

# Kernel machines : une nouvelle méthode pour l'optimisation de l'alignement des noyaux et l'amélioration des performances

Jean-Baptiste POTHIN, Cédric RICHARD

Institut des Sciences et Technologies de l'Information de Troyes (STIT, FRE CNRS 2732)

12 rue Marie Curie, BP 2060

10010 Troyes cedex, France

jean\_baptiste.pothin@utt.fr, cedric.richard@utt.fr

**Résumé** – On propose ici une méthode visant à optimiser, selon le critère d'alignement maximum, une combinaison linéaire de noyaux donnés. L'algorithme proposé s'inspire de la méthode SMO dédié à l'optimisation des SVMs : il segmente le problème d'optimisation initial en sous-problèmes pour lesquels on dispose d'une solution analytique, évitant ainsi par exemple le problème de Hessien mal conditionné. Son implémentation ne nécessite pas l'utilisation d'une bibliothèque d'optimisation particulière. Enfin, la solution obtenue s'avère en pratique plus parcimonieuse que celle trouvée par l'approche de type programmation quadratique, tout en offrant des performances similaires.

**Abstract** – In this paper, we propose a method for optimizing the alignment of a linear combination of kernels. The proposed algorithm is based on the well-known SMO idea : it solves a quadratic optimization problem by analytically solving subproblems. Ill-conditioned Hessian can thus be avoided. Its implementation does not require any optimization toolbox. Finally, we show that solutions obtained with our strategy are sparser than those obtained by quadratic programming, for equivalent performance.

## 1 Introduction

Dans le foisonnant domaine des méthodes dites à *noyau*, dont les SVMs sont le fer de lance, on constate que les efforts de recherche ont été essentiellement portés sur les applications, délaissant quelque peu l'étude des noyaux eux-mêmes. Cet état de fait est d'autant plus surprenant que ces derniers conditionnent largement l'efficacité des traitements, par leur aptitude à révéler les similarités existant au sein des données traitées. L'adaptation d'un noyau à l'intelligence des données relève de trois problématiques. La première concerne la définition de critères de choix d'un noyau, par exemple une mesure d'efficacité statistique maximale. La seconde vise à circonscrire une famille de noyaux apte à répondre au problème posé, et la dernière est consacrée à la détermination du meilleur noyau de la famille exhibée en procédant par exemple à l'optimisation de ses paramètres caractéristiques. On propose ici une méthode originale visant à répondre à ces deux dernières questions. L'article est organisé comme suit. Nous commençons par préciser le contexte de l'étude avant de passer à la présentation de l'algorithme proposé. Puis des résultats expérimentaux viennent illustrer les performances de notre approche, suivis d'une conclusion et de quelques perspectives.

### 1.1 Contexte

Soit  $\mathcal{X}$  un espace métrique compact. On appelle *noyau de Mercer* une fonction  $\kappa$  symétrique de  $\mathcal{X} \times \mathcal{X}$  dans  $\mathbb{R}$  vérifiant

$$\sum_{i=1}^n \sum_{j=1}^n a_i a_j \kappa(\mathbf{x}_i, \mathbf{x}_j) \geq 0 \quad (1)$$

pour tout  $n \in \mathbb{N}$ ,  $\mathbf{x}_1, \dots, \mathbf{x}_n \in \mathcal{X}$  et  $a_1, \dots, a_n \in \mathbb{R}$ . En d'autres termes, la matrice carrée  $\mathbf{K}$  de terme général  $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j)$ ,

appelée *matrice de Gram*, est symétrique, semi-définie positive pour tout  $\{\mathbf{x}_i\}_{i=1}^n \subseteq \mathcal{X}$ . Dans ces conditions, on montre qu'il existe une fonction  $\phi : \mathcal{X} \rightarrow \mathcal{F}$  telle que :

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_{\mathcal{F}}, \quad (2)$$

où  $\langle \cdot, \cdot \rangle_{\mathcal{F}}$  exprime le produit scalaire dans l'espace caractéristique  $\mathcal{F}$ , de dimension éventuellement infinie [1]. Un noyau fournit ainsi un moyen simple d'évaluer la similitude entre les images  $\phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \in \mathcal{F}$  pour toute paire  $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}$  sans avoir à expliciter  $\phi$  ou  $\mathcal{F}$ .

Supposons maintenant que l'on dispose d'un ensemble de  $n$  observations  $\mathbf{x}_i \in \mathcal{X}$  divisées en 2 classes, la variable  $y_i = \pm 1$ , stockée dans le vecteur colonne  $\mathbf{y}$ , désignant la classe d'appartenance de  $\mathbf{x}_i$ . Selon l'approche SVM standard [2], l'hyperplan séparateur optimal est à marge maximale. Il est trouvé dans l'espace caractéristique  $\mathcal{F}$  engendré par un noyau donné, par exemple le noyau gaussien, polynomial ou encore tangente hyperbolique :

$$\begin{aligned} \kappa_1(\mathbf{x}_i, \mathbf{x}_j) &= \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right) \\ \kappa_2(\mathbf{x}_i, \mathbf{x}_j) &= (\gamma \mathbf{x}_i^t \mathbf{x}_j + b)^d, \\ \kappa_3(\mathbf{x}_i, \mathbf{x}_j) &= \tanh(a \mathbf{x}_i^t \mathbf{x}_j + b). \end{aligned} \quad (3)$$

Ici,  $\Theta_1 = \sigma$ ,  $\Theta_2 = (\gamma, b, d)$  et  $\Theta_3 = (a, b)$  sont appelés *hyperparamètres*. Ils sont habituellement fixés *a priori*, c'est-à-dire avant apprentissage.

### 1.2 Noyau adapté par critère d'alignement

L'objectif de l'article est de trouver un noyau adapté aux données dans le cadre d'une approche de type *boosting*. On rappelle à ce propos que l'idée de base du *boosting* est de combiner des solutions de moindre qualité afin d'aboutir à une solution

plus performante. C'est notamment ainsi que l'on peut obtenir un classifieur compétitif étant donné des classifieurs aux performances modestes [3]. Ce principe, reformulé dans le contexte de l'ingénierie des noyaux, vise à combiner des noyaux génériques tels que ceux cités précédemment afin d'améliorer les performances [4]. Grâce aux propriétés algébriques des fonctions définies positives, il est facile de construire de nouveaux noyaux de Mercer à partir de noyaux donnés. Par exemple :

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \alpha_1 \kappa_1(\mathbf{x}_i, \mathbf{x}_j) + \alpha_2 \kappa_2(\mathbf{x}_i, \mathbf{x}_j), \quad (4)$$

est un noyau valide pour  $\alpha_1, \alpha_2$  positifs. De même, le produit de deux noyaux donne un noyau valide [5] :

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \kappa_1(\mathbf{x}_i, \mathbf{x}_j) \kappa_2(\mathbf{x}_i, \mathbf{x}_j). \quad (5)$$

Le modèle adopté pour la combinaison de noyaux est communément linéaire pour lequel il s'agit d'identifier les paramètres  $\alpha_i$  :

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \sum_{i=1}^M \alpha_i \kappa_i(\mathbf{x}_i, \mathbf{x}_j). \quad (6)$$

En remarquant que la matrice de Gram cible est définie par  $\mathbf{K}^* = \mathbf{y}\mathbf{y}^t$ , Cristianini *et al.* proposent de quantifier la qualité *a priori* de l'espace de représentation  $\mathcal{F}$  à l'aide de la mesure d'*alignement* suivante [6] :

$$0 \leq \mathcal{A}(\mathbf{K}, \mathbf{K}^*) = \frac{\langle \mathbf{K}, \mathbf{K}^* \rangle_F}{\|\mathbf{K}\|_F \|\mathbf{K}^*\|_F} \leq 1, \quad (7)$$

où  $\langle \cdot, \cdot \rangle_F$  désigne le produit scalaire de Frobenius. Si d'autres critères existent, on peut citer par exemple le *class separability criterion* [7] et le *data centering criterion* [8], l'avantage de l'alignement est que son optimisation relativement au modèle linéaire (6) se ramène à un problème d'optimisation convexe [9] :

$$\max_{\alpha} - \sum_{i,j} \alpha_i \alpha_j (\langle \mathbf{K}_i, \mathbf{K}_j \rangle_F + \lambda \delta_{ij}) + \sum_i \alpha_i \mathbf{y}^t \mathbf{K}_i \mathbf{y} \quad (8)$$

sous les  $M$  contraintes  $\alpha_i \geq 0$  afin de garantir la semi-définie positivité de la matrice de Gram  $\mathbf{K}$  associée au noyau (6). Notons que dans cette expression,  $\delta_{ij}$  désigne le symbole de Kronecker et  $\lambda > 0$  un paramètre de régularisation. En outre, on montre que l'alignement possède un biais nul et une variance faible pouvant être bornée [6]. Son lien avec l'erreur de généralisation a quant à lui été établi théoriquement et validé expérimentalement [9, 10, 11].

## 2 Méthode proposée

Dans le même esprit que la méthode SMO [12] dédiée à l'élaboration des SVMs, nous suggérons de segmenter le problème d'optimisation (8) en sous-problèmes pour lesquels on est en mesure d'exhiber une solution analytique.

### 2.1 Solution analytique dans le cas $M = 2$

D'après (8), la maximisation de l'alignement d'une combinaison linéaire de deux noyaux  $\kappa_1$  et  $\kappa_2$  revient à maximiser la fonction

$$W(\alpha_1, \alpha_2) = -\alpha_1^2 (\|\mathbf{K}_1\|_F^2 + \lambda) - \alpha_2^2 (\|\mathbf{K}_2\|_F^2 + \lambda) - 2\alpha_1 \alpha_2 \langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F + \alpha_1 \mathbf{y}^t \mathbf{K}_1 \mathbf{y} + \alpha_2 \mathbf{y}^t \mathbf{K}_2 \mathbf{y} \quad (9)$$

sous les contraintes  $\alpha_1, \alpha_2 \geq 0$ . Intéressons-nous dans un premier temps au problème sans contraintes. Le point stationnaire de (9) étant obtenu en annulant ses dérivées premières, on obtient pour conditions d'optimalité :

$$-2(\|\mathbf{K}_1\|_F^2 + \lambda)\alpha_1 - 2\alpha_2 \langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F + \mathbf{y}^t \mathbf{K}_1 \mathbf{y} = 0, \quad (10)$$

$$-2(\|\mathbf{K}_2\|_F^2 + \lambda)\alpha_2 - 2\alpha_1 \langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F + \mathbf{y}^t \mathbf{K}_2 \mathbf{y} = 0. \quad (11)$$

On aboutit ainsi au système suivant :

$$\begin{aligned} (\|\mathbf{K}_1\|_F^2 + \lambda)\alpha_1 + \langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F \alpha_2 &= \frac{\mathbf{y}^t \mathbf{K}_1 \mathbf{y}}{2} \\ \langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F \alpha_1 + (\|\mathbf{K}_2\|_F^2 + \lambda)\alpha_2 &= \frac{\mathbf{y}^t \mathbf{K}_2 \mathbf{y}}{2} \end{aligned} \quad (12)$$

qui admet pour solution :

$$\alpha_1^* = \frac{1}{2} \frac{\mathbf{y}^t \mathbf{K}_1 \mathbf{y} - 2\langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F \alpha_2^*}{\|\mathbf{K}_1\|_F^2 + \lambda} \quad (13)$$

$$\alpha_2^* = \frac{1}{2} \frac{(\|\mathbf{K}_1\|_F^2 + \lambda)\mathbf{y}^t \mathbf{K}_2 \mathbf{y} - \langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F \mathbf{y}^t \mathbf{K}_1 \mathbf{y}}{(\|\mathbf{K}_1\|_F^2 + \lambda)(\|\mathbf{K}_2\|_F^2 + \lambda) - \langle \mathbf{K}_1, \mathbf{K}_2 \rangle_F^2}. \quad (14)$$

D'après (13),  $\alpha_1^*$  et  $\alpha_2^*$  ne peuvent être nuls simultanément. De même, un raisonnement par l'absurde montre que  $\alpha_1^*$  et  $\alpha_2^*$  ne peuvent être strictement négatifs simultanément. Supposons pour cela  $\alpha_1^*, \alpha_2^* < 0$ . La combinaison linéaire des matrices  $\mathbf{K}_1$  et  $\mathbf{K}_2$  peut alors s'écrire  $-(|\alpha_1^*| \mathbf{K}_1 + |\alpha_2^*| \mathbf{K}_2) = -\mathbf{K}$ . En utilisant les propriétés  $\mathcal{A}(-\mathbf{K}, \mathbf{K}^*) = -\mathcal{A}(\mathbf{K}, \mathbf{K}^*)$  et  $0 \leq \mathcal{A}(\mathbf{K}, \mathbf{K}^*) \leq 1$ , on trouve qu'une solution de la forme  $\alpha_1^*, \alpha_2^* < 0$  est toujours associée à un alignement négatif. Or,  $(\alpha_1 = 1, \alpha_2 = 0)$  amène à un alignement positif. On en conclut que l'hypothèse de départ  $\alpha_1^*, \alpha_2^* < 0$  ne peut être vraie.

En dehors de cela, deux scénarios peuvent être observés :

- **Cas 1** :  $\alpha_1^*, \alpha_2^* > 0$

La solution (13)-(14) est admissible puisqu'elle vérifie les contraintes de positivité. En outre, cette solution est toujours associée à un alignement maximum, le Hessien de (9) étant défini négatif.

- **Cas 2** :  $\alpha_2^* < 0$ , à un échange près de  $\alpha_1^*$  et  $\alpha_2^*$ .

Le couple  $(\alpha_1^*, \alpha_2^*)$  n'est pas admissible. Partant d'une solution réalisable du plan  $(\alpha_1, \alpha_2)$ , on montre, par la méthode des directions réalisables [13], que l'optimum contraint est de la forme  $(\alpha_1^+ > 0, \alpha_2^+ = 0)$ . En d'autres termes, le noyau  $\kappa_2$  est écarté. Or, l'alignement est invariant par changement d'échelle, signifiant que l'on a  $\mathcal{A}(\alpha_1^+ \mathbf{K}_1, \cdot) = \mathcal{A}(\mathbf{K}_1, \cdot), \forall \alpha_1^+ > 0$ . On peut donc attribuer arbitrairement la valeur 1 à  $\alpha_1^+$ , indépendamment de la constante de régularisation  $\lambda$ .

En résumé, la combinaison linéaire de deux noyaux est d'alignement maximum pour :

$$(\alpha_1^+, \alpha_2^+) = \begin{cases} (\alpha_1^*, \alpha_2^*) & \text{si } \alpha_1^*, \alpha_2^* > 0 \\ (1, 0) & \text{si } \alpha_2^* \leq 0 \\ (0, 1) & \text{si } \alpha_1^* \leq 0. \end{cases} \quad (15)$$

Dans la section suivante, on exploite ce résultat pour traiter le problème (8) lorsque  $M$  est supérieur à 2.

Table 1: Optimisation de l’alignement par algorithme BB

Initialisations :

$$k = 0, I = \emptyset, \hat{\mathbf{K}} = \mathbf{0}, \mathcal{A}_{BB} = 0, \boldsymbol{\alpha} = \mathbf{0}$$

$$j^* = \arg \max_j \mathcal{A}(j), \alpha_{1,j^*}^+ = 0, \alpha_{2,j^*}^+ = 1.$$

**Tant que**  $\mathcal{A}_{BB} < \mathcal{A}(j^*)$ 

$$k = k + 1, I = I \cup \{j^*\}, \mathcal{A}_{BB} = \mathcal{A}(j^*)$$

$$\hat{\mathbf{K}}^{(k)} = \alpha_{1,j^*}^+ \hat{\mathbf{K}}^{(k-1)} + \alpha_{2,j^*}^+ \mathbf{K}_{j^*}$$

$$\boldsymbol{\alpha}^{(k)} = \alpha_{1,j^*}^+ \boldsymbol{\alpha}^{(k-1)}$$

$$\alpha_{j^*}^{(k)} = \alpha_{2,j^*}^+$$

**Pour tout**  $j \notin I$ 

$$\text{Calculer } \alpha_{1,j}^+, \alpha_{2,j}^+ \text{ et } \hat{\mathbf{K}}^{(k+1)}(j)$$

$$\text{Calculer } \mathcal{A}_{BB}(j) = \mathcal{A}(\hat{\mathbf{K}}^{(k+1)}(j), \mathbf{K}^*)$$

Choisir  $j^*$  tel que  $j^* = \arg \max_{j \notin I} \mathcal{A}(j)$ Retourner  $\boldsymbol{\alpha}$ 

## 2.2 Traitement du cas $M > 2$

Pour traiter le cas plus fréquent où  $M > 2$ , nous adoptons une stratégie de type *Branch and Bound* (BB) [14]. L’idée consiste à partir du meilleur noyau à disposition puis de sélectionner, parmi tous les noyaux restants, celui qui maximise l’alignement après combinaison. Ce choix est fait itérativement sur un horizon égal à 1 tant que l’alignement peut-être amélioré.

Supposons qu’à l’itération  $k$  de l’algorithme, on dispose de :

$$\hat{\mathbf{K}}^{(k)} = \sum_{i \in I} \alpha_i^{(k)} \mathbf{K}_i, \quad (16)$$

où  $I$  est un sous-ensemble de  $\{1, \dots, M\}$  et  $\mathbf{K}_i$  la matrice de Gram associée au noyau  $\kappa_i$ . L’objectif est d’atteindre, à l’itération  $k + 1$ , le meilleur alignement possible. D’après le paragraphe précédent, on est en mesure de déterminer la combinaison optimale

$$\hat{\mathbf{K}}^{(k+1)}(j) = \alpha_{1,j}^+ \left( \sum_{i \in I} \alpha_i^{(k)} \mathbf{K}_i \right) + \alpha_{2,j}^+ \mathbf{K}_j \quad (17)$$

au sens du critère d’alignement pour tout  $j$ . On détermine ainsi l’optimum local atteint à l’itération  $k + 1$  par :

$$\hat{\mathbf{K}}^{(k+1)} = \hat{\mathbf{K}}^{(k+1)}(j^*), \quad (18)$$

où

$$j^* = \arg \max_{j \notin I} \mathcal{A}(\hat{\mathbf{K}}^{(k+1)}(j), \mathbf{K}^*). \quad (19)$$

L’algorithme proposé Table 1 se résume alors ainsi : le noyau courant  $\hat{\kappa}^{(k)}$ , composé d’une combinaison linéaire des  $\kappa_{i, i \in I}$ , est amélioré en le combinant avec le noyau  $\kappa_{j^*}$  qui maximise l’alignement parmi tous les noyaux restants  $\kappa_{j, j \notin I}$ . En répétant cette stratégie jusqu’à ce que l’alignement ne progresse plus, on obtient, à défaut de l’optimum global, une combinaison (6) mieux adaptée aux données de l’ensemble d’apprentissage que chacun des noyaux  $\kappa_i$  pris individuellement.

Remarquons qu’un moyen simple d’accélérer l’algorithme et de précalculer les réels  $\langle \mathbf{K}_i, \mathbf{K}_j \rangle_F$ ,  $\|\mathbf{K}_i\|_F^2 + \lambda$  et  $\mathbf{y}^t \mathbf{K}_i \mathbf{y}$

Table 2: Alignements individuels

	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
I	.201	.208	.188	.151	.056	.136	.094	.080
B	<b>.079</b>	.207	.218	.189	.090	<b>.604</b>	.446	.095

I = ION, B = BCW

intervenant lors du calcul de (13) et (14). De même, notons que le critère d’arrêt peut facilement être modifié de manière à intégrer une contrainte sur le nombre  $M_{max} \geq 1$  de  $\alpha_i$  non nul souhaité. Enfin, on peut également envisager, grâce au paramètre  $0 \leq \eta \leq 1$ , de mettre fin prématurément à l’algorithme si l’amélioration n’est pas significative :

**Tant que**  $(k < M_{max})$  et  $(\frac{\mathcal{A}_{BB}}{\mathcal{A}(j^*)} < \eta)$ .

## 3 Résultats expérimentaux

Afin d’illustrer l’algorithme précédent, nous l’avons appliqué aux données<sup>1</sup> Ionosphere (ION) et Breast Cancer Wisconsin (BCW). Les noyaux considérés ont été, dans un premier temps, les noyaux polynômiaux de degré 1 (a), 2 (b), 3 (c) et 4 (d), et les noyaux gaussiens de paramètre  $\sigma$  égal à 0.1 (e), 5 (f), 10 (g) et 100 (h). Le paramètre de régularisation  $\lambda$  a quant à lui été fixé à 1 et l’on a considéré trois seuils  $\eta$  en deçà desquels l’amélioration de l’alignement n’a plus été jugée suffisante, provoquant l’arrêt prématuré de l’algorithme. Le tableau 2 fournit la valeur de l’alignement pour chacun des huit noyaux considérés individuellement.

L’alignement maximum pour les données ION, trouvé en résolvant (8) selon l’approche classique (QP : Quadratic Programming),<sup>2</sup> est  $\mathcal{A}_{QP} = .2105$ , une combinaison linéaire optimale associée étant  $.0041(a) + .0008(b) + .2443(e)$ . Pour BCW, l’optimum est  $\mathcal{A}_{QP} = .6277$ , obtenu pour  $3.4 * 10^{-09}(a) + .5675(f)$ . Le tableau 3 récapitule les résultats obtenus selon la stratégie proposée. Dans le premier cas (I), on remarque que l’alignement maximum  $\mathcal{A}_{QP} = .2105$  est sensiblement identiques à l’alignement du noyau (b) pris individuellement. On peut donc écarter les noyaux (a) et (e) sans dégradation notable (voir résultats avec  $\eta = .990$ ). Dans le second cas (B), la solution  $\mathcal{A}_{BB} = .6277$  que nous obtenons coïncide avec  $\mathcal{A}_{QP}$ . On notera également la faible contribution de (a) comparativement à (f) et leur combinaison qui semble indiquer que l’amélioration de l’alignement s’effectue en combinant des noyaux qui ne sont pas alignés entre eux.

Nous avons ensuite procédé à un certain nombre de simulations sur le jeu de donnée BCW en particulier (voir Table 4). Pour chaque simulation, nous avons sélectionné aléatoirement 30 vecteurs paramétrant les noyaux générique suivant : le noyau polynomial, gaussien et exponentiel. L’optimisation par combinaison linéaire des 30 noyaux générés s’est ensuite déroulée comme précédemment. La solution trouvée par l’algorithme proposé apparaît plus parcimonieuse que celle obtenue selon l’approche QP. On constate également que notre programme est plus stable dans le sens où l’approche QP est sensible au

<sup>1</sup>ftp://ftp.ics.uci.edu/pub/machine-learning-databases/

<sup>2</sup>implémentation réalisée sous MATLAB avec *quadprog*

Table 3: Solutions trouvées par l'algorithme BB

	$\eta$	$\mathcal{A}_{BB}$	noyaux retenus
I	.990	.2081	(b)
	.995	.2095	.0011(b) + .2266(e)
	1	.2103	.0036(a) + .0009(b) + .175(e)
B	.990	.6277	$3.4 * 10^{-09}(a) + .5675(f)$
	.995	.6277	$3.4 * 10^{-09}(a) + .5675(f)$
	1	.6277	$3.4 * 10^{-09}(a) + .5675(f)$

I = ION, B = BCW

Table 4: Résultats pour 10 simulations comprenant chacune 30 noyaux paramétrés aléatoirement - données BCW

$\mathcal{A}_{min}$	$\mathcal{A}_{max}$	$\mathcal{A}_{BB}$	$\mathcal{A}_{QP}$	$n_{BB}$	$n_{QP}$
0.0785	0.5953	0.6094	0.2081	4	30
0.0785	0.4952	0.4963	0.4963	2	6
0.0785	0.1433	0.1433	0.1433	1	1
0.0785	0.1872	0.1872	0.1872	1	1
0.0785	0.1975	0.1975	0.1975	1	1
0.0785	0.3225	0.3630	0.2081	3	30
0.0785	0.6040	0.6040	0.6040	1	1
0.0785	0.3225	0.3450	0.3450	2	2
0.0785	0.2938	0.3002	0.3002	2	6
0.0785	0.2290	0.2290	0.2290	1	1
0.0785	0.3390	0.3475	0.2919	1.8	7.9

$\mathcal{A}_{min(max)}$  est l'alignement individuel le plus petit (grand) parmi les 30 noyaux de la collection considérée,  $n_{BB}$  est le nombre de  $\alpha_i \neq 0$  obtenu par l'approche proposée et  $n_{QP}$  le nombre de  $\alpha_i \neq 0$  obtenu par l'approche QP.

conditionnement du Hessien régissant le problème, signifiant que le choix du paramètre  $\lambda$  peut être critique. On remarque d'ailleurs que  $\mathcal{A}_{QP}$  est quelques fois inférieur à  $\mathcal{A}_{max}$  (voir résultats pour lesquels  $n_{QP} = 30$ ). Cela correspond à des situations où le programme *quadprog* échoue numériquement.

## 4 Conclusion et perspectives

Nous avons mis en œuvre une stratégie simple pour l'optimisation, par combinaison linéaire, d'une collection de noyaux donnés. L'algorithme proposé permet de contrôler facilement la parcimonie de la solution, tout en restant stable numériquement. Les exemples choisis dans cet article sont restreints au domaine d'application qu'est la classification, même si une extension à la régression peut doré et déjà être envisagée [10]. Nous planifions également de comparer la méthode proposée ici avec la technique d'apprentissage utilisant le principe d'optimisation SDP (Semi-Definite Programming) [11]. Un autre axe de recherche est d'adopter un modèle additif/multiplicatif.

## References

- [1] N. ARONSZAJN. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, vol. 68, pages 337-404, 1950.
- [2] C. CORTES ET V. VAPNIK. Support Vector Networks. *Machine Learning*, 20 : 1-25, 1995.
- [3] M. COLLINS, R. E. SCHAPIRE ET Y. SINGER. Logistic regression, adaboost and bregman distances. *Machine Learning*, vol. 47, no 2-3, pages 253-285, 2002.
- [4] K. CRAMMER, J. KESHET ET Y. SINGER. Kernel design using boosting. *Advances in Neural Information Processing Systems 15*. Cambridge, MA : MIT Press, 2003.
- [5] B. SCHÖLKOPF ET A. SMOLA. Learning with kernels. MIT Press, Cambridge, MA, 2002.
- [6] N. CRISTIANINI, J. SHAWE-TAYLOR, A. ELISSEEFF ET J. KANDOLA. On kernel-target alignment. *Advances in Neural Information Processing Systems 14*, Cambridge, MA : MIT Press, pages 367-373, 2002.
- [7] L. WANG ET K. LUK CHAN. Learning kernel parameters by using class separability measure. *Neural Information Processing Systems (NIPS)*, Canada, 2002.
- [8] M. MEILA. Data centering in feature space. *Proceedings of Ninth International Workshop on Artificial Intelligence and Statistics*, 2003.
- [9] J. KANDOLA, J. SHAWE-TAYLOR ET N. CRISTIANINI. Optimizing kernel alignment over combinations of kernels. Technical Report 121, Department of Computer Science, University of London, <http://www.neurocolt.org>, 2002.
- [10] J. KANDOLA, J. SHAWE-TAYLOR ET N. CRISTIANINI. On the extensions of kernel alignment. Technical Report 120, Department of Computer Science, University of London, <http://www.neurocolt.org>, 2002.
- [11] G. LANCKRIET, N. CRISTIANINI, P. BARTLETT, L. EL GHAOUI ET M. I. JORDAN. Learning the kernel matrix with semi-definite programming. *Proceedings of the 19th International Conference on Machine Learning*, pages 323-330, 2002.
- [12] J. PLATT. Fast training of Support Vector Machines using Sequential Minimal Optimization. *Advances In Kernel Methods - Support Vector Learning*, chapitre 12, pages 185-208, MIT Press, 1999.
- [13] G. ZOUTENDIJK. Methods of feasible directions: a study in linear and non-linear programming. Elsevier, 1970.
- [14] E.L LAWLER ET D.E WOOD. Branch and Bound methods: a survey. *Operations Research*, vol. 149, pages 699-719, 1966.