Weighted One-Against-All

Alina Beygelzimer
IBM T. J. Watson Research Center
Hawthorne, NY 10532

John Langford
TTI-Chicago
Chicago, IL 60637

Bianca Zadrozny
IBM T. J. Watson Research Center
Yorktown Heights, NY 10598

Abstract

The one-against-all reduction from multiclass classification to binary classification is a standard technique used to solve multiclass problems with binary classifiers. We show that modifying this technique in order to optimize its error transformation properties results in a superior technique, both experimentally and theoretically. This algorithm can also be used to solve a more general classification problem "multi-label classification," which is the same as multiclass classification except that it allows multiple correct labels for a given example.

Introduction

In multiclass learning the goal is to train a classifier that predicts one of n labels for each test instance, given a set of labeled training examples. Many machine learning problems can be phrased in terms of multiclass classification with such examples as character recognition and document categorization. Binary classification, where the number of labels is two, is the simplest case that requires predicting just a single bit for each instance. For example, the question may be whether an object has a given property or not.

Given that we have many good binary learning algorithms and many multiclass classification problems, it is tempting to create meta-algorithms which use binary classifiers to make multiclass predictions.

Probably the simplest such method is the One-Against-All reduction (see, for example, (Dietterich & Bakiri 1995)) which creates one binary problem for each of the n classes. The classifier for class i is trained to predict "Is the label i or not?" thus distinguishing examples in class i from all other examples. Predictions are done by evaluating the n classifiers and randomizing over those which predict "yes," or over all n labels if all answers are "no". By analyzing the error transform properties of this reduction, we show that an average error rate ϵ of the learned binary classifiers induces an error rate of at most $(n-1)\epsilon$ for multiclass classification.

A careful consideration of the error transformation proof of One-Against-All reveals that the false positive and false negative failure modes are asymmetric. A false negative (i.e., predicting "no" when the correct label is "yes") is more disastrous than a false positive (i.e., predicting "yes" when the correct label is "no") because a false negative results in a $\frac{1}{n}$ probability of correct prediction, while for a false positive this probability is $\frac{1}{2}$ (provided that there are no other errors). Consequently, carefully making the learned classifier more prone to say "yes" results in a lower multiclass error rate. We accomplish this by first reducing to importance weighted binary classification, where each example has some importance, and the goal is to minimize the importance-weighted error rate. We then compose this reduction with the Costing reduction (Zadrozny, Langford, & Abe 2003) to remove the importances (by altering the underlying distribution using rejection sampling). The exact choice of importances is given by optimizing the main theorem so as to achieve a transformed error rate of roughly $\frac{n}{2}\epsilon$. Empirically, this algorithm results in superior performance on most tested binary learning algorithms and problems.

The setting we analyze actually applies to a more general problem of *multi-label classification* which is similar, except that any example may have multiple correct labels. This generalization essentially comes for free. Note that our analysis does not assume that the correct labels are independent (Rifkin & Klautau 2004).

There are many ways of reducing multiclass classification to a set of binary classifications. Perhaps the most mathematicaly compelling of these is the ECOC reduction (Dietterich & Bakiri 1995). The idea is to train a set of binary classifiers, each deciding membership in some subset of labels. So given a sequence of subsets, each of the n labels corresponds to a binary string (or a codeword) defined by the inclusion of this label in the sequence of subsets. A multiclass prediction is made by finding the codeword closest in Hamming distance to this sequence of binary predictions on the test example.

The error rate of the resulting multiclass classifier can be shown to be at most four times the average error rate of the individual binary classifiers (Guruswami & Sahai 1999; Beygelzimer *et al.* 2004). The proof of this statement is essentially the observation that there exist codes where the distance between any two codewords is at least 1/2. Consequently, at least 1/4 of the classifiers must err to induce a multiclass classification error, implying the bound.

Thus ECOC achieves an error transformation of 4ϵ independently of the number of labels, which is theoretically much more appealing than the $\frac{n}{2}\epsilon$ transformation achieved by Weighted One-against-all. However, the result presented here is still relevant for two reasons.

1. The transformations above are stated in terms of the average error rate ϵ of the learned binary classifiers. Thus nothing prevents a reduction from taking an "easy" multiclass problem and turning it into "hard" problems for the binary learner. And in fact, ECOC reductions with the dense codes required for the 4ϵ result tend to create artificially hard binary learning problems, as empirically observed by a number of authors (Rifkin & Klautau 2004; Guruswami & Sahai 1999; Allwein, Schapire, & Singer 2001). Consequently, the one-against-all reduction can empirically perform just as well (if not better) than the ECOC reduction.
2. The method used is orthogonal to the choice of reduction, and so it may be useful for optimizing ECOC and other reductions as well.

The paper has the following structure. First we introduce basic notions and analyze the original one-against-all reduction. We then present our new reduction and its analysis. Finally, we provide experimental support for the new reduction and conclude with a discussion of how this work relates to other approaches.

Basic Notions

A *binary classification learner* takes as input binary labeled training examples $(x_1, y_1), \ldots, (x_m, y_m)$ from $X \times \{0, 1\}$, where X is some feature space and $\{0, 1\}$ is the binary label. The goal of the learner is to use the examples to generate a classifier $h : X \rightarrow \{0, 1\}$ which minimizes the expected error rate

$$e(h, D) = E_{(x,y)\sim D}\left[I(h(x) \neq y)\right],$$

with respect to the test distribution D over $X \times \{0, 1\}$, where $I(\cdot)$ is the indicator function which is 1 when the argument is true, and 0 otherwise.

Multiclass classification is defined similarly except that the labels are in $\{1, \ldots, n\}$ for $n > 2$. In *multi-label classification* there are also more than two classes, but a given example can belong to more than one class. Thus examples are of the form (x, Y), where $x \in X$, and Y is a subset of $\{1, \ldots, n\}$. The expected error rate is then defined as

$$e_S(h, D) = E_{(x,Y)\sim D}\left[I(h(x) \notin Y)\right],$$

where the subscript S stands for "subset".

Importance weighted binary classification is an extension of binary classification where there is some importance associated with misclassifying each example. The learner gets to know the importances of training examples, and the goal is to minimize the expected importance-weighted loss,

$$e_W(h, D) = E_{(x,y,w)\sim D}\left[wI(h(x) \neq y)\right],$$

where the test distribution D is over $X \times \{0, 1\} \times [0, \infty)$.

We want to show how the performance of binary classifiers on subproblems generated by a given reduction translates into the performance on the multiclass problem. When performance is measured in terms of the expected error rate, such statements are called *error transformation bounds* of the reduction. To state the bounds, we will need to define how D induces a distribution for the created binary problems.

When analyzing error transformation properties, our goal will be to characterize the most efficient way in which any adversary can induce multiclass errors with a fixed budget on binary errors. The *error efficiency* of a reduction on a given example is defined as the maximum ratio of the probability of a multiclass error to the number of binary errors on the examples generated by the reduction.

Combining Multiple Binary Subproblems into One To simplify the description of our algorithms and analyses, we use a general transformation for turning multiple calls to a binary learning algorithm into a single call. We simply augment the feature space with the name of the call and then learn a combined classifier on the union of all training data.

The One-Against-All Reduction

In the one-against-all reduction, we learn n binary classifiers. For $i \in \{1, \ldots, n\}$, classifier b_i is trained using the mapping $(x, y) \rightarrow (x, I(y = i))$ from multiclass examples to binary examples. In order to construct a multiclass classifier from the binary classifiers, we use the following procedure: If there exists a label i such that $b_i(x) = 1$, then predict i, breaking ties randomly; predict randomly otherwise.

The algorithms formally specifying the reduction are given below. We combine the calls into one using the transformation discussed above.

1 OAA-Train (Set of n-class examples S, binary classifier learning algorithm B)

```
Set $S' = \emptyset$.
for all examples $(x, y)$ in $S$ do
    for all labels $i \in \{1, \ldots, n\}$ do
        Add a binary example $(\langle x, i\rangle, I(y = i))$ to $S'$.
    end for
end for
Return $h = B(S')$.
```

To state the transformation bound, we must define how a multiclass test distribution D induces a test distribution OAA-Train(D) over on the combined binary

```
2 OAA-Test (Binary classifier $h$, test example $x$)

Output $\text{argmax}_i h(\langle x, i\rangle)$ where $i \in \{1, \dots, n\}$ and ties are broken randomly.
```

classifier. To draw a sample from OAA-Train(D), we draw a multiclass sample (x, y) from D, a random index $i \in \{1, \dots, n\}$, and output $(\langle x, i\rangle, I(y = i))$. Recall that $e(h, D)$ denotes the error rate of classifier h on distribution D. The theorem below is known (Allwein, Schapire, & Singer 2001; Guruswami & Sahai 1999) for relating training set error rates. Modifying these results to relate test set error rates is straightforward, but the is instructive because it reveals the assymetry that we later exploit in the Weighted One Against All reduction.

Theorem 1. *(One-against-all error efficiency) Given any binary learner B and a set of examples S in $(X \times \{1, \dots, n\})^*$, let $h =$ OAA-Train(B, S). For all test distributions D on $X \times \{1, \dots, n\}$,*

$$e(\text{OAA-Test}(h), D) \leq (n-1)e(h, \text{OAA-Train}(D)).$$

Proof. We analyze how false negatives (predicting 0 when the correct label is 1) and false positives (predicting 1 when the correct label is 0) produced by the binary classifiers lead to errors in the multiclass classifier. A false negative produces an error in the multiclass classifier a $\frac{n-1}{n}$ fraction of the time (assuming all the other classifiers are correctly outputting 0), because we are choosing randomly between n labels and only one is correct. The other error modes to consider involve (possibly multiple) false positives. If there are k false positives, the error probability is either $\frac{k}{k+1}$ or 1 if there is also a false negative. The efficiency of these three modes in creating errors (i.e., the maximum ratio of the probability of a multiclass error to the number of binary errors) is $\frac{\frac{n-1}{n}}{1} = \frac{n-1}{n}$, $\frac{\frac{k}{k+1}}{k} = \frac{1}{k+1}$, and $\frac{1}{k+1}$, respectively. Taking the maximum, we get $\frac{n-1}{n}$. Multiplying by n (since we have n opportunities to err, one for each classifier), we get the result. ∎

Notice that the analysis actually shows that the multiclass error can be as high as $(n-1)\epsilon$.

Weighted One Against All

The key to improving the reduction above is an observation that the false positive and false negative failure modes are asymmetric: a false negative implies only a $1/n$ probability of correct prediction while a false positive implies only a $1/2$ probability of correct prediction. Thus one might hope that making the learned classifier more prone to output 1 will result in a lower multiclass error rate. We accomplish this by first reducing to importance weighted binary classification, and then composing this reduction with the Costing algorithm (Zadrozny, Langford, & Abe 2003) to reduce all the way to binary classification.

As mentioned in the introduction, the reduction actually applies to a more general problem of multi-label classification. This extra property comes for free.

A training example in multi-label classification is labeled by some subset Y of n possible labels. Let k be the number of labels in Y. The reduction maps each example (x, Y), to n examples of the form:

$$(\langle x, y\rangle, I(y \in Y), w_{I(y \in Y)}) \text{ for } y \in \{1, \dots, n\},$$

where $w_0 = \frac{n}{k+1}$, and $w_1 = \frac{n}{k+1}$ if $n \leq k^2 + k$, otherwise $w_1 = \frac{n-k}{k}$. The oracle uses these examples to construct a binary importance-weighted classifier h. To construct a multi-label classifier from h, we do the following: If there exists a label $y \in \{1, \dots, n\}$ such that $h(\langle x, y\rangle) = 1$, then predict y, breaking ties randomly; predict randomly otherwise.

The algorithms below describe the reduction more formally.

```
3 WOA-Train (Set of multi-label examples $S$, importance-weighted binary classifier learning algorithm $B$)

Set $S' = \emptyset$ and define
    $w_0 = w_0(k) = \frac{n}{k+1}$,
    $w_1 = w_1(k) = \begin{cases} n/(k+1), & \text{if } n \leq k^2 + k \\ (n-k)/k, & \text{otherwise} \end{cases}$
for all examples $(x, Y)$ in $S$ do
    for all labels $y \in \{1, \dots, n\}$ do
        Add a binary importance-weighted example
            $(\langle x, y\rangle, I(y \in Y), w_{I(y \in Y)})$ to $S'$.
    end for
end for
Return $h = B(S')$.
```

```
4 WOA-Test (Binary importance weighted classifier $h$, test example $x$)

Output $\text{argmax}_y h(\langle x, y\rangle)$ where $y \in \{1, \dots, n\}$ and ties are broken randomly.
```

Similarly to the one-against-all algorithm, for any distribution D on $X \times 2^{\{1, \dots, n\}}$ there is an induced distribution WOA-Train(D) defined on importance weighted samples $(X \times \{1, \dots, n\}) \times 0, 1 \times [0, \infty)$, where $(X \times \{1, \dots, n\})$ is the feature space augmented with the name of the call. To draw a sample from WOA-Train(D), we draw a multi-label sample (x, Y) from D, a random index $y \in \{1, \dots, n\}$, and output $(\langle x, y\rangle, I(y \in Y), w_{I(y \in Y)})$.

The following theorem gives an error transformation bound for this reduction.

Theorem 2. (WOA error efficiency) *Given any importance-weighted binary learner B and a set of examples S in $(X \times 2^{\{1,\ldots,n\}})^*$, let $h = \text{WOA-TRAIN}(B, S)$. For all test distribution D on $X \times 2^{\{1,\ldots,n\}}$,*

$$e_S(\text{WOA-TEST}(h), D) \leq e_W(h, \text{WOA-TRAIN}(D)).$$

Proof. There are two forms of errors, false positives and false negatives. First, we show that an adversarial binary classifier trying to induce multi-label errors with maximal efficiency has three possible strategies, then analyze these strategies.

First, notice that it is never more efficient for the adversary to invest into multiple false positives on a single example because the expected probability of a multi-label error grows sublinearly. For any number k of true positives, the probability of a multi-label error with two false positives is $\frac{2}{k+2}$, which is less than $\frac{2}{k+1}$, the probability of erring on two multi-label examples when investing one false positive in each. For $k = 0$, only one false positive is required to always err.

Now let l be the number of false negatives. If $l < k$ (note that l can be at most k), there must be one false positive; otherwise the error rate would be 0. For l false negatives and one false positive, we have an error rate of $\frac{1}{k-l+1}$ with the adversary paying importance $lw_1 + w_0$. To improve error efficiency, it is beneficial for the adversary to increase l if

$$\frac{\frac{1}{k-(l+1)+1}}{(l+1)w_1 + w_0} > \frac{\frac{1}{k-l+1}}{lw_1 + w_0},$$

or equivalently if $w_0 > w_1(k-2l)$. Otherwise, it is more efficient to decrease l. Thus an optimal adversary must choose either $l = 0$ or $l = k$.

For the $l = 0$ case, we have error efficiency $\frac{\frac{1}{k+1}}{w_0}$.

For the $l = k$ case, the adversary can have 0 or 1 false positives. These cases have multi-label error rates of $\frac{n-k}{n}$ and 1 with importance consumption of kw_1 or $kw_1 + w_0$, respectively.

Thus the adversary's most efficient strategy is given by

$$\max\left\{\frac{n-k}{knw_1}, \frac{1}{kw_1 + w_0}, \frac{1}{(k+1)w_0}\right\} = \frac{1}{n}.$$

Since the maximal error efficiency is $\frac{1}{n}$ and there are n classes, a binary importance weighted loss of ϵ implies a multi-label error rate of ϵ. ∎

Composition with Costing

The reduction above reduces to importance weighted binary classification. There are easy reductions from this problem to binary classification. For example, the Costing algorithm (Zadrozny, Langford, & Abe 2003) alters the underlying importance-weighted distribution D on $X \times \{0,1\} \times [0,\infty)$ using rejection sampling to produce a distribution $\text{COSTING}(D)$ on $X \times \{0,1\}$. The basic result is the following theorem.

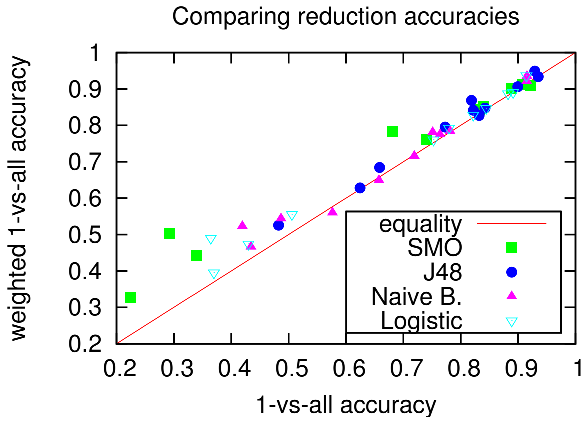

Figure 1: A figure showing comparative performance of the weighted-one-against-all reduction (+ costing) and the one-against-all reduction on several multiclass datasets with a Support Vector Machine (SMO), Naive Bayes, Decision tree (J48), and Logistic Regression classifier. Note that for these experiments we used one classifier per class rather than 1 classifier as in the theorem. Exact results are reported later.

Theorem 3. (Costing error efficiency) *Given any binary learner B and a set of examples S in $(X \times \{0,1\} \times [0,\infty))^*$, let $h = \text{COSTING}(B, S)$. For all test distributions D on $X \times 2^{\{1,\ldots,n\}}$,*

$$e_W(h, D) \leq e(h, \text{COSTING}(D))E_{(x,y,w)\sim D}[w].$$

When Weighted-One-Against-All is composed with this reduction we get the following corollary.

Theorem 4. *Given any binary learner B and a set of examples S in $(X \times 2^{\{1,\ldots,n\}})^*$, let $h = \text{WOA-TRAIN}(\text{COSTING}(B), S)$. Let $c(n,k) = \frac{n}{k+1}$ if $n \leq k^2 + k$, otherwise $c(n,k) = \frac{n+1}{k+1} - \frac{k}{n}$. For all test distribution D on $X \times 2^{\{1,\ldots,n\}}$,*

$$e_S(\text{WOA-TEST}(h), D)$$

$$\leq e(h, \text{COSTING}(\text{WOA-TRAIN}(D)))E_{(x,y)\sim D}[c(n,k)].$$

The theorem should be compared directly with the one-against-all reduction to binary classification which has error efficiency of $n-1$. Here, with $k = 1$ (i.e., a unique correct label), we get $c(n,1) = \frac{n+1}{2} - \frac{1}{n} = \frac{(n+2)(n-1)}{2n}$ implying error efficiency of about $\frac{n}{2}$, or about half the error rate of the original reduction.

Proof. The importance weighted loss is unnormalized since $\frac{k}{n}w_1 + \frac{n-k}{n}w_0 = \frac{n}{k+1}$ for $n \leq k^2 + k$ or $(\frac{n+1}{k+1} - \frac{k}{n})$ for $n > k^2 + k$. Taking an expectation over k according to D, we get the result. ∎

Experimental Results

Here we compare the performance of the One-Against-All (OAA) and the Weighted-One-Against-All (WOA) reductions on several multiclass datasets from the UCI Machine Learning Repository (Blake & Merz 1998). As binary classifier learners, we use four different learning algorithms available within Weka (Witten & Frank 2000): a decision tree learner (J48), a (linear) support vector machine learner (SMO), logistic regression and naive bayes. Note that for these experiments we use one binary classifier per class rather than one combined classifier as done in the analyses.

Four of the datasets we use have standard train/test splits (annealing, pendigits, satimage and soybean). For these datasets, we report the results on the test set. For the other datasets, we repeat the experiments on 20 random splits of the data (2/3 for training and 1/3 for testing) and report the average result on the test sets.

We use the default parameters in Weka, except with the Naive Bayes learner where we use the kernel estimation for modelling numeric attributes (option -K). We do not perform any kind of parameter optimization such as tuning the regularization parameters for support vector machines or the pruning parameters for decision trees. Our objective in performing these experiments is simply to compare the performance of the two reductions under the same conditions.

The test set error rates using each of the binary classifier learners are shown in Tables 1, 2, 3 and 4. From these results, we see that WOA generally results in better performance than OAA. Using SMO and logistic regression, WOA performs worse than OAA only for a single dataset (out of eleven). Using J48 and naive Bayes, WOA performs worse than OAA for two and three datasets (out of eleven), respectively.

We chose not to perform any kind of statistical significance test because the assumptions of independence and normality required by the usual tests are not satisfied here. We believe that the consistent results in favor of WOA across a range of different datasets and binary learners are enough to show that it is empirically superior to OAA.

Dataset	OAA	WOA
Annealing	0.1115	**0.0980**
Ecoli	0.3188	**0.2170**
Glass	0.7085	**0.4965**
Letter	0.6614	**0.5561**
Pendigits	0.1601	**0.1483**
Satimage	0.2593	**0.2391**
Solar flare	0.1665	**0.1576**
Soybean	**0.0793**	0.0907
Splice	0.0914	**0.0882**
Vowel	0.7753	**0.6742**
Yeast	0.8119	**0.5203**

Table 1: Test error rates using SMO as the binary learner.

Dataset	OAA	WOA
Annealing	0.0710	**0.0510**
Ecoli	0.2272	**0.2049**
Glass	0.3754	**0.3718**
Letter	0.1780	**0.1587**
Pendigits	0.1007	**0.0940**
Satimage	**0.1681**	0.1729
Solar flare	0.1576	**0.1528**
Soybean	0.1814	**0.1315**
Splice	**0.0650**	0.0660
Vowel	0.3415	**0.3158**
Yeast	0.5175	**0.4742**

Table 2: Test error rates using J48 as the binary learner.

Dataset	OAA	WOA
Annealing	0.0845	**0.0650**
Ecoli	0.2183	**0.2165**
Glass	0.5655	**0.5338**
Letter	**0.4235**	0.4399
Pendigits	0.2363	**0.2253**
Satimage	**0.2809**	0.2843
Solar flare	0.2489	**0.2192**
Soybean	**0.3426**	0.3508
Splice	0.0850	**0.0786**
Vowel	0.5132	**0.4562**
Yeast	0.5808	**0.4772**

Table 3: Test error rates using naive bayes as the binary learner.

Discussion

Summary We have shown that a simple modification of the common one-against-all reduction yields better performance in theory and in practice. The theory suggests an improvement by about a factor of 2 in the error rate, while the experimental results vary between negligible and significant improvement. Since the one-vs-all approach is a commonly used technique, this improvement is widely useful.

References

Allwein, E.; Schapire, R.; and Singer, Y. 2001. Reducing multiclass to binary: A unifying approach for

Dataset	OAA	WOA
Annealing	0.0855	**0.0635**
Ecoli	0.2205	**0.2071**
Glass	0.4944	**0.4444**
Letter	0.5700	**0.5257**
Pendigits	0.1555	**0.1527**
Satimage	0.2474	**0.2394**
Solar flare	0.1783	**0.1714**
Soybean	0.1174	**0.1129**
Splice	**0.1087**	0.1089
Vowel	0.6303	**0.6049**
Yeast	0.6358	**0.5096**

Table 4: Test error rates using logistic regression as the binary learner.

margin classifiers. *J. of Machine Learning Research* 1:113–141.

Beygelzimer, A.; Dani, V.; Hayes, T.; Langford, J.; and Zadrozny, B. 2004. Error limiting reductions between classification tasks. Technical Report TR04-077, Electronic Colloquium on Computational Complexity (ECCC).

Blake, C., and Merz, C. 1998. UCI repository of machine learning databases. `http://www.ics.uci.edu/~mlearn/MLRepository.html`.

Dietterich, T. G., and Bakiri, G. 1995. Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research* 2:263–286.

Guruswami, V., and Sahai, A. 1999. Multiclass learning, boosting, and error-correcting codes. In *Proceedings of the 12th Annual Conference on Computational Learning Theory (COLT)*, 145–155.

Rifkin, R., and Klautau, A. 2004. In defense of one-vs-all classification. *Journal of Machine Learning Research* 5:101–141.

Witten, I. H., and Frank, E. 2000. *Data Mining: Practical machine learning tools with Java implementations.* San Francisco: Morgan Kaufmann. `http://www.cs.waikato.ac.nz/ml/weka/`.

Zadrozny, B.; Langford, J.; and Abe, N. 2003. Cost sensitive learning by cost-proportionate example weighting. In *Proceedings of the 3rd ICDM Conference.*