donnera 2^p colonnes dans une table de taille $p \times 2^p$. Notons que $p = 2$ ne conduit à aucune amélioration par

combinaison, nous illustrons ainsi la méthode proposée dans le cas $p = 3$: pour trois classifieurs G_1, G_2, G_3, les 3 valeurs (± 1) binaires, qui sont pour chaque exemple i les signes de $y_i G_j(x_i)$, peuvent être représentées par une colonne à valeurs binaires qui appartient à un ensemble de $2^3 = 8$ colonnes possibles. Pour chacune des 8 colonnes, $\boldsymbol{\beta}^T \mathbf{G}$ est associé à l'une des 8 quantités $\pm X_0, \cdots, \pm X_3$ définies par

$$\begin{array}{rcl} X_1 & = & -\beta_1 + \beta_2 + \beta_3, \\ X_2 & = & +\beta_1 - \beta_2 + \beta_3, \\ X_3 & = & +\beta_1 + \beta_2 - \beta_3 \\ X_0 & = & +\beta_1 + \beta_2 + \beta_3 = X_1 + X_2 + X_3 \end{array} \quad (1)$$

Ainsi, quel que soit n, le comportement global du système composé de 3 classifieurs est caractérisé par un tableau à 3 lignes et 8 colonnes :

	n_0	m_0	n_1	m_1	n_2	m_2	n_3	m_3
G_1	-1	$+1$	$+1$	-1	-1	$+1$	-1	$+1$
G_2	-1	$+1$	-1	$+1$	$+1$	-1	-1	$+1$
G_3	-1	$+1$	-1	$+1$	-1	$+1$	$+1$	-1
$\boldsymbol{\beta}^T \mathbf{G}$	$-X_0$	X_0	$-X_1$	X_1	$-X_2$	X_2	$-X_3$	X_3

où les n_j, m_j comptent le nombre d'occurrences des configurations correspondantes dans $\mathcal{S}$. Les colonnes forment une partition de $\mathcal{S}$, donc $n = n_0+m_0+n_1+m_1+n_2+m_2+n_3+m_3$.

Par exemple, la colonne m_1 correspond aux exemples pour lesquels le classifieur G_1 est faux ($y_i G_1(x_i) = -1$) et les deux autres sont vrais ($y_i G_{2,3}(x_i) = +1$), tandis que pour la colonne n_1 c'est exactement le contraire (G_1 est vrai et les autres classifieurs sont faux).

Pour $\mathbf{G}$ donné, le comportement des 3 classifieurs sur $\mathcal{S}$ ainsi que le risque sont entièrement caractérisés par l'octuplet $(n_0, m_0, n_1, m_1, n_2, m_2, n_3, m_3)$.

Notons que si G_j fait partie d'une famille à paramètre continu, une faible variation du paramètre conduira au même octuplet.

Le fait que G_1 soit faible se traduit par $m_0+n_1+m_2+m_3 > n_0 + m_1 + n_2 + n_3 > 0$ (le dernière inégalité signifie que G_1 n'est pas parfait). Des conditions similaires valent pour G_2 et G_3. De plus, 2 classifieurs ne doivent pas être identiques.

Expression du risque associée à la table de vérité. Le risque s'écrit donc (hors frontières $X_j = 0$)

$$\begin{array}{rcl} n\mathcal{R}_{1_{\bullet<0}}(\boldsymbol{\beta}, \mathcal{S}) & := & nL_{\hat{\mathbb{P}}_n}(\mathrm{sign}(\boldsymbol{\beta}^T \mathbf{G})) \\ & = & \displaystyle\sum_{i=0}^{3} n_j 1_{X_j>0} + m_j 1_{X_j<0}, \end{array} \quad (2)$$

formulation qui apparie naturellement X_j and $-X_j$.

Remarque sur la généralisation au cas de p classifieurs. A partir de $\beta_1, \cdots, \beta_p$, on construit 2^{p-1} combinaisons linéaires de la forme $\sum_i \epsilon_i \beta_i$ avec $\epsilon_i = \pm 1$, étant entendu qu'à une combinaison est aussi associée son opposée, chaque combinaison est associée à un nombre d'exemples, ce qui permet de construire une généralisation de l'octuplet et donc du risque associé à cette nouvelle table de vérité. Dans la suite, on considère $p = 3$.

3 Minimiseur du risque empirique

D'après la table de vérité, le nombre d'erreurs de chaque classifieur peut s'exprimer par : $n_0 + m_1 + n_2 + n_3$ pour G_1, $n_0 + n_1 + m_2 + n_3$ pour G_2, $n_0 + n_1 + n_2 + m_3$ pour G_3. Ces expressions peuvent être étendues au calcul du nombre d'erreurs de tout classifieur de la forme $h = \mathrm{sign}(\boldsymbol{\beta}^T \mathbf{G})$ puisque, d'après (2), ce nombre se calcule directement en déterminant les signes des X_j associés aux β_j (hors frontières $X_j = 0$).

Classifieur logique et minimisation du risque empirique. Soient $(\beta_1, \beta_2, \beta_3) \in \mathbb{R}^3$ tels que $X_0 X_1 X_2 X_3 \neq 0$. Chacun des 4 termes $n_j 1_{X_j>0} + m_j 1_{X_j<0}$ qui composent $n\mathcal{R}_{1_{\bullet<0}}(\boldsymbol{\beta}, \mathcal{S})$ vaut n_j ou m_j selon que X_j est positif ou négatif.

Le risque $n\mathcal{R}_{1_{\bullet<0}}(\boldsymbol{\beta}, \mathcal{S})$ est donc une fonction des β_i constante par morceaux à valeurs dans un ensemble fini d'au plus 2^4 éléments. Mais le signe de $X_0 = X_1 + X_2 + X_3$ étant imposé lorsque X_1, X_2, X_3 sont de même signe, deux états sont impossibles et $n\mathcal{R}_{1_{\bullet<0}}(\boldsymbol{\beta}, \mathcal{S})$ appartient à la liste :

$$(L) \left\{ \begin{array}{cc} m_1 + n_2 + n_3 + n_0, & n_1 + m_2 + m_3 + m_0, \\ n_1 + m_2 + n_3 + n_0, & m_1 + n_2 + m_3 + m_0, \\ n_1 + n_2 + m_3 + n_0, & m_1 + m_2 + n_3 + m_0, \\ n_1 + n_2 + n_3 + n_0, & n_1 + m_2 + m_3 + n_0, \\ m_1 + n_2 + m_3 + n_0, & m_1 + n_2 + n_3 + m_0, \\ n_1 + n_2 + m_3 + m_0, & m_1 + m_2 + m_3 + m_0, \\ n_1 + m_2 + n_3 + m_0, & m_1 + m_2 + n_3 + n_0 \end{array} \right\}.$$

Les trois premiers éléments de chaque colonne correspondent aux nombres d'erreurs ou de décisions correctes de G_1, G_2, G_3 respectivement. A chacune de ces valeurs possibles du risque (nombre d'erreur du classifieur h sur S) correspondent des signes des X_j, donc une région pour $(\beta_1, \beta_2, \beta_3)$. Le classifieur associé à un $(\beta_1, \beta_2, \beta_3)$ pour lequel le risque empirique $\mathcal{R}_{1_{\bullet<0}}(\boldsymbol{\beta}, \mathcal{S})$ est minimum est appelé classifieur logique.

Remarquons que

— Chaque élément de la liste correspond à $\boldsymbol{\beta}$ dans une région de $\mathbb{R}^3$ donc à une infinité de $\boldsymbol{\beta}$.

— Le minimum peut être obtenu en plusieurs éléments de la liste. Dans ce cas, le domaine de définition en $\boldsymbol{\beta}$ du classifieur logique est l'union des régions associées à chaque élément.

Algorithme pour trois classifieurs. Pour trois classifieurs G_1, G_2, G_3 et une base d'exemples S donnés, le calcul d'un classifieur logique, c'est-à-dire d'un minimiseur du risque empirique sur S, s'effectue donc de la manière suivante :

1. calculer l'octuplet $(n_0, m_0, n_1, m_1, n_2, m_2, n_3, m_3)$ associé aux classifieurs G_j et à la base d'apprentissage $\mathcal{S}$,
2. calculer la liste (L) et déterminer la sous liste du (ou des) éléments de plus petite valeur dans (L),
3. **choisir** un élément dans cette sous liste,
4. déduire les signes des X_j associés à l'élément choisi,
5. **choisir** des X_j avec ces signes. Cette étape donne accès à l'infinité des solutions en $\boldsymbol{\beta}$.

6. calculer la valeur associée de $\boldsymbol{\beta}$, c'est-à-dire un classifieur, grâce — cf. (1) — aux relations :

$$\beta_1 = \frac{X_2 + X_3}{2}, \beta_2 = \frac{X_1 + X_3}{2}, \beta_3 = \frac{X_1 + X_2}{2}.$$

Quelques remarques à propos de cette procédure :

— Elle permet de déterminer une combinaison optimale des classifieurs sans avoir recours à quelque algorithme numérique que ce soit.

— Elle permet de minimiser la probabilité d'erreur sans faire appel à un quelconque substitut convexe.

— Des choix interviennent à deux étapes. Les classifieurs obtenus dépendent de ces choix, ils ne se valent pas pour ce qui est de l'erreur de généralisation.

— Lorsque l'on cherche à combiner un petit nombre de classifieurs (ici $p = 3$, on pourrait en prendre un peu plus) on réduit drastiquement la complexité algorithmique du problème puisque l'on se ramène à l'étude d'une fonction dans $\mathbb{R}^p$ (avec 2^p coefficients).

L'analyse du risque empirique pour déterminer le classifieur résultant pourrait s'arrêter là si il y a une seule valeur minimale sur les valeurs possibles, puisqu'on aurait dans ce cas déterminé le classifieur résultant.

Mais ce choix arbitraire des (β_j) associé aux signes des X_j et, plus encore, la possibilité d'avoir plusieurs régions donc plusieurs choix de classifieurs justifie l'utilisation d'une méthode de choix d'un $(\beta_1, \beta_2, \beta_3)$ unique si possible.

Robustesse de la prise de décision. Dans le cas où le minimum du risque empirique est atteint en plusieurs configurations (donc plusieurs classifieurs résultants), il suffit d'enlever ou d'ajouter un seul exemple bien choisi pour changer le comportement et fixer le classifieur logique à un des classifieurs résultants quelconques possibles. Ces situations pourraient être considérées comme instables ou pas du tout robustes.

Dans le cas où le minimum du risque empirique est unique, il suffit d'examiner la liste (L), de calculer le minimum $r > 0$ des différences entre ce minimum et les 13 autres valeurs et de remarquer qu'il suffit de supprimer $r + 1$ exemples bien choisis pour changer de minimum, donc de classifieur résultant. La valeur de r/n mesure la robustesse (fiabilité) de l'octuplet.

4 Convexification du risque

Minimiser ainsi le risque empirique est inhabituel du fait de sa non convexité qui empêche l'utilisation de méthodes d'optimisation numériques classiques. L'approche usuelle consiste à convexifier $\mathcal{R}_{1_{\bullet<0}}(\boldsymbol{\beta}, \mathcal{S})$ en remplaçant $1_{\bullet<0}$ par une fonction ϕ dite "calibrée pour la classification" [1] c'est-à-dire décroissante, strictement positive, strictement convexe avec $\phi(0) = 1$, et $\phi'(0) < 0$ et donc $\phi(x) \geq 1_{x<0}$. Le risque $\mathcal{R}_{1_{\bullet<0}}(\boldsymbol{\beta}, \mathcal{S})$ est alors remplacé par un risque plus grand

$$\mathcal{R}_\phi(\boldsymbol{\beta}, \mathcal{S}) = n^{-1} \sum_{i=1}^{n} \phi(y_i h(x_i)).$$

avec, ici, $h = \text{sign}(\boldsymbol{\beta}^T \mathbf{G})$. C'est une fonction sur $\mathbb{R}^3$ avec comme paramètres n_j/n, m_j/n, puisqu'elle s'écrit

$$\mathcal{R}_\phi(\boldsymbol{\beta}, \mathcal{S}) = \sum_{j=0}^{3} \frac{n_j}{n} \phi(-X_j) + \frac{m_j}{n} \phi(X_j).$$

Minimisation du risque convexifié versus ADABOOST. La fonction $x \to \phi(x) = e^{-x}$ est un choix classique utilisé dans les méthodes dites de boosting dont la plus connue s'appelle ADABOOST. Apparue en 1995 [3] d'après [5] qui a introduit l'idée de classifieur faible, ADABOOST a été largement utilisé. L'algorithme ADABOOST tient à jour un ensemble de poids et remplace, à chaque étape, le poids de chaque exemple par un poids plus grand pour les exemples mal classés et un poids plus petit pour les exemples bien classés par le classifieur faible utilisé. Cet algorithme est spécifique à la fonction exponentielle, Giraud [4] l'a identifié comme une méthode de minimisation. Cette méthode pourrait ressembler à une méthode classique d'optimisation : l'algoritme de relaxation (minimisation par coordonnées) sauf que toutes les coordonnées de $\boldsymbol{\beta}$ ne sont pas parcourues à chaque étape.

Même si ADABOOST est bien un algorithme de descente, il n'y a aucune information sur la convergence de la suite des $\boldsymbol{\beta}$.

Des expériences numériques explorant l'ensemble des octuplets de somme donnée sont présentées dans la section 5.

Un autre choix classique est $\phi(x) = \log_2(1 + e^{-x})$.

Nous appelerons Boost la solution obtenue par minimisation (en pratique par l'algorithme de relaxation) du risque convexifié pour $\phi(x) = \exp(-x)$, et Logit celle obtenue pour $\phi(x) = \log_2(1 + e^{-x})$.

Notons que l'algorithme de relaxation ne fait pas appel à un gradient, mais à la dérivée d'une fonction à valeurs réelles (et, dans les cas de Boost et de Logit, cela se résume à des égalités explicites résolubles par radicaux à chaque étape).

Vitesse de convergence de relaxation. Si $n_j m_j \neq 0$ pour au moins trois valeurs de j, le risque convexifié est une fonction infinie à l'infini et α-convexe sur tout compact de $\mathbb{R}^3$ donc **a un unique point de minimum** et l'algorithme de relaxation converge vers ce point. Ceci est valable pour toute fonction ϕ de classe C^2 calibrée pour la classification dont la partie paire est infinie à l'infini. La fonction Boost — $\exp(-x)$ — et la fonction logistique — $\log_2(1 + e^{-x})$ — entrent dans ce cadre. Sous ces hypothèses, l'algorithme de relaxation approche le minimum, à précision ϵ, en $O(\log(1/\epsilon))$ étapes.

Le point de minimum du risque convexifié est noté $\boldsymbol{\beta}_\phi$, et le classifieur qui en découle est $h_\phi^{\text{opt}} = \text{sign}(\boldsymbol{\beta}_\phi^T \mathbf{G})$.

Robustesse de la prise de décision. Les valeurs de $\boldsymbol{\beta}_\phi$ telles qu'au moins un des X_j est nul jouent un rôle très particulier, elles expriment quatre relations entre les n_j, m_j, elles ne dépendent que des valeurs $n_j/n, m_j/n$ et sont donc des fonctions C^1 sur $K \subset [0, 1]^8$, on les appelle les frontières associées à ϕ. Par exemple, si un point n_j, m_j est trop proche d'une de ces frontières il est possible qu'en changeant d'une unité une seule des valeurs de la liste (L), on change un des signes des X_j et donc le classifieur résultant. Cette répartition pourrait être

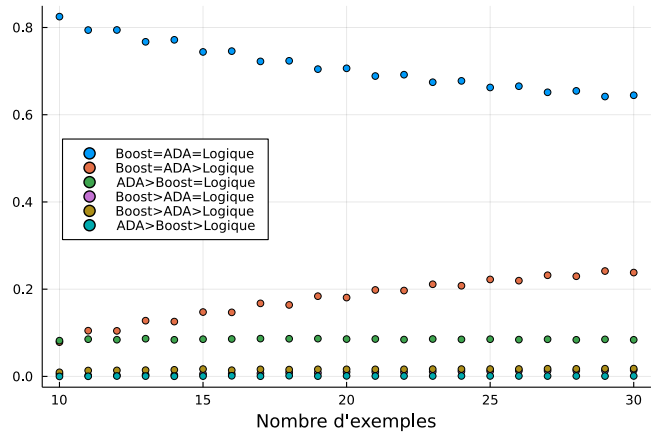

FIGURE 1 – Pourcentage des cas avec une hiérarchie donnée du nombre des erreurs entre ADABOOST (ADA), l'optimum pour le coût Boost (Boost) et le classifieur Logique (Logique), en fonction du nombre d'exemples.

considérée comme de mauvaise qualité. D'autre part, comme ϕ est supposée de classe C^2, un voisinage de 0 pour X_j, pour un j donné, correspond à un voisinage dans $[0,1]^8$ dont la taille est contrôlée par la taille du voisinage de 0 utlisé (dépendance C^1 du point de minimum par rapport aux paramètres $n_j/n, m_j/n$ par application du théorème des fonctions implicites).

Etant donné un point $(n_j/n, m_j/n)$, il existe un δ optimal ne dépendant que de ce point de $[0,1]^8$, de ϕ et de la norme choisie sur $\mathbb{R}^8$ tel que les signes des X_j ne changent pas dans $||(x_0 - \frac{n_0}{n}, \cdots, x_3 - \frac{n_3}{n}, y_0 - \frac{m_0}{n}, \cdots y_3 - \frac{m_3}{n})|| < \delta$.

De même δ mesure la fiabilité (ou robustesse) pour la norme choisie de l'octuplet $\phi-$convexifié.

5 Expériences numériques

Nombre d'erreurs des différentes solutions. En explorant tous les octuplets admissibles de somme donnée, le nombre d'erreurs des classifieurs calculés par ADABOOST, par la minimisation d'un coût convexe (boost ou logit) et le nombre d'erreurs du classifieur logique s'avèrent être ordonnées de multiples manières possibles mais pas dans les mêmes proportions.

Par exemple, pour $n = 30$ et l'octuplet $(4, 3, 1, 5, 3, 7, 2, 5)$, les nombres des erreurs obtenues sont les suivants : logique 10, Boost 13, Logit 10 et ADABOOST 13. Pour $(1, 5, 3, 2, 3, 2, 8, 6)$, logique 12, Boost 13, Logit 13 et ADABOOST 15.

La figure (1) trace l'évolution de ces proportions en fonction du nombre n d'exemples. Pour $n = 30$, dans 88 % des cas admissibles, le nombre d'erreurs est le même pour ADABOOST et Boost. Cependant, dans 8,4 % des cas ADABOOST est moins bon que Boost et dans 3,2 % des cas il est meilleur.

Comparaison des suites minimisantes. La limite de la suite $\boldsymbol{\beta}$ construite par ADABOOST ne converge pas toujours vers un point (on peut l'affirmer pour des suites extraites), et une suite extraite converge, dans beaucoup de cas, vers un point autre que le point de minimum.

Des expériences numériques pour $n \leq 31$ (taille raisonnable pour parcourir tous les octuplets admissibles, c'est-à-dire issus de classifieurs faibles) montrent que le classifieur résultant construit avec le point limite de l'algorithme ADABOOST conduit, dans un très grand nombre de cas, à un classifieur différent du classifieur obtenu avec le minimum (exact et unique) de la fonction $\mathcal{R}_\phi$. Les différents risques convexifiés conduisent à des valeurs de minima qui peuvent être très différentes. De plus, un peu moins de la moitié des points limites de l'algorithme ADABOOST se trouvent sur le bord d'équation $\beta_1\beta_2\beta_3 = 0$, alors que le point de minimum de $\mathcal{R}_\phi$ est presque toujours dans $\beta_1\beta_2\beta_3 \neq 0$. Il y a beaucoup de cas où l'algorithme ADABOOST ne combine qu'au plus deux classifieurs alors qu'extrêmement souvent le minimum du risque est atteint en un point qui combine les trois classifieurs.

6 Conclusion et perspectives

Tout en utilisant des outils et des méthodes classiques d'apprentissage supervisé, nous ajoutons une nouvelle structure basée sur une analyse logique de quelques classifieurs. Cela donne une analyse complète du classifieur résultant sans avoir à exécuter un algorithme. Cela permet de se poser la question de la robustesse d'un ensemble de trois classifieurs qui étend les conditions classiques sur chaque classifieur. Cette question de robustesse s'étend aux risques convexifiés (donc avec unique minimum), les conditions semblent différentes, ce qui peut éventuellement permettre d'améliorer les résultats en changeant la convexification ϕ. Cette méthode permet aussi de réduire drastiquement la complexité des méthodes numériques. L'extension à p classifieurs est immédiate. Pour p de l'ordre de $\log_2 n$, on a autant de classes que d'exemples donc beaucoup de risques qu'un certain nombre de classes soient vides, ce qui rend la description peu intéressante. Comme pour $p = 3$, pour p petit (moins de 10), les algorithmes (logique et relaxation) fonctionnent très bien théoriquement comme numériquement.

Références

[1] Peter L Bartlett, Michael I Jordan, and Jon D McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473) :138–156, 2006.

[2] Jean-Marc Brossier and Olivier Lafitte. Combining weak classifiers : a logical analysis. In *23rd International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, pages 178–181, 2021.

[3] Yoav Freund. Boosting a weak learning algorithm by majority. *Information and computation*, 121(2) :256–285, 1995.

[4] Christophe Giraud. Introduction to high-dimensional statistics. *Monographs on Statistics and Applied Probability*, 139 :139, 2021.

[5] Robert E. Schapire. The strength of weak learnability. *Machine learning*, 5(2) :197–227, 1990.