Sélection de modèles par des méthodes à noyaux pour la classification de données séquentielles

Trinh Minh Tri Do, Thierry Artières, Patrick Gallinari
LIP6, Université Pierre et Marie Curie
{Prénom.Nom}@lip6.fr

Ce travail concerne le développement de méthodes de classification discriminantes pour des données séquentielles. Quelques techniques ont été proposées pour étendre aux séquences les méthodes discriminantes, comme les machines à vecteurs supports, par nature plus adaptées aux données en dimension fixe. Elles permettent de classifier des séquences complètes mais pas de réaliser la segmentation, qui consiste à reconnaître la séquence d'unités, phonèmes ou lettres par exemple, correspondant à un signal. En utilisant une correspondance donnée / modèle nous transformons le problème de l'apprentissage des modèles à partir de données par un problème de sélection de modèles, qui peut être attaqué via des méthodes du type machines à vecteurs supports. Nous proposons et évaluons divers noyaux pour cela et fournissons des résultats expérimentaux pour deux problèmes de classification.

1 Introduction

Cette étude concerne l'intégration d'une information discriminante dans des systèmes de classification de données reposant sur des modèles génératifs et plus spécifiquement sur des mélanges de modèles génératifs. Dans la majorité des tâches de classification, on dispose de deux possibilités principales sur la nature de l'approche à employer, l'approche discriminante et l'approche générative. On peut utiliser un modèle discriminant -- réseau de neurones, classifieur linéaire, machine à vecteurs supports (MVS) -- dont l'apprentissage est focalisé sur ce qui différencie les différentes classes. D'un point de vue probabiliste, cela correspond à apprendre les lois de probabilités a posteriori des classes. La plupart de ces techniques discriminantes sont adaptées à des données en dimension fixe et sont plus délicates à utiliser avec des données séquentielles, de taille variable, comme la parole, l'écriture, etc. Une autre approche consiste à modéliser les classes indépendamment les unes des autres, et à apprendre pour chacune un modèle correspondant à sa densité de probabilité (e.g. modèle gaussien, modèle de Markov) avec un critère du type Maximum de Vraisemblance. On utilise un modèle génératif par classe, où chaque modèle est appris indépendamment des autres avec les données de sa classe. Ensuite, via le théorème de Bayes, on peut se ramener aux probabilités a posteriori et donc construire un système de classification optimal.

En règle générale, l'approche discriminante est plus performante. Cependant, on peut avoir intérêt à employer des mélanges de modèles génératifs dans certaines conditions. Les mélanges de modèles sont particulièrement adaptés lorsque les classes sont fortement multimodales (par exemple en écriture manuscrite, un « b » peut être écrit de différentes façons, on parle d'allographes). Les modèles génératifs sont eux particulièrement intéressants lorsque les données sont de dimension variable. Ce dernier cas correspond à

toutes les données disponibles sous forme de signaux ou séquences (parole, écriture, …). L'intérêt des modèles génératifs pour les données séquentielles est un intérêt « faute de mieux » et réside essentiellement dans l'inadéquation des techniques discriminantes pour ce type de données. Certaines techniques ont été proposées pour apprendre avec un critère discriminant des modèles traitant des données séquentielles. Ainsi, le noyau de Fisher (Jaakola et al. 1998) permet d'utiliser des machines à vecteur support pour des données séquentielles. Cependant, ce type de techniques permet d'étendre les méthodes discriminantes à la classification de séquences complètes mais pas à la segmentation. Or, la segmentation, qui consiste à reconnaître des séquences d'unités, phonèmes ou lettres en reconnaissance de la parole ou de l'écriture, est souvent la tâche la plus intéressante pour des données séquentielles.

Notre but est d'explorer diverses techniques permettant d'allier l'efficacité des méthodes discriminantes avec la souplesse de mélanges de modèles génératifs, capables de réaliser la segmentation pour des données très variables. Nous cherchons ici à utiliser des méthodes discriminantes pour construire des modèles génératifs performants pour la segmentation de données séquentielles. Dans toute la suite, nous considérons que nous souhaitons mettre au point un modèle génératif par classe de la forme suivante :

$$P(x/C) = \sum_{i}^{N_c} w_i^c P(x/\lambda_i^c) \qquad (1)$$

Où C est une classe, $P(x/C)$ est la vraisemblance d'une donnée x par le modèle de la classe C, les λ_i^c sont les modèles de la classe C, et les w_i^c sont les probabilités a priori, ou poids, des composantes du mélange et vérifient $\sum_{i=1}^{N_c} w_i^c = 1$ pour tout c. N^c est appelé dans la suite la taille du modèle de la classe C.

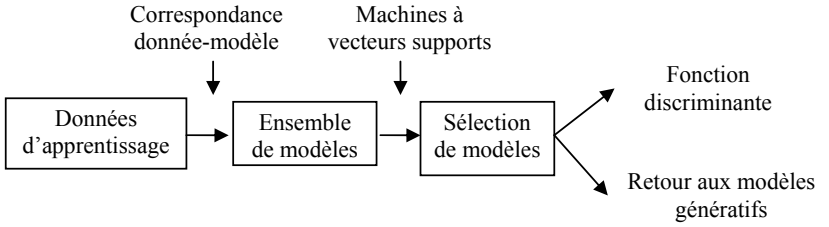

Fig. 1 - *Utilisation de machines à vecteurs supports pour l'apprentissage modèles génératifs.*

La figure 1 présente schématiquement notre approche. A partir d'un ensemble de données d'apprentissage, on change d'espace de représentation en utilisant une correspondance donnée-modèle, que nous expliciterons au §4, chaque donnée est ainsi transformée en un modèle. On peut alors utiliser des machines à vecteurs supports sur ces nouvelles « données » (i.e. des modèles) en définissant des noyaux sur les modèles, le but étant de définir des noyaux tels que les vecteurs supports correspondent à des modèles qui soient de bons candidats pour les λ_i^c . Le but de cette procédure est d'apprendre des modèles génératifs

obtenant une performance en classification la meilleure possible, notamment meilleure qu'un apprentissage classique avec un critère comme le Maximum de Vraisemblance.

Nous passons tout d'abord en revue les techniques utilisées pour exploiter des méthodes à noyaux dans la reconnaissance de données séquentielles. Puis nous définissons quelques noyaux sur les modèles que nous avons utilisés dans nos expériences. Enfin, nous décrivons des résultats expérimentaux sur des données en dimension fixe ainsi que sur des données manuscrites en ligne, permettant d'évaluer les qualités et pertinences des différentes méthodes envisagées.

2 Utilisation de noyaux pour des données séquentielles

Depuis quelques années, les Machines à Vecteurs de Support (MVS) (Vapnik 1995, Vapnik et al. 2001) sont devenues une approche classique et performante pour les problèmes de classification et de régression. Un classificateur MVS a la forme générale suivante pour un problème de classification à deux classes :

$$f(x) = \sum_{i=1}^{l} y_i \alpha_i \langle \phi(x_i).\phi(x) \rangle + b$$

Où x est une forme à classifier, les x_i sont les exemples d'apprentissage, dont la classe est identifiée par un label $y_i \in \{-1,+1\}$. Les coefficients α_i et b sont les paramètres de la fonction f, $\phi(x)$ représente la projection de x dans un espace de dimension élevé et $\langle \phi(x_i).\phi(x) \rangle$ représente le produit scalaire de deux vecteurs projetés. La classification d'un signal de test x est réalisée en déterminant le signe de $f(x)$. Sous certaines conditions, il n'est pas nécessaire d'expliciter la fonction de projection Φ car il existe une fonction, que l'on nomme fonction noyau telle que $K(x_i, x) = \langle \phi(x_i).\phi(x) \rangle$. L'équation précédente se réécrit, en notant K la fonction noyau :

$$f(x) = \sum_{i=1}^{l} y_i \alpha_i K(x_i, x) + b \qquad (2)$$

A partir d'un ensemble d'apprentissage $\{(x_i, y_i)\}$, l'apprentissage d'une MVS consiste à déterminer les paramètres de la fonction f (les α_i et b) les meilleurs au sens de la maximisation de la marge. Les coefficients α_i et b sont en fait les solutions d'un problème de programmation quadratique. Les α_i sont non nuls pour un sous ensemble des points d'apprentissage que l'on appelle les vecteurs supports (VS) et sont nuls pour les autres.

Différentes approches ont été proposées pour appliquer les MVS à des données séquentielles. Dans (Jaakola et al 1998, 1999) l'idée consiste à se ramener à une représentation en dimension fixe, en exploitant un modèle génératif. On utilise un modèle génératif appris sur l'ensemble des données, λ_0, défini par un ensemble de paramètres θ. Puis on définit la nouvelle représentation $U(x)$ d'une séquence x comme le gradient du logarithme de la vraisemblance de x par λ_θ :

$$U(x) = \vec{\nabla}_\theta \log(P(x / \lambda_\theta)$$

Intuitivement, plus la norme de $U(x)$ est importante, plus le modèle λ_θ devrait être changé pour produire le signal x avec une forte vraisemblance. On a donc changé l'espace de représentation de toutes les données (séquences) d'apprentissage, qui sont maintenant

représentées dans un espace de dimension fixe (le nombre de paramètres de λ_θ), et on peut utiliser des MVS sur ces représentations.

(Bahlman 2002) utilise un système basé sur des prototypes (des exemples représentatifs de chacune des classes) et définit un noyau sur les données séquentielles, à base d'une distance entre séquences très utilisée en reconnaissance de la parole, le DTW (Dynamic Time Warping). Dans ce cas, le noyau entre deux séquences est défini par :

$$K(x,y) = e^{-A.d_{dtw}(x,y)+B}$$

Où A et B sont des paramètres du système fixés empiriquement. Des techniques similaires sont discutées dans (Watkins 2003).

(Moreno 2003) explore une troisième technique qui consiste à utiliser une correspondance entre donnée et modèle. Pour la tâche d'identification du locuteur visée, chaque donnée est un signal de parole de quelques dizaines de secondes qui est transformé en un modèle génératif (un mélange de lois gaussiennes) en apprenant ce modèle sur la donnée. A partir de cette association d'un modèle λ_x à une donnée x, les auteurs proposent d'utiliser un noyau probabiliste entre deux données x et y, qui traduit la différence entre les modèles génératifs associés λ_x et λ_y. Ce noyau est basé sur la divergence de Kullback Leibler entre les distributions de probabilités définies par les deux modèles. Cette approche n'est bien entendu viable que si le modèle utilisé est relativement simple et peut être appris sur une seule donnée de test.

3 Sélection de modèles génératifs à l'aide de machines à vecteurs supports

Nos travaux s'inspirent de (Moreno 2003), nous cherchons à utiliser des machines à vecteurs supports en exploitant une correspondance donnée-modèle. Notre travail diffère en deux points. Tout d'abord, la correspondance donnée-modèle ne fait pas intervenir d'apprentissage mais exploite une information a priori fournie par le concepteur. Notre approche peut donc sembler moins générique que celle de (Moreno 2003) mais il faut noter que l'apprentissage d'un modèle à partir d'une donnée unique n'est généralement pas faisable sans information a priori. Ensuite, notre but n'est pas d'obtenir une fonction discriminante performante mais uniquement de sélectionner des modèles génératifs élémentaires performants (cf. Fig. 1).

Dans la suite, on suppose que l'on dispose d'une association donnée modèle, l'apprentissage est donc réalisé à partir d'une base de données d'apprentissage $\{(x_i, y_i)\}$ et d'une base de modèles associés $\{(\lambda_{x_i}, y_i)\}$. Nous discutons maintenant des noyaux que nous avons utilisés. Une première méthode que nous avons envisagée consiste à définir explicitement $\phi(x)$. L'idée est ici de représenter une donnée x par les vraisemblances de cette donnée calculée par l'ensemble des modèles de toutes les classes. Dans le cas bi-classe, et en notant p_j^i le modèle défini à partir de la $j^{ème}$ donnée d'apprentissage de la classe i, $\phi(x)$ est défini par :

$$\phi(\mathrm{x}) = \left(\log\left(\mathrm{P}(\mathrm{x}/\mathrm{p}_1^1)\right),\ldots,\log\left(\mathrm{P}(\mathrm{x}/\mathrm{p}_{\mathrm{N}_1}^1)\right),\log\left(\mathrm{P}(\mathrm{x}/\mathrm{p}_1^2)\right),\ldots,\log\left(\mathrm{P}(\mathrm{x}/\mathrm{p}_{\mathrm{N}_2}^2)\right)\right)$$

On utilise alors un noyau, par exemple gaussien, entre deux données $\phi(x)$ et $\phi(y)$ du type $K_\Phi(x,y) = e^{-\gamma\|\phi(x)-\phi(y)\|^2}$ où γ est une constante. Afin de simplifier la procédure, on peut construire $\phi(x)$ à partir des scores calculés par un sous-ensemble restreint de modèles de chaque classe. Dans nos expériences, nous choisissons aléatoirement par exemple *10* modèles par classe et définissons $\phi(x)$ comme un vecteur de *Nx10* scores, où *N* est le nombre de classes considérées. Nous appellerons cette méthode le *Noyau Φ*.

Nous avons également utilisé le noyau proposé par Moreno, exploitant la divergence symétrisée de Kullback-Leibler entre deux modèles génératifs, λ_X et λ_Y, construit à partir de deux données *x* et *y*, nous nommons cette méthode *Noyau KL*. La divergence symétrisée s'écrit :

$$KL_{sym}(\lambda_x, \lambda_y) = \frac{1}{2}(KL(\lambda_x | \lambda_y) + KL(\lambda_x | \lambda_y))$$

Dans nos expériences, nous avons estimé ces divergences de *KL* sur les données d'apprentissage par :

$$KL(\lambda_x | \lambda_y) = \sum_{z \in BA} P(z/\lambda_X) \log \frac{P(z/\lambda_x)}{P(z/\lambda_y)}$$

Où *z* décrit l'ensemble des données d'apprentissage. Le noyau utilisé est défini par :

$$K_{KL}(\lambda_x, \lambda_y) = e^{-A.KL_{sym}(\lambda_x, \lambda_y)+B}$$

Enfin, nous avons utilisé un noyau extrêmement simple (appelé *Noyau Proba*) qui contient une information du même ordre que le noyau précédent mais plus fruste, défini par :

$$K_{\Pr oba}(\lambda_x, \lambda_y) = P(x/\lambda_y) + P(y/\lambda_x)$$

Pour terminer, bien que le noyau de Fisher ait été la première méthode utilisée, nous ne fournissons pas ici de résultats avec cette méthode, la raison étant qu'elle est plus délicate à employer et qu'elle nécessite un réglage manuel important. Nous présentons donc des résultats avec les noyaux K_Φ, K_{KL} et K_{Proba}.

4 Apprentissage discriminant des coefficients de mélange

Notre but est d'apprendre des modèles génératifs qui soient le plus discriminant, ces modèles génératifs sont des modèles de mélange de la forme donnée dans l'équation (1). L'utilisation de machines à vecteurs supports, avec l'un des noyaux définis au §3, permet de sélectionner les composantes des modèles de mélange des classes, les λ_i^c. Il faut ensuite optimiser les coefficients de mélange, les w_i^c, afin d'obtenir la meilleure discrimination possible. L'algorithme classique consiste à apprendre ces paramètres avec un critère de Maximum de Vraisemblance. Dans ce cas, l'apprentissage est réalisé indépendamment pour chaque classe et on n'introduit pas d'information discriminante à ce niveau.

Nous proposons ici de chercher les coefficients des modèles de mélange optimisant le critère discriminant :

$$V = \prod_{(x_i, y_i) \in \mathrm{BA}} P(y_i/x_i)$$

C'est à dire le produit des probabilités a posteriori des données d'apprentissage, les λ_i^c étant fixés. En prenant le logarithme, on cherche donc à maximiser :

$$J = \sum_{x \in \mathrm{BA}_1} \log P(C_1 / x) + \ldots + \sum_{x \in \mathrm{BA}_N} \log P(C_N / x)$$

Cela est réalisé à l'aide d'un algorithme de gradient en dérivant le critère J par rapport aux w_i^c . Un des effets espéré, et observé dans certains cas, de l'optimisation de ce critère est qu'une partie des poids w_i^c convergent vers 0.

Nous comparerons cette méthode d'apprentissage des coefficients avec un apprentissage non discriminant optimisant le critère de Maximum de Vraisemblance.

5 Expériences sur des données de dimension fixe

Les tests réalisés ici, sur des données en dimension fixe, ont pur but de mettre en évidence les particularités des différentes méthodes envisagées. Nous discutons brièvement de la correspondance donnée modèle puis nous décrivons les données et fournissons enfin des résultats expérimentaux.

5.1 Base de données

Nous utilisons ici une base de données extraite de la base *pbvowel*[1] (Klautau 2002), contenant des signaux de parole (voyelles). Ces signaux ont été prétraités et sont représentés, après extraction de caractéristiques, en deux dimensions qui correspondent aux deux premiers formants. Il y a 10 classes et nous utilisons environ 600 exemples en apprentissage et 600 en test. Le taux de reconnaissance plafond donné par les auteurs sur ces données est de 81.7%, il est obtenu par la méthode des *K* plus proches voisins.

5.2 Correspondance donnée modèle

Dans le cas de données en dimension fixe, nous avons utilisé une correspondance extrêmement simple. Le modèle associé à une donnée (un point) est une loi gaussienne de moyenne le point et de variance proportionnelle à l'identité. Pour un point x, le modèle associé, λ_x, est donc une loi gaussienne $N(x, \sigma^2. Id)$. La variance est la même pour tous les modèles et a été fixée empiriquement.

5.3 Résultats

Le tableau 1 montre les résultats en classification des différentes approches envisagées : le système discriminant obtenu avec la MVS avec différents noyaux ; les systèmes obtenus en sélectionnant les modèles λ_i^c à l'aide des MVS et en déterminant les w_i^c avec différents schémas d'apprentissage. Afin de comparer équitablement les méthodes, nous avons réglé les

[1] Pour plus de détails : http://www.laps.ufpa.br/aldebaro/repository/

paramètres des MVS de façon à obtenir un nombre de vecteurs support par classe identique pour toutes les expériences. Dans le tableau 1, le nombre de VS par classe est de 40.

	Noyau Proba	Noyau KL	Noyau Φ
Fonction discriminante			
MVS	78.67	76.17	73.33
Nb VS / classe	40	40	40
Modèles génératifs Apprentissage des coefficients de mélange			
Maximum de Vraisemblance	77.2	77.7	78.2
Critère discriminant (§4)			
Performance	79.7	79.3	79.2
# modèles / classe	39	27	37

Tab. 1 - *Performances des fonctions discriminantes apprises (MVS) et des méthodes utilisant les MVS pour la sélection des modèles.*

On constate tout d'abord que les résultats des fonctions discriminantes ne sont pas tous identiques, le noyau K_{Proba} (le noyau plus simple) donnant des meilleures performances. Si l'on examine les résultats obtenus avec les modèles génératifs dont les modèles élémentaires ont été sélectionnés par les MVS, on s'aperçoit que leurs performances sont plus élevées que celles des MVS correspondantes que les coefficients de mélange soient appris avec un critère discriminant ou un critère non discriminant. Les noyaux utilisés ne sont donc peut-être pas tous bien adaptés pour construire une fonction discriminante mais ils permettent de sélectionner relativement efficacement des modèles pour concevoir des modèles génératifs. On constate également que, quelle que soit la méthode utilisée pour sélectionner des modèles, les systèmes génératifs obtenus après apprentissage des coefficients de mélange présentent des performances très similaires. Enfin, on note que l'apprentissage discriminant des coefficients de mélange permet d'obtenir dans tous les cas les meilleurs résultats.

A titre de référence, nous avons appris, avec un critère de Maximum de Vraisemblance, des modèles génératifs du type donné dans l'équation (1) pour des tailles variant de 15 à 60. Les taux de reconnaissance varient de 78 à 78.5%. Par ailleurs, rappelons que la performance plafond est de 81.7%. Comparativement à ces résultats, les systèmes génératifs dont les modèles sont sélectionnés par MVS et dont les coefficients de mélange sont appris avec un critère discriminant sont donc légèrement supérieurs. Il s'agit d'un résultat encourageant d'autant que, lors de l'apprentissage discriminant des coefficients de mélange, certains des coefficients convergent vers 0 si bien que la taille des modèles des classes est plus réduite, notamment avec le noyau K_{KL}.

Notons que l'on peut tirer le même type de conclusions pour d'autres expériences dans lesquelles le nombre de vecteurs support est plus élevé.

On peut étudier plus en détail le fonctionnement de l'approche en visualisant les points (ou plutôt les modèles) de la base d'apprentissage qui ont été choisis comme vecteurs support (Figure 1) pour la construction des modèles génératifs. La figure 2 montre l'ensemble des données des 10 classes, ainsi que, en gras, les modèles appris par la méthode du Maximum de Vraisemblance (les vecteurs moyens des gaussiennes). La figure 3 montre de la même façon, les modèles sélectionnés par une MVS utilisant le noyau Φ. On voit bien

sur ces figures que l'exploitation des MVS permet de récupérer les modèles qui se situent à la frontière des exemples des deux classes alors que les modèles appris avec un critère de Maximum de Vraisemblance sont des modèles modélisant les zones les plus denses de chaque classe.

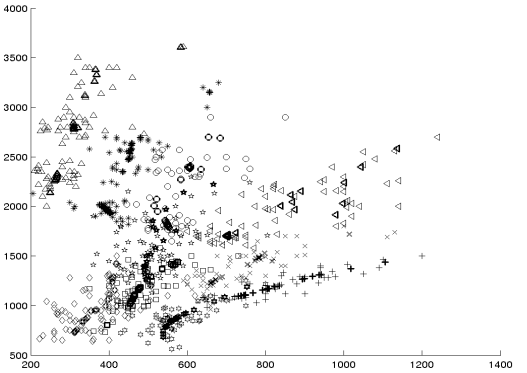

Fig. 2 - *Modèles élémentaires appris avec un critère non discriminant (MV) pour le problème de classification de voyelles.*

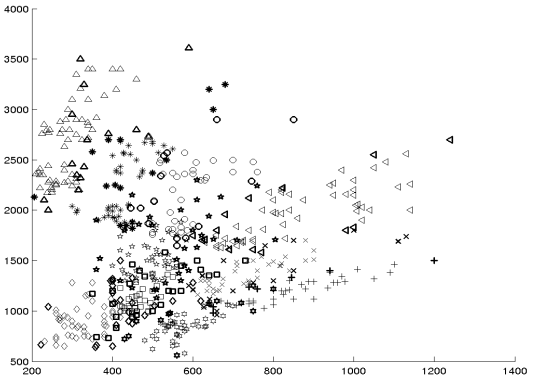

Fig. 3 - *Modèles élémentaires sélectionnés avec un MVS et le noyau K_Φ pour le problème de classification de voyelles.*

6 Expériences sur des données séquentielles

Nous présentons ici des tests réalisés sur des signaux d'écriture manuscrite en ligne. Un signal d'écriture manuscrite en ligne est un signal temporel constitué des coordonnées successives d'un stylo, il est capturé sur une tablette digitale ou via un stylo électronique. Nous présentons la base de données utilisée, discutons de la correspondance donnée modèle, puis nous fournissons des résultats expérimentaux.

6.1 Base de données

Nous avons travaillé sur une partie de la base UNIPEN (Guyon et al. 1994), qui est la base internationale de référence dans le domaine de l'écriture en ligne. Nos expériences portent sur des signaux correspondant aux 10 chiffres usuels écrits par environ 200 scripteurs. Nous utilisons 16000 exemples, 33% pour l'apprentissage et 66% pour le test. Chaque résultat d'expérience est un résultat moyen obtenu sur 3 expériences en faisant varier le tiers des données utilisées en apprentissage.

6.2 Correspondance donnée / modèle

Le signal d'écriture étant très variable (il existe de nombreux allographes pour tracer un même caractère) l'usage de modèles de mélanges ou de systèmes basés sur des prototypes de tracés typiques est extrêmement répandu. Un caractère est souvent modélisé par un mélange de modèles, par exemple des modèles Markoviens gauche droite, chacun de ces modèles correspondant à un allographe. L'apprentissage de ce type de modèles n'est pas aisé car le nombre d'allographes ainsi que la topologie des modèles Markoviens les modélisant doivent être fixés à la main. Un certain nombre de travaux ont été menés pour apprendre complètement les modèles de caractères à partir des données (Lee et al. 2001, Artières 2002, Marukatat et al. 2003). Ils sont basés sur la construction d'un MMC à partir d'un tracé et exploitent une représentation des tracés sous forme de séquence de codes directionnels.

Nous avons repris dans cette étude la procédure proposée dans (Artières 2002) que nous ne décrivons que brièvement ici. L'idée est de construire, à partir d'un tracé manuscrit en ligne originel, un MMC donnant une forte vraisemblance à des tracés ressemblant au tracé originel et de faibles vraisemblances à des tracés en différant. On note λ_x le modèle associé à un tracé x. La figure 4 présente schématiquement cette procédure. Le signal manuscrit est tout d'abord segmenté à l'aide d'un système markovien ergodique dans lequel chaque état représente un tracé élémentaire, il existe 36 tracés élémentaires (représentés dans la figure 5), des droites uniformément réparties entre 0° et 360° ainsi que des tracés légèrement convexes ou concaves. Le résultat de ce premier traitement est une représentation du tracé originel sous forme d'une séquence de tracés élémentaires (es_1, es_{21} et es_1 dans la figure 4). A partir de cette représentation, on construit un MMC gauche droite à trois états (à droite dans la figure 4). A chaque état est associée une loi de probabilité d'émission dérivée du tracé élémentaire correspondant. Par exemple, dans le premier et le troisième état du modèle de la Figure 4, le tracé élémentaire « idéal » est le tracé es_1 si bien que la loi d'émission donne de fortes probabilités pour des directions proches de la direction de ce tracé et faibles pour des directions différentes.

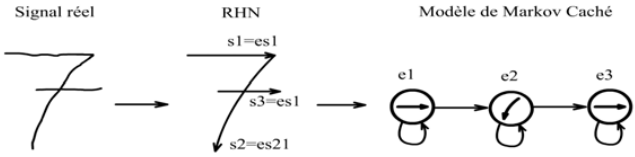

Fig. 4 - *Construction d'un MMC gauche droite à partir d'un tracé. Le signal manuscrit en ligne est segmenté en une séquence temporelle de tracés élémentaires, qui constitue un codage directionnel du tracé. Puis un modèle de Markov gauche-droite est construit à partir de cette séquence.*

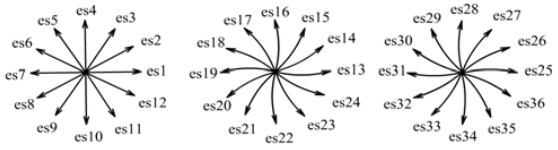

Fig. 5 - *Ensemble 36 tracés élémentaires utilisé pour représenter les formes en deux dimensions (Artières 2002). De gauche à droite : 12 tracés droits (notés es_1 à es_{12}), 12 convexes (es_{13} à es_{24}), 12 concaves (es_{25} à es_{36}).*

6.3 Résultats expérimentaux

Le tableau 2 résume les performances de différentes méthodes envisagées pour la classification des signaux des dix chiffres *0* à *9*. On remarque le même genre de phénomènes qu'avec les données en dimension fixe. Les performances des fonctions discriminantes à base de MVS sont très variables puisqu'elles vont de 86% jusqu'à presque 99% suivant le noyau utilisé. Il faut noter ici que la performance de 98 ,8 % obtenue par la fonction MVS avec le noyau K_{KL} est la meilleure performance en classification sur ces données, à notre connaissance. Par ailleurs, dans les systèmes génératifs pour lesquels les modèles élémentaires sont sélectionnés via des MVS, l'apprentissage des coefficients de mélange discriminant permet d'atteindre des résultats très supérieurs (pour le noyau K_{Proba}) ou avoisinant (pour les noyaux K_{KL} et K_{Φ}) les performances des fonctions discriminantes. Quel que soit le noyau utilisé, la performance des systèmes génératifs sont très proches, de l'ordre de 95% avec un apprentissage non discriminant et de l'ordre de 98% avec un apprentissage discriminant. Ainsi, si les noyaux ne sont pas forcément aussi bien adaptés pour construire une fonction discriminante, ils semblent équivalents pour la conception de systèmes génératifs. Pour terminer, il faut comparer ces résultats aux résultats obtenus par le système de (Marukatat et al, 2004) dont est inspiré ce travail sur la reconnaissance d'écriture en ligne. Ce système obtient une performance plafond de l'ordre de 97.5% sur ces données. Par rapport à ces performances, les systèmes obtenus ici réduisent le taux d'erreur de 20% pour les systèmes génératifs, et de 60% pour le système MVS avec le noyau K_{KL}.

Méthode	K_{Proba}	K_{KL}	K_{Φ}
Modèles génératifs Apprentissage des poids			
Max. de Vraisemblance	95.2	94.8	94.8
Critère discriminant			
Performance	97.7	97.9	98
Taille des modèles	73	87	50
Fonction discriminante (MVS)			
Performance	86.3	98.8	97.5
#VS / classe	90		
Méthode de référence (Marukatat 2004)			
Taille des modèles = 50	97.2		
Taille des modèles = 70	97.5		
Taille des modèles = 90	97.5		

Tab 2 - *Performance des méthodes utilisant les MVS comme fonction discriminante ou pour la sélection des modèles élémentaires (avec deux schémas d'apprentissage des coefficients de mélange) et du système de référence de (Marukatat 2004), non discriminant.*

7 Conclusion

Nous avons étudié dans ce papier des méthodes discriminantes pour l'apprentissage de mélanges de modèles génératifs. Ce type de modèle, particulièrement utilisé pour des données séquentielles, est généralement appris de façon non discriminante. Nous avons envisagé la possibilité de transformer le problème de façon à utiliser des machines à vecteurs support pour automatiquement sélectionner les modèles les plus pertinents pour la classification. Nous avons également proposé d'apprendre les coefficients de mélange des modèles avec un critère discriminant. Les performances obtenues montrent des résultats intéressants. Les systèmes génératifs obtenus sont plus performants que les modèles appris avec un critère non discriminant et présentent la même souplesse que ces derniers pour réaliser des tâches de segmentation de signaux.

Références

Artières T., Gallinari P. (2002), Stroke level HMMs for on-line handwriting recognition, International Workshop on Frontiers in Handwriting Recognition, pp 227-232.

Bahlmann C., Haasdonk B., Burkhardt H. (2002), On-line Handwriting Recognition using Support Vector Machines - A kernel approach. Int. Workshop on Frontiers in Handwriting Recognition, pp 49-54.

Chang Chih-Chung, Lin Chih-Jen, A library for Support Vector, http://www.csie.ntu.edu.tw/~cjlin/libsvm/

Guyon I., Schomaker L., Plamondon R., Liberman M., Janet S. (1994), UNIPEN project of on-line data exchange and recognizer benchmark, International Conference on Pattern Recognition.

Jaakkola T., Diekhans M., Haussler D. (1998), Exploiting generative models in discriminative classifiers, Advances in Neural Information Processing Systems 11, San Mateo, CA, pp 487-493.

Jaakkola T., Diekhans M., Haussler D. (1999), Using the Fisher kernel method to detect remote protein homologies, International Conference on Intelligent Systems for Molecular Biology, pp 149-158.

Klautau A. (2002), Classification of Peterson & Barney's vowels using Weka, Technical report, UFPA, (http://citeseer.nj.nec.com/klautau02classifcation.html).

Lee J.J., Kim J. and Kim J.H. (2001), Data-driven design of HMM toplogy for on-line handwriting recognition, International Journal of Pattern Recognition and Artificial Intelligence, Vol. 15, n° 1, pp 107-121.

Marukatat Sanparith (2004), Une approche générique pour la reconnaissance de signaux écrits en ligne, Thèse de doctorat, Université Paris 6, LIP6.

Moreno Pedro J., Ho Purdy P., Vasconcelos N. (2003), A Generative Model Based Kernel for SVM classification in Multimedia applications, NIPS.

Vapnik V, Ben-Hur A., Horn D., Siegelmann H.T (2001), A Support Vector Method for Hierarchical Clustering, Advances in Neural Information Processing Systems 13, pp 367-273.

Vapnik V. (1995), The Nature of Statistical Learning Theory, Eds Springer-Verlag, New York.

Watkins Chris (2003), Dynamic Alignment Kernels, Neural Information Processing Systems, Vancouver, Canada.

Summary

This paper investigates the development of discriminant methods for sequential data. A few techniques have been proposed to adapt discriminant models to such data, like Support Vector Machines. These techniques allow handling complete sequence classification but fail to perform segmentation tasks, i.e. recognizing the sequence of units (e.g. characters, phones) that correspond to a signal. Based on an association between data and models, we transform the problem of learning from training data into a problem of selecting appropriate generative models, thus enabling the use of Support Vector Machines for generative models learning. We compare a few kernels for this and report experimental results for two different classification problem.