# Ordinal Regression by Extended Binary Classification

**Ling Li** and **Hsuan-Tien Lin**

Learning Systems Group, California Institute of Technology

## Ordinal Regression

In an ordinal regression (ranking) problem, there is a total order on the labels (ranks).

1: infant | 2: child | 3: teenager | 4: adult | ?

Ordinal regression is between multiclass classification and metric regression:

- Ranks do carry ordering information: child is younger than adult.
- Ranks don't carry numerical information: child is not necessarily half as young as adult.

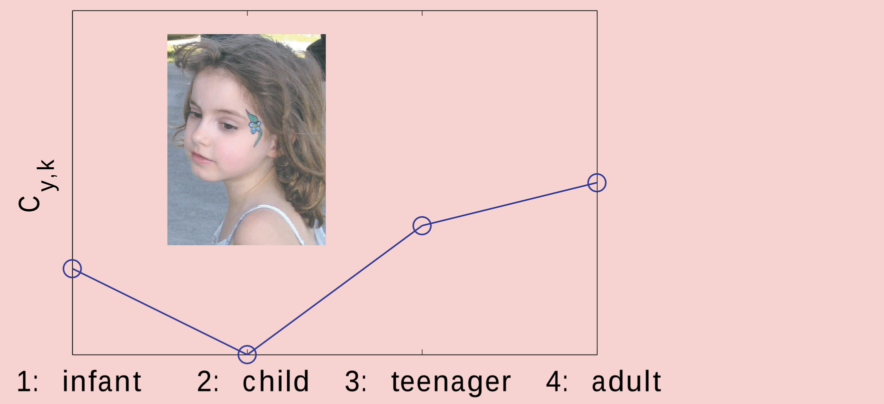
> Ordinal regression problem: Given a training set $\{(\mathbf{x}_n, y_n)\}$ of $N$ examples, find a ranking rule $r(\mathbf{x})$ that predicts the rank $y$ of unseen input $\mathbf{x}$ "well."

## Mislabeling Cost

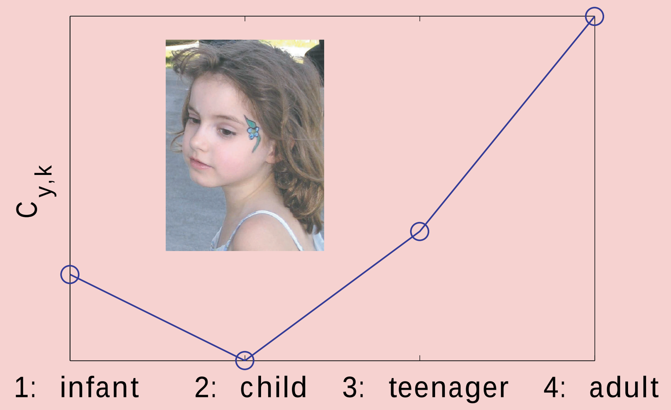
Predicting well: low expected mislabeling cost on all inputs $\mathbf{x}$ when using $r(\mathbf{x})$.

- We cannot compare rank 4 with rank 2 numerically, but we can artificially assign a cost when rank 2 is mislabeled as rank 4.
- Every kind of mislabeling $y \to k$ is assigned with a positive cost $\mathcal{C}_{y,k}$, e.g., $\mathcal{C}_{2,4}$: a child photo labeled as adult.
- Ordering information shall be encoded to make the costs different from those in multiclass.

Reasonable ordinal regression costs $\mathcal{C}_{y,k}$ for a given $y$:




V-shaped: pay more when predicting farther away




Convex: pay increasingly more when predicting farther away

The costs can be organized in a matrix.

$$\begin{pmatrix}0&1&1&1\\1&0&1&1\\1&1&0&1\\1&1&1&0\end{pmatrix} \qquad \begin{pmatrix}0&1&2&3\\1&0&1&2\\2&1&0&1\\3&2&1&0\end{pmatrix} \qquad \begin{pmatrix}0&1&4&9\\1&0&1&4\\4&1&0&1\\9&4&1&0\end{pmatrix}$$

classification cost | absolute cost | squared cost

## Reduction

- Designing new algorithms for ordinal regression takes much effort.
- Researchers usually borrow ideas from binary classification algorithms.

A general framework to systematically reduce ordinal regression to binary classification is very useful.

## Ranking Through Associated Binary Problem

The total order allows us to compare an example to a rank class:

$$f_b(\mathbf{x}, k) = [\![ f(\mathbf{x}, k) > 0 ]\!]\text{: Is the rank of } \mathbf{x} \text{ greater than } k?$$

$\mathbf{x} =$ 

| $k$ | 1: infant | 2: child | 3: teenager |
|---|---|---|---|
| consistent answers | Yes (1) | No (0) | No (0) |
| inconsistent answers | Yes (1) | No (0) | Yes (1) |

Consistent answers lead to a ranking rule that finds the first No,

$$r(\mathbf{x}) = \min\{k\colon f_b(\mathbf{x}, k) = 0\} = 1 + \sum_{k=1}^{K-1} f_b(\mathbf{x}, k).$$

This construction rule can also be used for inconsistent answers.

## Extended Examples

- Extended examples $(\mathbf{x}^{(k)}, y^{(k)})$ with weights $w_{y,k}$ :

$$\mathbf{x}^{(k)} = (\mathbf{x}, k), \quad y^{(k)} = 2[\![k < y]\!] - 1, \quad w_{y,k} = |\mathcal{C}_{y,k} - \mathcal{C}_{y,k+1}|.$$

- The binary label $y^{(k)}$ reflects the desired consistent answer for the associated binary problem.
- The weight $w_{y,k}$ is the additional cost that the binary classifier $f_b$ pays for wrong prediction on $(\mathbf{x}^{(k)})$.
- If $f_b$ gives consistent answers, or $\mathcal{C}$ contains convex rows, for any $(\mathbf{x}, y)$ and its extended examples $(\mathbf{x}^{(k)}, y^{(k)})$

$$\mathcal{C}_{y,r(\mathbf{x})} \le \sum_{k=1}^{K-1} w_{y,k} [\![ y^{(k)} f(\mathbf{x}^{(k)}) \le 0 ]\!].$$

> The reduction framework:
>
> 1. Transform training examples $(\mathbf{x}_n, y_n)$ to extended training examples $(\mathbf{x}_n^{(k)}, y_n^{(k)})$ with weights $w_{y_n,k}$.
> 2. Use a binary classification algorithm to learn $f(\mathbf{x}^{(k)})$ using the weighted extended training examples.
> 3. Construct a ranking rule $r(\mathbf{x})$ from $f(\mathbf{x}^{(k)})$ for prediction.

## Generalization Bounds

- $(\mathbf{X}, Y) = (\mathbf{x}^{(k)}, y^{(k)})$ can be thought as outcomes of $(\mathbf{x}, y) \sim P$ and $k \sim \Pr(k \mid y) \propto w_{y,k}$.
- Performing well in binary classification implies performing well in ordinal regression.

| bound | binary classification | ordinal regression |
|---|---|---|
| generalization error | $\mathbb{E}_{(\mathbf{X},Y)} [\![ Y f(\mathbf{X}) \le 0 ]\!]$ is small. | $\mathbb{E}_{(\mathbf{x},y)}\, \mathcal{C}_{y,r(\mathbf{x})}$ is small. |
| data-dependent large-margin bound | (Bartlett98) $f(\mathbf{X}) = \langle \mathbf{u}, \phi(\mathbf{X}) \rangle$ with bounded $\mathbf{X}$ and normalized $\mathbf{u}$: $\mathbb{E}_{(\mathbf{X},Y)} [\![ Y f(\mathbf{X}) \le 0 ]\!] \le \frac{1}{N}\sum_{n=1}^{N} [\![ Y_n f(\mathbf{X}_n) \le \Delta ]\!] + O\left(\frac{\log N}{\sqrt{N}}, \frac{1}{\Delta}, \sqrt{\log\frac{1}{\delta}}\right)$ | $f(\mathbf{x}^{(k)}) = \langle (\mathbf{u}, -\boldsymbol{\theta}), (\phi(\mathbf{x}), \mathbf{e}_k) \rangle$ with bounded $\mathbf{x}^{(k)}$ and normalized $(\mathbf{u}, -\boldsymbol{\theta})$: $\mathbb{E}_{(\mathbf{x},y)}\, \mathcal{C}_{y,r(\mathbf{x})} \le \frac{\beta}{N} \cdot \sum_{n=1}^{N}\sum_{k=1}^{K-1} w_n^{(k)} [\![ y_n^{(k)} f(\mathbf{x}_n^{(k)}) \le \Delta ]\!] + O\left(\frac{\log N}{\sqrt{N}}, \frac{1}{\Delta}, \sqrt{\log\frac{1}{\delta}}\right)$ |

## Algorithms

Ordinal regression algorithm $\Longleftarrow$ reduction + cost matrix + encoding of $\mathbf{x}^{(k)}$ + binary classification algorithm

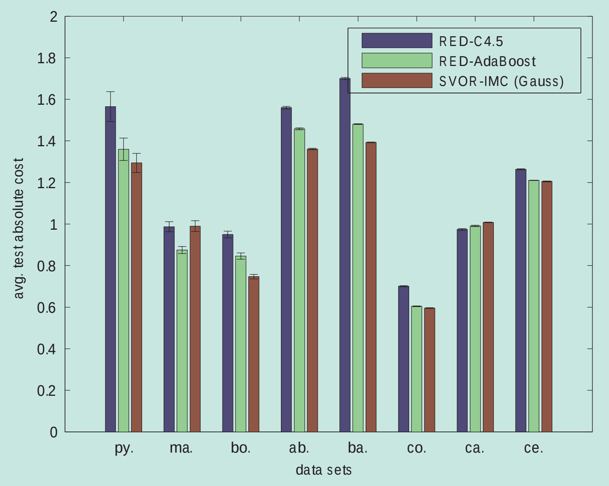
$$\mathbf{e}_k = (\overbrace{0, \cdots, 0, 1}^{k-1}, 0, \cdots, 0)$$

| ordinal regression alg. | cost | $\mathbf{x}^{(k)}$ | binary classification algorithm |
|---|---|---|---|
| thresholded ranking | any convex one | $(\mathbf{x}, \mathbf{e}_k)$ | any algorithm for $f(\mathbf{x}^{(k)}) = g(\mathbf{x}) - \langle \boldsymbol{\theta}, \mathbf{e}_k \rangle$ |
| perceptron ranking (Crammer02) | absolute | $(\mathbf{x}, \mathbf{e}_k)$ | modified perceptron learning rule for $f(\mathbf{x}^{(k)}) = \langle (\mathbf{u}, -\boldsymbol{\theta}), \mathbf{x}^{(k)} \rangle$ |
| kernel-based ranking (Rajaram03) | classification | $(\mathbf{x}, \sum_{i=1}^{k} \mathbf{e}_i)$ | modified hard-margin SVM for $f(\mathbf{x}^{(k)}) = \langle (\mathbf{u}, -\boldsymbol{\theta}), (\phi(\mathbf{x}), \mathbf{e}_k) \rangle$ |
| SVOR-EXP (Chu05) | classification | $(\mathbf{x}, \mathbf{e}_k)$ | modified soft-margin SVM with ordered $\boldsymbol{\theta}$ for $f(\mathbf{x}^{(k)}) = \langle (\mathbf{u}, -\boldsymbol{\theta}), (\phi(\mathbf{x}), \mathbf{e}_k) \rangle$ |
| SVOR-IMC (Chu05) | absolute | $(\mathbf{x}, \mathbf{e}_k)$ | modified soft-margin SVM for $f(\mathbf{x}^{(k)}) = \langle (\mathbf{u}, -\boldsymbol{\theta}), (\phi(\mathbf{x}), \mathbf{e}_k) \rangle$ |
| Reduction-SVM | absolute | $(\mathbf{x}, \mathbf{e}_k)$ | standard soft-margin SVM for $f(\mathbf{x}^{(k)}) = \langle (\mathbf{u}, -\boldsymbol{\theta}), (\phi(\mathbf{x}), \gamma \cdot \mathbf{e}_k) \rangle$ |
| Reduction-C4.5 | absolute | $(\mathbf{x}, \mathbf{e}_k)$ | standard C4.5 for decision trees |
| Reduction-AdaBoost | absolute | $(\mathbf{x}, \mathbf{e}_k)$ | standard AdaBoost for decision stump ensembles |

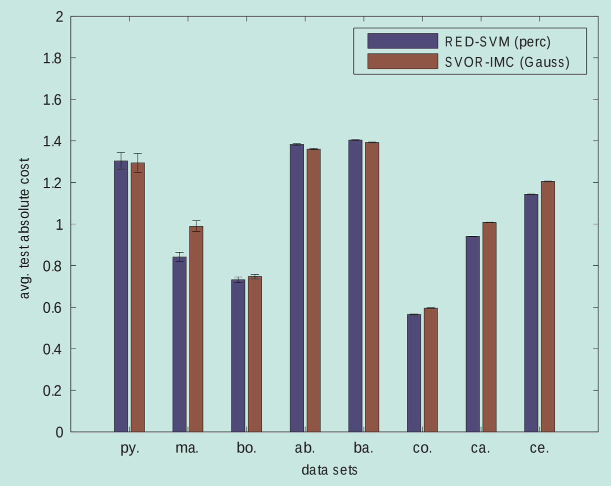
Our framework simplifies the analysis and the tuning of ordinal regression algorithms:

- Mistake bound for perceptron ranking is an easy extension of perceptron mistake bound.
- Improvements in binary classifier (e.g., faster optimization procedure for SVM) can be immediately inherited.

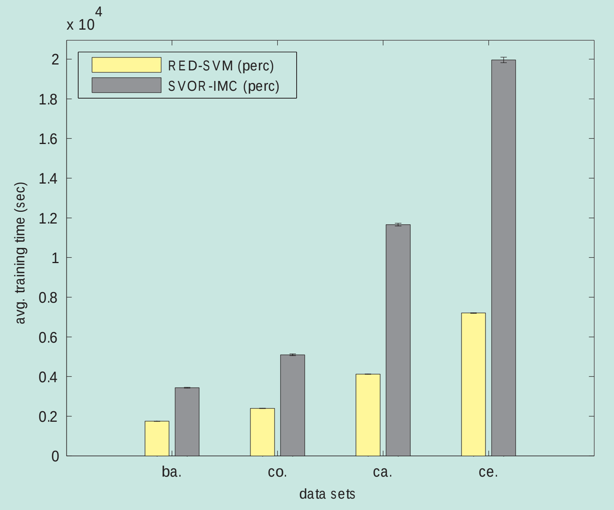
## Experimental Results




Simple reduction to popular and standard algorithms beats state-of-the-art SVOR-IMC some time.




Reduction to standard SVM with perceptron kernel is often significantly better than SVOR-IMC.




Reduction to standard SVM is much faster than reduction to modified SVM (SVOR-IMC).

## Summary

With our reduction framework from ordinal regression to binary classification:

- New generalization bounds for ordinal regression can be easily derived from known bounds for binary classification, which saves tremendous efforts in theoretical analysis.
- Well-tuned binary classification approaches can be readily transformed into good ordinal regression algorithms, which saves immense efforts in design and implementation.