SÉLECTION DE VARIABLES POUR LA CLASSIFICATION BINAIRE EN GRANDE DIMENSION : COMPARAISONS ET APPLICATION AUX DONNÉES DE BIOPUCES

Badih GHATTAS*, Anis BEN ISHAK** [1]

RÉSUMÉ

Dans cet article nous nous proposons de comparer trois méthodes récentes de sélection de variables dans le cadre de la classification binaire. Le contexte auquel nous nous intéressons ici est celui où le nombre de variables est très grand et beaucoup plus important que le nombre d'observations, comme c'est le cas pour les données issues des biopuces. Les approches comparées sont de type SVM, GLM sous contraintes de type L_1 et Forêts Aléatoires.

Mots-clés : Biopuces, Bootstrap, Forêts Aléatoires, Hiérarchies de variables, Machines à Vecteurs Supports, Sélection de variables, Méthodes séquentielles, Modèles Linéaires Généralisés, Validation croisée.

ABSTRACT

In this paper we compare three methods for selecting important features in binary classification. We focus on the case where the sample size is smaller than the number of variables. The three approaches used are based on Support Vector Machines, L_1 constrained Generalized Linear Models and Random Forests.

Keywords : Bootstrap, Cross validation, Feature selection, Forward selection, GLM-path, Microarray data, Random Forests, Ranking rules, Support Vector Machines, SVM-based criteria.

* Institut de Mathématiques de Luminy (IML), CNRS Marseille, France, 163, avenues de Luminy, Marseille Cedex 9. ghattas@iml.univ-mrs.fr

** BESTMOD – Université de Tunis, Institut Supérieur de Gestion, 41, avenue de la liberté, Cité Bouchoucha, le Bardo 2000, Tunisie. ishak@iml.univ-mrs.fr

1. La réalisation de ce travail a bénéficié du soutien de l'ambassade de France en Tunisie et du CMCU.

1. Introduction

Supposons que nous disposons d'un échantillon $(X_1, Y_1), (X_2, Y_2), ..., (X_n, Y_n)$ d'observations indépendantes identiquement distribuées d'un couple de variables aléatoires $(X, Y) \in (\mathbb{R}^p, \{-1, 1\})$. Nous souhaitons ajuster un modèle de régression de la variable Y en fonction du vecteur $p-$dimensionnel des variables explicatives X :

$$Y = f(X) + \epsilon$$

où f est une fonction inconnue, et ϵ une erreur aléatoire. Certaines composantes de X peuvent être redondantes ou même parfois nuisibles pour la prévision de Y et peuvent donc être assimilées à du bruit qui nuit aux performances prédictives du modèle obtenu.

Nous nous intéressons ici à la réduction de la dimension de X, sans pour autant transformer ses composantes comme par exemple dans les méthodes factorielles. Dans ce contexte les méthodes existantes sont réparties en trois grandes catégories (cf. Kohavi *et al.* (1997), Guyon *et al.* (2003)) selon le type du critère de sélection et la façon dont il est pris en compte dans la procédure de classification. La première catégorie, dite « *filter* » (Dudoit *et al.* (2002)), évalue l'importance des variables en utilisant un critère statistique indépendant a priori du modèle. La deuxième catégorie, dite « *wrapper* », intègre les performances prédictives de la règle de classification dans la procédure de recherche et d'évaluation des sous-ensembles de variables pertinentes. Les méthodes wrapper utilisent certaines propriétés du modèle utilisé. Quant à la troisième catégorie, dite « *embedded* » (Weston *et al.* (2003)), elle combine la sélection de variables et l'estimation du modèle en une seule tâche. Nous nous plaçons dans la deuxième catégorie, dans le cas où la variable à prédire Y est binaire et p très grand.

La sélection de variables dans ce sens a fait l'objet de nombreux travaux, anciens pour certains. L'approche « *stepwise* » [1], d'introduction séquentielle de variables est la technique la plus courante dans ce contexte. Elle a été utilisée dans les modèles linéaires, la régression logistique et l'analyse discriminante.

Pour des modèles non paramétriques, peu d'outils permettent d'établir une sélection. Les arbres de décision (CART, Breiman *et al.* 1984) et les forêts aléatoires (FA, Breiman (2001)) offrent une possibilité d'établir une hiérarchie des variables explicatives très liée à la structure du modèle. Plus récemment Guyon *et al.* (2002) et Rakotomamonjy (2003) ont suggéré des scores pour chaque variable explicative utilisée dans un modèle de type machine à vecteurs supports (SVM), permettant ainsi d'établir une hiérarchie des variables.

Une fois une hiérarchie des variables obtenue, il est nécessaire de choisir celles à garder dans le modèle. En se basant sur un score calculé à partir des SVM, Guyon *et al.* (2002) ont proposé un algorithme d'élimination récursive des variables, nommé SVM-RFE. Ben Ishak *et al.* (2005) ont mis au point une

1. Nous avons testé les procédures les plus récentes de cette catégorie, les SFFS (Stepwise Forward Floating Selection, *cf.* Somol *et al.* (1999)), mais nous les avons abandonnées en raison de leur forte dépendance de l'ordre des variables dans les données d'origine.

procédure du type stepwise, plus fine que la précédente, et se basant sur différents scores estimés par bootstrap.

Récemment Park *et al.* (2006) ont abordé la sélection de variables par une approche originale consistant à introduire une pénalité dans le critère d'optimisation utilisé dans la méthode d'estimation des paramètres d'un modèle linéaire. C'est le principe de base de la technique « LARS » (Least Angle Regression [2], Efron *et al.* (2004)) en régression pour le critère des moindres carrées, mais aussi de son équivalent pour les modèles linéaires généralisés pénalisant le critère de vraisemblance, technique dite GLMpath.

Ici nous comparons trois approches différentes (SVM, FA, GLMpath) pour parvenir au même objectif : évaluer la capacité de chaque méthode à établir une « bonne » hiérarchie pour les variables explicatives et en sélectionner les essentielles pour le modèle. Les comparaisons seront effectuées d'abord sur des données simulées où six variables sont pertinentes et les autres assimilées à du bruit, puis sur des données réelles de biopuces issues de la base de données publique d'apprentissage. Nous nous focalisons sur le cas où le nombre de variables est beaucoup plus élevé que le nombre d'observations.

Les premières sections introduisent les approches utilisées, la cinquième présente les données et les résultats des comparaisons.

2. Sélection de variables basée sur les SVM

Les SVM (cf. Boser *et al.* (1992), Vapnik (1995)) 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 travail nous nous limitons au cadre de la classification linéaire binaire.

2.1. Les SVM en classification

Pour le problème de la classification binaire, on suppose que l'on dispose d'un échantillon $\mathcal{E}$ de n observations *i.i.d.* de $\mathcal{X} \subseteq \mathbb{R}^p$ appartenant chacune à l'une des deux classes de $\mathcal{Y} = \{-1, +1\}$:

$$\mathcal{E} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \subseteq (\mathcal{X} \times \mathcal{Y})^n .$$

Nous nous plaçons dans le cas où les deux classes sont linéairement séparables, *i.e.* qu'il existe un hyperplan de $\mathcal{X}$, qui permet de séparer parfaitement les deux classes. L'hyperplan est défini par le couple $(w, b) \in \mathbb{R}^p \times \mathbb{R}$ où $\langle w.x \rangle + b = 0$, où $\langle . \rangle$ représente le produit scalaire usuel et $\|.\|$ la norme associée dans $\mathbb{R}^p$.

Les SVM consistent à rechercher l'hyperplan de $\mathcal{X}$ qui maximise la marge $\gamma = \frac{1}{\|w\|}$ et qui satisfait le problème d'optimisation suivant :

2. La lettre «S» qui apparaît dans l'abréviation fait allusion aux méthodes «Lasso» et «Stagewise» qui sont des cas particuliers de «LARS».

$$\begin{array}{ll} Minimiser_{w,b} & \dfrac{\|w\|^2}{2} \\ \textit{Sous les contraintes} & y_i(\langle w.x_i\rangle + b) \geqslant 1,\ \forall i \in \{1,\ldots,n\} \end{array} \quad (1)$$

Ce problème quadratique convexe se résout dans l'espace dual en introduisant les multiplicateurs de Lagrange $\alpha_i \geqslant 0$, $i \in \{1,\ldots,n\}$. La solution optimale $\alpha^* = (\alpha_1^*,\ldots,\alpha_n^*)$ permet de déterminer les paramètres (w^*, b^*) de l'hyperplan optimal :

$$\begin{aligned} w^* &= \sum_{i=1}^{n} \alpha_i^* y_i x_i = \sum_{i\in sv} \alpha_i^* y_i x_i \\ b^* &= -\frac{\max\limits_{y_i=-1} (\langle w^*.x_i\rangle) + \min\limits_{y_i=+1} (\langle w^*.x_i\rangle)}{2} \end{aligned}$$

où $sv = \{i \in \{1,\ldots,n\}\,;\alpha_i^* \neq 0\}$. Les vecteurs x_i pour lesquels $\alpha_i^* \neq 0$ sont dits *support vectors* [3] et correspondent aux observations les plus proches de l'hyperplan séparateur.

La règle de classification f associée à un modèle SVM pour une observation $x \in \mathcal{X}$ s'écrit :

$$f(x) = sign\left(\sum_{i\in sv} \alpha_i^* y_i \langle x_i.x\rangle + b^*\right).$$

Dans le cas non linéairement séparable, les contraintes du problème (1) peuvent être relâchées en introduisant des variables d'écarts, ou en plongeant les données, par des transformations non-linéaires, dans des espaces de dimension plus élevée où un séparateur linéaire peut être trouvé (cf. Cristianini *et al.* (2000)).

2.2. Bornes sur l'erreur de généralisation pour les SVM

Les performances prédictives d'un modèle sont souvent évaluées par des bornes de risque majorant ses taux d'erreurs. L'estimateur du taux d'erreur utilisé pour les bornes de risque des SVM est celui obtenu par *leave-one-out* [4], car il est sans biais (Luntz et Brailovsky, 1969).

Les plus connues de ces bornes sont basées sur la *marge* de l'hyperplan séparateur et sur une statistique appelée l'*étendue* [5].

- Vapnik (1998) a fourni la borne supérieure suivante :

$$\mathcal{L} \leqslant \frac{R^2}{\gamma^2} = R^2 \|w^*\|^2, \quad (2)$$

où $\mathcal{L}$ est le nombre d'observations mal classées par leave-one-out, γ est la marge et R le rayon de la plus petite boule recouvrant $\mathcal{E}$.

3. Les observations pour lesquelles $\alpha_i^* = 0$ peuvent être retirées de l'échantillon sans affecter la solution.

4. Validation croisée où les échantillons tests sont constitués d'une seule observation à la fois.

5. *Span bound* en anglais.

- Une borne similaire mais plus stricte a été fournie par Vapnik et Chapelle (2000),

$$\mathcal{L} \leqslant \sum_{i \in sv} \alpha_i^* S_i^2, \tag{3}$$

où l'étendue S_i est la distance entre le vecteur support x_i et un ensemble de combinaisons linéaires contraintes des autres vecteurs supports.

2.3. Scores pour le calcul du rang d'importance des variables

Les bornes de risque introduites dans le paragraphe précédent ont servi à définir trois critères permettant d'établir une hiérarchie sur les variables explicatives. Ces critères ont été introduits par Guyon *et al.* (2002) et Rakotomamonjy (2003). Nous noterons ces critères comme suit :

$$W = \|w^*\|^2, RW = R^2 \|w^*\|^2 \text{ et } Spb = \sum_{i=1}^{n} \alpha_i^* S_i^2$$

L'idée principale est d'évaluer la contribution de chacune des variables explicatives à chacun de ces critères. Une variable est d'autant plus importante que sa contribution au critère est forte. Cette contribution est mesurée de trois manières différentes, donnant ainsi lieu à trois scores :

– Le score d'*ordre zéro* d'une variable est égal à la valeur du critère calculée en omettant la variable en question. Cette méthode de calcul mesure l'ampleur du critère en l'absence de chaque variable.
– Le score *par différences* d'une variable est égal à la différence entre la valeur du critère calculée avec toutes les variables et sa valeur calculée en omettant la variable en question. Cette méthode de calcul mesure la variation d'un critère suite à l'élimination de chaque variable.
– Le score d'*ordre un* est issu de la dérivée du critère par rapport à un vecteur de pondérations artificielles des variables. Ce score quantifie la sensibilité du critère vis à vis d'une légère variation de chaque variable.

Pour plus de détails sur les différentes méthodes adoptées dans le calcul de ces scores voir Ben Ishak (2007).

Les scores d'ordre zéro et par différences peuvent être calculés de deux manières différentes : avec ou sans réapprentissage. Ben Ishak et Ghattas (2005) ont analysé les deux approches et ont montré que les résultats avec ou sans réapprentissage ne sont pas les mêmes pour tous les critères. De plus, ils ont démontré des équivalences entre certains scores.

Les mêmes auteurs ont mis en évidence l'instabilité des scores dérivés des SVM suite à une légère perturbation des données et ont proposé de les estimer par bootstrap afin d'améliorer leurs performances.

Dans ce travail, le score d'importance d'une variable sera égal à la moyenne de ses valeurs calculées sur $B = 200$ échantillons bootstrap tirés à partir de l'échantillon d'apprentissage. Les hiérarchies seront par la suite établies sur la

base des scores moyens. Des expériences de simulations menées préalablement ont montré qu'au-delà de 200 échantillons bootstrap les hiérarchies restent relativement stables.

2.4. Sélection de modèle

Nous utiliserons ici deux procédures pour la sélection des variables basées sur les hiérarchies établies avec les scores définis dans le paragraphe précédent.

– La première utilise l'algorithme SVM-RFE (Rakotomamonjy, 2003) décrit dans le tableau 1.

TABLEAU 1. — SVM-RFE : Elimination récursive des variables.

Tant qu'il reste des variables :
  Ordonner les variables selon le critère $||w^*||^2$ calculé par différence.
    Si le nombre de variables est supérieur à 100 :
      Eliminer la moitié des variables. (les moins importantes)
    Sinon
      Eliminer la variable la moins importante.
  Estimer le taux d'erreur du modèle SVM restreint aux variables importantes.
Retenir le modèle ayant le taux d'erreur minimal.

Cet algorithme a l'avantage d'être rapide puisqu'il élimine la moitié des variables à chaque étape. Par contre, le critère est réestimé à chaque étape. Le modèle conservé est celui pour lequel le taux d'erreur estimé par leave-one-out ou sur 100 échantillons tests stratifiés est minimum.

– La deuxième procédure de sélection de variables a été introduite par Ghattas et Oppenheim (2001) en régression et par Poggi et Tuleau (2006) dans un contexte similaire. Elle est décrite dans le tableau 2 et comporte deux étapes : d'abord une hiérarchie des variables est établie par bootstrap, puis les variables sont introduites séquentiellement dans le modèle, dans l'ordre décroissant d'importance. On obtient ainsi une suite croissante de modèles emboîtés. La performance de chaque modèle de la suite est évaluée par plusieurs partages aléatoires stratifiés des données. Celui réalisant le taux d'erreur minimum est retenu comme étant celui ayant le nombre optimal de variables.

Ceci peut être réalisé pour toute hiérarchie disponible et nous conservons pour les comparaisons faites ultérieurement les trois scores à l'ordre un basés sur le poids (∂W), le rayon (∂RW) et l'étendue (∂Spb). En plus de leur fiabilité, ces scores ont l'avantage majeur d'être faciles à calculer. Pour des données disposant d'un grand nombre de variables cette procédure peut être

TABLEAU 2. — Procédure de sélection de variables à partir d'une hiérarchie. À la sortie de la procédure on récupère le nombre optimal de variables.

D = données disponibles. $B = 200$ Nombre d'échantillons bootstrap.

Calcul des scores moyens sur (D, B) pour obtenir une hiérarchie $X^{(1)}, \ldots, X^{(p)}$.

Pour $k = 1, \ldots, p$

Pour $l = 1, \ldots, 50$

Réaliser un partage aléatoire stratifié de $D = A_l \cup T_l$

A_l est l'échantillon d'apprentissage et T_l est l'échantillon test.

$M_l^k = f\left(X^{(1)}, \ldots, X^{(k)}, A_l\right)$ (construire un modèle M_l^k à partir de A_l en n'utilisant que les k variables les plus importantes).

$Er_l^k = Test\left(M_l^k, T_l\right)$ (taux d'erreur du modèle M_l^k estimé sur T_l).

$Er^k = \frac{1}{50}\sum_{l=1}^{50} Er_l^k$ (taux d'erreur moyen des modèles M_l^k).

$kopt = Arg\min_k \left\{Er^k\right\}$. (nombre optimal de variables à retenir dans le modèle).

---

accélérée en introduisant les variables par paquets de taille croissante, avec une croissance très faible en début de procédure, et de plus en plus importante au fur et à mesure.

La deuxième procédure est plus fine que SVM-RFE, et calcule la hiérarchie des variables une seule fois avant l'introduction séquentielle et par bootstrap. De plus, Svetnik *et al.* (2004) ont montré que le fait de recalculer la hiérarchie des variables à chaque étape après élimination de sous-ensembles de telles variables, introduit un fort biais dans le calcul de leurs importances.

Nous avons programmé les différentes méthodes sous Matlab et sous R en nous appuyant sur quelques librairies existantes.

3. Forêts aléatoires

Les forêts aléatoires (FA) combinent un grand nombre K d'arbres de décisions binaires construits sur des échantillons bootstrap de l'échantillon d'apprentissage. Ces techniques d'apprentissage par agrégation de modèles sont populaires et sont utilisées dans des applications provenant de domaines très variés. Les particularités des FA sont les suivantes :

– Dans la construction des arbres, à chaque nœud un nombre faible de variables est tiré au hasard et la recherche de la meilleure règle de partage est faite sur ce sous-ensemble de variables.

– Les arbres construits sur chaque échantillon bootstrap ne sont pas optimisés, en particulier ils sont maximaux et non élagués.

– Pour chaque arbre, la partie de l'échantillon d'apprentissage non utilisée pour la construction de l'arbre, dite « out of bag sample » (OOB), sert à l'évaluation de l'importance des variables.

Notons que deux versions des FA existent : l'une dite « *Random Input* » qui utilise une seule variable pour chaque règle de décision, et l'autre dite « *Random Features* » qui utilise une combinaison linéaire des variables sélectionnées à chaque nœud, avec des coefficients tirés aussi au hasard. Les bonnes performances des FA s'expliquent par deux propriétés essentielles : la bonne performance des arbres individuels (qui ont un biais très faible mais une forte variance), et la faible corrélation entre les arbres de la forêt. La corrélation entre arbres est définie comme celle de leurs prévisions sur les échantillons OOB. Le fait qu'un faible nombre de variables soit utilisé à chaque nœud des arbres construits, permet de réduire considérablement la complexité algorithmique des FA.

3.1. Hiérarchie des variables

Les forêts aléatoires fournissent un moyen original pour le calcul d'un indice d'importance pour les variables. La procédure utilisée est décrite dans le tableau 3. L'indice d'importance d'une variable correspond à la diminution en moyenne de la performance d'un arbre de la forêt quand on perturbe aléatoirement les valeurs observées pour cette variable dans l'échantillon OOB. Cet indice peut aussi être basé sur la diminution moyenne d'un autre critère, comme par exemple le critère de Gini utilisé dans la construction des arbres.

TABLEAU 3. — Importance des variables dans les forêts aléatoires. OOB_k est constitué des observations de l'échantillon d'apprentissage qui ne sont pas utilisées dans l'arbre k de la forêt.

Initialiser $N_i = 0$, $M_i = 0$ et $M_i^j = 0$, pour $i = 1, n$ et $j = 1, p$
N_i = Nombre de fois où l'observation i apparaît dans un échantillon OOB.
M_i = Nombre de fois où l'observation i apparaît dans un échantillon OOB, et est mal classée
M_i^j = Nombre de fois où l'observation i apparaît dans un échantillon OOB, et est mal classée après permutation des valeurs de la variable j dans OOB.
Pour chaque variable $j = 1, p$
  Pour chaque arbre de la forêt $k = 1, K$
    Si l'observation i est dans OOB_k, $N_i = N_i + 1$
    Si l'observation i est dans OOB_k et est mal classée, $M_i = M_i + 1$
    Permuter aléatoirement les valeurs de la variable j dans OOB_k
    Si l'observation i est dans OOB_k et est mal classée après permutation, $M_i^j = M_i^j + 1$
L'importance de la variable j est : $\frac{1}{n}\sum_{i=1}^{n} Z_i(j)$ où $Z_i(j) = (M_i^j - M_i)/N_i$.

Les forêts aléatoires dépendent de trois paramètres : le nombre d'arbres, le nombre de variables testées à chaque nœud d'un arbre et le nombre d'observations minimal dans les feuilles des arbres. Nous avons utilisé des résultats de Díaz-Uriarte et Alvarez de Andrés (2006) et réalisé quelques simulations préalables afin de choisir un réglage optimal pour ces trois paramètres.

- Dans nos expériences 200 arbres ont été construits pour chaque forêt. Au delà le gain en performances dans nos simulations était négligeable.
- Le nombre de variables testées pour chaque nœud d'un arbre est égal à $\sqrt{p}$, où p est le nombre initial de variables. Cette valeur suggérée par Breiman (2001) en classification, a été confirmée par plusieurs travaux (Liaw et Wiener (2002), Díaz-Uriarte et Alvarez de Andrés (2006)) qui ont montré son optimalité en terme de performance des forêts sur les échantillons OOB. Une forte diminution de ce paramètre réduit les chances que des variables importantes soient sélectionnées dans les arbres individuels, et dégrade les performances des forêts.
- Le nombre d'observations minimum par feuille a été fixé à cinq. La réduction à un de cette valeur n'a pratiquement aucun effet sur l'amélioration des taux d'erreurs des forêts et augmente légèrement le temps de calcul.

D'autre part nous avons pu constater aussi que l'importance des variables dans les forêts aléatoires est :

- insensible à la nature du rééchantillonnage utilisé (échantillon bootstrap avec ou sans remise),
- stable en présence de variables explicatives corrélées,
- invariante vis à vis de la normalisation (par l'écart type des $Z_i(j)$ calculée dans le tableau 3),
- stable vis à vis de faibles perturbations des données. Il est donc inutile de la calculer par bootstrap.

3.2. Sélection de modèle

Nous avons utilisé la procédure séquentielle décrite dans le tableau 2 en partant de la hiérarchie des variables calculée sur toutes les données sans bootstrap. Notons que l'avantage des forêts aléatoires dans ce contexte est la possibilité de les utiliser en classification multiclasse mais aussi en régression, ce qui n'est pas le cas des procédures basées sur les SVM. L'inconvénient majeur est le temps de calcul important, essentiellement quand on dispose de plusieurs milliers de variables explicatives. Notons que dans ce contexte Díaz-Uriarte et Alvarez de Andrés (2006) ont utilisé une procédure séquentielle descendante où les variables les moins importantes sont éliminées successivement, et le modèle optimal retenu est celui qui minimise l'erreur estimée sur les échantillons OOB. Les auteurs ont signalé que leur procédure a un double défaut : elle a tendance à sélectionner très peu de gènes, et les erreurs calculées sur les échantillons OOB sont sous estimées.

4. Modèles linéaires généralisés sous contraintes L_1

Les modèles linéaires généralisés (GLM), très largement utilisés depuis leur introduction en statistique (McCullagh et Nelder (1989)), sont définis par :

$$g(\mu) = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p$$

où $\mu = E(Y) = P[Y = 1]$ est l'espérance de la variable $Y \in \{0, 1\}$, et g est une fonction dite de lien. Le cas le plus connu que nous utiliserons ici est celui qui correspond à $g(\mu) = \dfrac{\mu}{1-\mu}$, *i.e.* au modèle logistique. L'estimation des paramètres est obtenue par maximum de vraisemblance.

4.1. Régularisation pour le choix du modèle

Park et Hastie (2006) ont suggéré l'estimation des paramètres β_k du modèle, sous contrainte de type L_1, en pénalisant la vraisemblance :

$$\widehat{\beta}(\lambda) = argmin_\beta \left\{ -logL(\mathbf{x}; \beta) + \lambda \left\| \beta \right\|_1 \right\}$$

où $\lambda > 0$ est un paramètre de régularisation, L la vraisemblance, et $\beta = (\beta_0, ..., \beta_p)$ le vecteur de paramètres à estimer. La suite $\widehat{\beta}(\lambda), 0 < \lambda < \infty$, est appelée le *path*.

Pour une valeur infinie de λ tous les coefficients sont nuls. L'augmentation de la valeur de λ contraint plus de coefficients à devenir négligeables, voire nuls. Un algorithme dit predictor-corrector est utilisé pour estimer la suite $\widehat{\beta}(\lambda)$ pour différentes valeurs de λ. Cette estimation se fait en trois étapes. Park et Hastie (2006) ont démontré que les valeurs $\widehat{\beta}(\lambda)$ sont constantes par morceaux en λ et il suffit donc de repérer les seuils de changement pour λ. Ils ont procédé par itération de quatre étapes :

À chaque étape k, on dispose d'une valeur pour λ, notée λ_k, et des valeurs β_k associées.

1 Calcul du pas nécessaire pour atteindre λ_{k+1}.

2 étape « predictor » : Calcul d'une approximation linéaire β^{k+}, de β^{k+1}.

3 étape « corrector » : Calcul par optimisation convexe de β^{k+1}, utilisant comme valeur initiale β^{k+}.

4 Tester si l'ensemble des variables actives (de coefficient non nul) doit être modifié.

À chaque itération l'ensemble des variables actives est modifié et on dispose d'une valeur du paramètre de régularisation λ_k et du modèle qui lui est associé (basé sur l'ensemble des variables actives correspondantes). Le choix du meilleur modèle, donc de la valeur optimale de λ, peut être obtenu par validation croisée, en optimisant soit le taux d'erreur de prévision, soit la vraisemblance. En fin de parcours on ne peut retrouver plus de variables que d'observations dans le modèle. Cette technique est mise en œuvre dans la librairie *glmpath* du logiciel libre *R*.

4.2. Hiérarchie des variables

Contrairement aux approches décrites dans les sections précédentes, GLMpath ne propose pas un moyen direct pour calculer un score pour chacune des variables. Parmi les p variables disponibles, seules $p' \leqslant n$ sont conservées dans le modèle, et les autres ont donc un coefficient nul. L'idée est de classer les variables actives selon la valeur de leur coefficient dans le modèle. Nous avons pu constater que l'ensemble des variables actives change considérablement si nous perturbons les données.

Ainsi, pour établir une hiérarchie stable des variables nous avons choisi d'utiliser B échantillons bootstrap de $\mathcal{E}$. Sur chaque échantillon un modèle optimal est recherché, et la valeur des coefficients pour toutes les variables est conservée. L'ensemble des variables actives varie selon les échantillons bootstrap. Nous avons calculé pour chaque variable j la valeur moyenne de son coefficient obtenue à partir des B échantillons bootstrap, notée $\hat{\beta}_j^B$. Les variables sont ensuite ordonnées selon la valeur absolue de $\hat{\beta}_j^B$. Le nombre de variables ayant un coefficient nul (estimé par bootstrap) décroît avec le nombre B d'échantillons bootstrap utilisés.

5. Comparaison des méthodes

Pour comparer les méthodes décrites ci-dessus nous avons utilisé des simulations et des données réelles de Biopuces. Pour les simulations, l'objectif est de montrer la capacité de chacune des trois méthodes à retrouver d'une part le bon ordre des variables, et d'autre part le « bon modèle », au sens du bon nombre de variables à conserver. L'effet de la taille de l'échantillon et du nombre de variables est analysé. Pour les données réelles nous n'avons pu que nous limiter à comparer les hiérarchies et les performances des trois méthodes.

Nous avons retenu à titre de comparaison le critère de discrimination de Fisher comme un score d'importance supplémentaire. Ce critère se calcule par :

$$FDS(i) = \left| \frac{\mu_i^+ - \mu_i^-}{\eta_i^+ + \eta_i^-} \right| \;\; ; \;\; i = 1, 2, \ldots p,$$

où $\mu_i^{\pm}$ est la valeur moyenne de la $i^{ème}$ variable respectivement dans la classe positive ($Y = 1$) et négative ($Y = -1$) et $\eta_i^{\pm}$ désigne l'écart type correspondant. Les expériences de simulations que nous avons menées montrent que le score FDS et celui utilisé par Dudoit *et al.* (2002) donnent lieu à des hiérarchies très similaires. Nous allons donc nous limiter au score FDS dans nos applications. La variable la plus importante selon ce critère est celle qui en maximise la valeur. L'intérêt de ce critère est qu'il n'est basé sur aucun modèle.

5.1. Résultats pour les données simulées : Toys data

Ces données ont été introduites par Weston *et al.* (2003). Dans le cas de classification binaire avec des données linéairement séparables, six variables déterminent entièrement le modèle, les autres peuvent être assimilées à du bruit. Les deux classes $\{-1, 1\}$ sont équiprobables.

- Pour 70% des observations, les trois premières variables suivent une loi gaussienne dépendant du signe de y, $x_i \sim yN(i, 1), i = 1, 2, 3$ et les trois suivantes $x_i \sim yN(0, 1), i = 4, 5, 6$.
- Pour les 30% restantes, $x_i \sim yN(0, 1)$ pour les trois premières $i = 1, 2, 3$, et $x_i \sim yN(i - 3, 1)$, pour $i = 4, 5, 6$.
- Les autres variables constituent du bruit, $x_i \sim N(0, 20), i = 7 \ldots, p$.

Ces données sont linéairement séparables avec une forte probabilité, qui est d'autant plus grande que l'échantillon est de faible taille. Les données sont normalisées en centrant et réduisant toutes les variables.

Dans un premier temps, nous vérifions la capacité des différentes méthodes à retrouver les variables importantes en présence de bruit en modifiant la taille de l'échantillon et le nombre de variables. Dans un deuxième temps, nous évaluons la capacité de chacune de ces méthodes à repérer un sous-ensemble optimal de variables.

5.1.1. Hiérarchie des variables

Nous fixons le nombre de variables à $p = 200$ et nous fixons successivement la taille de l'échantillon n à 50, 100 et 200. Les hiérarchies obtenues par les quatre premiers scores (FDS, ∂W, ∂RW, ∂Spb) sont calculées sur la base de 200 échantillons bootstrap. Pour GLMpath nous avons utilisé $B = 500$ échantillons bootstrap [6] pour garantir la stabilité des estimations des coefficients.

Les résultats sont présentés dans le tableau 4. Pour chaque taille d'échantillon utilisée, nous donnons les rangs auxquels sont apparues dans la hiérarchie quatre puis cinq puis les six variables importantes. Nous remarquons clairement que les rangs des variables importantes s'améliorent en augmentant la taille de l'échantillon. Cette caractéristique semble moins vraie pour la hiérarchie obtenue par les forêts aléatoires.

Dans le tableau 5 nous fixons la taille de l'échantillon à $n = 50$ et le nombre de variables p à 500 puis 1000. En augmentant le nombre de variables, aucune méthode n'arrive à bien classer plus de quatre variables parmi les six importantes. Deux variables parmi les six importantes apparaissent d'autant plus tard dans la hiérarchie que le nombre de variables est plus élevé. Dans ce cas le modèle linéaire généralisé arrive à retrouver les variables importantes plus facilement que les autres techniques.

6. Nous avons constaté qu'au delà de 500 échantillons bootstrap, l'ensemble des variables ayant un coefficient $\hat{\beta}_j^B$ non nul est stable.

TABLEAU 4. — Pour 50, 100 et 200 observations chaque ligne donne le rang auquel quatre, cinq puis six variables parmi les variables importantes sont apparues dans la hiérarchie. Le nombre de variables est fixé à 200. La hiérarchie est établie sur 200 échantillons bootstrap pour les quatre premiers scores et sur 500 échantillons bootstrap pour GLMpath.

$n/Score$	FDS	∂W	∂RW	∂Spb	FA	$GLMpath$
50	4	4	4	4	4	4
	6	5	5	5	6	5
	13	17	16	12	12	8
100	4	4	4	4	4	4
	5	5	5	5	5	5
	6	7	6	6	6	6
200	4	4	4	4	4	4
	5	5	5	5	5	5
	6	6	6	6	6	6

TABLEAU 5. — Pour 500 et 1000 variables, chaque ligne donne le rang auquel quatre, cinq puis six variables importantes sont apparues dans la hiérarchie. La taille de l'échantillon est fixée à 50.

$p/Score$	FDS	∂W	∂RW	∂Spb	FA	$GLMpath$
500	4	4	4	4	4	4
	5	7	7	5	12	5
	18	13	12	11	42	6
1000	4	4	4	4	4	4
	34	33	32	31	20	35
	173	194	202	224	206	38

5.1.2. Sélection du modèle optimal

Nous évaluons ici la capacité de chaque méthode à trouver le modèle optimal. La figure 1 montre l'évolution du taux d'erreur moyen pour les différents scores utilisés. Les variables sont introduites séquentiellement, une par une, dans le modèle. Toutes ces courbes ont la même allure, elles décroissent pour atteindre un certain minimum global à partir duquel elles croissent. Chaque point de ces courbes indique le taux d'erreur moyen (en ordonnée) calculé sur les 50 échantillons tests pour le modèle utilisant les k variables (en abscisse) les plus importantes.

Les trois premières colonnes du tableau 6 donnent le taux d'erreur moyen minimal ainsi que le nombre de variables qui le réalise pour les différentes tailles utilisées. Il est clair que le taux d'erreur moyen diminue lorsque la taille

de l'échantillon augmente. Le taux d'erreur des forêts aléatoires est souvent supérieur à celui obtenu par chacune des autres méthodes. Pour les autres méthodes l'examen des résultats ne nous permet pas de les hiérarchiser.

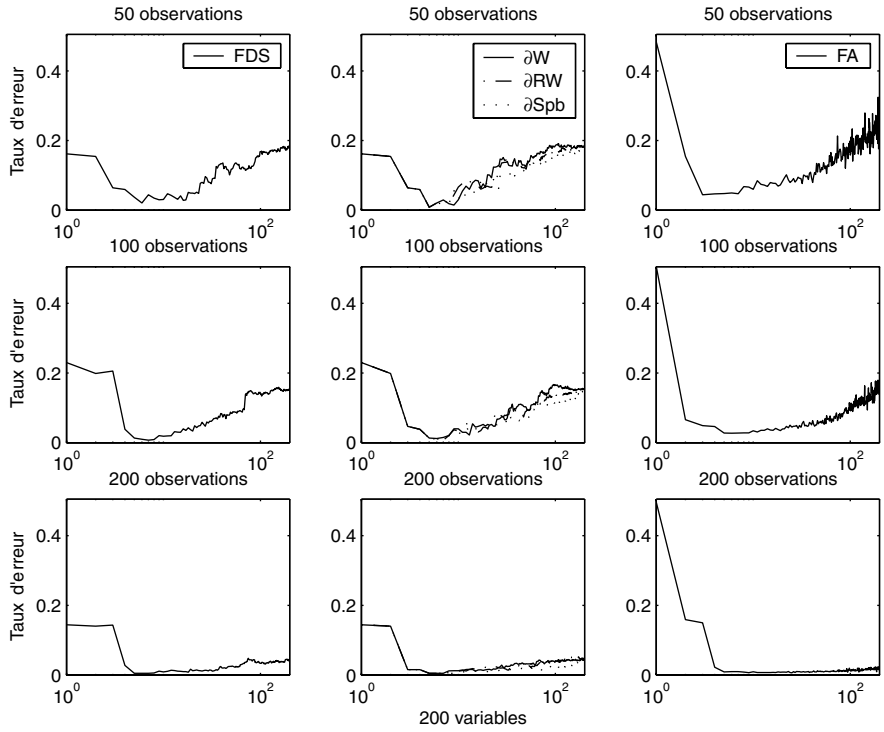

FIG 1. — Effet de la taille de l'échantillon. Taux d'erreur moyen calculé sur 50 échantillons tests pour différentes tailles. Le nombre de variables est fixé à 200.

TABLEAU 6. — Taux d'erreur moyen calculé sur 50 échantillon tests obtenu suite à l'introduction séquentielle des variables selon l'ordre d'importance décroissant. Le nombre optimal de variables est mis entre parenthèses. Pour la méthode GLMpath le taux d'erreur est obtenu par validation croisée sur l'échantillon d'apprentissage.

$Score/(n,p)$	(50,200)	(100,200)	(200,200)	(50,500)	(50,1000)
FDS	0.0208(6)	0.0072(7)	0.0048(7)	0.0044(5)	0.0084(5)
∂W	0.0084(5)	0.012(6)	0.0048(7)	0.008(7)	0.0084(5)
∂RW	0.0084(5)	0.0072(7)	0.0048(7)	0.008(7)	0.0076(6)
∂Spb	0.0084(5)	0.0096(6)	0.0044(8)	0.0044(5)	0.0084(5)
$SVM\text{-}RFE$	0.0476(8)	0.016(8)	0.006(4)	0.0132(8)	0.0104(4)
$GLMpath$	0.0188(1)	0.0252(3)	0.0074(4)	0.008(4)	0.0192(2)
FA	0.044(3)	0.0272(6)	0.0064(25)	0.0252(12)	0.0656(4)

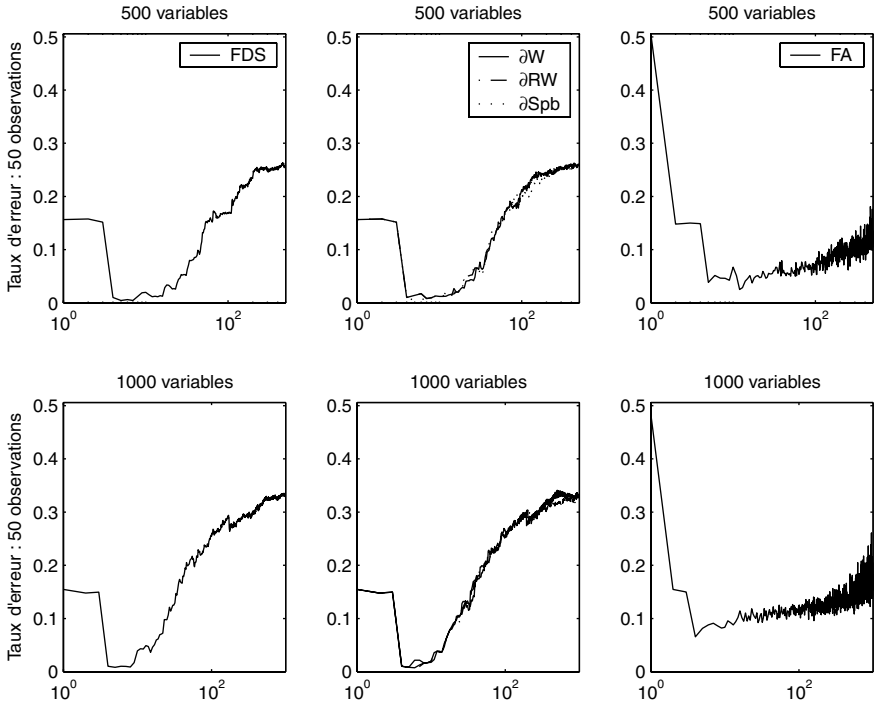

FIG 2. — Effet du nombre de variables. Taux d'erreur moyen calculé sur 50 échantillons tests en utilisant 500 variables (les panneaux de la première ligne) et 1000 variables (les panneaux de la deuxième ligne). La taille de l'échantillon est fixée à 50.

La figure 2 donne une idée sur l'allure globale du taux d'erreur moyen en présence d'un grand nombre de variables constituant du bruit. Nous retrouvons la même forme de courbe que précédemment sauf que la phase de croissance est beaucoup plus importante surtout pour les quatre premiers scores.

Les deux dernières colonnes du tableau 6 contiennent les résultats pour une taille de l'échantillon égale à 50 et un nombre de variables valant 500 puis 1000. Nous remarquons que les taux d'erreur sont légèrement différents de ceux obtenus pour 200 variables (voir les trois premières colonnes du même tableau). Ceci est dû au fait qu'avec 500 et 1000 variables au moins quatre variables parmi les six importantes ont été retenues dans l'ensemble des méthodes. Les forêts aléatoires réalisent un taux d'erreur souvent supérieur à celui des autres scores.

5.2. Résultats pour les données de Biopuces

Nous avons utilisé des données de biopuces très répandues dans la littérature scientifique. Elles relèvent toutes d'un problème de discrimination binaire, et

disposent d'un grand nombre de variables, les gènes, et de peu d'observations. Notre objectif ici n'est pas d'interpréter les résultats obtenus quant aux gènes sélectionnés, mais seulement de comparer les hiérarchies et les performances des méthodes.

5.2.1. Descriptif des données

Nous donnons une brève description des données que nous avons utilisées, leur origine, les pré-traitements qu'elles ont subis par les fournisseurs et le problème qu'elles soulèvent. Pour une description plus détaillée des traitements statistiques préalables appliqués souvent sur des données de biopuces, voir Dudoit *et al.* (2002).

- **Colon** : Ce jeu de données est constitué de 62 profils d'expression issus de deux populations : 40 tissus tumoraux et 22 tissus sains. Chaque profil comporte 2000 niveaux d'expression de gènes. Ces données ont été introduites pour la première fois par Alon *et al.* (1999) et elles ont subi une transformation logarithmique et des standardisations dans le travail de Guyon *et al.* (2002) dans le but de diminuer l'effet des valeurs aberrantes. Ces données ont été téléchargées à partir de http ://www.kyb.tuebingen.mpg.de/bs/people/weston/l0/index.html.

- **Lymphoma** : Le problème de discrimination lié à ce jeu de données est décrit en détail dans Alizadeh (2000). Ce jeu de données est constitué de 96 profils d'expression issus de deux populations : 62 cas sont du type « DLCL », « FL » ou « CLL » (maligne) et les 34 restants sont normaux. Chaque profil comporte 4026 gènes. Ces données n'ont subi aucun type de pré-traitement et elles ont été téléchargées à partir de http ://www.kyb.tuebingen.mpg.de/bs/people/weston/l0/index.html.

- **Prostate** : Dans ce jeu de données le niveau d'expression de 12600 gènes est mesuré sur 102 tissus. L'objectif est de séparer les tissus normaux (52) des cancéreux (50). On trouvera une description complète de ces données dans Singh *et al.* (2002). Ces données n'ont subi aucun type de prétraitement et elles sont disponibles à http ://homes.esat.kuleuven.be/ npochet/Bioinformatics/.

- **Leukemia** : Ce jeu de données est constitué de 72 profils d'expression issus de deux populations : 47 tissus atteints de Leucémie lymphoblastique aiguë (ALL) et 25 tissus atteints de Leucémie myéloïde aiguë (AML). Il est à noter que ce jeu de données peut aussi être considéré comme problème de discrimination multiclasse dans la mesure où les 47 tissus ALL se subdivisent en deux populations selon que les cellules analysées sont de type B (38 cas) ou de type T (9 cas). Chaque profil comporte 7129 niveaux d'expression de gènes. L'échantillon test est de taille 34 (20 ALL/14 AML) alors que celui d'apprentissage est de taille 38 (27 ALL/11 AML). Ces données ont subi un pré-traitement de seuillage et de transformation logarithmique dans l'article d'origine de Golub *et al.* (1999). Ces données ont été téléchargées à partir de

  http ://homes.esat.kuleuven.be/ npochet/Bioinformatics/.

Avant d'appliquer notre procédure de sélection de variables nous avons commencé par normaliser tous les jeux de données en centrant et réduisant toutes les variables (gènes) dans le but d'éliminer l'effet d'échelle.

TABLEAU 7. — Description des données réelles : nombre de variables, tailles des échantillons d'apprentissage et de test (dans le cas où on en dispose), effectifs de chaque classe.

Données	*# de variables*	*apprentissage*	*test*	# d'observations *+1/-1*
Colon	2000	62	–	22/40
Lymphoma	4026	96	–	62/34
Prostate	12600	102	–	52/50
Leukemia	7129	38	34	27/11 - 20/14

5.2.2. Hiérarchie des variables

Nous menons ici les mêmes expériences sur les quatre jeux de données décrits ci-dessus. Pour Leukemia l'échantillon test fourni sera utilisé pour les comparaisons des méthodes sur ce jeu de données.

Pour comparer les méthodes nous nous basons sur le nombre de variables communes à leurs hiérarchies. Nous réalisons ces comparaisons uniquement pour les variables dont les coefficients $\hat{\beta}_j^B$ (estimés avec $B = 500$ échantillons bootstrap) dans les modèles GLMpath sont différents de zéro. Le nombre de variables ainsi intervenant dans les comparaisons des hiérarchies dépend du jeu de données : 999 pour Colon, 1376 pour Lymphoma, 1190 pour Leukemia, et 2234 pour Prostate.

Les figures 3 et 4 montrent pour les quatre jeux de données des courbes de similarités entre les différentes hiérarchies comparées. L'axe des abscisses correspond au rang dans la hiérarchie et celui des ordonnées indique le nombre de variables communes aux hiérarchies jusqu'à ce rang. Nous avons divisé les abscisses et les ordonnées par le nombre total de variables afin de se situer dans le carré $[0, 1] \times [0, 1]$. Nous avons effectué quatre comparaisons : les critères SVM entre eux, les critères SVM avec les forêts aléatoires, les SVM avec GLMpath, et GLMpath avec les forêts aléatoires. Les courbes correspondant à ces quatre comparaisons sont superposées. Plus la courbe est proche de la bissectrice, plus les hiérarchies sont voisines. La première partie de la courbe est particulièrement pertinente pour ces comparaisons, elle concerne la comparaison des variables les plus importantes (en tête de chaque hiérarchie). Pour les quatre jeux de données, nous remarquons que les hiérarchies basées sur les SVM sont très proches les unes des autres. Ensuite, celles fournies par les SVM et GLMpath semblent être aussi voisines. En effet, comme les quatre jeux de données sont linéairement séparables, ces deux techniques semblent être mieux adaptées. Les forêts aléatoires donnent des hiérarchies assez différentes. Ce résultat sera retrouvé par la suite dans la comparaison des performances des modèles obtenus à partir de ces hiérarchies. Ces résultats sont conformes à ceux que nous avons obtenus sur les simulations.

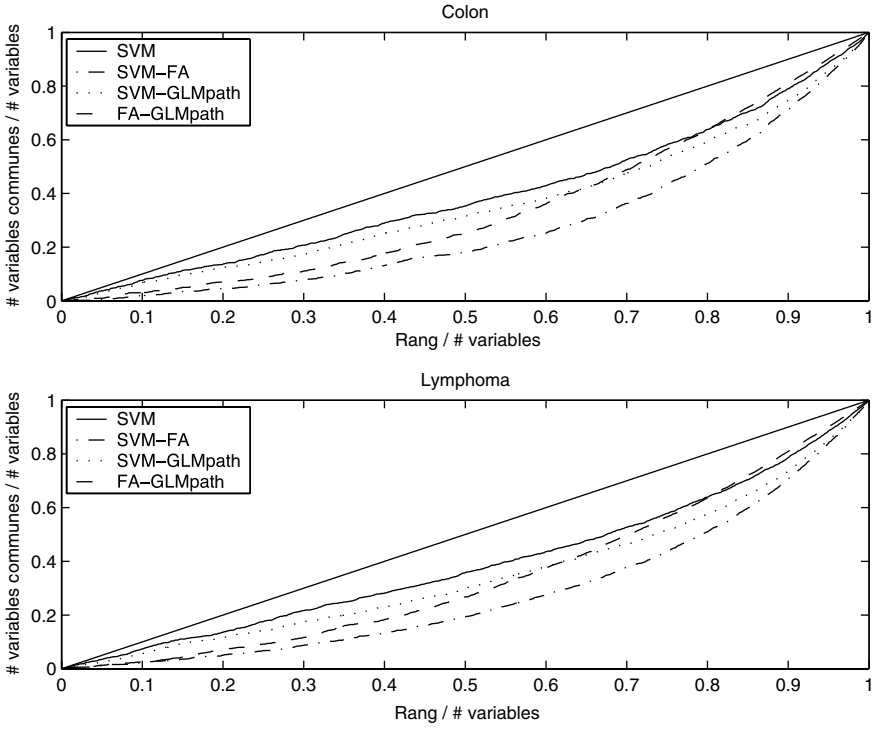

FIG 3. — Courbes de similarités entre les hiérarchies SVM, SVM-FA, SVM-GLMpath et FA-GLMpath pour les deux jeux de données Colon et Lymphoma. L'axe des abscisses indique le rang (divisé par le nombre total de variables) et celui des ordonnées donne le nombre de variables communes (divisé par le nombre total de variables) pour les hiérarchies comparées. Plus la courbe est proche de la première bissectrice plus les hiérarchies correspondantes sont voisines.

Le tableau 8 donne le nombre de variables communes pour les différentes comparaisons illustrées dans les graphiques précédents pour les 50 variables importantes. Nous remarquons que le nombre de variables communes est en général supérieur à 50% pour les hiérarchies données par les SVM et celles des SVM et de GLMpath. Ce taux est beaucoup plus élevé pour les données colon et Lymphoma, qui ont en l'occurrence un nombre de variables plus faible que les deux autres jeux de données.

5.2.3. Sélection de modèle

Pour les quatre premiers scores utilisés nous avons appliqué la procédure décrite dans le tableau 2. L'introduction séquentielle des variables a été réalisée par paquets de taille croissante. La taille de ces paquets a été choisie telle que leur nombre soit constant pour tous les jeux de données (environ 700) et que presque la moitié d'entre eux, ceux du début, ne contiennent qu'une seule variable.

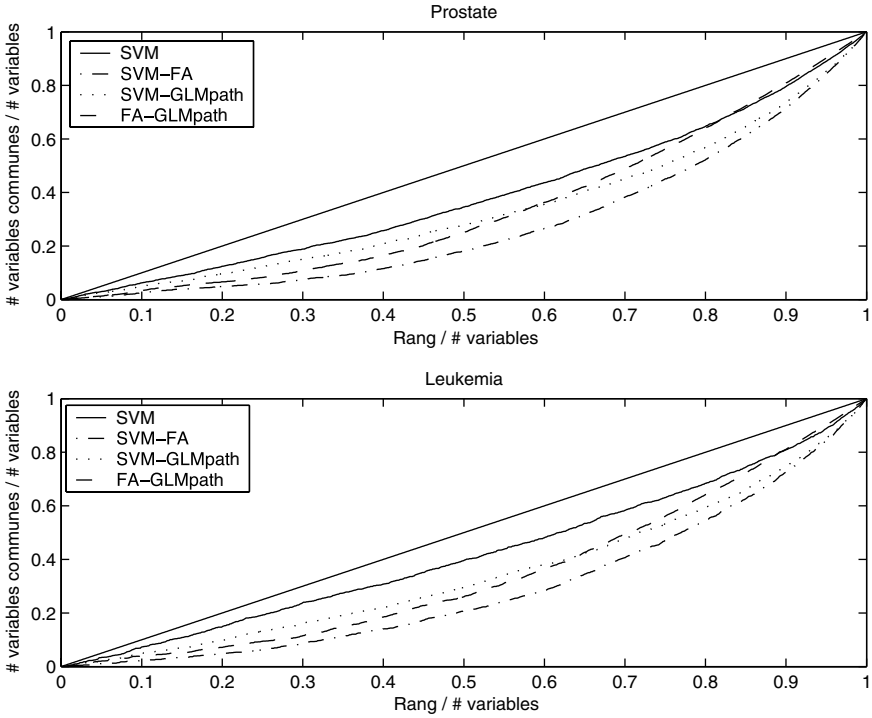

FIG 4. — Courbes de similarités entre les hiérarchies SVM, SVM-FA, SVM-GLMpath et FA-GLMpath pour les deux jeux de données Leukemia et Prostate.

TABLEAU 8. — Nombre de variables communes parmi les 50 les plus importantes pour les quatre comparaisons établies.

Comparaison/Jeu de données	Colon	Lymphoma	Prostate	Leukemia
SVM	37	37	32	30
SVM/GLMpath	33	26	24	21
SVM/FA	4	9	12	9
FA/GLMpath	10	12	16	21

Le tableau 9 présente les résultats des applications menées sur les quatre jeux de données réelles. On donne entre parenthèses, pour chaque méthode, le nombre optimal de variables pour lequel le taux d'erreur est minimal. Le taux d'erreur est estimé par 50 partages aléatoires stratifiés sauf pour Leukemia dont l'échantillon test est fourni à part. Nous remarquons que le nombre optimal de variables varie d'une méthode à l'autre sur un même jeu de données. De plus, les performances de ces méthodes diffèrent d'un jeu de données à un autre mais sur l'ensemble des résultats nous relevons une légère

supériorité de GLMpath et de notre procédure (décrite dans le tableau 2) utilisant les scores SVM.

TABLEAU 9. — Résultats des applications sur les données biopuces. On donne entre parenthèses le nombre minimal de variables pour lequel le taux d'erreur moyen atteint son minimum. Ce taux d'erreur est calculé sur 50 échantillons tests obtenus par partages aléatoires stratifiés. On garde le même partage pour les différentes méthodes utilisées. Pour le jeu de données Leukemia le taux d'erreur est estimé sur l'échantillon test.

Score/Données	Colon	Lympoma	Prostate	Leukemia
FDS	0.1219(3)	0.0436(200)	0.0371(315)	0.0882(7)
∂W	0.0009(31)	0(186)	0.0269(83)	0.1176(2)
∂RW	0.0029(33)	0(60)	0.0269(902)	0.0882(22)
∂Spb	0.0029(34)	0.0006(118)	0.0109(45)	0.1176(11)
$SVM-RFE$	0.0057(32)	0(64)	0(64)	0.0882(1)
$GLMpath$	0.064(2)	0(3)	0(3)	0(1)
FA	0.0962(55)	0.0588(73)	0.0554(7)	0.0588(103)

Notons que les taux d'erreur moyens réalisés sur ces jeux de données en utilisant toutes les variables avec les SVM linéaires sont : Colon : 0.17, Leukemia : 0.20588, Lymphoma : 0.06, Prostate : 0.075.

5.2.4. Biais de sélection

Nous considérons que les résultats obtenus dans le paragraphe précédent sont optimistes et présentent donc un biais de sélection. Ceci est dû principalement au fait que la hiérarchie des variables a été calculée à partir de toutes les données (*cf.* Ambroise et MacLachlan (2002), Reunanen (2003)). L'idée est donc de réaliser une validation croisée de la procédure décrite dans le tableau 2. Les données disponibles sont partitionnées en $V = 10$ parts égales par stratification. Chaque partie joue le rôle d'échantillon test. Son complémentaire est utilisé dans la procédure initiale du tableau 2. La procédure tenant compte du biais de sélection est décrite dans le tableau 10.

Ainsi la hiérarchie des variables est calculée V fois, et V modèles optimaux avec leurs nombres de variables et leurs performances sont obtenus. Le nombre moyen de variables et le taux d'erreur minimal moyen sont présentés dans le tableau 11.

Nous confrontons ces résultats à ceux obtenus dans le tableau 9. Les performances des modèles sont systématiquement dégradées. La dégradation pour les scores basés sur les SVM est d'autant plus significative que le jeu de données comporte moins de variables. Le nombre moyen de variables sélectionnées par GLMpath est similaire à celui obtenu sans validation croisée pour les trois jeux de données. La dégradation des performances des forêts aléatoires est très faible.

Les taux d'erreurs sont plus réalistes que les résultats obtenus sans validation croisée. Cependant, les gènes sélectionnés et leur nombre sont différents et très variables pour chaque échantillon de validation croisée. Ces taux sont donc des moyennes de performances de modèles très différents a priori les uns des autres n'utilisant pas les mêmes sous-ensembles de variables.

TABLEAU 10. — 10-validations croisées de la procédure de sélection de variables décrite dans le tableau 2.

Soit D le jeu de données, et B le nombre d'échantillons bootstrap.

Partitionner D avec stratification, $D_1, ..., D_{10}$.

Soit $D_{-j} = D - D_j$.

Pour $j = 1, \ldots, 10$

Score(D_{-j},B) et conserver la hiérarchie $X^{(1)}, ..., X^{(p)}$

Pour $k = 1, \ldots, p$

$M^k = f\left(X^{(1)}, ..., X^{(k)}\right)$

$Er^k = Test_{RS}(M^k, D_{-j})$

$kopt_j = Argmin_k\{Er^k\}$

$er_j =$ Erreur moyenne de M^{kopt_j} sur D_j.

Calcul de $\bar{er} = \frac{1}{10}\sum_{j=1}^{10} er_j$.

TABLEAU 11. — Biais de sélection : Erreur estimée par validation croisée pour le meilleur modèle sélectionné pour les données réelles. Le nombre moyen de variables sélectionnées est entre parenthèses.

Données	Colon	Lymphoma	Prostate
FDS	0.1595(15.1)	0.1233(83.7)	0.0882(126.4)
∂W	0.233 (35.1)	0.051 (86.5)	0.054 (756.6)
∂RW	0.214 (43.3)	0.042 (71)	0.053 (573.3)
∂Spb	0.197 (31.8)	0.073 (70.5)	0.052 (95.5)
$SVM - RFE$	0.1452(26.4)	0.0878(16.8)	0.0582(43.2)
$GLMpath$	0.1809 (1.3)	0.0522 (2.8)	0.05909 (1.6)
FA	0.106 (49.8)	0.052 (65.9)	0.059 (81)

6. Conclusions et perspectives

La comparaison des méthodes de sélection de variables a montré que les résultats obtenus avec les SVM sont assez voisins quels que soient les scores utilisés. Le modèle linéaire généralisé sous contrainte L_1 sur les coefficients du modèle donne des résultats proches de ceux des SVM, et paraît même plus performant dans le cas où p est très grand. Les forêts aléatoires semblent être moins performantes pour accomplir ces tâches, mais paraissent plus stables que les autres méthodes. Les résultats obtenus sur les données réelles confirment ceux obtenus par simulations.

Notons que nous nous sommes limités ici aux situations où les données sont linéairement séparables et la variable à prédire est binaire. Ceci est le cas des données simulées utilisées mais aussi celui des données réelles (les quatre jeux de données sont parfaitement séparables par les SVM avec un noyau linéaire). Plusieurs extensions de ces travaux sont en cours : celle au cas non linéairement séparable d'une part, et celle au cas multiclasse d'autre part. Les scores basés sur les SVM nécessitent une adaptation dépendant de l'approche multiclasse utilisée. Les équivalences entre les scores dans le cas linéaire ne sont pas toujours valides dans le cas non linéaire. Par contre, pour les modèles linéaires généralisés et les forêts aléatoires, les approches que nous avons employées ici peuvent être utilisées directement dans le cas multiclasse et le cas non linéaire.

Remerciements : Les auteurs remercient les rapporteurs et l'éditeur associé pour leurs nombreuses remarques et commentaires. Ils remercient également Adele Culter pour les précisions qu'elle a fournies au sujet des forêts aléatoires, et Claude Deniau pour ses précieux commentaires sur leur travail.

Références

Alizadeh A. A. (2000). Distinct types of diffues large b-cell lymphoma identified by gene expression profiling. *Nature*, 403 : 503-511.

Alon U., N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon cancer tissues probed by oligonucleotide arrays. *Cell Biology*, 96(12) : 6745-6750.

Ambroise C. and G. Maclachlan (2002). Selection Bias in gene extraction on the basis of microarray gene expression data. *Proceedings of the National Academic Science, USA*, 99(10) :6562-6566.

Ben Ishak A. and B. Ghattas (2005). An efficient method for variable selection using svm-based criteria. *Pré-publication de l'Institut de Mathématiques de Luminy*, Marseille, France.

Ben Ishak A. (2007). Séléction de variables par les machines à vecteurs supports pour la discrimination binaire et multiclasse en grande dimension. *Thèse soutenue à l'Université de la Méditerranée le 06 Spetembre 2007.* (http ://lumimath.univ-mrs.fr/ ghattas/theseAnisBenIshak.pdf)

Boser A., I. Guyon, and V. Vapnik (1992). A training algorithm for optimal margin classifiers. *In Fifth Annual Workshop on Computational Learning Theory*, pages 144-152, Pittsburgh. ACM.

Breiman L., J. H. Friedman, R. A. Olshen, and C. J. Stone (1984). *Classification and Regression Trees*. Wadsworth and Brooks.

Breiman L. (2001). Random forests. *Machine Learning Journal*, 45 :5–32.

Cristianini N. and J. Shawe-Taylor (2000). *Introduction to Support Vector Machines*. Cambridge University Press.

Díaz-Uriarte R. and S. Alvarez de Andrés (2006). Gene Selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7 :3, pp 1-13.

Dudoit S., J. Fridlyand, and T. Speed (2002). Comparison of discrimination methods for the classification of tumors using gene expression data, *J. Amer. Stat. Assoc.*.

Efron B., T. Hastie, I. Johnstone, and R. Tibshirani (2004). Least angle regression. *Annals of Statistics*, 32(2) :407-499.

Ghattas B. et G. Oppenheim (2001). Etude de faisabilité : Modèles globaux pour la mise au point moteur. *Rapport technique Renault*, 56 pages.

Golub T. R., D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander (1999). Molecular classification of cancer : Class discovery and class prediction by gene expression monitoring. *Science*, 286 : 531-537.

Guyon I. and A. Elisseff (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3 : 1157-1182.

Guyon I., J. Weston, S. Barnhill, and V. Vapnik (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1-3) : 389-422.

Kohavi R. and G. H. John (1997). Wrappers for Feature Subset Selection. *Artificial Intelligence*, 97(1-2) : 273-324.

Liaw A. and M. Wiener (2002). Classification and Regression by Random Forest. *Rnews*, 2 :18-22.

Luntz A. and V. Brailovsky (1969). On estimation of characters obtained in statistical procedure of recognition. *Technicheskaya Kibernetica*, 3.

Mccullagh P. and J. Nelder (1989). *Generalized Linear Models*. CHAPMAN & HALL/CRC, Boca Raton.

Park M. Y. and T. Hastie (2006). L_1 Regularization Path Algorithm for Generalized Linear Models. *Technical report*, Stanford University.

Poggi J. M. et C. Tuleau (2006). Classification supervisée en grande dimension. Application à l'agrément de conduite automobile. *Revue de Statistique Appliquée*, LIV (4), 39-58.

Rakotomamonjy A. (2003). Variable selection using SVM-based criteria. *Journal of Machine Learning Research*, 3 : 1357-1370.

Reunanen J. (2003). Overfitting in Making Comparisons Between Variable Selection Methods. *Journal of Machine Learning Research*, 3 :1371-1382.

Singh D., P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'amico, J. P. Richie, E. S. Lander, M. Loda, P. W. Kantoff, T. R. Golub, and W. R. Sellers (2002). Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1(2) : 203-209.

SOMOL P., P. PUDIL, J. NOVOVIČOVÁ, and P. PACLIK (1999). Adaptive floating search methods in feature selection. *Pattern Recognition Letters*, 20 :1157-1163.

SVETNIK V., A. LIAW, C. TONG, and T. WANG (2004). Application of Breiman's random forest to modeling structure-activity relashionships of pharmaceutical molecules. *Multiple Classifier Systems. Lecture Notes in Computer Science, Springer*, 3077 :334-343.

VAPNIK V. (1995). *The Nature of Statistical Learning Theory*. Springer Verlag, New York.

VAPNIK V. (1998). *Statistical Learning Theory*. John Wiley and Sons, New York.

VAPNIK V. and O. CHAPELLE (2000). Bounds on error expectation for support vector machines. *Neural Computation*, 12 : 9.

WESTON J., A. ELISSEFF, B. SCHOELKOPF, and M. TIPPING (2003). Use of the zero norm with linear models and kernel methods. *Journal of Machine Learning Research*, 3 : 1439-1461.