PAKDD Data Mining Competition 2009: New Ways of Using Known Methods

Chaim Linhart[1], Guy Harari[1], Sharon Abramovich[2], and Altina Buchris[2]

[1] School of Computer Science, Tel Aviv University, Tel Aviv 69978, Israel
chaiml@post.tau.ac.il
[2] Department of Statistics and Operations Research, Tel Aviv University,
Tel Aviv 69978, Israel

Abstract. The PAKDD 2009 competition focuses on the problem of credit risk assessment. As required, we had to confront the problem of the robustness of the credit-scoring model against performance degradation caused by gradual market changes along a few years of business operation. We utilized the following standard models: logistic regression, KNN, SVM, GBM and decision tree. The novelty of our approach is two-fold: the integration of existing models, namely feeding the results of KNN as an input variable to the logistic regression, and re-coding categorical variables as numerical values that represent each category's statistical impact on the target label. The best solution we obtained reached 3rd place in the competition, with an AUC score of 0.655.

Keywords: data mining, logistic regression, KNN, credit risk assessment.

1 Introduction

The offer of credit for potential clients is a very important service for stimulating consumption in the market. One main difficulty credit scoring modelers have to contend with is gradual market changes which occur during the collection of data. This difficulty increases the risk when the credit is lent for long term payment.

The PAKDD 2009 data mining competition focused on the model's robustness against performance degradation caused by market gradual changes along several business years [1]. We participated in this competition as part of the requirements of the Data Mining course given by Professor Yoav Benjamini in Tel-Aviv University.

The challenge was as follows. We were given three datasets, which were collected over different years and consist of 30 explanatory variables and one binary target variable. The first dataset, which was used for model selection, is labeled and contains 50,000 samples collected during 2003. The second dataset consists of 10,000 unlabeled samples collected during 2005 and was used for model evaluation. After selecting a model we could apply it on this dataset, submit the results to the leaderboard web-site, and compare its performance to the scores attained by other teams. The third dataset, called the prediction data, consists of 10,000 unlabeled samples from 2008, and was used for grading the performance of the final models of all competitors. Performance of each model was evaluated by area under ROC curve (AUC, in short).

T. Theeramunkong et al. (Eds.): PAKDD Workshops 2009, LNAI 5669, pp. 99–105, 2010.

2 Data Preparation

Initial observations revealed that some of the explanatory variables are not useful for analysis, since they are constant in either the modeling or prediction data. A small number of samples in the modeling dataset have unreasonable or missing values, so we ignored them. We replaced unreasonable and missing values in the prediction data, as detailed below. We also tried to remove samples with area and profession codes that are absent from the leaderboard or prediction data. This gave better results on the modeling data, but performed worse on the leaderboard dataset, so we abandoned this approach.

We noticed that some variables have a significantly different distribution in the modeling data than in the leaderboard and/or prediction data. For example, *AREA_CODE_RESIDENCIAL_PHONE* is “50” in 22%, 5%, and 15% of the samples in the modeling, leaderboard, and prediction datasets, respectively. Another example is *PAYMENT_DAY*, which receives the value “15” in only 0.2% of the modeling samples, and 21% of the prediction samples. These differences might lead to degraded performance on the prediction data – the models are fitted to data with certain characteristics, and tested on data with different distributions.

Numerical variables: Unreasonable values, such as age 0 or extremely high income, were replaced by the median value of the corresponding variable. In order to account for possible changes in the value of the local currency over time (e.g., due to inflation), we standardized the two income variables to mean 0 and standard deviation 1. We also experimented with other transformations, such as logarithm and square root.

We noticed that some samples contain 0 in the income variables *PERSONAL_NET_INCOME* and *MATE_INCOME*. The distribution of the target variable suggests that at least some of these values do not really represent zero income. For example, when *PERSONAL_NET_INCOME* is 0, the target variable is 0 in 83% of the cases; when the income is 50-150, it’s 77%; for 150-250, it’s 76%; and for 350-450 (approximately the mean income) it’s 79%. This suggests that a value of zero indicates either no income, or a missing value. Therefore, as with several other variables, we replaced the 0’s by the mean value (not including 0’s).

Textual variables: We replaced the two personal reference textual variables by a single numerical variable that holds the sum of their lengths. This was done since we discovered a relationship between the length of the personal references and the target variable.

Categorial variables: Boolean variables and categorical variables with a small number of categories (such as *MARITAL_STATUS*) are easily handled by all the models we applied – each category is replaced by several boolean indicator variables, one per category. Variables with a large number of categories, such as *ID_SHOP* and *PROFESSION_CODE*, pose a difficult challenge. We first added indicator variables for the most frequent values of each such variable. However, using many such variables in a logistic regression, for example, is prone to over-fitting. On the other hand, using only a small number of the indicator variables utilizes the information in the corresponding categories while effectively ignoring the information in the rest of the categories. We thus developed a method for transforming these categorial variables into numerical variables in a similar approach taken in [2]. These variables are called here "*P-VAL* variables" and are described in what follows.

The target variable *TARGET_LABEL_BAD* gets the values 0 (good) and 1 (bad) in 40,105 and 9,868 (legal) modeling samples, respectively. Given a categorial variable *X*, we compared the distribution of *TARGET_LABEL_BAD* in each category of *X* to that of the entire data. We tried the following three transformations:

I. “Probs”: The proportion of 0's (good clients) among each category, that is, for a sample with $X=c$ we replaced the category c by the fraction of 0’s in *TARGET_LABEL_BAD* among all the samples with $X=c$.

II. “P-values”: The probability to obtain at least/most the observed number of *TARGET_LABEL_BAD*=0 in a category, given the total number of 0’s and 1’s in *TARGET_LABEL_BAD*. Assume there are K samples with $X=c$, out of which K_1 have *TARGET_LABEL_BAD*=1 and K_0 have *TARGET_LABEL_BAD*=0. We can view these K samples as a series of samples from the whole set of samples without replacement, and thus we may use the hypergeometric distribution to test whether the K samples were drawn randomly from the entire set. We use a two-sided test to detect a tendency both to 0 and to 1. In order to preserve this information in the numerical variable, we replaced categories with a tendency to 0 by the above p-value, and categories with many 1’s by one minus the p-value.

III. “Logit”: As in II, but taking the logit of the p-value for categories with over-representation of 0’s in *TARGET_LABEL_BAD*, and taking –logit(p-value) for categories with tendency to 1’s. Thus, categories with a similar 0/1 distribution to that of the entire dataset, as well as very rare categories (that are present in only a couple of samples), are replaced by values close to 0. Categories in which there are statistically many samples with *TARGET_LABEL_BAD*=0 are replaced by very small (negative) values. Likewise, categories with a strong tendency for *TARGET_LABEL_BAD*=1 are replaced by large (positive) values.

Table 1. Main pre-processing steps performed on the data

The problem/issue	Variables involved	Our solution
Constant in modeling or prediction data	QUANT_BANKING_ACCOUNTS FLAG_MOBILE_PHONE FLAG_CONTACT_PHONE COD_APPLICATION_BOOTH FLAG_CARD_INSURANCE_OPTION FLAG_OTHER_CARD QUANT_DEPENDANTS EDUCATION	Omit variables
Illegal values in modeling data	SEX	Remove samples with illegal values
Unreasonable values	AGE MONTHS_IN_RESIDENCE PERSONAL_NET_INCOME MATE_INCOME	Replace unreasonable values with the median of the variable

Table 1. (*continued*)

Different categories and distribution of values in model and prediction datasets	SHOP_RANK	Omit the variable
Categorial variable with many categories	AREA_CODE_RESIDENCIAL_PHONE PROFESSION_CODE ID_SHOP PAYMENT_DAY	1. Create indicator variables for the most frequent categories 2. Transform to numerical variables using one of three methods: Probs, P-values, Logit
Currency changes over time (inflation)	PERSONAL_NET_INCOME MATE_INCOME	Standardize the variables to mean 0 and std 1
Textual variables	PERSONAL_REFERENCE_1 PERSONAL_REFERENCE_2	Replace by sum of lengths

3 Modeling

We used the cross validation approach (5-fold CV) to estimate the performance of our models. Note, however, that since the modeling, leaderboard and prediction datasets were not sampled from the same distribution, better performance on the modeling data does not guarantee improved results on the other two datasets.

Logistic model: We fitted logistic models using the glm() function in R [3], starting with single variables, and went on to include interactions between variables. We found that using our transformation of categorial variables into numerical variables solves the difficulty of ranking the categories - which is necessary for a monotonous relation, as the one the model tries to fit.

KNN: We implemented our own KNN function. For each test sample, it first identifies the training samples with: (a) The same sex, (b) The same marital status, (c) A similar age (ages different by less than some predefined threshold), and (d) A similar income (salaries that are bounded from both sides by some pre-defined multiplicative factor). It then computes the distance between the test sample and each of these training samples, using different weights for the various variables. It is worth mentioning that different types of variables require a different distance metric. For numeric variables we used the Euclidean distance, whereas categorial variables got a zero weight when levels were equal and some positive weight otherwise. Finally, the procedure reports the fraction of the k nearest neighbors with *TARGET_LABEL_BAD=1*, as well as the logit of its p-value (as described above for the "P-VAL" variables).

Logistic + KNN combined model: We combined the KNN and logistic models by feeding the results of KNN as input to the logistic model. In other words, we added two new variables, called *KNN_PROBS* and *KNN_PROBS_PVALS*, that contained the results of our KNN procedure in “Probs” and “Logit” transformation, respectively (notice that the KNN procedure was executed on both the training and test sets, in order to obtain the value of the two aforementioned variables for all samples – the training samples, to which the logistic model is fitted, and the test samples, on which it is tested).

We also experimented with several other models, such as decision tree, SVM (support vector machines) and GBM (generalized boosted models) as implemented in R [3]. However, they did not yield good results.

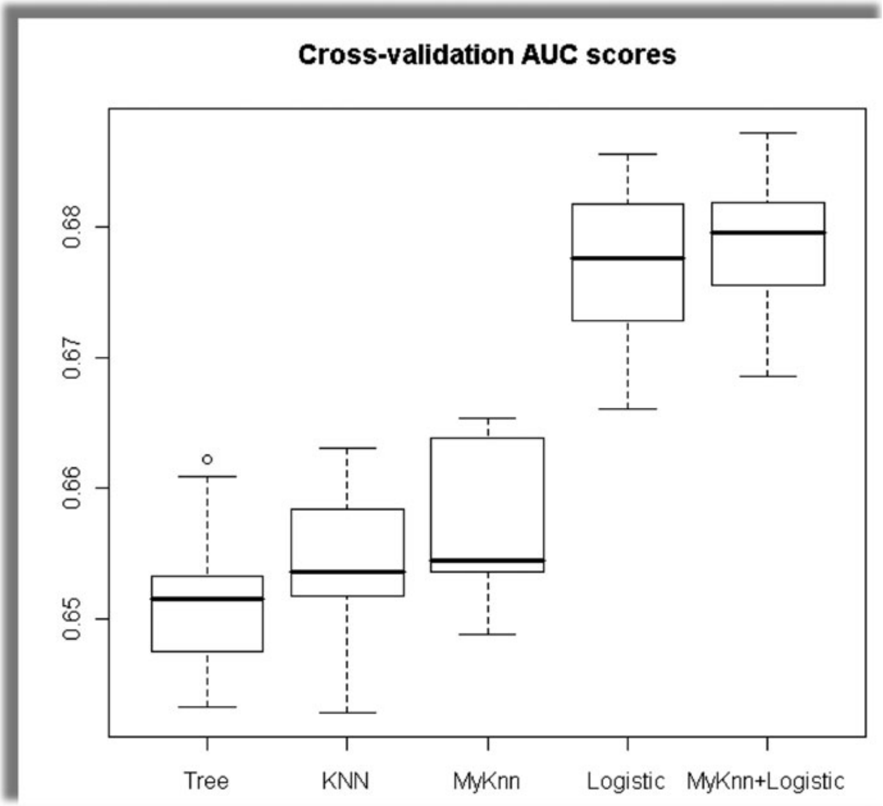

Fig. 1. AUC scores of the main models we studied. Scores were obtained using two iterations of 5-fold cross validation tests on the entire modeling dataset. "MyKnn" refers to our implementation of KNN. "MyKnn + Logistic" is the combined model.

4 Results

Logistic model: The logistic model gave reasonable results on the leaderboard data. Once we included the "P-VAL" variables, the score improved further. Interestingly, when we tested which type performs best, the results of the cross validation procedure indicated that the transformation of type "Probs" outperforms the others. However, the "Logit" transformation yielded the best score on the leaderboard dataset. A possible explanation is that the exact ratio of 0's and 1's in the target variable change over time, whereas statistically significant tendencies do not. The best logistic model we obtained reached an AUC score of 0.677 on the modeling data (see Figure 1), and 0.6125 on the leaderboard data.

An important observation is that a logistic model with many variables tended to return degraded results on the leaderboard data, even though it improved the results as assessed by the cross validation procedure on the modeling data. This might indicate an over-fitting of the model to specific characteristics of the modeling data, which change over time (recall that the leaderboard samples are two years after the modeling samples).

KNN: Our KNN procedure with *k*=250 attained an AUC score of 0.654 on the modeling data (Fig. 1) – less promising that the logistic model. However, the two models received the same score on the leaderboard data. Surprisingly, this was achieved by our KNN implementation using cutoffs and weights that were set by mere intuition on which variables are more important for predicting the target variable. Due to lack of time, we did not implement any procedure for optimizing the parameters of the KNN model. However, based on a couple of experiments, we believe that small changes to these parameters have very little effect on the results.

Logistic + KNN combined model: Combining the two models, as explained above, gave the best results. Our final logistic model consisted of 43 variables, including two variables that contained the results of our KNN procedure (*KNN_PROBS* and *KNN_PROBS_PVALS*), seven "P-VAL" variables (of type “Logit”) and two indicator variables for frequent categories (area code 31 and profession 950); the rest of the variables were original variables (after the transformations we applied) and interactions between several pairs of variables (e.g., all pairwise interactions of *AGE*, *SEX*, and *MARITAL_STATUS*). The AUC score of the final model is 0.68 on the modeling data, 0.6177 on the leaderboard data and 0.655 on the prediction data – which is ranked 3rd in the competition.

5 Conclusions and Summary

We conclude that both KNN and logistic models describe the data quite well. However, these results may be misleading since the long execution time of KNN compelled us to attempt it with very few combinations of parameters and variables. Also, since we have limited experience with SVM and GBM, we cannot conclude whether they can or cannot model the data in the competition as well as KNN and logistic regression. Interestingly, the logistic model attained higher scores than the KNN approach in the CV test on the modeling data (see Figure 1), but both methods performed equally well on the leaderboard data, indicating perhaps that the logistic model is more over-fitted to the characteristics of the modeling data than KNN. Combining KNN with a logistic model gave the best results in our experiments.

Some variables that may have some influence on the target variable were omitted from our analysis for technical reasons, such as different names of categories between the modeling and prediction data. Replacement of missing or unreasonable values could be performed by a more suitable procedure, such as maximum-likelihood based methods.

We believe that the method we described for transforming categorial variables into numerical variables, as well as our combination of KNN with logistic regression, are interesting and could be applied on other datasets. Another interesting approach could be feeding the logistic model with results from other models, such as SVM or neural networks. Due to lack of time, we paid little attention to the issue of feature selection, which could have enhanced the performance of our models.

Acknowledgments. We gratefully acknowledge the help, support and advice of Prof. Yoav Benjamini from the department of Statistics and Operations Research at Tel Aviv University. The presentation of this work was supported by the Edmond and Beverly Sackler Chair in Bioinformatics, the school of Computer Science, and the Sackler Faculty of exact sciences, all in Tel Aviv University.

References

1. PAKDD data mining competition 2009, Credit risk assessment on a private label credit card application (2009), http://sede.neurotech.com.br/PAKDD2009
2. Ritchie, M.D., Hahn, L.W., Roodi, N., Bailey, L.R., Dupont, W.D., Parl, F.F., Moore, J.H.: Multifactor-dimensionality reduction reveals high-order interactions among estrogen-metabolism genes in sporadic breast cancer. Am. J. Hum. Genet. 69(1), 138–147 (2001)
3. R Development Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria (2009), http://www.R-project.org