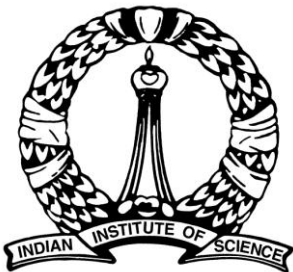


# Support Vector Algorithms for Optimizing the Partial AUC

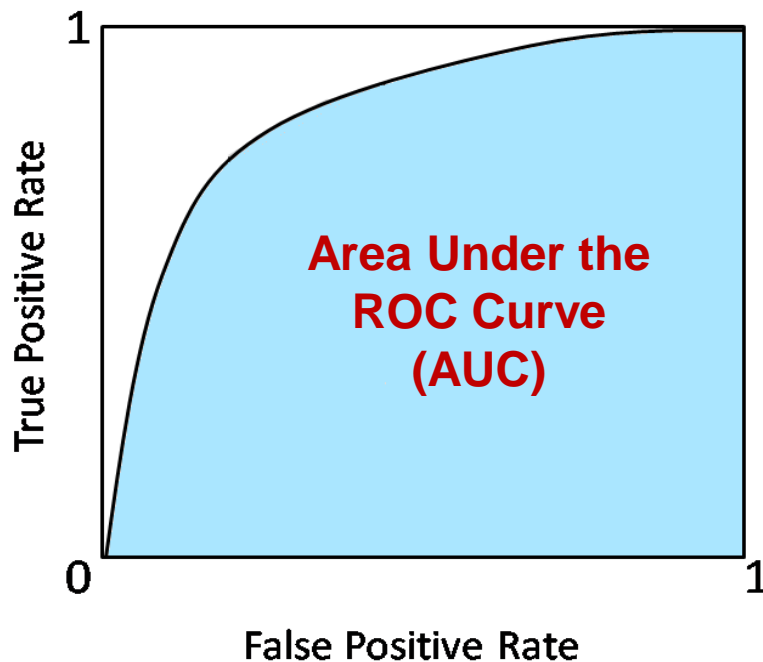
**Harikrishna Narasimhan** and Shivani Agarwal



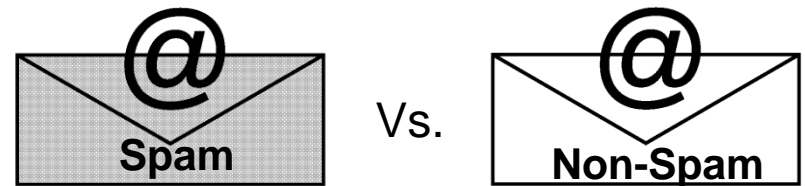
Department of Computer Science and Automation  
Indian Institute of Science, Bangalore

Based on work in ICML 2013 and KDD 2013

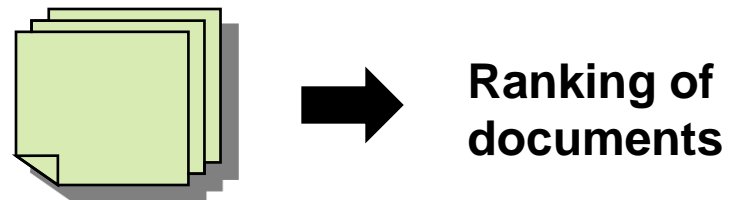
# Receiver Operating Characteristic Curve



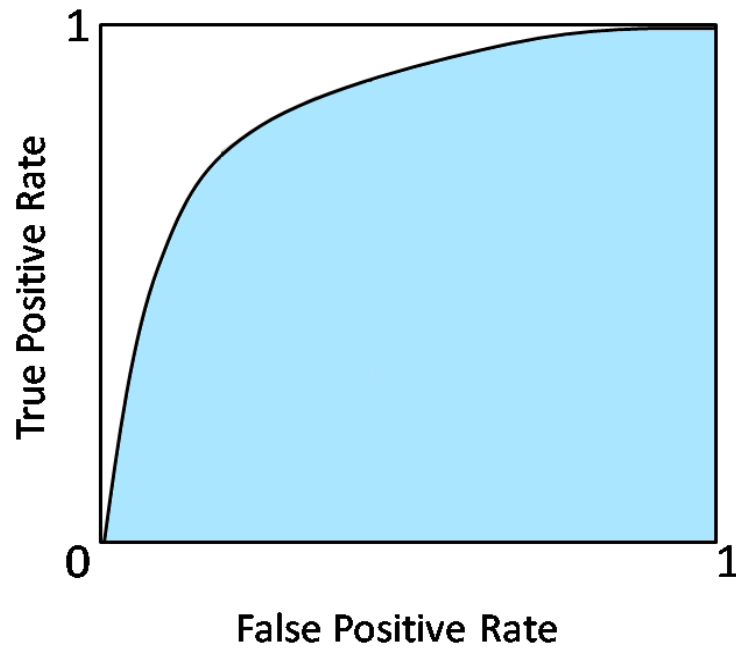
## Binary Classification



## Bipartite Ranking

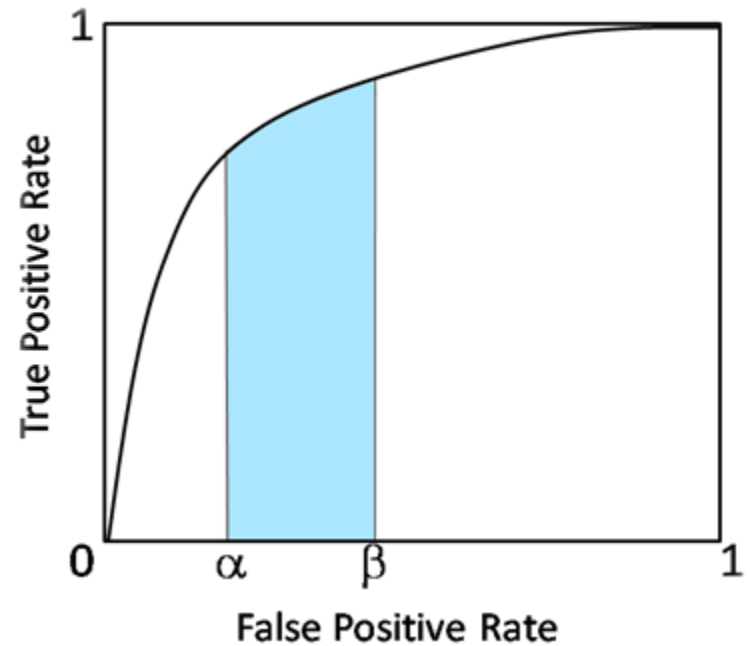


# Partial AUC?



**Full AUC**

**Vs**



**Partial AUC**

# Ranking



learning to rank

Search

About 216,000,000 results (0.23 seconds)

Web

Images

Maps

Videos

News

More

Bangalore, Karnataka

Change location

The web

Pages from India

More search tools

[Learning to rank - Wikipedia, the free encyclopedia](#)  
[en.wikipedia.org/wiki/Learning\\_to\\_rank](http://en.wikipedia.org/wiki/Learning_to_rank)

**Learning to rank** or machine-learned ranking (MLR) is a type of supervised or semi-supervised machine learning problem in which the goal is to automatically ...

[Applications](#) - [Feature vectors](#) - [Evaluation measures](#) - [Approaches](#)

[Yahoo! Learning to Rank Challenge](#)

[learningtorankchallenge.yahoo.com/](http://learningtorankchallenge.yahoo.com/) - United States

**Learning to Rank** Challenge is closed! Close competition, innovative ideas, and fierce determination were some of the highlights of the first ever Yahoo!

[PDF] [Learning to Rank for Information Retrieval This Tutorial](#)

[www2009.org/.../T7A-LEARNING%20TO%20RANK%20TUTORIA...](http://www2009.org/.../T7A-LEARNING%20TO%20RANK%20TUTORIA...)

File Format: PDF/Adobe Acrobat - [Quick View](#)

12 Apr 2009 - **Learning to Rank** for Information Retrieval. Tie-Yan Liu. Microsoft Research Asia. A tutorial at WWW 2009. This Tutorial. • **Learning to rank** for ...

[LETOR: A Benchmark Collection for Research on Learning to Rank ...](#)

[research.microsoft.com/~letor/](http://research.microsoft.com/~letor/)

This website is designed to facilitate research in **Learning TO Rank** (LETOR). Much information about **learning to rank** can be found in the website, including ...

[PDF] [Large Scale Learning to Rank](#)

[www.eecs.tufts.edu/~dsculley/papers/large-scale-rank.pdf](http://www.eecs.tufts.edu/~dsculley/papers/large-scale-rank.pdf)

File Format: PDF/Adobe Acrobat - [Quick View](#)

by D Sculley - [Cited by 19](#) - [Related articles](#)

Pairwise **learning to rank** methods such as RankSVM give good performance, ... In this paper, we are concerned with **learning to rank** methods that can learn on ...

[PDF] [Metric Learning to Rank](#)

[www.icml2010.org/papers/504.pdf](http://www.icml2010.org/papers/504.pdf)

File Format: PDF/Adobe Acrobat - [Quick View](#)

by B McFee - [Cited by 21](#) - [Related articles](#)

**Metric Learning to Rank**. Brian McFee [bmcfee@cs.ucsd.edu](mailto:bmcfee@cs.ucsd.edu). Department of Computer Science and Engineering, University of California, San Diego, CA 92093 ...

[PDF] [Yahoo! Learning to Rank Challenge Overview](#)

[jmlr.csail.mit.edu/proceedings/papers/v14/.../chapelle11a.pdf](http://jmlr.csail.mit.edu/proceedings/papers/v14/.../chapelle11a.pdf)

File Format: PDF/Adobe Acrobat - [Quick View](#)

by O Chapelle - [Cited by 23](#) - [Related articles](#)

**Learning to rank** for information retrieval has gained a lot of interest in the ... field in which machine learning algorithms are used to learn this ranking function.

[PDF] [Future directions in learning to rank](#)

[jmlr.csail.mit.edu/proceedings/papers/v14/.../chapelle11b.pdf](http://jmlr.csail.mit.edu/proceedings/papers/v14/.../chapelle11b.pdf)

File Format: PDF/Adobe Acrobat - [Quick View](#)

True Positive Rate

1

0

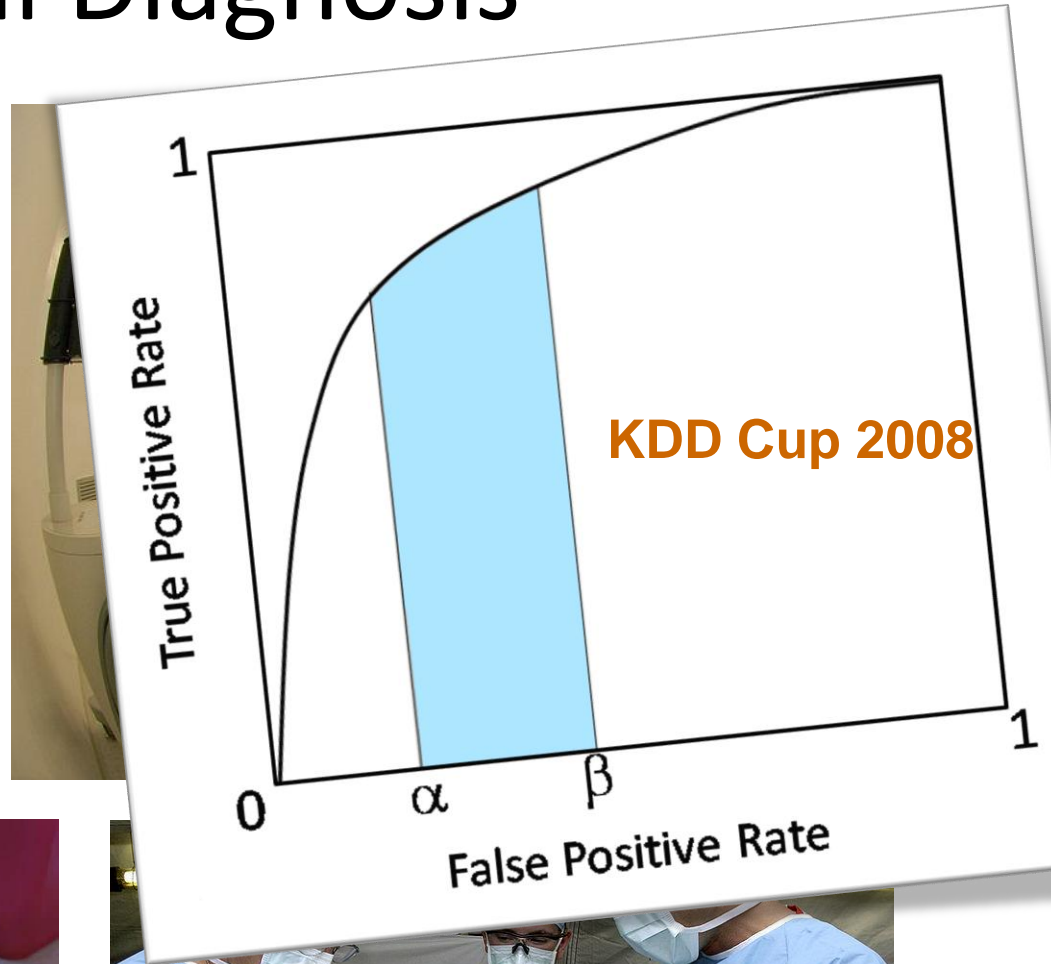
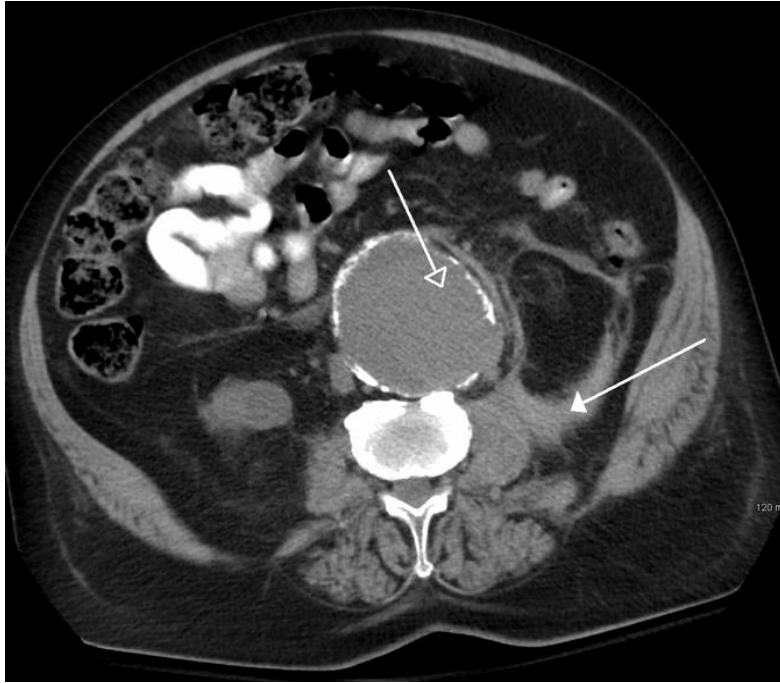
$\beta$

False Positive Rate

1

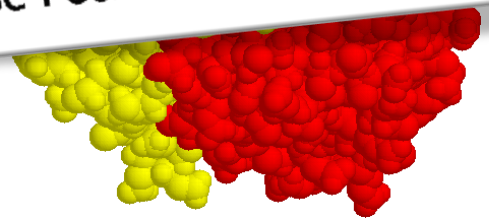
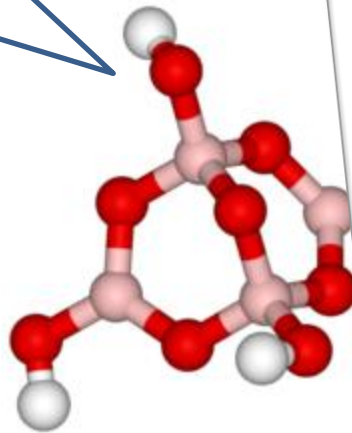
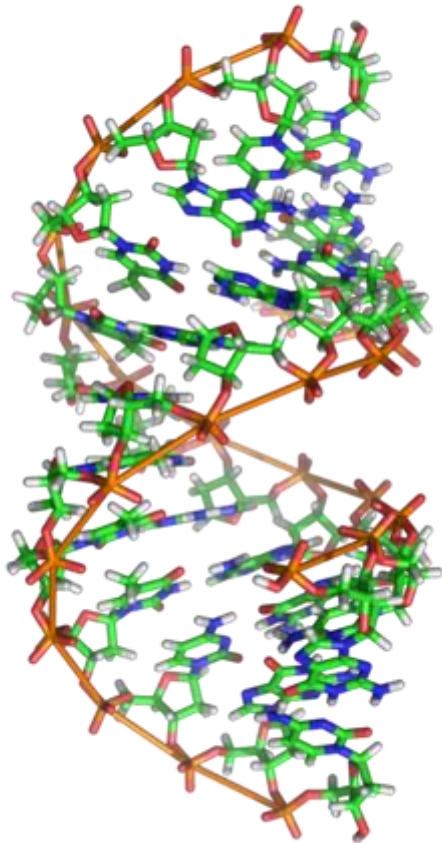
<http://www.google.com/>

# Medical Diagnosis



# Bioinformatics

- Drug Discovery
- Gene Prioritization
- Protein Interaction Prediction
- .....



Partial Area Under the ROC Curve is critical  
to many applications

# Partial AUC Optimization

- Asymmetric SVM:
  - Wu, S.-H., Lin, K.-P., Chen, C.-M., and Chen, M.-S. Asymmetric support vector machines: low false-positive learning under the user tolerance. In KDD, 2008.
- Boosting style algorithm:
  - Komori, O. and Eguchi, S. A boosting method for maximizing the partial area under the ROC curve. *BMC Bioinformatics*, 11:314, 2010.
  - Takenouchi, T., Komori, O., and Eguchi, S. An extension of the receiver operating characteristic curve and AUC-optimal classification. *Neural Computation*, 24, (10):2789–2824, 2012.
- Several heuristic approaches:
  - Pepe, M. S. and Thompson, M. L. Combining diagnostic test results to increase accuracy. *Biostatistics*, 1(2):123–140, 2000.
  - Ricamato, M. T. and Tortorella, F. Partial AUC maximization in a linear combination of dichotomizers. *Pattern Recognition*, 44(10-11):2669–2677, 2011.



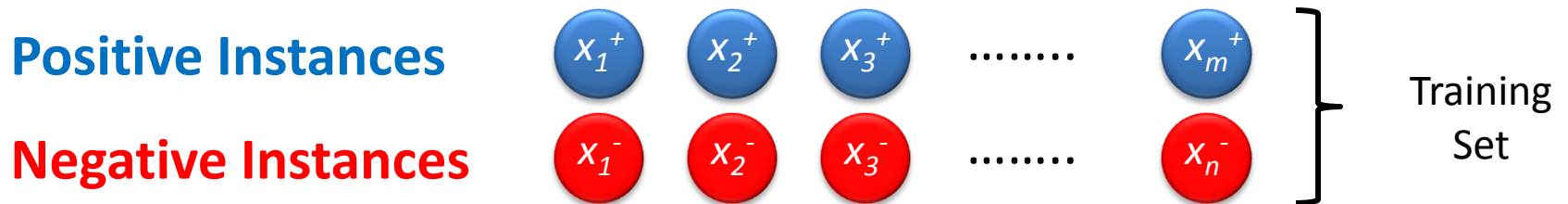
# Partial AUC Optimization

- Many of the existing approaches are either heuristic or solve special cases of the problem.
- **Our contribution:** New support vector methods for optimizing the general partial AUC measure.
- Based on Joachims' Structural SVM approach for optimizing full AUC, but leads to a trickier inner combinatorial optimization problem.
  - Joachims, T. A Support Vector Method for Multivariate Performance Measures. ICML, 2005.
  - Joachims, T. Training linear SVMs in linear time. KDD, 2006.
- Improvements over baselines on several real-world applications

# Outline

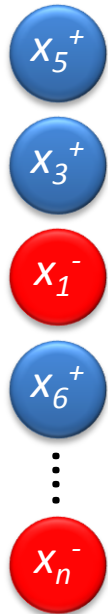
- Problem Setup
- **First cut:** Structural SVM Approach for Optimizing Partial AUC
- **Better Formulation:** Tighter Upper Bound on the Partial AUC Loss
- Experiments

# Receiver Operating Characteristic Curve



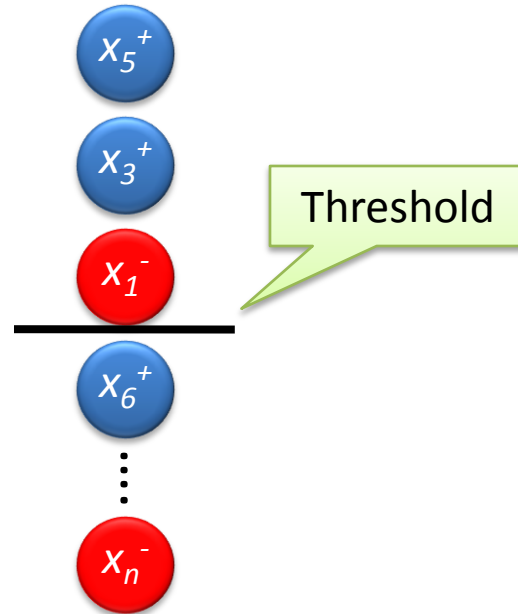
**GOAL?** Learn a scoring function  $f : X \rightarrow \mathbb{R}$

**Rank objects**

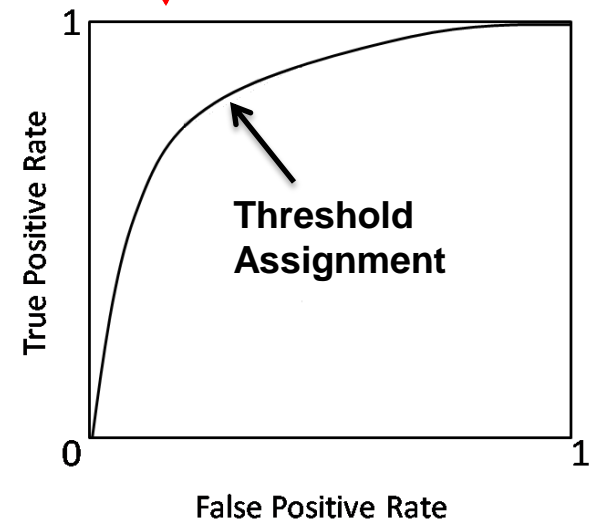


or

**Build a classifier**

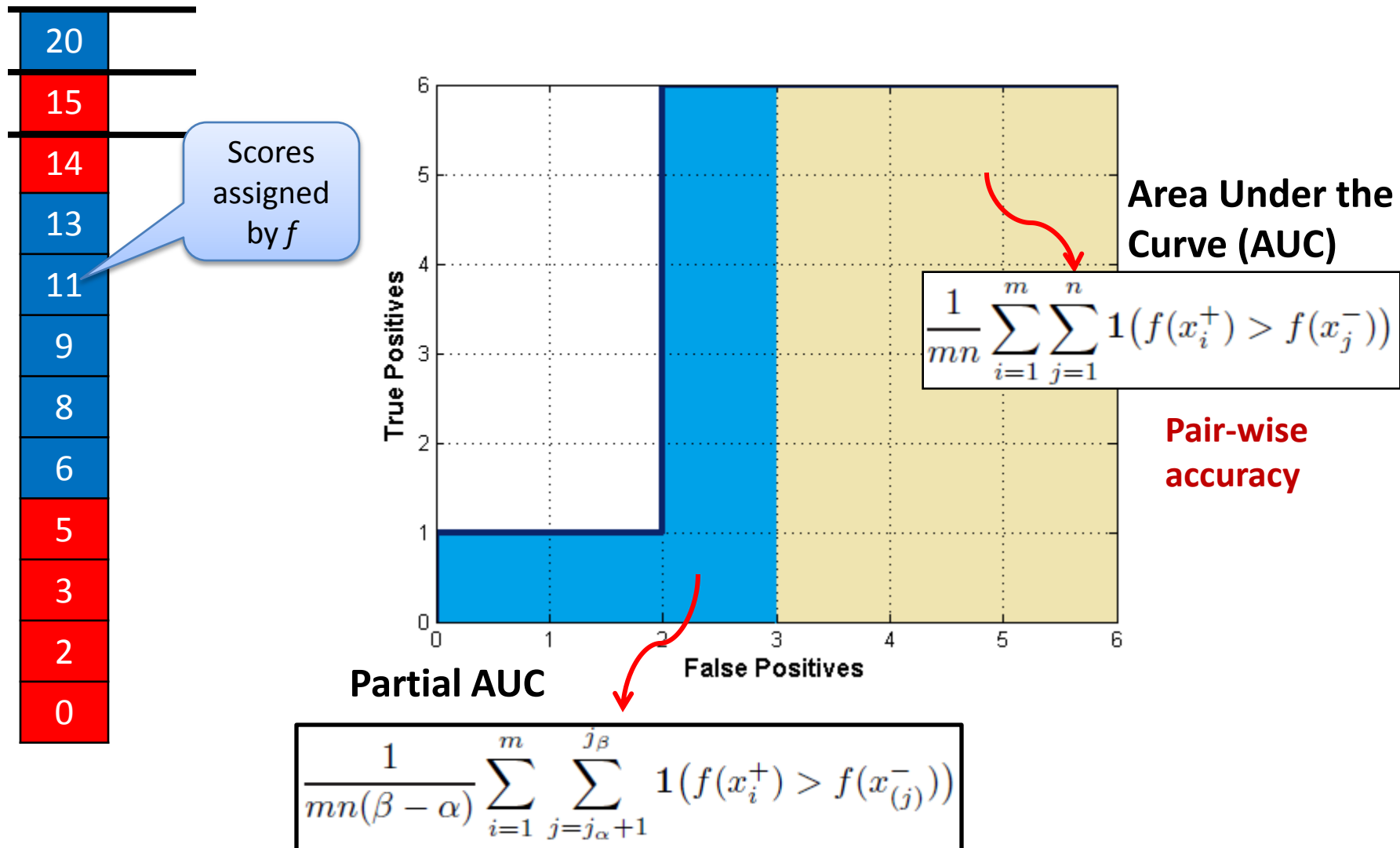


Quality of scoring function?



# ROC Curve

## Receiver Operating Characteristic Curve



# Partial AUC Optimization

Minimize:

$$1 - \widehat{\text{pAUC}}_f(\alpha, \beta)$$

Discrete and  
Non-differentiable



**Convex Upper Bound on** “ $1 - \widehat{\text{pAUC}}_f(\alpha, \beta)$ ” **+ Regularizer**

**Structural SVM Based Approach**

- Extends Joachims' approach for full AUC optimization, but leads to a trickier combinatorial optimization step.
- Efficient solver with the **same/lesser time complexity** compared to that for full AUC.

# Outline

- Problem Setup
- **First cut:** Structural SVM Approach for Optimizing Partial AUC
- **Better Formulation:** Tighter Upper Bound on the Partial AUC Loss
- Experiments

# Structural SVM Based Approach

Ordering of  $\{x_1, x_2, \dots, x_s\}$

$n$

$m$

0	0	0	0	0
1	1	0	0	0
1	1	0	0	1
1	1	0	0	1

$\pi$

compared with

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

$\pi^*$

IDEAL

Upper Bound on  $(1 - \text{pAUC})$

Regularizer

$$\min_{w, \xi \geq 0} \frac{1}{2} ||w||^2 + C\xi$$

s.t.

pAUC Loss

$$\forall \pi \in \Pi_{m,n} : w^\top (\phi(S, \pi^*) - \phi(S, \pi)) \geq \Delta_{\text{pAUC}(\alpha, \beta)}(\pi^*, \pi) - \xi$$

Exponential  
Number of Output  
Matrices!!

# Cutting-plane Solver

Converges in  
constant number  
of iterations

Repeat:

1. Solve OP for a subset of constraints.
2. Add the most violated constraint.

$$\min_{w, \xi \geq 0} \frac{1}{2} \|w\|^2 + C\xi$$

s.t.  $\forall \pi \in \mathcal{C} :$

$$w^\top (\phi(S, \pi^*) - \phi(S, \pi)) \geq \Delta_{\text{pAUC}(\alpha, \beta)}(\pi^*, \pi) - \xi$$

$$\operatorname{argmax}_{\pi} \Delta_{\text{pAUC}(\alpha, \beta)}(\pi^*, \pi) + w^\top (\phi(S, \pi^*) - \phi(S, \pi))$$

**Break down!**

**Full AUC**

0	1	0	1	0
1	1	0	0	0
1	1	0	0	1
1	1	0	0	1

**Partial AUC**

0	1	0	1	0
1	1	0	0	0
1	1	0	0	1
1	1	0	0	1

?



# Trickier Optimization Problem

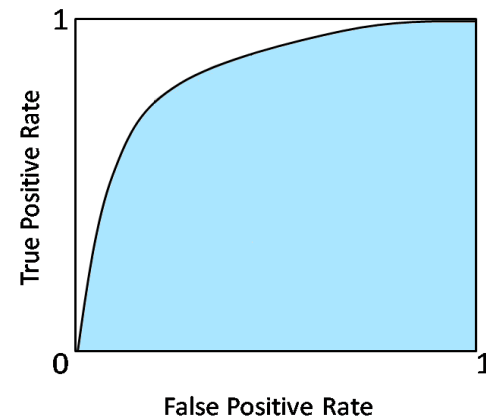
## Full AUC

All Pairs

$$\frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n \mathbf{1}(f(x_i^+) > f(x_j^-))$$

0	1	0	1	0
1	1	0	0	0
1	1	0	0	1
1	1	0	0	1

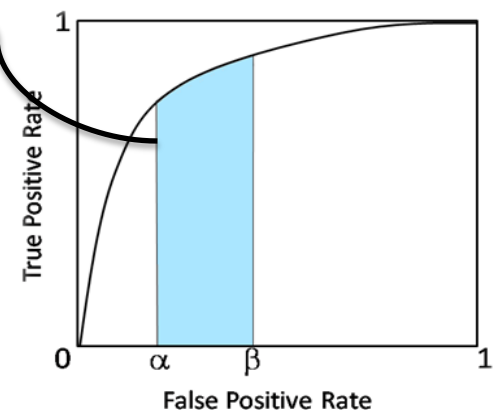
→  $\Sigma$  → AUC



## Partial AUC

$$\frac{1}{mn(\beta - \alpha)} \sum_{i=1}^m \sum_{j=j_\alpha+1}^{j_\beta} \mathbf{1}(f(x_i^+) > f(x_{(j)}^-))$$

Subset of negative instances in the  
FPR range  $[\alpha, \beta]$  – **changes with  
ordering**



# Trickier Optimization Problem

Full AUC

All Pairs

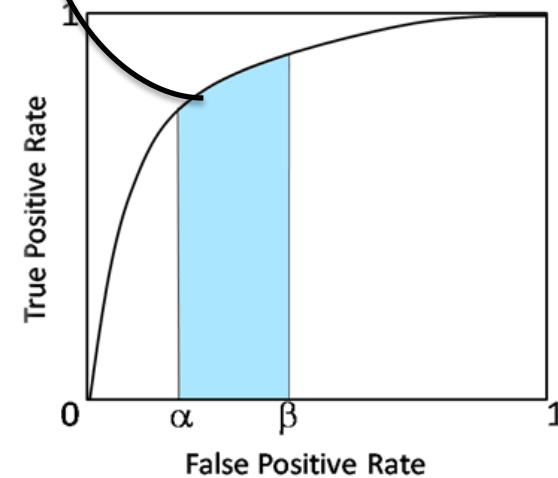
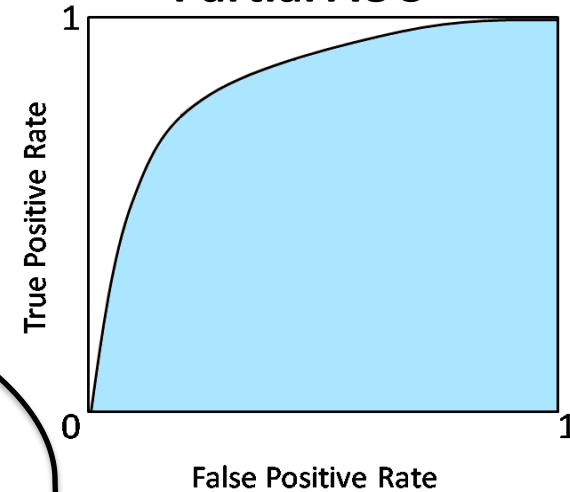
$$\frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n \mathbf{1}(f(x_i^+) > f(x_j^-))$$

Partial AUC

$$\frac{1}{mn(\beta - \alpha)} \sum_{i=1}^m \sum_{j=j_\alpha+1}^{j_\beta} \mathbf{1}(f(x_i^+) > f(x_{(j)}^-))$$

Subset of negative instances in the  
FPR range  $[\alpha, \beta]$  – **changes with  
ordering**

Partial AUC



# Trickier Optimization

Can be implemented in  $O((m+n) \log(m+n))$  time complexity

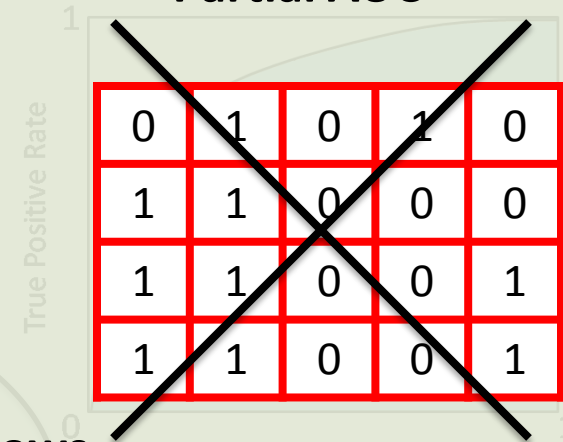
- 1: Inputs:  $S = (S_+, S_-)$ ,  $\alpha$ ,  $\beta$ ,  $w$
- 2: For  $i = 1, \dots, m$  do
- 3: Optimize over  $r_i \in \{0, \dots, j_\alpha - 1\}$ :  

$$\pi_{i,(j)_w}^{(1)} = \begin{cases} 1(w^\top x_{i,(j)_w}^\pm \leq 0), & j \in \{1, \dots, j_\alpha - 1\} \\ 0, & j \in \{j_\alpha, \dots, n\} \end{cases}$$
- 4: Optimize over  $r_i \in \{j_\alpha\}$ :  

$$\pi_{i,(j)_w}^{(2)} = \begin{cases} 1, & j \in \{1, \dots, j_\alpha\} \\ 0, & j \in \{j_\alpha + 1, \dots, n\} \end{cases}$$
- 5: Optimize over  $r_i \in \{j_\alpha + 1, \dots, n\}$ :  

$$\pi_{i,(j)_w}^{(3)} = \begin{cases} 1, & j \in \{1, \dots, j_\alpha + 1\} \\ 1(w^\top x_{i,(j)_w}^\pm \leq 1), & j \in \{j_\alpha + 2, \dots, j_\beta\} \\ 1(w^\top x_{i,(j)_w}^\pm \leq n\beta - j_\beta), & j = j_\beta + 1 \\ 1(w^\top x_{i,(j)_w}^\pm \leq 0), & j \in \{j_\beta + 2, \dots, n\} \end{cases}$$
- 6:  $\bar{k} = \operatorname{argmax}_{k \in \{1,2,3\}} \left\{ \begin{array}{l} \text{term inside sum over } i \text{ in} \\ \text{Eq. (4) evaluated at } \pi_i^{(k)} \end{array} \right\}$
- 7:  $\bar{\pi}_i = \pi_i^{(\bar{k})}$
- 8: End For
- 9: Output:  $\bar{\pi}$

## Partial AUC



Optimize rows independently

0	1	0	1	0
1	1	0	0	0
1	1	0	0	1
1	1	0	0	1

# Outline

- Problem Setup
- **First cut:** Structural SVM Approach for Optimizing Partial AUC
- **Better Formulation:** Tighter Upper Bound on the Partial AUC Loss
- Experiments

# Better Formulation

- Tighter upper bound on partial AUC loss
- Lesser time for finding most-violated constraint!
- Better guarantee on number of cutting-plane iterations!

$$\max_{z \in \mathcal{Z}_\beta} \sum_{x_j^- \in z} \sum_{i=1}^m \mathbf{1}(f(x_i^+) < f(x_j^-))$$

Max over subsets of negative instances

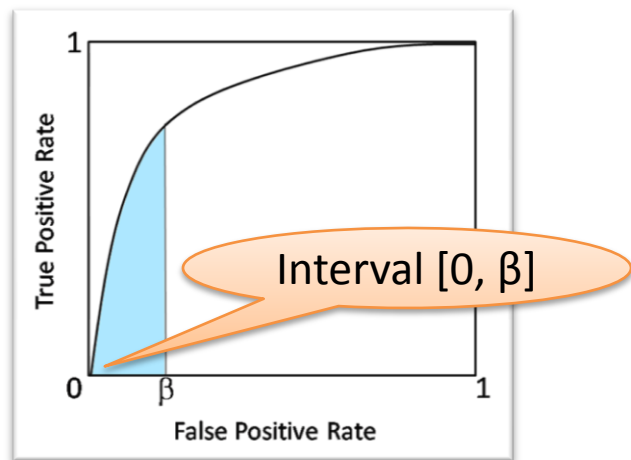
Truncated form of earlier objective

H. Narasimhan and S. Agarwal. *SVM\_pAUC<sup>tight</sup>: A New Support Vector Method for Optimizing Partial AUC Based on a Tight Convex Upper Bound*. KDD, 2013. To appear.

# Outline

- Problem Setup
- **First cut:** Structural SVM Approach for Optimizing Partial AUC
- **Better Formulation:** Tighter Upper Bound on the Partial AUC Loss
- Experiments

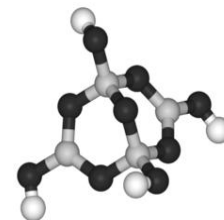
# SVMpAUC<sup>struct</sup> vs. Baseline Methods



## Drug Discovery

50 active compounds / 2092 inactive compounds

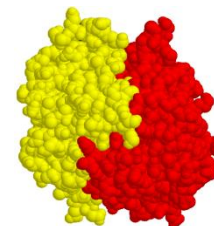
	pAUC(0, 0.1)
<b>SVM<sub>pAUC</sub>[0,0.1]</b>	<b>65.25</b>
SVM <sub>AUC</sub>	62.64 *
ASVM[0,0.1]	63.80
pAUCBoost[0,0.1]	43.89 *
Greedy-Heuristic[0,0.1]	8.33 *



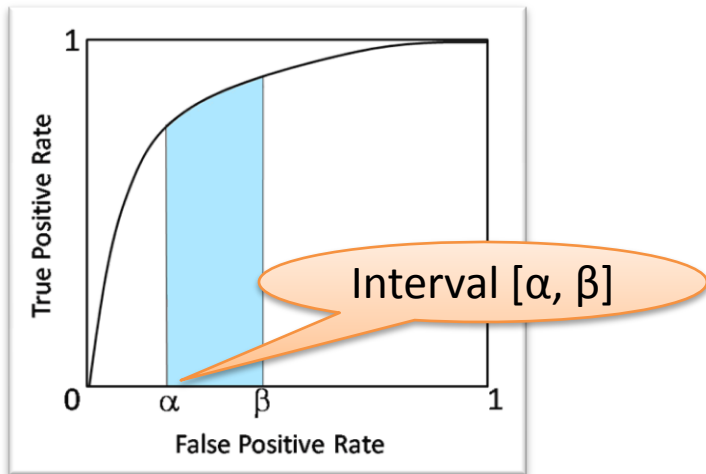
## Protein-Protein Interaction Prediction

$\sim 3 \times 10^3$  interacting pairs /  $\sim 2 \times 10^5$  non-interacting pairs

	pAUC(0, 0.1)
<b>SVM<sub>pAUC</sub>[0,0.1]</b>	<b>51.79</b>
SVM <sub>AUC</sub>	39.72 *
ASVM[0,0.1]	44.51 *
pAUCBoost[0,0.1]	48.65 *
Greedy-Heuristic[0,0.1]	47.33 *



# SVMpAUC<sup>struct</sup> vs. Baseline Methods



KDD Cup 2008

Breast Cancer Detection

~600 malignant ROIs /  $\sim 10^5$  benign ROIs

	$\hat{pAUC}(0.2s, 0.3s)$
SVM <sub>pAUC</sub> [0.2s, 0.3s]	51.44
SVM <sub>AUC</sub>	50.50
pAUCBoost[0.2s, 0.3s]	48.06 *
Greedy-Heuristic[0.2s, 0.3s]	46.99 *



# SVMpAUC<sup>tight</sup> vs. SVMpAUC<sup>struct</sup>

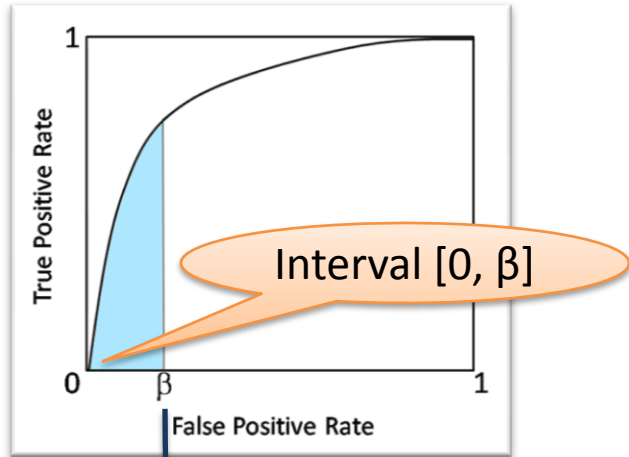
## Partial AUC in $[0, \beta]$

	SVM <sup>tight</sup> <sub>pAUC</sub> [0, 0.1]	SVM <sup>struct</sup> <sub>pAUC</sub> [0, 0.1]	SVM <sub>AUC</sub>
PPI	<b>52.95</b>	51.96 *	39.72 *
Cheminformatics	<b>65.30</b>	65.28	62.78
KDD Cup 2001	69.91	<b>70.12</b>	62.23 *
Leukemia	<b>30.44</b>	24.64 *	28.83
Ovarian Cancer	91.84	91.84	<b>92.17</b>

## Partial AUC in $[\alpha, \beta]$

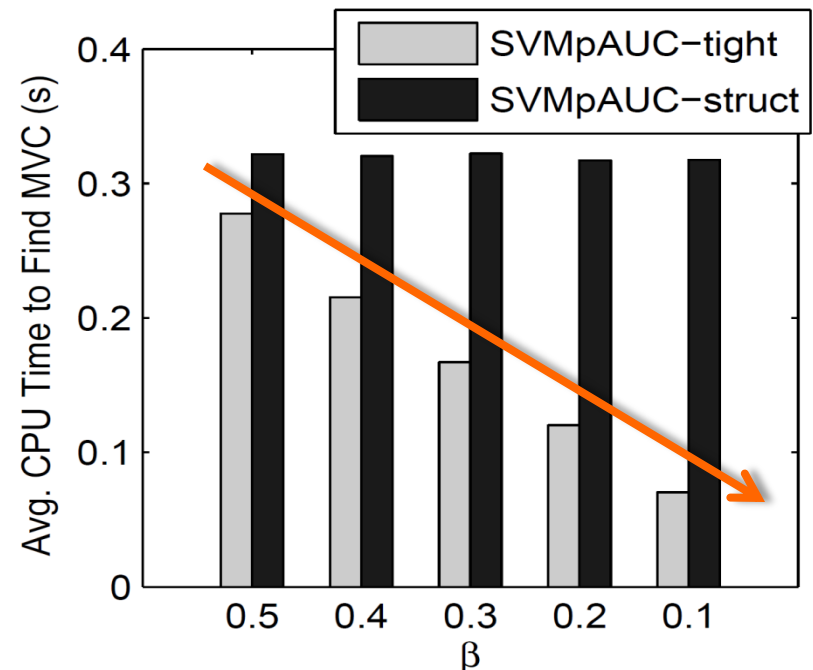
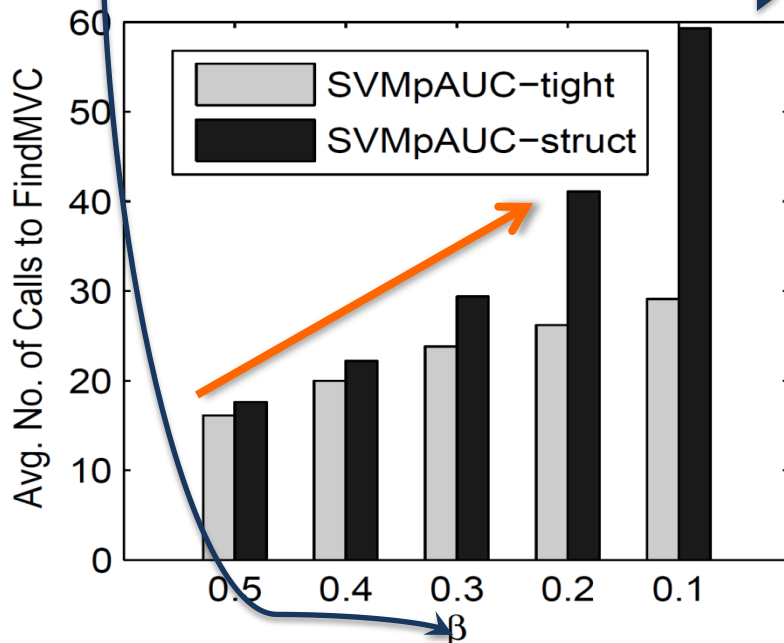
	SVM <sup>tight</sup> <sub>pAUC</sub> [0.2s, 0.3s]	SVM <sup>struct</sup> <sub>pAUC</sub> [0.2s, 0.3s]	SVM <sub>AUC</sub>
KDD Cup 2008	<b>53.43</b>	51.89	50.66

# Run-time Analysis



Repeat:

1. Solve OP for a subset of constraints.
2. Add the **most violated constraint**.



# Conclusions

- A **new structural SVM based** approach for optimizing partial AUC
- Efficient algorithm for solving the inner combinatorial optimization step
- Improved algorithm that optimizes a **tighter upper bound** on the partial AUC loss
- Experimental results confirm the **effectiveness** of our methods

Questions?