ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence
and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

Multicriteria optimized MLP for imbalanced learning

Paavo Nieminen and Tommi Kärkkäinen

Department of Mathematical Information Technology
P.O. Box 35, 40014 University of Jyväskylä - Finland

Abstract. Classifier construction for data with imbalanced class frequencies needs special attention if good classification accuracy for all the classes is sought. When the classes are not separable, i.e., when the distributions of observations in the classes overlap, it is impossible to achieve ideal accuracy for all the classes at once. We suggest a versatile multicriteria optimization formulation for imbalanced classification and demonstrate its applicability using a single hidden layer perceptron as the classifier model.

1 Introduction

In many practical applications of classification, the prior classwise frequencies of labelled data vary a lot [1]. Special techniques are constantly emerging to address and successfully manage this well-known situation, e.g., [2]. This paper outlines our current approach to handle imbalanced classification in a proper and natural way, using multilayered perceptron (MLP) as the classifier model and Pareto-based memetic multiobjective optimization (MOO) for learning. An overview of Pareto-based learning is given by [3]. Benefits of the approach include a clean separation of the classification accuracies of all the classes, allowing classifiers to be constructed independently of class frequencies, without the need to specify importances or weights *a priori*, and without external re-sampling techniques. Further possibilities include gaining insights about the properties of the dataset under examination, such as its separability and complexity.

Restricting ourselves to fully supervised classification of continuous input data, we assume that a training dataset of N observations and corresponding integral target class labels $\mathbf{X} = \{(\mathbf{x}_i, t_i) \mid \mathbf{x}_i \in \mathbb{R}^n,\ t_i \in \{1, \ldots, K\}\}_{i=1}^N$ is available, and the task is to use these examples to learn how any $\mathbf{x} \in \mathbb{R}^n$ should be assigned to one of the K classes. By imbalance we mean that the numbers of labelled observations in each class $N_c = |\mathbf{X}_c|$, where $\mathbf{X}_c = \{(\mathbf{x}_i, t_i) \in \mathbf{X} \mid t_i = c\}$, or, equivalently, their frequency of occurrence in the whole dataset, $\phi_c = N_c/N$, are different from each other. Cost-sensitive learning [4] and ROC curves [5] are traditional ways of dealing with the situation. In binary classification, different costs may apply to false positive and false negative predictions. With more classes, even more complex trade-offs must be addressed.

Multiobjective, or multicriteria, optimization (MOO) [6] augments the classical field of optimization by considering multiple objective functions (aka. cost functions, fitness functions, criteria) simultaneously. Without loss of generality, we can consider minimization only. For m objectives, we are to solve

$$\min_{X \in \Omega} \quad \mathbf{f}(X) = (f_1(X), f_2(X), \ldots, f_m(X)),$$

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

where $f_i(\cdot)$ denotes the ith real-valued objective function and Ω the set of admissible values for the unknown X. We adopt the concept of *Pareto-optimality* based on the *dominance relation*. A solution image $\mathbf{f}(X')$ is said to dominate $\mathbf{f}(X)$, notated $\mathbf{f}(X') \preceq \mathbf{f}(X)$, if $f_i(X') \leq f_i(X)$ for all $i \in \{1, \ldots, m\}$ and $f_i(X') < f_i(X)$ for at least one $i \in \{1, \ldots, m\}$. Instead of a single optimum, of interest is the *Pareto set* (PS) of *nondominated* solutions for which no dominating solutions exist, $P = \{X \in \Omega \mid \neg\exists Y \in \Omega : \mathbf{f}(Y) \preceq \mathbf{f}(X)\}$. The image of the PS in the objective space $\mathbf{f}(P)$ is called the *Pareto Front* (PF). Intuitively, a solution not in the PS is suboptimal because improvements are possible without compromises. In MOO methods based on sampling the (generally infinite) PS, an approximation of the PF is presented to the user, who ultimately has to select a preferred compromise solution. The PF exploration itself can reveal insights into the MOO problem.

Multiobjectivity pervades machine learning: For example, any MLP learning algorithm using a complexity penalty such as “weight decay”, can be considered an instance of MOO where a single solution of the PS is sampled via scalarization using pre-assigned weights [6]. Furthermore, different measures of complexity may be mutually conflicting, e.g., an MLP with fewer hidden neurons (less complex) may require larger weights (more complex) to fit the training data. Clearly, imbalanced classification is also an instance of MOO whenever a single solution cannot achieve full accuracy for all the classes. Based on these observations, we argue that MOO provides a holistic and preferrable framework for neural network design.

A multi-class MOO interpretation of the ROC is used by [7] to minimize the $K \times (K-1)$ misclassification rates possible with K classes, i.e., predictions to class i when j would be correct. The resulting quadratic increase in the number of objectives for multi-class scenarios is known to be problematic for traditional MOO algorithms. An extreme solution [8] is to reduce the information by considering only the worst class-wise misclassification rate. The number of objectives remains small, but all information about the performance on different classes is lost when $K > 2$. We opt for a mid-way solution briefly hinted to by [7].

The inherent MOO nature of the popular support vector machine (SVM) has been used for Pareto-based imbalanced classification for example in [9]. However, SVM is inherently a binary classification technique although it can be extended for multiple classes and nonlinear regression with special enlargements. MLP with its basic structure is readily suitable to represent as many classes (or regression outputs) as needed, which is why we prefer it. In MLP training, it is customary (e.g., [8]) to combine a global evolutionary optimization framework and a traditional gradient-based local search. Such hybrids fall within the category of memetic algorithms (MA) [10] which we assume as the methodological basis in this work.

In some works (e.g., [11]), an evolutionary MOO step is used to handle class imbalance as an intermediate phase, but the algorithm still remains single-objective in the sense that a final solution is eventually selected that minimizes the total accuracy. The recent survey [12] lists further applications of evolutionary MOO in classification and other data mining tasks. The multiclass approach followed here was not included in the findings of [12], which makes us more confident of our view that the community may have recently been less interested in some earlier ideas [3, 7] that should be revived and ultimately combined with the newer developments. From [12] we observe that usual candidates for individual MOO criteria for MLP training are related to total accuracy and network complexity, but with fewer proposals to divide the classwise error rates and to connect the ROC methodology to scenarios with more than two classes. NSGA-II [13] seems to be the most popular MOO algorithm [12, 14], hybridized with a local solver as in [15].

Next, in Section 2, we introduce our method. In Section 3, we demonstrate its use and identify future possibilities. We briefly conclude in Section 4.

2 The method

Algorithm 1 outlines a Pareto-based MOO strategy that combines the generic memetic operations [10] of "Initialization", "Cooperation" and "Improvement" of a population of solution candidates with the non-dominated sorting (NS) approach of NSGA-II [13]. By leaving MOO up to NS and adopting the established point of view and terminology of MAs, we can focus directly on the rich possibilities in defining specific operators for our application of interest, which is now MLP classifier training for imbalanced data.

The *problem definition* consists of the objective functions. Given a multiclass dataset, we wish to minimize the classwise numbers of misclassifications $f_c = |\{(\mathbf{x}_i, t_i) \mid t_i = c \neq \text{prediction}(\mathbf{x}_i)\}|$ for $c = 1, \ldots, K$, yielding K objectives.

For simplicity, we restrict ourselves to the single-hidden-layer feedforward network (SLFN) as the classifier model. Its action for a given vector $\mathbf{x} \in \mathbb{R}^{n_0}$ can be formalized (see [16]) as $\mathbf{W}\mathcal{F}(\mathbf{V}\tilde{\mathbf{x}})$, where $\mathbf{V} \in R^{n_1 \times (n_0+1)}$ contains the hidden weights with bias values in its first column, $\mathbf{W} \in R^{n_2 \times (n_1+1)}$ the outer-layer weights with biases, $\tilde{\mathbf{x}}$ denotes the enlargement of a vector as $[1\ \mathbf{x}^T]^T$ for

Algorithm 1: General memetic MOO with non-dominated sorting

```
input  : Definition of the problem, encodings, parameters
output: Approximated sample of the Pareto set
parents ← Initialize() ;
parents ← Improve(parents) ;
AssignRankAndCrowdingDistance(parents) ;
repeat
    children ← Cooperate(parents) ;
    children ← Improve(children) ;
    parents ← NondominatedSort(parents ∪ children) ;
until Iteration limit reached;
return parents;
```

the bias computation, and $\mathcal{F}$ denotes the application of an activation function (logistic sigmoid here) and enlargening of the result similarly to $\tilde{\mathbf{x}}$. Thus, our set of admissible solutions is $\Omega = \mathbb{R}^{n_2\times(n_1+1)} \times \mathbb{R}^{n_1\times(n_0+1)}$. The *encoding* of such a solution is simply the storage of the real-valued weights. For the experiment here, we use a population of 100 individual solution candidates encoded in this way. Other *parameters* controlling the method steps will be explained below.

We *initialize* the first population by assigning uniform random weights from the distribution $U([-1,1])$. Then, the first round of *improving* the population is performed by running at most 200 steps of the conjugate gradient method on the single-objective (scalarized) cost function

$$\mathcal{J}(\mathbf{V},\mathbf{W}) = \sum_{c=1}^{K} \lambda_c \sum_{i=1}^{N_c} \left\| \mathbf{W}\mathcal{F}(\mathbf{V}\tilde{\mathbf{x}}_{p_c(i)}) - \mathbf{t}_{p_c(i)} \right\|^2 + \mu \sum_{w\in\{V,W\}} |w|^2,$$

where $\lambda_c \sim U([0.1,1])$ and $\mu \sim U([10^{-6},10^{-4}])$ are random values, p_c an index mapping to reach observations of class c, vector $\mathbf{t}_{p_c(i)} \in \mathbb{R}^K$ is the usual representation where the $t_{p_c(i)}$th component is 1 and other components are 0. This traditional local search restricts a solution to a meaningful MLP after disturbance by the evolutionary operations that facilitate global exploration. The complexity penalty imposes a necessary soft constraint. We hasten to emphasize that while the internal improvement step is single-objective and uses a function *different from the objectives* evaluated in the evolution, the *randomized weights keep the overall search multiobjective*, and the user is free from having to make strong assumptions of the problem before viewing the resulting PF. Even a very simple equation like this seems to work in practice, as illustrated in Section 3.

In the example here, the *cooperation* step uses only a unary mutation operator that perturbs each weight by noise drawn from $n(0, 0.4)$ with a probability of 10%. Extending the cooperation step with crossover operators designed specifically for MLP classifiers is another main focus of our current research.

3 Experimental evidence of suitability

Figure 1 illustrates the class imbalance problem with a simulated dataset in which $n = 2, K = 3, N = 400, \phi_1 = 5\%, \phi_2 = 10\%$, and $\phi_3 = 85\%$. The case is made difficult by letting all of the classes overlap in the middle, but none of them contain subconcepts or noise (cf. [1]) by design. The decision boundaries of different classifier candidates from the approximated PS are also shown, indicated by colors and numerical identity tags. The MLP layers had $n_0 = 2$, $n_1 = 20$, and $n_2 = 3$ units. Not one of the classifiers can be deemed better than the other without assigning some preferred costs to misclassifications in one class versus the other, so the pure multiobjective nature of the problem can be clearly seen. Improvement in one class implies degradation in another. Figure 2 shows a parallel coordinate plot where the trade-offs between objective functions can be compared visually.

The solutions were obtained by running the described memetic MOO optimization algorithm for 100 iterations. Noteworthy with regard to the prospects

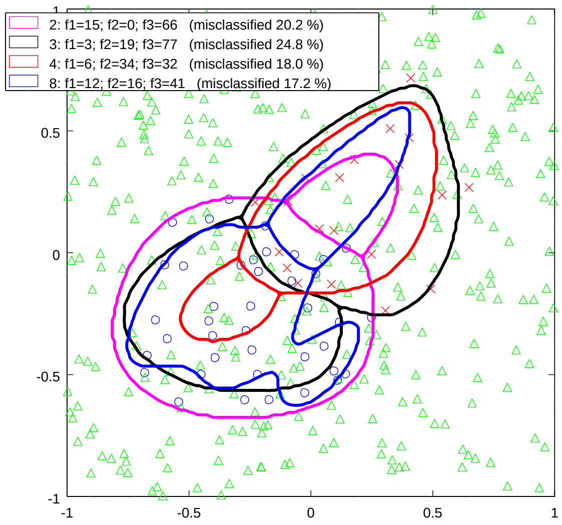

Fig. 1: Post optimization exploration of pareto-optimal classifiers.

of our approach is the fourth solution (ID tag 8). It contains sharper edges and seems to be overfitting the example data. With less versatile learning frameworks one needs to compete against this kind of behavior by pre-assigning constraints or penalties to the network complexity. In a MOO setting, it will be trivial to add network complexity measure(s) as additional objectives, turning the danger of overfitting into the prospect of gaining valuable information of the complexity of the data itself. The memetic approach in general does not require limitations to be set *a priori*, even if we have been using them in this early demonstration. To really set the framework and its user free, further MLP-specific development of the memetic improvement (lifetime learning) and cooperation (recombination) operators and solution encoding needs to be done. We shall pursue these in follow-up publications.

4 Conclusions and future work

In this paper, we demonstrated how Pareto-based MOO provides a natural approach to handle imbalanced classification using a mid-way approach among proposed alternatives. The application shown is one part of our current pursuit of a combined methodology involving objective functions to jointly and explicitly handle also MLP complexity, generalization, and the possibility of mislabeled data. Within the past decade, after the proliferation of MOO methods for MLP learning, holistic approaches appear to have been considered surprisingly little, as only a few aspects have been explicitly formulated as separate objectives in each study. Therefore, we see it fit to remind the community about the possibilities in Pareto-based multiobjective learning.

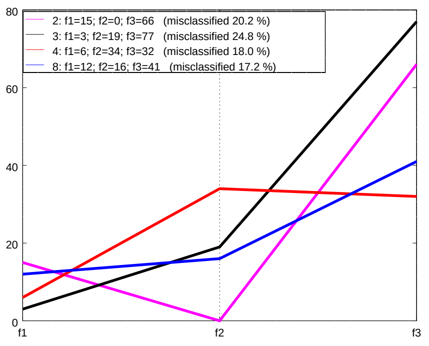

Fig. 2: Parallel coordinates view of the solutions in Figure 1.

References

[1] H. He and E. A. Garcia. Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.*, 21(9):1263–1284, 2009.

[2] W. Zong, G.-B. Huang, and Y. Chen. Weighted extreme learning machine for imbalance learning. *Neurocomputing*, 101:229–242, 2013.

[3] Y. Jin and B. Sendhoff. Pareto-based multiobjective machine learning: An overview and case studies. *IEEE Trans. Syst., Man, Cybern., Part C*, 38(3):397–415, 2008.

[4] C. Elkan. The foundations of cost-sensitive learning. In *IJCAI'01*, pages 973–978, 2001.

[5] T. Fawcett. An introduction to ROC analysis. *Pattern Recognit. Lett.*, 27(8):861–874, 2006.

[6] M. Ehrgott. *Multicriteria Optimization*. Springer-Verlag New York, Inc., 2005.

[7] R.M. Everson and J.E. Fieldsend. Multi-class ROC analysis from a multi-objective optimisation perspective. *Pattern Recognit. Lett.*, 27(8):918 – 927, 2006.

[8] J.C. Fernandez-Caballero, F.J. Martinez, C. Hervas, and P.A. Gutierrez. Sensitivity versus accuracy in multiclass problems using memetic pareto evolutionary neural networks. *IEEE Trans. Neural Netw.*, 21(5):750–770, 2010.

[9] Ayşegül Aşkan and Serpil Sayın. SVM classification for imbalanced data sets using a multiobjective optimization framework. *Ann. Oper. Res.*, 216(1):191–203, 2013.

[10] F. Neri, C. Cotta, and P. Moscato, editors. *Handbook of Memetic Algorithms*. Springer Publishing Company, Inc., 2012.

[11] S. García, R. Aler, and I. M. Galván. Using evolutionary multiobjective techniques for imbalanced classification data. In *ICANN 2010*, pages 422–427, 2010.

[12] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, and C. A. Coello Coello. A survey of multiobjective evolutionary algorithms for data mining: Part I. *IEEE Trans. Evol. Comput.*, 18(1):4–19, 2014.

[13] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.*, 6(2):182–197, 2002.

[14] S. Poles, M. Vassileva, and D. Sasaki. *Multiobjective Optimization Software*, pages 329–348. Springer-Verlag, 2008.

[15] H. A. Abbass. Speeding up backpropagation using multiobjective evolutionary algorithms. *Neural Computation*, 15:2705–2726, 2003.

[16] T. Kärkkäinen. MLP in layer-wise form with applications in weight decay. *Neural Computation*, 14:1451–1480, 2002.