Système de classification à deux niveaux de décision combinant approche par modélisation et machines à vecteurs de support.

Jonathan Milgram, Robert Sabourin, Mohamed Cheriet

Système de classification à deux niveaux de décision combinant approche par modélisation et machines à vecteurs de support

Jonathan Milgram — Robert Sabourin — Mohamed Cheriet

Laboratoire d'Imagerie, de Vision et d'Intelligence Artificielle

École de Technologie Supérieure de Montréal

milgram@livia.etsmtl.ca - {robert.sabourin, mohamed.cheriet}@etsmtl.ca

Résumé : *Il est possible de distinguer deux types de données pouvant causer des problèmes à un classifieur : les données ambiguës et les données aberrantes. Or, les algorithmes de classification peuvent être séparés en deux grandes catégories. Les approches agissant par séparation ont pour objectif de minimiser le premier type d'erreur, mais ne permettent pas de rejeter efficacement le deuxième type de données. Par contre, les approches agissant par modélisation sont adaptées à ce type de rejet, mais s'avèrent généralement peu discriminantes. Dans cet article nous proposons donc de combiner approche par modélisation et machine à vecteurs de support (SVM) au sein d'un système de classification à deux niveaux de décision. En outre, cette combinaison présente l'avantage de réduire la complexité de calcul associée à la prise de décision des SVM. Ainsi, nos expériences sur la base MNIST montrent qu'il est possible de maintenir les performances associées aux SVM, tout en réduisant significativement la complexité et en rendant possible la détection de données aberrantes.*

Mots-clés : *Système de classification, combinaison de classifieurs, machine à vecteurs de support, approche par modélisation, détection de données aberrantes.*

1 Introduction

Lors de la conception d'un système de reconnaissance de formes, l'objectif principal est de minimiser les erreurs de classification. Cependant, un autre critère important est la capacité à estimer une mesure de confiance dans la décision prise par le système. En effet, une telle mesure est essentielle pour permettre de ne pas prendre de décision lorsque le résultat de la classification est incertain. Ainsi, il est important de différentier deux types de rejet, correspondant à deux catégories de données délicates. Le rejet d'ambiguïté consiste, comme son nom l'indique, à filtrer les exemples ambigus. Le second type de rejet concerne les données aberrantes qui ne correspondent à aucune des classes du problème. On parle alors de détection d'« *outliers* », de rejet d'ignorance ou de rejet de distance. Or, parmi l'ensemble des techniques de classification, il est possible de distinguer deux catégories d'approches, celles agissant par modélisation et celles agissant par séparation.

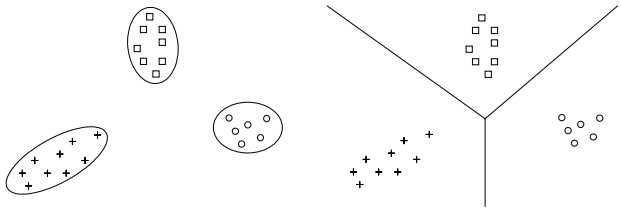

FIG. 1 – Deux catégories d'approches de classification

Le premier type d'approche cherche à déterminer un modèle le plus fidèle possible de chacune des classes, alors que l'objectif du second type est d'optimiser des frontières de décision de manière à séparer au mieux les classes. La décision est alors prise dans le second cas en se basant sur la position de l'exemple par rapport aux frontières et dans le premier cas en utilisant une mesure de similarité pour comparer la donnée à classifier à chacun des modèles.

Ainsi, comme il est montré dans [LIU 02], de part leur nature discriminante, les approches par séparation sont plus performantes pour traiter les données ambiguës, mais peu aptes à gérer les outliers. Par contre, les approches par modélisation permettent la détection de ces données aberrantes, mais s'avèrent peu discriminantes. A partir de ces constatations, les auteurs proposent deux options : soit fusionner les deux approches de manière interne au sein d'un système hybride, soit les combiner de manière externe. Ainsi, dans un article plus récent [LIU 03], les mêmes auteurs présentent un système hybride qui utilise un apprentissage discriminant pour améliorer les performances de leur approche par modélisation. Mais, bien que très satisfaisants les taux de reconnaissance obtenus restent inférieurs à ceux rendus possible par l'utilisation de machines à vecteurs de support (SVM).

Par conséquent, nous proposons de combiner approche par modélisation et SVM au sein d'un système de classification à deux niveaux de décision. L'idée consiste alors à utiliser dans un premier niveau de décision une approche par modélisation pour rejeter les outliers, classer les données ne présentant aucune ambiguïté et isoler les classes en conflits. Le second niveau de décision, utilisera ensuite les SVM appropriés pour

permettre une meilleure classification. De plus, cette combinaison présente l'avantage de réduire le principal fardeau des SVM : la complexité de calcul nécessaire à la prise de décision.

Bien qu'un certain nombre d'idées similaires aient été introduites dans des articles récents [BEL 03][PRE 03][VUU 03], notre système reste différent et original. En effet, une première combinaison entre approche par modélisation et approche par séparation a été proposée dans [PRE 03], mais les auteurs n'utilisent alors que quelques MLP pour améliorer les performances de leur premier classifieur et ne s'intéressent pas à la notion d'outlier. D'autre part, si le problème de la complexité liée aux SVM est traité dans [BEL 03], le système proposé ne possède qu'un seul niveau de décision. En effet, le MLP qui est utilisé comme premier classifieur sélectionne automatiquement ce qui lui semble être le « bon » SVM. De plus, l'utilisation de deux approches par séparation ne permet pas le rejet d'outliers. Pour résoudre ce problème, les auteurs proposent d'utiliser un autre SVM. Mais ceci nécessite alors de disposer d'une base conséquente d'outliers et risque de s'avérer très coûteux en terme de complexité. Enfin, plusieurs méthodes de détection de conflits ne se limitant pas à deux classes sont proposées dans [VUU 03]. Or, le premier niveau de décision utilise un ensemble de classifieurs qui s'avère particulièrement lourd. Ainsi, il est possible de se demander s'il ne serait pas préférable dans ce cas d'utiliser directement l'ensemble des SVM.

2 Approche par modélisation

2.1 Caractérisation du problème de classification

Bien que peu discriminante, ce type d'approche peut servir de premier niveau de décision et permettre de caractériser le problème de classification. Le degré d'appartenance à chacune des classes peut être évalué indépendamment par le biais de la distance entre l'exemple traité et le modèle de la classe considérée.

Trois cas de figure sont alors envisageables :

– Toutes les distances sont très grandes. Il s'agit alors vraisemblablement d'un *outlier* qui pourra être rejeté.

– Une seule distance s'avère faible. Il s'agit d'une donnée facile à classer. La décision peut donc être prise directement.

– Plusieurs distances sont faibles. Il s'agit d'une donnée ambiguë. Le conflit sera alors réglé dans un second temps par le ou les SVM appropriés.

Un exemple simple est présenté FIG. 2. Les distances à chacun des modèles des deux classes sont représentées par des lignes de niveaux en (a) et (b), alors que la combinaison de ces deux mesures montre en (c) comment il est possible d'isoler les outliers en utilisant le maximum des deux distances et en (d) comment détecter les cas d'ambiguïté en utilisant le minimum.

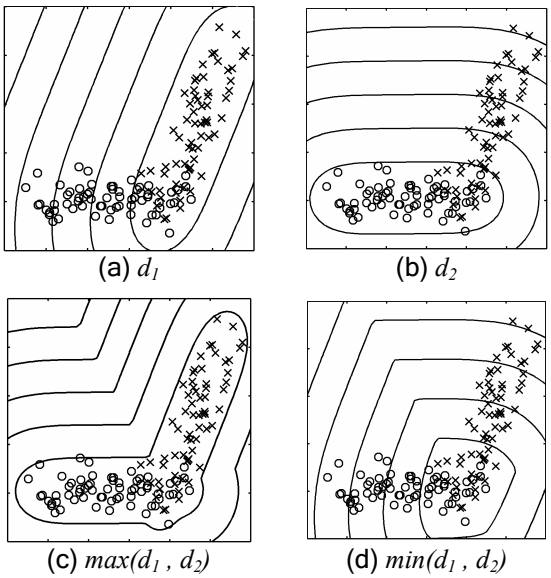

FIG. 2 – Utilisation d'une approche par modélisation pour caractériser le problème de classification.

D'autre part, la modularité de ce type d'approche présente l'avantage de permettre de traiter efficacement des problèmes où le nombre de classes est très grand. En effet, comme il est montré dans [OH 02] l'utilisation d'une approche globale tel qu'un réseau MLP s'avère inefficace lorsque le nombre de classes augmente comme dans le cas des 352 caractères coréens utilisés dans les adresses postales.

2.2 Modélisation à l'aide d'hyperplans

Afin de modéliser les différentes classes, nous avons choisi une méthode simple qui consiste à utiliser des hyperplans. Chaque classe ω_i est alors modélisée par l'hyperplan défini par les k premiers vecteurs propres extraits de la matrice de covariance Σ_i et passant par la moyenne μ_i des données de la classe. Le principal avantage d'une telle méthode réside dans sa capacité à interpoler les données de manière à obtenir des modèles très compacts et donc extrêmement légers en terme de complexité de calcul. Ainsi, pour tout point x de l'espace de représentation, le degré d'appartenance à une classe ω_i peut être évalué en calculant la distance d_i entre le point x considéré et sa projection P_i sur l'hyperplan modélisateur. La FIG. 3 illustre ceci à travers un exemple en deux dimensions.

$$d_i(x) = \left\| x - P_i(x) \right\| \quad [1]$$

$$P_i(x) = (x - \mu_i)\Psi_i\Psi_i^T + \mu_i \quad [2]$$

où Ψ_i représente la matrice contenant les k premiers vecteurs propres.

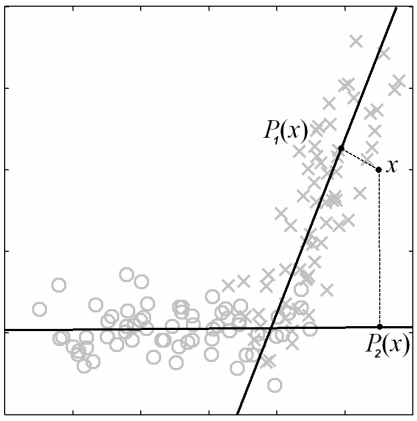

FIG. 3 – Exemple de modélisation d'un problème à deux dimensions. Les hyperplans modélisateurs sont alors définis par l'axe principal (k = 1) des données

En outre, cette approche ne nécessite que l'optimisation du paramètre k correspondant à la dimension des hyperplans modélisateurs. Toutefois, comme nous le verrons expérimentalement, il est important de ne pas négliger ce paramètre qui joue un rôle important dans la qualité de la modélisation. En effet, si k est trop petit, la perte d'information est importante et la modélisation peu précise. Si l'on considère le cas extrême où k = 0, la classe n'est alors modélisée que par le prototype μi correspondant à la moyenne. Par contre, si k est trop grand le modèle engendré ne sera plus discriminant. Si l'on considère l'autre cas extrême où $k = d$, d étant le nombre de caractéristiques, l'hyperplan englobe alors tous les points de l'espace de représentation et la distance de projection sera donc nulle quelque soit le point x.

3 Combinaison avec une approche par séparation

3.1 Détection de conflits

La première étape consiste donc à détecter les données ambiguës. Il nous semble alors préférable d'utiliser un nombre p de classes en conflit qui pourra varier dynamiquement suivant le cas de figure. La procédure que nous proposons pour déterminer la liste L_ω des classes en conflit consiste à normaliser les distances di à l'aide d'une fonction « softmax » [4] de manière à obtenir des mesures s_i d'appartenance aux classes ω_i. Puis, les différentes classes ω_i seront ordonnées de manière décroissante en fonction de leur valeur s_i. Enfin, on déterminera le nombre minimum p de classe nécessaire pour vérifier le critère suivant :

$$1 - \sum_{i=1}^{p} s_i < \varepsilon \qquad [3]$$

$$s_i = \frac{e^{-\alpha d_i}}{\sum_j e^{-\alpha d_j}} \qquad [4]$$

Le seuil ε contrôle donc la tolérance du premier niveau de décision. Plus sa valeur est petite, plus le nombre p de classes en conflit aura tendance à être grand et plus la décision sera reportée sur le second niveau de décision. Ainsi, une valeur de ε trop grande aura pour effet de ne quasiment jamais faire appel à l'approche discriminante. Par contre, une valeur trop petite entraînera une utilisation superflue du second niveau de classification et donc des temps de traitement excessifs.

3.2 Utilisation de machines à vecteurs de support

L'objectif du second niveau de décision est de retraiter les données ambiguës à l'aide de classifieurs discriminants de manière à prendre la décision parmi les p classes en conflits. Il semble donc préférable d'adopter une approche modulaire tel que la stratégie « *pairwise* » qui consiste à décomposer un problème à n classes en $n(n\text{-}1)/2$ sous problèmes binaires. Dans ce contexte, il est donc particulièrement intéressant d'utiliser des machines à vecteurs de support. En effet, les SVM sont des classifieurs binaires très discriminants. Les algorithmes que nous avons alors utilisés pour l'apprentissage et le test des SVM sont décrits dans [CHA 01]. Ainsi, nous avons entraîné les SVM correspondant à toutes les paires de classes. Mais, lors de la classification seuls les $p(p\text{-}1)/2$ classifieurs définis par la liste L_ω seront utilisés. La décision sera alors prise en effectuant un vote majoritaire et en cas d'égalité nous choisirons la classe ayant la plus petite distance d_i.

4 Résultats expérimentaux

De manière à tester l'approche proposée, nous avons choisi de nous intéresser à un problème de reconnaissance de formes très classique : la reconnaissance d'images de chiffres manuscrits isolés.

4.1 Base de données

Nos expériences ont été réalisées sur la base de données MNIST [LEC 98]. Il s'agit d'une base publique couramment utilisée et dont les résultats pour de nombreux classifieurs sont disponibles. Les images ont été normalisées en dimension (20×20) puis centrées dans une rétine 28×28 en faisant coïncider le centre de gravité du caractère avec le centre géométrique de la rétine. Les 50 000 premiers exemples de la base d'apprentissage seront utilisés pour l'entraînement de nos classifieurs et les 10 000 suivants pour la validation. Enfin, la base de test qui est composée de 10 000 exemples sera exclusivement réservée à l'évaluation des résultats finaux.

4.2 Approche par modélisation

Dans un premier temps, il est nécessaire de fixer la dimension k des hyperplans modélisateurs. Pour ce faire, nous avons utilisé la base de validation pour estimer l'effet de k sur les performances en classification (voir FIG. 4). Le meilleur résultat en validation (3.82 %) est alors obtenu pour k = 25 et a permis d'obtenir un taux d'erreur de 4.09 % sur la base de test.

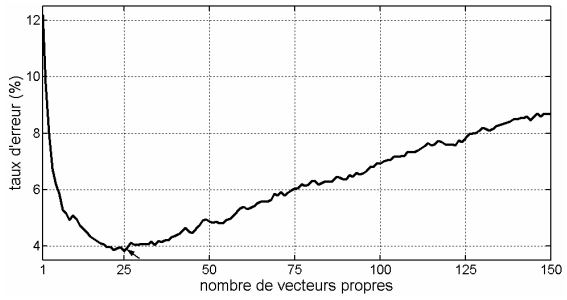

FIG. 4 – Effet de la dimension *k* des hyperplans modélisateurs sur les performances en classification

D'autre part, comme nous pouvons le constater TAB. 1, le label de l'exemple traité ne se trouve pas nécessairement parmi les deux premières solutions isolées par l'approche par modélisation. Ceci justifie donc l'utilisation d'une valeur dynamique du nombre *p* de classes en conflit.

position du bon label	1	2	3	>3
% de la base de test	95.91	2.82	0.72	0.55

TAB. 1 – Résultats de l'approche par modélisation sur la base de test.

Ainsi, bien que peu discriminante, cette approche très simple devrait permettre de classifier de manière fiable une grande partie des données et de réduire le nombre de classes à traiter par les SVM dans les cas ambigus.

4.3 Machines à vecteurs de support

L'apprentissage et le test des SVM ont été réalisés à l'aide du logiciel LIBSVM dont les algorithmes sont décrits dans [CHA 01]. Nous avons choisi d'utiliser le C-SVM avec un noyau gaussien : $K(x,y) = e^{-\gamma\|x-y\|^2}$. Les hyper paramètres γ et C ont été déterminés empiriquement en cherchant à minimiser le taux d'erreur sur la base de validation. Les valeurs retenues (C = 10 et γ = 0.0185) ont permis d'obtenir un taux d'erreurs de 1.47 % sur la base de validation. Cet ensemble de SVM utilise 11 118 vecteurs de support et permet d'obtenir un taux d'erreur de 1.54 % sur la base de test. Notons que ce résultat a été obtenu sans utiliser de connaissances *a priori* sur le type d'invariances des données.

4.4 Système de classification à deux niveaux de décision

Lors de l'implantation d'un système de classification, il peut être nécessaire de faire un compromis entre fiabilité et complexité. Le seuil de tolérance ε permet alors de contrôler ce type de compromis (voir FIG. 5). Ainsi, la base de validation pourra être utilisée pour fixer ce paramètre en fonction des contraintes liées à l'application. Notons que la complexité relative à la procédure de test est évaluée par le biais de la valeur moyenne du nombre de vecteurs de support distincts utilisés pour classifier les exemples de la base de données.

Par ailleurs, la valeur du paramètre α de la fonction « *softmax* » a été fixée à 6.0, ce qui correspond à la valeur minimisant l'erreur quadratique moyenne sur la base de validation.

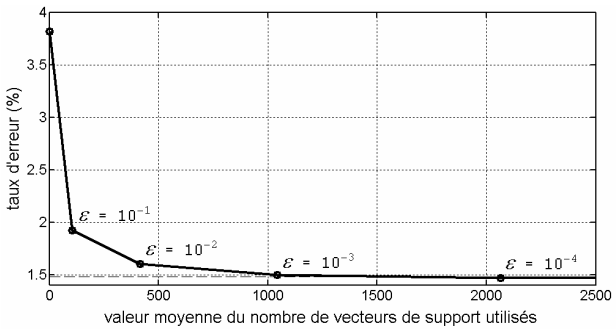

FIG. 5 – Utilisation de la base de validation pour évaluer l'effet du seuil ε

A partir des résultats obtenus sur la base de test (voir TAB. 2 et FIG. 6) il est possible de dégager un certain nombre de conclusions. Premièrement, il est intéressant de constater qu'en agissant seulement sur environ 10 % des données, il est déjà possible d'améliorer grandement les performances de l'approche par modélisation. En effet, lors de l'utilisation d'un seuil ε de 10^{-1}, le taux d'erreur passe de 4.09 % à 2.03 % en n'utilisant en moyenne que 0.18 SVM. Deuxièmement, il est important de constater qu'il est possible de maintenir les performances obtenues en utilisant tous les SVM « *pairwise* » en utilisant moins de 10 % de la complexité initiale. Effectivement, si l'on fixe $\varepsilon = 10^{-3}$, la moyenne du nombre de vecteurs de support utilisé est alors de 1 054.3 contre 11 118 initialement.

Seuil ε	10^{-1}	10^{-2}	10^{-3}	10^{-4}
Taux d'erreur (%)	2.03	1.62	1.53	1.5
# SVM utilisés	0.18	0.81	2.38	5.13
# VS utilisés	108.2	416.0	1054.3	2026.5

TAB. 2 – Résultats sur la base de test de notre système de classification à deux niveaux de décision

Ainsi, le nombre de SVM utilisés lors du test varie dynamiquement. L'histogramme correspondant est présenté FIG. 6. On peut alors constater qu'il est parfois nécessaire d'utiliser bien plus d'un SVM pour résoudre les conflits. Ceci prouve donc que notre approche par modélisation n'est pas assez précise.

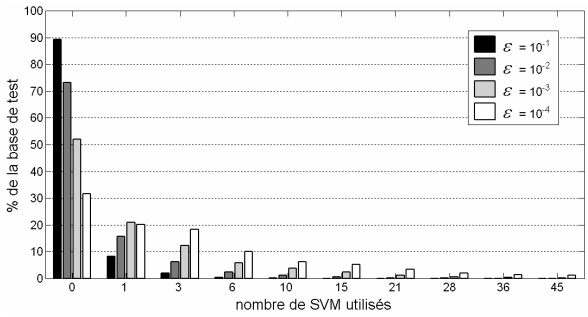

FIG. 6 – Effet du seuil ε sur la distribution du nombre de SVM utilisés

5 Conclusions et perspectives

Nous avons présenté dans cet article une nouvelle architecture qui présente plusieurs propriétés intéressantes pour la reconnaissance de caractères. Elle possède tout d'abord l'avantage d'être parfaitement modulaire et donc de pouvoir être appliquée à des problèmes où le nombre de classes est très grand. De plus, le système proposé combine les avantages des approches par modélisation, tel que la possibilité de détecter les outliers, avec l'important pouvoir discriminant des SVM tout en réduisant énormément le temps de traitement lié aux SVM.

Dans le futur, il sera intéressant de tester la capacité du système à effectuer les deux types de rejet. Les distances d_i pourront alors être utilisées pour le rejet d'ignorance. D'autre part, le schéma de vote majoritaire utilisé par le deuxième niveau de décision sera remplacé par une méthode d'estimation de probabilités, ce qui permettra d'effectuer efficacement le rejet d'ambiguïté.

Enfin, la principale limitation de notre système se situe au niveau du premier niveau de décision. De part sa linéarité, l'approche par modélisation que nous utilisons s'avère peu précise. En effet, la présence d'allographes de caractères peut conduire à ce que la distribution des données soit multimodale. Il semble donc préférable d'utiliser plus d'un hyperplan par classe. Ainsi, en améliorant le premier niveau de décision, il devrait être possible de réduire d'avantage la complexité du système.

Références

[BEL 03] BELLILI A., GILLOUX M., GALLINARI P., « An MLP-SVM combination architecture for offline handwritten digit recognition », *International Journal on Document Analysis and Recognition*, 2003, p. 244-252.

[CHA 01] CHANG C.-C., LIN C.-J., « LIBSVM: a library for support vector machines », rapport technique, national taiwan university, 2001. Logiciel disponible en ligne (http://www.csie.ntu.edu.tw/~cjlin/libsvm/).

[LEC 98] LECUN Y., BOTTOU L., BENGIO Y., HAFFNER P., « Gradient-based learning applied to document recognition », *Proceedings of IEEE,* vol. 86, n° 11, 1998, p. 2278-2324. Base de données Mnist disponible en ligne (http://yann.lecun.com/exdb/mnist/).

[LIU 02] LIU C.-L., SAKO H., FUJISAWA H., « Performance evaluation of pattern classifiers for handwritten character recognition », *International Journal on Document Analysis and Recognition*, 2002, p. 191-204.

[LIU 03] LIU C.-L., SAKO H., FUJISAWA H., « Handwritten digit recognition: benchmarking of state-of-the-art techniques », *Pattern Recognition*, 2003, p. 2271-2285.

[OH 02] OH I.-S., SUEN C., « A class-modular feedforward neural network for handwriting recognition », *Pattern Recognition*, vol. 35, 2002, p. 229-244.

[PRE 03] PREVOST L., MICHEL-SENDIS C., MOISES A., OUDOT L., MILGRAM M., « Combining model-based and discriminative classifiers: application to handwritten character recognition », *Int. Conference on Document Analysis and Recognition*, 2003, p. 31-35.

[VUU 03] VUURPIJL L., SCHOMAKER L., VAN ERP M., « Architectures for detecting and solving conflicts: two-stage classification and support vector classifiers », *International Journal on Document Analysis and Recognition,* 2003, p. 213-223.

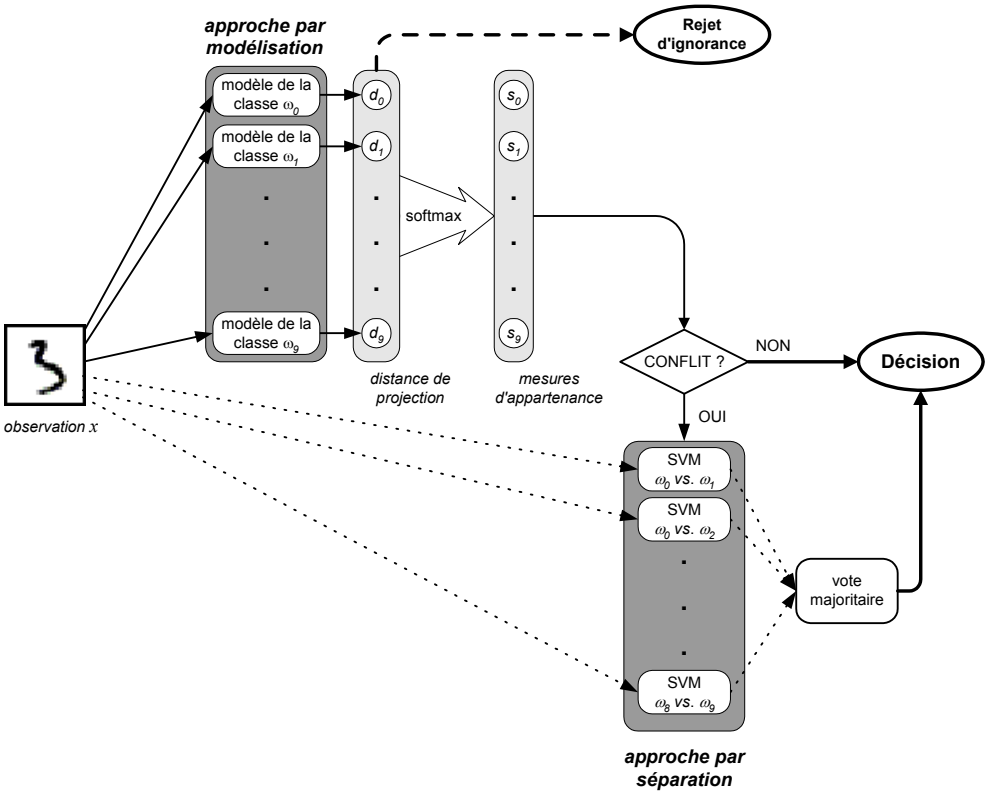

FIG. 7 – Vue d'ensemble de notre système de classification à deux niveaux de décision