Improving Resampling-based Ensemble in Churn Prediction

Bing Zhu ZHUBING1866@HOTMAIL.COM
Business School, Sichuan University, Chengdu 610064, P.R. China

Seppe vanden Broucke SEPPE.VANDENBROUCKE@KULEUVEN.BE
Department of Decision Sciences and Information Management, KU Leuven, Leuven 3000, Belgium

Bart Baesens BART.BAESENS@KULEUVEN.BE
Department of Decision Sciences and Information Management, KU Leuven, Leuven 3000, Belgium

Sebastián Maldonado SMALDONADO@UANDES.CL
Facultad de Ingeniería y Ciencias Aplicadas Universidad de los Andes, Santiago, Chile

Editors: Luís Torgo, Bartosz Krawczyk, Paula Branco and Nuno Moniz.

Abstract

Dealing with class imbalance is a challenging issue in churn prediction. Although resampling-based ensemble solutions have demonstrated their superiority in many fields, previous research shows that they cannot improve the profit-based measure in churn prediction. In this paper, we explore the impact of the class ratio in the training subsets on the predictive performance of resampling-based ensemble techniques based on experiments on real-world churn prediction data sets. The experimental results show that the setting of the class ratio has a great impact on the model performance. It is also found that by choosing suitable class ratios in the training subsets, UnderBagging and Balanced Random Forests can significantly improve profits brought by the churn prediction model. The demonstrated results provide some guidelines for both academic and industrial practitioners.

Keywords: Churn prediction, Class imbalance, Profit-based measure, Resampling-based ensemble

1. Introduction

Churn prediction aims at identifying potential churning customers based on past information and prior behaviors and is an important issue in customer relationship management. Due to fierce market competitions customers can easily switch between competitors. The literature has suggested that attracting new customers is five to six times more expensive than serving current customers (Colgate and Danaher, 2000). Therefore, companies have been forced to pay more attention to retaining existing customers than acquiring new customers. Accurate churn prediction can help companies to focus their retention efforts towards the right customers at the right time and we have witnessed various applications of data mining techniques in this field.

One important issue in churn prediction is to deal with class imbalance. The number of churners is usually significantly outnumbered by the non-churners in many industries. In the telecom sector for instance, the monthly churn rate is 2%. The class imbalance problem has a negative influence on the standard classification algorithms as they tend to misclassify the minority instances more often than the majority instances. In extreme cases, they may simply classify all instances to the majority class, resulting in high overall precision but unacceptably low accuracy with respect to minority class of interest. To deal with this issue, many solutions have been proposed, which can be roughly categorized into data-level, algorithm-level and ensemble solutions (Branco et al., 2016; Krawczyk, 2016). Ensemble solutions try to develop or modify existing ensemble learning algorithms to improve the performance on imbalanced data sets. This approach has demonstrated its superiority in many fields, especially the resampling-based ensemble which integrates sampling strategies into the base classifiers training stage. However, our previous research has shown that ensemble solution shows no improvement of the profits brought by the churn models (Zhu et al., 2017), which is arguably the main metric of interest in the context of churn and other business-oriented data mining models, rather than accuracy or related metrics.

In this paper, we explore the impact of the class ratio in the training subsets on the behavior of resampling-based ensemble models and try to find a way to improve the model performance of resampling-based ensembles in terms of the profit-based measure. Most current resampling-based solutions assume equally balanced post-sampling class ratios for base classifiers training as our previous empirical research has done (Zhu et al., 2017). However, some scholars have have found that such a balanced class ratio is not optimal when using sampling as a preprocessing step to learn single classifiers (Albisua et al., 2013). Therefore, one reasonable conjecture is that the class ratio does not need to be balanced when training the base classifiers in a resampling-based ensemble. In order to justify this conjecture, we carry out a set of experiments on real-world churn prediction data sets to study the relationship between the class ratio in the training subsets and the model performance of resampling-based ensemble. A recently developed profit-based measure called expected profit measure (EMP) is used as the measure in the experiment to calculate the profit brought by the churn prediction model. Our experiments show that the class ratio in the training subsets has a significant impact on the model performance. By choosing a suitable ratio, UnderBagging and Balanced Random Forest can gain significant improvements in terms of the profit-based measure.

The remainder of this paper is organized as follows. In Section 2, we will briefly review related work in churn prediction and class imbalance. Then, Section 3 presents the experimental methodology and Section 4 discusses the experimental results. Finally, Section 5 gives conclusions and directions for further research.

2. Related Work

Class imbalance is intrinsic to many data sets used in churn prediction. Most standard classification algorithms exhibit a bias towards the majority class and hence suffer from class imbalance (Sun et al., 2009). Many solutions have been proposed to address this issue, which can be roughly divided into three categories: data-level, algorithm-level and ensemble solutions. The data-level solution tries to re-balance the class distribution by resampling the original unbalanced data sets. This solution consists of many different forms of resampling techniques such as random over-sampling, random under-sampling, and the synthetic minority over-sampling technique (SMOTE). Algorithm-level solutions attempt to adapt existing learning algorithms to strengthen their learning ability with regards to the minority class. Cost-sensitive learning, active learning, one-class learning are typical methods in this group.

Ensemble solutions have become increasingly popular in recent years. They either modify existing ensemble learning algorithms at the data-level to preprocess the data before the learning base classifiers or they embed a cost-sensitive framework in the ensemble learning process. Consequently, ensemble solutions can be further divided into cost-sensitive ensembles and resampling-based ensembles. Cost-sensitive ensembles assign different misclassification costs to different classes in the ensemble algorithm, so that a heavier penalty is placed for misclassifying the minority class. The main difficulty with such techniques is then to configure suitable misclassification costs for the different classes, which is a non-trivial task.

Resampling-based ensemble techniques embeds data-preprocessing into the training of the base classifiers. It is preferred for its simplicity and effectiveness. Many ensemble solutions based on Bagging, Boosting, and Random Forests techniques have been proposed. For example, UnderBagging and OverBagging (Wang and Yao, 2009) use random undersampling and oversampling respectively so that the class distribution is balanced when building each base model in the ensemble. Roughly Balance Bagging (Hido et al., 2009) is also based on undersampling, but the size of the majority examples is determined probabilistically according to a negative binomial distribution, whereas the number of minority examples is always the same. In SMOTEBagging (Wang and Yao, 2009), each training data subset is composed of a bootstrap sample taken from the majority class, and a sample of the minority class created through a combination of random oversampling and SMOTE. Chawla et al. proposed an approach called SMOTEBoost by the combination of SMOTE sampling and a boosting procedure (Chawla et al., 2003), in which SMOTE sampling is used before computing the new instance weights in each iteration. Seiffert et al. (2010) presented a different ensemble method named RUSBoost, which removes instances from the majority class by random undersampling in each iteration of the boosting procedure. Chen et al. (2004) proposed the Balanced Random Forest (BRF), which combines random undersampling with the Random Forest approach. For each repetition, the same number of instances from the minority class and the majority class are drawn at random. Galar et al. (2012) developed a thorough empirical comparison by the consideration of a wide range of ensemble solutions. One of their main conclusions states that resampling-based solutions provide better results than cost-sensitive ensembles.

In churn prediction, the techniques to deal with class imbalance do not go beyond the scope of data-level, algorithm-level and ensemble-based solutions. Burez and Van de Poel (2009) performed experiments to investigate how to better handle class imbalance in churn prediction, where random undersampling and Weighted Random Forests show their effectiveness in terms of the AUC measure. Meanwhile, our previous research shows no improvement for the ensemble-based solutions when a profit-based measure is considered and the predictive performance of cost-sensitive ensembles are worse than the resampling-based ensembles (Zhu et al., 2017).

Table 1: Summary of the churn data sets used in the experiment

Data set	Source	Region	#Obs.	#Att.	Churn Rate (%)
Chile	Operator	South American	5300	41	5.66
Duke1	Duke	North American	51306	173	1.80
Duke2	Duke	North American	100462	173	1.80
KDDcup	KDD CUP 2009	Europe	50000	231	7.34
K1	Operator	East Asia	2019	10	3.96
K2	Operator	East Asia	2941	14	4.42
K3	Operator	East Asia	5990	36	4.34
K4	Operator	East Asia	2183	9	4.58
K5	Operator	East Asia	26224	11	4.19
Tele1	Operator	Europe	4350	87	8.05
UCI	UCI ML repository	-	3333	19	14.5

3. Experimental Framework

Ensemble algorithms usually consist of two stages, i.e. base model training and aggregation, so that the improvement of model performance can be done in both stages. In this paper, we focus on the base classifier training stage and try to investigate how changes of the class ratio influence the final predictive performance based on experiments on the real-world data sets.

The profits brought by the model are of main concern in the context of churn prediction. However, the commonly used measure metric AUC does not take this factor into account. To measure the profit, Neslin et al. (2006) presented a framework and established the following expression to get the total profit for a retention campaign:

$$P = N\alpha[\beta\gamma(CLV - c - \delta) + \beta(1-\gamma)(-c) + (1-\beta)(-c-\delta)] - A \tag{1}$$

where P is the profit of the retention campaign, N is the number of customers in the current customer base, α is the fraction of the target customers in the retention campaign, β is the portion of churners within the targeted customers, γ is the probability of would-be churners accepting the offer and staying in the company, CLV is the average customer lifetime value of retained customers, c is the cost of the incentive when a customer accepts the offer, δ is the cost of contacting a customer to offer the incentive, and A is the fixed administrative cost of running a retention program. By taking into account the uncertainty of the probability of accepting incentive offer, Verbraken developed the expected maximum profit (EMP) (Verbraken et al., 2013), which is calculated as follows:

$$EMP = \int_{\gamma} P_c(T(\gamma); CLV, \delta, \phi) \cdot h(\gamma) d\gamma \tag{2}$$

where $h(\gamma)$ is the probability density function for γ. Given a γ value, $T(\gamma)$ is the optimal threshold and $P_c(T(\gamma); CLV, \delta, \phi)$ is the maximum profit obtained by selecting an optimal targeting size α in equation (1). γ is assumed to follow a Beta distributed random variable

$B(\alpha, \beta)$. We use the EMP measure as the main performance measure in our experiments. The parameters CLV, δ, and c were set to be 200, 10 , and 1, respectively. The two parameters in $B(\alpha, \beta)$ were set to be 6 and 14. The parameter settings were based on previous scientific literature (Verbraken et al., 2013; Jahromi et al., 2014) and discussion with data scientists in the telecommunication industry.

We considered four different options for the minority-to-majority class ratios in the experiments: 1:1, 1:1.5, 1:3, and cost-weighted ratio $W = 1 : \frac{C_{-} * N_{-}}{C_{+} * N_{+}}$. C_{+} and C_{-} are the mean misclassification costs for the minority and majority class while N_{+} and N_{-} are the number of instances of the two classes. According to Equation (1), the mean misclassification costs of both class are $C_{-} = \delta + c = 11$ and $C_{+} = E(\gamma(CLV - \delta) - c) = 56$. We use these options to determine the class ratio of the training data subsets used to build base classifiers. For OverBagging and UnderBagging, Oversampling and undersampling were used in combination with bootstrap sampling to generate the training subsets so that the class ratio in each subset reaches the considered value. In SMOTEBagging, the number of minority instances created through a combination of random oversampling and SMOTE were determined by each class ratio setting. The implementation of BRF is similar to UnderBagging and undersampling was used in each repetition to randomly select majority instances to get the desired class ratio. In RUSBoost and SMOTEBoost, we used undersampling to remove majority instances or SMOTE to generate new minority instances in each iteration to reach the ratio value before computing the new instance weights.

Eleven real-world data sets coming from the telecommunication industry were used in the experiments. Table 1 summarizes the main characteristics of the dataets, where each column presents the name, data source, region, number of observation (#Obs.) and attributes (#Att.) as well as the churn rate. Usually, the customer behavioral attributes used to churn predication are temporal. They have been transformed into static ones through aggregation, so all the attributes in the eleven data sets are static. As Table 1 shows, the churn rates range from 1.8% to 14.5%. In order to avoid the "curse of dimensionality" problem, a feature selection procedure was conducted as a step of data preparation. The feature selection approach based on the Fisher score was used on five data sets: Chile, Duke1, Duke2 , KDD and Tele1. Features with highest Fisher scores are retained and the numbers of variables in these data sets were reduced to 30.

A 5*2 cross-validation strategy was applied in our study as follows: each original data set was randomly split into two equally-sized parts. First, one part was used as training data to build models, while the other part acted as test data to calculate the performance.Then, the two parts reversed their role. This process was repeated five times to get average values of the EMP measure. The state-of-the-art ensemble techniques used in our experiments are three Bagging-based ensembles (UnderBagging, OverBagging and SMOTEBagging), two Boosting-based ensembles (RUSBoost, SMOTEBoost), and one Random-Forest-based ensembl(Balance Random Forest). We also included the original Bagging, AdaBoost and Random Forest algorithms to allow for a comparison with a solid base. In all ensemble techniques, C4.5 decision trees were used as a base classifier. We set the confidence level to 0.25. The minimum number of instances per leaf was set to 2 and pruning was used to obtain the final tree. We set the number of bags in Bagging-based ensembles, the number of iterations in Boosting-based ensembles and the number of trees in RandomForest-based ensembles to be 40, which is consistent with previous research (Galar et al., 2012; Liu and Zhou, 2009).

4. Experimental Results and Discussion

Our analysis consists of three consequential steps.

- First, we compare the performance of different setting within each technique to check whether the class ratio has a significant, overall influence.
- Then, we perform a comparison between techniques that are influenced by the class ratio and the original ensemble algorithm without considering class imbalance to find whether the technique at hand is well-suited to increase the EMP measure.
- Finally, we further analyze the techniques that show such improvement over the original method and investigate how their performance changes with different class ratios.

Let us first check the overall influence of different class ratios. Table 2 to Table 5 provide the detailed results grouped by the ensemble strategies. Each entry in these tables provides the mean value of the EMP measure in the 5*2 cross validation and the best option for a method among the four class ratios is highlighted in bold in each row. To judge the statistical significance of the difference among different class ratios, the Friedman test with Iman-Davenport extension are used, the last rows provide the p-values of the Iman-Davenport test. We can see from these tables that there is no universally “winning” setting and the optimal class ratio depends on the data sets. From the experimental results of the Bagging-based ensemble shown in Table 2 and 3, we can see there is no statistical difference at the 5% significance level among different class ratios for OverBagging and SMOTEBagging according to the Iman-Davenport test. With respect to UnderBagging, on the contrary, there is an obvious statistical difference among different ratios. The EMP measure favors the cost-weighted ratio for UnderBagging, which obtains the best results on six out of eleven data sets. Next, we will analyze the Boosting-based ensemble solutions in Table 4. For RUSBoost, there is no evidence of statistical significance among different class ratios. Meanwhile, a cost-weighted ratio seems to be a good option in SMOTEBoost and gets the best results on seven out of the eleven data sets. The results in Table 5 demonstrate that a significant difference exists within different class ratios for Balanced Random Forest, with the cost-weighted ratio or class ratio of “1:3” giving better results, the latter outperforming the former with a small margin.

After finding the best setting for each solution, we now compare their performance with their corresponding original algorithms without considering class imbalance to see whether they can improve the profit-based measure. We only choose the techniques that show a statistically significant difference among different class ratio settings. The Wilcoxon test is exploited to perform a pairwise statistical comparison. Table 6 presents the results, where the second column gives the profit gains over the original method with the best class ratio setting, the third column provides the p-values of Wilcoxon test, and the last column presents the decision on the null hypothesis of equivalence. As can be seen from Table 6, both UnderBagging and BRF improve the profit significantly over the original method. SMOTEBoost can improve the result, but the improvement is not significant.

Table 2: Experimental results of Bagging-based ensemble

Data set	Bagging	OverBagging				UnderBagging			
		1:1	1:1.5	1:3	W	1:1	1:1.5	1:3	W
Chile	0.8749	**0.9115**	0.9078	0.8534	0.8667	0.8429	0.8151	0.8491	**0.8632**
Duke1	0.0009	0.0012	0.0010	0.0006	**0.0015**	0.0004	0.0005	0.0005	**0.0012**
Duke2	0.0063	0.0020	0.0019	0.0024	**0.0034**	0.0020	0.0049	0.0048	**0.0053**
KDD	0.4295	**0.2827**	0.2806	0.2822	0.2706	**0.4684**	0.4588	0.4414	0.4576
K1	0.1902	0.1345	0.1790	**0.2276**	0.1988	0.2484	0.3688	**0.3699**	0.3122
K2	0.4695	0.4297	0.4025	**0.4417**	0.4125	0.4635	0.5008	0.5005	**0.5413**
K3	1.4552	**1.3570**	1.3442	1.3500	1.3189	1.4510	1.4788	1.4785	**1.4856**
K4	2.2814	2.2498	2.2416	2.2451	**2.2623**	2.2587	2.2807	**2.2863**	2.2840
K5	0.6525	0.5786	0.5700	0.5841	**0.5915**	0.6267	0.6548	0.6680	**0.6753**
Tele1	1.5778	1.4740	**1.5564**	1.5273	1.5328	1.5325	1.6160	1.6773	**1.7323**
UCI	6.2587	6.1710	6.1445	**6.2188**	6.2080	6.2632	6.2627	**6.3331**	6.2467
Iman-Davenport		$p = 0.2457$				$p = 0.0074$			

Table 3: Experimental results of Bagging-based ensemble (Continued)

Data set	Bagging	SMOTEBagging			
		1:1	1:1.5	1:3	W
Chile	0.8749	0.8706	0.8571	**0.8738**	0.8599
Duke1	0.0009	0.0003	0.0006	**0.0008**	0.0004
Duke2	0.0063	**0.0044**	0.0024	0.0033	0.0031
KDD	0.4295	0.2851	0.3012	0.3311	**0.3389**
K1	0.1902	0.2306	0.1952	0.2053	**0.2394**
K2	0.4695	0.4072	0.4233	0.4095	**0.4261**
K3	1.4552	1.3867	1.4091	1.4443	**1.4454**
K4	2.2814	2.2769	**2.2836**	2.2689	2.2921
K5	0.6525	0.5805	0.5912	**0.6083**	0.6000
Tele1	1.5778	**1.5442**	1.5147	1.5198	1.5127
UCI	6.2587	6.1891	6.2792	**6.3255**	6.1995
Iman-Davenport		$p = 0.1346$			

Now we will further explore how the predictive performance varies with class ratio. Beside the four above mentioned class ratios, we also consider the two new ones around the cost-weighted ratio: $W^- = 1 : (\frac{C_- * N_-}{C_+ * N_+} - 1)$ and $W^+ = 1 : (\frac{C_- * N_-}{C_+ * N_+} + 1)$. Figure Figure 1 to Figure 11 illustrate how EMP measure varies with class ratios. As these figures shown, EMP measure first increases and then decreases on most data sets. The optimal class ratio is not a fixed value but usually reaches the maximum EMP value around the cost-weighted ratio. Seven out of eleven data sets follow this pattern. There are also some exceptions on

Table 4: Experimental results of Boosting-based ensemble

Data set	AdaBoost	RUSBoost				SMOTEBoost			
		1:1	1:1.5	1:3	W	1:1	1:1.5	1:3	W
Chile	0.9058	**0.6833**	0.6424	0.6794	0.6800	0.8574	0.8637	**0.9126**	0.8933
Duke1	0.0004	0.0002	0.0004	**0.0006**	0.0004	0.0003	0.0004	**0.0007**	**0.0007**
Duke2	0.0019	0.0018	**0.0025**	0.0021	0.0017	0.0021	0.0020	0.0018	**0.0023**
KDD	0.2191	**0.2580**	0.2293	0.1820	0.1826	0.2001	0.2099	0.2125	**0.2194**
K1	0.1656	**0.3252**	0.2772	0.2050	0.1656	**0.2132**	0.1451	0.1408	0.2026
K2	0.2687	0.3723	**0.3946**	0.3348	0.2741	0.3585	0.4007	0.3441	**0.4093**
K3	1.2387	1.1767	**1.2061**	1.1327	1.0657	1.2703	1.2664	1.2694	**1.3378**
K4	2.2859	2.2004	2.1879	2.2285	**2.2395**	2.2503	2.2657	2.2631	**2.2682**
K5	0.5450	**0.5578**	0.5457	0.5497	0.5361	0.5222	0.5086	**0.5336**	0.5304
Tele1	1.3578	**1.4773**	1.3733	1.4059	1.4482	1.2680	1.2871	1.3305	**1.3458**
UCI	6.2028	**5.8458**	5.7820	1.5662	5.7438	6.2616	6.3116	**6.3491**	6.2590
Iman-Davenport		$p = 0.1149$				$p = 0.0164$			

Table 5: Experimental results of RandomForest-based ensemble

Data set	RandomForest	BRF			
		1:1	1:1.5	1:3	W
Chile	0.8534	0.7034	0.7403	**0.8254**	0.8110
Duke1	0.0002	0.0010	0.0012	**0.0014**	0.0012
Duke2	0.0038	0.0016	0.0042	0.0060	**0.0065**
KDD	0.3227	0.3798	0.4231	**0.4421**	0.4404
K1	0.2179	0.2667	**0.3388**	0.3112	0.3264
K2	0.2995	**0.4402**	0.4289	0.4050	0.4155
K3	1.3235	1.1396	1.3283	1.3881	**1.3931**
K4	2.2328	2.0986	2.1095	2.1576	**2.2404**
K5	0.6188	0.5979	0.6239	0.6416	**0.6589**
Tele1	1.4450	1.3831	1.5211	**1.6084**	1.5458
UCI	6.2960	6.1104	6.1546	**6.3039**	6.1244
Iman-Davenport		$p = 0.0003$			

three data sets. For data set UCI, the class imbalance level is low, so the changes of class ratio have limited influence on the EMP measure. On data set KDD, Chile and K4, there is no obvious difference across different class ratios for UnderBagging. The performance of BRF seems to decrease as the class ratio becomes more balanced on Chile and K4.

To summarize, we can draw the following conclusions:

- The optimal class ratio is not a fixed value, and depends on the data sets.

Table 6: Pairwise comparison with base ensemble algorithm

Methods	Profit gain	p-value	Action
SMOTEBoost vs AdaBoost	0.0252	0.3652	Not reject
UnderBagging vs Bagging	0.0371	0.0674	Reject
BRF vs RandomForest	0.0434	0.0537	Reject

- Some ensemble solutions can increase the performance of the EMP measure by altering the class ratio. UnderBagging and BRF can gain a statistically significant increase.
- For UnderBagging and BRF, a generally well-performing setting for the class-ratio is around the cost-weighted ratio.

Although the above interesting findings have been found, several issues need to be further discussed. The first one is the coherence of optimal class ratio of training data subsets used to generate base classifiers. The experimental results indicate that the optimal class ratio is around the cost-weighted ratio on most data sets. But there it is not a fixed value and there are also some exceptions. Therefore, some other factors that measure the characteristic of each data set should be considered to help finding the optimal class ratio. One possible measure may be the local class distribution near the class boundary. The second is the impact of base classifier. Decision tree is almost the standard configuration for ensemble solutions in imbalance learning. However, weather the change of base classifier will bring different results needs further exploration. The final issue is about the aggregation strategy. In this paper, we use default aggregation strategy of the original ensemble methods. The experiments show the Boost-based solutions do not significantly improve the performance in terms of profit-based measure. The results may be caused by their usage of weighted average to aggregate the base classifiers. How do different aggregation strategies influence the performance would be of great interest.

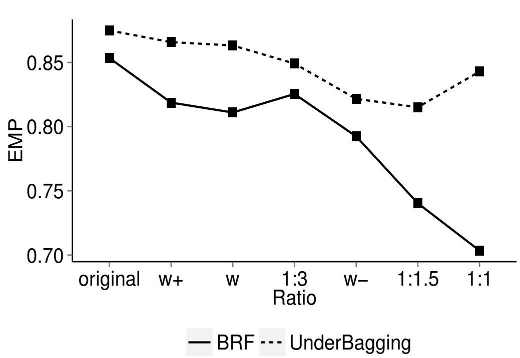

Figure 1: Model performance on Chile

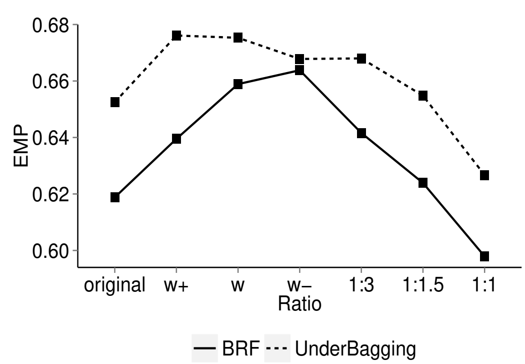

Figure 2: Model performance on Duke1

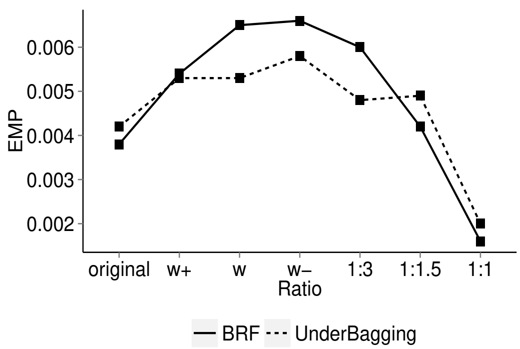

Figure 3: Model performance on Duke2

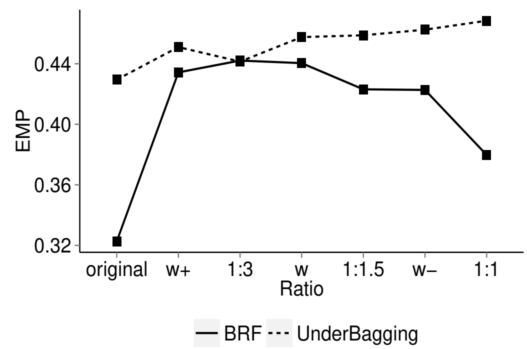

Figure 4: Model performance on KDD

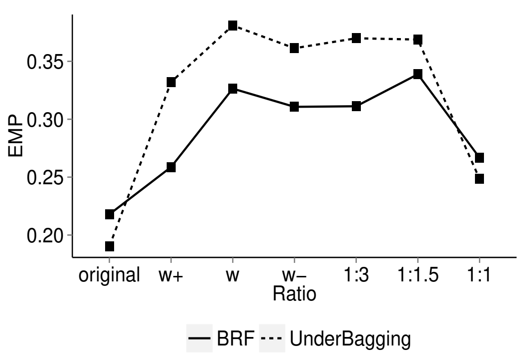

Figure 5: Model performance on K1

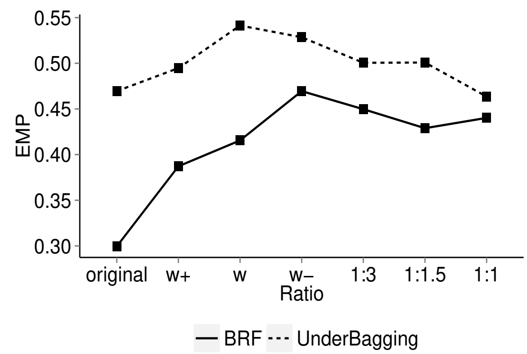

Figure 6: Model performance on K2

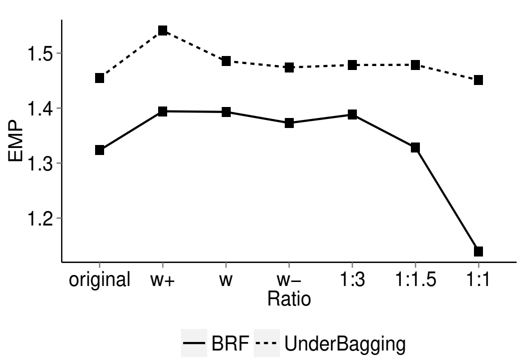

Figure 7: Model performance on K3

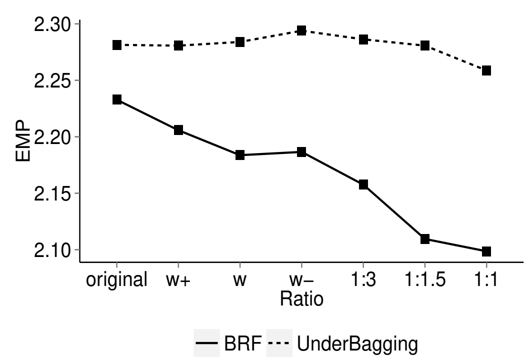

Figure 8: Model performance on K4

5. Conclusion

Churn prediction is an important part in customer relationship management. To address the class imbalance issue, many resampling-based ensemble solutions have been proposed. In this paper, we have explored the relationship between profit-based measure and class ratio

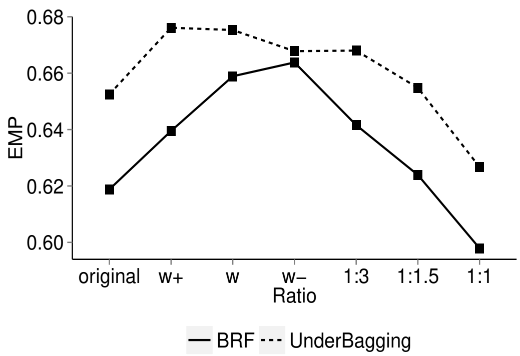

Figure 9: Model performance on K5

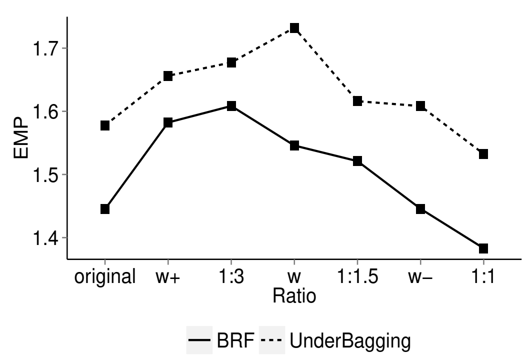

Figure 10: Model performance on Tele1

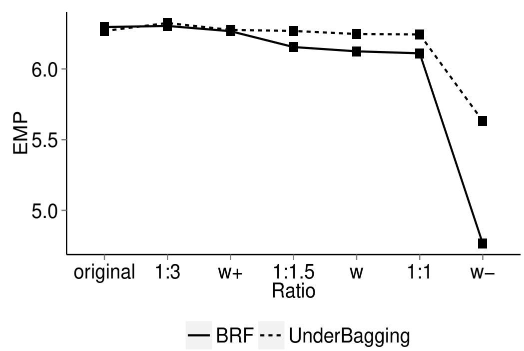

Figure 11: Model performance on UCI

in the training subsets. The experiments on real-world data sets show that the data ratio in the training subsets has a great impact on the profit-based measure for some resampling-based ensemble solutions. By altering the class ratio, UnderBagging and Balanced Random Forest can improve the profit-based measure. In future work, we plan to perform further studies to find deeper explanations behind the experimental results and to propose a new resampling-based ensemble solution by developing new aggregation strategy.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant No. 71401115), the MOE (Ministry of Education in China) Youth Project of Humanities and Social Sciences (Grant No. 13YJC630249). This work is also funded by Sichuan University (Grant No. skqy201742).

References

I. Albisua, O. Arbelaitz, I. Gurrutxaga, A. Lasarguren, J. Muguerza, and J. Pérez. The quest for the optimal class distribution: an approach for enhancing the effectiveness of learning via resampling methods for imbalanced data sets. *Progress in Artificial Intelligence*, 2 (1):45–63, 2013.

P. Branco, L. Torgo, and R. Ribeiro. A survey of predictive modeling on imbalanced domain. *ACM Computing Surveys (CSUR)*, 49(2):Art. 31, 2016.

J. Burez and D de Poel. Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, 36(3):4626–4636, 2009.

N. Chawla, A. Lazarevic, L. Hall, and K.W. Bowyer. Improving prediction of the minority class in boosting. Proceeding of ECML/PKDD2003, pages 107–119, 2003.

C. Chen, A. Liaw, and L. Breiman. Using random forests to learn imbalanced data. Technical report, Statistics Department of University of California at Berkeley, 2004.

M. Colgate and P. Danaher. Implementing a customer relationship strategy: The asymmetric impact of poor versus excellent execution. *Journal of the Academy of Marketing Science*, 28(3):375–387, 2000.

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. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 41(4):463–484, 2012.

S. Hido, H. Kashima, and Y. Takahashi. Roughly balanced bagging for imbalanced data. *Statistical Analysis and Data Mining*, 2(5-6):412–426, 2009.

A.T. Jahromi, S. Stakhovych, and M. Ewing. Managing b2b customer churn, retention and profitability. *Industrial Marketing Management*, 43(7):1258–1268, 2014.

B. Krawczyk. Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4):221–232, 2016.

X. Liu and Z. Zhou. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(2):539–550, 2009.

S. Neslin, S. Gupta, W. Kamakura, J. Lu, and C. Mason. Detection defection: Measuring and understanding the predictive accuracy of customer churn models. *Progress in Artificial Intelligence*, 43(2):204–211, 2006.

C. Seiffert, T. M. Khoshgoftaar, and J. Van Hulse. Rusboost: A hybrid approach to alleviating class imbalance. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 40(1):185–197, 2010.

Y. Sun, A.K.Wong, and M.S. Kamel. Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(4):687–719, 2009.

T. Verbraken, W. Verbeke, and B. Baesense. A novel profit maximizing metric for measuring classification performance of customer churn prediction models. *IEEE Transactions on Knowledge and Data Engineering*, 25(5):961–973, 2013.

S. Wang and X. Yao. Diversity analysis on imbalanced data sets by using ensemble models. IEEE Symposium on Computational Intelligence, pages 324–331, 2009.

B. Zhu, B. Baesens, and S. vanden Broucke. An empirical comparison of techniques for the class imbalance problem in churn prediction. *Information Sciences*, 408:84–99, 2017.