# Cost-sensitive Multiclass Classification Using One-versus-one Comparisons

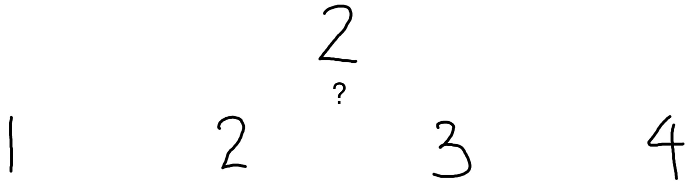
Hsuan-Tien Lin

Assistant Professor
Dept. of Computer Science and Information Engineering
National Taiwan University

Talk at ICISE2, 06/24/2012

*Based on the technical report "A Simple Cost-sensitive Multiclass Classification Algorithm Using One-versus-one Comparisons", Lin 2010.*

# Which Digit Did You Write?

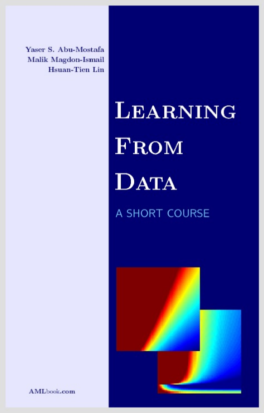


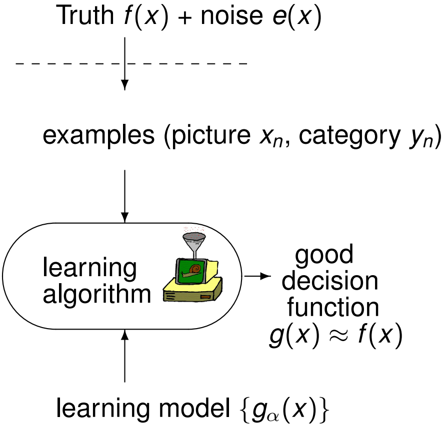

one (1) two (2) three (3) four (4)

- a **classification** problem
  —grouping “pictures” into different “categories”

**How can machines learn to classify?**

# Learning from Data (Abu-Mostafa, Magdon-Ismail and Lin, 2012)




challenge:

see only $\{(x_n, y_n)\}$ without knowing $f(x)$ or $e(x)$

$\overset{?}{\Longrightarrow}$ **generalize** to unseen $(x, y)$ w.r.t. $f(x)$

# Mis-prediction Costs ($g(x) \approx f(x)$?)

?

- ZIP code recognition:
  1: wrong; 2: right; 3: wrong; 4: wrong
- check value recognition:
  1: one-dollar mistake; 2: no mistake;
  3: one-dollar mistake; 4: **two**-dollar mistake
- evaluation by formation similarity:
  1: not very similar; 2: very similar;
  3: somewhat similar; 4: a **silly** prediction

**different applications evaluate mis-predictions differently**

# ZIP Code Recognition

?

1: wrong; 2: right; 3: wrong; 4: right

- **regular** classification problem: only right or wrong
- wrong cost: 1; right cost: 0
- prediction error of $g$ on some $(x, y)$:

$$\text{classification cost} = [\![y \neq g(x)]\!]$$

regular classification: **well-studied**, many good algorithms

# Check Value Recognition

?

1: one-dollar mistake; 2: no mistake;
3: one-dollar mistake; 4: **two**-dollar mistake

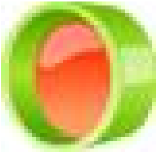
- **cost-sensitive** classification problem:
  different costs for different mis-predictions
- e.g. prediction error of $g$ on some $(x, y)$:

$$\text{absolute cost} = |y - g(x)|$$

cost-sensitive classification: **new**, need more research

# What is the Status of the Patient?

?



H1N1-infected

cold-infected

healthy

- another **classification** problem
  —grouping "patients" into different "status"

**Are all mis-prediction costs equal?**

# Patient Status Prediction

error measure = society cost

$\mathcal{C} =$

| actual \ predicted | H1N1 | cold | healthy |
|---|---|---|---|
| H1N1 | 0 | 1000 | **100000** |
| cold | 100 | 0 | 3000 |
| healthy | 100 | 30 | 0 |

- H1N1 mis-predicted as healthy: **very high cost**
- cold mis-predicted as healthy: high cost
- cold correctly predicted as cold: no cost

human doctors consider costs of decision;
**can computer-aided diagnosis do the same**?

# Cost Matrix $\mathcal{C}$

**regular classification**

$\mathcal{C}$ = classification cost $\mathcal{C}_c$:

$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

**cost-sensitive classification**

$\mathcal{C}$ = anything other than $\mathcal{C}_c$:

$$\begin{pmatrix} 0 & 1 & 4 & 5 \\ 1 & 0 & 1 & 3 \\ 3 & 1 & 0 & 2 \\ 5 & 4 & 1 & 0 \end{pmatrix}$$

regular classification:
**special case** of cost-sensitive classification

# Cost-sensitive Classification Setup

## Given

$N$ examples, each (input $x_n$, label $y_n$) $\in \mathcal{X} \times \{1, 2, \ldots, K\} \times R^K$; cost matrix $\mathcal{C}$

- $K = 2$: binary; $K > 2$: **multiclass**
- will assume $\mathcal{C}(y, y) = \min_{1 \le k \le K} \mathcal{C}(y, k)$

## Goal

a classifier $g(x)$ that pays a small cost $\mathcal{C}(y, g(x))$ on future **unseen** example $(x, y)$

cost-sensitive classification:
**more realistic than regular one**

# Our Contribution

| | binary | multiclass |
|---|---|---|
| regular | well-studied | well-studied |
| cost-sensitive | known (Zadrozny, 2003) | ongoing (our work, among others) |

*a theoretical and algorithmic study of cost-sensitive classification, which ...*

- introduces a methodology for extending regular classification algorithms to cost-sensitive ones with **any cost**
- provides **strong theoretical support** for the methodology
- leads to some promising algorithms with **superior experimental results**

will describe the methodology
and a concrete algorithm

# Central Idea: Reduction

(iPod)

complex cost-sensitive problems

(adapter)


(reduction)

(cassette player)

simpler regular classification problems with well-known results on models, algorithms, and theories

**If I have seen further it is by standing on the shoulders of Giants—I. Newton**

# Cost-Sensitive Binary Classification (1/2)

medical profile $x$

?

medical profile $x_1$
H1N1 (1)

medical profile $x_2$
NOH1N1 (2)

- predicting H1N1 as NOH1N1: serious consequences to public health
- predicting NOH1N1 as H1N1: not good, but less serious
- cost-sensitive $\mathcal{C}$: $\begin{pmatrix} 0 & 1000 \\ 1 & 0 \end{pmatrix}$
- regular $\mathcal{C}_c$: $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$

**how to change the entry from 1 to 1000?**

# Cost-Sensitive Binary Classification (2/2)

**copy each case labeled H1N1** 1000 **times**

**original problem**

evaluate w/ $\begin{pmatrix} 0 & 1000 \\ 1 & 0 \end{pmatrix}$

$(x_1, \text{H1N1})$
$(x_2, \text{NOH1N1})$
$(x_3, \text{NOH1N1})$
$(x_4, \text{NOH1N1})$
$(x_5, \text{H1N1})$

**equivalent problem**

evaluate w/ $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$

$(x_1, \text{H1N1}), \cdots, (x_1, \text{H1N1})$
$(x_2, \text{NOH1N1})$
$(x_3, \text{NOH1N1})$
$(x_4, \text{NOH1N1})$
$(x_5, \text{H1N1}), \cdots, (x_5, \text{H1N1})$

mathematically:

$$\begin{pmatrix} 0 & 1000 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1000 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

# Key Idea: Cost Transformation

$$\underbrace{\begin{pmatrix} 0 & 1000 \\ 1 & 0 \end{pmatrix}}_{\mathcal{C}} = \underbrace{\begin{pmatrix} 1000 & 0 \\ 0 & 1 \end{pmatrix}}_{\#\text{ of copies}} \cdot \underbrace{\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}}_{\mathcal{C}_c}$$

$$\underbrace{\begin{pmatrix} 0 & 1 & 1 & 1 \\ 3 & 2 & 3 & 4 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}}_{\mathcal{C}} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\text{mixture weights } Q} \cdot \underbrace{\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}}_{\mathcal{C}_c, \text{invertible}}$$

- **split** the cost-sensitive example:

$$(x, 2)$$

$\Longrightarrow$ a mixture of regular examples $\{(x,1),(x,2),(x,2),(x,3)\}$

or a weighted mixture $\{(x,1,1),(x,2,2),(x,3,1)\}$

**why split?**

# Cost Equivalence by Splitting

$$\underbrace{\begin{pmatrix} 0 & 1 & 1 & 1 \\ 3 & 2 & 3 & 4 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}}_{\mathcal{C}} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\text{mixture weights } Q} \cdot \underbrace{\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}}_{\mathcal{C}_c}$$

- $(x, 2)$
  $\Longrightarrow$ a weighted mixture $\{(x, 1, 1), (x, 2, 2), (x, 3, 1)\}$
- **cost equivalence**: for any classifier $g$,

$$\mathcal{C}(y, g(x)) = \sum_{\ell=1}^{K} Q(y, \ell)\mathcal{C}_c(\ell, g(x))$$

$$\begin{array}{lll} & \min_g \text{ expected LHS} & \text{(cost-sensitive)} \\ = & \min_g \text{ expected RHS} & \text{(regular when } Q(y, \ell) \geq 0) \end{array}$$

# Cost Transformation Methodology: Preliminary

1. split each training example $(x_n, y_n)$ to a weighted mixture $\left\{(x_n, \ell, Q(y_n, \ell))\right\}_{\ell=1}^{K}$
2. apply regular classification algorithm on the weighted mixtures $\bigcup_{n=1}^{N} \left\{(x_n, \ell, Q(y_n, \ell))\right\}_{\ell=1}^{K}$

- by cost equivalence,

$$\begin{aligned} & \text{good } g \text{ for new regular classification problem} \\ = \quad & \text{good } g \text{ for original cost-sensitive classification problem} \end{aligned}$$

- regular classification: needs $Q(y_n, \ell) \geq 0$

**but what if $Q(y_n, \ell)$ negative?**

# Similar Cost Vectors

$$\underbrace{\begin{pmatrix} 1 & 0 & 1 & 2 \\ 3 & 2 & 3 & 4 \end{pmatrix}}_{\text{costs}} = \underbrace{\begin{pmatrix} 1/3 & 4/3 & 1/3 & -2/3 \\ 1 & 2 & 1 & 0 \end{pmatrix}}_{\text{mixture weights } Q(y,\ell)} \cdot \underbrace{\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}}_{\text{classification costs}}$$

- negative $Q(y, \ell)$: cannot split
- but $\hat{\mathbf{c}} = (1, 0, 1, 2)$ is **similar** to $\mathbf{c} = (3, 2, 3, 4)$: for any classifier $g$,

$$\hat{\mathbf{c}}[g(x)] + \text{constant} = \mathbf{c}[g(x)]$$

- constant can be dropped during minimization

> shifting cost matrix by constant rows does not affect minimization

# Cost Transformation Methodology: Revised

$$\underbrace{\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}}_{\mathcal{C}} + \underbrace{\begin{pmatrix} 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}}_{\text{constant rows}} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\text{mixture weights } Q} \cdot \underbrace{\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}}_{\mathcal{C}_c}$$

1. shift each row of original cost to a similar and "splittable" $\mathcal{C}(y, :)$, i.e., with $Q(y_n, \ell) \geq 0$
2. split $(x_n, y_n)$ to weighted mixture $\left\{(x_n, \ell, Q(y_n, \ell))\right\}_{\ell=1}^{K}$
3. apply regular classification algorithm on the weighted mixtures $\bigcup_{n=1}^{N} \left\{(x_n, \ell, Q(y_n, \ell))\right\}_{\ell=1}^{K}$

good $g$ for new regular classification problem
$=$ good $g$ for cost-sensitive classification problem

# Uncertainty in Mixture

- a single example $\{(x, 2)\}$
  —**certain** that the desired label is 2
- a mixture $\{(x, 1, 1), (x, 2, 2), (x, 3, 1)\}$ sharing the same $x$
  —**uncertainty** in the desired label (25%: 1, 50%: 2, 25%: 3)
- over-shifting adds unnecessary mixture uncertainty:

$$\underbrace{\begin{pmatrix} 3 & 2 & 3 & 4 \\ 33 & 32 & 33 & 34 \end{pmatrix}}_{\text{costs}} = \underbrace{\begin{pmatrix} 1 & 2 & 1 & 0 \\ 11 & 12 & 11 & 10 \end{pmatrix}}_{\text{mixture weights}} \cdot \underbrace{\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}}_{\mathcal{C}_c}$$

> should choose a similar and splittable **c**
> with **minimum mixture uncertainty**

# Cost Transformation Methodology: Final

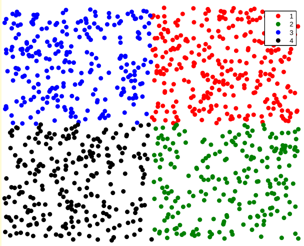
1. shift original cost to a similar and splittable $\mathcal{C}$ with minimum "mixture uncertainty"
2. split $(x_n, y_n)$ to a weighted mixture $\left\{(x_n, \ell, Q(y_n, \ell)\right\}_{\ell=1}^{K}$ with $\mathcal{C}$
3. apply regular classification algorithm on the weighted mixtures $\bigcup_{n=1}^{N} \left\{(x_n, \ell, Q(y_n, \ell))\right\}_{\ell=1}^{K}$

- mixture uncertainty: entropy of each normalized $Q(y, :)$
- a simple and unique optimal shifting exists for every $\mathcal{C}$ —$Q(y, k) = \max_\ell \mathcal{C}(y, \ell) - \mathcal{C}(y, k)$

$$\begin{aligned} & \text{good } g \text{ for new regular classification problem} \\ = \; & \text{good } g \text{ for cost-sensitive classification problem} \end{aligned}$$

# Unavoidable (Minimum) Uncertainty

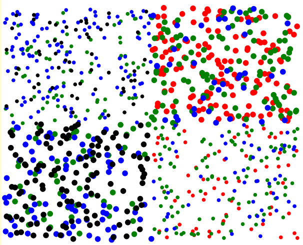
**Original Cost-Sensitive Classification Problem**




individual examples with certainty

+ absolute cost =

**New Regular Classification Problem**



mixtures with unavoidable uncertainty

- new problem usually **harder** than original one

**need robust regular classification algorithm to deal with uncertainty**

# From OVO to CSOVO

**One-Versus-One: A Popular Classification Meta-Method**

1. for a pair $(i, j)$, take all examples $(x_n, y_n)$ that $y_n = i$ or $j$
2. train a binary classifier $g^{(i,j)}$ using those examples
3. repeat the previous two steps for all different $(i, j)$
4. predict using the votes from $g^{(i,j)}$

$$\begin{array}{rl} & \text{cost-sensitive multiclass classification} \\ \overset{\text{cost transformation}}{\Longrightarrow} & \text{regular (weighted) multiclass classification} \\ \overset{\text{OVO decomposition}}{\Longrightarrow} & \text{regular (weighted) binary classification} \end{array}$$

**cost-sensitive one-versus-one:**
**cost transformation + one-versus-one**

# Cost-Sensitive One-Versus-One (CSOVO)

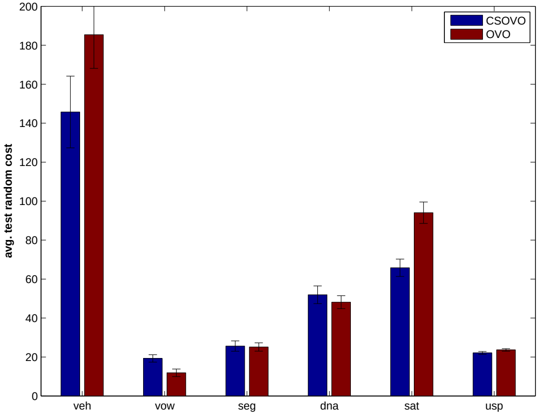
1. for a pair $(i, j)$, transform all examples $(x_n, y_n)$ to $\left(x_n, \operatorname*{argmin}_{k \in \{i,j\}} \mathcal{C}(y_n, k)\right)$ with weight $\left|\mathcal{C}(y_n, i) - \mathcal{C}(y_n, j)\right|$
2. train a binary classifier $g^{(i,j)}$ using those examples
3. repeat the previous two steps for all different $(i, j)$
4. predict using the votes from $g^{(i,j)}$

- comes with **good theoretical guarantee**:

$$\text{test cost of final classifier} \leq 2 \sum_{i<j} \text{test cost of } g^{(i,j)}$$

**simple**, **efficient**, and
takes original OVO as **special case**

# CSOVO v.s. OVO




- OVO: popular regular classification meta-method, NOT cost-sensitive
- couple both meta-methods with SVM

**CSOVO often better suited for cost-sensitive classification**

# Conclusion

- **cost transformation** methodology:
  makes **any** (robust) regular classification algorithm cost-sensitive
- theoretical guarantee: **cost equivalence**
- algorithmic use: a **novel and simple** algorithm CSOVO
- experimental performance of CSOVO: **superior**

**Thank you for your attention!**