Advanced Course in Machine Learning Spring 2010

Multiclass prediction

Handouts are jointly prepared by Shie Mannor and Shai Shalev-Shwartz

In this lecture we study the problem of multiclass prediction, in which we should learn a function $h : \mathcal{X} \to \mathcal{Y}$, where $\mathcal{X}$ is an instance space and $\mathcal{Y} = \{1, \ldots, k\} = [k]$ is the target space. We start with describing reduction techniques: assuming we have a learning algorithm for binary classification, we will show how to construct a learning algorithm for multiclass categorization. Next, we will shift our focus to specific hypothesis classes that are based on linear predictors. Finally, we will discuss cases in which the number of classes is very large but has some structure that leads to efficient learning.

1 Error Correcting Output Codes (ECOC)

The idea of ECOC is to associate each class $r \in [k]$ with a row of a "coding matrix" $M \in \{-1, 0, 1\}^{k \times l}$ for some integer l. For each $s \in [l]$, a binary learning algorithm is trained on an induced binary problem in which each multiclass example (x, r) is converted into a binary example (x, y) as follows: If $M_{r,s} \in \{\pm 1\}$ then $y = M_{r,s}$. Otherwise, if $M_{r,s} = 0$, then the example is ignored in the training process of the binary classifier associated with the s column of M. This creates binary classifiers $h_1, \ldots, h_l$. At test time, we calculate the word $(h_1(x), \ldots, h_m(x))$ and then predict the multiclass label as follows:

$$\hat{y} = \operatorname*{argmin}_r \sum_{s=1}^{l} |h_s(x) - M_{r,s}| \ .$$

Two well known examples are:

- **One vs. rest:** Setting $l = k$ and M is the matrix that has 1 on diagonal elements and -1 on off diagonal elements leads to the one vs. rest approach. That is, each binary classifier discriminates between one class to the rest of the classes.
- **All pairs:** Setting $l = \binom{k}{2}$ and M is the matrix such that for each pair $(r_1, r_2) \in [k]^2$ there exists a column $s \in [l]$ in M with zero everywhere except 1 on the $M_{r_1,s}$ element and -1 on the $M_{r_2,s}$ element. That is, each binary classifier discriminates between class r_1 and class r_2.

1.1 Analysis

Allwein, Schapire, and Singer analyzed the multiclass training error as a function of the average binary training error of the binary classifiers. In particular, they showed that the multiclass error is at most

$$\frac{1}{\rho} \sum_{s=1}^{l} \frac{1}{m} \sum_{i=1}^{m} (1 - \text{sign}(M_{y_i,s} f_s(x_i))) \ ,$$

where m is the number of examples and

$$\rho = \min_{r_1, r_2} \sum_{s=1}^{l} \frac{1 - M_{r_1,s} M_{r_2,s}}{2} \ ,$$

is the minimal "distance" between two codewords.

Note that for the one vs. rest approach, $\rho = 2$. It follows that even if the binary error of each binary predictor is ϵ, the multiclass error can be as large as $k\epsilon$.

In fact, Langford and Beygelzimer proved that for some distributions, even if the binary classification algorithm is guaranteed to be consistent, no matrix over $\{\pm 1\}^{k \times l}$ will lead to a consistent multiclass predictor.

In the all pairs approach, the matrix is over $\{-1, 0, 1\}$ so the above claim does not hold. However, it is possible to show that the multiclass error can be as large as $k\epsilon$. That is, the error of the binary predictor deteriorates linearly with the number of classes.

2 Error Correcting Tournaments

In this section we describe the filter tree method of Beygelzimer, Langford, Ravikumar (2009). This method enables to construct a multiclass predictor using a binary learning algorithm, where the regret of the multiclass predictor is guaranteed to be bounded by $\log(k)$ times the regret of the binary predictor. Note that the dependence on k is exponentially better than the dependence in the ECOC method. On the flip side, the analysis is based on a reduction to a no-regret binary learning algorithm w.r.t. the class of all functions from $\mathcal{X}$ to $\{\pm 1\}$. From no-free-lunch theorems, we know that such an algorithm does not exist unless we make serious distributional assumptions.

2.1 Regret Reduction

Let $\mathcal{D}$ be a distribution over $\mathcal{X} \times \mathcal{Y}$. The error of a multiclass predictor $h : \mathcal{X} \to \mathcal{Y}$ is $\mathrm{err}_{\mathcal{D}}(h) = \mathbb{P}[h(x) \neq y]$. The multiclass *regret* is defined as

$$\mathrm{reg}_{\mathcal{D}}(h) = \mathrm{err}_{\mathcal{D}}(h) - \min_{g:\mathcal{X} \to \mathcal{Y}} \mathrm{err}_{\mathcal{D}}(g) \ .$$

Clearly, the above definition subsumes binary classification as a special case ($k = 2$).

Suppose that we have a binary learning algorithm with a regret rate of ϵ. Can we construct a multiclass learning algorithm with a low regret?

2.2 First try: Divide and Conquer Tree

A straightforward attempt will be to split the labels in half, learn a binary classifier to distinguish between the two subsets, and repeat recursively until each subset contains one label. This tree reductions transforms the original distribution $\mathcal{D}$ into a distribution $\mathcal{D}_T$ (where T is the tree) over binary labeled examples by drawing $(x, y) \sim \mathcal{D}$ and a random non-leaf node i from the tree and constructing the binary example $((x, i), b)$ where $b = 1$ if y is in the left subtree of node i and $b = 0$ otherwise.

The following theorem shows that even if the binary learning algorithm is optimal (i.e. has no regret), the multiclass predictor can have non vanishing regret.

Theorem 1 *For all $k \geq 3$, for all binary trees over the labels, there exists a multiclass distribution $\mathcal{D}$ such that* $\mathrm{reg}_{\mathcal{D}}(T(f^\star)) > 0$ *for all $f^\star \in \mathrm{argmin}_f \, \mathrm{err}_{\mathcal{D}_T}(f)$.*

Proof W.l.o.g. we can assume $k = 3$ and the root node of the tree decides between $y \in \{1, 2\}$ or $y \in \{3\}$. Set $\mathcal{D}$ to be concentrated on a single instance and set the label to be 1 w.p. $1/4 + \epsilon$, 2 w.p. $1/4 + \epsilon$, and 3 w.p. $1/2 - 2\epsilon$. Clearly, the best multiclass predictor always predict the label 3 and has an error of $1/2 + 2/\epsilon$. In contrast, any optimal binary classifier f will prefer to decide $y \in \{1, 2\}$ over $y \in \{3\}$ at the root node. Thus, it'll err on all examples with label $y = 3$ and on half of the rest of examples. So, its multiclass error will be $1/2 - 2\epsilon + 1/4 + \epsilon = 3/4 - \epsilon$. It follows that the regret is $3/4 - \epsilon - (1/2 + 2\epsilon) = 1/4 - 3\epsilon$ which is non-negative for $\epsilon \in (0, 1/12)$. ■

2.3 The filter tree

The filter tree method works in round. For concreteness, suppose $k = 7$. In the first round, the labels are paired arbitrarily, e.g. "1 vs. 2", "3 vs. 4", "5 vs. 6", "7". A classifier is trained for each pair to predict which of the two labels is more likely. The winning labels from the first round are in turn paired in the second round, and a classifier is trained to predict whether the winner of one pair is more likely than the winner of the other. This process continues until we reach the root node.

The binary classifiers at the internal nodes are trained based on examples for which the multiclass label is one of the labels in the sub-tree associated with the node. E.g., the classifier "winner of 1 vs. 2 vs. winner of 3 vs. 4" is trained by examples for which $y \in \{1, 2, 3, 4\}$. If $y \in \{1, 2\}$ then the binary classifier gets the label -1 and if $y \in \{3, 4\}$ then the classifier gets the label 1. Other examples with $y \notin \{1, 2, 3, 4\}$ are rejected and are not used in the training process of this node.

At prediction time, the classifiers are applied from the root node to a leaf, and the label of the leaf is the multiclass prediction.

Beygelzimer et al proved that if the regret of the binary classification learning algorithm (against the class of all classifiers from $\mathcal{X}$ to $\{\pm 1\}$) is ϵ, then the multiclass regret is $\log(k)\epsilon$. As mentioned before, the assumption is quite strong and therefore we omit this result.

3 Linear Multiclass Predictors

In this section we describe a specific hypothesis class and learning algorithms for multiclass prediction, which is based on linear transformations.

3.1 The Multi-vector Construction

Suppose that $\mathcal{X} \subset \mathbb{R}^d$ and define the hypothesis class:

$$\{x \mapsto \operatorname*{argmax}_r (Wx)_r : W \in \mathbb{R}^{k \times d} \; \|W\| \le B\} .$$

For now, we intentionally did not specify which norm we refer to. The idea is as follows: Each row of W corresponds to one of the classes and x is projected on each of the rows. The row for which the value of the projection is largest is selected as the prediction.

To learn a matrix W from examples one can solve the ERM problem w.r.t. the above class. However, the optimization problem is non-convex. To overcome this obstacle, we can follow a technique similar to the use of Hinge loss in SVM and define the multiclass hinge loss as follows:

$$\ell(W, (x, y)) = \max_r \mathbb{1}_{[r \neq y]} - ((Wx)_y - (Wx)_r) .$$

It is easy to verify that $\ell(W, (x, y))$ is convex w.r.t. W and that

$$\mathbb{1}_{[y \neq \operatorname{argmax}_r (Wx)_r]} \le \ell(W, (x, y)) .$$

This, this loss function is a convex surrogate loss function for the multiclass problem.

Now, the problem of ERM w.r.t. the multiclass hinge loss becomes a convex optimization problem that can be solved efficiently.

Analysis It is possible to derive a generalization bound for learning the above class with the multiclass hinge-loss function. Choosing the norm of W to be the Frobenoius norm, the bound behaves like $BX/\sqrt{m}$ where m is the number of examples, and X is the maximal Euclidean norm of an instance $x \in \mathcal{X}$. It is also possible to derive different bounds assuming different norm constraints on W and on $x \in \mathcal{X}$.

3.2 The Single-vector Construction

The Multi-vector construction is a special case of a more general class of linear based multiclass predictors, which we define now. Let $\phi(x, y)$ be a feature mapping that takes $x \in \mathcal{X}$ and $y \in [k]$ and returns a vector in $\mathbb{R}^n$. Then, we define the hypothesis class:

$$\{x \mapsto \underset{r}{\operatorname{argmax}} \langle w, \phi(x, r)\rangle : w \in \mathbb{R}^n \; \|w\| \leq B\} \; .$$

It is an easy exercise to show that with an adequate definition of ϕ, this class becomes the multi-vector class described in the previous sub-section.

We can define the multiclass hinge-loss accordingly,

$$\ell(w, (x, y)) = \max_r \mathbb{1}_{[r \neq y]} - \langle w, \phi(x, y) - \phi(x, r)\rangle \; .$$

This loss function is again convex w.r.t. w and thus the ERM can be solved efficiently. It is also possible to derive a generalization bound which behaves like $BX/\sqrt{m}$ where now X is a bound on $\|\phi(x, y) - \phi(x, r)\|_\star$.

4 Structured Output

We refer the reader to Sections 8-9 in
`http://www.cs.huji.ac.il/~shais/papers/CrammerDeKeShSi06.pdf`.