Bagging and Boosting Classification Trees to Predict Churn

Aurélie Lemmens[1] and Christophe Croux[1]

Katholieke Universiteit Leuven

[1] K.U. Leuven, Department of Applied Economics, Naamsestraat 69, B-3000 Leuven, Belgium. Email: aurelie.lemmens@econ.kuleuven.ac.be and christophe.croux@econ.kuleuven.ac.be,
Phone: +32-16-326960, Fax: +32-16-326732

The authors thank Marnik Dekimpe for his valuable and helpful comments, as well as the Teradata Center for Customer Relationship Management at Duke University for the data and comments. This research has been funded by the Research Fund K.U.Leuven and the "Fonds voor Wetenschappelijk Onderzoek".

Bagging and Boosting Classification Trees to Predict Churn

ABSTRACT

In this paper, bagging and boosting techniques are proposed as performing tools for churn prediction. These methods consist of sequentially applying a classification algorithm to resampled or reweigthed versions of the data set. We apply these algorithms on a customer database of an anonymous U.S. wireless telecom company. Bagging is easy to put in practice and, as well as boosting, leads to a significant increase of the classification performance when applied to the customer database. Furthermore, we compare bagged and boosted classifiers computed, respectively, from a balanced versus a proportional sample to predict a rare event (here, churn), and propose a simple correction method for classifiers constructed from balanced training samples.

KEYWORDS: bagging, boosting, classification, churn, gini coefficient, rare events, sampling, top decile.

1. INTRODUCTION

Marketers are regularly confronted with classification issues, such as: who are our most profitable customers, what is our stable customer base, which customers would buy this specific product (see e.g. Bolton, Kannan and Bramlett 2000; Ganesh, Arnold and Reynolds 2000; Reinartz and Kumar 2002)? Classification techniques answer these kinds of questions by predicting, on the basis of several relevant predictive variables like personal social-demographic characteristics or past purchase behaviour, the group which a specific individual belongs to.

Logistic regression, discriminant analysis, classification trees or neural networks are common classification methods. In this paper, a technique originating from statistical machine learning, namely bagging (Breiman 1996), will be investigated. It consists of sequentially computing a base classifier from resampled versions of the training sample in order to obtain a committee of classifiers. The final classifier is then obtained by taking the average over all committee members. Bagging is very simple and easy to put in practice: it only requires a bit more computation time, but no more information than the one contained in the training sample is needed. Moreover, there is a growing literature showing that committees usually perform better than the base classifiers. As base classifier, we select a classification tree, as recommended by Breiman (1996), even if other choices are possible as well. More sophisticated versions of bagging, using weighted sampling schemes, exist under the name of boosting. Here we will compare bagging with two versions of boosting, i.e. the Real Adaboost (Freund

and Schapire 1996; Schapire and Singer 1998) and the Stochastic Gradient Boosting (Friedman 2002).

In this paper, bagging and boosting are applied to a customer database of an anonymous U.S. wireless telecom company[2]. The classification task consists in predicting churn based on personal social-demographic characteristics and past purchase behaviour. Churn, and especially voluntary churn, is a marketing-related term characterising whether a current customer decides to take his business elsewhere (i.e. to defect from one mobile service provider to another) or voluntary terminate their service[3]. Churn prediction does not only forecast whether a customer will defect or not during the following months, but also attaches a probability of churn to each customer. As such, a scoring of the risk of churn is obtained allowing the company to rank their customers according to this risk. Therefore, the marketing mix can be more adequately adapted to these customers which are the most disposed to churn with the purpose of improving retention. For example, specific incentives could be undertaken towards the most risky group (i.e. customers which are the most inclined to leave the company), such as sending personalized promotional offers via SMS, hoping that these targeted customers would remain loyal.

In the wireless telecom sector, churn is a major concern. Indeed, domestic monthly churn rates in the U.S. wireless telecom industry rose to 2 or 3% in 2001 (Telephony Online 2002) and stabilized at about 2.6% at the end of 2002 (Hawley 2003), corresponding to an annual churn rate of about 30%. In Western Europe,

[2] Provided by the Teradata Center for Customer Relationship Management at Duke University in the context of the Churn Modelling Tournament. See Gupta et al. (2003) for a study on the predictive performance of different classification methods using this database.

the situation is similar, with monthly churn rates between 2 and 3% (Hawley 2003). The main reason to this churn intensification is the increased competition between wireless providers, due to the difficulty for providers to differentiate their offers, but also to the saturation of the wireless customer base. For example, in 2002, subscriber yearly growth rates decreased from 50% to 15%-20%, while analysts predicted a 10% growth rate only for 2003 (Business Week Online 2002).

Churn is extremely damageable for companies. Indeed, *The Wall Street Journal Europe* reported that the loss of a customer cost a US wireless company an average of $676 in 1998 (2000, September 18) while *Wireless Week* mentioned that the annual turnover of nearly 40 million customers for the North American wireless industry amounts to $10 billion annually (2002, May 27). In fact, *"the top six US wireless carriers would have saved $207 million if they had retained an additional 5% of customers open to incentives but who switched plans in the past year"* (Reuters 2002). As a comparison, in the U.K., the cost of replacement of a lost wireless customer amounts to £200 to £400 in terms of sales support, marketing, advertising and commissions (SAS, 2001). Therefore, the wireless industry strategy has moved from an acquisition orientation to a retention policy. Predicting churn makes this strategy feasible. Other scientific studies also pointed out the economic value of customer retention. For example, Athanassopoulus (2000), Bhattacharya (1998), as well as Colgate and Danaher (2000) illustrated the advantage of customer retention as a low-cost operation, compared to the cost necessitated for attracting new customers. Clearly, customer retention via churn

[3] Involuntary churn is the escape or defection of consumers for reasons that are independent of the willingness (e.g. the death). Since there is no real marketing-based purpose of predicting involuntary churn because of its unavoidable nature, marketers should focus on predicting voluntary churn.

forecasting is an efficient way to maintain a sustainable level of profitability in a highly competitive world such as the Telco industry.

To evaluate the performance of a churn prediction rule, as well as the potential financial gains that would derive from it, the specificity of the problem needs to be taken into account. Only looking at the misclassification rate of the prediction rule is often misleading (see Section 5), and other criteria like the gini coefficient or the top decile, would be therefore more appropriate. These criteria will be reviewed in Section 5.

The rarity of the churn event, despite its high financial consequences, implies another issue for churn prediction. Indeed, if the sample used to build a classifier (i.e. *the training sample*) is randomly drawn from the customers' population, the percentage of churners in this training sample would be relatively low. To compensate for the low proportion of churners in such a *proportional training sample*, marketing researchers would need a sufficiently large sample size. However, this involves several drawbacks: data gathering becomes more costly and statistical estimation more time-consuming. Therefore, selective sampling is often advised as a way to avoid this efficiency loss. Here, the training sample is stratified according to outcome of interest, i.e. churn, in such a way that the two strata (churners versus non-churners) contain an equal number of customers. Such a sampling scheme will be called *balanced sampling*. Several methods exist to correct classifiers computed from disproportional training samples, hereby taking the real-life proportion of churners into account (see e.g. Cosslett 1993; Donkers, Franses and Verhoef 2003; Franses and Paap 2001; Imbens and Lancaster 1996; Scott and Wild 1997). However, no such correction has yet been provided for

bagging and boosting. In Section 4, we discuss two easy correction methods from which marketers may take profit to predict churn.

Knowing (i) that churn prediction is *crucial* for the financial wealth of companies which are therefore seeking for more performing classification methods, (ii) that a balanced sampling requires an *appropriate correction* that does not exist yet for bagging and boosting, and (iii) that *proportional or balanced* sampling could yield different performances for predicting the rare event in question, i.e. churn, we came up with three research questions that will be investigated throughout this paper:

- Q1: Does bagging, Real Adaboost or Stochastic Gradient Boosting improve the performance of the initial base classifier for churn prediction, and by how much? If so, marketers could also expect financial gains from this improvement. To answer this question, a proportional training sample will be used.

- Q2: Does the use of a balanced training sample request an appropriate correction? How do perform both corrections to be discussed in Section 4? The best correction method will then be used for the question 3.

- Q3: How do the classifiers computed from the proportional and the balanced training data perform comparatively? This question will be answered for bagging as well as boosting.

The paper is organized as follows. Section 2 contains a description of the data as well as the data preparation step. Section 3 outlines the bagging procedure, as well as the Adaboost algorithm. Section 4 presents the correction methods to be applied to the predictions computed from a balanced training

sample. Section 5 provides the assessment criteria used to evaluate the different classification rules. Results are then reported in Section 6, where the different research questions are addressed, and Section 7 provides a conclusion.

2. THE DATA

2.1. Data description

The study is performed on a dataset provided by the Teradata Center at Duke University. This database contains three datasets of respectively 51,306, 100,000 and 100,462 observations. These represent mature subscribers (i.e. customers who were with the company for at least six months) of a major U.S. wireless telecom carrier. The two first datasets will be used as training samples. They contain data extracted from the months of July, September, November and December 2001. The third dataset contains a different set of customers selected at a future point in time. A classifier will be constructed using data from one of the aforementioned training samples from which the probability to churn will be subsequently predicted for every customer in the third dataset.

The variable to predict is whether a subscriber will churn during the period 31-60 days after the sampling date. This variable is observed one to two months after the predictive variables because of practical concern. Indeed, a few weeks would actually be needed to score customers and implement proactive marketing incentives. The churn response is coded as a dummy variable with $y = 1$ if the customer churns, and $y = -1$ otherwise.

The test set contains a huge number of observations, allowing to measure the performance of the classification procedure very precisely. The proportion of churners in the test set is about 1.8% (1808 churners), meaning that churn is indeed a rather rare event.

The first training sample is a *proportional training sample.* Hence, the proportion of churners in the sample is about 1.8% (924 churners), like the actual monthly churn rate in this anonymous wireless telecom company. The second training set contains an *oversampled* number of churners such that the number of churners is equal to, i.e. *balanced by*, the number of non-churners. Theoretically, a potentially better performing classifier could be obtained from such a sample. When comparing the quality of predictions from the proportional and the balanced training sets, the number of observations of the latter one will be reduced to 51,306 observations to equal the size of both training sets.

2.2 Variables selection and transformation

To predict the churn potential of customers, U.S. wireless operators today take into account from 50 to 300 subscriber variables as explicative factors (Hawley 2003). From the high number of explicative variables contained in the initial database (171 variables), we retain 43 variables, including 31 continuous and 13 categorical variables.

Selection of the variables is based on a two-step procedure. A descriptive analysis provides a first insight about the nature of the data and the number of missing values. Variables containing more than 30% of missing values are excluded from the analysis.

In a second step, and following an unreported preliminary analysis, a subset of 43 variables was selected. According to the well-known (RFM) trilogy introduced by Cullinan (1977) and further developed by Bauer (1988), we retain recency, frequency, and monetary value variables. Other potentially important variables, like the mean unrounded minutes of customer care calls, the number of adults in the household, the education level of the customer, etc. were added as well. Most of these variables were included in similar previous studies (see Vandepoel 2003 for a review). All methods will be applied on the same set of selected variables, hereby creating a benchmark dataset making the comparison between different methods easy. In this paper, we do not study the individual contributions of each explicative variable, but focus on the predictive performance of the classification rule.

An extra variable is added indicating whether one missing value was found for at least one of the continuous variables. One could indeed imagine that not answering a question may also be informative. For categorical variables, an extra level is created indicating whether the value is missing or not.

The reasons for carrying out a variable selection procedure are mainly computational. Bagging and boosting require repetitive application of the base classifier, and working with fewer variables therefore reduces computation time. Some experiments indicated that whether a variable selection procedure was implemented or not, did, in fact, hardly changed the performance criteria of the classification rules.

3. THE BAGGING AND BOOSTING METHODOLOGY

Bagging and boosting both originate from the machine learning research community, and are based on the principle of aggregating classifiers. In a seminal paper of Breiman (1996), he found gains in accuracy by aggregating predictors built from perturbed versions of the training set in order to construct a final rule. Over the recent years, bagging and boosting received increasing attention, and were applied to various fields of application (e.g. Friedman, Hastie and Tibshirani 2000, for the UCI machine learning archive; Nardiello, Sebastiani and Sperduti 2003, for text categorisation; Varmuza, He and Fang 2003, in chemometrics; or Viane, Derrig and Dedene 2002, for an application in fraud claim detection). Statistical theory has recently been elaborated and is still under development to provide more theoretical background to these techniques (e.g. Bühlmann and Yu 2002, for bagging; Friedman, Hastie and Tibshirani 2000, for boosting). Most recent state-of-the-art supervised classification techniques can be found in Hastie, Tibshirani, and Friedman (2001), an already renowned reference in the field.

3.1. Bagging

Bagging is, by far, the simplest technique to upgrade, or to "boost", the performance of a classifier. It only requires repeated applications of the initial classifier on resampled versions of the training sample, while no additional information has to be provided in the training sample, neither in the form of additional variables, nor in terms of extra observations.

We select the decision tree as base classifier (Breiman et al. 1984). Decision trees are powerful nonparametric classification methods, available in most

software packages. According to Breiman (1996), they are adequate candidates for the bagging procedure since they are highly performing, but unstable classifiers. "Instability" refers to classifiers that significantly change when small changes in the dataset are performed. Because bagging, i.e. Bootstrap AGGregatING, averages predictions over a collection of bootstrap samples, it reduces the variance of the prediction (Bauer and Kohavi 1999) allowing for an upgrade of the predictive performance of the classifier.

Denote the training sample by $Z = \{(x_1, y_1), \ldots, (x_i, y_i), \ldots, (x_N, y_N)\}$, where N is the number of observations in the training sample. In this expression, $x_i = (x_{i1}, x_{i2}, \ldots, x_{iK})$ represents a vector containing the K explicative variables for individual i, while y_i (equal to 1 or –1) indicates whether this individual i will churn or not. A base classifier $\hat{f}$ is computed from these training data. In the case of a customer whose value of the churn variable is unknown, the base classifier returns a value $\hat{f}(x)$, with x the characteristics of this customer. This value can be considered as the score associated with the customer, i.e. a measure of its associated risk to churn. Classification into one of the two groups is accomplished by computing

$$\hat{c}(x) = sign\left(\hat{f}(x) - \tau_B\right), \tag{1}$$

giving values +1 or –1, where τ_B is a cut-off value. If $\hat{f}(x_i)$ is larger than τ_B, customer i will be classified as churner, while, if $f(x_i)$ is smaller than τ_B, it will be predicted as non-churner. When using a classification tree, the score is given by $\hat{f}(x) = 2\hat{p}(x) - 1$, where $\hat{p}(x)$ is the probability to churn as estimated by the

tree. A natural value for τ_B is $\tau_B = 0$, while in the case of non proportional sampling, the value of τ_B could vary (see Section 4).

From this original training set Z, we draw B bootstrap samples $Z_b^*, b = 1, 2, \ldots, B$. A bootstrap sample is created by randomly drawing with replacement N observations from Z. Therefore, it contains the same number of elements as the training sample, but some observations could be drawn more than once, and others could not be represented in the bootstrap sample at all. For each bootstrap sample Z_b^*, a classifier is estimated, giving B score functions $\hat{f}_1^*(x), \ldots, \hat{f}_b^*(x), \ldots, \hat{f}_B^*(x)$. These functions are then aggregated into the final score

$$\hat{f}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}_b^*(x). \tag{2}$$

Classification can then be carried out via

$$\hat{c}_{bag}(x) = sign\left(\hat{f}_{bag}(x) - \tau_B\right), \text{ with } \hat{c}_{bag}(x) \in \{-1,1\}. \tag{3}$$

Again, τ_B is a selected cut-off value, equal to zero for proportional samples. Note that bagging simply consists of B repeated applications of a classification rule on resampled training sets. The unique remaining question therefore relies to the choice of the constant B, i.e. the number of bootstrap samples to be built. A strategy consists of computing a performance criterion on the training sample, like the error rate (i.e. the apparent error rate), and to select B such that the difference between the error rates at iterations B and $B+1$ is

negligible. In our application, after $B = 50$, the performance on the training sample becomes stable.

3.2. Boosting

A comparison will be done between bagging and one of the most well-known boosting algorithms, i.e. Real Adaboost (Freund and Schapire 1996; Schapire and Singer 1998). We also include Stochastic Gradient Boosting (Friedman 2002), a new and more advanced variant of boosting, and also the winner of the Teradata Churn Modelling Tournament. Many other versions of boosting exist and are regularly proposed. Among others, LogitBoost (Friedman, Hastie and Tibshirani 2000), Random Forest (Breiman 2001), Gradient Boosting (Friedman 2001) are different variants of boosting. Research on this issue is ongoing in the machine learning and statistical community. Note that, in contrast to bagging, not all of these algorithms are straightforward to implement. We restrict the analysis to two representative variants of boosting, the Real Adaboost and the Stochastic Gradient Boosting. The first is based on the classical boosting scheme while the second is one of the most recent and advanced development in the field.

The general principle of boosting consists of sequentially applying the base learner to *adaptively* reweighted versions of the initial dataset Z_b^*, $b = 1, 2, ..., B$. Previously misclassified observations get an increased weight on the next iteration, while weights given to previously correctly classified observations are reduced. The idea is to force the classification procedure to concentrate on the hard-to-classify observations. Note that, unlike bagging, the

boosting procedure requires software that allows assigning weights to the observations of the training sample when computing the base classifier.

Another difference with bagging is that the initial classification rule is preferably a "weak" learner, i.e. a classifier that has a slightly lower error rate than random guessing. Hastie, Tibshirani and Friedman (2001) advised to use decision stumps, i.e. binary trees with only two terminal nodes, for Real Adaboost, and Friedman (2002) *k*-node trees for Stochastic Gradient Boosting where k is about 6. Other authors (e.g. Ting and Zheng 1999) suggest a naïve Bayes learner as base classifier. Such a "weak" base classifier would have a low variance, but a high bias. After B iterations of the boosting algorithm, the bias should be reduced, while the variance would remain moderate. In principle, boosting should therefore outperform bagging since it not only reduces the variance, but also the bias.

Real Adaboost allows working with score predictions $\hat{f}$, unlike Discrete Adaboost that only produces binary classification rules $\hat{c}$. In the first step of the Real Adaboost, i.e. $b = 1$, estimated probabilities $\hat{p}_1^*(x_i)$ are computed from the training set using a decision stump with equal weights for every observation:

$$w_{i,1} = 1/N, \text{ with } i = 1, \ldots, N. \tag{4}$$

Weights are computed in the further iterations on the basis of the probabilities estimated in the previous iteration. Suppose that, in step b, probabilities $\hat{p}_b^*(x_i)$, for $i = 1, 2, \ldots, N$, have been computed. The scores $\hat{f}_b(x_i)$ are then obtained by calculating the half logit-transform of the probability estimates as:

$$\hat{f}_b(x_i) = \frac{1}{2}\log\left(\frac{\hat{p}_b^*(x_i)}{1-\hat{p}_b^*(x_i)}\right). \tag{5}$$

Weights for step $b+1$ are afterwards updated by the formula

$$w_{i,b+1} = w_{i,b}\exp\left(-y_i\hat{f}_b(x_i)\right),\ i = 1,2,...,N, \tag{6}$$

and normalized such that the sum of all weights equals one. The probability estimates for iteration $b+1$, $\hat{p}_{b+1}^*(x_i)$ are then computed applying the base classifier on the weighted training sample, using the above-defined weights $w_{i,b+1}$. The procedure is repeated for $b = 1,2,\ldots,B$. The final prediction consists again of a majority vote using the scores, so

$$\hat{c}_{boost}(x) = sign\left[\sum_{b=1}^{B}\hat{f}_b(x) - \tau_B\right], \text{ with } \hat{c}_{boost}(x) \in \{-1,1\}. \tag{7}$$

Again, τ_B is a correction term for balanced training sample (see Section 4). When a proportional sample is used, $\tau_B = 0$.

The Stochastic Gradient Boosting algorithm is more evolved than Real Adaboost. We prefer not to outline it here, and refer for details to Friedman (2002).

4. CORRECTION FOR A BALANCED TRAINING SAMPLE

Even if churn has very damageable consequences for companies, it is, statistically speaking, a rare event. It concerns, per month, about 1.8% of the customers of the U.S. wireless telecom company under consideration. The rarity of the event could make it difficult to be predicted. Indeed, the group of churners in

a proportional training sample would be far smaller than the group of non-churners since it is constituted by random drawing of customers from the whole population. One may therefore fear that the characteristics driving the defection of a customer could well be difficult to be detected in such a sample, and that the vast majority of non-churners in the sample will dominate the statistical analysis.

A simple solution to handle this problem could consist in creating a balanced training sample where the proportion of churners equals the proportion of non-churners. Nevertheless, a classifier trained on such a *balanced* sample would overestimate the proportion of churners when applied on new real-life observations. In this case, an appropriate correction needs to be carried out. While such methods already exist for some common classifiers (see Section 1), we did not find any correction method for the more recent bagging and boosting classification methods.

A first solution consists in attaching a weight to the observations of the balanced training sample. Marketers or managers generally have a priori idea about the churn rate π_c, i.e. the proportion of churners, among their customers. For example, it can be estimated by the empirical frequency of churners in a proportional sample. In our case, π_c is taken as 1.8%. Let $N_c^{balanced}$ be the number of churners in the balanced sample, with N the total size of this sample. One may weight observations of a balanced training sample by attaching the weights

$$w_i^c = \frac{\pi_c}{N_c^{balanced}} \quad \text{and} \quad w_i^{nc} = \frac{1-\pi_c}{N - N_c^{balanced}} \tag{8}$$

to the churners, respectively the non-churners. In a perfectly balanced training sample,

$$N_c^{balanced} = N - N_c^{balanced}\,. \tag{9}$$

The sum of the weights defined in (8) is always equal to one. Moreover, the sum of the weights associated to the churners equals the real-life proportion of churners

$$\sum_{i=1}^{N_c^{bal.}} w_i^c = \pi_c\,. \tag{10}$$

When applying this weighting correction to bagging and Stochastic Gradient Boosting, a sequence of weighted decision trees are computed, where the weights remain fixed through iterations. For the Real Adaboost procedure however, the initial weights are now given by (8), instead of taking them all equal, as stated in (4). For the next iterations, adaptively reweighted classification rules are then computed in the same way as explained in Section 3.

Rather than weighting the observations of a balanced sample, one may consider to take a non-zero cut-off value τ_B in the bagging and boosting algorithms. The value of τ_B is taken such that the proportion of predicted churners in the training sample $\hat{\pi}_c$ equals the actual a priori proportion of churners π_c. This correction is achieved for bagging (and similarly for boosting)

by first sorting the values of $\hat{f}_{bag}(x)$ in the training sample from the largest to the smallest value, $\hat{f}_{bag}\left(x_{(1)}\right) \geq \hat{f}_{bag}\left(x_{(2)}\right) \geq \ldots \geq \hat{f}_{bag}\left(x_{(N)}\right)$, and taking

$$\tau_B = \hat{f}_{bag}\left(x_{(j)}\right) \text{ , with } j = N_c^{balanced} . \tag{11}$$

This latter correction method can also be called intercept correction, by analogy to the correction carried out for the logistic regression model when oversampling the population of $y = 1$ (see e.g. Franses and Paap 2001, pp. 73-75). Both correction methods will be compared in Section 6.

Note that, given the high amount of observations in the Teradata database, the absolute number of churners in the training set under consideration is never insignificant. For example, the proportional training sample still contains 924 churners. Therefore, even if churn is a rare event, a proportional sampling could still be efficient, while, for smaller databases, stratified sampling could be more relevant. The third research question (Section 6.3) addresses this issue.

5. THE ASSESSMENT CRITERIA

After building a classifier on a training set with bagging or boosting, marketers may use it to predict the future churn behaviour of their customers. To assess the precision of such predictions, one has to use a test set, as the one included in the Teradata database (see Section 2). The principle is that this test set has not been used for constructing the classification rules, and will therefore give reliable indicators of performance for marketers. Indeed, if one assesses a classifier on its respective training set, his judgment could be biased because of the overfitting problem. Overfitting means that a classifier fits the idiosyncrasies

of the training set too closely. It leads to lower error rates on the training set, but at the same time much higher error rates on the test set (more details in Berry and Linoff, 1997, pp.79-80).

Denote $\{(x_1, y_1),\ldots,(x_i, y_i),\ldots,(x_M, y_M)\}$ the test set. The scores computed for a given classifier are denoted by $\hat{f}(x_i)$, and the classifications themselves as $\hat{c}(x_i)$, for $i = 1,\ldots,M$ where M is the size of the test sample. The traditional performance criterion is the error rate, counting the percentage of incorrectly classified observations in the test set:

$$Error\ Rate = \frac{1}{M}\sum_{i=1}^{M} I[\hat{c}(x_i) \neq y_i], \tag{12}$$

where $I[A]$ is an indicator function, equal to 1 when A is true and zero otherwise. For rare events, the error rate is often inappropriate, as already noticed by Morrison (1969). For example, a prediction rule stating that nobody churns would have an expected error rate of about 1.8% only, which could be falsely considered as sufficiently good. Indeed, marketers would not be satisfied with such a classification rule that simply does not make any distinction between customers. Since all are expected not to churn, no targeted incentives could be undertaken to touch the riskiest of the customers. Moreover, error rates do not take the scores $\hat{f}(x_i)$ into account, while it is relevant to check whether future churners indeed receive the highest scores. In other words, it is important to know if the customers that would be targeted with special offerings are indeed the most inclined to churn. The top decile and the gini coefficient are appropriate measures for assessing this power of "discriminability".

5.1. Top decile

The top decile only focuses on the top 10% of customers that are predicted as most likely to churn. Potentially, this segment is the perfect target for a retention marketing campaign. The top decile measures the proportion of churners in this top 10% divided by the total proportion of churners in the whole test set. Practically, customers are first sorted from the predicted most likely to churn to the predicted least likely to churn, i.e. $\hat{f}(x_1) \geq \hat{f}(x_2) \geq \ldots \geq \hat{f}(x_M)$. Then, the exact proportion of actual churners *from the top 10%* predicted most likely to churn is computed as

$$\hat{\pi}_{10\%} = \frac{1}{m}\sum_{i=1}^{m} I[y_i = 1], \tag{13}$$

where m is the number of observations in this top 10% most risky customers, and the churn rate across *all* customers is estimated by

$$\hat{\pi} = \frac{1}{M}\sum_{i=1}^{M} I[y_i = 1]. \tag{14}$$

Finally, the top decile is obtained by computing the ratio between both proportions:

$$Top\ decile = \frac{\hat{\pi}_{10\%}}{\hat{\pi}}. \tag{15}$$

The higher the top decile, the better the classifier.

5.2. Gini coefficient

Another possible measure is the gini coefficient (e.g. Hand 1997, p.134). We first determine the fraction of *all subscribers* having a predicted churn probability above a certain threshold. A whole sequence of thresholds is considered, each of them given by a predicted score $\hat{f}(x_I)$, for $I = 1, 2, ..., \mathrm{M}$, resulting in M proportions

$$\pi_I = \frac{1}{M} \sum_{i=1}^{M} I\left[\hat{f}(x_i) > \hat{f}(x_I)\right]. \tag{16}$$

The fraction of *all churners* having predicted churn probability above this threshold is also computed for each threshold

$$\pi'_I = \frac{1}{M_c} \sum_{i=1}^{M_c} I\left[\hat{f}(x_i) > \hat{f}(x_I) \text{ and } y_i = 1\right], \tag{17}$$

with M_c the total number of actual churners in the test set. The gini coefficient is then computed as

$$\textit{Gini coefficient} = \frac{2}{M} \sum_{I=1}^{M} (\pi'_I - \pi_I). \tag{18}$$

The idea behind the gini coefficient consists in giving a larger penalization for misclassified customers having high associated predicted probability to churn, than for those with low associated predicted probability. For marketers, it is indeed important to maximize the number of future churners that would be targeted by incentives and, on the other hand, to minimize the amount of customers that would be targeted while they are not potential churners. The latter could have as a consequence that a company would "waste" money by

offering special incentives to customers that would not be willing to leave the company in the next future.

6. RESULTS

This section addresses the three research questions exposed in Section 1. It will be shown (i) that both bagging and boosting techniques significantly improve the classification performance, (ii) that the correction methods for a balanced training sample reduce the classification error rate, and (iii) that the use of a balanced training sample improves the forecasting accuracy of the bagging procedure, while this was not confirmed for boosting. Details for the computation of these classifiers were described in the methodological Section 3.

6.1. Do bagging and boosting improve the performance of the initial base classifier for churn prediction?

Classification or regression trees, also known as CART, are common classifiers, elaborated by Breiman et al. (1984). Decision trees can be used in several marketing applications such as market segmentation, consumer choice modelling or purchase timing modelling (see e.g. Baines 2003; Currim, Meyer and Le 1988; Haughton and Oulabi 1993; Newman and Staelin 1971). Bagging and boosting will be done by sequentially applying the CART algorithm[4] to the resampled or reweighted training data, resulting in a bagged or boosted classifier. This classifier gives an associated score $f(x)$, i.e. the propensity to churn, to each observation, as well as a classification outcome $c(x)$, the latter being +1 or –1.

[4] As implemented in the statistical software package Splus.

We compute the gini coefficient and the top decile on the test set in order to evaluate the gain in performance obtained from bagging and boosting. In the bagging and boosting literature, results are usually presented as a function of B, the number of iterations performed in the procedure. Figure 1 represents both performance criteria against the number of iterations of bagging or boosting (Real Adaboost and Stochastic Gradient Boosting). The training set under consideration is the proportional sample.

[Insert Figure 1 about here]

The horizontal line in Figure 1 represents the performance of the initial classifier. It is constant since it is only computed once. Bagging and boosting clearly outperform such a classical decision tree, confirming hereby many other examples (e.g. Hastie, Tibshirani and Friedman 2001, pp.246-249 & 299-345). Bagging and boosting already perform better than the CART classifier after a few iterations, with respect to the gini coefficient as well as the top decile. When the number of iterations approaches 50, the value of both criteria stabilizes, confirming that $B = 50$ is an acceptable default choice.

The Stochastic Gradient Boosting classifier achieves here the best performance, for the gini coefficient as well as the top decile. Figure 2 reports the error rate (on the test set) as a function of the number of iterations. Due to the rarity of the churn event, all error rates are small and difficult to distinguish. As explained in Section 5, only looking at the error rate is misleading here. Indeed, no clear difference between the performances of the three final classifiers appears, while a significant difference was clearly visible for the gini coefficient and the top decile measures.

[Insert Figure 2 about here]

6.2. Does the use of a balanced training sample request an appropriate correction? How do perform the two corrections discussed in Section 4?

Two corrections were envisaged to adapt the predicted probabilities obtained by using a balanced training sample. Using any of these two corrections provides a very significant improvement of the error rate, as illustrated in Figure 3. Without a correction, the classification error is unreasonably high, i.e. about 40%. Using any of both correction methods brings the classification error rate to more reasonable proportions.

[Insert Figure 3 about here]

Since a correction is necessary, we also would like to assess the relative performance of both corrections: the correction by weighting of the sample versus the use of a non-zero cut-off value τ_B. The assessment is first done for the bagging classifier. While the error rate hardly distinguished which correction method works best (see Figure 3), the gini coefficient and the top decile provide more evidence about the best correction to use. Figure 4 reports the gini coefficient and the top decile for a bagged classifier trained on the balanced training sample, respectively corrected by weighting or with intercept correction. One may observe that the intercept correction outperforms the weighting correction.

[Insert Figure 4 about here]

For boosting, both correction methods comparatively perform (see Figure 5). Note that, since the gini coefficient and the top decile only use the relative ranking of the attributed scores, their values for uncorrected bagging/boosting and bagging/boosting with intercept correction are identical.

[Insert Figure 5 about here]

Reweighting the observations from a balanced training sample makes it representative of the real-life population of customers. This approach is statistically valid, but cancels the advantage relative to the balanced sampling. Further experiments showed that reweighting observations of a balanced training sample, in fact, gave similar results than working with a proportional training sample of the same size. Given these results, the intercept correction will be applied to handle the third question.

6.3. How do the predictions from the proportional and the balanced training data comparatively perform?

It is often advised to construct a more efficient classifier from a balanced training sample when the variable to be predicted consists of a rare event, like churn. This third research question puts this statement into question when using bagging and boosting classifiers. We compare the performance of balanced and proportional classifiers on two criteria, the gini coefficient and the top decile. As already mentioned above, we build the classifiers on the same number of observations (i.e. 51,306 customers) in order to ensure a fair comparison.

Figure 6 represents the gini coefficient as well as the top decile of two classifiers constructed on the basis of the bagging algorithm. Results indicate that the balanced sampling scheme is recommended. For the error rate (not reported here), results for both classifiers are again very close to each other.

[Insert Figure 6 about here]

Figure 7 reports similar results as Figure 6, but now for the Stochastic Gradient Boosting algorithm[5], respectively computed from the balanced and the proportional training samples. Fixing the number of iterations at $B = 50$, Figure 7 illustrates that a proportional sample provides, in this case, better results. Moreover, we did find out that the initial weighting correction gives slightly better results than intercept correction for boosting. When comparing bagging and Stochastic Gradient Boosting applied to the balanced sample, it appears that bagging performs slightly better here, while, for the proportional sample, Boosting clearly outperforms bagging. Note that it is not possible to find a universally best classifier, since performance always depends on the dataset under consideration.

[Insert Figure 7 about here]

7. MARKETING IMPLICATIONS AND CONCLUSIONS

The aim of this paper was to bring some new developments from the machine learning and statistical classification literature under the attention of marketing researchers. We used one of the simplest versions of aggregated

[5] For Real Adaboost, we do not wish to generalize these statements, given the observed instabilities during the boosting iterations.

classifiers, i.e. bagging, one of the most standard version of boosting, i.e. Real Adaboost, as well as one of most recent algorithms of this emerging research field, i.e. Stochastic Gradient Boosting. Bagging is an easy procedure aimed at increasing the classification performance of an initial classifier, by repeatedly applying this classifier to bootstrapped versions of the training sample. Boosting algorithms are most sophisticated versions using weighted sampling schemes.

In this paper, we found that, when predicting churn, bagging and boosting provide substantially better classifiers than a CART decision tree. It has been shown that the Stochastic Gradient Boosting scheme yields superior results in some cases. Our feeling, however, is that the simple bagging algorithm already gives reliable and stable results. Moreover, it is a transparent procedure in the tradition of the bootstrap (more details in Efron and Tibshirani 1993) and well suited for practitioners which are non-experts in modern classification techniques.

Another contribution of this paper is the study of the appropriateness of a balanced training sample compared to a proportional sample, when predicting rare events, like churn. For smaller data sets, it is clear that a balanced sample can increase precision (see e.g. Donkers, Franses and Verhoef 2003). However, for churn prediction in the telecom industry, one typically has enormous databases, allowing to have enough churners even under a proportional sampling scheme. We found that bagging still found profit from balanced samples, while selective sampling did not increase efficiency for boosting. For bagging, in contrast to boosting, balanced sampling indeed provides better gini coefficients and top deciles than proportional sampling. However, to maintain the classification error rate at a reasonable level, it is indispensable to correct the predictions obtained from a balanced sample with an intercept correction, as discussed in Section 4.

Companies should profit from the bagging and boosting algorithms in the elaboration of their retention strategy. For example, it was shown (Figure 1) that the gini coefficient, an appropriate measure in the context of churn prediction, got a *relative* increase of about 55% after running the bagging iterations with respect to a single tree, and about 110% for Stochastic Gradient Boosting. The relative increase for the top decile measure was about 28% with respect to a single tree classifier, about 69% for Stochastic Gradient Boosting. Moreover, this performance gain only requires an increased computing time, linearly growing with the number of iteration steps B.

If companies are able to identify more accurately the potential churners, they should be able to more precisely target these with special incentives and to translate the improved prediction accuracy into real profits. Indeed, if churn prediction belongs to a fully integrated Business Intelligence process, including an efficient data management, a proper data analysis (e.g. bagging or boosting), and the appropriate subsequent marketing decisions about adequate incentives aimed at future churners, profits can be huge.

Figure 1: Gini coefficient (left) and top decile (right) on the test set for bagging, Real Adaboost, Stochastic Gradient Boosting and a single CART tree

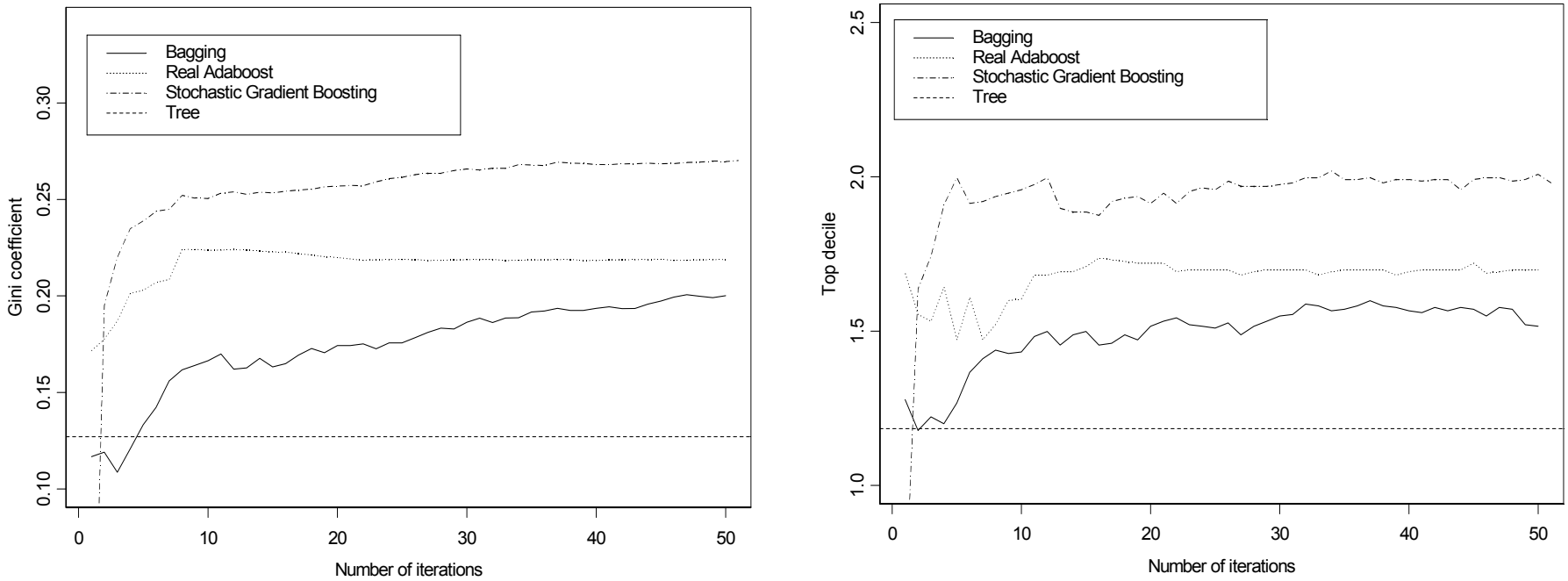


Figure 2: Error rate on the test set for bagging, Real Adaboost, Stochastic Gradient Boosting and a single CART tree

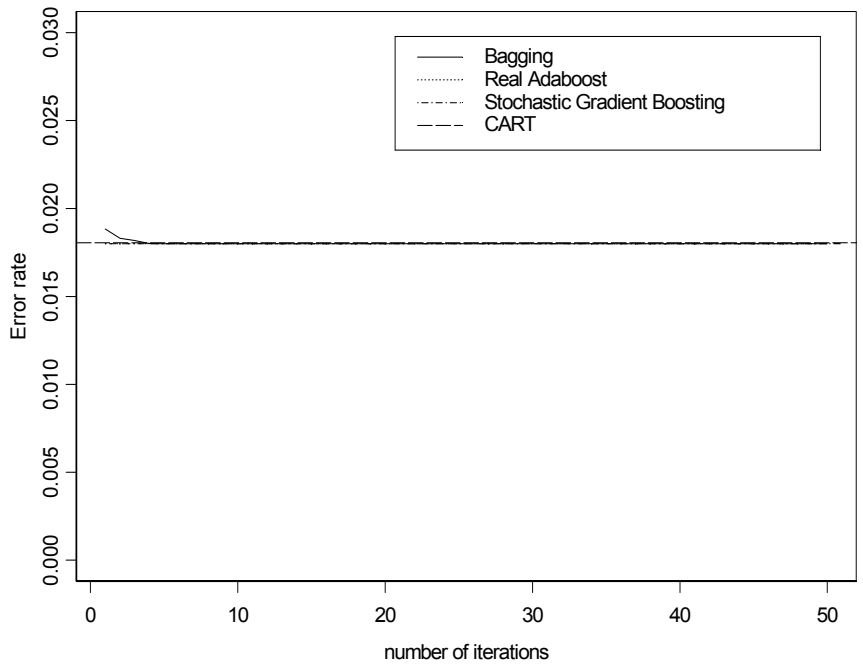


Figure 3: Error rate on the test set for bagging with intercept correction, for reweighted bagging and for uncorrected bagging.

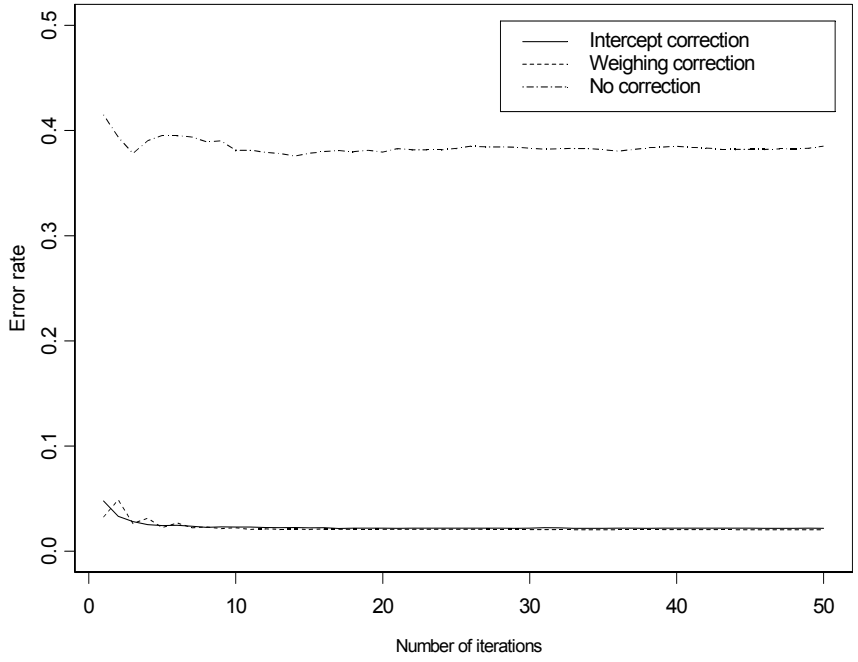


Figure 4: Gini coefficient (left) and top decile (right) on the test set for bagging with intercept correction and for reweighted bagging.

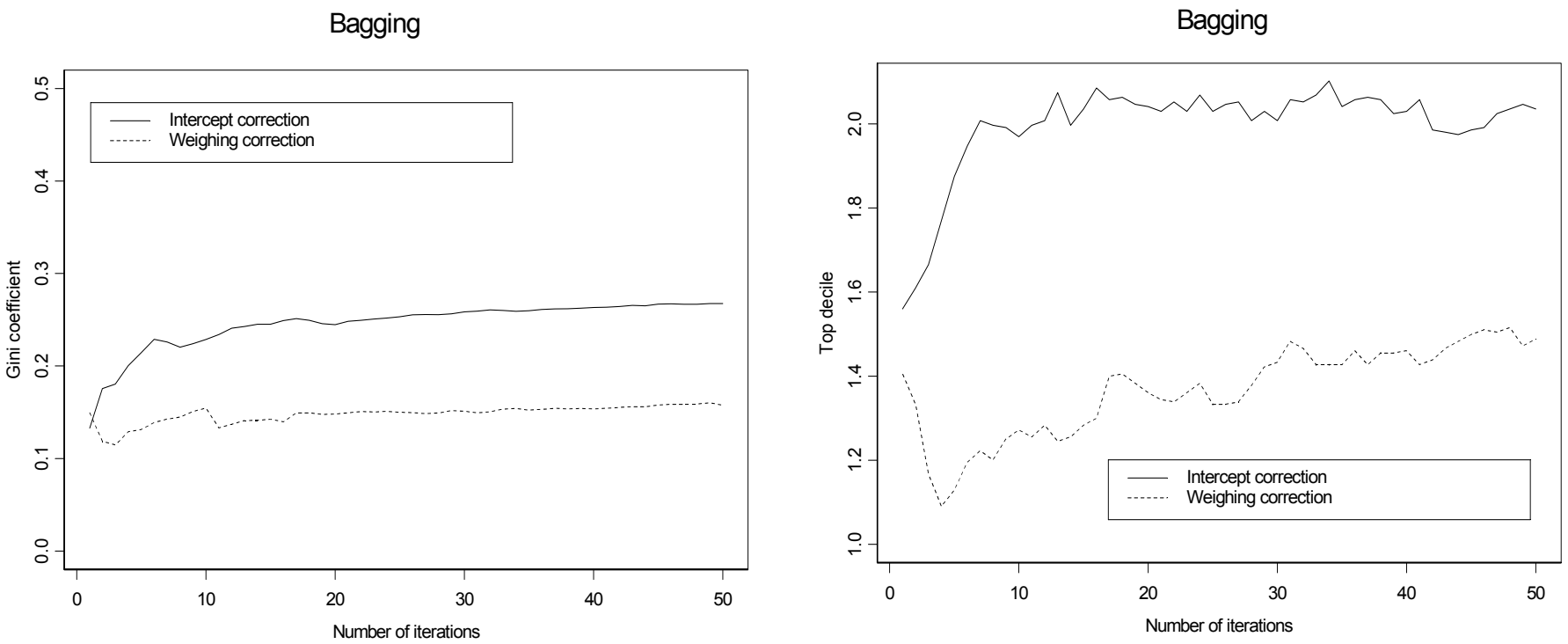


Figure 5: Gini coefficient (left) and top decile (right) on the test set for intercept corrected or reweighted Stochastic Gradient Boosting.

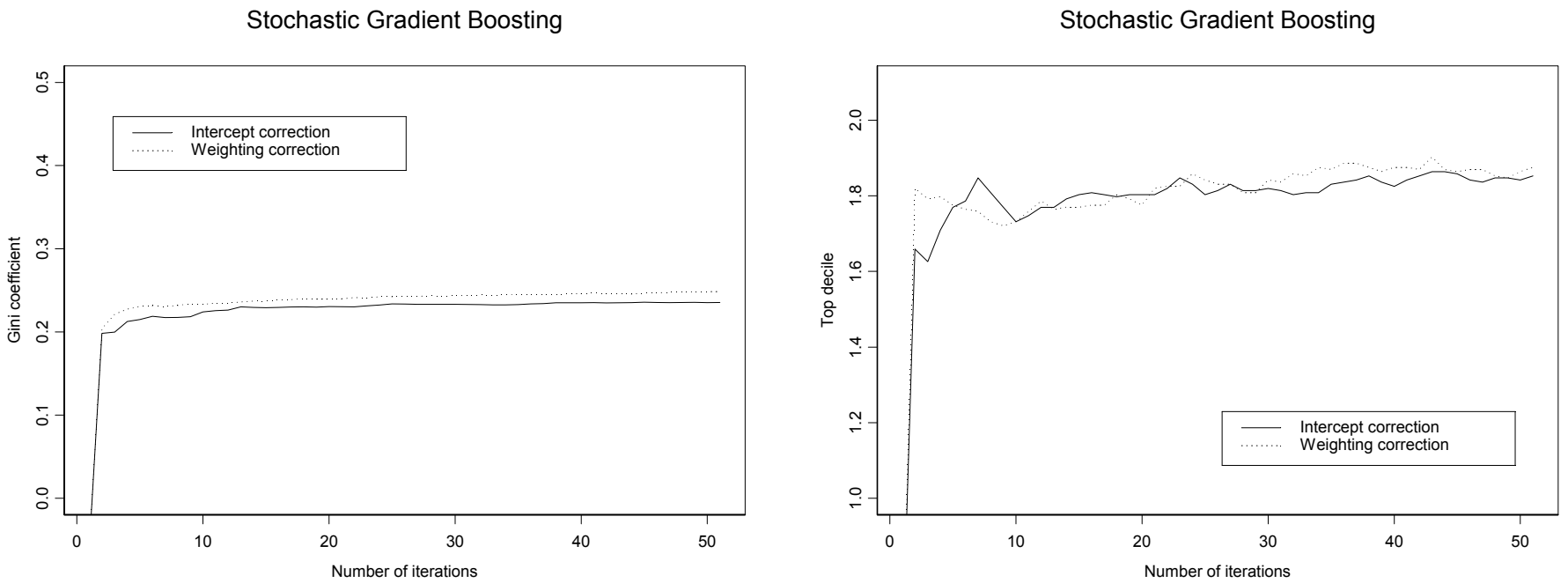
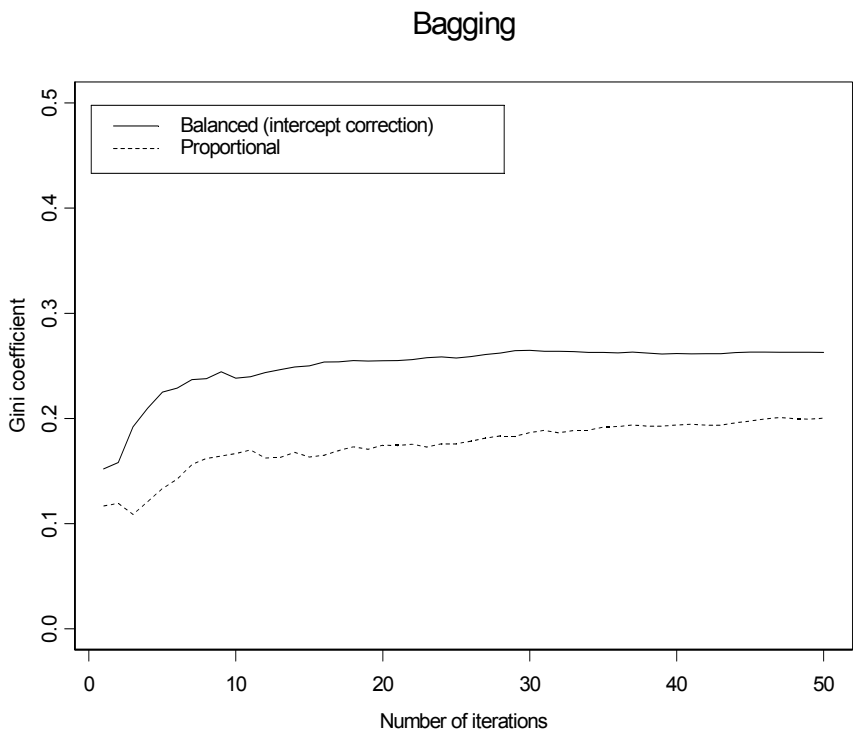


Figure 6: Gini coefficient (left) and top decile (right) on the test set for bagging with a balanced or a proportional training sample



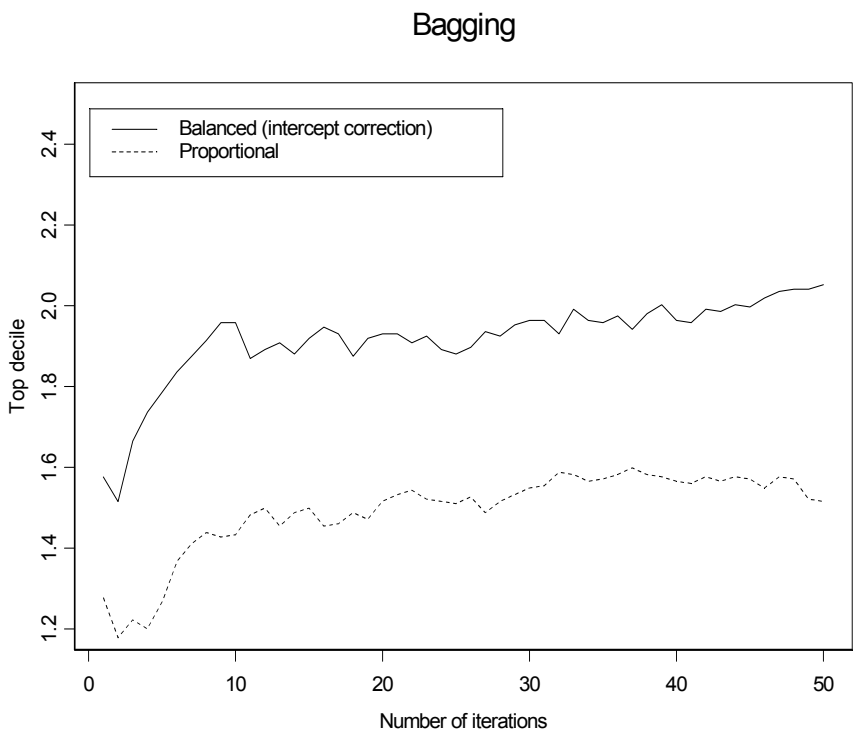
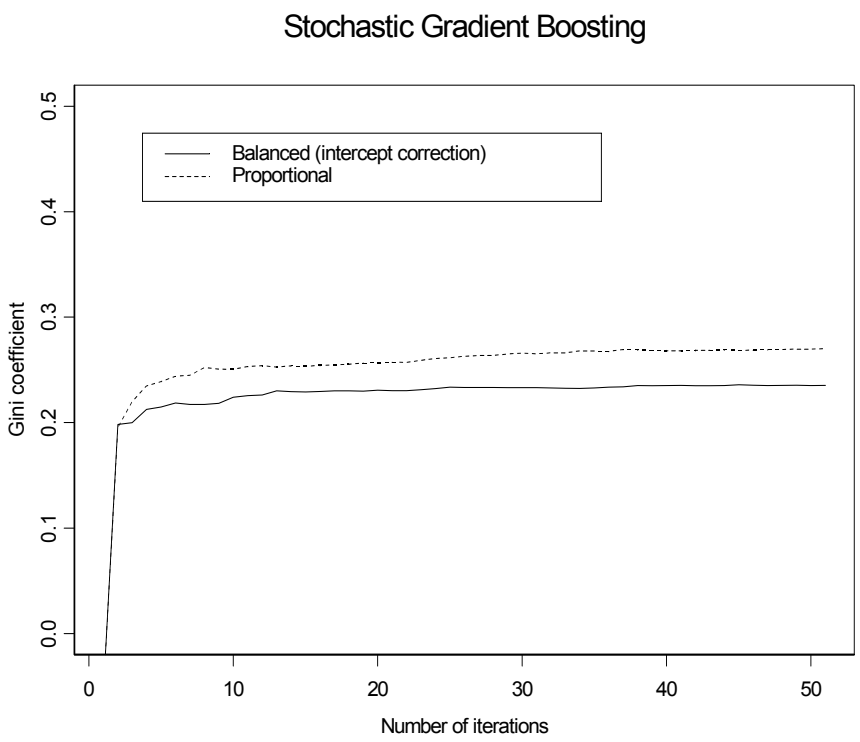
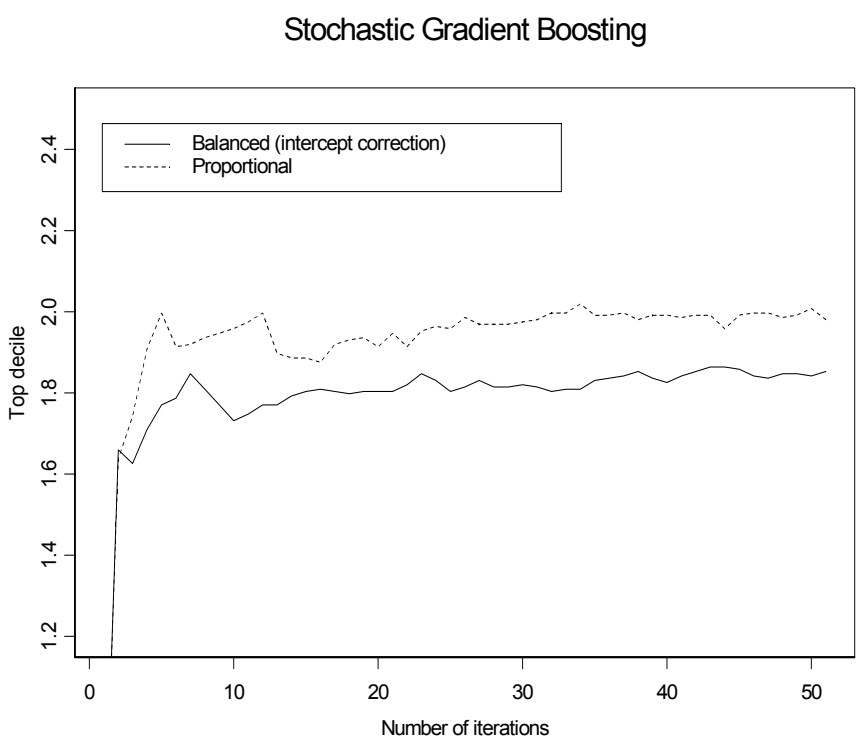


Figure 7: Gini coefficient (left) and top decile (right) on the test set for Stochastic Gradient Boosting with a balanced and a proportional training sampling





REFERENCES

Athanassopoulos, Antreas D. (2000), "Customer Satisfaction Cues to Support Market Segmentation and Explain Switching Behavior", *Journal of Business Research*, 47, 191-207.

Baines Paul R., Worcester Robert M., Jarrett David and Roger Mortimore (2003), "Market Segmentation and Product Differentiation in Political Campaigns: A Technical Feature Perspective", *Journal of Marketing Management*, 19, 225-249.

Bauer Eric and Ron Kohavi (1999), "An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting and Variants", *Machine Learning*, 36, 105-142.

Bauer, Connie L. (1988), "A Direct Mail Customer Purchase Model", *Journal of Direct Marketing*, 2, 16-24.

Berry Michael J.A. and Gordon Linoff (1997), *Data Mining Techniques for Marketing, Sales and Customer Support*, New York: Wiley Computer Publishing.

Bhattacharya, C.B. (1998), "When Customers are Members: Customer Retention in Paid Membership Contexts", *Journal of the Academy of Marketing Science*, 26, 31-44.

Bolton Ruth N., P.K. Kannan and Matthew D. Bramlett (2000), "Implications of Loyalty Program Membership and Service Experiences for Customer Retention and Value", *Journal of the Academy of Marketing Science*, 28, 95-108.

Breiman, Leo (1996), "Bagging Predictors", *Machine Learning*, 26, 123-140.

---- (1999), "Random Forests - Random Features", *Technical Report* 567, Department of Statistics, University of California, Berkeley, September.

----, Jerome .H. Friedman, Richard A. Olshen, and Charles J. Stone (1984), *Classification and Regression Trees*, Wadsworth Belmont CA.

Bühlmann Peter and Bin Yu. (2002), "Analysing Bagging", *Annals of Statistics*, 30, 927-961.

Business Week Online (2002), "Who'll Survive the Cellular Crisis", February 15.

Colgate Mark R. and Peter J. Danaher (2000), "Implementing a Customer Relationship Strategy: The Asymmetric Impact of Poor versus Excellent Execution", *Journal of the Academy of Marketing Science*, 28, 375-387.

Cosslett, S.R. (1993), "Estimation from Endogenously Stratified Samples", in *Maddala G.S., C.R. Rao and H.D. Vinod* (eds), Handbook of Statistics, Vol. 11, Elsevier Science Publishers.

Cullinan, G. J. (1977), *Picking Them by Their Batting Averages' Recency – Frequency – Monetary Method of Controlling Circulation*, manual release 2103, New York: Direct Mail/Marketing Association.

Currim Imran S., Meyer Robert J. and Nhan T. Le (1988), "Disaggregate Tree-Structure Modelling of Consumer Choice Data", *Journal of Marketing Research*, 25, 253-265.

Donkers Bas, Philip H. Franses, and Peter Verhoef (2003), "Selective Sampling for Binary Choice Models", *Journal of Marketing Research*, 40, 492-497.

Efron Brad and Robert Tibshirani (1993), *An Introduction to the Bootstrap*, New York: Chapman and Hall.

Franses Philip H. and Richard Paap (2001), *Quantitative Models for Marketing Research*, Cambridge: Cambridge University Press, 73-75.

Freund Yoav & Robert E. Schapire (1996), "Experiments with a New Boosting Algorithm", In *Proceedings of the 13th International Conference on Machine Learning*, 148-156.

Friedman, Jerome H. (2002), "Stochastic Gradient Boosting", *Computational Statistics and Data Analysis*, 38, 367-378

----, Trevor Hastie & Robert Tibshirani (2000), "Additive Logistic Regression: a Statistical View of Boosting", *The Annals of Statistics*, 28, 337-407.

---- (2001), "Greedy Function Approximation: A Gradient Boosting Machine", *The Annals of Statistics*, 29, 1189-1232.

Ganesh Jaishankar, Mark J. Arnold and Kristy E. Reynolds (2000), "Understanding the Customer Base of Service Providers: An Examination of the Differences between Switchers and Stayers", *Journal of Marketing*, 65, 65-87.

Gupta Sunil, Wagner Kamakura, Junxiang Lu, Charlotte Mason and Scott A. Neslin (2003), "Churn Modelling Tournament", paper presented at *Marketing Science Conference*, University of Maryland.

Hand, David J. (1997), *Construction and Assessment of Classification Rules*, Chichester: Wiley Series in Probability and Statistics.

Hastie Trevor, Robert Tibshirani and Jerome Friedman (2001), *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, New York: Springer-Verlag.

Haughton Dominique and Samer Oulabi (1993), "Direct marketing modelling with CART and CHAID", *Journal of Direct Marketing*, 7, 16- 26.

Hawley, David (2003), "International Wireless Churn Management Research and Recommendations", *Yankee Group report*, June.

Imbens Guido W. and Tony Lancaster (1996), "Efficient Estimation and Stratified Sampling", *Journal of Econometrics*, 74, 289-318.

Lysne, Alan (2002), "Chopping Churn With Care", *Wireless Week*, May 27.

Morrison, Donald G. (1969), "On the Interpretability of Discriminant Analysis", *Journal of Marketing Research*, 6, 156-163.

Nardiello Pio, Fabrizio Sebastiani, and Alessandro Sperduti (2003), "Discretizing Continuous Attributes in AdaBoost for Text Categorization", *Proceedings of ECIR-03, 25th European Conference on Information Retrieval*, Pisa, Italy, 320-334.

Newman Joseph W. and Richard Staelin (1971), "Multivariate Analysis of Differences in Buyer Decision Time", *Journal of Marketing Research*, 8, 192-198.

Reinartz Werner and Vishesh Kumar (2002), "The Mismanagement of Customer Loyalty", *Harvard Business Review*, July, pp. 86-97

SAS Institute (2001), "Enterprise Miner Reference", SAS.

Schapire Robert E. and Yoram Singer (1998), "Improved Boosting Algorithms using Confidence-rated Predictions", in *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*.

Scott Alastair J. and Chris J. Wild (1997), "Fitting Regression Models to Case-Control Data by Maximum Likelihood", *Biometrika*, 84, 57-71.

Snel, Ross (2000), "Fighting the Fickle", *The Wall Street Journal Europe*, September 18.

Telephony Online (2002), "Standing by Your Carrier", March 19.

Ting Kai Ming and Zijian Zheng (1999), "Improving the Performance of Boosting for Naive Bayesian Classification", In *Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining '99 Berlin*, Springer-Verlag.

Van den Poel, Dirk (2003), "Predict Mail-Order Repeat Buying: Which Variables Matter?", *Tijdschrift voor Economie en Management*, 48, 371-403.

Varmuza Kurt, Ping He and Kai-Tai Fang (2003), "Boosting Applied to Classification of Mass Spectral Data", *Journal of Data Science*, 1, 391-404.

Viaene Stijn, Richard A. Derrig and Guido Dedene (2002). "Boosting Naive Bayes for Claim Fraud Diagnosis", in *Lecture Notes in Computer Science 2454*, Berlin: Springer.