Cost-Sensitive Neural Network with ROC-Based Moving Threshold for Imbalanced Classification

Bartosz Krawczyk(✉) and Michał Woźniak

Department of Systems and Computer Networks,
Wroclaw University of Technology, Wrocław, Poland
{bartosz.krawczyk,michal.wozniak}@pwr.edu.pl

Abstract. Pattern classification algorithms usually assume, that the distribution of examples in classes is roughly balanced. However, in many cases one of the classes is dominant in comparison with others. Here, the classifier will become biased towards the majority class. This scenario is known as imbalanced classification. As the minority class is usually the one more valuable, we need to counter the imbalance effect by using one of several dedicated techniques. Cost-sensitive methods assume a penalty factor for misclassifying the minority objects. This way, by assuming a higher cost to minority objects we boost their importance for the classification process. In this paper, we propose a model of cost-sensitive neural network with moving threshold. It relies on scaling the output of the classifier with a given cost function. This way, we adjust our support functions towards the minority class. We propose a novel method for automatically determining the cost, based on the *Receiver Operating Characteristic* (ROC) curve analysis. It allows us to select the most efficient cost factor for a given dataset. Experimental comparison with state-of-the-art methods for imbalanced classification and backed-up by a statistical analysis prove the effectiveness of our proposal.

Keywords: Machine learning · Neural networks · Imbalanced classification · Cost-sensitive · Moving threshold

1 Introduction

Machine learning algorithms have been effectively used for classification purposes in last decades. However, new problems continuously emerge that pose challenge to pattern classifiers. Often these difficulties are embedded in the nature of the data. They may be connected with the volume, non-stationary nature, some specific characteristics, or differing quality of examples. One of such problems is the imbalance between the class representatives.

Standard classifiers assume, that the distribution of objects between classes is roughly equal. When this assumption is violated, they tend to get biased towards the class with higher quantity of examples. This deteriorates the performance over the minority class. Therefore, in order to get a well-balanced classifier that is competent over all of classes, we need to counter the imbalanced distribution.

K. Jackowski et al. (Eds.): IDEAL 2015, LNCS 9375, pp. 45–52, 2015.
DOI: 10.1007/978-3-319-24834-9_6

There is a plethora of methods proposed, usually focusing on some data pre-processing methods or guiding the training process towards the minority examples. Former group concentrates on operations on the dataset in order to re-balance it, and then uses standard classifiers. Latter group works on the original dataset, but changes the training procedure of the classifier. Both of these approaches are considered as an effective aid in the imbalanced classification domain.

Between these two groups of methods lies the cost-sensitive solution. Usually during the design of the classifier one assumes that all classes are identically important and calculate the training error on the basis of quantity of misclassified examples (e.g., as in 0–1 loss function). However, in imbalanced domain usually the minority class is the more interesting one. Therefore, one may associate a higher misclassification cost with the minority class examples in order to boost their recognition rate. This will penalize the training process for errors on the minority objects and counter the problem of uneven class representations. Cost-sensitive paradigm has been successfully introduced to some types of classifiers, like decision trees or neural networks.

In this paper, we concentrate on the design of cost-sensitive neural networks. We propose to work with the neural classifiers that use the moving threshold approach for incorporating the classification cost. Here instead of re-balancing the training set or modifying the learning procedure, we scale the continuous output of a neural network. Therefore, we modify the classification phase instead of the training phase. Such a scaling forces the classification boundary to be moved towards the objects with higher cost, thus alleviating the bias towards the majority class.

However, the cost factor has a crucial influence on the performance of such a neural classifier and the problem lies in establishing its value. Some problems have the cost supplied by an expert (like in medical domains), but mainly we have no prior information on how to set it. We propose a fully automatic method, based on *Receiver Operating Characteristic* (ROC) curve [4] analysis. We use it to select the best cost factor for a given dataset, that returns balanced performance on both classes.

Our ROC-based cost-sensitive neural network is compared with a set of reference approaches for handling the class imbalance problem. Experimental analysis carried over a set of datasets with varying imbalance ratio proves the usefulness of our approach.

2 Imbalanced Classification

The performance and quality of machine learning algorithms is conventionally evaluated using predictive accuracy. However, this is not appropriate when the data under consideration is strongly imbalanced, since the decision boundary may be strongly biased towards the majority class, leading to poor recognition of the minority class. Disproportion in the number of class examples makes the learning task more complex [12], but is not the sole source of difficulties for machine learning algorithms. It is usually accompanied by difficulties embedded in the structure of data such as small sample size (very limited availability of minority examples), small disjuncts (minority class can consist of several sub-concepts) or class overlapping.

Over the last decade there was developed a number of dedicated techniques for handling such difficulties [1]. They can be divided into two major categories. First one consist of data-level approaches that in the pre-processing stage aim at re-balancing the original distribution [3]. Classifier-level approaches try to adapt existing algorithms to the problem of imbalanced datasets and alleviate bias towards the majority class. Third group relies on cost-sensitive classification and assign higher misclassification cost for minority class, while classification is performed so as to reduce the overall learning cost.

Ensemble systems have also been successfully applied to this domain, and mainly combine a committee learning algorithm (such as Bagging or Boosting) with one of the above mentioned methods [5]. One may also propose a hybrid training procedure for such a combined classifier that will apply cost-sensitive learning locally (for each base classifier) and globally (in the ensemble pruning step) [7].

3 Proposed Cost-Sensitive Neural Network

In this paper, we propose to investigate the cost-sensitive neural network model, based on moving threshold [13].This algorithm concentrates on modifying the output of a classifier, instead of changing the structure of the training data or the training algorithm. The cost-sensitive modification is introduced during the classification step - this the model is being trained in a traditional manner.

Let us assume, that we have a binary imbalanced problem with *majority* and *minority* classes. Then the continuous output of two neurons in the final layer of a neural network for object x can be denoted as $O_{maj}(x)$ and $O_{min}(x)$, where $O_{maj}(x)+O_{min}(x) = 1$ and both outputs are bounded within [0, 1]. In canonical neural network models, the final class is selected according to *winner-takes-all* (WTA) procedure: $\Psi(x) = \arg\max_{m \in \{maj,min\}} O_m(x)$.

However, in moving-threshold model we modify the outputs of the neural network, thus denoting them as $O^*_{maj}(x)$ and $O^*_{min}(x)$. In cost-sensitive threshold-moving model, we compute the output as follows:

$$O^*_{maj}(x) = \eta O_{maj} \mathrm{Cost}[maj, min], \tag{1}$$

and

$$O^*_{min}(x) = \eta O_{min} \mathrm{Cost}[min, maj], \tag{2}$$

where $\mathrm{Cost}[m, n]$ is the misclassification cost between m-th and n-th class, and η is a normalization parameter such that $O^*_{maj}(x)+O^*_{min}(x) = 1$ and both outputs are bounded within [0,1].

One should note, that threshold-moving approaches for neural networks have been overlooked for a long time and is not even close in popularity to sampling-based methods for class imbalance [5]. However, some studies report its high usefulness for dealing with datasets with skewed distributions [9]. Other works

Algorithm 1. Cost-sensitive neural network with moving threshold.

```
Require: training set $\mathcal{TS}$,
    validation set $\mathcal{VS}$,
    neural network training procedure ()
    cost matrix

Training phase:
Train a neural network on $\mathcal{TS}$ using supplied classifier training procedure
Optimize cost parameters using ROC curve analysis on the basis of $\mathcal{VS}$

Classification phase:
obtain continuous outputs $O_{maj}(x)$ and $O_{min}(x)$
$O^*_{maj}(x) \leftarrow$ modify $O_{maj}(x)$ according to Eq. (1)
$O^*_{min}(x) \leftarrow$ modify $O_{min}(x)$ according to Eq. (2)
Apply the WTA procedure on cost-sensitive outputs
```

report, that simply changing the data distribution without considering the imbalance effect on the classification threshold (and thus adjusting it properly) may be misleading [10].

As we can see, the proper settings of cost values has a crucial effect on the performance of this algorithm. Too low cost would lead to insignificant improvements over the standard methods, while too high cost would degrade the performance over the majority class. One must remember, that we cannot sacrifice the majority class, as our ultimate goal is to obtain a classifier with good performance on both classes. In optimal scenario the cost would be supplied by a domain expert according to his/her knowledge. However, in most of real-life imbalanced applications we do not have an access to a pre-defined cost and thus must set it manually. This procedure can be time-consuming, difficult and may lead to an increased classification error when conducted erroneously. So far only simple heuristics were used to calculate the cost, like setting it equal to class imbalance ratio [8].

In this paper, we propose a new method for cost-sensitive neural network training based on ROC curve analysis [7]. Here, we use different values of cost parameter as cut-off points for plotting a ROC curve. Then, we select such a setting of neural classifier that offers the best ratio between the *True Positive rate* and *False Positive rate* (in practice - point located closest to the top left corner of the ROC plot). This allows us for an automatic cost selection that offers a balanced performance on both classes. User only needs to supply a search range and the tuning procedure is conducted in a fully automatic manner.

The detailed steps of the cost-sensitive moving-threshold classifier are presented in a form of pseudo-code in Algorithm 1.

4 Experimental Study

The aims of the experiments were to establish the usefulness of the cost-sensitive moving-threshold neural network, compare it with-state-of-the-art methods for imbalanced classification, and asses the quality of ROC-based cost parameter selection.

In the experiments we have used 10 binary imbalanced datasets from the KEEL repository[1]. Their details are given in Table 1.

Table 1. Details of datasets used in the experimental investigations, with the respect to no. of objects, no. of features, no. of classes, and imbalance ratio (IR).

No	Name	Objects	Features	Classes	IR
1	Haberman	306	3	2	2.78
2	Segment0	2308	19	2	6.02
3	Page-blocks0	5472	10	2	8.79
4	Vowel0	988	13	2	9.98
5	Glass4	214	9	2	15.47
6	Abalone9-18	731	8	2	16.40
7	Yeast6	1484	8	2	41.40
8	Ecoli-0-1-3-7_vs_2-6	281	7	2	39.14
9	Poker-8_vs_6	1477	10	2	85.88
10	Kddcup-rootkit-imap_vs_back	2225	41	2	100.14

As a base classifier we use a single-layer neural network trained with resilient backpropagation algorithm [11]. The number of input neurons is equal to the number of features, output neurons to the number of classes, and the number of neurons in the hidden layer is equal to $\frac{neurons_{input}+neurons_{output}}{2}$. Each neural network is trained for 1000 iterations.

As reference methods for dealing with class imbalance, we combine neural networks with Random Oversampling (NN + OS), Random Undersampling (NN + US), Synthetic minority over-sampling technique (NN + SMOTE) [3], and cost-sensitive moving threshold method with Cost$[minority, majority] = IR$ [8] (NN + MV(IR)).

The proposed method with ROC-based cost optimization (NN + MV(ROC)) uses [5, 200] as a possible range of possible cost parameter values.

We use 5×2 CV F-test for training / testing and pairwise statistical analysis, while Friedman ranking test and Shaffer post-hoc tests are applied for statistical comparison over multiple datasets [6].

The results are given in Table 2. The output of Shaffer post-hoc test is reported in Table 3.

From the obtained results one may see, that the proposed ROC-based cost-sensitive neural network outperforms all other methods in 6 out of 10 cases. What is highly interesting it always delivers superior performance in comparison with the cost selected on the basis of the imbalance ratio. This shows, how important is the proper selection of the cost parameter and that imbalance ratio is not

[1] http://sci2s.ugr.es/keel/imbalanced.php.

Table 2. Results according to G-mean [%] for each examined methods over 10 datasets. Small numbers under proposed method stands for the indexes of reference classifiers that were statistically inferior in the pairwise 5×2 CV F-test. Last row represents the ranks after Friedman test.

Data	NN + OS^1	NN + US^2	NN + SMOTE^3	NN + MT(IR)^4	NN + MT(ROC)^5
1	61.18	61.76	**62.27**	60.46	62.02 1,2,4
2	98.87	98.36	**99.18**	97.00	97.97 –
3	92.73	94.11	93.88	93.64	**94.82** 1,3,4
4	93.18	94.36	94.84	93.88	**95.80** *ALL*
5	81.18	82.74	**83.98**	81.06	82.39 1,4
6	57.44	58.92	57.87	58.14	**60.31** *ALL*
7	82.44	81.89	83.59	83.20	**85.07** *ALL*
8	74.18	76.39	77.99	76.82	**78.38** 1,2,4
9	83.17	**85.66**	85.13	82.78	84.19 1,4
10	71.87	73.22	74.04	73.86	**76.08** *ALL*
Rank	4.25	3.50	2.15	3.80	1.40

the best indicator of the proper misclassification cost. This can be explained by the fact, that IR is not the sole reason behind the imbalanced difficulty. There are other factors, embedded in the nature of data [2]. Therefore, one can easily imagine two hypothetical datasets with identical IR, but completely different classification difficulty. In such situations misclassification cost based purely on IR will definitely fail. This is further confirmed by Shaffer post-hoc test.

When comparing the proposed ROC-based neural network to other solutions, we can clearly see that it easily outperforms Random Oversampling. This is because our cost parameter was optimized to balance the performance on both classes, while RO multiplies the minority class without considering the importance of objects.

On some datasets, the proposed method was inferior to Random Undersampling and SMOTE. This can be explained by the lack of selectiveness of our procedure - it modifies the output for all of examples. Thus it may happen that a correctly recognized examples is weighted towards the incorrect class, which

Table 3. Shaffer test for comparison between the proposed ROC-based cost-sensitive neural network and reference methods over multiple datasets. Symbol '+' stands for a situation in which the proposed method is superior, '−' for vice versa, and '=' represents a lack of statistically significant differences.

Hypothesis	*p*-value
NN + MT(ROC) vs NN + OS	+ (0.0196)
NN + MT(ROC) vs NN + US	+ (0.0348)
NN + MT(ROC) vs NN + SMOTE	+ (0.0402)
NN + MT(ROC) vs NN + MT(IR)	+ (0.0097)

may result in an increased rate of false positives (majority examples misclassified as minority ones). To counter this problem, one would need to introduce a sample selection mechanism, that would apply the cost modification only on examples that are potentially uncertain.

5 Conclusions and Future Works

In this paper we have presented a modification of cost-sensitive neural network classifier based on moving threshold for imbalanced learning domain. We proposed to augment this model with automatic procedure for parameter selection. We applied a ROC-based parameter selection to chose an optimal cut-off point that determined the selected value of misclassification penalty. This allowed for selecting such a parameter, that would offer a balanced performance on both classes.

Experimental evaluation carried out on a number of datasets with varying imbalance ratio confirmed the usefulness of the proposed approach. Our method was always better than the normally used approach for cost parameter estimation. Additionally, in 6 out of 10 cases it was able to outperform the reference methods based on data sampling. This was further backed-up with a thorough statistical analysis. However, the experiments revealed the weak side of our method, that is lack of selectiveness when modifying the output of the classifier.

In future works we plan to develop an active learning solution for selecting important samples to modify the threshold, and propose a dynamic ensemble system based on this classifier.

Acknowledgments. This work was supported by the Polish National Science Center under the grant no. DEC-2013/09/B/ST6/02264.

References

1. Antonelli, M., Ducange, P., Marcelloni, F.: An experimental study on evolutionary fuzzy classifiers designed for managing imbalanced datasets. Neurocomputing **146**, 125–136 (2014)

2. Błaszczyński, J., Stefanowski, J.: Neighbourhood sampling in bagging for imbalanced data. Neurocomputing **150**, 529–542 (2015)
3. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
4. Flach, P.A.: The geometry of ROC space: understanding machine learning metrics through ROC isometrics. In: Proceedings of the Twentieth International Conference on Machine Learning, ICML 2003, 21–24 August 2003, Washington, DC, USA, pp. 194–201 (2003)
5. Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for the class imbalance problem: bagging-, boosting- and hybrid-based approaches. IEEE Trans. Syst. Man Cybern. C Appl. Rev. **42**(4), 463–484 (2012)
6. García, S., Fernández, A., Luengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. Inf. Sci. **180**(10), 2044–2064 (2010)
7. Krawczyk, B., Woźniak, M., Schaefer, G.: Cost-sensitive decision tree ensembles for effective imbalanced classification. Appl. Soft Comput. **14**, 554–562 (2014)
8. Lopez, V., Fernandez, A., Moreno-Torres, J.G., Herrera, F.: Analysis of preprocessing vs. cost-sensitive learning for imbalanced classification open problems on intrinsic data characteristics. Expert Syst. Appl. **39**(7), 6585–6608 (2012)
9. Maloof, M.A.: Learning when data sets are imbalanced and when costs are unequal and unknown. In: ICML-2003 Workshop on Learning from Imbalanced Data Sets II (2003)
10. Provost, F.: Machine learning from imbalanced data sets 101. In: Proceedings of the AAAI 2000 Workshop on Imbalanced Data Sets, pp. 1–3 (2000)
11. Riedmiller, M., Braun, H.: A direct adaptive method for faster backpropagation learning: the RPROP algorithm. In: 1993 IEEE International Conference on Neural Networks, pp. 586–591 (1993)
12. Sun, Y., Wong, A.K.C., Kamel, M.S.: Classification of imbalanced data: a review. Int. J. Pattern Recogn. Artif. Intell. **23**(4), 687–719 (2009)
13. Zhou, Z.-H., Liu, X.-Y.: Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Trans. Knowl. Data Eng. **18**(1), 63–77 (2006)