Une méthode de classification supervisée sans paramètre pour l'apprentissage sur les grandes bases de données

Marc Boullé*

*Orange Labs
2 avenue Pierre Marzin
22300 Lannion
marc.boulle@orange-ftgroup.com,
http://www.francetelecom.com/fr/groupe/rd/

Résumé. Dans ce papier, nous présentons une méthode de classification supervisée sans paramètre permettant d'attaquer les grandes volumétries. La méthode est basée sur des estimateurs de densités univariés optimaux au sens de Bayes, sur un classifieur Bayesien naïf amélioré par une sélection de variables et un moyennage de modèles exploitant un lissage logarithmique de la distribution a posteriori des modèles. Nous analysons en particulier la complexité algorithmique de la méthode et montrons comment elle permet d'analyser des bases de données nettement plus volumineuses que la mémoire vive disponible. Nous présentons enfin les résultats obtenu lors du récent PASCAL Large Scale Learning Challenge, où notre méthode a obtenu des performances prédictives de premier plan avec des temps de calcul raisonnables.

1 Introduction

La phase de préparation des données est particulièrement importante dans le processus data mining (Pyle, 1999). Elle est critique pour la qualité des résultats, et consomme typiquement de l'ordre de 80% du temps d'une étude data mining. Dans le cas d'une entreprise comme France Télécom, le data mining est appliqué dans de nombreux domaines : marketing, données textuelles, données du web, classification de trafic, sociologie, ergonomie. Les données disponibles sont hétérogènes, avec des variables numériques ou catégorielles, des variables cibles comportant de multiples classes, des valeurs manquantes, des distributions bruitées et déséquilibrées, des nombres de variables et d'instances pouvant varier sur plusieurs ordres de grandeurs. Ce contexte industriel impose des contraintes telles que le potentiel des données collectées dans les systèmes d'information est largement sous-utilisé. Cette situation s'aggrave années après années, suite à des vitesses d'évolution divergentes des capacités des systèmes d'information, en augmentation très rapide pour le stockage et "seulement" rapide pour le traitement, des capacités de modélisation des méthodes d'apprentissage statistique, en progression lente, et de la disponibilité des analystes de données, au mieux constante. Dans ce contexte, les solutions actuelles sont impuissantes à répondre à la demande rapidement croissante de l'utilisation de techniques data mining. Les projets, en surnombre, sont abandonnés ou traités

sous-optimalement. Pour résoudre ce goulot d'étranglement, nous nous intéressons ici au problème de l'automatisation de la phase de préparation des données du processus data mining. Dans cet article, nous présentons une méthode [1] qui vise à automatiser les phases de préparation des données et de modélisation du processus data mining, y compris dans le cas de la très grande volumétrie.

Après avoir présenté dans la partie 2 les principes de notre méthode, nous introduisons dans la partie 3 les algorithmes permettant de traiter des bases nettement plus volumineuses que la mémoire vive disponible. La partie 4 présente les résultats obtenus lors du Large Scale Learning challenge. Enfin, la partie 5 présente un résumé et des perspectives.

2 Une méthode de classification entièrement automatique

Notre méthode, introduite dans (Boullé, 2007), étend le classifieur Bayesien naïf grâce à une estimation optimale des probabilité conditionnelles univariées, à une sélection de variables selon une approche Bayesienne et un moyennage de modèle exploitant un lissage logarithmique de la distribution a posteriori des modèles.

2.1 Discrétisation optimale

Le classifieur Bayesien naif a démontré son efficacité sur de nombreuses applications réelles (Langley et al., 1992; Hand et Yu, 2001). Ce classifieur, qui suppose que les variables explicatives sont conditionnellement indépendantes, nécessite uniquement l'estimation des probabilités conditionnelles univariées. Les études menées dans la littérature (Liu et al., 2002) ont démontré l'intérêt des méthodes de discrétisation pour l'évaluation de ces probabilités conditionnelles. Dans l'approche MODL (Boullé, 2006), la discrétisation supervisée est posée sous la forme d'un problème de sélection de modèle, résolu selon une approche Bayesienne. Un espace des modèles de discrétisation est défini, en prenant pour paramètres le nombre d'intervalles, les bornes des intervalles et la distribution des classes par intervalle. Une distribution a priori est proposée sur cet espace de modèles, en exploitant la hiérarchie des paramètres, avec un choix uniforme à chaque étage de la hiérarchie, et des distributions multinomiales de classes par intervalles supposées indépendantes entre elles. Le meilleur modèle de discrétisation est choisi selon une approche MAP (maximum a posteriori), qui consiste a maximiser la probabilité $p(Model|Data)$ d'un modèle connaissant les données. En se basant sur la définition explicite de l'espace de modélisation et sur la distribution a priori des modèles, la formule de Bayes permet d'obtenir une expression analytique exacte pour l'évaluation de la probabilité a posteriori d'un modèle de discrétisation. Des heuristiques d'optimisation efficaces sont mises en oeuvre pour rechercher le meilleur modèle de discrétisation.

Le cas des variables descriptives catégorielles est traité selon la même approche (Boullé, 2005), en se basant sur une famille de modèles d'estimation de densité conditionnelle qui partitionne l'ensemble des valeurs descriptives en groupes de valeurs.

[1] L'outil est disponible en shareware sur http ://perso.rd.francetelecom.fr/boulle/

2.2 Sélection de variables selon une approche Bayesienne

L'hypothèse naive d'indépendance peut dégrader les performances prédictives si elle n'est pas respectée. Pour éviter le problème des variables fortement corrélées, le classifieur Bayesien naif sélectif (Langley et Sage, 1994) exploite une approche enveloppe (Kohavi et John, 1997) pour sélectionner un sous-ensemble de variables de façon à optimiser le taux de bonne classification. Bien que cette méthode améliore notablement les performances sur des jeux de données de petite taille, elle n'est pas applicable dans le cas de la grande volumétrie, pour des jeux de données comportant des centaines de milliers d'instances et des milliers de variables. Le problème provient à la fois de l'algorithme d'optimisation, dont la complexité est quadratique avec le nombre de variables, et du critère d'évaluation d'une sélection qui est sujet au sur-apprentissage. Dans (Boullé, 2007), le problème de sur-apprentissage est abordé suivant une approche Bayesienne, où le meilleur modèle de sélection de variable est le modèle MAP. Les paramètres de modélisation sont le nombre de variables et le sous-ensemble de variables sélectionnées, en utilisant à nouveau un a priori hiérarchique. La vraisemblance conditionnelle des modèles dérive des probabilités conditionnelles prédites par le classifieur Bayesien naif. On obtient alors un calcul exact de la probabilité a posteriori des modèles. Des algorithmes d'optimisation de complexité super-linéaire en nombre de variables et d'instances sont utilisés, sur la base d'heuristiques stochastiques d'ajout ou surpression de variables.

2.3 Moyennage de modèles par taux de compression

Le moyennage de modèles a été appliqué avec succès dans le cas du bagging (Breiman, 1996), qui exploite un ensemble de classifieurs appris sur des ensembles ré-échantillonnés. A l'opposé de cette approche où chaque classifieur à le même poids dans le classifieur moyenné, le moyennage Bayesien de modèles (Hoeting et al., 1999) pondère les classifieurs selon leur probabilité a posteriori. Dans le cas du classifieur Bayesien naif sélectif, une analyse des modèles optimisés révèle que leur distribution a posteriori est si piquée que le moyennage Bayesien revient pratiquement au choix du modèle MAP, ce qui lui enlève son intérêt. De façon intermédiaire entre les poids uniformes du bagging et les poids fortement déséquilibrés du moyennage Bayesien, une nouvelle approche est proposée dans (Boullé, 2007), sur la base d'un lissage logarithmique de la distribution a posteriori des modèles.

3 Complexité algorithmique de la méthode

Dans cette partie, nous rappelons d'abord la complexité algorithmique des méthodes détaillées dans (Boullé, 2005, 2006, 2007) dans le cas ou toutes les données tiennent en mémoire vive, puis présentons l'extension de la méthode au cas de la grande volumétrie. La méthode comprend trois étapes : prétraitement univarié sur la base de discrétisations ou groupements de valeurs, sélection de variables et moyennage de modèles. L'étape de prétraitement est super-linéaire en temps de calcul et a une complexité de $O(KN \log N)$, où K est le nombre de variables et N le nombre d'instances. Dans l'étape de sélection de variables, la méthode alterne des passes d'ajout ou suppression rapide de variables (toute amélioration est immédiatement acceptée), sur la base de réordonnancements aléatoires des variables. Le processus est répété plusieurs fois de façon à mieux explorer l'espace de modélisation et à

réduire la variance causée par l'ordre d'évaluation des variables. Le nombre de passes sur les variables est fixé à $\log N + \log K$, de telle façon que la complexité globale de cette étape soit $\mathrm{O}(KN(\log K + \log N))$, ce qui est comparable à l'étape de prétraitement. L'étape de moyennage de modèles consiste à récolter l'ensemble des modèles évalués pendant l'étape de sélection de variables et à les moyenner selon un lissage logarithmique de leur probabilité a posteriori, ceci sans impact sur la complexité algorithmique. Globalement, l'algorithme d'apprentissage à une complexité algorithmique de $\mathrm{O}(KN(\log K + \log N))$ en temps et de $\mathrm{O}(KN)$ en espace.

Quand les données dépassent la capacité en mémoire vive. Avec la complexité de $\mathrm{O}(KN)$ en espace, les grands jeux de données ne tiennent pas en mémoire vive et ne peuvent plus être traités par l'algorithme précédant. Pour dépasser cette limite, nous proposons une amélioration de nos algorithmes sur la base d'un partitionnement de l'ensemble des variables. Dans un premier temps, il importe de distinguer les temps d'accès à la mémoire vive (t_1) et au disques dur, en accès séquentiel (t_2) ou aléatoire (t_3). Pour les ordinateurs modernes (année 2008), t_1 est de l'ordre de 10 nanosecondes. Les accès séquentiels au disque dur sont si rapides (sur la base de taux de transfert d'environ 100 Mb/s) que t_2 est du même ordre que t_1 : le temps de traitement du processeur est souvent un facteur limitant, quand des opérations de transcodage de valeur sont impliquées. Le temps d'accès aléatoire au disque t_3 est de l'ordre de 10 millisecondes, un million de fois plus lent que t_1 ou t_2. Ainsi, la seule voie pour traiter de très gros volumes de données est d'utiliser les disques durs de manière séquentielle.

Soit S=(KN) la taille du jeu de données à traiter et M la taille de la mémoire vive disponible. Pour l'étape de prétraitement de notre méthode, chaque variable est analysée une seule fois après avoir été chargée en mémoire vive. On partitionne l'ensemble des variables en P parties de K_P variables, de telle façon que $K_P N < M$ et $P > S/M$. L'étape de prétraitement est une boucle sur ces P sous-ensembles de variable, et à chaque itération de la boucle, l'algorithme lit les données, transcode uniquement les variables à traiter, les charge en mémoire vive, les analyse de façon à inférer les tables de probabilités conditionnelles et enfin libère la mémoire vive correspondante. Dans l'étape de sélection de variables, l'algorithme remplace dans un premier temps la valeur de chaque variable par son index dans la table de probabilités conditionnelles correspondante issue du prétraitement, et crée autant de fichiers prétraités temporaires que nécessaire. Chaque fichier prétraité peut être chargé en mémoire vive intégralement, et très rapidement puisqu'il s'agit de fichiers d'index, ne nécessitant aucune opération de transcodage. L'algorithme de sélection de variables boucle sur l'ensemble de ces fichiers en ordre aléatoire. Chaque fichier prétraité (comportant une partie des variables) est chargé en mémoire, analysé, puis libéré de la mémoire.

4 Résultats sur le Large Scale Learning Challenge

Le PASCAL Large Scale Learning Challenge[2], organisé à l'occasion de la conférence ICML 2008, a pour objectif de permettre une comparaison directe des méthodes d'apprentissage dans le cas de la grande volumétrie. Les jeux de données, présentés dans la table 1, représentent une variété de domaines, avec des jeux de données artificiels (alpha à zeta), de

[2]Web site : http ://largescale.first.fraunhofer.de/about/

TAB. 1 – *Jeux de données du challenge et performance en test de notre méthode (MODL).*

DOMAINE	INSTANCES	VARIABLES	AOPRC(BEST)	AOPRC(MODL)	RANG(MODL)
ALPHA	500000	500	0.0854	0.2528	22
BETA	500000	500	0.4577	0.4627	2
GAMMA	500000	500	0.0116	0.0116	1
DELTA	500000	500	0.0801	0.0801	1
EPSILON	500000	2000	0.0341	0.0656	8
ZETA	500000	2000	0.0115	0.0333	8
FD	5469800	900	0.1838	0.1838	1
OCR	3500000	1156	0.1584	0.1640	7
DNA	50000000	200	0.8030	0.8612	2
WEBSPAM	350000	VARIABLE	0.0004	0.0033	5

détection de visage (fd), reconnaissance de caractère (ocr), prédiction de point de coupure adn ou détection de webspam. Ils contiennent jusqu'à plusieurs millions d'instances, milliers de variables et dizaine de gigaoctets d'espace disque. Bien que notre méthode ne soit pas conçue pour être compétitive avec les méthodes "online" d'un point de vue temps d'apprentissage, les jeux de données du challenge sont intéressants pour évaluer la tenue de charge de notre méthode "offline", quand la taille des données dépasse d'un ordre de grandeur celle de la mémoire vive. Nous avons appliqué notre méthode de façon entièrement automatique, sans aucun ajustement de paramètre ni utilisation d'ensemble de validation. Dans nos expérimentations, nous avons utilisé un PC 3 Ghz sous Windows XP avec 2 Go RAM. La table 1 présente nos résultats en test sur le challenge, pour le critère de performance aoPRC (area over the precision recall curve) retenu par les organisateurs. A part pour le jeu de données alpha, notre méthode obtient toujours des performances prédictives compétitives, en tête sur trois jeux de données et en seconde position sur deux autres. Ceci est d'autant plus remarquable que la méthode est entièrement automatique et que sa capacité est limitée par l'hypothèse Bayesienne naive. Il est également à noter que quand la taille des jeux de données dépasse celle de la mémoire vive, le surcôut en temps d'apprentissage (tout compris : accès disque et traitements processeur) n'est que d'un facteur deux. Comme attendu, notre temps d'apprentissage (CPU uniquement) est nettement plus long que celui des méthodes online utilisées par la plupart des autres compétiteurs, d'environ deux ordres de grandeurs. Néanmoins, si l'on prend en compte le processus complet d'apprentissage, avec les temps d'accès au disque, de préparation des données et d'ajustement des paramètres des méthodes, notre temps d'apprentissage devient compétitif, avec environ une heure par gigaoctet de données analysé. En résumé, notre méthode est capable de traiter de manière entièrement automatique de très grandes volumétries et obtient des performances prédictives particulièrement compétitives.

5 Conclusion

Nous avons présenté une méthode de classification entièrement automatique qui exploite l'hypothèse Bayesienne naive. Elle estime les probabilités conditionnelles univariées au moyen de la méthode MODL, avec des discrétisations et groupements de valeurs optimaux pour les

variables numériques et catégorielles. Elle recherche un sous-ensemble de variables consistant avec l'hypothèse Bayesienne naive, en utilisant un critère d'évaluation selon une approche Bayesienne de la sélection de modèles et des heuristiques efficaces d'ajout/supression de variables. Enfin, elle moyenne l'ensemble des modèles évalués en exploitant un lissage logarithmique de la distribution a posteriori des modèles.

Les résultats obtenu lors du Large Scale Learning Challenge démontrent que notre méthode gère la très grande volumétrie et construit automatiquement des classifieurs performants. Quand la taille de données à traiter dépasse celle de la mémoire vice disponible d'un ordre de grandeur, notre méthode exploite une stratégie efficace de partitionnement des données sur disque, avec un surcoût en temps de traitement de seulement un facteur deux. Dans des travaux futurs, notre objectif est d'étendre la méthode aux cas de la régression et de la classification avec grand nombre de valeurs à expliquer.

Références

Boullé, M. (2005). A Bayes optimal approach for partitioning the values of categorical attributes. *Journal of Machine Learning Research 6*, 1431–1452.

Boullé, M. (2006). MODL : a Bayes optimal discretization method for continuous attributes. *Machine Learning 65*(1), 131–165.

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

Breiman, L. (1996). Bagging predictors. *Machine Learning 24*(2), 123–140.

Hand, D. et K. Yu (2001). Idiot bayes ? not so stupid after all ? *International Statistical Review 69*(3), 385–399.

Hoeting, J., D. Madigan, A. Raftery, et C. Volinsky (1999). Bayesian model averaging : A tutorial. *Statistical Science 14*(4), 382–417.

Kohavi, R. et G. John (1997). Wrappers for feature selection. *Artificial Intelligence 97*(1-2), 273–324.

Langley, P., W. Iba, et K. Thompson (1992). An analysis of Bayesian classifiers. In *10th national conference on Artificial Intelligence*, pp. 223–228. AAAI Press.

Langley, P. et S. Sage (1994). Induction of selective Bayesian classifiers. In *Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence*, pp. 399–406. Morgan Kaufmann.

Liu, H., F. Hussain, C. Tan, et M. Dash (2002). Discretization : An enabling technique. *Data Mining and Knowledge Discovery 4*(6), 393–423.

Pyle, D. (1999). *Data preparation for data mining*. Morgan Kaufmann Publishers, Inc. San Francisco, USA.

Summary

In this paper we present a parameter-free scalable classification method. The method is based on Bayes optimal univariate conditional density estimators, naive Bayes classification

enhanced with a Bayesian variable selection scheme, and averaging of models using a logarithmic smoothing of the posterior distribution. We focus on the complexity of the algorithms and show how they can cope with datasets that are far larger than the available central memory. We finally report results on the Large Scale Learning challenge, where our method obtains state of the art performance within practicable computation time.