Probabilistic Error Correcting Output Codes

John Langford

(and Alina Beygelzimer)

Outline

1. The Problem: Multiclass classification

2. A past approach: ECOC

3. A new approach: PECOC

4. Proof of optimality

Multiclass Classification

Given: $(X, \{1, ..., k\})^*$ = sequence of training examples

Find: $c_M : X \to \{1, ..., k\}$ = multiclass classifier approximately minimizing $\Pr_{x,y\sim D}(c_M(x) \neq y)$ = probability of error.

Note: D is unknown. (Key difference between learning and optimization)

Note: No assumption here that D generating training examples = D tested on.

ECOC: A Tempting Solution

Use Binary learning algorithms $A : (X, \{0,1\})^* \rightarrow (X \rightarrow \{0,1\})$ to solve Multiclass classification.

Label

Binary Problem	1	2	3	4
	1	1	0	0
	1	0	1	0
	1	0	0	1

Binary predictions

0	0	0
1	1	0
0	1	0
3	134	1234

Multiclass prediction

Binary training data $= \{(x, I(y_M \in \text{set}) : (x, y_M) \in \text{multiclass data}\}$

Multiclass prediction = closest label vector (ties $\Rightarrow$ randomize).

ECOC Analysis and Issues

Theorem: For all k, For all D on (x, y_M),

$$\text{Pr(ECOC errs)} \leq \text{4(average binary error rate)}$$

Proof: Exists code such that $< \frac{1}{4}$ binary errs $\Rightarrow$ no error

Caveats:

1. Arbitrary ECOC may make "impossible" binary problems.

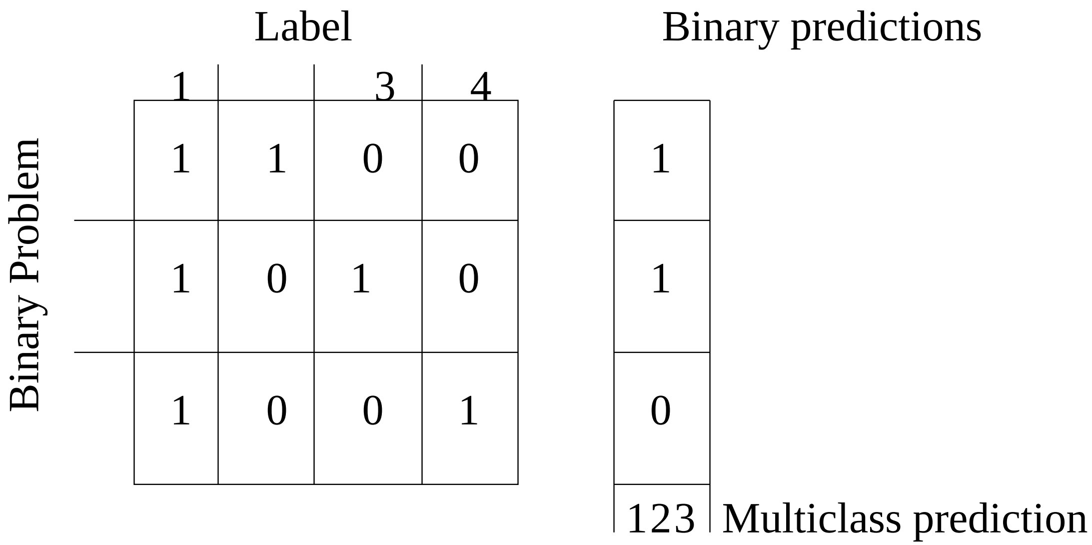
2. For arbitrary D binary error rate ≤ 0.25 might be impossible.

PECOC addresses 2 but not 1.

ECOC Inconsistency

Theorem: There exists a family of D s.t. ECOC is suboptimal even with perfect binary classifiers.

Suppose: $D(1|x) = 0.46$, $D(2|x) = 0.45$, $D(3|x) = 0.09$



Finite set of bits $\Rightarrow$ not enough bits to make optimal decision. PECOC fixes this.

PECOC, Defined

As ECOC, except binary classifiers make *probabilistic* predictions.

Still reduction to binary classification with Probing subroutine.

To predict: Use l_1 closest label/codeword.

	Label				Binary predictions
	1		3	4	
Binary Problem	1	1	0	0	0.91
	1	0	1	0	0.55
	1	0	0	1	0.46
					1 Multiclass prediction

PECOC Optimality

Theorem: For all k, For all D on (x, y_M),

$$\text{Pr(PECOC errs)} - \text{Pr(Optimal errs)}$$

$$\leq 2\sqrt{2(\text{average Binary error} - \text{minimum average Binary error})}$$

$$\equiv 2\sqrt{2(\text{relative Binary error})}$$

- Binary consistency $\Rightarrow$ Multiclass consistency

- Binary error rate $= 0.25$ OK whe minimum error rate $= 0.25$

Proof: when things are perfect

Proof: Pick code using all $\binom{k}{k/2}$ subsets.

For all labels l, without errors,

$$\sum_{b \in \text{Binary Problems}} \text{Probing}_b(\text{set containing } l|x)$$

$$= \sum_{b \in \text{Binary Problems}} \left[\sum_{l' \in \text{set containing l}} \Pr(l'|x) \right]$$

$$= |\{b\}|(\Pr(l|x) + \frac{1}{2} \sum_{l' \neq l} \Pr(l'|x)) = |\{b\}| \frac{\Pr(l|x) + 1}{2}$$

Proof: The real case

From Probing analysis,

$$E_{x,y\sim D}|D(1|x) - \text{Probing}(x)| \le \sqrt{2\text{relative Binary error}}$$

Most efficient method to disturb $\sum$ by $|\{b\}|\frac{\epsilon}{2}$ is for each call to probing to err by $|D(1|x) - \text{Probing}(x)| = \frac{\epsilon}{2}$.

Changing sum by $|\{b\}|\frac{\epsilon}{2} \Rightarrow$ choosing class with error rate at most 2ϵ worse than minimum.

Adding pieces together $\Rightarrow$ proof.

Bonus: Probabilistic Output

Instead of outputting

$$\arg\max_l \sum_{b \in \text{Binary Problems}} \text{Probing}_b(\text{set containing } l|x)$$

Output $\hat{p}(l|x) = \frac{2\sum_{b\in\text{Binary Problems}} \text{Probing}_b(\text{set containing } l|x)}{\{|b|\}} - 1$

Bonus Theorem: For all k, For all D on (x, y_M),

$$\forall l \ |\hat{p}(l|x) - D(l|x)| \leq \sqrt{2(\text{relative Binary error}}$$

Proof: Already done.

Remaining Issues

Experiments: How well does this work in practice?

Which code do you use in practice?

Shannon $\Rightarrow$ a random code of size $O(\log k)$ is near optimal for above theory.

... but a random code may make hard binary problems.