Design and Analysis of the KDD Cup 2009

Fast Scoring on a Large Orange Customer Database

Isabelle Guyon *
Clopinet
955, Creston Rd
Berkeley, California

Vincent Lemaire &
Marc Boullé
Orange Labs
Lannion, France

Gideon Dror
Academic College of
Tel-Aviv-Yaffo
Tel Aviv, Israel

David Vogel
Data Mining
Solutions
Orlando, Florida

ABSTRACT

We organized the KDD cup 2009 around a marketing problem with the goal of identifying data mining techniques capable of rapidly building predictive models and scoring new entries on a large database. Customer Relationship Management (CRM) is a key element of modern marketing strategies. The KDD Cup 2009 offered to participants an opportunity to work on a large marketing database from the French Telecom company Orange. The tasks were to predict the propensity of customers to switch provider (churn), buy new products or services (appetency), or buy upgrades/add-ons proposed to them to make the sale more profitable (up-selling). The challenge, which lasted from March 10 to May 11, 2009, attracted over 450 participants from 46 countries. We attribute its popularity to several factors: (1) A generic problem relevant to the Industry (a classification problem), but presenting a number of scientific and technical challenges, including many missing values (about 60%), a large number of features (15000) and a large number of training examples (50000), unbalanced class proportions (fewer than 10% of the examples of the positive class), noisy data, and the presence of categorical variables with many different values. (2) Prizes (Orange offers 10000 Euros in prizes). (3) A well designed protocol and web site (we benefitted from past experience). (4) An effective advertising campaign using mailings and a teleconference to answer potential participants questions. The results of the challenge were discussed at the KDD conference (June 28, 2009). The principal conclusions are that ensemble methods are very effective and that ensemble of decision trees offer off-the-shelf solutions to problems with large numbers of samples and attributes, mixed types of variables, and lots of missing values. The data and the platform of the challenge remain available for research and educational purposes at `http://www.kddcup-orange.com/`.

1. INTRODUCTION

Customer Relationship Management (CRM) is a key element of modern marketing strategies. The KDD Cup 2009 offered the opportunity to work on large marketing databases from the French Telecom company Orange to predict the propensity of customers to switch provider (churn), buy new products or services (appetency), or buy upgrades or add-ons proposed to them to make the sale more profitable (up-selling).

The most practical way to build knowledge on customers in a CRM system is to produce scores. A score is the output of a predictive model, which uses a number of explanatory variables or features, extracted from a customer's record, to predict an outcome of interest, *e.g.,* churn, appetency or up-selling. The scores produced by the model are then used by the information system (IS), for example, to personalize the customer relationship. The rapid and robust detection of the most predictive variables is a key factor in a marketing application. An industrial customer analysis platform developed at Orange Labs, capable of building predictive models for datasets having a very large number of input variables (thousands) and instances (hundreds of thousands), is currently in use by Orange marketing. A key requirement is the complete automation of the whole process. The system extracts a large number of variables from a relational database, selects a subset of informative variables and instances, and efficiently builds in a few hours an accurate classifier. When the models are deployed, the platform exploits sophisticated indexing structures and parallelization in order to compute the scores of millions of customers, using the best representation. More details are found in [10].

The challenge was to beat the in-house system developed by Orange Labs. It was an opportunity for participants to prove that they could handle a very large database, including heterogeneous noisy data (numerical and categorical variables), and unbalanced class distributions. Time efficiency is often a crucial point. Therefore part of the competition was time-constrained to test the ability of the participants to deliver solutions quickly. The fast track of the challenge lasted five days only. To encourage participation, the slow track of the challenge allowed participants to continue working on the problem for an additional month. A smaller database was also provided to allow participants with limited computer resources to enter the challenge.

2. BACKGROUND AND MOTIVATIONS

This challenge uses important marketing problems to benchmark classification methods in a setting, which is typical of large-scale industrial applications. A large database was made available by the French Telecom company, Orange with tens of thousands of examples and variables. This dataset is unusual in that it has a large number of variables making the problem particularly challenging to many state-of-the-art machine learning algorithms. The challenge

*Corresponding author
isabelle@clopinet.com

participants were provided with masked customer records and their goal was to predict whether a customer will switch provider (churn), buy the main service (appetency) and/or buy additional extras (up-selling), hence solving three binary classification problems. Churn is the propensity of customers to switch between service providers, appetency is the propensity of customers to buy a service, and up-selling is the success in selling additional good or services to make a sale more profitable. Although the technical difficulty of scaling up existing algorithms is the main emphasis of the challenge, the dataset proposed offers a variety of other difficulties: heterogeneous data (numerical and categorical variables), noisy data, unbalanced distributions of predictive variables, sparse target values (only 1 to 7 percent of the examples examples belong to the positive class) and many missing values.

3. EVALUATION

There is value in a CRM system to evaluate the propensity of customers to buy. Therefore, tools producing scores are more usable that tools producing binary classification results. The participants were asked to provide a score (a discriminant value or a posterior probability $P(Y = 1|X)$), and they were judged by the area under the ROC curve (AUC). The AUC is the area under the curve plotting sensitivity *vs.* (1− specificity) when the threshold θ is varied (or equivalently the area under the curve plotting sensitivity *vs.* specificity). We call "sensitivity" the error rate of the positive class and "specificity" the error rate of the negative class. The AUC is a standard metric in classification. There are several ways of estimating error bars for the AUC. We used a simple heuristic, which gives us approximate error bars, and is fast and easy to implement: we find on the AUC curve the point corresponding to the largest balanced accuracy BAC = 0.5 (sensitivity + specificity). We then estimate the standard deviation of the BAC as:

$$\sigma = \frac{1}{2}\sqrt{\frac{p_+(1-p_+)}{m_+} + \frac{p_-(1-p_-)}{m_-}}\ , \qquad (1)$$

where m_+ is the number of examples of the positive class, m_- is the number of examples of the negative class, and p_+ and p_- are the probabilities of error on examples of the positive and negative class, approximated by their empirical estimates, the sensitivity and the specificity [14].

The fraction of positive/negative examples posed a challenge to the participants, yet it was sufficient to ensure robust prediction performances (as verified in the beta tests). The database consisted of 100,000 instances, split randomly into equally sized train and test sets:

- **Churn problem:** 7.3% positive instances (3672 / 50000 on train).
- **Appetency problem:** 1.8% positive instances (890 / 50000 on train).
- **Up-selling problem:** 7.4% positive instances (3682 / 50000 on train).

On-line feed-back on AUC performance was provided to the participants who made correctly formatted submissions, using only 10% of the test set. There was no limitation on the number of submissions, but only the last submission on the test set (for each task) was taken into account for the final ranking.

The score used for the final ranking was the average of the scores on the three tasks (churn, appetency, and up-selling).

4. DATA

Orange (the French Telecom company) made available a large dataset of customer data, each consisting of:

- **Training :** 50,000 instances including 15,000 inputs variables, and the target value.
- **Test :** 50,000 instances including 15,000 inputs variables.

There were three binary target variables (corresponding to churn, appentency, and up-selling). The distribution within the training and test examples was the same (no violation of the i.i.d. assumption - independently and identically distributed). To encourage participation, an easier task was also built from a reshuffled version of the datasets with only 230 variables. Hence, two versions were made available ("small" with 230 variables, and "large" with 15,000 variables). The participants could enter results on either or both versions, which corresponded to the same data entries, the 230 variables of the small version being just a subset of the 15,000 variables of the large version. Both training and test data were available from the start of the challenge, without the true target labels. For practice purposes, "toy" training labels were available together with the training data from the onset of the challenge in the fast track. The results on toy targets did not count for the final evaluation. The real training labels of the tasks "churn", "appetency", and "up-selling", were later made available for download, half-way through the challenge.

The database of the large challenge was provided in several chunks to be downloaded more easily and we provided several data mirrors to avoid data download congestion. The data were made publicly available through the website of the challenge `http://www.kddcup-orange.com/`, with no restriction of confidentiality. They are still available to download for benchmark purpose. To protect the privacy of the customers whose records were used, the data were anonymized by replacing actual text or labels by meaningless codes and not revealing the meaning of the variables.

Extraction and preparation of the challenge data:

The Orange in-house customer analysis platform is devoted to industrializing the data mining process for marketing purpose [10]. Its fully automated data processing machinery includes: data preparation, model building, and model deployment. The data preparation module was isolated and used to format data for the purpose of the challenge and facilitate the task of the participants. Orange customer data are initially available in a relational datamart under a star schema. The platform uses a feature construction language, dedicated to the marketing domain, to build tens of thousands of features to create a rich data representation space. For the challenge, a datamart of about one million of customers was used, with about ten tables and hundreds of fields. The first step was to resample the dataset, to obtain 100,000 instances with less unbalanced target distributions. For practical reasons (the challenge participants had to download the data), the same data sample was used for the three marketing tasks. In a second step, the feature construction language was used to generate 20,000 features and obtain a tabular representation. After discarding constant features and removing customer identifiers, we narrowed down the feature set to 15,000 variables (including 260 categorical variables). In a third step, for privacy reasons, data was anonymized, discarding variables names, randomizing the order of the variables, multiplying each continuous variable by a random factor and recoding categorical variable with randomly generated category names. Finally, the data sample was split randomly into equally sized train and test sets. A random subset of 10% of the test set was designated to provide immediate performance feed-back.

5. BETA TESTS

The website of the challenge http://www.kddcup-orange.com/ was thoroughly tested by the KDD cup chairs and volunteers. The datasets were downloaded and checked. Baseline methods were tried to verify the feasibility of the task. A Matlab® version of the data was made available and sample code were provided to format the results. A sample submission of random results was given as example and submitted to the website. The results of the Naïve Bayes method were also uploaded to the website to provide baseline results.

Toy problem:

The Toy problem on the LARGE dataset consisted of one single predictive continuous variable (V5963) uniformly distributed on the interval $[0, 2.0]$. The target value was obtained by thresholding V5963 at 1.6 and adding 20% noise. Hence for 80% of the instances, lying in interval $[0, 1.6]$, the fraction of positive examples is 20%; for the remaining 20% lying in interval $]1.6, 2.0]$, the fraction of positive examples is 80%. The expected value of the AUC (called "true AUC") can easily be computed[1]. Its value is approximately 0.7206. Because of the variance in the sampling process, the AUC effectively computed using the optimal decision rule (called "optimal AUC") is 0.7196 for the training set and a 0.7230 for the test set. Interestingly, as shown in Figure 1, the optimal solution was outperformed by many participants, up to 0.7263. This illustrates the problem of multiple testing and shows how the best test performance overestimates both the expected value of the AUC and the performance of the optimal decision rule, increasingly with the number of challenge submissions.

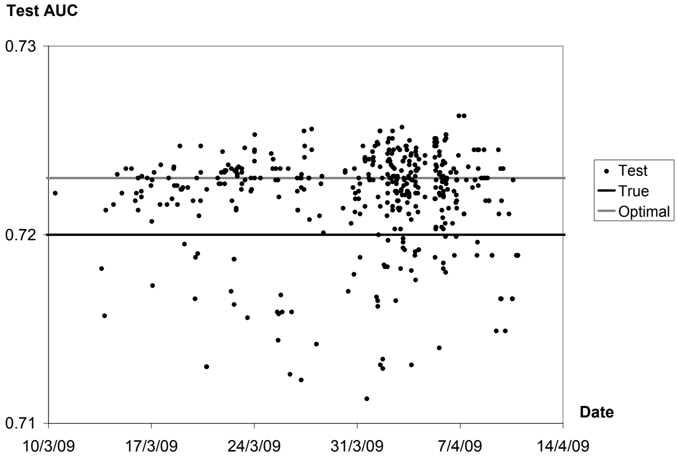


Figure 1: *Toy problem test results.*

Basic Naïve Bayes classifier:

The basic Naïve Bayes classifier (see [19]) makes simple independence assumptions between features and votes among features with a voting score capturing the correlation of the feature to the target. No feature selection is performed and there are no hyper-parameters to adjust.

For the LARGE dataset, the overall score of the basic Naïve Bayes classifier is 0.6711, with the following results on the test set:

- **Churn problem :** AUC = 0.6468;
- **Appetency problem :** AUC = 0.6453;
- **Up-selling problem :** AUC=0.7211;

As per the rules of the challenge, the participants had to outperform the basic Naïve Bayes classifier to qualify for prizes.

Orange in-house classifier:

The Orange in-house classifier is an extension of the Naïve Bayes classifier, called "Selective Naïve Bayes classifier" [3]. It includes an optimized preprocessing, variable selection, and model averaging. It significantly outperforms the basic Naïve Bayes classifier performance, which was provided to the participants as baseline, and it is computationally efficient: The results were obtained after 3 hours using a standard laptop, considering the three tasks as three different problems. The models were obtained by applying the training process Khiops® only once since the system has no hyper-parameter to adjust. The results of the in-house system were not revealed until the end of the challenge. An implementation of the method is available as shareware from http://www.khiops.com; some participants downloaded it and used it.

The requirements placed on the in-house system are to obtain a high classification accuracy, under the following constraints:

- Fully automatic: absolutely no hyper-parameter setting, since hundred of models need to be trained each month.

[1] If we call T the total number of examples, the (expected value of) the total number of examples of the positive class P is the sum of the number of positive examples in the first and the second intervals, *i.e.,* $P = (0.2 \times 0.8 + 0.8 \times 0.2)\ T = 0.32\ T$. Similarly, the total number of negative examples is $N = (0.8 \times 0.8 + 0.2 \times 0.2)\ T = 0.68\ T$. If we use the optimal decision rule (a threshold on V5963 at 1.6) the number of true positive examples is the sum of the number of true positive examples in the two intervals, *i.e.,* $TP = 0 + (0.2 \times 0.8)\ T = 0.16\ T$. Similarly, the number of true negative examples is $TN = (0.8 \times 0.8)\ T = 0.64\ T$. Hence, the true positive rate is $TPR = TP/P = 0.16/0.32 = 0.5$ and the true negative rate is $TNR = TN/N = 0.64/0.68 \simeq 0.9412$. The balanced accuracy (or the AUC because $BAC = AUC$ in this case) is therefore: $BAC = 0.5\ (TPR + TNR) = 0.5\ (0.5 + 0.9412) = 0.7206$.

- Fast to train: the three challenge marketing problems were trained in less than 3 hours on a mono-processor laptop with 2 Go RAM.
- Efficient after deployment: models need to process rapidly up to ten million instances.
- Interpretable: selected predictive variables must provide insight.

However, for the challenge, the participants were not placed under all these constraints for practical reasons: it would have been both too constraining for the participants and too difficult to enforce for the organizers. The challenge focused on maximizing accuracy under time constraints.

For the LARGE dataset, the overall score of the Orange in-house classifier is 0.8311, with the following results on the test dataset:

- **Churn problem :** AUC = 0.7435;
- **Appetency problem :** AUC = 0.8522;
- **Up-selling problem :** AUC=0.8975;

The challenge was to beat these results, but the minimum requirement to win prizes was only to outperform the basic Naïve Bayes classifier.

6. SCHEDULE AND PROTOCOL

The key elements of our design were:

- To make available the training and test data three weeks before the start of the "fast challenge" to allow participants to download the large volume of data, read it and preprocess it without the training labels.
- To make available "toy" training labels during that period so participants could finalize their methodology and practice using the on-line submission system.
- To put participants under time pressure once the training labels were released (produce results in five days) to test their ability to produce results in a timely manner.
- To continue the challenge beyond this first milestone for another month (slow challenge) to give the opportunity to participants with less computational resources to enter the challenge.
- To provide a down-sized version of the dataset for the slow challenge providing an opportunity for participants with yet less computational resources to enter the challenge.
- To provide large prizes to encourage participation (10,000 Euros donated by Orange), without any strings attached (no legal constraint or commitment to release code or methods to download data or participate).

The competition rules were inspired from previous challenges we organized [8]. The full rules are available from the website of the challenge `http://www.kddcup-orange.com/`. They were designed to attract a large number of participants and were successful in that respect: Many participants did not participate in the fast challenge on the large dataset, but entered in the slow track, either on the small or the large dataset (or both). There was one minor design mistake: the small dataset was derived from the same data as the large one and, despite our efforts to disguise the identity of the features, it was possible for some entrants to match the features and entries in the small and large dataset. This provided a small advantage, *in the slow track only*, to the teams who did that data "unscrambling": they could get feed-back on 20% of the data rather than 10%.

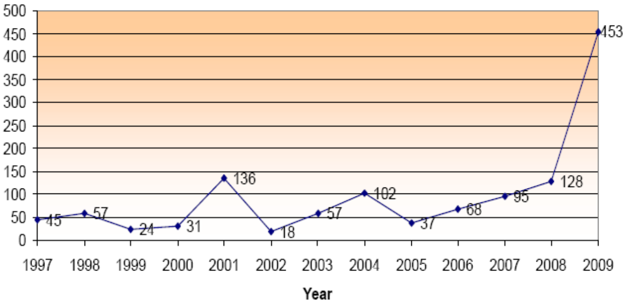


Figure 2: *KDD Cup Participation by year (number of teams).*

The schedule of the challenge was as follows (Dates in 2009):

- **March 10** - Start of the FAST large challenge. Data tables without target values were made available for the large dataset. Toy training target values were made available for practice purpose. Objective: participants can download data, ask questions, finalize their methodology, try the submission process.
- **April 6** - Training target values were made available for the large dataset for the real problems (churn, appetency, and upselling). Feed-back: results on 10% of the test set available on-line when submissions are made.
- **April 10** - Deadline for the FAST large challenge. Submissions had to be received before midnight, time zone of the challenge web server.
- **April 11** - Data tables and training target values were made available for the small dataset. The challenge continued for the large dataset in the slow track.
- **May 11** - Deadline for the SLOW challenge (small and large datasets). Submissions had to be be received before midnight, time zone of the challenge web server.

7. RESULTS

The 2009 KDD Cup attracted 1299 teams from 46 different countries. From those teams, 7865 valid entries were submitted by 453 different teams. The participation was more than three times greater than any KDD Cup in the past. Figure 2 represents the KDD Cup participation by year. A large participation was a key element to validate the results and for Orange to have a ranking of its in-house system; the challenge was very successful in that respect.

Table 1: **Prize winners.** Top 3: fast track. Bottom 3: slow track.

Team	Fast track		Slow track	
	Rk	Score	Rk	Score
IBM Research, USA	1	0.8493	1	0.8521
ID Analytics, USA	2	0.8448	3	0.8479
Slate & Frey, USA	3	0.8443	8	0.8443
U. Melbourne, Australia	27	0.8250	2	0.8484
FEG Inc., Japan	4	0.8443	4	0.8477
NTU, Taiwan	20	0.8332	5	0.8461

7.1 Winners

The overall winner is the IBM Research team [16] who ranked first in both tracks. Six prizes were donated by Orange to top ranking participants in the fast and the slow tracks (see Table 1). As per the rules of the challenge, the same team could not earn two prizes. If the ranking of a team entitled it to two prizes, it received the best of the two and the next best ranking team received the other prize.

All the winning teams scored best on the large dataset (and most participants obtained better results on the large dataset then on the small dataset). IBM Research, ID Analytics, and National Taiwan University (NTU) “unscrambled” the small dataset. This may have provided an advantage only to NTU since “unscrambling” affected only the slow track and the two other teams won prizes in the fast track. We briefly comment on the results of the winners.

Fast track:

- **IBM Research:** The winning entry [16] consisted in an ensemble of a wide variety of classifiers, following [7; 6]. Effort was put into coding (most frequent values coded with binary features, missing values replaced by mean, extra features constructed, etc.)
- **ID Analytics, Inc.:** One of the only teams to use a wrapper feature selection strategy, following a filter [22]. The classifier was built from the commercial TreeNet software by Salford Systems: an additive boosting decision tree technology. Bagging was also used to gain additional robustness.
- **David Slate & Peter Frey (Old dogs with new tricks):** After a simple preprocessing (consisting in grouping of modalities or discretizing) and filter feature selection, this team used ensembles of decision trees, similar to Random Forests [5].

Slow track:

- **University of Melbourne:** This team used for feature selection a cross-validation method targeting the AUC and, for classification, boosting with classification trees and shrinkage, using a Bernoulli loss [18].
- **Financial Engineering Group, Inc.:** Few details were released by the team about their methods. They used grouping of modalities and a filter feature selection method using the AIC criterion [1]. Classification was based on gradient tree-classifier boosting [12].
- **National Taiwan University:** The team averaged the performances of three classifiers [17]: (1) The solution of the joint multiclass problem with an L1-regularized maximum entropy model. (2) AdaBoost with tree-based weak learners [11]. (3) Selective Naïve Bayes [3], which is the in-house classifier of Orange (see Section 5).

Table 2: **Best results and baselines.** In the top table, we show the best score $TAUC^*$ (averaged over the three tasks), over increasing periods of time $[0:t]$. The best overall performance is $TAUC^{**} = TAUC^*(36d)$. The relative performance difference $\delta^* = (TAUC^{**} - TAUC^*)/TAUC^{**}$ is given in parenthesis (in percentage). The bottom table gives the relative performance difference $\delta^* = (TAUC^{**} - TAUC)/TAUC^{**}$ for the two reference results: the basic Naïve Bayes classifier (NB) and the in-house Orange system (SNB).

$TAUC$ (δ^*%)	$TAUC^*$ $12h$	$TAUC^*$ $24h$	$TAUC^*$ $5d$	$TAUC^{**}$
Churn	0.7467 (2.40)	0.7467 (2.40)	0.7611 (0.52)	0.7651 (0)
Appetency	0.8661 (2.17)	0.8714 (1.57)	0.8830 (0.26)	0.8853 (0)
Up-selling	0.9011 (0.89)	0.9011 (0.89)	0.9057 (0.38)	0.9092 (0)
Average	0.8380 (1.65)	0.8385 (1.60)	0.8493 (0.33)	0.8521 (0)

$TAUC$ (δ^*%)	$TAUC$ NB	$TAUC$ SNB
Churn	0.6468 (15.46)	0.7435 (2.82)
Appetency	0.6453 (27.11)	0.8522 (3.74)
Up-selling	0.7211 (20.69)	0.8975 (1.29)
Average	0.6711 (21.24)	0.8311 (2.46)

7.2 Performance statistics

We now turn to the statistical analysis of the results of the participants. The main statistics are summarized in Table 2. In the figures of this section, we use the following color code:

1. Blue: **Overall best submissions received**. Referred to as $TestAUC^{**}$.
2. Red: **Baseline result**, obtained with the basic Naïve Bayes classifier or NB, provided by the organizers (see Section 5). The organizers consider that this result is easy to improve. They imposed that the participants would outperform this result to win prizes to avoid that a random submission would win a prize.
3. Green: **Orange system result**, obtained by the in-house Orange system with the Selective Naïve Bayes classifier or SNB (see Section 5).

Progress in performance

A good result, better than the baseline result, was obtained **within one hour** of the start of the challenge and the in-house system was slightly outperformed **after seven hours**. The improvement during the first day of the competition, after the first 7 hours, was small: from 0.8347 to 0.8385. Over the first 5 days (FAST challenge), the performance progressed from 0.8385 to 0.8493. Considering only the submission with Test AUC > 0.5 in the first 5 days, 30% of the submissions had worse results than the baseline (basic Naïve Bayes) and 91% had worse results than the in-house system (AUC=0.8311). **Only and 9% of the submissions had better results than the in-house system**. These results, and an examination of the fact sheets of the challenge filled out by the participants, reveal that:

- There are available methods, which can process fast large databases using today's available hardware in both Academia and Industry.

- Several teams were capable of adapting their methods to meet the requirements of the challenge and reach quickly good performances, yet the bulk of the participants did not.

- The protocol of the challenge was well designed: the month given to download the data and play with the submission protocol (using the toy problem) allowed us to monitor progress in performance, not the time required to get ready for the challenge.

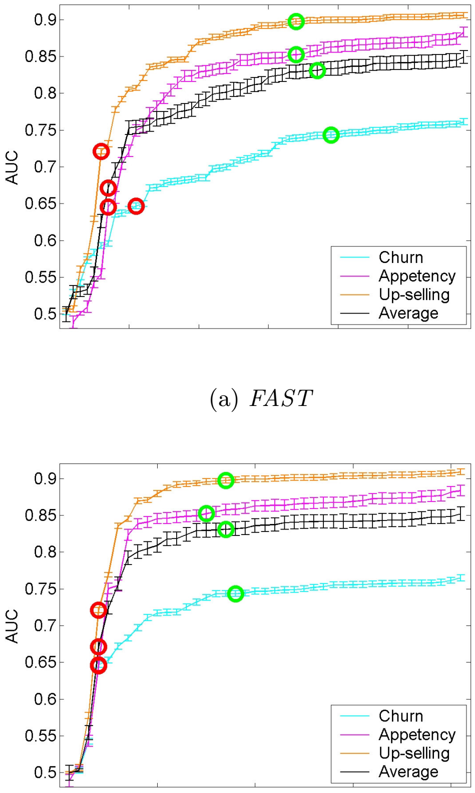
This last point was important for Orange to assess the time taken for generating state-of-the-art models, since speed of model generation is a key requirement in such applications. Furthermore, very small improvements (from 0.8493 to 0.8521) were made after the 5^{th} day (SLOW challenge)[2].

One of the aims of the KDD cup 2009 competition was to find whether there are data-mining methods which are significantly better than others. To this end we performed a significance analysis on the final results (last submission before the deadline, the one counting towards the final ranking and the selection of the prize winners) of both the SLOW and FAST track. Only final results reported on the large dataset were included in the analysis since we have realized that submissions based on the small dataset were considerably inferior.

To test whether the differences between the teams are statistically significant we followed a two step analysis that is specifically designed for multiple hypothesis testing when several independent task are involved [9]: First we used the Friedman test [13], to examine the null hypothesis H_0, which states that the AUC values of the three tasks, (Churn, Appetency and Up-Selling) on a specific track (FAST or SLOW) are all drawn from a single distribution. The Friedman test is a non-parametric test, based on the average ranking of each team, where AUC values are ranked for each task separately. A simple test-statistic of the average ranks is sufficient to extract a p-value for H_0; In the case when H_0 is rejected, we use a two tailed Nemenyi test [20] as a post-hoc analysis for identifying teams with significantly better or worse performances.

Not surprisingly, if one takes all final submissions, one finds that H_0 is rejected with high certainty (p-value $< 10^{-12}$). Indeed, significant differences are observed even when one inspects the average final AUCs (see Figure 3), as some submissions were not substantially better than random guess, with an AUC near 0.5. Of course, Figure 3 is much less informative than the significance testing procedure we adopt, which combines the precise scores on the three tasks, and not each one separately or their averages.

Trying to discriminate among the top performing teams is more subtle. When taking the best 20 submissions per track (ranked by the best average AUC) - the Friedman test still rejects H_0 with p-values 0.015 and 0.001 for the FAST and SLOW tracks respectively. However, the Nemenyi tests on these reduced data are not able to identify significant differences between submissions, even with a significance level of $\alpha = 0.1$!



(a) *FAST*

(b) *SLOW*

Figure 3: **Sorted final scores:** The sorted AUC values on the test set of each of the three tasks, together with the average of AUC on the three tasks. Only final submissions are included. (a) FAST track and (b) SLOW track. The baseline results for Basic Naïve Bayes and Selective Naïve Bayes are superposed on the corresponding tasks.

The fact that one does not see significant differences among the top performing submissions is not so surprising: during the period of the competition more and more teams have succeeded to cross the baseline, and the best submissions tended to accumulate in the tail of the distribution (bounded by the optimum) with no significant differences. This explains why the number of significant differences between the top 20 results decreases with time and number of submissions.

Even on a task by task basis, Figure 3 reveals that the performance of the top 50% AUC values lie on an almost horizontal line, indicating there are no significant differences among these submissions. This is especially marked for the SLOW track.

From an industrial point of view, this result is quite interesting. In an industrial setting many criteria have to be considered (in addition to prediction performance), including automation of the data mining process, training time, and deployment time. These put constraints on the algorithms

[2] This improvement may be partly attributed to "unscrambling"; unscrambling was not possible during the fast track of the challenge (first 5 days).

employed. In the SLOW track, the participants were largely free of such constraints and many used abundant computer and human resources. Our analysis shows that significant improvements in performance are difficult to obtain, even at the expense of a huge deterioration of the other criterions.

Rapidity of model building

Figure 4.d gives a comparison between the submissions received and the best overall result over increasing periods of time: 12 hours, one day, 5 days, and 36 days. We compute the relative performance difference

$$\delta^* = (TestAUC^{**} - TestAUC)/TestAUC^{**} \ , \qquad (2)$$

where $TestAUC^{**}$ is the best overall result. The values of δ^* for the best performing classifier in each interval and for the reference results are found in Table 2. On each box, the central mark is the median, the edges of the box are the 25^{th} and 75^{th} percentiles, the whiskers extend to the most extreme data points not considered to be outliers; the outliers are plotted individually as crosses.

The following observations can be made:

- there is a wide spread of results;
- the median result improves significantly over time, showing that it is worth continuing the challenge to give to participants an opportunity of learning how to solve the problem (the median beats the baseline on all tasks after 5 days but keeps improving);
- but the best results do not improve a lot after the first day;
- and the distribution after 5 days is not very different from that after 36 days.

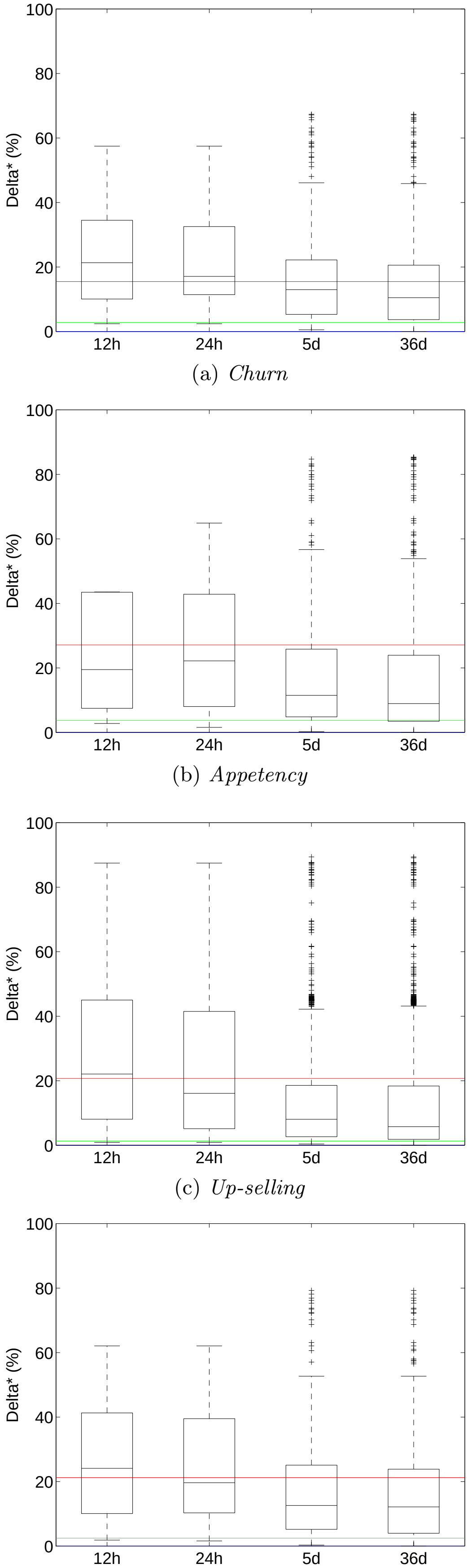
Table 2 reveals that, at the end of the challenge, for the average score, the relative performance difference between the baseline model (basic Naïve Bayes) and the best model is over 20%, but only 2.46% for SNB. For the best ranking classifier, only 0.33% was gained between the fifth day (FAST challenge) and the last day of the challenge (SLOW challenge). After just one day, the best ranking classifier was only 1.60% away from the best result. The in-house system (selective Naïve Bayes) has a result less than 1% worse than the best model after one day.

We conclude that the participants did very well in building models fast. Building competitive models is one day is definitely doable and the Orange in-house system is competitive, although it was rapidly beaten by the participants.

Individual task difficulty

To assess the relative difficulty of the three tasks, we plotted the relative performance difference δ^* (Equation 2) for increasing periods of time, see Figure4.[a-c].

The *churn* task seems to be the most difficult one, if we consider that the performance at day one, 0.7467, only increases to 0.7651 by the end of the challenge (see Table 2 for other intermediate results). Figure 4.a shows that the median performance after one day is significantly worse than the baseline (Naïve Bayes), whereas for the other tasks the median was already beating the baseline after one day.



(a) *Churn*

(b) *Appetency*

(c) *Up-selling*

(d) *Average*

Figure 4: **Performance improvement over time.**

The *appetency* task is of intermediate difficulty. Its day one performance of 0.8714 increases to 0.8853 by the end of the challenge. Figure 4.b shows that, from day one, the median performance beats the baseline method (which performs relatively poorly on this task).

The *up-selling* task is the easiest one: the day one performance 0.9011, already very high, improves to 0.9092 (less that 1% relative difference). Figure 4.c shows that, by the end of the challenge, the median performance gets close to the best performance.

Correlation between $TestAUC$ and $ValidAUC$:

There is a good correlation between the results on the test set (100% of the test set), $TestAUC$, and the results on the validation set (10% of the test set used to give a feed back to the competitors), $ValidAUC$. We computed the Pearson correlation coefficient after removing the result having a test AUC lower than 0.5 (probably corresponding to errors in submissions). We obtained 0.9960 ± 0.0005 (95% confidence interval) for the first 5 days and of is of 0.9959 ± 0.0003 for the 36 days. This seems to indicate that, (i) the validation set provided useful feedback to the participants, without compromising the test set; (ii) the participants did not overfit the validation set.[3] The analysis of correlation task by task gives the same information, on the entire challenge (36 days) the correlation coefficient is for the Churn task: 0.9860 ±0.001; for the Appetency task: 0.9875 ±0.0008 and for the Up-selling task: 0.9974 ±0.0002.

We also asked the participants to return training set prediction results, hoping that we could do an analysis of overfitting by comparing training set and test set performances. Unfortunately, because the training set results did not affect the ranking score, some participants did not return real predictions made by their classifier, but rather random results or the target labels. However, if we exclude extreme performances (random or perfect), we can observe that (i) a fraction of the models performing well on test data have a good correlation between training and test performances; (ii) there is a group of models performing well on test data and having an AUC on training examples significantly larger. Large margin models like SVMs [2] or boosting models [11] behave in this way. Among the models performing poorly on test data, some clearly overfitted (had a large difference between training and test results).

7.3 Methods employed

We analyzed the information provided by the participants in the fact sheets to determine which methods were employed to tackle the challenge:

- **Preprocessing:** Few participants did not use any preprocessing. A large fraction of the participants replaced missing values by the mean or the median or a fixed value. Some added an additional feature coding for the presence of a missing value. This allows linear classifiers to automatically compute the missing value by selecting an appropriate weight. Decision tree users did not replace missing values. Rather, they relied on the usage of "surrogate variables": at each split in a dichotomous tree, if a variable has a missing value, it may be replaced by an alternative "surrogate" variable. Discretization was the second most used preprocessing. Its usefulness for this particular dataset is justified by the non-normality of the distribution of the variables and the existence of extreme values. The simple bining used by the winners of the slow track proved to be efficient. For categorical variables, grouping of under-represented categories proved to be useful to avoid overfitting. The winners of the fast and the slow track used similar strategies consisting in retaining the most populated categories and coarsely grouping the others in an unsupervised way. Simple normalizations were also used (like dividing by the mean). Principal Component Analysis (PCA) was seldom used and reported not to bring performance improvements.
- **Feature selection:** Feature ranking and other filter methods were the most widely used feature selection methods. Most participants reported that wrapper methods overfitted the data. The winners of the slow track method used a simple technique based on cross-validation classification performance of single variables.
- **Classification algorithm:** Ensembles of decision trees were the most widely used classification method in this challenge. They proved to be particularly well adapted to the nature of the problem: large number of examples, mixed variable types, and lots of missing values. The second most widely used method was linear classifiers, and more particularly logistic regression (see *e.g.,* [15]). Third came non-linear kernel methods (*e.g.,* Support Vector Machines, [2]). They suffered from higher computational requirements, so most participants gave up early on them and rather introduced non-linearities by building extra features.
- **Model selection:** The majority of the participants reported having used for model selection the on-line performance feed-back on 10% of the test set (called here *validation set*), at least to some extent. However, after analyzing the variance in performance estimation with such a small portion of the test data, many participants preferred using cross-validation (ten-fold or five-fold) to select hyper-parameters and perform model selection, to avoid overfitting the validation set. Model selection was to a large extent circumvented by the use of ensemble methods. Three ensemble methods have been mostly used by top ranking participants: boosting [11; 12], bagging [4; 5], and heterogeneous ensembles built by forward model selection [7; 6].

Surprisingly, less than 50% of the teams reported using regularization [21]. Perhaps this is due to the fact that many ensembles of decision trees do not have explicit regularizers, the model averaging performing an implicit regularization. The wide majority of approaches were frequentist (non Bayesian). Little use was made of the unlabeled test examples for training and no performance gain was reported.

We also analyzed the fact sheets with respect to the software and hardware implementation:

- **Hardware:** While some teams used heavy computational apparatus, including multiple processors and

[3] Many participants took the precaution of using cross-validation on training data for model selection, rather than using the validation set performance, to avoid overfitting the validation set.

lots of memory, the majority (including the winners of the slow track) used only laptops with less than 2 Gbytes of memory, sometimes running in parallel several models on different machines. Hence, even for the large dataset, it was possible to provide competitive solutions with inexpensive computer equipment. In fact, the in-house system of Orange computes its solution in less than three hours on a laptop.

- **Software:** Even though many groups used fast implementations written in C or C++, packages in Java (Weka) and libraries available in Matlab® or "R", presumably slow and memory inefficient, were also widely used. Users reported performing first feature selection to overcome speed and memory limitations. Windows was the most widely used operating system, closely followed by Linux and other Unix operating systems.

8. CONCLUSION

The results of the KDD cup 2009 exceeded our expectations in several ways. First we reached a very high level of participation: over three times as much as the most popular KDD cups so far. Second, the participants turned in good results very fast: within 7 hours of the start of the FAST track challenge they exceeded the baseline provided by Orange. The performances were only marginally improved in the rest of the challenge, showing the maturity of data mining techniques. Ensemble of decision trees offer off-the-shelf solutions to problems with large numbers of samples and attributes, mixed types of variables, and lots of missing values. Ensemble methods proved to be effective for winning, but single models are still preferred by many customers. Further work include matching the performances of the top ranking participants with single classifiers.

8.1 Acknowledgements

We are very grateful to the Orange company who donated the data, the computer servers, many hours of engineer time, and the challenge prizes. We would like to thank all the Orange team members, including Fabrice Clérot and Raphael Féraud. We also gratefully acknowledge the ACM SIGKDD and the Pascal2 European network of excellence who supported the work of the KDD Cup 2009 co-chairs Isabelle Guyon and David Vogel and the web site development. The support of Google and Health Discovery Corporation allowed students to attend the workshop. We are very grateful to the technical support of MisterP and Pascal Gouzien.

9. REFERENCES

[1] H. Akaike. Information theory and an extension of the maximum likelihood principle. In B. Petrov and F. Csaki, editors, *2nd International Symposium on Information Theory*, pages 267–281. Akademia Kiado, Budapest, 1973.

[2] B. E. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *COLT*, pages 144–152, 1992.

[3] M. Boullé. Compression-based averaging of Selective Naïve Bayes classifiers. *JMLR*, 8:1659–1685, 2007.

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

[5] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[6] R. Caruana, A. Munson, and A. Niculescu-Mizil. Getting the most out of ensemble selection. In *Proceedings of the 6th International Conference on Data Mining (ICDM '06)*, December 2006. Full-length version available as Cornell Technical Report 2006-2045.

[7] R. Caruana and A. Niculescu-Mizil. Ensemble selection from libraries of models. In *Proceedings of the 21st International Conference on Machine Learning (ICML'04)*, 2004.

[8] Clopinet. Challenges in machine learning.

[9] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, 7:1–30, 2006.

[10] R. Féraud, M. Boullé, F. Clérot, F. Fessant, and V. Lemaire. The orange customer analysis platform. In *JMLR W&CP*, volume 7, KDD cup 2009, Paris, 2009.

[11] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *ICML*, pages 148–156, 1996.

[12] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2000.

[13] M. Friedman. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, 32:675–701, 1937.

[14] I. Guyon, A. Saffari, G. Dror, and J. Buhmann. Performance prediction challenge. In *IEEE/INNS conference IJCNN 2006*, Vancouver, Canada, July 16-21 2006.

[15] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning, Data Mining, Inference and Prediction.* Springer Verlag, 2000.

[16] IBM Research. Winning the KDD cup orange challenge with ensemble selection. In *JMLR W&CP*, volume 7, KDD cup 2009, Paris, 2009.

[17] H.-Y. Lo et al. An ensemble of three classifiers for KDD cup 2009: Expanded linear model, heterogeneous boosting, and selective naïve Bayes. In *JMLR W&CP*, volume 7, KDD cup 2009, Paris, 2009.

[18] H. Miller et al. Predicting customer behaviour: The University of Melbourne's KDD cup report. In *JMLR W&CP*, volume 7, KDD cup 2009, Paris, 2009.

[19] T. Mitchell. *Machine Learning.* McGraw-Hill Co., Inc., New York, 1997.

[20] P. B. Nemenyi. *Distribution-free multiple comparisons.* Doctoral dissertation, Princeton University, 1963.

[21] V. Vapnik. *Statistical Learning Theory.* John Wiley and Sons, N.Y., 1998.

[22] J. Xie et al. A combination of boosting and bagging for KDD cup 2009 - fast scoring on a large database. In *JMLR W&CP*, volume 7, KDD cup 2009, Paris, 2009.