Optimisation directe des poids de modèles dans un prédicteur Bayésien naïf moyenné

Romain Guigourès*, Marc Boullé*

*Orange Labs
2 avenue Pierre Marzin
22307 Lannion Cedex

{romain.guigoures, marc.boulle}@orange-ftgroup.com

Résumé. Le classifieur Bayésien naïf est un outil de classification efficace en pratique pour de nombreux problèmes réels, en dépit de l'hypothèse restrictive d'indépendance des variables conditionnellement à la classe. Récemment, de nouvelles méthodes permettant d'améliorer la performance de ce classifieur ont vu le jour, sur la base à la fois de sélection de variables et de moyennage de modèles. Dans cet article, nous proposons une extension de la sélection de variables pour le classifieur Bayésien naïf, en considérant un modèle de pondération des variables utilisées et des algorithmes d'optimisation directe de ces poids. Les expérimentations confirment la pertinence de notre approche, en permettant une diminution significative du nombre de variables utilisées, sans perte de performance prédictive.

1 Introduction

Le classifieur naïf Bayésien est un outil largement utilisé dans les problèmes de classification supervisée. Il a pour avantage de se montrer efficace pour de nombreux jeux de données réels (Hand et Yu, 2001). Cependant, l'hypothèse naïve d'indépendance des variables peut, dans certains cas, dégrader les performances du classifieur.

Ainsi, des méthodes proposant de réaliser de la sélection de variables ont vu le jour (Langley et Sage, 1994). Elles consistent en la mise en place d'heuristiques d'ajout et de suppression de variables afin de sélectionner le meilleur sous-ensemble de variables maximisant un critère et donc les performances du classifieur, selon une approche wrapper (Guyon et Elisseeff, 2003). Il a été montré par Boullé (2007) que moyenner un grand nombre de classifieurs Bayésiens naïfs sélectifs, réalisés avec différents sous-ensembles de variables, revenait à ne considérer qu'un seul modèle avec une pondération sur les variables.

Dans cet article, on se propose de trouver un modèle optimal par optimisation directe des poids des variables. On introduit un critère, basé sur la vraisemblance d'un modèle, fonction continue d'un vecteur de poids. Une descente de gradient est ensuite utilisée pour l'optimisation, le critère étant continu et dérivable sur l'ensemble de définition du vecteur de poids des variables. Une méthode de régularisation est introduite pour minimiser le nombre de variables

sélectionnées sans dégrader les performances du classifieur. Le problème de ce type d'optimisation est qu'il existe des optima locaux, des multistarts sont alors réalisés afin de trouver un optimum satisfaisant.

La deuxième partie de ce papier introduit les notations utilisées tout au long de l'article et revient sur les principes des classifieurs Bayésiens naïfs et les différentes approches basées sur la pondération des variables. La troisième partie définit le critère optimisable par descente de gradient, ainsi qu'une pénalisation permettant de maximiser les performances du classifieurs avec un minimum de variables. Puis des expérimentations sont présentées afin de montrer les performances de l'approche. Enfin, une conclusion fera le bilan des différents points évoqués dans cet article.

2 Classifieurs Bayesiens naïfs et SNB

Notations : Soient $X = \{X_1, X_2, ..., X_K\}$ un vecteur de K variables explicatives et une variable de classe Y ayant J valeurs $\{y_1, y_2, ..., y_J\}$. On note $D = \{D_1, D_2, ..., D_N\}$ la base de données contenant N instances, identifiées de la façon suivante : $D_n = (x_1^{(n)}, ..., x_K^{(n)}, y^{(n)})$ et simplifié en $D_n = (x^{(n)}, y^{(n)})$ pour une meilleure lisibilité.

Un modèle M_m est décrit par un vecteur de K poids $W = \{w_1, w_2, ..., w_K\}$. En effet chaque poids est associé à une variable tel que w_k pondère X_k dans un modèle M_m.

Notons $P(y_j)$ la probabilité à priori que la classe Y vaille y_j et $P(X_k|y_j)$ la probabilité conditionnelle de la $kième$ variable connaissant la valeur de la classe. Ces deux probabilités sont considérées comme initialement connues grâce à un pré-traitement par discrétisation ou groupement de valeurs.

Classifieur Bayesien naïf : Le classifieur Bayesien prédit la classe y_j pour chacune des instances tel que soit maximale la probabilité conditionnelle $P(y_j|X)$. L'hypothèse naïve dans un classifieur bayésien est de considérer indépendantes les variables explicatives conditionnellement aux classes (Duda et al., 2000). On obtient alors

$$P(y_j|X) = \frac{P(y_j)\prod_{k=1}^{K} P(X_k|y_j)}{\sum_{i=1}^{J} P(y_i)\prod_{k=1}^{K} P(X_k|y_i)}$$

Classifieur SNB (Selective Naive Bayes) : Bien que le classifieur Bayesien naïf soit efficace dans de nombreux cas, l'hypothèse d'indépendance des variables conditionnellement à la classe peut, dans certains cas, déteriorer les performances du classifieur. Le classifieur SNB propose de sélectionner un sous-ensemble de variables afin de maximiser les performances. On réduit ainsi le biais apporté par l'hypothèse naïve du classifieur (Langley et Sage, 1994). Plus formelement, cela revient à fixer une pondération booléenne $W = \{w_1, w_2, ..., w_K\} \in \{0, 1\}^K$ sur chacune des probabilités conditionnelles des variables connaissant la classe.

$$P(y_j|X) = \frac{P(y_j)\prod_{k=1}^{K} P(X_k|y_j)^{w_k}}{\sum_{i=1}^{J} P(y_i)\prod_{k=1}^{K} P(X_k|y_i)^{w_k}}$$

Plusieurs méthodes ont été développées en exploitant des heuristiques proposant de faire de l'ajout ou de la supression de variables en optimisant un critère tel que la précision ou l'aire sous la courbe de ROC.

Approche MAP (Maximum A Posteriori) : Cette approche propose de déterminer le meilleur sous-ensemble de variables en maximisant la vraisemblance, pénalisée par un a priori hiérarchique sur les paramètres de sélection du nombre de variables puis sur les sous-ensembles de variables. (Boullé, 2007).

Approche BMA (Bayesian Model Averaging) : Alors que l'approche MAP permet de déterminer le modèle le plus probable a posteriori, l'approche BMA, quant à elle, se propose de tenir compte de tous les modèles et de les moyenner, en les pondérant par leur probabilité a posteriori, afin d'obtenir un modèle plus performant (Hoeting et al., 1999). Il a été démontré par Boullé (2007) que moyenner un grand nombre de classifieurs sélectifs revenait à élaborer un seul classifieur dans lequel chaque variable aurait un poids compris entre 0 et 1.

Approche CMA (Compression Model Averaging) : Cette technique est proche de la précédente, à la différence qu'elle se base sur le taux de compression pour moyenner les modèles, ce qui revient à un lissage logarithmique des probabilités a posteriori (Boullé, 2007). Cette approche aboutit à un autre schéma de pondération des variables, dont les performances surpassent significativement celles de l'approche BMA.

3 Optimisation directe des poids

Le but est de maximiser la vraisemblance d'un modèle en optimisant directement ses poids. On verra par la suite comment améliorer le critère d'optimisation afin de réduire au maximum le nombre de variables.

Descente de gradient : Reprenons le classifieur SNB dont le principe est de minimiser le coût d'un modèle défini par un vecteur de poids, booléens dans l'approche MAP, et compris entre 0 et 1 dans le cas du moyennage de modèles. Ici, on considère le coût d'un modèle comme une fonction à K variables définies sur $[0,1]^K$, ces variables correspondant aux poids.

Soit C la fonction objectif à optimiser, logarithme négatif de la vraisemblance du modèle M_m décrit par le vecteur de poids W. C est fonction de $W = \{w_1, w_2, ..., w_K\}$ $\in [0,1]^K$, dérivable sur son ensemble de définition.

$$C(W) = -\sum_{n=1}^{N} logP_m(y^{(n)}|x^{(n)})$$

$$C(W) = -\sum_{n=1}^{N}\left(logP(y^{(n)}) + \sum_{k=1}^{K} w_k logp(x_k^{(n)}|y^{(n)}) - log\sum_{j=1}^{J} P(y_j)\prod_{k=1}^{K} p(x_k^{(n)}|y_j)^{w_k}\right)$$

Soit t l'indice marquant l'itération, on notera $W^{(t)} = \{w_1^{(t)}, w_2^{(t)}, ..., w_K^{(t)}\}$ la valeur du vecteur de poids au temps t. On va itérer jusqu'à optimiser la fonction C (Duda et al., 2000), en utilisant l'algorithme de descente de gradient :

$$W^{(t+1)} = W^{(t)} - \eta^{(t)}\nabla C(W^{(t)})$$

avec $\eta^{(t)} = \{\eta_1^{(t)}, \eta_2^{(t)}, ..., \eta_K^{(t)}\}$ le vecteur du pas pour chacune des variables au temps t. Le pas de chacune des variables est initialisé à 10^{-2}, puis ce pas sera adaptatif et déterminé par

l'algorithme RPROP (Riedmiller, 1994) :

$$\eta_\gamma^{(t)} = \begin{cases} \eta_\gamma^{(t-1)} \times 1.2 & \text{si } \frac{\partial C}{\partial w_\gamma}(W^{(t-1)}) \times \frac{\partial C}{\partial w_\gamma}(W^{(t)}) > 0 \\ \eta_\gamma^{(t-1)} \times 0.5 & \text{sinon} \end{cases}$$

Une contrainte est ajoutée, elle consiste à réduire l'espace d'optimisation à chaque itération. En effet, si une variable prend une valeur nulle et que le gradient du critère à optimiser selon cette même variable reste négatif à l'itération suivante, alors elle sera supprimée du modèle. Ceci permet de converger vers une solution optimale plus rapidement et avec un nombre de variables réduit. La condition d'arrêt de l'algorithme est fixée par la convergence des poids à $\frac{1}{N}$ près. Cet algorithme a une complexité en $O(KN)$ par passe, donc linéaire par rapport au nombre d'instances et de variables.

Méthode de régularisation : Bien que la descente de gradient soit efficace pour optimiser la vraisemblance, on aura tendance à voir une répartition homogène des poids sur l'ensemble des variables du modèle, et finalement une sélection de variables plutôt inefficace. En étudiant de manière plus approfondie l'évolution du critère en fonction des poids, on se rend compte qu'il existe plusieurs optima locaux. Les solutions optimales qui nous intéressent sont celles présentant le moins de variables. On va donc introduire une probabilité a priori sur la distribution des poids favorisant les fortes pondérations et les pondérations nulles. Soient $P(M)$ la distribution a priori du modèle et $P(D|M)$ la vraisemblance optimisée précédemment.

$$P(M|D) = P(M)P(D|M)$$

On choisit de définir $P(M) = f(W)$ comme la moyenne de deux lois normales centrées en 0 et 1, projetée sur l'intervalle $[0, 1]$.

$$f(W) = \frac{2}{erf(1)\sigma\sqrt{2\pi}} \left(\frac{1}{2} e^{-\frac{1}{2}\left(\frac{W}{\sigma}\right)^2} + \frac{1}{2} e^{-\frac{1}{2}\left(\frac{W-1}{\sigma}\right)^2} \right)$$

avec $erf(x)$, la fonction d'erreur Gaussienne en x et σ la variance devant être importante afin de ne pas déteriorer le critère de base. On définit alors une nouvelle fonction D à optimiser.

$$D(W) = C(W) - logf(W)$$

Multistarts : Le critère précédent est conçu de manière à réduire le nombre de variables, mais l'ajout de la pénalisation crée des optima locaux. Il n'est ainsi pas garanti que la selection de variables ait été optimale. C'est pourquoi on propose de réaliser des multistarts. Le principe est de relancer l'algorithme plusieurs fois sur les données, en réinitialisant les poids des variables non nuls de manière aléatoire.

4 Expérimentations

Présentations des données et conditions expérimentales : Les expérimentations sont menées sur dix validations croisées. Les données choisies seront de différents types, et proviennent soit de l'UCI (Frank et Asuncion, 2010) soit du challenge KDD 2009. Le pré-traitement des données est effectué par l'approche MODL. (Boullé, 2006)

Trois algorithmes sont comparés. Le premier est un classifieur bayesien naïf classique (NB). Le second est un classifieur obtenu par moyennage de modèles basé sur la Compression (CMA). Et enfin, la méthode d'optimisation directe des poids par descente de gradient (Grad) pour laquelle on réalise 5 starts.

Données	Nombre de variables		Nombre de valeurs de classe	Nombre d'instances
	Numériques	Catégorielles		
Satimage	36	0	6	6435
Segmentation	19	0	7	2310
SickEuthyroid	7	18	2	3163
Small KDD 09	190	41	2	50000
Spam	57	0	2	4307
Thyroid	21	0	3	7200
Vehicle	18	0	4	846
Waveform	21	0	3	5000

TAB. 1 – *Caractéristiques des jeux de données utilisés pour les expérimentations*

Résultats : La réalisation d'une descente de gradient nécessite, en fonction du jeu de données, entre 20 et 60 passes par start avant la convergence d'un vecteur de poids. Le premier start élimine la majorité des variables, les suivants permettent d'affiner la sélection.

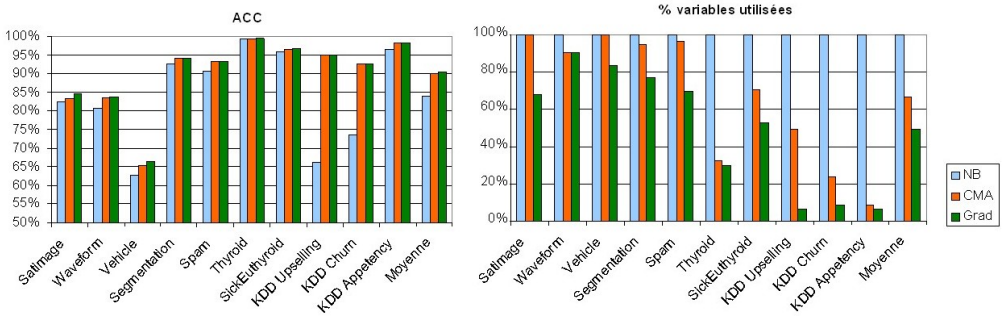

FIG. 1 – *Précision en test (ACC) et pourcentage de variables embarquées par le classifieur (NB = Bayésien naïf, CMA = Compression based averaging, Grad = Descente de gradient)*

Globalement, d'après la Figure 1, les performances prédictives en test sont équivalentes pour la méthode SNB et pour l'optimisation directe des poids par descente de gradient et meilleures qu'avec un simple classifieur Bayesien naïf. Le plus gros avantage que l'on puisse tirer d'une optimisation directe des poids est la diminution du nombre de variables de variables. En effet alors que l'approche SNB(CMA) permettait déjà une bonne sélection des variables, l'optimisation directe est encore plus efficace, diminuant de façon importante leur nombre.

5 Conclusion

Dans cet article, une méthode d'optimisation directe des poids des variables dans un classifieur bayesien naïf a été proposée, celle-ci consistant à minimiser un critère basé sur la vrai-

semblance d'un modèle dont les variables sont pondérées par des réels compris entre 0 et 1. Une distribution des poids a priori a été introduite afin de rendre la sélection de variables plus efficace. L'optimisation par descente de gradient possède l'avantage d'être algorithmiquement peu coûteuse. D'autre part, elle peut être adaptée aux bases de données de grande taille et se faire sur un nombre réduit d'instances (gradient stochastique) d'après Bottou et Le Cun (2005). La pénalisation bayésienne du critère à optimiser permet, quant à elle, de sélectionner un nombre très réduit de variables sans déteriorer les performances des approches SNB discutées dans (Boullé, 2007). Les résultats expérimentaux présentent des modèles bien plus parcimonieux et tout aussi performants que les modèles réalisés par sélection de variables et moyennage de modèles, ce qui améliore d'une part l'interprétabilité des modèles, d'autre part, l'efficacité du deploiement.

Références

Bottou, L. et Y. Le Cun (2005). On-line learning for very large data sets : Research articles. *Appl. Stoch. Model. Bus. Ind. 21*(2).

Boullé, M. (2006). Modl : A bayes optimal discretization method for continuous attributes. *Mach. Learn. 65*(1), 131–165.

Boullé, M. (2007). Compression-based averaging of selective naive bayes classifiers. *Journal of Machine Learning Research 8*, 1659–1685.

Duda, R. O., P. E. Hart, et D. G. Stork (2000). *Pattern Classification (2nd Edition)* (2 ed.). Wiley-Interscience.

Frank, A. et A. Asuncion (2010). UCI machine learning repository.

Guyon, I. et A. Elisseeff (2003). An introduction to variable and feature selection. *J. Mach. Learn. Res. 3*, 1157–1182.

Hand, D. J. et K. Yu (2001). Idiot's bayes - not so stupid after all ? *International Statistical Review 69*(3), 385–398.

Hoeting, J. A., D. Madigan, A. E. Raftery, et C. T. Volinsky (1999). Bayesian model averaging : A tutorial. *Statistical Science 14*(4), 382–401.

Langley, P. et S. Sage (1994). Induction of selective bayesian classifiers. In *Conference on uncertainty in artificial intelligence*, pp. 399–406. Morgan Kaufmann.

Riedmiller, M. (1994). Rprop - description and implementation details.

Summary

The naive Bayes classifier is very effective on many datasets in which each variable is assumed independent compared to each other. Approaches improving the performance by weighting the variables have been developed lately. In this paper, we describe a method that directly optimizes the weights and maximizes the classifier performance. A regularization technique is also introduced in order to make an effective feature selection. Experimental results are presented and discussed so that the efficiency of this approach could be proved.