Améliorer les performances d'un modèle prédictif: perspectives et réalité

Stéphane TUFFERY

6, rue Gaston Turpin - 44000 Nantes
stephane.tuffery@univ-rennes1.fr, data.mining@free.fr
http://data.mining.free.fr

Résumé. Dans cet article, nous montrons que les performances d'un modèle prédictif dépendent généralement plus de la qualité des données et du soin apporté à leur préparation et à leur sélection, que de la technique de modélisation elle-même. Entre deux techniques, l'écart de performance est souvent négligeable en regard des incertitudes résultant de la définition de la variable à expliquer et de la représentativité de l'échantillon d'étude. Toutefois, le rééchantillonnage et l'agrégation de modèles peuvent permettre de réduire drastiquement la variance et parfois même le biais de certains modèles. De bons résultats peuvent aussi être obtenus simplement par la partition de modèles, c'est-à-dire en partitionnant en classes l'échantillon initial et en construisant un modèle sur chaque classe.

1 Introduction

Le foisonnement de découvertes statistiques de ces dernières années (modèle additif généralisé, séparateurs à vaste marge, régression logistique PLS, etc.) ne doit pas nous faire oublier que dans la plupart des problèmes réels de classement[1], notamment ceux qui se posent dans l'assurance, la banque et le marketing[2], les performances d'un modèle dépendent souvent moins de la technique de modélisation que de la nature et de la qualité des données. De nombreux comparatifs basés sur un ou plusieurs jeux de données mettent en évidence des écarts de performance très limités entre les techniques. Ces écarts sont si ténus qu'ils sont parfois dérisoires en regard des incertitudes qui pèsent sur eux. Première incertitude : la définition de la variable à expliquer, qui est parfois beaucoup moins naturelle qu'elle peut l'être dans des domaines comme la médecine. Deuxième incertitude : la représentativité de l'échantillon d'étude (dont sont extraits les échantillons d'apprentissage et de test) par rapport à une population dont tous les individus n'ont pas été observés (biais de sélection) ou qui a pu évoluer depuis l'observation (la modélisation s'appuie sur des échantillons rétrospectifs). Troisième incertitude :

[1] Attention à la terminologie : les statisticiens francophones appellent "classement" (technique supervisée) ce que les anglo-saxons- et certains data miners français- appellent "classification". Quand au terme français "classification" (technique non supervisée), il se traduit en anglais par "clustering".

[2] Nous parlons de ces domaines qui sont le sujet du présent numéro de la RNTI et qui sont aussi du ressort de l'auteur ; certaines conclusions sont généralisables à d'autres domaines, mais pas toutes : ainsi la parcimonie dans le nombre de variables du modèle est moins pertinente dans certains domaines comme la génomique ou la chimiométrie.

l'influence sur les performances du choix des échantillons d'apprentissage et de test, et la variance qui en résulte, surtout quand on observe des phénomènes rares. Nous pouvons citer une quatrième source d'incertitude même si elle n'est pas étudiée dans cet article : l'influence du statisticien sur les performances, dans la mesure où il optimisera mieux une technique qu'il connaît mieux. En raison de ces incertitudes, la légère sur-performance d'un modèle complexe peut être illusoire, et la part, à la fois la plus grande et la plus robuste, de la performance est souvent obtenue aussi bien par des techniques simples aux hypothèses restrictives, telle l'analyse discriminante linéaire, que par des techniques plus sophistiquées mais aussi plus sujettes au surapprentissage. Cela est d'autant plus vrai que des modèles simples peuvent voir leurs performances augmenter, cette fois-ci de façon sensible, par un travail approprié sur les données et par le recours aux méthodes générales que sont la partition et l'agrégation de modèles.

Nous commençons l'article par les questions soulevées par la définition de la variable à expliquer (section 2) et par la représentativité de l'échantillon d'étude (section 3). Puis nous abordons plusieurs sections sur la sensibilité des performances d'un modèle : mesure de performance fournie par les courbes ROC et de lift (section 4), sensibilité au nombre de variables explicatives (section 5), sensibilité au choix des échantillons d'apprentissage et de test (section 6) et sensibilité au choix de la méthode de modélisation (section 7). La troisième partie de l'article est consacrée aux méthodes générales d'amélioration des performances : discrétisation des variables (section 8), partition de modèles (section 9) et rééchantillonnage bootstrap et agrégation de modèles (section 10).

2 Définition de la variable à expliquer

La définition de la variable à expliquer s'impose naturellement dans un problème médical comme celui de la prédiction d'un cancer : le patient a une tumeur ou n'en a pas. Mais pour un établissement financier, qu'est-ce qu'un client non risqué ? Un client qui n'a pas connu d'incident de paiement, qui n'en a connu qu'un seul, un client qui n'a pas été au contentieux ni fiché à la Banque de France, un client dont la dette a été apurée. . . ? Fréquemment, la définition de la variable à expliquer par le modèle, la variable « cible », n'est pas complètement naturelle et imposée par le contexte. Plusieurs définitions analogues pourraient être aussi valables, et pourtant il faut en choisir une, et ce choix aura une grande influence sur le modèle obtenu, plus que la méthode de modélisation. L'arbitraire de ce choix existe notamment quand la variable cible binaire est obtenue en fixant un seuil à une variable quantitative : risqué au delà de trois impayés, non risqué en deçà. Et si le seuil n'était pas trois, mais deux ou quatre ? Dans certains modèles de *credit scoring*, il arrive que, pour faciliter la discrimination, on sépare les « bons »des « mauvais », en définissant les premiers comme étant par exemple ceux qui ont au plus un impayé, les seconds comme étant ceux qui ont au moins trois impayés, et en considérant comme « indéterminés »et exclus de la modélisation ceux qui ont exactement deux impayés.

3 Représentativité de l'échantillon d'étude

3.1 Constitution de l'échantillon d'étude

Une hypothèse fondamentale dans l'élaboration d'un modèle prédictif est que l'échantillon d'étude disponible est représentatif de la population à laquelle sera appliqué le modèle, c'est-à-dire que tout individu de la population a une probabilité non nulle d'appartenir à l'échantillon. Si cet échantillon n'est pas représentatif, le modèle élaboré ne se généralisera pas bien à la population tout entière et les futures prédictions manqueront de fiabilité.
Cela ne signifie pas que l'échantillon d'étude soit nécessairement une reproduction exacte de la population tout entière, c'est-à-dire le fruit d'un échantillonnage aléatoire simple. En effet, pour faciliter l'apprentissage, en particulier d'un réseau de neurones ou d'un arbre de décision, l'échantillon d'apprentissage est parfois volontairement biaisé, de façon à contenir la même proportion de chacune des classes à prédire, même si ce n'est pas le cas de l'ensemble de la population. Si l'on veut classer les individus en deux catégories, les positifs et les négatifs, l'échantillon contiendra 50% de positifs et 50% de négatifs : l'échantillon est ainsi stratifié sur la variable à expliquer. La nécessité d'un tel redressement de l'échantillon est évidente dans le cas d'un arbre de décision CART (Breiman et al., 1984). Si l'on analyse les résultats d'une campagne de marketing direct dont le taux de retour est de 3%, on ne pourra visiblement pas construire un arbre CART sachant classer 3% d'acheteurs et 97% de non-acheteurs. En effet, puisque CART utilise un critère de division basé sur la pureté, la division sera impossible dès la racine de l'arbre : l'arbre dira que personne n'est acheteur, avec un taux d'erreur très convenable de 3%.
Le redressement d'échantillon est aussi parfois préconisé dans le cas d'une analyse discriminante linéaire réalisée avec des variables explicatives n'ayant pas la même variance dans les différentes classes à discriminer : l'hétéroscédasticité est moins gênante quand les effectifs ont même taille.

3.2 Le problème de l'évolution de la population

La première entorse à l'hypothèse de représentativité de l'échantillon d'étude vient du principe même de la plupart des techniques de modélisation (Tufféry, 2007), qui déduisent du lien entre les variables explicatives observées pendant la période $T-1$ et la variable cible observée pendant la période T, un modèle prédisant la valeur de la variable cible pendant la période $T+1$ (par exemple, les achats ou les impayés dans les 12 mois à venir) à partir de la connaissance des variables explicatives pendant la période T (par exemple, au cours des 12 derniers mois). Cette utilisation d'échantillons rétrospectifs suppose implicitement une stabilité de la population étudiée et de la distribution des variables du modèle. Or, entre $T-1$ et $T+1$, les individus ont pu changer, l'environnement économique aussi, et un modèle très performant sur des données anciennes peut l'être moins aujourd'hui. Comme le remarque justement David J. Hand (2005), l'avantage d'une méthode avancée de modélisation sur une méthode plus simple (linéaire, par exemple) réside souvent dans une meilleure modélisation des petites particularités de l'échantillon d'étude : comme ces idiosyncrasies résistent mal au passage du temps, il est inutile de chercher à les modéliser et l'apport des méthodes avancées peut alors s'avérer illusoire.

3.3 Le problème du biais de sélection

La seconde entorse à l'hypothèse de représentativité vient de ce que, dans la population à modéliser, tous les individus n'ont pas toujours pu être observés. C'est ce que l'on appelle le biais de sélection, auquel ne résistent pas non plus les petites particularités de l'échantillon d'étude. Ce problème se pose dans le domaine médical et dans celui de l'assurance et du crédit. Le biais de sélection survient notamment dans la construction des modèles d'appétence, lorsque les politiques commerciales et les ciblages marketing font que certains clients reçoivent systématiquement moins de propositions que les autres : ils souscriront donc en moyenne moins de contrats et biaiseront les modèles d'appétence. De façon encore plus radicale, ce problème de biais de sélection survient dans l'élaboration de modèles de risque, et tout particulièrement dans le credit scoring, puisque le client peut se voir interdire de souscrire un crédit, ce qui est bien plus discriminant que de ne pas s'en voir proposer.
La prise en compte en credit scoring des dossiers refusés, que l'on appelle inférence des refusés ("reject inference"), pose problème puisque l'organisme prêteur ne leur a pas laissé la possibilité d'exister et de se révéler "bons" ou "mauvais", ce qui fait que la variable cible n'est pas connue. Or, ils n'ont pas été refusés au hasard, mais sur la base de procédures internes ou de préjugés des analystes. De même que l'application à tous les clients d'un modèle d'appétence construit sur les seuls clients précédemment ciblés, l'application à cette population d'un modèle construit sur une population de demandeurs acceptés n'est par conséquent pas complètement correcte. Ajoutons qu'aux refusés par l'organisme, doivent s'ajouter les dossiers classés "sans suite" en raison de refus de clients, quand ceux-ci ont par exemple trouvé de meilleures conditions à la concurrence : c'est une nouvelle source de biais. À l'heure actuelle, malgré de nombreuses tentatives[3] , l'inférence des refusés ne reçoit pas encore de solution statistique complètement satisfaisante. Plusieurs approches existent, dont nous décrivons maintenant les plus courantes.

3.4 Quelques approches de l'inférence des refusés

Première approche : ignorer l'existence des "refusés" et se contenter de modéliser les "acceptés", ce qui équivaut, dans le domaine de l'appétence, à ne construire le modèle que sur les clients ayant déjà fait l'objet d'une proposition commerciale. Cette approche est assez courante dans le domaine du risque, où elle est plus ou moins appropriée, selon le taux de refusés. Elle est moins courante dans le domaine de l'appétence mais peut être pertinente quand des campagnes commerciales suffisamment répétées font que de nombreux clients ont déjà reçu des propositions.

Deuxième approche : considérer tous les dossiers refusés comme "mauvais". Cette approche est évidemment d'autant plus acceptable que le taux de refusés est bas. C'est la position analogue à celle qui consiste à considérer comme "mauvais" pour l'appétence tout non-acheteur, qu'il ait fait l'objet ou non d'une proposition commerciale. L'inconvénient de cette approche est qu'elle rend les résultats complètement tributaires des campagnes commerciales déjà effectuées. Le passé pèse de tout son poids sur le modèle, qui ne pourra que reproduire les règles de ciblage des précédentes campagnes. Si, par préjugé, on n'a jamais ciblé les clients de plus de 50 ans, un certain nombre d'entre eux ont pu devenir clients d'un concurrent plus entreprenant

[3] Lire par exemple Crook et Banasik (2002).

et le modèle enregistrera pourtant que les clients de plus de 50 ans ont une mauvaise appétence.

Pour expliquer la troisième approche ("augmentation") (Hsia, 1978), une des plus répandues, il faut décomposer la probabilité $P(b|x)$ pour qu'un dossier de profil x soit "bon", en l'écrivant sous la forme :

$$P(b|x) = P(b|x,a).P(a|x) + P(b|x,r).P(r|x), \quad (1)$$

où $P(r|x)$ (resp. $P(a|x)$) est la probabilité pour qu'un dossier de profil x soit refusé (resp. accepté). Dans cette méthode, on suppose que la probabilité inconnue $P(b|x,r)$ vérifie l'égalité :

$$P(b|x,r) = P(b|x,a). \quad (2)$$

Cette égalité signifie que le risque d'un dossier ne dépend que de ses caractéristiques propres et non du fait qu'il ait été accepté ou refusé. On pourrait donc faire l'apprentissage du score sur les dossiers acceptés et refusés, après avoir classé chaque refusé en "bon" ou "mauvais" en fonction des dossiers acceptés qui lui ressemblent, en utilisant par exemple la méthode des k-plus proches voisins pour le rapprocher du profil le plus voisin. On voit qu'un dossier accepté isolé au milieu de dossiers refusés aura un poids important dans le modèle inféré, puisque tous ces dossiers refusés se verront affecter la valeur cible du dossier accepté. Plus précisément, chaque dossier accepté entre dans le modèle avec un poids inversement proportionnel à la probabilité $P(a|x)$ qu'il avait d'être accepté. La méthode d'augmentation est donc traditionnellement mise en oeuvre en deux temps : d'abord, on détermine un modèle d'acceptation pour calculer la probabilité $P(a|x)$ de chaque dossier, puis on pondère chaque dossier accepté par $\frac{1}{P(a|x)}$ et on construit le modèle définitif sur les dossiers acceptés ainsi pondérés. C'est cette pondération qui permet en quelque sorte de "compenser" l'absence des refusés.

Quatrième approche ("iterative reclassification") (Joanes, $1993/4$) : l'apprentissage du modèle commence par se faire sur les dossiers acceptés "bons" et "mauvais" ; après cela, on applique le modèle obtenu aux "refusés" pour les classer en "bons" ou "mauvais" ; ensuite, on recalcule le modèle en ajoutant aux acceptés "bons" (resp. "mauvais"), les refusés prédits "bons" (resp. "mauvais") par le premier modèle ; on applique à nouveau ce modèle aux refusés et on réitère la démarche jusqu'à stabilisation des modèles obtenus.

Cinquième approche : définir *a priori* des critères, s'appuyant sur des données disponibles, permettant d'affirmer qu'un dossier "refusé" est bon ou mauvais. On peut de même dire que tous les clients non-acheteurs non ciblés n'ont pas une mauvaise appétence, mais seulement ceux qui vérifient certaines conditions supplémentaires : on peut ainsi supposer qu'en deçà d'un certain revenu ou au-dessus d'un certain âge, un client même s'il avait été ciblé n'aurait probablement pas acheté.

Sixième approche : si l'on suppose que les refusés l'ont été au hasard, on peut les affecter aléatoirement à la catégorie "bon" ou "mauvais" en conservant la même proportion de ces deux catégories que dans la population des acceptés.

Septième approche (si le sujet et la législation le permettent) : compléter la connaissance des "refusés" par des données externes, en s'assurant toutefois que les définitions interne et externe

de la variable cible coïncident. En s'adressant à un credit bureau (pas en France), un établissement peut obtenir des informations sur la santé financière de ses refusés qui ont éventuellement pu être acceptés pour le même produit dans un autre établissement.

Huitième approche (probablement la plus fiable) : accepter quelques centaines de dossiers qui auraient dû être refusés. Très peu d'établissements s'y risquent, mais les pertes engendrées seraient peut-être inférieures aux gains découlant d'un modèle plus fiable. Sur un portefeuille trop risqué, il est aussi envisageable de n'accepter qu'une partie des refusés, tirés au hasard. Cette méthode est évidemment plus facile à appliquer aux modèles d'appétence.

4 Rappel sur l'évaluation des modèles de scoring : courbe de lift et courbe ROC

Avant d'aborder, dans les prochaines sections, la question de la sensibilité des performances des modèles prédictifs de scoring, nous commençons par indiquer comment seront mesurées ces performances.
Face à la multiplicité des méthodes de modélisation, qui chacune possède ses propres indicateurs statistiques de qualité (par exemple, le lambda de Wilks d'une analyse discriminante ou la log-vraisemblance d'une régression logistique), le statisticien a recherché des critères universels de performance d'un modèle. Le critère le plus courant, quand la variable cible est binaire et quand le modèle de score doit permettre de classer tout individu dans un groupe ou dans l'autre, est le taux de bon classement : on convient de fixer un seuil au score, au-delà duquel on est classé dans un groupe (disons, les positifs) et en deçà duquel on est classé dans l'autre (disons, les négatifs) ; on compare le résultat de ce classement à la réalité et on obtient le taux de bon classement. L'inconvénient de cette mesure est qu'elle dépend du seuil de score s choisi. Comme le taux de bon classement est la somme :

- de la proportion (par rapport à la population totale) des positifs détectés (score $\geq s$) ;
- et de la proportion des négatifs détectés (score $< s$) ;

et que ces deux proportions évoluent en sens contraire quand évolue la valeur de s, on a eu l'idée de les observer en faisant varier le seuil de score. Plus exactement, on normalise ces proportions en définissant :

- la sensibilité $\alpha(s)$, qui est la proportion de positifs détectés parmi les positifs, soit $\text{Pro}(\text{score}(x) \geq s | x = \text{positif})$;
- la spécificité $\beta(s)$, qui est $\text{Pro}(\text{score}(x) < s | x = \text{négatif})$.

La courbe ROC est alors définie comme la courbe dont l'ordonnée est la sensibilité du modèle et dont l'abscisse est 1 - spécificité, autrement dit la proportion de tous les négatifs ayant un score $\geq s$ (ce sont les "faux positifs"). Le modèle est d'autant meilleur que nous avons de grandes ordonnées associées à de petites abscisses, c'est-à-dire une aire importante entre la courbe ROC et l'axe des abscisses. Cette aire est comprise entre 0 et 1 et s'interprète comme la probabilité que $\text{score}(x) > \text{score}(y)$, si x est tiré aléatoirement parmi les positifs et y parmi les négatifs.

Il existe une courbe similaire, très utilisée en marketing, la courbe de lift, qui a même ordonnée mais dont l'abscisse est la proportion des individus sélectionnés, c'est-à-dire ayant un score s.

Puisque la courbe de lift a généralement une abscisse plus grande que la courbe ROC pour une même ordonnée, la courbe de lift est sous la courbe ROC. L'aire AUL sous la courbe de lift et l'aire AUC sous la courbe ROC sont liées par la formule :

$$AUL = \frac{p}{2} + (1-p)AUC, \tag{3}$$

où p désigne la probabilité *a priori* d'être positif. En particulier, si $AUC = 1$ (modèle séparant parfaitement) alors $AUL = p/2 + (1-p) = 1 - p/2$. Autre cas particulier, si $AUC = 0,5$ (prédiction aléatoire) alors

$$AUL = p/2 + 1/2 - p/2 = 0,5. \tag{4}$$

Une autre conséquence de la formule (3) est rassurante : pour décréter qu'un modèle est supérieur à un autre, il est équivalent mesurer l'aire sous la courbe de lift ou l'aire sous la courbe ROC.

5 Le nombre de variables explicatives du modèle

Le data miner est parfois représenté comme jonglant avec des centaines de variables dans de gigantesques bases de données en vue de prédire au mieux la variable cible de son problème. Il faut ici insister sur le fait que, si le nombre de variables explicatives candidates peut être énorme, le nombre de variables explicatives finalement retenues pour modéliser la variable cible est généralement très réduit : souvent moins de dix variables dans un modèle courant de score. Non seulement un petit nombre de variables suffit à bien expliquer un phénomène, mais on nuirait à la qualité du modèle en introduisant plus de variables explicatives.
En pratique, le choix des variables entrant dans le modèle se fait généralement pas à pas, en commençant par rechercher et sélectionner la variable expliquant le mieux la variable cible, puis en recherchant la seconde variable qui, jointe à la première, explique le mieux la cible, etc. Dans ce processus cumulatif, il faut s'attendre à ce que la seconde variable sélectionnée ne soit pas la seconde variable la plus discriminante dans l'absolu : c'est la seconde meilleure compte tenu du choix de la première variable. Si la seconde meilleure variable dans l'absolu est très corrélée à la première, presque toute l'information qu'elle serait susceptible d'apporter au modèle sera déjà contenue dans la première variable et sa sélection ne présente pas d'intérêt et ne sera pas opérée. Il sera plus intéressant de sélectionner une variable moins discriminante prise isolément si elle est moins corrélée aux variables déjà présentes dans le modèle. Comme les variables nettement corrélées entre elles sont assez nombreuses dans les problèmes courants, on pressent qu'elles pourront être beaucoup moins nombreuses à entrer dans le modèle.

Une petite illustration peut en être donnée en superposant plusieurs courbes ROC qui montrent l'apport progressif de chaque variable explicative dans un modèle. La figure 1 montre qu'une seule variable, bien choisie, explique déjà en bonne partie le phénomène à prédire, et que plus les variables ajoutées au modèle sont nombreuses, plus faible est le gain marginal qu'elles apportent.
David J. Hand (2005) précise ce phénomène en prenant l'exemple d'une régression et en calculant la proportion $1 - R^2$ de la variance non expliquée (par la régression) en fonction du nombre de prédicteurs dans le modèle. Il prend pour hypothèse un coefficient de corrélation

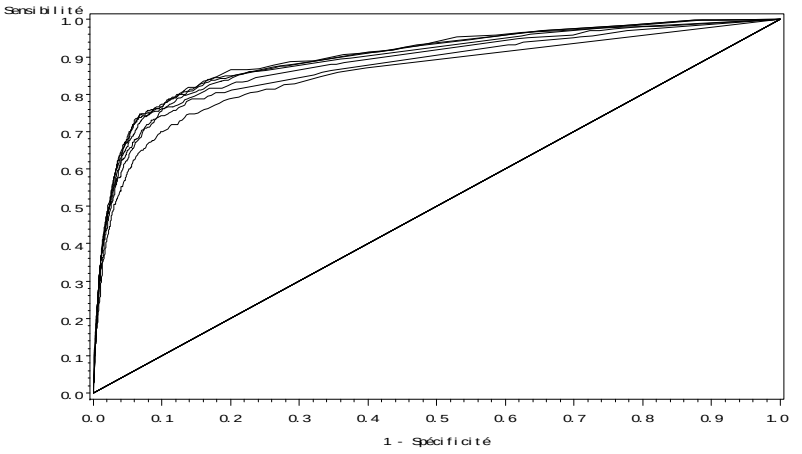

FIG. 1 – *Courbes ROC après entrées successives des variables dans le modèle.*

$= 0,5$ entre chaque prédicteur et la variable cible, un coefficient de corrélation ρ entre prédicteurs, et il en déduit une formule dont il dérive plusieurs résultats en faisant varier ρ entre $0,1$ et $0,9$. La figure 2 représente la baisse de la variance non expliquée pour les différentes valeurs de ρ . Lorsque les prédicteurs sont très corrélés ($\rho = 0,9$), presque toute la variance expliquée l'est par la première variable. À l'opposé, une corrélation nulle entre les prédicteurs donnerait une droite diagonale entre le coin supérieur gauche et le coin inférieur droit. Ce résultat de David Hand est assez spectaculaire, et pourtant il est obtenu en supposant que tous les prédicteurs ont la même corrélation avec la cible. On imagine la situation lorsque, comme dans une sélection pas à pas, la corrélation des prédicteurs successifs avec la cible décroît rapidement.
Quel coefficient de corrélation peut-on donc tolérer entre les prédicteurs ? Cela dépend évidemment de la quantité et de la qualité des variables candidates à notre disposition : plus elles seront importantes et plus nous pourrons être exigeants. Quand les variables disponibles sont peu nombreuses et quand nous peinons à obtenir un pouvoir prédictif suffisant (parfois imposé par les autorités ou par le contexte), nous pourrons tolérer un coefficient de corrélation $\rho = 0,7$ voire $0,8$ en faisant très attention. Dans tous les cas, un coefficient $\rho \geq 0,9$ est prohibitif.
S'il faut vraiment manipuler des variables explicatives très corrélées entre elles, une solution peut être le recours à la régression PLS. Celle-ci fonctionne en réalisant un compromis entre deux objectifs : maximiser la variance expliquée des variables explicatives (principe de l'ACP) et maximiser la variance expliquée de la variable cible (principe de la régression). Pour son aptitude à traiter des variables explicatives nombreuses et corrélées, la régression logistique PLS est particulièrement appréciée en chimie, spectrométrie, industrie du pétrole, cosmétique et biologie. Son apport dans les contextes classiques de la banque, de l'assurance et du marketing est sans doute moindre.

Dans les problèmes courants, il est donc préférable de limiter le nombre de variables d'un modèle. Une raison subsidiaire en est que, plus les variables sont nombreuses dans un modèle, plus le risque est grand de voir l'une de ces variables provoquer une défaillance du modèle car elle aura été un jour mal alimentée par les programmes informatiques de constitution de la base de données. Il en va des modèles comme des mécaniques : plus elles sont complexes, plus elles risquent de tomber en panne.

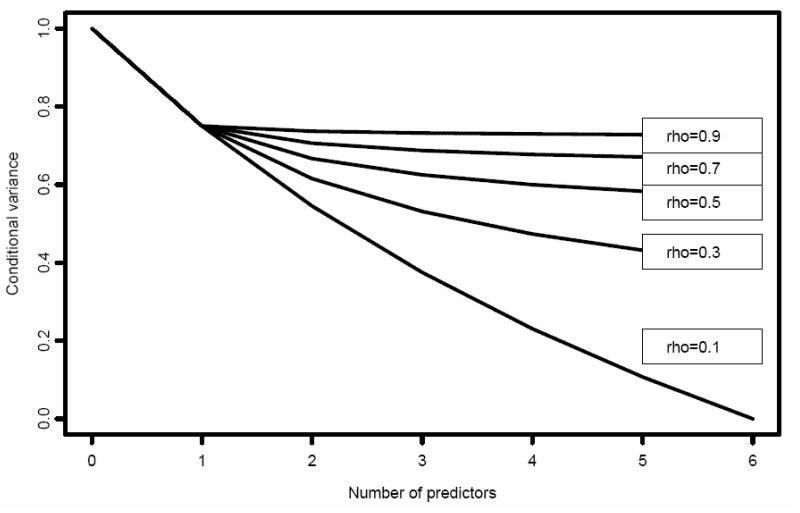

FIG. 2 – *Proportion de la variance non expliquée en fonction du nombre de prédicteurs.*

Nous concluons cette section en mentionnant la limite des méthodes de sélection pas à pas des variables. Comme dans les arbres de décision, même si c'est de façon moins marquée, ces méthodes accordent une importance plus grande à la première variable sélectionnée, puisque celle-ci peut occulter une variable qui lui est fortement corrélée. On constate parfois qu'en forçant le choix d'une autre première variable, les variables ensuite sélectionnées conduisent à un modèle plus prédictif. Dans le cas d'une variable à expliquer binaire et de variables explicatives continues, on peut recourir à l'algorithme de sélection globale *leaps and bound* (Furnival et Wilson, 1974), qui cherche à calculer les meilleures régressions pour $1, 2, \ldots, k$ variables explicatives, en comparant une partie de tous les modèles possibles et en éliminant les moins intéressants *a priori*. Un logiciel de statistique n'est pas toujours pourvu de cet algorithme, mais, s'il est rapide et pourvu d'un langage de programmation, on peut aussi calculer et évaluer automatiquement tous les modèles à $1, 2, \ldots, k$ variables explicatives, en prenant toutes les combinaisons possibles de variables.

6 Sensibilité des performances au choix des échantillons

La figure 1 montre le faible gain de performance qu'il faut parfois attendre de l'augmentation du nombre de variables du modèle. Nous allons voir dans cette section que, non seulement ce gain est faible, mais qu'il est en partie illusoire. Quand un modèle est construit sur un échantillon d'apprentissage et validé par une mesure de l'aire sous la courbe ROC sur un échantillon de test, la valeur de cette aire, dans une certaine mesure, dépend moins du nombre de variables que du choix des échantillons d'apprentissage et de test !

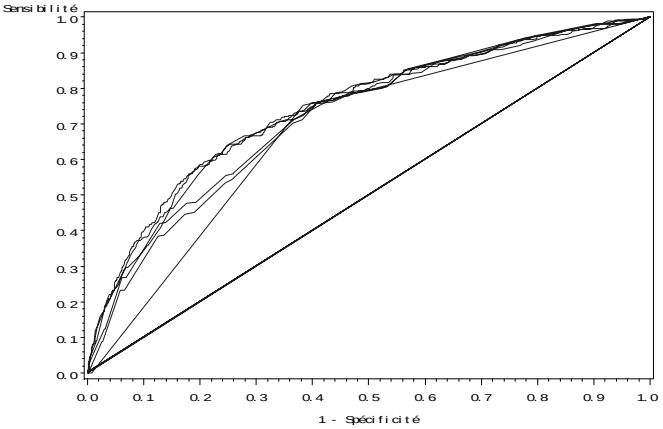

FIG. 3 – *Courbes ROC après entrées successives des six variables dans le modèle.*

Nous allons le montrer sur l'exemple d'un modèle de prédiction de souscription d'assurance caravane. Nous nous appuyons sur un fichier de 5822 assurés fourni à partir de cas réels par la compagnie néerlandaise Sentient Machine Research. Ce fichier a été utilisé dans un concours de data mining prédictif en 2000, l'objet de la compétition étant de construire le meilleur modèle prédictif (P. van der Putten et M. van Someren, 2000).
On commence par travailler sur l'ensemble du fichier. Une sélection pas à pas des variables nous fournit des modèles de régression logistique à $1, 2, \ldots, 6$ variables explicatives, dont les aires sous la courbe ROC croissent ainsi : $0,680$ pour 1 variable, $0,715$ pour 2 variables, ensuite $0,725$, puis $0,738$, puis $0,742$ et enfin $0,744$ pour 6 variables. Les courbes ROC correspondantes sont représentées sur la figure 3. On voit que le pouvoir prédictif du modèle logistique n'augmente plus beaucoup à partir de la quatrième variable, une fois atteinte l'aire $0,738$.
On tire ensuite mille échantillons aléatoires de 65% de la population, qui servent à construire des modèles avec les six variables précédemment sélectionnées, tandis que les 35% restants servent au test des modèles. On mesure pour chaque échantillon d'apprentissage et de test les deux aires correspondantes sous la courbe ROC, puis on mesure la moyenne, l'écart-type et d'autres caractéristiques de ces aires.
Le tableau 1 montre l'énorme dispersion des aires dans l'ensemble des échantillons. Dans ces conditions, non seulement on peut se demander ce que signifie la performance du modèle,

Paramètres	AUC apprentissage	AUC test
moyenne	0,745	0,741
écart-type s	0,0107	0,0199
coef. de variation	1,44%	2,69%
moyenne + 1,96s	0,766	0,780
moyenne - 1,96s	0,724	0,702
minimum	0,715	0,664
maximum	0,780	0,800
centiles :		
1	0,719	0,690
2,5	0,724	0,702
5	0,727	0,707
25	0,738	0,728
50	0,744	0,742
75	0,752	0,754
95	0,763	0,773
97,5	0,766	0,780
99	0,772	0,789

TAB. 1 – *Distribution des aires* AUC *sur* 1000 *échantillons* $65 - 35$ *sans remise.*

quand l'aire AUC de l'échantillon de test a un intervalle de confiance à 95% allant de $0,702$ à $0,780$, mais la sélection même des variables peut être sujette à caution, car elle n'aurait pas nécessairement été la même sur un autre échantillon initial, hormis pour les premières variables dont le pouvoir discriminant s'impose (c'est ainsi le cas de la première variable du modèle, la détention d'une assurance automobile, qui est de loin le facteur le plus discriminant). On voit que le modèle final à 6 variables $(AUC = 0,744)$ et le modèle à 4 variables $(AUC = 0,738)$ ne sont séparés que par un quartile dans le tableau 1. On sera donc tenté de se limiter à ces quatre variables dans la construction du modèle.

7 Sensibilité des performances au choix de la méthode de modélisation

Nous avons vu que l'essentiel des performances des modèles de score était obtenu avec un petit nombre de variables explicatives, et qu'il était vain de vouloir attendre des performances très supérieures. Ceci est d'autant plus vrai que des incertitudes importantes pèsent souvent, nous l'avons vu en début d'article, sur les conditions d'obtention des modèles. Nous pourrions espérer obtenir un gain substantiel de performance par le recours à des techniques subtiles de modélisation. Là encore, de nombreux comparatifs montrent que les écarts de performance entre les différentes techniques sont ténus. Un exemple de comparatif est celui de Saporta et Niang (2006) effectué sur des données d'assurance automobile. Dans cette étude de cas, ils obtiennent les valeurs suivantes de l'aire sous la courbe ROC :

- régression logistique : $0,933$;
- régression logistique PLS : $0,933$;
- analyse discriminante DISQUAL : $0,934$;
- analyse discriminante barycentrique : $0,935$.

Ils citent d'ailleurs Hastie, Tibshirani et Friedman (Hastie et al., 2001, p. 105) : "It is generally felt that logistic regression is a safer, more robust bet than the LDA model, relying on fewer assumptions. It is our experience that the models give very similar results, even when LDA is

used inappropriately, such as with qualitative variables."
Ce constat est aussi celui de nombreux praticiens, qui ont remarqué combien les performances d'un modèle dépendaient souvent relativement peu du choix de la technique de modélisation et ne suffisaient pas à elles seules à motiver ce choix.
En vertu du principe de parcimonie (prédire le plus possible le plus simplement possible), et compte tenu des incertitudes mentionnées, nous serons donc souvent amenés à privilégier des techniques simples et classiques, qui fournissent l'essentiel si ce n'est la totalité de la performance des modèles, et permettent en outre de limiter les risques de sur-apprentissage.
Si la performance d'un modèle de score ne peut pas beaucoup s'améliorer en augmentant le nombre de variables explicatives ou en recourant à une technique sophistiquée de modélisation, n'y a-t-il aucune piste d'amélioration ? Au contraire, nous allons voir, dans les dernières sections de cet article, que le statisticien et le data miner ont à leur disposition plusieurs méthodes efficaces d'amélioration des modèles, applicables à tout type de classifieur de base.

8 La discrétisation des variables continues

En réalité, le point essentiel dans la performance d'un modèle est la qualité de la préparation des données. Nous n'énumérerons pas ici l'ensemble des opérations à accomplir, de fiabilisation des données, de croisement, de recodage, de normalisation, etc. Les interactions de variables, surtout qualitatives, peuvent être intéressantes. Qu'il nous suffise ici de citer une opération : la discrétisation, qui consiste à découper en plusieurs tranches les valeurs d'une variable continue. Il n'est pas immédiat d'admettre que le découpage en tranches fait perdre moins d'information qu'il n'en fait gagner. Pourtant, quand la méthode de modélisation se prête à la prise en compte de variables explicatives catégorielles (régression logistique, analyse discriminante DISQUAL), on peut trouver au moins six raisons de discrétiser les variables explicatives continues :

- la prise en compte des valeurs manquantes (dont l'imputation est toujours délicate) qui se trouvent regroupées dans une modalité spécifique ;
- la neutralisation des outliers (individus hors norme) qui se trouvent intégrés à la première ou la dernière modalité ;
- la prise en compte de la non-monotonie et de la non-linéarité (avec autant de coefficients que de modalités, au lieu d'en avoir un seul) ;
- la prise en compte des ratios dont le numérateur et le dénominateur peuvent être tous deux > 0 ou < 0 (on trouve de tels ratios en analyse financière) ;
- une plus grande robustesse (on constate souvent que 2 ou 3 modalités permettent d'augmenter l'aire sous la courbe ROC, du moins sur l'échantillon de test, par rapport à 4 ou 5 modalités) ;
- la prise en compte simultanée des variables qualitatives et quantitatives, la discrétisation de ces dernières permettant de les mettre sur le même plan que les variables qualitatives (et d'effectuer éventuellement une ACM sur l'ensemble).

Dougherty et al. (1995) classent les méthodes de discrétisation selon trois critères : méthodes supervisées vs non supervisées, globales vs locales et dynamiques vs statiques.

Les méthodes supervisées tiennent compte de l'existence d'une classification de la population (par exemple, selon la présence ou l'absence de risque) pour discrétiser les variables continues. Le découpage en modalités de chaque variable continue est effectué en sorte d'optimiser une mesure de la liaison entre la variable continue discrétisée et la variable de classe. Cette mesure peut reposer sur l'entropie (Fayyad et Irani, 1993) et il s'agira de minimiser la somme des entropies des modalités, comme dans l'arbre $C4.5$ (Quinlan, 1993). Une autre approche est l'indice de pureté de Gini qu'il s'agira de minimiser comme dans l'arbre CART (Breiman et al., 1984). On peut aussi envisager d'utiliser l'arbre 1R à un niveau de Holte (1993), même s'il a tendance à produire un grand nombre de classes et à être sujet au sur-apprentissage. Cela tient à son principe : découper la variable continue chaque fois que la variable de classe change, en imposant toutefois un minimum (déterminé empiriquement par Holte) de 6 individus dans chaque modalité. La mesure à optimiser peut aussi être le critère du χ^2 comme dans l'algorithme $ChiMerge$ (Kerber, 1992) et l'arbre CHAID qui découpe d'abord la variable continue en modalités avant de fusionner deux à deux les plus proches en terme de χ^2. Similaire est l'algorithme $StatDisc$ (Richeldi et Rossotto, 1995) qui remplace χ^2 par Φ et peut fusionner plus de deux modalités à chaque fois. Dans ces deux derniers algorithmes, le nombre de modalités de la variable continue est contrôlé par le seuil fixé au χ^2 et à Φ . Nous avons mentionné l'implémentation sous forme d'arbres de décision des algorithmes précédents, car elle permet une mise en oeuvre opérationnelle facilitée par les logiciels de data mining incluant ces arbres, que l'on peut utiliser au besoin en écrivant des macros.
Les points délicats des méthodes supervisées sont, d'une part une éventuelle tendance au sur-apprentissage en cas de découpage trop fin (à juger en fonction du volume de données), et d'autre part un temps de calcul qui peut devenir trop long si les variables à découper sont nombreuses et la population importante. On pourra dans ce cas, échantillonner la population et limiter le nombre de variables en éliminant les moins prédictives ou les moins utilisables *a priori*.
Les méthodes non-supervisées ne posent pas la même difficulté. Certaines méthodes s'appuient sur des algorithmes de classification automatique comme les k-means pour obtenir des groupes d'individus homogènes. Les deux méthodes non-supervisées les plus simples et les plus rapides sont celles qui consistent à découper l'étendue de la variable continue en n tranches de même largeur ou de même effectif (n-tiles). La seconde méthode est préférable car elle n'est pas sensible à la présence d'outliers.
Même si le découpage des variables continues par une méthode non-supervisée peut déjà améliorer sensiblement les performances d'un modèle prédictif, en découpant par exemple les variables en quartiles, les meilleurs résultats sont évidemment obtenus par les méthodes supervisées, qui découpent les variables en fonction de la variable à expliquer.

Un autre axe d'analyse oppose les méthodes globales aux méthodes locales de discrétisation. Dans une méthode globale, le découpage de chaque variable continue est le même dans tout l'espace des individus, ce qui signifie qu'elle est découpée indépendamment des autres variables. Dans une méthode locale, le découpage varie selon les régions de l'espace, comme dans un arbre de décision où une variable apparaissant dans deux noeuds ne sera pas forcément découpée deux fois à l'identique. Les arbres de décision mentionnés plus haut sont susceptibles de produire des discrétisations locales lorsqu'ils ont au moins deux niveaux de profondeur. Les méthodes locales peuvent paraître plus puissantes mais elles sont aussi plus longues en temps

de calcul, moins robustes sur de petits effectifs et souvent guère plus performantes, comme le montre le comparatif de Dougherty et al. (1995) établi sur 16 jeux de données. De ce fait, la plupart des méthodes employées sont globales.

La dernière distinction opérée par Dougherty et al. (1995) est intéressante : ils distinguent les méthodes "statiques" usuelles, dans lesquelles le nombre de modalités de chaque variable discrétisée est déterminé indépendamment des autres, des méthodes "dynamiques" dans lesquelles les nombres de modalités interagissent. De telles méthodes sont encore à développer, mais elles sont prometteuses, tant l'on constate dans le travail courant de modélisation qu'il n'est pas toujours possible de discrétiser le plus finement toutes les variables continues (risque de sur-apprentissage et de manque de robustesse du modèle sur de faibles effectifs) ni de les discrétiser le plus grossièrement (c'est-à-dire de les binariser), car dans ce dernier cas, le modèle manque de finesse et de précision. L'arbitrage entre précision et robustesse impose un arbitrage entre le nombre de modalités de chaque variable du modèle.

Pour montrer l'intérêt de la discrétisation d'une variable continue, nous avons comparé les aires sous la courbe ROC de régressions logistiques construites avec les quatre mêmes variables explicatives :

- laissées sous leur forme initiale continue ($\text{AUC} = 0,820$) ;
- découpées en modalités considérées comme ordonnées ($\text{AUC} = 0,834$) ;
- découpées en modalités considérées comme nominales ($\text{AUC} = 0,836$).

La discrétisation a été effectuée à l'aide d'une méthode basée sur le χ^2. On voit que le découpage en modalités nominales est le plus performant. Cela vient de ce que, au lieu d'avoir un seul odds-ratio, ce qui suppose que les probabilités évoluent identiquement entre 0 et 1, 1 et 2, 2 et 3 . . ., on a autant de odds-ratios que de modalités, ce qui permet de prendre en compte des réponses non linéaires, et même non monotones. Le découpage en modalités ordonnées signifie que l'on considère le numéro des modalités comme une variable discrète $1, 2, \ldots$ à laquelle est associé un seul odds-ratio. Par rapport au découpage en modalités nominales, on perd la puissance de modélisation des odds-ratios multiples, mais par rapport à la variable continue d'origine, on gagne en robustesse, ce qui explique la performance très honorable de ce modèle. Comme pour illustrer ce que nous avons déjà dit dans les sections précédentes, la légère surperformance du découpage nominal par rapport au découpage ordinal ne justifie peut-être pas la complexité des odds-ratios multiples.
En tout cas, cet exemple montre que l'on peut gagner autant en pouvoir prédictif, voire plus, en discrétisant les variables continues qu'en ajoutant plusieurs nouvelles variables, tout en conservant, bien entendu, une plus grande robustesse du modèle.

9 La partition de modèles

La partition (ou stratification) de modèles consiste à faire précéder la modélisation d'une classification de la population, puis à construire un modèle différent pour chacune des classes, avant d'en faire la synthèse. Puisque ajouter le plus grand nombre de variables possible n'améliore pas, mais généralement détériore un modèle, il est intéressant de partitionner la population avant de la modéliser, afin de pouvoir travailler sur des groupes homogènes, nécessitant moins de variables pour les décrire. Ceci permet fréquemment d'améliorer notablement les résultats.

Il existe plusieurs façons de procéder à cette classification de la population ; elles dépendent des données et de la méthode de classification qui sont utilisées. Le tableau 2 présente une combinatoire possible. On peut ainsi effectuer une classification selon des caractéristiques di-

		Méthode de classification	Méthode de classification
		Classification statistique	Règles d'experts
Données utilisées liées à l'événement à prédire	Oui	solution efficace	solution pouvant être efficace
Données utilisées liées à l'événement à prédire	Non	solution risquant d'être moyennement efficace	solution pouvant être efficace si les données sont bien choisies

TAB. 2 – *Solutions pour la partition de modèles.*

rectement liées à l'événement à prédire (partition supervisée). Généralement, une partition supervisée est plus efficace car elle contient une partie du pouvoir discriminant demandé au modèle. Elle peut être mise en oeuvre à partir de règles de métier ou de bon sens, ou par une méthode statistique, par exemple un arbre de décision à un ou deux niveaux de profondeur. Un arbre de décision à un seul niveau de profondeur permet déjà de séparer en deux classes (ou un peu plus) bien différenciées la population à modéliser, sans présenter l'instabilité propre aux arbres plus profonds. Une règle de bon sens pourra être d'établir un score de risque à la carte de crédit sur une population segmentée par niveau d'utilisation de la carte : faible, moyenne ou forte. Un score d'appétence au crédit à la consommation pourra être élaboré sur deux segments : celui des clients déjà équipés d'un tel crédit, et les autres. On remarque que dans ces deux exemples les règles peuvent s'exprimer par un arbre de décision.
On peut aussi effectuer une classification selon des caractéristiques générales (partition non supervisée). Il peut s'agir des caractéristiques sociodémographiques (âge, profession. . .) d'une personne physique. Pour une entreprise, il peut s'agir de sa taille, de sa nature juridique ou de son secteur d'activité. Cette méthode peut se justifier dans une approche marketing par l'existence d'offres spécifiques orientées vers certains segments de clientèle : programmes "jeunes", "seniors", etc. Du point de vue statistique, si elle n'est pas toujours optimale car les comportements ne sont pas forcément liés aux caractéristiques générales, cette méthode peut parfois s'avérer très efficace. Cela est notamment le cas de la modélisation de la santé financière des entreprises, laquelle s'appuie sur des ratios financiers dont les valeurs et l'interprétation dépendent beaucoup du secteur d'activité de l'entreprise. D'ailleurs, depuis plusieurs années, la Banque de France élabore des modèles par secteurs d'activité : industrie, commerce, transports, hôtels, restaurants, construction, services. Nous avons testé un partitionnement simple de petites entreprises, celles-ci étant réparties en seulement deux classes d'activité : en construisant un modèle pour chacune des deux sous-populations, nous avons obtenu une aire sous la courbe ROC égale à $0,749$, alors qu'un modèle unique appliqué à l'ensemble de ces entreprises avait une aire sous la courbe ROC égale à $0,726$. Et pourtant, les taux de défaillance étaient comparables (moins de 7% d'écart entre les deux classes) et plus de la moitié des variables était commune aux deux sous-populations, avec des discrétisations et des coefficients différents, il est vrai.
Dans tous les cas, la classification devra s'exprimer selon des règles explicites permettant de classer tout individu. Une fois la partition de la population effectuée et un modèle construit sur chaque classe, une étape technique consiste à partir des fonctions de score des différentes classes pour calculer une fonction de score unique définie sur l'ensemble de la population.

Cette opération est plus aisée quand le modèle est une régression logistique, puisque le score est alors une probabilité, qui est par essence normalisée et se prête à la même interprétation dans toutes les classes.
La partition de modèles atteint évidemment ses limites lorsque le volume de données est faible et ne permet pas d'avoir dans chaque classe suffisamment d'individus de chaque catégorie à prédire.

10 L'agrégation de modèles

10.1 Le recours au bootstrap

Un problème classique rencontré en statistique et plus généralement en data mining est celui de l'estimation d'un paramètre statistique θ. Un tel paramètre est défini dans une population globale Ω, et il est une fonction de la loi statistique F définie sur Ω. Ce paramètre peut ainsi être la moyenne μ de F. Or, la population globale et la loi F sont généralement inconnues, d'autant que la population (par exemple, un ensemble de clients) peut être en évolution perpétuelle ou qu'il peut exister des erreurs de mesure, de saisie, etc. Quand nous travaillons sur un jeu de données, il s'agit donc presque toujours d'un échantillon $S = \{x_1, x_2, \ldots, x_n\}$ tiré d'une population globale inconnue, et l'on cherche à approcher le paramètre θ par un estimateur $\hat{\theta}$ défini sur l'échantillon S, cet estimateur $\hat{\theta}$ étant obtenu en remplaçant la loi inconnue F par la loi dite "empirique", qui est la loi discrète donnant une probabilité $\frac{1}{n}$ à chaque x_i. Cet estimateur est appelé estimateur "plug-in" ; on le note $\hat{\theta} = s(S)$ pour signifier qu'il dépend de l'échantillon S. Ainsi, $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i$ est un estimateur "plug-in" de la moyenne μ. Si F est la loi normale de moyenne μ_F et de d'écart-type σ_F(ou si $n > 30$), on connaît la distribution des estimateurs : elle suit la loi normale de moyenne μ_F et de d'écart-type $\frac{\sigma_F}{\sqrt{n}}$. Puisque la moyenne des $\hat{\mu}$ est μ, on dit que $\hat{\mu}$ est un estimateur sans biais. De plus, il est donné par une formule explicite, de même que son écart-type.
Plus généralement, pour un paramètre autre que la moyenne, se pose la question de la précision et de la robustesse de l'estimateur, c'est-à-dire de son biais et de son écart-type, lesquels ne sont pas généralement pas donnés par une formule explicite. Pour calculer l'écart-type de l'estimateur, il faudrait pouvoir déterminer l'estimateur sur un grand nombre d'échantillons $S', S'' \ldots$. Or, souvent un seul échantillon S nous est donné ; c'est typiquement le cas d'un sondage, mais pas seulement. L'idée de Bradley Efron (1979) en inventant le " bootstrap " fut de reproduire le passage de la population Ω à l'échantillon S étudié, en faisant jouer à $S = \{x_1, x_2, \ldots, x_n\}$ le rôle d'une nouvelle population et en obtenant les échantillons souhaités $S', S'' \ldots$ par des tirages aléatoires avec remise des n individus $x_1, x_2, \ldots, x_n$. On appelle "échantillon bootstrap" un tel échantillon obtenu par tirage avec avec remise de n individus parmi n. Dans un échantillon bootstrap, un x_i peut être tiré plusieurs fois ou ne pas être tiré. La probabilité qu'un x_i donné soit tiré est égale à $1 - (1 - 1/n)^n$, qui tend vers $0,632$ quand n tend vers $+\infty$. Quand on a tiré un certain nombre B (en général $B \geq 100$) d'échantillons bootstrap S^* et calculé sur chacun d'eux l'estimateur " plug-in " $\hat{\theta}^* = s(S^*)$, on obtient une distribution des estimateurs " plug-in " bootstrap centrée autour de la moyenne $\frac{1}{B}\sum_{b=1}^{B} \hat{\theta}^{*b}$, d'où l'on déduit un écart-type qui fournit l'approximation recherchée de l'écart-type de l'estimateur $\hat{\theta}$. On peut aussi déduire des intervalles de confiance des quantiles de la distribution : on prendra B assez grand, par exemple $B = 1000$ (c'est un minimum selon Efron) et on regar-

dera la 25^e plus faible valeur $Q_{2,5}$ et la 25^e plus forte valeur $Q_{97,5}$ de l'estimateur bootstrap pour avoir une idée de l'intervalle de confiance $[Q_{2,5}; Q_{97,5}]$ à 95% de l'estimateur. Quant au biais, son approximation bootstrap est

$$\frac{1}{B}\sum_{b=1}^{B}\hat{\theta}^{*b} - \hat{\theta}, \tag{5}$$

c'est la différence entre la moyenne des estimateurs bootstrap et l'estimateur calculé sur S.
La philosophie du bootstrap est en résumé celle-ci : faire jouer à l'échantillon S le rôle de la population globale Ω , et faire jouer à l'échantillon bootstrap le rôle de l'échantillon S, sachant que $\hat{\theta}^*$ se comporte par rapport à $\hat{\theta}$ comme $\hat{\theta}$ par rapport à θ et que la connaissance de $\hat{\theta}^*$ (distribution, variance, biais) contribue à celle de $\hat{\theta}$.
Dans les problèmes de scoring, les paramètres θ que l'on cherche à estimer peuvent être :

- le taux d'erreur (ou de bon classement) ou une autre mesure de performance du modèle de score (aire sous la courbe ROC, indice de Gini ...) ;
- les coefficients de la fonction de score ;
- les prédictions (probabilités a *posteriori* d'appartenance à chaque classe à prédire).

Comme la population globale sur laquelle devrait être construit le modèle est inconnue, les paramètres précédents ne peuvent être qu'estimés. On commence par construire B échantillons bootstrap à partir de l'échantillon initial, après quoi on construit un modèle sur chaque échantillon bootstrap. On obtient B modèles de score. Le bootstrap sur le taux d'erreur ou l'aire sous la courbe ROC permet d'obtenir des intervalles de confiance de ces indicateurs de performance du modèle. La situation est représentée dans la figure 4. À noter que la moyenne des taux d'erreur sur les échantillons bootstrap est une estimation biaisée par optimisme, dans la mesure où ces taux d'erreur sont calculés par resubstitution sur les individus qui ont servi à l'apprentissage du modèle. Une variante représentée dans la figure 5 consiste à calculer les erreurs sur les seuls individus n'appartenant pas à l'échantillon bootstrap : on parle d'estimation "out-of-bag". Comme cette estimation est cette fois-ci biaisée par pessimisme, Efron et Tibshirani ont proposé de pallier simultanément le biais optimiste de l'estimation par resubstitution et le biais pessimiste de l'estimation "out-of-bag" par la "formule magique" du ".632-bootstrap" :

$$\text{Estimation}_{.632} = 0,368 \times \text{estimation(resubstitution)} + 0,632 \times \text{estimation(bootstrap-oob)}.$$

Cette formule tient compte de la probabilité $= 0,632$ de sélection de chaque individu dans un des différents échantillons bootstrap (autrement dit : un individu appartient en moyenne à $0,632 \times B$ échantillons bootstrap[4]), ce qui est la cause de la fluctuation trop importante de l'estimateur "out-of-bag". On peut l'appliquer à un indicateur de performance telle l'aire sous la courbe ROC. Comme cette estimation est parfois elle-même trop optimiste dans les situations de fort sur-apprentissage, les mêmes auteurs en ont proposé une variante plus élaborée appelée ".632 + bootstrap" (Efron et Tibshirani, 1997).

Appliquant la formule du ". 632-bootstrap" au modèle de prédiction de souscription d'assurance caravane décrit précédemment dans la section 6, nous obtenons avec 1000 itérations une

[4]Réciproquement, un échantillon contient en moyenne $0,632 \times n$ individus différents

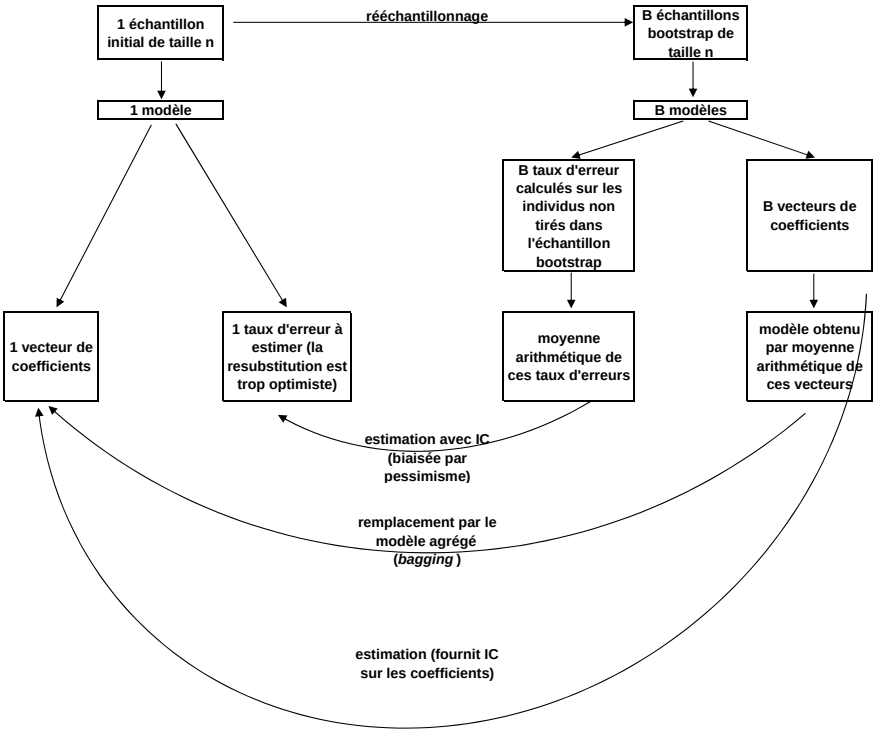

FIG. 4 – *Rééchantillonnage boostrap et bagging.*

estimation de l'aire AUC du modèle logistique à six variables, qui est égale à :

$$0,368 \times \text{estimation(resubstitution)} + 0,632 \times \text{estimation(bootstrap-oob)}$$
$$= (0,368 \times 0,744) + (0,632 \times 0,740) = 0,741.$$

Notons que la distribution des aires AUC (tableau 3) est proche de celle obtenue en recourant à des échantillons aléatoires $65 - 35$ sans remise (voir le tableau 1 plus haut). Il est vrai que le bootstrap fournit des échantillons à peu près $63 - 37$.
Le bootstrap sur les coefficients de la fonction de score est notamment utilisé avec l'analyse discriminante linéaire pour obtenir des intervalles de confiance des coefficients et pouvoir juger de l'apport véritable de chaque variable explicative.

10.2 L'agrégation par bagging

Quant au bootstrap sur les prédictions, il est connu depuis Leo Breiman sous le nom de *bagging*, ou "Bootstrap AGGregatING" (Breiman, $1996a$). Les prédictions de chaque échantillon bootstrap sont agrégées par un vote (classement) ou une moyenne (régression). La moyenne

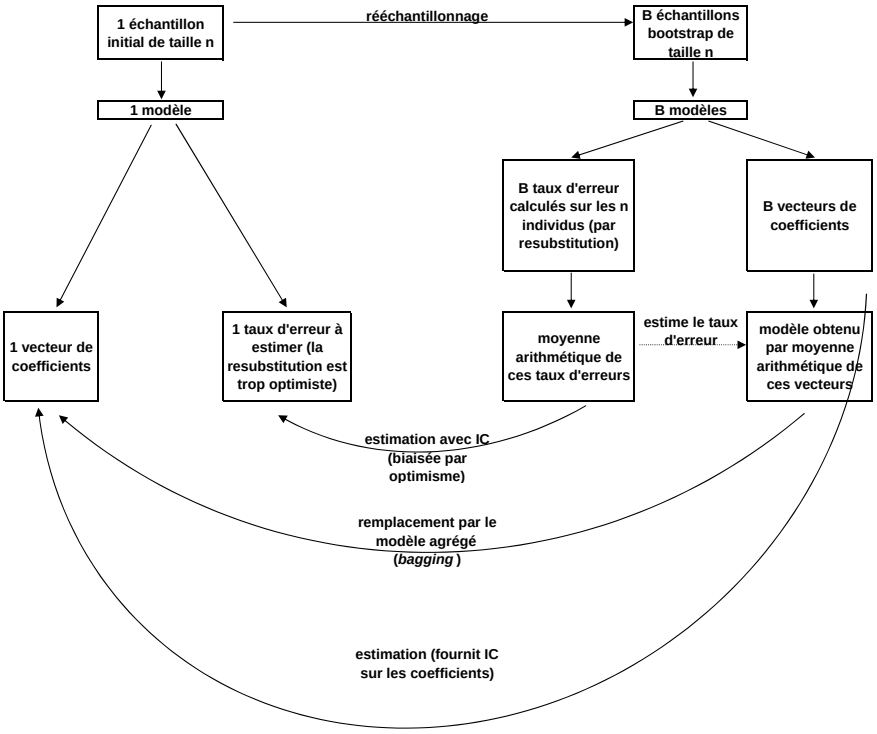

FIG. 5 – *Rééchantillonnage boostrap avec estimation "out-of-bag".*

	AUC resubstitution	AUC "out-of-bag"
moyenne	0,744	0,740
écart-type s	0,0113	0,0192
coef. de variation	1,52%	2,60%
moyenne + 1,96s	0,767	0,778
moyenne - 1,96s	0,722	0,702
minimum	0,709	0,680
maximum	0,787	0,805
centiles :		
1	0,718	0,690
2,5	0,723	0,702
5	0,727	0,707
25	0,737	0,727
50	0,744	0,740
75	0,752	0,753
95	0,763	0,770
97,5	0,767	0,776
99	0,773	0,784

TAB. 3 – *Distribution des aires AUC sur 1000 échantillons bootstrap.*

peut aussi être utilisée avec une régression logistique en calculant les moyennes des probabilités a posteriori fournies par le modèle ; dans ce cas, les moyennes des probabilités sont approchées par les probabilités fournies par le modèle "moyen", c'est-à-dire le modèle dont les coefficients sont pour chaque variable la moyenne des coefficients des différents modèles.
Le *bagging* permet de diminuer la variance mais non le biais d'un modèle, et il est surtout intéressant pour corriger le manque de robustesse des classifieurs instables comme les arbres de décision et les réseaux de neurones. Dans le cas des arbres, il est cependant important de remarquer que l'agrégation de plusieurs arbres détruit la structure simple d'arbre de décision et fait disparaître le principal intérêt des arbres : leur lisibilité. En revanche, dans le cas de l'analyse discriminante et de la régression logistique, le bagging n'augmente pas la complexité du modèle par rapport au modèle de base, puisque le modèle résultant est simplement le modèle "moyen". Pour ces classifieurs stables, le bagging présente toutefois moins d'intérêt : la variance du modèle est abaissée mais dans une moindre mesure. Il n'est toutefois pas inintéressant car il assure une meilleure généralisation du modèle. En effet, dans la distribution des estimateurs bootstrap (voir le tableau 3), les modèles dont l'aire AUC "out-of-bag" est maximale ne sont généralement pas ceux dont l'aire sera maximale sur un nouvel échantillon indépendant de l'échantillon initial (par exemple, un échantillon constitué à une date ultérieure). En d'autres termes, les aires AUC "out-of-bag" et les aires AUC mesurées sur un échantillon indépendant sont faiblement corrélées (coefficient couramment inférieur à $0,1$). Quant au modèle agrégé, le modèle "moyen", dans les exemples que nous avons traités, son aire AUC sur l'échantillon initial est très proche de la moyenne des aires AUC "in-the-bag", et son aire AUC sur un échantillon de test indépendant se situe environ au niveau du 173^e sur 500 modèles bootstrap. Si l'on répartit les 500 modèles en déciles selon leur aire AUC "out-of-bag" et que l'on classe ces déciles selon leur aire AUC moyenne sur l'échantillon indépendant de test, on s'aperçoit que ces déciles ne sont pas classés dans l'ordre : les 50 meilleurs modèles selon l'aire AUC "out-of-bag" ne sont pas les 50 meilleurs sur l'échantillon indépendant. En définitive, le modèle agrégé n'est pas le plus performant sur un échantillon de test indépendant, mais sa performance est nettement supérieure à la moyenne et la médiane, et de toutes façons le meilleur modèle n'est pas prédictible et n'est en tout cas pas le plus performant selon un estimateur "in-the-bag" ou "out-of-bag". On a donc intérêt à utiliser le modèle agrégé, pour un gain de performance qui sans être colossal n'est pas toujours négligeable.

On peut attribuer une partie de l'efficacité du bagging au fait que les éventuels outliers de l'échantillon initial ne se retrouvent que dans certains échantillons bootstrap, et que la moyenne des estimations bootstrap fait perdre de leur nuisance à ces outliers. Cette efficacité repose aussi simplement sur le fait que la moyenne de deux estimateurs sans biais $\hat{\theta}_1$ et $\hat{\theta}_2$ du même paramètre θ , non corrélés et de même variance $v = Var[\hat{\theta}_1] = Var[\hat{\theta}_2]$, est un estimateur sans biais de θ de variance $v/2$. En moyennant deux estimateurs sans biais, on a divisé la variance par 2.

Le moindre apport de l'agrégation de modèles aux classifieurs stables peut s'interpréter en songeant à un résultat de Hansen et Salamon (1990), selon lesquels l'agrégation de p classifieurs, dont les erreurs sont indépendantes et ont un taux $< 50\%$, conduit à un taux d'erreur pour le modèle agrégé qui tend vers 0 quand p tend vers l'infini. Or, moins un classifieur sera stable, moins ses erreurs sur des échantillons différents seront corrélées. L'idée générale de

l'agrégation de modèles est de combiner des modèles en désaccord dans leurs prédictions, et pour cela de perturber l'apprentissage, soit en changeant l'échantillon d'apprentissage, ce qui est le plus courant, soit en conservant le même échantillon et en faisant varier les paramètres d'apprentissage.

Un algorithme particulier de bagging est celui appliqué à des arbres de décision avec introduction d'une sélection aléatoire des variables explicatives. Il permet d'éviter de voir apparaître toujours les mêmes variables. On a dans ce cas une double randomisation et on parle de forêts aléatoires (Breiman, 2001). Contrairement au simple bagging, on peut l'appliquer avec succès aux arbres limités à deux feuilles (souches ou " stumps "), sans voir apparaître des arbres utilisant si souvent les mêmes variables qu'ils en deviennent trop corrélés.

Dans une autre variante, Opitz et Maclin (1999) proposent l'agrégation de réseaux de neurones construits, non pas sur des échantillons différents, mais sur le même échantillon en ne faisant varier que les paramètres et la topologie de chaque réseau de base.

10.3 L'agrégation par boosting

Une nouvelle approche de l'agrégation de modèles est venue de l'apprentissage machine ("machine learning") et est due à Yoav Freund et Robert E. Schapire : il s'agit du boosting (Freund et Schapire, 1996). Contrairement au bagging qui s'appuie sur un processus purement aléatoire, le boosting est un processus adaptatif et souvent déterministe. Comme dans le bagging, on construit un ensemble de modèles dont on agrège ensuite les prédictions, mais ici :

- on n'utilise pas nécessairement des échantillons bootstrap mais plus souvent l'échantillon initial complet à chaque itération (sauf dans quelques versions des algorithmes AdaBoost et Arcing) ;
- chaque modèle est une version adaptative du précédent, l'adaptation consistant à augmenter le poids des individus précédemment mal classés tandis que le poids des bien classés n'augmente pas ;
- l'agrégation finale des modèles est réalisée par une moyenne de tous les modèles dans laquelle chacun est généralement (sauf dans l'algorithme Arcing) pondéré par sa qualité d'ajustement.

La deuxième de ces différences entre le principe du bagging et celui du boosting en entraîne une autre, d'un point de vue calculatoire : un algorithme de bagging peut être parallélisé, non un algorithme de boosting.

On peut donc considérer que le boosting concentre ses efforts sur les individus difficiles à modéliser et dont le comportement est plus malaisé à prédire. De ce fait, à une itération ayant mal classé des outliers succède une itération les classant bien mais classant du coup moins bien le reste de la population. L'itération suivante rétablit la balance, mais va à son tour mal classer certains outliers. On a ainsi un balancement au fil des itérations, dans lequel, contrairement au bagging, les modèles ne sont que localement (sur certains individus) et non globalement optimaux. C'est leur agrégation qui elle est globalement optimale, sauf en présence de bruit. Ce dernier point a déjà été relevé par Opitz et Maclin (1999) : la sensibilité du boosting au bruit et aux outliers fait que, si le boosting est globalement supérieur au bagging, il peut dans

certains cas lui être inférieur, voire même augmenter la variance du classifieur de base. La sensibilité du boosting au bruit a deux causes :

- le dispositif adaptatif du boosting met l'accent sur les individus difficiles à classer (en les pondérant plus ou en les choisissant plus souvent) qui par nature sont plutôt les outliers ;
- la pondération finale lors de l'agrégation des modèles peut contribuer au sur-apprentissage, plus qu'une agrégation sans pondération comme celle de l'Arcing.

Attention, en présence de bruit, l'erreur du modèle boosté augmente avec le nombre de modèles agrégés.

Il faut répondre à plusieurs questions quand on met en oeuvre un algorithme de boosting :

- Faut-il utiliser des échantillons bootstrap ou l'échantillon initial complet ?
- Quelle fonction d'erreur de classement utilise-t-on pour pondérer les individus (cela peut être le résidu de la déviance pour un modèle linéaire généralisé, ou $|Y_{\text{obs}} - Y_{\text{pred}}|$ pour un arbre de décision) ?
- Faut-il à chaque itération n'utiliser que l'erreur de l'itération précédente, ou la multiplier par l'erreur de toutes les itérations antérieures, ce qui peut avoir pour effet de " zoomer " excessivement sur les individus outliers mal classés ?
- Que fait-on des individus exagérément mal classés à l'itération i, faut-il borner leur erreur (par exemple en limitant à 2 le résidu de la déviance), leur interdire de participer à l'itération $i + 1$, ou ne rien faire ?
- Comment réaliser l'agrégation finale et quelle fonction de qualité d'ajustement utilise-t-on ? Faut-il prendre en compte tous les modèles ou écarter ceux qui s'ajustent trop mal ?

Le boosting est couramment utilisé quand le modèle de base est un arbre de décision, mais peut être valablement appliqué à d'autres modèles de base, surtout ceux qui sont instables. Il améliore souvent la robustesse des modèles (diminue leur variance), comme le fait le bagging, mais contrairement au bagging, il peut aussi permettre d'améliorer leur précision (diminuer leur biais). Autrement dit, le taux de bon classement et l'aire sous la courbe ROC augmentent sur l'échantillon de test (précision), et l'écart de ces mesures entre apprentissage et test diminue (robustesse). Cependant, le boosting, en raison des réglages susmentionnés à effectuer, est plus délicat à mettre en oeuvre que le bagging.

Dans le cas d'un arbre, on obtient déjà des résultats non triviaux avec un arbre de base à seulement deux feuilles ("stump"), même si la valeur recommandée pour le nombre de feuilles de l'arbre de base est comprise entre 4 et 8, ou égale à la racine carrée du nombre de variables explicatives. On trouve dans (Hastie et al., $2001, p.302$) l'exemple d'un "stump" dont le taux d'erreur (en test) est de 46% (à peine mieux qu'une prédiction au hasard) mais descend à $12,2\%$ après 400 itérations de boosting, alors qu'un arbre de décision classique a dans ce cas un taux d'erreur de 26%. Il est à noter que, comme les forêts aléatoires et contrairement au simple bagging, le boosting est efficace sur les "stumps", ce qui est une autre dissemblance entre ces deux techniques d'agrégation.

Plusieurs algorithmes basés sur le principe du boosting ont vu le jour : entre autres le Discrete AdaBoost, le Real AdaBoost, le LogitBoost, le Gentle AdaBoost et l'Arcing (Adaptative Resampling and Combining). L'Arcing (Breiman, 1996b) présente cette particularité que

l'agrégation finale des modèles se fait en donnant le même poids à tous les modèles, quelle que soit leur qualité d'ajustement. À chaque itération, le poids d'un individu est proportionnel à la somme de 1 et des puissances $4^{\text{èmes}}$ des nombres d'erreurs de classification des itérations précédentes. Cette formule est simple, efficace et facile à implémenter. Breiman a choisi la puissance $4^{\text{ème}}$ de façon empirique après avoir testé plusieurs valeurs.

Voici le principe du Real AdaBoost (Schapire et Singer, 1998). On a un échantillon de taille n, et, pour chaque $i = 1, \ldots, n$, un vecteur x_i de variables explicatives et une valeur à prédire $y_i \in \{-1, +1\}$. On cherche à construire un modèle $F(x)$ à partir des observations $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ et en procédant à M itérations. On commence par initialiser tous les poids $w_i = 1/n$. On répète pour $m = 1$ à M :

- calculer la probabilité $p_m(x) = P_w(y = 1|x)$ sur l'échantillon pondéré par $w = (w_i)$;
- poser $f_m(x) = \frac{1}{2} \log \left(\frac{p_m(x)}{1-p_m(x)} \right)$;
- mettre à jour les poids $w_i = w_i \exp(-y_i f_m(x_i))$ et les renormaliser pour que $\sum w_i = 1$.

Le résultat cherché $F(x)$ est le signe de $\sum_{m=1}^{M} f_m(x)$.

10.4 Derniers exemples et récapitulation

Opitz et Maclin (1999) ont testé le bagging et le boosting de réseaux de neurones et d'arbres de décision sur 23 jeux de données. Leur article est intéressant à plusieurs titres, et notamment car les réseaux de neurones sont plus rarement évoqués dans la littérature sur l'agrégation de modèles. À cela, deux raisons au moins : tout d'abord, les temps de traitement plus longs des réseaux de neurones sont problématiques quand on travaille sur des échantillons multiples ; ensuite, les performances des réseaux de neurones sont assez sensibles à des paramètres qui ne sont pas toujours faciles à choisir. Cependant, les réseaux de neurones sont intéressants par leur large spectre d'applications et par leurs performances parfois supérieures à celles des arbres de décision.

Sur le nombre de modèles à agréger, selon ces auteurs il dépend de la méthode :

- une dizaine suffit pour le bagging et le boosting des réseaux de neurones, et le bagging des arbres de décision ;
- pour le boosting des arbres de décision, il faut compter 25 modèles pour obtenir l'essentiel de la baisse de l'erreur.

De leur côté, Bauer et Kohavi (1999) étudient en détail sur 12 jeux de données le bagging et le boosting d'arbres de décision mais non de réseaux de neurones. Ils s'intéressent en revanche au classifieur naïf de Bayes, qui, en tant que classifieur très stable, montre qu'il ne faut pas s'attendre à ce que le bagging ou le boosting en fasse réellement baisser la variance (même si le boosting en diminue le biais).

Néanmoins, s'intéressant à un classifieur stable, Decourt et Despiegel (2003) ont appliqué des stratégies de bagging et de boosting (avec rééchantillonnage) à un modèle de base qui est une analyse discriminante de Fisher, sur des données d'assurance automobile. Ils ont mesuré le taux de bon classement (voir tableau 4) du modèle de base et des modèles résultant du bagging et du boosting à 10 et 500 itérations. Le bénéfice du bagging est mince, et la robustesse semble même pâtir d'un trop grand nombre d'itérations (ce n'est pas toujours le cas). En revanche,

le gain du boosting est net, tant en précision qu'en robustesse, et cela même en dix itérations seulement.

		Bien classés (jeu de test)	Écart entre entraînement et test
Modèle de base	Modèle de base	69,76 %	7,24%
Bagging	10	69,84%	6,29%
Bagging	500	70,01%	6,37%
Boosting	10	72,74%	3,14%
Boosting	500	73,90%	2,35%

TAB. 4 – *Taux de bon classement selon la méthode d'agrégation de modèles.*

Dans d'autres exemples que nous avons testés en boostant 500 fois une régression logistique, sur un échantillon indépendant de l'échantillon initial le modèle agrégé par boosting se situait à peu près au niveau d'AUC du modèle agrégé par bagging.

Qu'il s'agisse du bagging ou du boosting, ces techniques d'agrégation de modèles permettent d'améliorer parfois très nettement la qualité des prédictions, notamment en termes de variance. Le revers de la médaille est la perte de lisibilité des arbres de décision, dans certains cas aussi (arbres de décision et réseaux de neurones) la nécessité de stocker tous les modèles afin de pouvoir les combiner (tandis qu'un modèle logistique ou discriminant "moyen" est aussi concis que le modèle de base), et de façon générale l'importance du temps de calcul sur un ordinateur, qui peut devenir prohibitif quand on dépasse plusieurs centaines d'itérations. Le poids des avantages est toutefois tel que ces techniques font l'objet de nombreux travaux théoriques et commencent à apparaître dans les logiciels commerciaux. Nous avons résumé leurs grandes caractéristiques dans le tableau 5.

11 Conclusion : Influence relative des données et des méthodes

Nous conclurons en disant que les performances d'un modèle prédictif dépendent finalement beaucoup plus des données que de la méthode de modélisation employée, ce qui rend primordiales les étapes préliminaires d'exploration, d'analyse et de recodage des données, et même celles, dévolues à l'informatique, de collecte et de stockage des données. Ce travail, même s'il représente un coût en développement informatique, est la condition de la mise au point de modèles prédictifs performants.
Alors que le gain d'une méthode par rapport à une autre se mesure souvent en millièmes d'aire sous la courbe ROC, l'ajout d'une nouvelle variable devenue disponible dans l'entrepôt de données peut apporter un gain de plusieurs centièmes.

Enfin, un troisième facteur intervient dans la performance du modèle, plus que la méthode de modélisation et autant que les données disponibles : la problématique étudiée. Ainsi, il est intrinsèquement plus facile de prédire le risque d'impayés que l'appétence à l'achat : cette dernière est moins bien cernée par les données couramment disponibles car elle est plus liée à la psychologie de l'individu.

BAGGING	BOOSTING
Caractéristiques	**Caractéristiques**
Le bagging est aléatoire	Le boosting est adaptatif et généralement déterministe
On utilise des échantillons bootstrap	On utilise généralement l'échantillon initial complet
Chaque modèle produit doit être performant sur l'ensemble des observations	Chaque modèle produit doit être performant sur certaines observations ; un modèle performant sur certains outliers sera moins performant sur les autres individus
Dans l'agrégation, tous les modèles ont le même poids	Dans l'agrégation, les modèles sont généralement pondérés selon leur qualité d'ajustement (sauf l'Arcing)
Avantages et inconvénients	**Avantages et inconvénients**
Technique de réduction de la variance par moyenne de modèles	Peut diminuer la variance et le biais du classifieur de base Mais la variance peut augmenter avec un classifieur de base stable
Perte de lisibilité quand le classifieur de base est un arbre de décision	Perte de lisibilité quand le classifieur de base est un arbre de décision
Inopérant sur les " stumps " (sauf double-randomisation des forêts aléatoires)	Efficace sur les " stumps "
Possibilité de paralléliser l'algorithme	Algorithme séquentiel ne pouvant être parallélisé
Pas de sur-apprentissage : supérieur au boosting en présence de " bruit "	Risque de sur-apprentissage mais globalement supérieur au bagging sur des données non bruitées (l'Arcing est moins sensible au bruit)
En résumé, le bagging fonctionne plus souvent que le boosting. . .	. . . mais quand le boosting fonctionne, il fonctionne mieux que le bagging.

TAB. 5 – *Comparatif du bagging et du boosting*

Néanmoins, nonobstant la prépondérance des données et de la problématique, il est presque toujours possible d'obtenir un modèle plus précis et surtout plus robuste : soit par partition de modèles après une préclassification, soit (nous n'en avons pas parlé ici) par combinaison de plusieurs modèles obtenus par des techniques différentes appliquées au même échantillon, soit par combinaison (l'agrégation exposée plus haut) de plusieurs modèles obtenus par la même technique appliquée à plusieurs échantillons, ce qui peut se présenter ainsi :

Appliquer :	**Le même échantillon**	**Des échantillons différents**
La même technique	Modèle simple	Agrégation de modèles
Des techniques différentes	Combinaison de modèles	Mélange (*)

(*) Il pourrait s'agir d'une suite d'échantillons bootstrap auxquels seraient chaque fois appliqués un arbre de décision et un réseau de neurones.

Remerciements

L'auteur remercie les referees pour leur lecture attentive, leurs remarques pertinentes et leurs suggestions utiles.

Références

Bauer, E. and R. Kohavi (1999). *An empirical comparison of voting classification algorithms : Bagging, boosting, and variants*. Machine Learning 36 (1-2), 105-139.

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

Breiman, L. (1996a). *Bagging Predictors*. Machine Learning 26 :2, 123-140.

Breiman, L. (1996b). *Bias, variance, and arcing classifiers*. Tech. rep. 460, UC-Berkeley, Berkeley, CA.

Breiman, L. (2001). *Random Forests*. Machine Learning 45 :1, 5-32.

Crook, J. and J. Banasik (2002). *Does reject inference really improve the performance of application scoring models ?* Working Paper 02/3, Credit Research Centre, University of Edinburgh.

Decourt, O. et N. Despiegel (2003). *Le bootstrap expliqué par l'exemple*. Paris : Club SAS.

Dougherty, J., R. Kohavi, and M. Sahami (1995). *Supervised and unsupervised discretization of continuous features*. In Proceedings of the Twelfth International Conference on Machine Learning. Los Altos, CA : Morgan Kaufman.

Efron, B. (1979). *Bootstrap methods : another look to Jackknife*. Ann. Statist. 7, 1-26.

Efron, B. and R. J. Tibshirani (1997). *Improvements on cross-validation : The .632+ bootstrap method*. J. of the American Statistical Association, 92, 548-560.

Fayyad, U. M. and K. B. Irani (1993). M*ulti-interval discretization of continuous-valued attributes for classification learning*. In Proc. 13th International Joint Conference on Artificial Intelligence, 1022-1027. Los Altos, CA : Morgan Kaufmann Publishers

Freund, Y. and R. E. Schapire (1996). *Experiments with a new boosting algorithm*. Machine Learning : Proceedings of the Thirteenth International Conference on Machine Learning, 148-156.

Furnival, G. M. and R. W. Wilson (1974). *Regressions by Leaps and Bounds*. Technometrics, 16, 499-511.

Hand, D. J. (2005). *Classifier technology and the illusion of progress*. London : Technical Report. Imperial College, Department of Mathematics.

Hansen, L. and P. Salamon (1990). *Neural network ensembles*. IEEE Transactions on Pattern Analysis and Machine Intelligence 12, 993-1001.

Hastie, T., R. Tibshirani, J.-H. Friedman (2001). *The Elements of Statistical Learning : Data Mining, Inference and Prediction*. Berlin : Springer Series in Statistics, Springer Verlag.

Holte, R. C. (1993). *Very simple classification rules perform well on most commonly used datasets*. Machine Learning, 11, 63-91.

Hsia, D. C. (1978). *Credit scoring and the Equal Credit Opportunity Act*. The Hastings Law Journal, 30, November : 371-448.

Joanes, D. N. (1993/4). *Reject inference applied to logistic regression for credit scoring*. IMA Journal of Mathematics Applied in Business and Industry, 5 : 35-43.

Kerber, R. (1992). *ChiMerge : Discretization of numeric attributes*. In Proc. Tenth National Conference on Artificial Intelligence, 123-128. MIT Press.

Opitz, D. and R. Maclin (1999). *Popular Ensemble Methods : An Empirical Study*. Journal of Artificial Intelligence Research 11, 169-198.

Quinlan, J. R. (1993). *Programs for Machine Learning*. Los Altos, CA : Morgan Kaufman.

Richeldi, M. and M. Rossotto (1995). *Class-Driven statistical discretization of continuous attributes*. In Machine Learning : ECML-95 Proceedings European Conference on Machine Learning, Lecture Notes in Artificial Intelligence 914, 335-338. Springer Verlag.

Saporta, G. and N. Niang (2006). *Correspondence analysis and classification*. In J. Blasius and M. Greenacre (editors), Multiple Correspondence Analysis and Related Methods, Chapman and Hall.

Schapire, R. and Y. Singer (1998). *Improved boosting algorithms using confidence-rated prediction*. In Proceedings of the Eleventh Annual Conference on Machine Learning.

Tufféry, S. (2007). *Data mining et Statistique décisionnelle*. Paris : Éditions Technip.

Van der Putten, P. and M. van Someren (eds) (2000). *CoIL Challenge 2000 : The Insurance Company Case*. Technical Report 2000-09, Leiden Institute of Advanced Computer Science.

Summary

In this paper, we show that the performance of a predictive model depends generally more on the quality of the data and on the care taken with their preparation and with their selection, than on the method of modelling. The performance difference between two methods is often negligible compared with the uncertainties resulting from the definition of the dependent variable and from the fact that the design set is possibly not representative of future data. However, the resampling and ensemble classifiers approach can lead to a dramatic reduction in the vari-

ance and the bias of some models. Good results can be also obtained simply by clustering the design sample and building up a model on each cluster.