

11ème Colloque Africain sur la Recherche en Informatique et Mathématiques

Approche de sélection d'attributs pour la classification basée sur l'algorithme RFE-SVM

Mouhamadou Lamine SAMB*, Fodé CAMARA*, Samba NDIAYE*

Yahya SLIMANI** --- Mohamed Amir ESSEGHIR***

* Département Mathématiques-Informatique, Faculté des Sciences et Techniques, Université Cheikh Anta Diop de Dakar SENEGAL

{mouhamadou81.samb, fode.camara, [samba.ndiaye](mailto:samba.ndiaye@ucad.edu.sn)}@ucad.edu.sn

** Département d'Informatique, Faculté des Sciences Université Tunis TUNISIE
yahya.slimani@fst.rnu.tn

*** Laboratoire de Génie Informatique et d'Automatique de l'Artois France mohamedemir@yahoo.fr

RÉSUMÉ. La problématique de filtrage de données ou de sélection d'attributs pour la classification représente un axe de recherche très actif dans le domaine du datamining (fouille de données) et en optimisation. La nature combinatoire du problème impose le développement de techniques spécifiques (utilisation de filtres, d'enveloppes etc.) ou hybrides combinant plusieurs procédés d'optimisation. Dans ce contexte, l'approche basée sur l'algorithme *RFE-SVM (Recursive Feature Elimination with Support Vector Machine)* s'est distinguée par l'intégration du filtrage dans le processus d'apprentissage basé sur les machines à vecteurs de supports ou *SVM*. Cependant, l'utilisation de l'algorithme *RFE-SVM* a montré quelques limites comme le fait d'être complètement glouton, c'est-à-dire ne permettant pas les retours en arrière. Dans cet article, nous avons proposé une approche de sélection d'attributs pour pallier cette limite de l'algorithme *RFE-SVM*. Notre approche consiste à combiner l'algorithme *RFE-SVM* avec des opérateurs de recherche locale, issus des domaines de la recherche opérationnelle et de l'intelligence artificielle.

ABSTRACT. The feature selection for classification is a very active research field in data mining and optimization. Its combinatorial nature requires the development of specific techniques (such as filters, wrappers, genetic algorithms, and so on) or hybrid approaches combining several optimization methods. In this context, the support vector machine recursive feature elimination (SVM-RFE), is distinguished as one of the most effective methods. However, the RFE-SVM algorithm is a greedy method that only hopes to find the best possible combination for classification. To overcome this limitation, we propose an alternative approach with the aim to combine the RFE-SVM algorithm with local search operators based on operational research and artificial intelligence.

MOTS-CLÉS : Datamining, Classification, Apprentissage supervisé, Sélection d'attributs, Machines à Vecteurs de Supports (SVM), Recursive Feature Elimination (RFE), Recherche locale.

KEYWORDS: Data mining, Classification, Supervised classification, Feature selection, Support Vector Machines, Recursive Feature Elimination (RFE), Local search.

CARI' 2012 - Algiers, Algeria

1. Introduction

La sélection d'attributs est un sujet de recherche très actif depuis une dizaine d'années dans les domaines de l'apprentissage artificiel, de la fouille de données, du traitement d'images, et de l'analyse de données en bioinformatique [1, 2, 3]. Elle consiste à choisir parmi un ensemble d'attributs de grande taille, un sous-ensemble d'attributs intéressants pour le problème étudié. Cette problématique peut concerner différentes tâches d'apprentissage ou de fouille de données, mais nous parlerons dans ce papier de la sélection d'attributs réalisée pour la classification supervisée. Dans ce cas, l'objectif de la sélection est de trouver un sous-ensemble optimal d'attributs qui ait les propriétés suivantes : il doit être composé d'attributs pertinents et doit chercher à éviter les attributs redondants. De plus, cet ensemble doit permettre de satisfaire au mieux l'objectif fixé, à savoir la précision de l'apprentissage, la rapidité de l'apprentissage ou bien encore l'applicabilité du classifieur proposé [3, 6]. Dans ce papier, nous avons choisi d'étudier, l'algorithme *RFE-SVM* [1] qui est un algorithme de type *embedded* [4] basé sur l'élimination *backward*¹ et exploitant les *SVM* [9], de façon récursive pour sélectionner un sous-ensemble d'attributs optimal. Il a été utilisé en bioinformatique pour l'analyse du niveau d'expression de gènes et dans l'analyse de données de transcriptome [10]. Ceci a montré que *RFE-SVM* [1] sélectionne de très bons sous-ensembles d'attributs. Cependant, son utilisation a montré quelques limites comme le fait d'être complètement glouton. Dans cet article, nous avons proposé une approche de sélection d'attributs pour pallier cette limite. Afin d'évaluer les apports de notre approche, nous avons réalisé une série d'expérimentations sur trois bases de données disponibles dans le site *UCI Machine Learning Repository* [5]. Les résultats de ces expérimentations sont prometteurs et confirment l'intérêt d'utiliser des recherches locales. Le reste du papier est structuré comme suit. La section 2, introduit les concepts et les notions utilisés dans ce papier. Nous décrivons dans cette section, les algorithmes *SVM* [9], *RFE-SVM* [2], et les opérateurs de recherche locale [7, 8], qui seront utilisés dans la suite de ce papier. La proposition est présentée dans les sections 3 et 4. Dans la section 3, nous expliquons l'approche de sélection d'attributs proposée dans ce papier. Les résultats obtenus lors des expérimentations sur des bases de données benchmarks sont détaillés dans la section 4. Enfin, nous concluons ce papier par le bilan des apports de notre contribution ainsi que par la présentation rapide de quelques perspectives ouvertes par notre travail.

¹ L'ensemble total des attributs est considéré au départ de la procédure itérative, et à chaque itération on supprime un attribut. On l'appelle également approche descendante.

2. Préliminaires

2.1. La sélection d'attributs

Les algorithmes de sélection permettent d'extraire une information non redondante et pertinente, en vue d'une exploitation efficace des bases de données. Ils font l'objet d'une littérature abondante depuis une dizaine d'années [1, 2, 3, 4] ; ils sont répartis en trois groupes principaux : les « filters » [1], les « wrappers » [1] et les « embedded » [4]. Les méthodes filtres (*filter methods*) opèrent directement sur le jeu de données et fournissent une pondération, un classement ou un ensemble de variables en sortie. Ces méthodes ont l'avantage d'être rapides et indépendantes du modèle de classification, mais au prix de résultats inférieurs. Les méthodes enveloppes (*wrapper methods*) effectuent une recherche dans l'espace des sous-ensembles de variables, guidée par le résultat du modèle, par exemple les performances en validation croisée sur les données d'apprentissage. Elles ont souvent de meilleurs résultats que les méthodes de filtrage, mais au prix d'un temps de calcul plus important [1, 2]. Enfin, les méthodes embarquées (*Embedded methods*) utilisent l'information interne du modèle de classification (par exemple, le vecteur de poids dans le cas des *SVM*), ces méthodes sont donc proches des méthodes d'enveloppes, du fait qu'elles combinent le processus d'exploration avec un algorithme d'apprentissage sans étape de validation, pour maximiser la qualité de l'ajustement et minimiser le nombre d'attributs [1, 4].

2.2. Algorithme RFE-SVM

RFE-SVM [1] est un algorithme de sélection d'attributs pour la classification supervisée. Cet algorithme fait partie des algorithmes de type « *Embedded* ». En effet, il intègre le filtrage dans le processus d'apprentissage *SVM* dans le but d'évaluer chaque sous-ensemble grâce à un classifieur *SVM* mais aussi pour avoir des informations sur la contribution de chaque attribut sur la construction de l'*hyperplan séparateur*.

2.1.1. Machine à vecteurs de support (SVM)

Les machines à vecteurs de support [9] (*SVM* pour « Support Vector Machines ») sont des techniques largement répandues en apprentissage statistique, elles ont eu beaucoup de succès dans quasiment tous les domaines où elles ont été appliquées. Dans ce papier, nous nous limitons au cadre de la classification linéaire. Dans ce cas, l'échantillon d'apprentissage est de la forme D avec :

$$D = \{(x_1, y_1), \dots, (x_n, y_n)\} \text{ et } y_i = \{\pm 1\}, \quad (1)$$

Le problème consiste donc à trouver un hyperplan tel que : les données des étiquettes de classe +1 et -1 se trouvent de chaque côté de l'hyperplan et la distance des vecteurs les plus proches de l'hyperplan (pour chacune des deux classes) est maximale. Ces vecteurs sont appelés *vecteurs de support* et la distance de ceux-ci par rapport à l'hyperplan constitue la *marge optimale*. D'une façon plus formelle, notre objectif est de trouver un hyperplan $w \cdot x + b$, $w \in R$ et $b \in R$ qui sépare les deux

classes avec la plus grande marge. La recherche de la marge optimale permettant de déterminer les paramètres w et b de l'hyperplan conduit à un problème d'optimisation quadratique qui consiste (dans le cadre général) à minimiser :

$$\{\|w\|^2 + C \sum \epsilon_i |y_i (w \cdot \Phi(x_i) + b)| \geq 1 - \epsilon_i, \epsilon_i \geq 0\} \quad (2)$$

où C est un paramètre de compromis entre la marge et les erreurs², ϵ_i est une variable ressort associée à l'observation x_i , et ϕ est une transformation. Le problème peut être résolu (entre autre) par la méthode Lagrangienne d'optimisation quadratique avec contraintes (formulation duale) pour maximiser la marge [9] :

$$\{\sum \alpha_i - (1/2 \sum \alpha_i \alpha_j y_i y_j K(x_i, x_j)) \mid 0 \leq \alpha_i \leq C, \sum \alpha_i y_i = 0\} \quad (3)$$

où α_i est le multiplicateur Lagrangien associé au vecteur x_i . Si la valeur de α_i est non-nul alors x_i est un vecteur de support et $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ est le noyau de transformation. Le noyau [9] d'un SVM est une fonction symétrique définie-positive qui permet de projeter les données dans un espace transformé de grande dimension dans lequel s'opère plus facilement la séparation des classes.

La décision est obtenue selon (le signe de) la fonction :

$$f(x) = \text{sign}[\alpha_i y_i K(x_i, x) + b] \quad (4)$$

2.1.2. RFE-SVM

L'algorithme RFE-SVM est une méthode de type *embedded* basée sur l'élimination *backward* et utilisant les SVM pour sélectionner un sous-ensemble d'attributs optimal non redondants. La méthode repose sur l'estimation de poids relatifs à l'optimisation d'un problème de discrimination linéaire, ce problème étant résolu à l'aide d'une machine à vecteurs de support (SVM). Il est montré dans [1] que le coût de suppression d'une caractéristique est de l'ordre de $\{w_j^2\}_{j=1}^D$ où w_j est le poids attribué à l'attribut j lors de la phase de construction du SVM [9]. La procédure de sélection est décrémente et élimine progressivement les attributs de faible poids. L'algorithme est décomposé en trois étapes :

Tant qu'il reste des attributs,

- Entraîner le classifieur SVM
- Calcul des w_j^2
- Supprimer le (les) attribut(s) correspondant au(x) poids le(s) plus faible(s).

2.1.3. Procédure de recherche locale

Les opérateurs de recherche locale [7, 8] sont souvent présentés comme des méthodes qui peuvent être à la fois intuitives et très efficaces pour résoudre de manière

² Le choix de C est critique si les données sont bruitées.

approchée des instances de problèmes combinatoires, lorsque ces instances possèdent des solutions. Ces opérateurs de recherche locale s'appuient sur le principe suivant : partant d'une solution quelconque s , qui peut être prise au hasard ou peut être construite par un algorithme dans l'espace de recherche S (dans notre cas l'algorithme *RFE-SVM*), ces méthodes cherchent à améliorer, de façon itérative, la solution existante en explorant le voisinage de celle-ci. Un opérateur de recherche locale se caractérise par deux éléments : une fonction d'évaluation et une fonction de voisinage. D'ailleurs le nom des fonctions de voisinage est utilisé pour désigner chaque opérateur utilisé dans ce travail. Il s'agit des opérateurs *Bit-Flip*[7] et *Attribute-Flip*[7]. Nous allons dans ce qui suit donner les détails de chacun des deux opérateurs cités précédemment.

- Opérateur *Attribute-Flip* : Dans le cas de l'opérateur *Attribute-Flip*, la fonction de voisinage est basée sur l'opération enlever/ajouter qui enlève un attribut f_j de la solution X et ajoute un autre attribut f_i appartenant aux sous-ensembles d'attributs supprimés. L'opérateur *Attribute-Flip* est illustré par l'équation 5.

$$NB_{AF} = \{X | X = S \cup \{f_i\} - \{f_j\}, \forall f_i \in X, f_j \notin X\} \quad (5)$$

- Opérateur *Bit-Flip* : L'opérateur *Bit-Flip* lui aussi est basé sur l'opération enlever/ajouter qui retire un attribut ai de la solution X ou ajoute un autre attribut aj appartenant au sous-ensemble d'attributs supprimés. Ici on ne fait pas l'échange, mais cet opérateur va explorer le voisinage de la solution retournée par *RFE-SVM* en ajoutant ou en supprimant un attribut. La fonction de voisinage de l'opérateur *Bit-Flip* (*BF*) est illustrée par l'équation 6.

$$NH_{BF}(S) = \{X | X = NH^+(S) \cup NH^-(S)\} \quad (6)$$

avec $NH^+(S)$ et $NH^-(S)$ qui désignent, respectivement, les solutions voisines obtenues en ajoutant ou en supprimant un attribut à la solution courante.

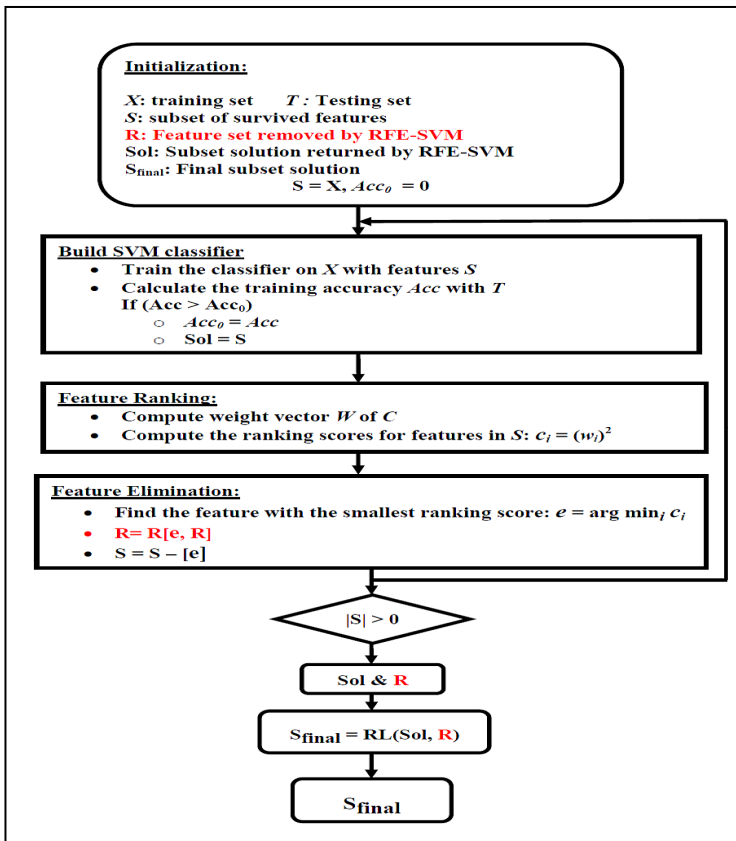
3. Notre Proposition

3.1. Définition du problème

Nous pouvons formaliser notre problème comme suit : soit X l'ensemble d'apprentissage, R le sous ensemble d'attributs éliminés par l'algorithme *RFE-SVM* et S la solution retournée par ce dernier, qui nous le rappelons est un algorithme complètement glouton. Notre approche consiste donc à chercher $S' \in N(S)$ tel que $SVM(S') > SVM(S)$ et $\forall S'' \in N(S), SVM(S') > SVM(S'')$. Notons que $N(S)$ représente le voisinage de la solution S . Pour atteindre cet objectif, nous utilisons les opérateurs *Bit-Flip* et *Attribute-Flip* décrits ci-dessus.

3.2. Notre algorithme

Nous proposons une démarche de sélection d'attributs suivant un processus à deux phases (voir l'organigramme ci-dessous). La première phase, appelée **phase d'initialisation**, permet de construire une solution initiale pour débiter la recherche. Nous proposons d'appliquer l'algorithme *RFE-SVM* [1] afin d'obtenir une solution initiale qui ne soit pas trop mauvaise. Rappelons que *RFE-SVM* est très utilisé dans la littérature mais le caractère glouton de cet algorithme fait qu'il ne retourne pas la solution optimale. A la deuxième phase, que nous appelons **phase d'amélioration**, nous allons introduire de la recherche locale [7, 8] afin d'améliorer la qualité de la solution retournée par *RFE-SVM*. L'organigramme de notre algorithme de sélection d'attributs et le code source de l'opérateur de recherche locale que nous avons utilisé sont donnés ci-dessous:



Algorithme 1 Procédure de recherche locale

input :

S: Solution retournée par RFE-SVM
S': Attributs supprimés par RFE-SVM
Cla: le classifieur utilisé

output :

S_{local} : solution retournée par la recherche locale

Begin

$S_1 \leftarrow S, S_{best} \leftarrow S_1$

$Stop \leftarrow false$

repeat

$Sol_{list} \leftarrow NH(S_1, S')$

$\forall Sol \in Sol_{list} Evaluate(Sol, Cla)$

$S_1 \leftarrow getBest(Sol_{list})$

if $S_1.fitness > S_{best}.fitness$ **then**

$S_{best} \leftarrow S_1$

else

$Stop \leftarrow true$

end if

until ($Stop = true$)

$S_{local} \leftarrow S_{best}$

Return (S_{local})

End

Afin de comparer les résultats de notre algorithme présenté ci-dessus, nous avons choisi d'évaluer les performances en terme de précision du classifieur construit à l'aide de trois jeux de données publics, facilement accessibles et qui sont utilisés dans de nombreux travaux concernant la classification des données. En pratique, nous divisons chaque base de données comme suit: les deux tiers des données sont utilisées pour la sélection des caractéristiques et l'apprentissage des classifieurs, la partie restante étant utilisée pour l'évaluation i.e comme ensemble test. La performance est mesurée par le taux de bonne classification moyennée sur les ensembles de test. Les résultats sont présentés dans le tableau 1, où N désigne le nombre d'attributs retenu par l'algorithme de la première ligne et P la précision du classifieur construit sur les N attributs.

<i>Datasets</i>	<i>RFE -SVM</i>	<i>RFE -SVM +BF</i>	<i>RFE -SVM +AF</i>
<i>Ionosphere</i>	N = 5, P = 85.714%	N= 6, P = 85.714%	N = 5, P = 86.286%
<i>SpamBase</i>	N = 39, P=89.606%	N= 38, P=89.729%	N = 39, P=89.704%
<i>Spect Heart Data</i>	N = 5, P = 71.123%	N= 6, P = 72.727%	N = 5, P = 73.262%

Tableau 1. Comparaisons des résultats de classification.

En nous basant sur les résultats issus des expériences que nous avons réalisées, nous pouvons affirmer que notre approche (*RFE-SVM+BF* et *RFE-SVM+AT*) de sélection d'attributs est très bien fondée. En effet, sur les trois ensembles de données que nous avons utilisés dans ce papier, notre algorithme de sélection arrive à améliorer la qualité de la solution retournée par l'algorithme *RFESVM*, et nous avons ainsi produits des classifieurs plus performants que ceux produit par l'algorithme *RFE-SVM* de base..

5. Conclusion

Dans ce papier, nous avons proposé une variante de l'algorithme *RFE-SVM* qui est un algorithme de sélection d'attributs pour la classification supervisée. Celui-ci exploite essentiellement la richesse du bagage théorique sur lequel sont basées les machines à vecteurs supports (*SVM*). L'analyse de l'existant nous a permis de noter que la principale faiblesse de *RFE-SVM* est liée au fait qu'il soit glouton (i.e. ne fait pas de retours en arrière). La variante que nous avons proposée dans ce papier, autorise des retours en arrière dans le but d'améliorer la qualité de la solution retournée par l'algorithme *RFE-SVM*. Les résultats issus des expérimentations ont été concluants et confirment l'intérêt d'utiliser des recherches locales. En effet, malgré quelques erreurs de classification, nous avons pu montrer que la réinsertion d'attributs éliminés lors du processus de classification contribuait à l'amélioration de la qualité de la solution retournée par l'algorithme *RFE-SVM*, produisant ainsi un classifieur plus performant.

Bien que les résultats obtenus soient intéressants et encourageants, beaucoup de points sont susceptibles d'être étudiés dans le cadre de travaux futurs. Donc en guise de perspectives, nous comptons continuer à expérimenter notre algorithme sur d'autres datasets benchmarks afin de voir le comportement de notre approche sur d'autres bases de très grandes tailles. Mais aussi, utiliser au sein de notre procédure de sélection d'attributs toutes les variantes de *RFE-SVM*. Enfin, nous envisageons d'étudier la classification multi-classe, pour voir comment se comporte notre algorithme dans ce cas de figure.

6. Bibliographie

- [1] Guyon, Weston, Barnhill, and Vapnik, “*Gene selection for cancer classification using support vector machines*,” *MACHLEARN: Machine Learning*, vol. 46, 2002.
- [2] P. A. Mundra and J. C. Rajapakse, “*SVM-RFE with relevancy and redundancy criteria for gene selection*,” in *PRIB*, J. C. Rajapakse, B. Schmidt, and L. G. Volkert, Eds., vol. 4774. Springer, 2007, pp. 242–252.
- [3] H. Liu and L. Yu, “*Toward integrating feature selection algorithms for classification and clustering*,” *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 4, pp. 491–502, 2005.
- [4] T. N. Lal, O. Chapelle, J. Weston, and A. Elisseeff, *Embedded Methods*. Springer, Nov. 20 2004.
- [5] “Uci machine learning repository,” <http://archive.ics.uci.edu/ml/datasets>.
- [6] A. K. Jain and D. E. Zongker, “*Feature selection: Evaluation, application, and small sample performance*,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 2, pp. 153–158, 1997.
- [7] M. A. Esseghir, G. Goncalves, and Y. Slimani, “*Memetic feature selection: Benchmarking hybridization schemata*,” in *HAI(1)*, 2010, pp. 351–358.
- [8] J. C. H. Hernandez, B. Duval, and J.-K. Hao, “*Svm based local search for gene selection and classification of microarray data*,” in *BIRD*, 2008, pp. 499–508.
- [9] V. N. Vapnik, “*The nature of statistical learning theory*”. New York, NY, USA: Springer-Verlag New York, Inc., 1995.
- [10] Sridhar Ramaswamy, Pablo Tamayo, Ryan Rifkin, Sayan Mukherjee, Michael Angelo, Christine Ladd, Michael Reich, P. Mesirov, Tomaso Poggio, William Gerald, Massimo Loda, Eric S. L., and Todd R. Golub. *SUPPLEMENTARY INFORMATION multi-class cancer diagnosis using tumor gene expression signatures*, December 27 (2001).