

University of Groningen

Classification System for Mortgage Arrear Management

Sun, Zhe; Wiering, Marco; Petkov, Nicolai

Published in:

IEEE Computational Intelligence for Financial Engineering and Economics

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version

Early version, also known as pre-print

Publication date:

2014

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Sun, Z., Wiering, M., & Petkov, N. (2014). Classification System for Mortgage Arrear Management. In IEEE Computational Intelligence for Financial Engineering and Economics IEEE (The Institute of Electrical and Electronics Engineers).

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Classification System for Mortgage Arrear Management

Z. Sun*, M.A. Wiering[†], N. Petkov*

* Johann Bernoulli Institute of Mathematics and Computing Science, University of Groningen, Groningen, the Netherlands

[†] Institute of Artificial Intelligence and Cognitive Engineering, University of Groningen, Groningen, the Netherlands

Abstract—Due to the economic recession in the recent years, more and more mortgage customers default on the payments. This brings tremendous losses to banks and forces their arrear management departments to develop more efficient processes. In this paper, we propose a classification system to predict the outcome of a mortgage arrear. Each customer who delays a monthly mortgage rate payment is assigned a label with two possible values: a delayer, who will pay the rate before the end of the month, and a defaulter, who will fail to do so. In this way, the arrear management department only needs to treat defaulters intensively. We use arrear history records obtained from a data warehouse of one Dutch bank. We apply basic classifiers, ensemble methods and sampling techniques to this classification problem. The obtained results show that sampling techniques and ensemble learning improve the performance of basic classifiers considerably. We choose balanced random forests to build the ultimate classification system. The resulting system has already been deployed in the daily work of the arrear management department of the concerned bank, and this leads to huge cost savings.

Index Terms—arrear management, classification, sampling, ensemble learning, balanced random forests

I. INTRODUCTION

Mortgages are one of the main products in retail banking. While getting profit from mortgage loan interest, banks also take the risks that mortgage customers can default on the payment. Arrears bring banks tremendous costs, e.g., interest losses, loan-loss provision and expected losses. The arrear management departments of banks are in charge of restructuring, recovering and collecting the arrears of mortgage payments. Commonly, they start tracing the customers when they have had arrears for a certain period. All customers in arrears will be treated in the same way: letters, emails or SMS will be sent with the purpose of drawing the attention of the customers; if the customers still fail to pay, they will be reached by phone calls so that the reasons of an arrear can be figured out and further treatments can be executed such as rescheduling the payment, fining or collecting the mortgages.

Due to the economic recession in the Netherlands in the recent years, more and more Dutch mortgage customers experience financial distress and default on their mortgage payments [1], which pushes the arrear management departments to adopt more efficient strategies and processes. One possible approach is treating the customers differently. In the Netherlands, the Dutch loan-loss provision (“Mutatie in voorzieningen” in Dutch) regulates that if a mortgage customer misses more than one monthly payment, the bank has to reserve a certain

percentage of the potential collection loss of the mortgage as a guarantee, which means customers who default for the short term are less harmful to banks than ones who default on more than one monthly payment. We define two kinds of arrear customers based on the duration of the arrears: *delayers*, who do not stay in arrears longer than one month, and *defaulters*, who have arrears longer than one month. Most of the mortgage customers are delayers just because they have temporary financial constraints or even simply forget to pay, while a minority of the customers are defaulters. If delayers and defaulters can be predicted accurately, the arrear management departments can only contact the defaulters intensively as soon as they are in arrears, while giving the delayers loose treatments. This would save considerable costs.

In this paper, we report on the design of an automatic system for the classification of customers who fail to pay on the due date at the beginning of a month. Based on a set of customer features, i.e. attributes that characterize the customer, the system will classify him either as a delayer, i.e. one who will pay before the end of the month, or as a defaulter, i.e. one who will fail to do so. We use arrear history records obtained from a data warehouse of one Dutch bank, and assign around 2,000 features to each customer. Feature selection and data preprocessing are executed first. Then, we test and compare several popular basic classifiers such as k-nearest neighbours (KNN), Naive Bayes, decision trees, logistic regression, and also some ensemble methods like bagging, random forests, boosting, voting and stacking. Since the two classes are highly imbalanced with the ratio of defaulters to delayers being around 1:9, sampling techniques are employed. We also consider cost analysis and feature importance.

This paper consists of five sections. Section II provides a literature review about classification techniques that are used in banking. Section III outlines the data, the classifiers and the assessment metrics. Section IV compares the results achieved with various classifiers and contains the cost matrix analysis. Section V states the conclusion.

II. LITERATURE REVIEW

Statistics and machine learning have been widely adopted in banking for decades. The most popular and successful application is credit scoring, which was first used by Altman to predict the default risk of firms in 1968 [2]. In mortgage

management, there are also some applications, such as mortgage default factors analysis and visualization [3], mortgage customer default classification [4] [5], and mortgage risk management [6]. All these researches address the problem if a given mortgage customer will default or not. Compared to previous work, our study is the first academic study of short period behaviour prediction of arrear customers. The surprising absence of previous such studies stems probably from a lack of motivation to optimize the working process of the arrear management before the global economic slowdown in the recent years.

At the beginning of applying prediction or classification systems in banking, researchers focused on statistical or operations research methods, including discriminant analysis, linear regression and linear programming. Gradually, more and more machine learning approaches were imported into this field [7]. Basic classifiers, such as case-based reasoning, Naive Bayes, decision trees and logistic regression have already been successfully applied to various applications, e.g., [4], [8]–[11]. Although bagging [12], random forests [13], AdaBoost [14] and other ensemble techniques have great success in the machine learning community in the recent ten years, ensemble learning seems not to draw enough attention in banking. For example, Ngai et al. investigated the techniques in financial fraud detection [15] and only one out of 75 articles between 1997 and 2008 used an ensemble method. In this paper, we use both basic classifiers and ensemble methods and we determine which method gives the best results for the application at hand.

The two classes in the concerned application are imbalanced. Nowadays, it has been the common understanding in the machine learning community that most traditional machine learning methods are affected by imbalanced data [16]–[19]. The ways to overcome the problem of class imbalance are of different levels according to the phases in learning, i.e., data level methods for handling imbalance, which create changing class distributions mainly by re-sampling techniques and feature selection, classifier level methods by manipulating classifiers internally and ensemble learning level methods [20]. Among these methods, sampling methods seem to be the dominant approach as changing class distributions is the most natural and straightforward solution [21]. Random undersampling and random oversampling are the most basic sampling methods. Undersampling eliminates majority-class examples while oversampling duplicates minority-class examples randomly. Both sampling techniques decrease the overall level of class imbalance, thereby making the minority class examples less rare. Synthetic sampling with data generation techniques has also attracted much attention. The synthetic minority oversampling technique (SMOTE) algorithm is the most popular approach, which oversamples by introducing new, non-replicated minority class examples [22].

III. METHODS AND DATA SET

A. The Data

This study explores arrear history records obtained from a data warehouse of a Dutch bank. The data cover the period

from November 2011 to March 2013, a total of around 420,000 anonymous observations (one customer might correspond to multiple observations, because he/she might be in and out of arrears repeatedly). A label of either delayer or defaulter is assigned to each observation according to whether the customer stays in arrears less or longer than one month, respectively. The ratio of the number of defaulters to delayers is around 1:9. The initial customer characterization contains around 2,000 features, which cover personal information, mortgage information and payment records, other products such as bank account and credit card, and some external data.

B. The Classification System

Figure 1 illustrates the workflow of the classification system. It consists of data selection, preprocessing, classification and evaluation blocks. We will describe them one by one.

After data collection and aggregation, we select appropriate features. Before selecting features by using a machine learning approach, we use domain knowledge to come to a better set of ad-hoc features [23]. Table I shows some empirical reasons why customers stay in arrears, which come from the investigation of customer service clerks in the arrear management department. The corresponding features in the right column in the table will be employed in the system regardless of the result of automatic feature selection.

TABLE I
DOMAIN KNOWLEDGE ON THE REASONS OF DEFAULT AND
CORRESPONDING FEATURES.

Reasons	Features
Lost job, or the self-employed company has problems	Salary; median salary in the last 7 months; unemployment status; unemployment benefit;
Divorce or separation	The status of marriage; the change in marital status;
The customer buys a second house, but has not sold the first one.	Number of mortgages; National mortgage guarantee.
Has to pay other debts	The balance of credit card;
Other extreme expenditures	Large amount cash withdrawals; large amount money transactions;
Higher monthly mortgage payment	Interest rate of the debt
If customers were in arrears once, they are more prone to be distressed again.	History of arrears in last 3/6/12 months;
There is a high risk of default when a customer borrows a loan larger than the appraisal value of the property. He is more likely to default when equity decreases.	LTV (Loan To Value ratio)
There is a high risk of default when a customer borrows a large loan compared to his/her income.	LTI (Loan To Income ratio)

The filter method is a feature selection method which is independent of the learning algorithm that is adopted to build a classifier. All input variables are ranked on the basis of their utility for meeting the classification goal using statistical tests [24]. The filter method is computationally convenient especially for large data sets (we use 2,000 initial features for around 420,000 customer cases). Common feature ranking techniques are information gain, Gini-index, relief,

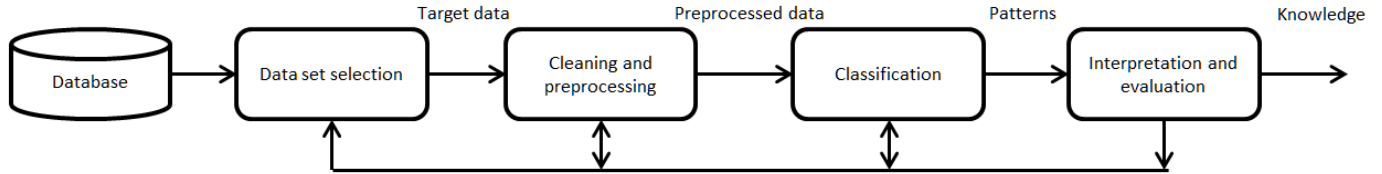


Fig. 1. The work flow of the classification system design.

χ^2 , correlation criterion, etc. In our system, we adopt the weighted voting approach of [25]: Consider an ensemble E consisting of s feature selectors, $E = \{F_1, F_2, \dots, F_s\}$. Each F_i provides a feature ranking $f_i = (f_i^1, \dots, f_i^N)$, and the individual rankings of the different selectors are aggregated into a consensus feature ranking f by equal weighted voting:

$$f^l = \sum_{i=1}^s w(f_i^l)$$

where $w(\cdot)$ denotes a weighting function. In the first selection step, we choose information gain, Gini-index and χ^2 as basic rankers and use equal weights.

This procedure results in the selection of 100 features, which include around 20 domain knowledge features. We perform the necessary data cleaning, because discrepancies, inconsistencies and missing data always exist in real banking databases. Then, missing values imputation, discretization, normalization and scaling are performed before the data is fed into a classifier. Next, the data set is used to train classifiers that can predict the labels of arrear customers. Each classifier is then tested with a test data set to evaluate its performance. At last, some classifiers can be translated into rules or meaningful business knowledge such as cost analysis and feature importance so that they can be applied into business processes.

C. Experiments

We apply several basic classifiers, such as case based reasoning (CBR), Naive Bayes (NB), decision trees (DT) and logistic regression (LR). Then we explore the impact of sampling techniques, ensemble methods and balanced ensemble methods. Next, we find the best classifier and select it for the classification system. At last, feature importance analysis and cost matrix analysis are investigated.

- 1) Basic classifiers and sampling technique: four basic classifiers are tested first. Then, three types of sampling methods: random undersampling, random oversampling and SMOTE are employed to comparatively study what the classification performances are.
- 2) Ensemble methods: we first study the impact of bagging on the basic classifiers. Bagging is configured with 50 bootstrap samples. Then, random forests with 50 trees and AdaBoost with 50 boosting iterations are tested to compare with the performance of bagging. Decision stump, also called 1-rules [26], is used in conjunction with AdaBoost.
- 3) Balanced ensemble methods: sampling techniques with ensemble methods have arisen as a possible solution to

the class imbalance problem [27]. We test symmetric bagging [28], balanced random forests [29], EasyEnsemble [30] and BalanceCascade [30]. The configurations of these methods are: symmetric bagging with 50 bootstrap samples, balanced random forests with 50 trees, both EasyEnsemble and BalanceCascade with 50 bootstrap samples and 20 boosting iterations.

- 4) Finding the best classifier: according to the performance, we will select the best performing, robust, meaningful and fastest classifier in the classification system.
- 5) Cost matrix analysis is executed in order to estimate the expected cost reduction and decide on the optimal classification. We also analyse the feature importance for a better understanding of which characteristics are most important.

The normal k-fold cross validation would bring the situation that the distribution of defaulters and delayers in each fold are different. In order to reduce the deviation of the test results in different folds, stratified k-fold cross validation is adopted to ensure the numbers of instances in both majority and minority class are strictly equal in each fold. Following the common practise, we use 10 folds in our experiments.

D. Assessment Metrics

Classification performance can be formulated by a confusion matrix, as illustrated in Table II. Singular assessment metrics such as accuracy, precision, recall, F-measure and G-means can be computed from a confusion matrix. They are frequently used in two-class classification problems. Among these metrics, the F-measure is defined as

$$F_\beta = \frac{(1 + \beta^2) \cdot Recall \cdot Precision}{\beta^2 \cdot Recall + Precision},$$

where β is a coefficient to adjust the relative importance of precision versus recall (usually, $\beta = 1$). We will use the F_1 -measure to evaluate to what extent one classifier is biased towards the majority class.

TABLE II
CONFUSION MATRIX FOR PERFORMANCE EVALUATION

		Predicted class	
		Defaulter	Delayer
Actual class	Defaulter	true positive (TP)	false negative (FN)
	Delayer	false positive (FP)	true negative (TN)

However, the singular metrics are not suitable to compare holistic performance of different classifiers in an imbalanced classification problem [31]. In this paper, we use the Receiver

Operating Characteristic (ROC) curve and the Area under the Curve (AUC) value as the assessment metrics to compare different classifiers. An ROC graph plots the true positive rate on the y-axis versus the false positive rate on the x-axis. One confusion matrix corresponds to one point on the ROC graph. Changing the decision threshold value means moving from one point to another point, and by traversing all threshold, an ROC curve is generated. An ROC curve does not assume any particular misclassification costs or class prior probabilities. The area under the curve (AUC), is a common method to convert the ROC curve to a single scalar representing performance. The AUC value is always between 0 and 1. In general, the AUC gives a general idea of the predictive potential of a classifier. A higher AUC value indicates a better average performance. We want to note that the AUC measure has been criticized, especially when it is used for problems with large class imbalance as in our case [32]. Furthermore, some alternative metrics have been proposed in literature, such as the AUK [33]. We still used the AUC metric, because it is well known and we do not think our conclusions would change significantly when using a newer assessment metric.

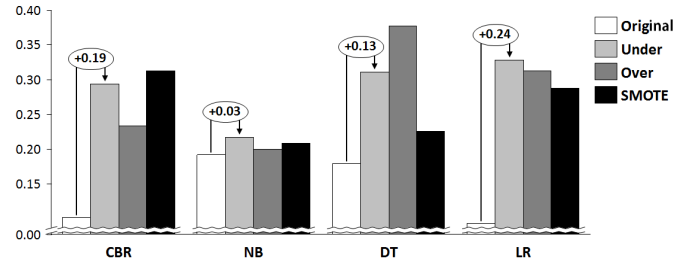
IV. RESULTS AND ANALYSIS

A. Basic Classifiers and Sampling Techniques

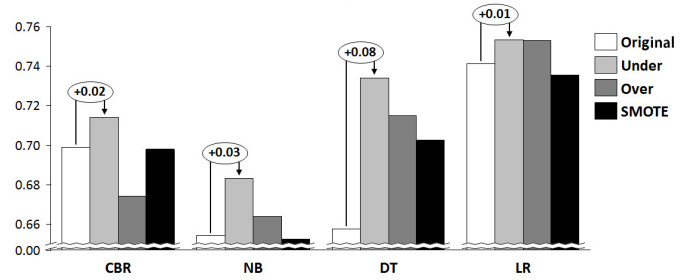
Figure 2a plots the F-measure of four classifiers for four sampling techniques. For each group of bars, it is clear that the performance of the original classifier (the leftmost white bar) is significantly lower than the performance of the classifiers with sampling techniques (other three bars). This performance increase is due to the fact that the detection of defaulters (minority class) becomes more accurate and more effective.

Figure 2b compares AUC values of four classifiers for four kinds of sampling techniques. All four groups of bars indicate that undersampling (the second left bar) outperforms the original classifiers (the left most bar) significantly. In comparison to random oversampling and SMOTE, undersampling performs also better or equally well (the AUC of logistic regression with random oversampling is close to undersampling). Another interesting result is that SMOTE does not improve the performance substantially. The AUC result of SMOTE on Naive Bayes and logistic regression even decreases slightly. A plausible explanation can be found in a study of SMOTE for high-dimensional class-imbalanced data [34]. On high dimensional and imbalanced data, they conclude that SMOTE has hardly any effect on most classifiers trained on high-dimensional data.

From these experimental results, we can conclude that: 1) Imbalanced data cause the basic classifiers to bias to the majority class; 2) Sampling makes the classes more balanced and increases AUC. Random undersampling works better than the other three techniques; 3) Logistic regression with undersampling is the best classifier so far. It gives an AUC of 0.7531 and outperforms the other tested classifiers significantly. Bolton indicated in [35] that logistic regression is the most favored method in practice of credit score prediction due to (almost) no assumptions imposed on variables, with



(a) Bar charts of F-measure.



(b) Bar charts of AUC.

Fig. 2. Original, Under, Over and SMOTE in the legend stand for basic classifier, randomly undersampling, randomly oversampling and SMOTE, respectively.

the exception of missing values and multicollinearity among variables.

B. Ensemble Methods

Figure 3 shows the test results of ensemble methods. The four groups of bars from the left plot the AUC values of basic classifiers with and without bagging. They illustrate that all results of bagging (right bars) exceed the performance of basic classifiers (left bars). If using a student t-test here to compare the difference of AUC with and without bagging, the p -values are 0.4592 for case-based reasoning, 0.1037 for Naive Bayes, 0.0000 for the decision tree and 0.3198 for logistic regression. Although bagging helps all four basic classifiers, applying it to the Decision tree gives the most significant difference. The results fit the theoretical analysis in [12]. A decision tree, which is a kind of unstable classifier, can benefit more from bagging.

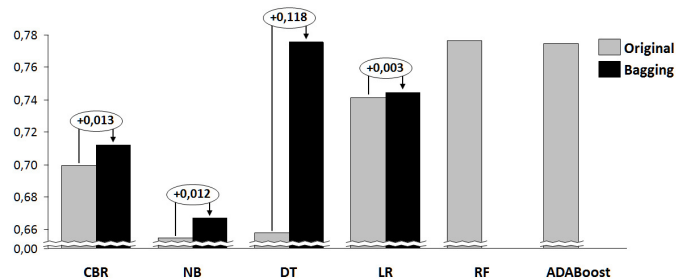


Fig. 3. Bar charts of the AUC value of ensemble methods.

The two bars on the right in Figure 3 plot the AUC values of random forests and AdaBoost. It is clear that random forests,

AdaBoost and bagging with decision trees generate the highest AUC values and outperform the basic classifiers remarkably.

C. Balanced Ensemble Methods

In this subsection, we first empirically discuss the impact of undersampling on bagging methods, then we compare the performance of symmetric bagging, balanced random forests, EasyEnsemble and BalanceCascade.

Bagging with decision trees, bagging with logistic regression and random forests are tested with three kinds of sampling ratios, 1:9 (original), 1:5 (around half undersampling delayers) and 1:1 (balanced sampling). Figure 4a illustrates the different AUC values and all three groups of results show the same trend that balanced sampling does help the classification. The original distribution (the left most bars) obtains the lowest AUC values, 1:5 ratio (the bars in the middle) improves the AUC values, and 1:1 symmetric sampling (the rightmost bars) gives the highest scores. Our testing results are consistent with former studies of symmetric bagging [28].

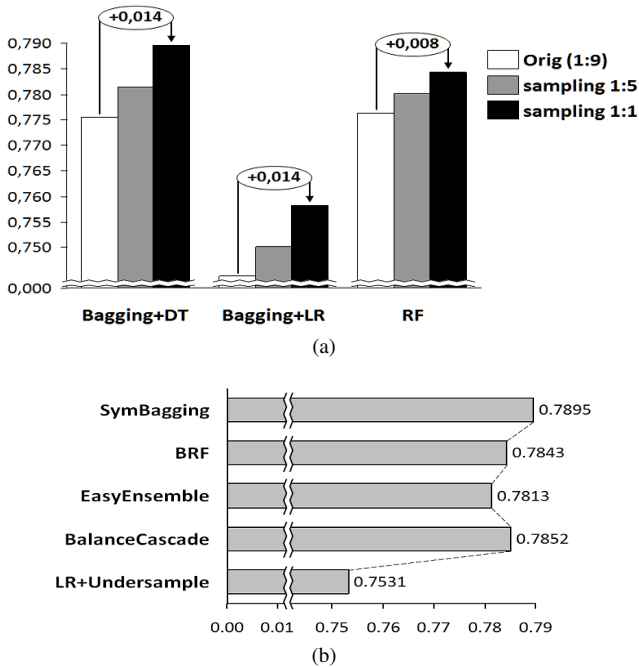


Fig. 4. (a) Bar chart of AUC values with different sampling ratios. (b) Bar chart of AUC values of balanced ensemble methods. The AUC value of logistic regression with undersampling is also plotted here.

Symmetric bagging, balanced random forests, EasyEnsemble and BalanceCascade are compared in Figure 4b. The performances of all four classifiers are close to each other. Although symmetric bagging is a bit higher than the other three methods, the student t-test does not show significant differences between them (p -value between symmetric bagging and balanced random forests is 0.6415). We also put the result of logistic regression with undersampling in the figures as baseline, which performs the best of all tested basic classifiers with sampling techniques. Apparently its performance is exceeded by all four balanced ensemble methods.

D. Finding the Best Classifier

So far, we have already tested and compared several approaches and all results are summarized in Table III. The experiments show that balanced ensemble methods give the best results. For the sake of building a robust, meaningful and fast model, balanced random forests (BRF) is selected as the final classifier of the system. The reasons are as following: 1) As a variant of random forests, BRF can handle thousands of variables efficiently. It needs less data preprocessing, because random forests can handle both discrete and continuous data, is not sensitive to outliers, and does not need variable deletion [13]. 2) BRF is a fast method because it handles less data instances (undersampling the majority class in each bootstrap sample), less features (only uses a part of the features but not the full set while constructing each split node) and does not need to prune trees. 3) BRF has only two parameters to tune, i.e., the number of trees (N_{tree}) and the number of variables randomly sampled as candidates at each split (m_{try}).

After tuning parameters in the 10-fold stratified cross validation, we use balanced random forests with $N_{tree}=2000$ trees and $m_{try}=70$ to build the final classifier. The achieved AUC value is 0.8002.

E. Cost Matrix Analysis

Both from the customer and bank perspective, the objective is to save as much on risk costs as possible. In addition to this, the bank needs to balance these risk costs with operational costs, such as employing customer service clerks and system capabilities. In this section, we analyse these aspects. A cost matrix is proposed first, then we decide on the best cut-off threshold to make minimal global costs.

After getting the classification results, the following actions of the arrears management department will yield both an operational cost, which is the overhead of treatments, and a risk cost, which is caused by giving the wrong treatment due to misclassifications. They are calculated in the following way. Due to confidentiality issues, we use some symbols in the formulas.

- Operational cost: automatic treatments like emails, SMS and letters will be sent to all arrear customers, no matter defaulters or delayers. The average cost of this treatment is $\text{€}A$ per customer. Predicted defaulters receive an additional treatment by a phone call and the estimated cost of the treatment is around $\text{€}4.3A$, including the personnel costs of the bank staff.
- Risk cost: as mentioned in section I, loan-loss provision is the main source of the risk cost. If a customer is a delayer, no matter what kind of classification result, the customer will not be in arrears. So, there is no risk cost for the misclassification of an actual delayer. If a customer is an actual defaulter and is misclassified as a delayer, he/she will miss the intensive treatments and will probably bring the loan-loss provision by misclassification. Suppose the loan-loss provision is $\text{€}B$ per arrear customer and the rate that a defaulter goes back to a healthy status with inten-

TABLE III
TESTING RESULTS OF BASIC CLASSIFIERS W/O SAMPLING TECHNIQUES, ENSEMBLE METHODS AND BALANCED ENSEMBLE METHODS.

Methods	Original	Under	Over	SMOTE	Bagging(1:9)	Bagging(1:5)	Bagging(1:1)
CBR	0.6989±0.018	0.7140±0.050	0.6742±0.015	0.6977±0.010	0.7017±0.083	0.7098±0.054	0.7217±0.063
NB	0.6540±0.012	0.6830±0.023	0.6638±0.009	0.6521±0.019	0.6664±0.026	0.6748±0.021	0.6903±0.017
DT	0.6574±0.018	0.7339±0.008	0.7147±0.009	0.7023±0.049	0.7754±0.012	0.7813±0.028	0.7895±0.024
LR	0.7412±0.017	0.7531±0.017	0.7529±0.013	0.7354±0.029	0.7442±0.011	0.7500±0.016	0.7581±0.020

Methods	RF	RF(1:5)	BRF	AdaBoost	EasyEnsemble	BalanceCascade
AUC	0.7763±0.016	0.7801±0.010	0.7843±0.009	0.7747±0.013	0.7813±0.032	0.7852±0.013

sive treatment is β , then the risk cost of a misclassified defaulter is $\epsilon\beta B$.

TABLE IV
COST MATRIX.

		Predict class	
		Defaulter	Delayer
Actual class	Defaulter	4.3A	A + βB
	Delayer	4.3A	A

By summing up the operational cost and risk cost, we get the cost matrix as shown in Table IV. Then, we can calculate the minimal global cost to determine the best cut-off threshold. A confusion matrix is generated by a given cut-off threshold. Let us denote the threshold as θ , and $TP(\theta)$, $FP(\theta)$, $TN(\theta)$ and $FN(\theta)$ as the four elements in the confusion matrix. Since the elements in the cost matrix represent the average unit cost per customer, we multiply element-wise the cost matrix and the confusion matrix and sum the products to obtain the total cost.

$$C_{total}(\theta) = 4.3A \cdot TP(\theta) + (A + \beta B) \cdot FN(\theta) + 4.3A \cdot FP(\theta) + A \cdot TN(\theta)$$

By using the same way of plotting an ROC curve, different costs can be calculated by traversing each threshold on a cost curve. Then, the minimal cost can be determined and the corresponding threshold is just the optimal threshold. The cost curve plotted in figure 5 can provide us with some insight. The threshold 0.178 gives us the minimal cost marked by a cross in Figure 5a, but the corresponding positive rate is around 0.80 marked by the cross in Figure 5b. In other words, 80% of the arrear customers are classified as defaulters. (We remind that the actual percentage of defaulters is around 11%.) This reflects the high risk cost of misclassifying a defaulter as a delayer: for the total costs it is of advantage to chose a lower threshold that will lead to many false positives but will reduce the number of false negatives (missed defaulters). Although the total cost is smallest, the arrear management department normally does not have enough capacity to handle (call) 80% of the arrear customers. The cost curve in Figure 5b is monotonically decreasing before the lowest cost point (the cross), so in the real deployment of the classification system the chosen threshold corresponds to the maximum capacity of customer service clerks, which can handle around 25% to 30% of all arrear customers. (We note that this is still 2 to 3 times

more than the percentage of defaulters.) The default cut-off threshold value 0.5 (the dot in Figure 5a) of balanced random forests classifies around 25% of arrear customers as defaulters (the dot in Figure 5b), which just fits the current contacting capability. This analysis also shows a shortcoming of the AUC metric. Because the whole curve cannot be used due to the limit imposed by available bank personnel, the measure should only consider a part of the curve. However, since the best classifiers usually dominate worse classifiers on a very large part of the curve, we do not think this significantly changes our conclusions.

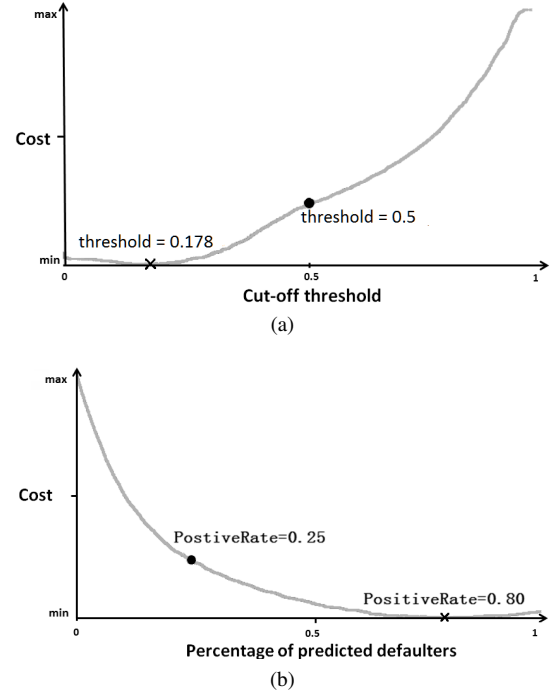


Fig. 5. Cost curve plotting. (a) X-axis is the cut-off threshold. Y-axis is the total cost. (b) X-axis is the percentage of predicted defaulters of all arrear customers. Y-axis is the total cost.

F. Feature Importance

Since we have already selected balanced random forests as the classifier, it is natural to employ built-in functionalities of random forests to analyze the data further. A way to evaluate feature importance was proposed in [13]. The top 30 features are plotted in Figure 6. The names of the features are omitted due to confidentiality issues.

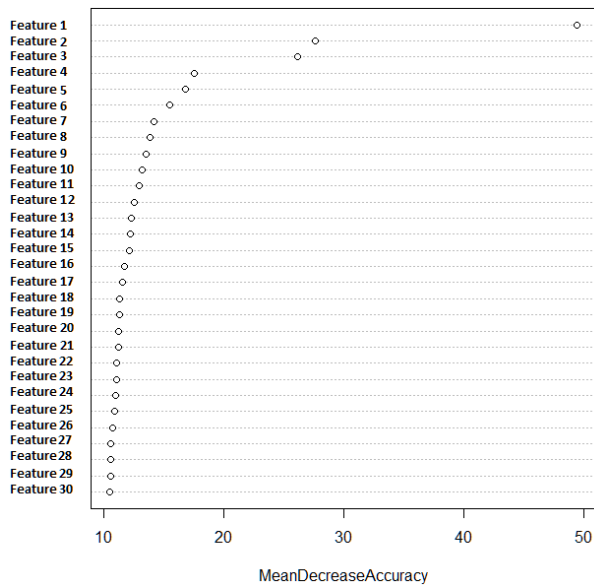


Fig. 6. Dot chart of feature importance. The Y-axis shows the anonymized features. They are ordered top-to-bottom as most-to-least important. The X-axis shows the mean decrease in accuracy as determined during the out of bag error calculation phase. The more the accuracy of the random forest decreases due to the addition of a single variable, the more important the variable is deemed, and therefore variables with a large mean decrease in accuracy are more important for classification of the data.

It is clear from the figure that the top 3 features are strong predictors, which are far beyond all the other features. The customer service clerks can communicate with arrear customers effectively with the guidance of these top features. Then, the importance decreases dramatically from the fourth feature, and keeps diminishing gradually. It implies that this classification problem is a difficult one, because most of the features are latent factors and only have weak correlations with the class labels, although they are selected from 2,000 initial features by the feature selection algorithm and the domain knowledge.

V. CONCLUSION

In this paper we presented a classification system for mortgage arrear management. Our experiments showed that sampling techniques and ensemble methods play the key role to achieve good performance and overcome the class imbalance. We chose balanced random forests as the classifier. The system has already been deployed in the arrear management department of a Dutch bank for several months. A new working process was also developed. Comparing with the old one, the new process gives intensive treatments such as phone calls to predicted defaulters at the very beginning, meanwhile the predicted delayers are only treated in automatic ways like emails, letters and SMS. We know the real class labels of the customers after one month, so the real labels can be used for validation. The validation AUC result of May 2013 is 0.7714, which is promising and consistent with the test results. Useful knowledge is also discovered, such as feature importance and cost matrix analysis. They can guide the daily work of the

arrears management department and provide insight. Compared to the old process, the classification system and the new process can push on average around 19% to 30% (varies in different months) more defaulters out of arrears, which saves a huge amount of costs for the bank.

REFERENCES

- [1] W. Vandevyvere and A. Zenthöfer, "The housing market in the Netherlands," Directorate General Economic and Monetary Affairs, European Commission, Tech. Rep., 2012.
- [2] E. I. Altman, "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy," *The journal of finance*, vol. 23, no. 4, pp. 589–609, 1968.
- [3] G. H. John and Y. Zhao, "Mortgage data mining," in *Computational Intelligence for Financial Engineering (CIFER), Proceedings of the IEEE/IAFE 1997*. IEEE, 1997, pp. 232–236.
- [4] D. Feldman and S. Gross, "Mortgage default: classification trees analysis," *The Journal of Real Estate Finance and Economics*, vol. 30, no. 4, pp. 369–396, 2005.
- [5] E. Scheurmann and C. Matthews, "Neural network classifiers in arrears management," in *Artificial Neural Networks: Formal Models and Their Applications-ICANN 2005*. Springer, 2005, pp. 325–330.
- [6] Q. Gan, B. Luo, and Z. Lin, "Risk management of residential mortgage in China using data mining a case study," in *New Trends in Information and Service Science, 2009. NISS'09. International Conference on*. IEEE, 2009, pp. 1378–1383.
- [7] L. C. Thomas, "A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers," *International Journal of Forecasting*, vol. 16, no. 2, pp. 149–172, 2000.
- [8] G. H. Lee, "Rule-based and case-based reasoning approach for internal audit of bank," *Knowledge-Based Systems*, vol. 21, no. 2, pp. 140–147, 2008.
- [9] S. Sarkar and R. S. Sriram, "Bayesian models for early warning of bank failures," *Management Science*, vol. 47, no. 11, pp. 1457–1475, 2001.
- [10] B. Baesens, T. Van Gestel, S. Viaene, M. Stepanova, J. Suykens, and J. Vanthienen, "Benchmarking state-of-the-art classification algorithms for credit scoring," *Journal of the Operational Research Society*, vol. 54, no. 6, pp. 627–635, 2003.
- [11] S. Nettleton and R. Burrows, "Mortgage debt, insecure home ownership and health: an exploratory analysis," *Sociology of health & illness*, vol. 20, no. 5, pp. 731–753, 1998.
- [12] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.
- [13] —, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [14] Y. Freund, R. E. Schapire *et al.*, "Experiments with a new boosting algorithm," in *International Conference on Machine Learning*, vol. 96, 1996, pp. 148–156.
- [15] E. Ngai, Y. Hu, Y. Wong, Y. Chen, and X. Sun, "The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature," *Decision Support Systems*, vol. 50, no. 3, pp. 559–569, 2011.
- [16] G. M. Weiss, "Mining with rarity: a unifying framework," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, pp. 7–19, 2004.
- [17] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Handling imbalanced datasets: A review," *GESTS International Transactions on Computer Science and Engineering*, vol. 30, no. 1, pp. 25–36, 2006.
- [18] Y. Sun, A. K. Wong, and M. S. Kamel, "Classification of imbalanced data: A review," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, no. 04, pp. 687–719, 2009.
- [19] C. Lemnar and R. Potolea, "Imbalanced classification problems: systematic study, issues and best practices," in *Enterprise Information Systems*. Springer, 2012, pp. 35–50.
- [20] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou, "On the class imbalance problem," in *Natural Computation, 2008. ICNC'08. Fourth International Conference on*, vol. 4. IEEE, 2008, pp. 192–201.
- [21] H. He and Y. Ma, *Imbalanced Learning: Foundations, Algorithms, and Applications*. Wiley.com, 2013.
- [22] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, no. 1, pp. 321–357, 2002.

- [23] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.
- [24] R. N. Khushaba, A. Al-Ani, and A. Al-Jumaily, "Differential evolution based feature subset selection," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE, 2008, pp. 1–4.
- [25] Y. Saeys, T. Abeel, and Y. Van de Peer, "Robust feature selection using ensemble feature selection techniques," in *Machine Learning and Knowledge Discovery in Databases*. Springer, 2008, pp. 313–325.
- [26] R. C. Holte, "Very simple classification rules perform well on most commonly used datasets," *Machine learning*, vol. 11, no. 1, pp. 63–90, 1993.
- [27] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 42, no. 4, pp. 463–484, 2012.
- [28] S. Hido, H. Kashima, and Y. Takahashi, "Roughly balanced bagging for imbalanced data," *Statistical Analysis and Data Mining*, vol. 2, no. 5-6, pp. 412–426, 2009.
- [29] C. Chen, A. Liaw, and L. Breiman, "Using random forest to learn imbalanced data," *University of California, Berkeley*, 2004.
- [30] X.-Y. Liu, J. Wu, and Z.-H. Zhou, "Exploratory undersampling for class-imbalance learning," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 39, no. 2, pp. 539–550, 2009.
- [31] H. He and E. A. Garcia, "Learning from imbalanced data," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 21, no. 9, pp. 1263–1284, 2009.
- [32] D. Hand, "Measuring classifier performance: a coherent alternative to the area under the ROC curve," *Machine Learning*, vol. 77, pp. 103–123, 2009.
- [33] U. Kaymak and A. B.-D. aand R. Potharst, "The AUK: A simple alternative to the AUC," *Engineering Applications of Artificial Intelligence*, vol. 25, pp. 1082–1089, 2012.
- [34] L. Lusa *et al.*, "SMOTE for high-dimensional class-imbalanced data," *BMC bioinformatics*, vol. 14, no. 1, pp. 1–16, 2013.
- [35] C. Bolton, "Logistic regression and its application in credit scoring," Ph.D. dissertation, University of Pretoria, 2009.