AUC-based gradient boosting for imbalanced classification

Publisher:
Centro Editoriale di Ateneo Università di Cassino e del Lazio Meridionale

Published version:
DOI:

AUC-BASED GRADIENT BOOSTING FOR IMBALANCED CLASSIFICATION

Martina Dossi[1], Giovanna Menardi[2]

[1] European Central Bank[†] (e-mail: martina.dossi@ecb.europa.eu)

[2] Dip. di Sc. Statistiche, Università di Padova (e-mail: menardi@stat.unipd.it)

ABSTRACT: Classification problems with imbalanced class distributions are pervasive in a plurality of real-world applications, such as network intrusion detection, fraud detection and rare disease diagnosis. In this context, most of standard classification models are heavily compromised, as they tend to focus on the majority class, yet the minority class is often the one of greatest importance. To tackle the problem, we combine *XGBoost*, a powerful and recent formulation of the *gradient boosting*, with a loss function specifically derived to optimise the Area Under the ROC curve, an evaluation metric more robust towards class imbalance.

KEYWORDS: AUC, boosting, classification, class imbalance

1 Introduction

Class imbalance refers to all supervised classification tasks which suffer of uneven class distributions. The issue has gained ground with some further implicit assumptions, such that imbalanced data are expected to have rare instances belonging to the class of greatest interest and a (relatively) large number of units from the other classes. An imbalanced class distribution may severely affect the performance of classification algorithms, by interfering with both model estimation and accuracy evaluation phases. Disregarding each model own specificities, model estimation is typically driven by the optimisation of a global loss function, which favours classification rules ignoring the rare units as overwhelmed by the prevalent class. A number of techniques have been developed to cope with imbalanced classes: data level approaches attempt to re-balance the class distribution before building learning models, whereas classifier level approaches aim to adapt existing algorithms to focus on the minority class. The latter group includes cost-sensitive techniques, methods that replace the loss function with more meaningful measures and combinations of classifiers, that follow the logic of *boosting*, *bagging* and *random forests*.

[†] Disclaimer: this document reflects authors' views, not necessarily shared by ECB.

Under imbalanced scenarios, assessing the performance of a classifier plays a role that is at least as crucial as its estimation. Accuracy, which is the most commonly used metric for classification tasks, is not sufficient, as it is governed by the majority class. Other performance metrics which account for the class distribution are preferred in this context, as the G-mean, the F-measure, and especially the Area Under the ROC Curve (AUC). See Menardi & Torelli (2014) for a more comprehensive discussion about the imbalance problem.

Within the logic of the approaches at a classifier level, in this work we derive a differentiable loss function that optimises the AUC to train a gradient-based model within the *boosting* family, in order to extend the benefits of the AUC as evaluation metric to the phase of model estimation. After presenting the building blocks relevant for a full comprehension of the proposed method, we discuss our contribution and show some numerical results.

2 Gradient boosting optimisation based on the AUC

Given a training set $\mathcal{T}_n$ containing n *i.i.d.* pairs $(\mathbf{x}_i, y_i)$, $i = 1, \dots, n$, where $\mathbf{x}_i \in \mathbb{R}^d$ is a vector of attributes and $y_i \in \{\mathcal{Y}_0, \mathcal{Y}_1\}$ is a response variable whose classes are conventionally labeled as negative and positive respectively, a classifier $\mathcal{H} : \mathcal{X} \mapsto \mathbb{R}$ is a function that allows to predict the response variable y, based on the observed $\mathbf{x}$. The output $\mathcal{H}(\mathbf{x})$ measures the confidence of $\mathbf{x}$ belonging to the positive class, whereas the predicted label $\hat{y}$ is defined on the basis of a threshold $k \in \mathbb{R}$ such that $\hat{y} = \mathcal{Y}_0$ if $\mathcal{H}(\mathbf{x}) < k$ and $\hat{y} = \mathcal{Y}_1$ otherwise. A non-negative loss function $\mathcal{L}(y, \hat{y})$, that measures the discrepancy between observed and fitted values, is used either to optimize the classifier during the learning process and to assess the performance of the model.

Even if not specifically developed to tackle the class imbalance problem, the *gradient boosting* (Friedman, 2001) has showed to achieve competitive results in this domain. In broad terms, it exploits the connection between *AdaBoost*, the first applicable approach of *boosting*, that relies on the idea of increasing the weight of the hardest to classify units, and a forward-stagewise additive modeling approach. At each iteration of the algorithm, a functional gradient descent optimisation is applied to a loss function, in the n-dimensional space of the fitted values, and it is then approximated by some simple model. The final rule is a linear combination of all the previous estimated functions. A specific formulation of the *gradient boosting* is *XGBoost* (Chen & Guestrin, 2016), which, at each iteration, approximates the objective loss function by a second order Taylor's series expansion, and estimates a classification model via its minimisation. This implementation easily supports different loss functions, as it is sufficient to provide the algorithm with its first two derivatives.

The rationale behind the proposed approach is to integrate into the *XGBoost* a loss function independent on the class distribution. In this perspective, the AUC - its ones' complement, in fact - represents a sensible candidate.

Let n_+ and n_- be the sample size of positive and negative observations respectively, and assume that $\mathcal{H}(\mathbf{x}_i^+)$ and $\mathcal{H}(\mathbf{x}_j^-)$ are the fitted scores respectively for the i-th positive and the j-th negative instances. The AUC is equivalent to the normalized Wilcoxon Mann-Whitney statistic, in the form:

$$AUC = \frac{1}{n_+n_-}\sum_{i=1}^{n_+}\sum_{j=1}^{n_-}\mathbb{I}_{0.5}(\mathcal{H}(\mathbf{x}_i^+) - \mathcal{H}(\mathbf{x}_j^-)), \tag{1}$$

where $\mathbb{I}_{0.5}(t)$ is 0 if $t < 0$, 0.5 if $t = 0$, 1 otherwise. The AUC estimates the probability that a positive unit receives a higher score than a negative one by means of comparisons between instances belonging to different classes. While the global accuracy of a classifier depends on the choice of a classification threshold, the AUC evaluates its discriminating ability as the threshold varies over all its range. This allows to cater for the presence of rare units as, by construction, it does not place more emphasis on one class over the other.

Unfortunately, two issues prevent the expression (1) from being directly used as a loss function: first and foremost, the function is non differentiable, secondly, its argument is not the single observation but rather refers to pairs of instances. To overcome the first limitation, we consider the following differentiable approximation (Yan *et al.*, 2003):

$$\mathcal{U}_s = \frac{1}{n_+n_-}\sum_{i=1}^{n_+}\sum_{j=1}^{n_-}\mathcal{S}(\mathcal{H}(\mathbf{x}_i^+), \mathcal{H}(\mathbf{x}_j^-)), \text{ where:} \tag{2}$$

$$\mathcal{S}(\mathcal{H}(\mathbf{x}_i^+), \mathcal{H}(\mathbf{x}_j^-)) = \begin{cases} (-(\mathcal{H}(\mathbf{x}_i^+) - \mathcal{H}(\mathbf{x}_j^-) - \tau))^p & \text{if } \mathcal{H}(\mathbf{x}_i^+) - \mathcal{H}(\mathbf{x}_j^-) < \tau, \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

for a given $\tau \in (0,1]$ and $p > 1$ selected by the user. A pair of observations contributes to the loss function when the score of a positive unit exceeds the one of a negative unit by τ. The authors suggest to choose $\tau \in [0.1, 0.7]$ and $p \in \{2,3\}$. The quantity $\mathcal{U}_s$ is then reformulated to refer to unique instances:

$$\mathcal{U}_s = \frac{1}{n_+n_-}\sum_{i=1}^{n}\left[\mathbb{I}_{(y_i=1)}\sum_{i'=1}^{i-1}\mathcal{S}_{i'}^{+} + \mathbb{I}_{(y_i=-1)}\sum_{i'=1}^{i-1}\mathcal{S}_{i'}^{-}\right], \text{ where:} \tag{4}$$

$\mathcal{S}_{i'}^{+} = \mathbb{I}_{(y_{i'}=-1)}\mathcal{S}(\mathcal{H}(\mathbf{x}_i), \mathcal{H}(\mathbf{x}_{i'}))$ and $\mathcal{S}_{i'}^{-} = \mathbb{I}_{(y_{i'}=1)}\mathcal{S}(\mathcal{H}(\mathbf{x}_{i'}), \mathcal{H}(\mathbf{x}_i))$. Once the parameters are defined, the computation of the first two derivatives is straightforward and the method can be implemented.

Empirical results reveal that the proposed approach outperforms many other competitive classifiers, especially in scenarios of extreme rarity and nontrivial data patterns. In the bidimensional setting illustrated in Figure 1, as well as in its generalisation in 5 dimensions, rare units lie in small disjunct sets, overlapping with the majority class at the margins of each box. The results of the analysis are outlined in Table 1. As expected, standard models as the logistic regression and the classification tree fail in this domain. The algorithm *SMOTEBoost* (Chawla *et al.*, 2003), specifically developed to address the imbalance, performs even worse than the original *AdaBoost*. Conversely, the modified *XGBoost* achieves better results in the majority of the cases, including the hardest.

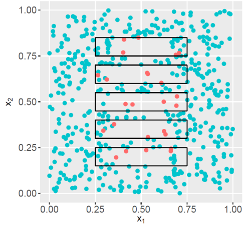

Figure 1: Simulated data in the bidimensional space. Red dots represent the rare instances.

%	**dim.**	*Logistic Reg.*	*Tree (Gini)*	*Ada-Boost*	*SMOTE-Boost*	*Gradient boosting*	*Modified XGBoost*
0.6	2	0.500 (0.004)	0.500 (0.000)	0.772 (0.047)	0.602 (0.059)	0.782 (0.043)	**0.790** (0.041)
	5	0.500 (0.012)	0.500 (0.000)	0.721 (0.041)	0.563 (0.059)	0.712 (0.040)	**0.736** (0.042)
1	2	0.500 (0.003)	0.501 (0.008)	0.830 (0.034)	0.632 (0.059)	**0.838** (0.030)	0.833 (0.030)
	5	0.499 (0.008)	0.501 (0.014)	0.786 (0.032)	0.609 (0.067)	0.777 (0.030)	**0.790** (0.031)

Table 1: Average AUC (and standard deviation) over 300 Monte Carlo samples of size 1000, with dimension 2 and 5, rare class frequency of 0.6% and 1%. For *boosting* algorithms 200 iterations were considered; for the AUC-based loss function $\tau = 0.7$ and $p = 2$.

References

CHAWLA, N. V., *et al.* 2003. SMOTEBoost: Improving prediction of the minority class in boosting. *European conference on principles of data mining and knowledge discovery*, Springer, 107-119.

CHEN, T. & GUESTRIN, C. 2016. XGBoost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, ACM, 785-794.

FRIEDMAN, J. H. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189-1232.

MENARDI, G., & TORELLI, N. 2014. Training and assessing classification rules with imbalanced data. *Data Mining and Knowledge Discovery*, **28(1)**, 92-122.

YAN, L., *et al.* 2003. Optimizing classifier performance via an approximation to the Wilcoxon-Mann-Whitney statistic. *Proceedings of the 20th International Conference on Machine Learning*, 848-855.