# Performance-Aligned Learning Algorithms with Statistical Guarantees

Rizal Zaini Ahmad Fathony

Committee: Prof. Brian Ziebart (Chair)
Prof. Bhaskar DasGupta
Prof. Xinhua Zhang
Prof. Lev Reyzin
Prof. Simon Lacoste-Julien

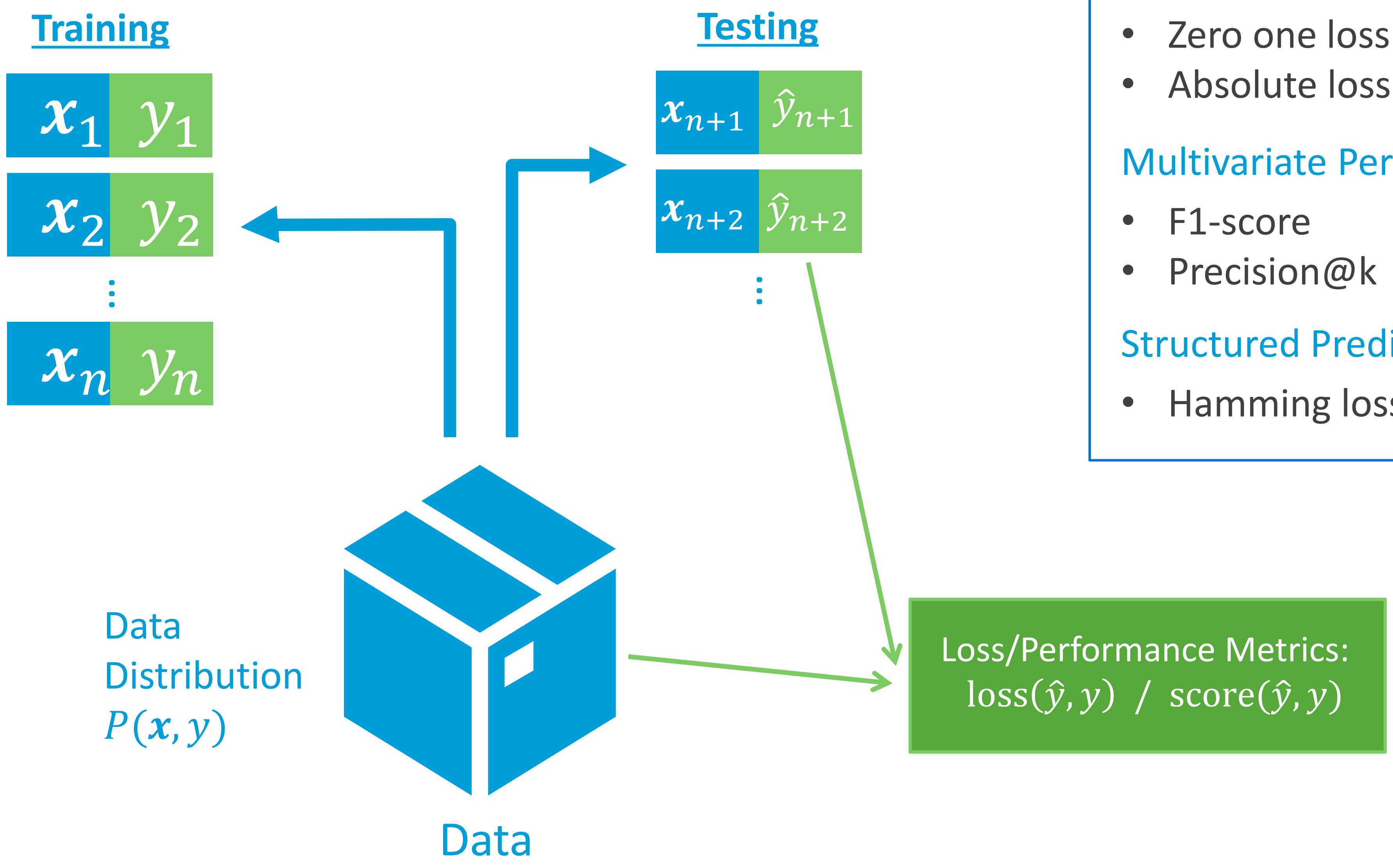
UIC COMPUTER SCIENCE

# Outline

"New learning algorithms that align with performance/loss metrics and provide the statistical guarantees of Fisher consistency"

# Introduction and Motivation

# Supervised Learning




### Multiclass Classification

- Zero one loss / accuracy metric
- Absolute loss (for ordinal regression)

### Multivariate Performance

- F1-score
- Precision@k

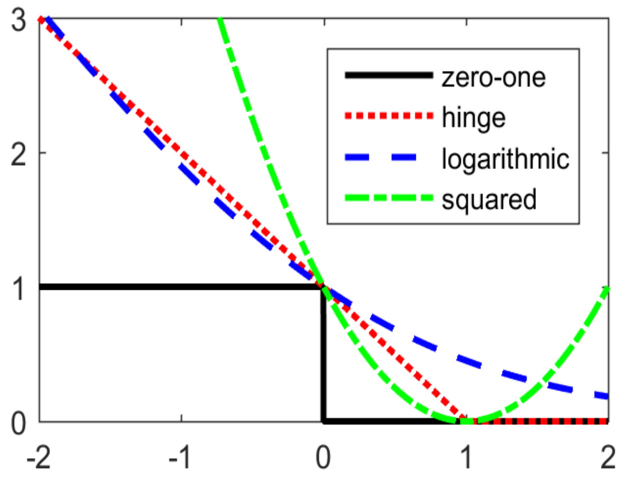
### Structured Prediction

- Hamming loss (sum of 0-1 loss)

# Empirical Risk Minimization (ERM) (Vapnik, 1992)

- Assume a family of parametric hypothesis function $f$ (e.g. linear discriminator)
- Find the hypothesis $f^*$ that minimize the empirical risk:

$$\min_f \frac{1}{n} \sum_{i=1}^{n} \text{loss}(f(\mathbf{x}_i), y_i) = \min_f \mathbb{E}_{\mathbf{X},Y\sim\tilde{P}} [\text{loss}(f(\mathbf{X}), Y)]$$




Non-convex, non-continuous metrics → Intractable optimization

Convex surrogate loss need to be employed!

A desirable property of convex surrogates:

## Fisher Consistency

Under ideal condition: optimize surrogate → minimizes the loss metric

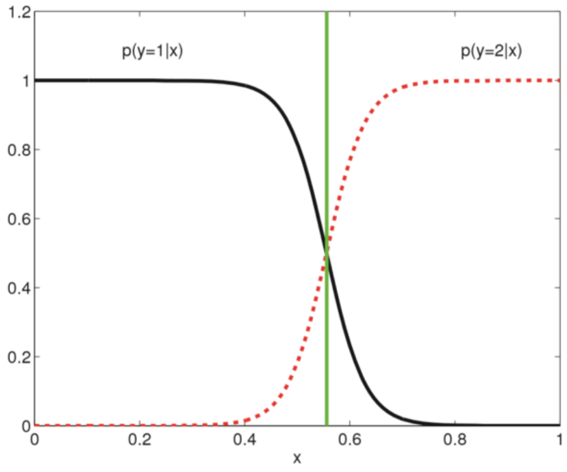
(given the true distribution and fully expressive model)

# Two Main Approaches

## 1 Probabilistic Approach

- Construct prediction probability model
- Employ the logistic loss surrogate

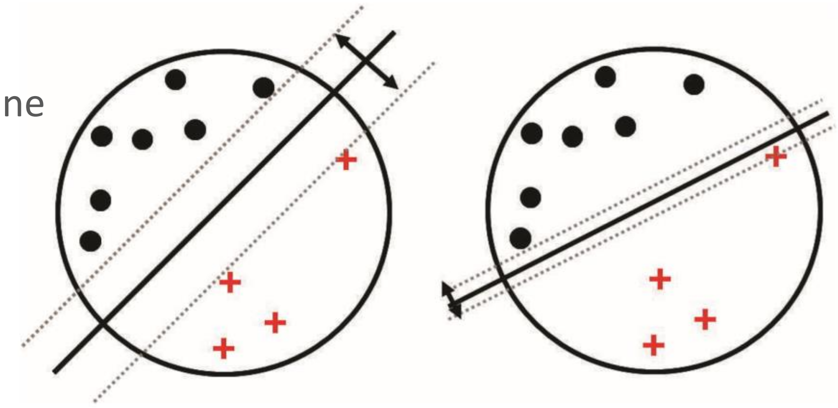
Logistic Regression, Conditional Random Fields (CRF)




## 2 Large-Margin Approach

- Maximize the margin that separates correct prediction from the incorrect one
- Employ the hinge loss surrogate

Support Vector Machine (SVM), Structured SVM



* Pictures are taken from MLPP book (Kevin Murphy)

# Multiclass Classification | Logistic Regression vs SVM

## 1 Multiclass Logistic Regression

✓ Statistical guarantee of Fisher consistency
(minimizes the zero-one loss metric in the limit)

✗ No dual parameter sparsity

## 2 Multiclass SVM

✗ Current multiclass SVM formulations:
- Lack Fisher consistency property, or
- Doesn't perform well in practice

✓ Computational efficiency
(via the kernel trick & dual parameter sparsity)

# Structured Prediction| CRF vs Structured SVM

## 1 Conditional Random Fields (CRF)

- ✓ Statistical guarantee of Fisher consistency
- ✗ No easy mechanism to incorporate customized loss/performance metrics
- ✗ Computation of the normalization term may be intractable

## 2 Structured SVM

- ✗ No Fisher consistency guaranteees
- ✓ Flexibility to incorporate customized loss/performance metrics
- ✓ Relatively more efficient in computation

# New Learning Algorithms?

- ✓ Align better with the loss/performance metric
  (by incorporating the metric into its learning objective)
- ✓ Provide Fisher consistency guarantee
- ✓ Computationally efficient
- ✓ Perform well in practice

## How?

Robust adversarial learning approach

*"What predictor best maximizes the performance metric (or minimizes the loss metric) in the worst case given the statistical summaries of the empirical distributions?"*

# Performance-Aligned Surrogate Losses for General Multiclass Classification

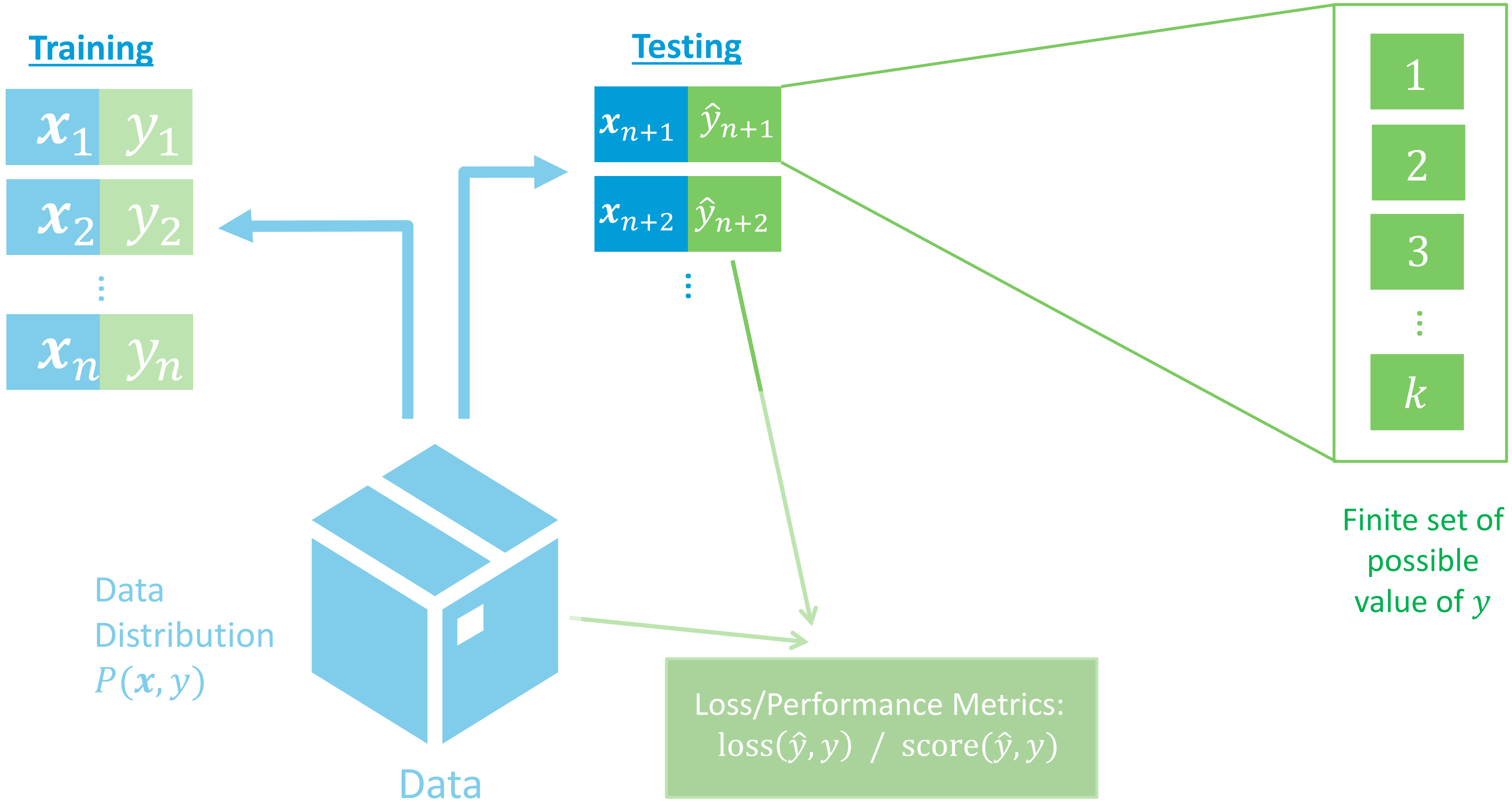
Based on:

**Fathony, R**., Asif, K., Liu, A., Bashiri, M. A., Xing, W., Behpour, S., Zhang, X., and Ziebart, B. D.: *Consistent robust adversarial prediction for general multiclass classification*. arXiv preprint arXiv:1812.07526, 2018. (Submitted to JMLR).

**Fathony, R.**, Liu, A., Asif, K., and Ziebart, B.: *Adversarial multiclass classification: A risk minimization perspective*. NIPS 2016.

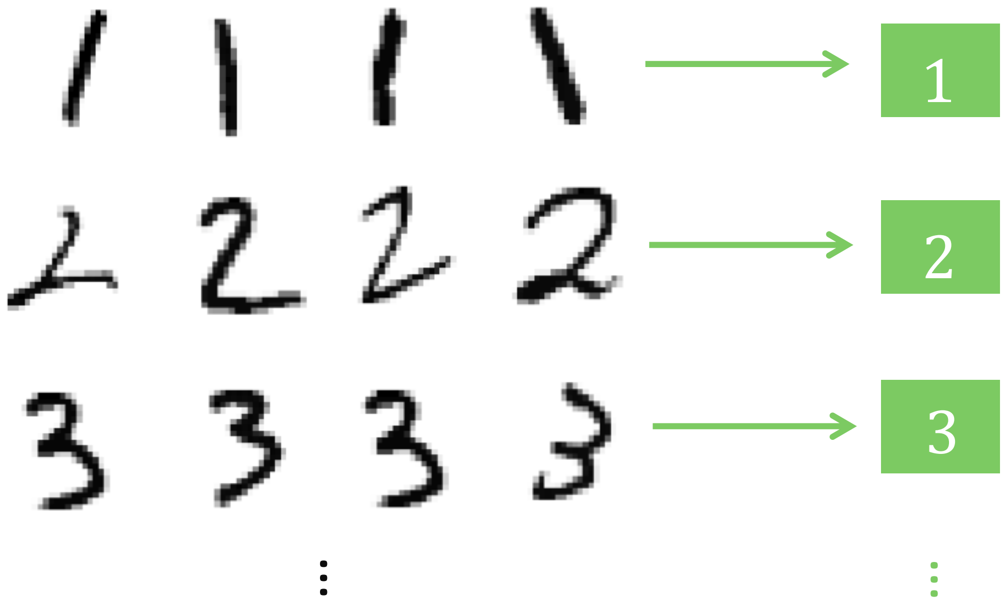
**Fathony, R**., Bashiri, M. A., and Ziebart, B.: Adversarial surrogate losses for ordinal regression. NIPS 2017.

# Supervised Learning | Multiclass Classification



# Multiclass Classification | Zero-One Loss

## Example: Digit Recognition




## Loss Metric: Zero-One Loss

Loss Metric:
$\text{loss}(\hat{y}, y) = I(\hat{y} \neq y)$

$$\mathbf{L} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

# Multiclass Classification | Ordinal Classification

## Example: Movie Rating Prediction


→ 1 ★☆☆☆☆


→ 2 ★★☆☆☆

⋮


→ 5 ★★★★★

## Loss Metric: Absolute Loss

Loss Metric:
$$\mathrm{loss}(\hat{y}, y) = |\hat{y} - y|$$

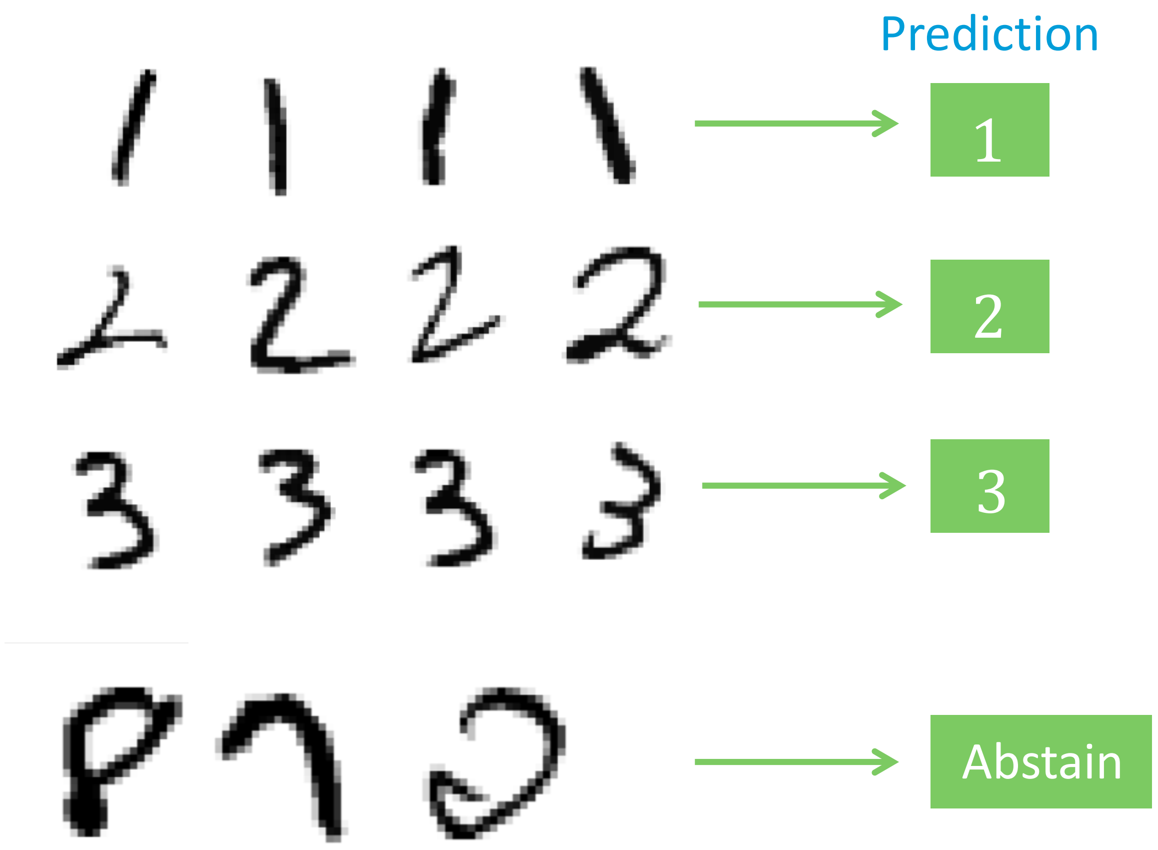
$$\mathbf{L} = \begin{bmatrix} 0 & 1 & 2 & 3 & 4 \\ 1 & 0 & 1 & 2 & 3 \\ 2 & 1 & 0 & 1 & 2 \\ 3 & 2 & 1 & 0 & 1 \\ 4 & 3 & 2 & 1 & 0 \end{bmatrix}$$

Predicted vs Actual Label:

Distance ↑ → Loss ↑

# Multiclass Classification | Classification with Abstention

## Predictor can say 'abstain'




## Loss Metric: Abstention Loss

Loss Metric:

$$\text{loss}(\hat{y}, y) = \begin{cases} \alpha & \text{if abstain} \\ I(\hat{y} \neq y) & \text{otherwise} \end{cases}$$

$$\mathbf{L} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ \alpha & \alpha & \alpha & \alpha & \alpha \end{bmatrix}$$

# Multiclass Classification | Other Loss Metrics

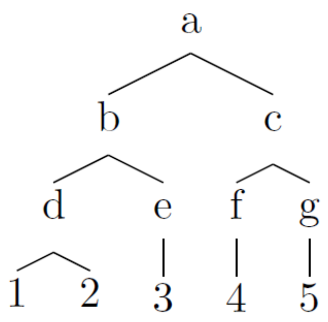
## Squared loss metric

$$\text{loss}(\hat{y}, y) = (\hat{y} - y)^2$$

$$\begin{bmatrix} 0 & 1 & 4 & 9 & 16 \\ 1 & 0 & 1 & 4 & 9 \\ 4 & 1 & 0 & 1 & 4 \\ 9 & 4 & 1 & 0 & 1 \\ 16 & 9 & 4 & 1 & 0 \end{bmatrix}$$

## Taxonomy-based loss metric

$$\text{loss}(\hat{y}, y) = h - v(\hat{y}, y) + 1$$




$$\begin{bmatrix} 0 & 1 & 2 & 3 & 3 \\ 1 & 0 & 2 & 3 & 3 \\ 2 & 2 & 0 & 3 & 3 \\ 3 & 3 & 3 & 0 & 2 \\ 3 & 3 & 3 & 2 & 0 \end{bmatrix}$$

## Cost-sensitive loss metric

$$\text{loss}(\hat{y}, y) = \mathbf{C}_{\hat{y},y}$$

$$\begin{bmatrix} 0 & 3 & 2 & 3 \\ 1 & 0 & 7 & 4 \\ 3 & 2 & 0 & 2 \\ 5 & 4 & 3 & 0 \end{bmatrix}$$

# Robust Adversarial Learning

# Robust Adversarial Learning (Grunwald & Dawid, 2004; Delage & Ye, 2010; Asif et.al, 2015)

## Original Loss Metric

Non-convex, non-continuous

$$\min_f \mathbb{E}_{\mathbf{X},Y\sim\tilde{P}}\left[\text{loss}(f(\mathbf{X}),Y)\right]$$

Approximate the loss with convex surrogates →

## Empirical Risk Minimization

$$\min_f \mathbb{E}_{\mathbf{X},Y\sim\tilde{P}}\left[\text{surrogate}(f(\mathbf{X}),Y)\right]$$

$$\hat{y} = \operatorname*{argmax}_j f_j(\mathbf{x})$$

↓ Probabilistic prediction $\hat{P}(\hat{Y}|\mathbf{X})$

$$\min_{\hat{P}(\hat{Y}|\mathbf{X})} \mathbb{E}_{\mathbf{X},Y\sim\tilde{P};\hat{Y}|\mathbf{X}\sim\hat{P}}\left[\text{loss}(\hat{Y},Y)\right]$$

↓ Evaluate against an adversary, instead of using empirical data

Adversary's probabilistic prediction $\check{P}(\check{Y}|\mathbf{X})$

$$\min_{\hat{P}(\hat{Y}|\mathbf{X})} \max_{\check{P}(\check{Y}|\mathbf{X})} \mathbb{E}_{\mathbf{X},Y\sim\tilde{P};\hat{Y}|\mathbf{X}\sim\hat{P};\check{Y}|\mathbf{X}\sim\check{P}}\left[\text{loss}(\hat{Y},\check{Y})\right]$$

→ **Constraint** the **statistics** of the **adversary**'s distribution to match the **empirical** statistics

## Robust Adversarial Learning

$$\min_{\hat{P}(\hat{Y}|\mathbf{X})} \max_{\check{P}(\check{Y}|\mathbf{X})} \mathbb{E}_{\mathbf{X},Y\sim\tilde{P};\hat{Y}|\mathbf{X}\sim\hat{P};\check{Y}|\mathbf{X}\sim\check{P}}\left[\text{loss}(\hat{Y},\check{Y})\right]$$

$$\text{s.t. } \mathbb{E}_{\mathbf{X}\sim\tilde{P};\check{Y}|\mathbf{X}\sim\check{P}}[\phi(\mathbf{X},\check{Y})] = \mathbb{E}_{\mathbf{X},Y\sim\tilde{P}}[\phi(\mathbf{X},Y)]$$

# Robust Adversarial Dual Formulation

Primal:

$$\min_{\hat{P}(\hat{Y}|\mathbf{X})} \max_{\check{P}(\check{Y}|\mathbf{X})} \mathbb{E}_{\mathbf{X}\sim\tilde{P};\hat{Y}|\mathbf{X}\sim\hat{P};\check{Y}|\mathbf{X}\sim\check{P}}\left[\text{loss}(\hat{Y},\check{Y})\right]$$

$$\text{subject to: } \mathbb{E}_{\mathbf{X}\sim\tilde{P};\check{Y}|\mathbf{X}\sim\check{P}}[\phi(\mathbf{X},\check{Y})] = \mathbb{E}_{\mathbf{X},Y\sim\tilde{P}}[\phi(\mathbf{X},Y)]$$

Lagrange multiplier, minimax duality

Dual:

$$\min_{\theta} \mathbb{E}_{\mathbf{X},Y\sim\tilde{P}} \max_{\check{P}(\check{Y}|\mathbf{X})} \min_{\hat{P}(\hat{Y}|\mathbf{X})} \mathbb{E}_{\hat{Y}|\mathbf{X}\sim\hat{P};\check{Y}|\mathbf{X}\sim\check{P}}\left[\text{loss}(\hat{Y},\check{Y}) + \theta^{\intercal}\left(\phi(\mathbf{X},\check{Y}) - \phi(\mathbf{X},Y)\right)\right]$$

ERM with the adversarial surrogate loss (AL):

$$AL(\mathbf{x},y,\theta) = \max_{\check{P}(\check{Y}|\mathbf{x})} \min_{\hat{P}(\hat{Y}|\mathbf{x})} \mathbb{E}_{\hat{Y}|\mathbf{x}\sim\hat{P};\check{Y}|\mathbf{x}\sim\check{P}}\left[\text{loss}(\hat{Y},\check{Y}) + \theta^{\intercal}\left(\phi(\mathbf{x},\check{Y}) - \phi(\mathbf{x},y)\right)\right]$$

Convex in $\theta$

Simplified notation

$$AL(\mathbf{f},y) = \max_{\mathbf{q}\in\Delta} \min_{\mathbf{p}\in\Delta} \mathbf{p}^{\intercal}\mathbf{L}\mathbf{q} + \mathbf{f}^{\intercal}\mathbf{q} - f_y$$

where:

$$p_i = \hat{P}(\hat{Y} = i|\mathbf{x})$$

$$q_i = \check{P}(\check{Y} = i|\mathbf{x})$$

$$f_i = \theta^{\intercal}\phi(\mathbf{x},i)$$

# Adversarial Surrogate Loss

## Adversarial Surrogate Loss

$$AL(\mathbf{f}, y) = \max_{\mathbf{q}\in\Delta} \min_{\mathbf{p}\in\Delta} \mathbf{p}^\intercal \mathbf{L}\mathbf{q} + \mathbf{f}^\intercal \mathbf{q} - f_y$$

## Convert to a Linear Program

$$\begin{aligned} AL(\mathbf{f}, y) = \max_{\mathbf{q}, v} \; & v + \mathbf{f}^\intercal \mathbf{q} - f_y \\ \text{s.t.: } & \mathbf{L}_{(i,:)}\mathbf{q} \geq v \quad \forall i \in [k] \\ & q_i \geq 0 \quad \forall i \in [k] \\ & \mathbf{q}^\intercal \mathbf{1} = 1 \end{aligned}$$

→ LP Solver $O(k^{3.5})$

## Convex Polytope formed by the constraints

$$\mathbb{C} = \left\{ \begin{bmatrix} \mathbf{q} \\ v \end{bmatrix} \middle| \mathbf{A} \begin{bmatrix} \mathbf{q} \\ v \end{bmatrix} \geq \mathbf{b}, \quad \text{where} \quad \mathbf{A} = \begin{bmatrix} \mathbf{L} & -\mathbf{1} \\ \mathbf{I} & \mathbf{0} \\ \mathbf{1}^\intercal & 0 \\ -\mathbf{1}^\intercal & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ 1 \\ -1 \end{bmatrix} \right\}$$

## Example for a four class classification

$$\begin{array}{l} \text{1st block} \\ \\ \text{2nd block} \\ \\ \text{3rd block} \end{array} \begin{bmatrix} 0 & 1 & 1 & 1 & -1 \\ 1 & 0 & 1 & 1 & -1 \\ 1 & 1 & 0 & 1 & -1 \\ 1 & 1 & 1 & 0 & -1 \\ \hdashline 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \hdashline 1 & 1 & 1 & 1 & 0 \\ -1 & -1 & -1 & -1 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ v \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}$$

## Extreme points of the (bounded) polytope

There is always an optimal solution that is an extreme point of the domain.

Computing AL =
finding the best extreme point

# Zero-One Loss : AL$^{0\text{-}1}$ | Convex Polytope

## Convex Polytope of the AL$^{0\text{-}1}$

$$\mathbb{C} = \left\{ \begin{bmatrix} \mathbf{q} \\ v \end{bmatrix} \middle| \mathbf{A} \begin{bmatrix} \mathbf{q} \\ v \end{bmatrix} \geq \mathbf{b}, \quad \text{where} \quad \mathbf{A} = \begin{bmatrix} \mathbf{L} & -\mathbf{1} \\ \mathbf{I} & \mathbf{0} \\ \mathbf{1}^\intercal & 0 \\ -\mathbf{1}^\intercal & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ 1 \\ -1 \end{bmatrix} \right\}$$

$$\mathbf{L} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

## Extreme points of the polytope

$$D = \left\{ \begin{bmatrix} \mathbf{q} \\ v \end{bmatrix} = \frac{1}{|S|} \begin{bmatrix} \sum_{i \in S} \mathbf{e}_i \\ |S| - 1 \end{bmatrix} \middle| \emptyset \neq S \subseteq [k] \right\}$$

$\boldsymbol{e}_i$ is a vector with a single 1 at the *i*-th index, and 0 elsewhere.

$$[k] \triangleq \{1, \ldots, k\}$$

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

## The Adversarial Surrogate Loss for Zero-One Loss Metrics (AL$^{0\text{-}1}$)

$$AL^{0\text{-}1}(\mathbf{f}, y) = \max_{S \subseteq [k],\, S \neq \emptyset} \frac{\sum_{i \in S} f_i + |S| - 1}{|S|} - f_y$$

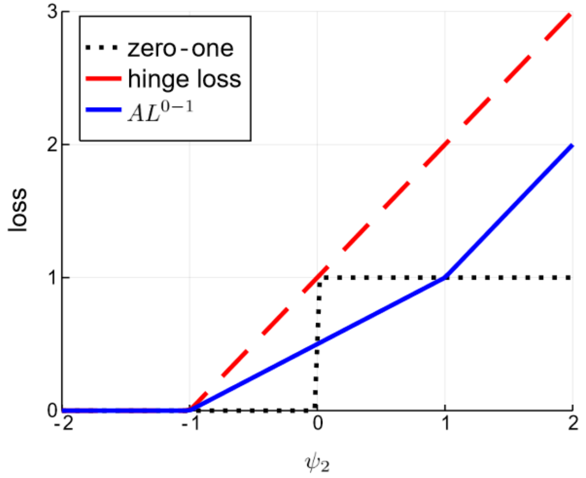
## Computation of AL$^{0\text{-}1}$

- Sort $f_i$ in non-increasing order
- Incrementally add potentials to the set $S$, until adding more potential decrease the loss value

O$(k \log k)$, where $k$ is the number of classes

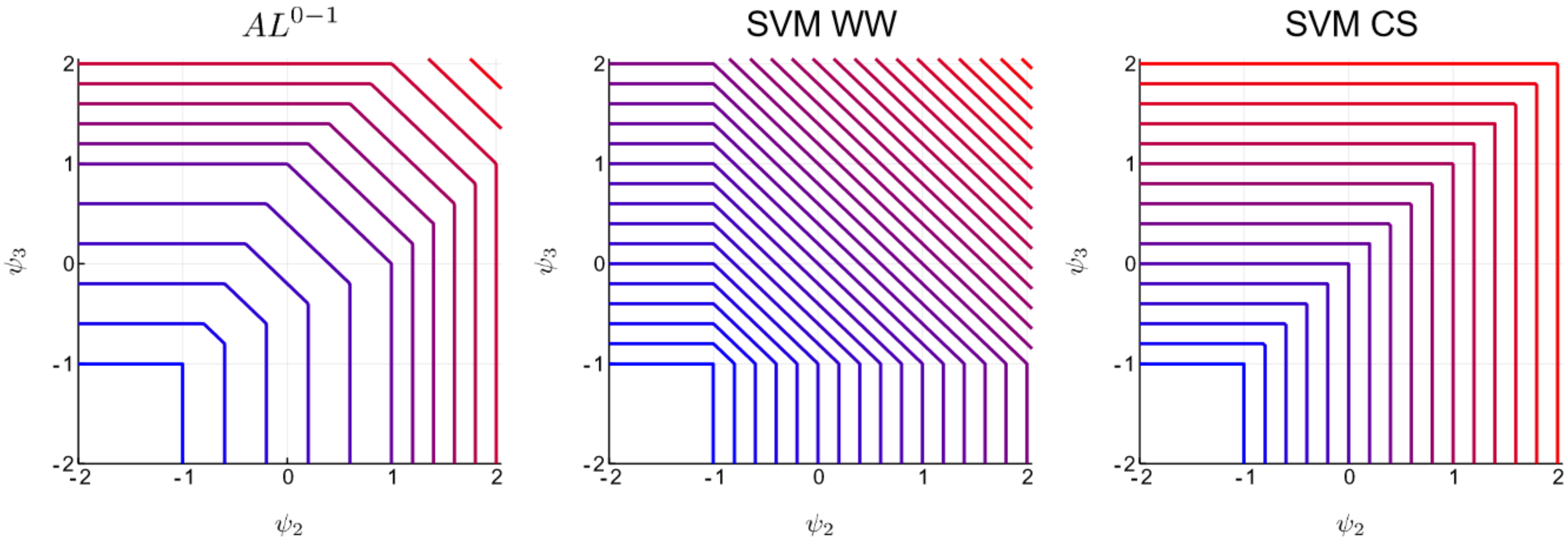
# $AL^{0\text{-}1}$| Loss Surface

## Binary Classification




- Plots over the space of potential differences $\psi_i = f_i - f_y$
- The true label is $y = 1$

## Three Class Classification



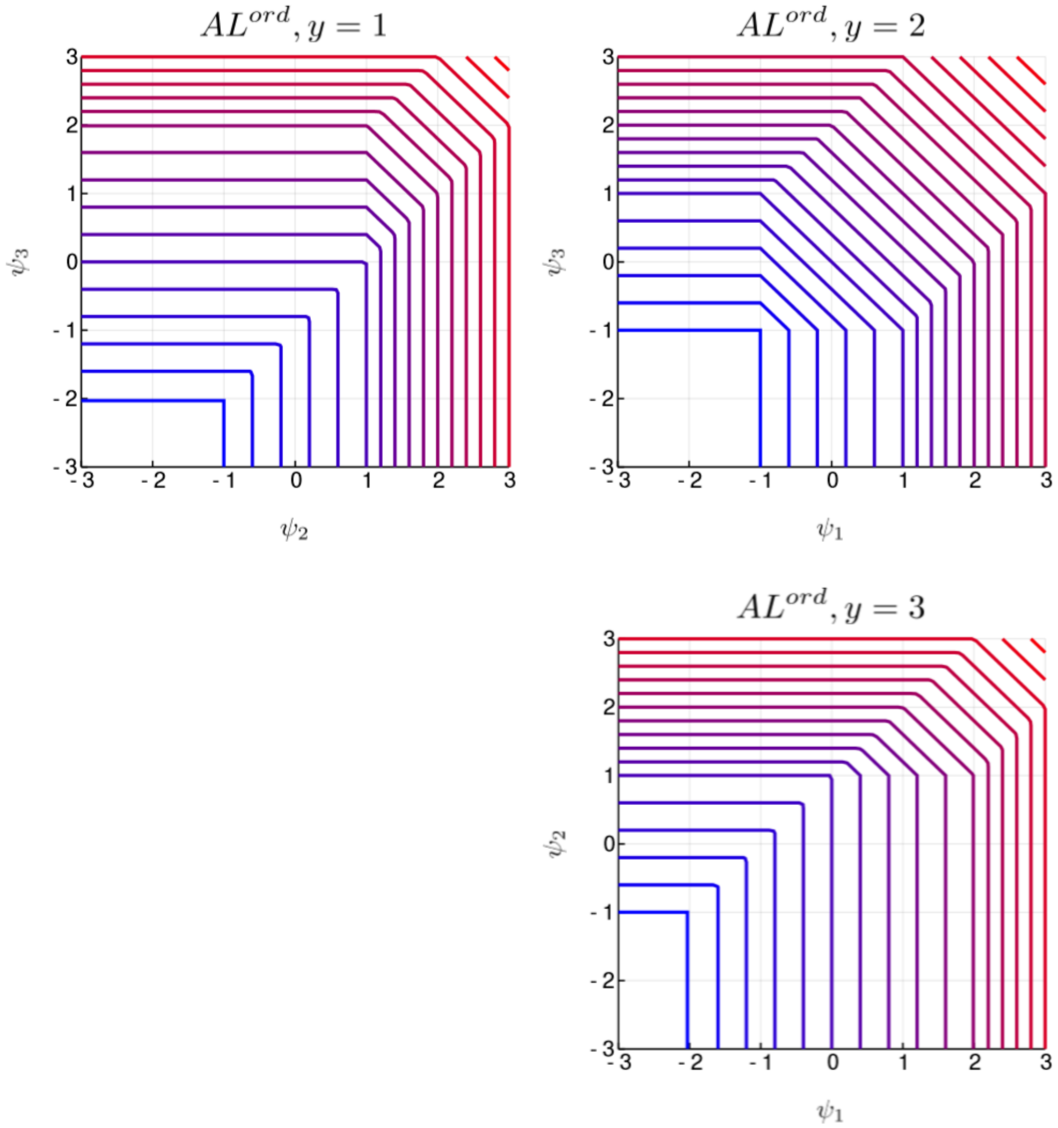
# Other Multiclass Loss Metrics

## Ordinal Regression with Absolute Loss Metric

Extreme points of the polytope:

$$D = \left\{ \begin{bmatrix} \mathbf{q} \\ v \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \mathbf{e}_i + \mathbf{e}_j \\ j - i \end{bmatrix} \middle| \; i, j \in [k]; i \leq j \right\}$$

$\boldsymbol{e}_i$ is a vector with a single 1 at the *i*-th index, and 0 elsewhere.




Adversarial Surrogate Loss $AL^{ord}$:

$$AL^{ord}(\mathbf{f}, y) = \max_{i,j \in [k]} \frac{f_i + f_j + j - i}{2} - f_y$$

Computation cost:
$O(k)$, where $k$ is the number of classes

# Other Multiclass Loss Metrics

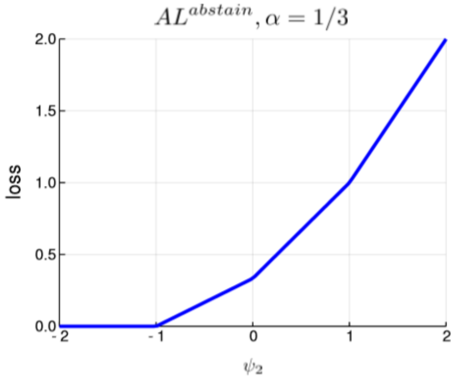
$$\begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ \alpha & \alpha & \alpha & \alpha & \alpha \end{bmatrix}$$

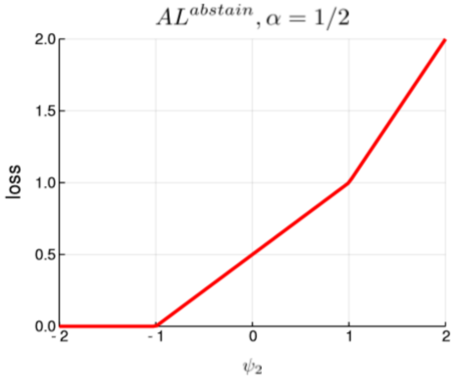
## Classification with Abstention $(0 \leq \alpha \leq 0.5)$

### Extreme points of the polytope:

$$D = \left\{ \begin{bmatrix} \mathbf{q} \\ v \end{bmatrix} = (1-\alpha) \begin{bmatrix} \mathbf{e}_i \\ 0 \end{bmatrix} + \alpha \begin{bmatrix} \mathbf{e}_j \\ 1 \end{bmatrix} \middle| \begin{array}{c} i, j \in [k] \\ i \neq j \end{array} \right\} \cup \left\{ \begin{bmatrix} \mathbf{q} \\ v \end{bmatrix} = \begin{bmatrix} \mathbf{e}_i \\ 0 \end{bmatrix} \middle| \, i \in [k] \right\}$$

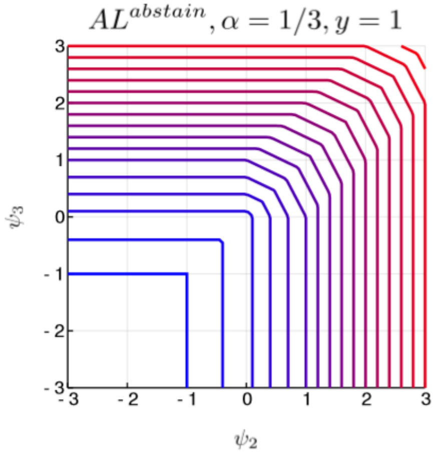
$\boldsymbol{e}_i$ is a vector with a single 1 at the *i*-th index, and 0 elsewhere.

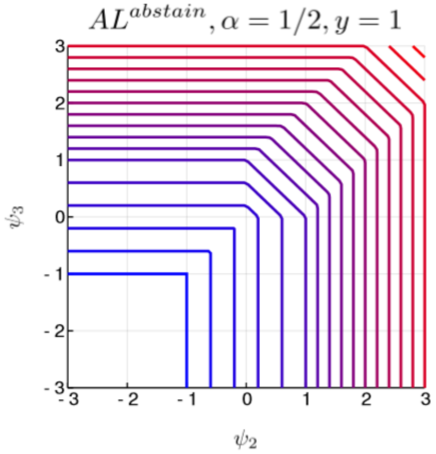



### Adversarial Surrogate Loss AL$^{abstain}$:

$$AL^{abstain}(\mathbf{f}, y, \alpha) = \max \left\{ \max_{i,j \in [k], i \neq j} (1-\alpha) f_i + \alpha f_j + \alpha, \ \max_i f_i \right\} - f_y$$




Computation cost:
O$(k)$, where $k$ is the number of classes

# Fisher Consistency

## Fisher Consistency Requirement in Multiclass Classification

$$f^* \in \mathcal{F}^* \triangleq \operatorname*{argmin}_f \mathbb{E}_{Y|\mathbf{x}\sim P}\left[\mathrm{AL}_f(\mathbf{x}, Y)\right]$$

- $P(Y|\boldsymbol{x})$ is the true conditional distribution
- $f$ is optimized over all measurable functions

$$\Rightarrow \operatorname*{argmax}_y f^*(\mathbf{x}, y) \subseteq \mathcal{Y}^\diamond \triangleq \operatorname*{argmin}_{y'} \mathbb{E}_{Y|\mathbf{x}\sim P}[\mathrm{loss}(y', Y)]$$

→ Bayes risk minimizer

## Minimizer Property

$$\mathbf{f}^* \in \operatorname*{argmin}_{\mathbf{f}} \max_{\mathbf{q}\in\Delta} \min_{\mathbf{p}\in\Delta} \{\mathbf{f}^\intercal\mathbf{q} + \mathbf{p}^\intercal\mathbf{L}\mathbf{q} - \mathbf{d}^\intercal\mathbf{f}\} = \operatorname*{argmin}_{\mathbf{f}} \max_{\mathbf{q}\in\Delta} \left\{\mathbf{f}^\intercal\mathbf{q} + \min_y(\mathbf{L}\mathbf{q})_y - \mathbf{d}^\intercal\mathbf{f}\right\}$$

- $\mathbf{d}$ is the true conditional distribution
- $y^\diamond$ is the Bayes optimal predictor

Under $\mathbf{f}^*$: → $\mathbf{f}^* + \mathbf{L}_{(y^\diamond,:)}{}^\intercal$ is a uniform vector

## Consistency

$\mathbf{f}^* + \mathbf{L}_{(y^\diamond,:)}{}^\intercal$ is a uniform vector → $\operatorname{argmax}_y f^*(\mathbf{x}, y) = \operatorname{argmin}_y \mathbf{L}_{(y^\diamond, y)}$ → Fisher consistent

# Optimization

## Sub-gradient descent

$$Q^* = \operatorname*{argmax}_{\mathbf{q}\in\Delta} \min_{\mathbf{p}\in\Delta} \left\{ \mathbf{p}^\intercal \mathbf{L} \mathbf{q} + \theta^\intercal \left[ \textstyle\sum_j q_j \phi(\mathbf{x}, j) - \phi(\mathbf{x}, y) \right] \right\}$$

$$\partial_\theta \, AL(\mathbf{x}, y, \theta) = \mathbf{conv} \left\{ \textstyle\sum_j q_j^* \phi(\mathbf{x}, j) - \phi(\mathbf{x}, y) \;\middle|\; \mathbf{q} \in Q^* \right\}$$

Example: $AL^{0\text{-}1}$

$$\partial_\theta \, AL^{0\text{-}1}(\mathbf{x}, y, \theta) \ni \frac{1}{|S^*|} \textstyle\sum_{j \in S^*} \phi(\mathbf{x}, j) - \phi(\mathbf{x}, y)$$

$S^*$ is the set that maximize $AL^{0\text{-}1}$

## Incorporate Rich Feature Spaces via the Kernel Trick

input space $\boldsymbol{x}_i$ → rich feature space $\omega(\boldsymbol{x}_i)$

Compute the dot products

$$K(\mathbf{x}_i, \mathbf{x}_j) = \omega(\mathbf{x}_i) \cdot \omega(\mathbf{x}_j)$$

1. Dual Optimization (benefit: dual parameter sparsity)
2. Primal Optimization (via PEGASOS (Shalev-Shwartz, 2010))

# Experiments:
## Example: Multiclass Classification (0-1 loss)

# Multiclass Classification | Related Works

## Multiclass Support Vector Machine (SVM)

1. The WW Model (Weston et.al., 2002)

$$\text{loss}_{\text{WW}}(\mathbf{x}_i, y_i) = \sum_{j \neq y_i} [1 - (f_{y_i}(\mathbf{x}_i) - f_j(\mathbf{x}_i))]_+$$

Relative Margin Model

2. The CS Model (Crammer and Singer, 1999)

$$\text{loss}_{\text{CS}}(\mathbf{x}_i, y_i) = \max_{j \neq y_i} [1 - (f_{y_i}(\mathbf{x}_i) - f_j(\mathbf{x}_i))]_+$$

Relative Margin Model

3. The LLW Model (Lee et.al., 2004)

$$\text{loss}_{\text{LLW}}(\mathbf{x}_i, y_i) = \sum_{j \neq y_i} [1 + f_j(\mathbf{x}_i)]_+$$

with: $\sum_j f_j(\mathbf{x}_i) = 0$

Absolute Margin Model

| Model | **Fisher Consistent?** (Tewari and Bartlett, 2007) (Liu, 2007) | **Perform well** in low dimensional feature? (Dogan et.al., 2016) |
|---|---|---|
| 1. The WW Model | ✗ | ✓ |
| 2. The CS Model | ✗ | ✓ |
| 3. The LLW Model | ✓ | ✗ |

# $AL^{0\text{-}1}$ | Experiments

Dataset properties and $AL^{0\text{-}1}$ constraints

| Dataset | Properties | | | | SVM | $AL^{0\text{-}1}$ constraints added and active | | | |
|---|---|---|---|---|---|---|---|---|---|
| | #class | #train | # test | #feat. | constraints | Linear kernel | | Gauss. kernel | |
| (1) iris | 3 | 105 | 45 | 4 | 210 | 213 | 13 | 223 | 38 |
| (2) glass | 6 | 149 | 65 | 9 | 745 | 578 | 125 | 490 | 252 |
| (3) redwine | 10 | 1119 | 480 | 11 | 10071 | 5995 | 1681 | 3811 | 1783 |
| (4) ecoli | 8 | 235 | 101 | 7 | 1645 | 614 | 117 | 821 | 130 |
| (5) vehicle | 4 | 592 | 254 | 18 | 1776 | 1310 | 311 | 1201 | 248 |
| (6) segment | 7 | 1617 | 693 | 19 | 9702 | 4410 | 244 | 4312 | 469 |
| (7) sat | 7 | 4435 | 2000 | 36 | 26610 | 11721 | 1524 | 11860 | 6269 |
| (8) optdigits | 10 | 3823 | 1797 | 64 | 34407 | 7932 | 597 | 10072 | 2315 |
| (9) pageblocks | 5 | 3831 | 1642 | 10 | 15324 | 9459 | 427 | 9155 | 551 |
| (10) libras | 15 | 252 | 108 | 90 | 3528 | 1592 | 389 | 1165 | 353 |
| (11) vertebral | 3 | 217 | 93 | 6 | 434 | 344 | 78 | 342 | 86 |
| (12) breasttissue | 6 | 74 | 32 | 9 | 370 | 258 | 65 | 271 | 145 |

12 datasets

dual parameter sparsity

# $AL^{0\text{-}1}$ | Experiments | Results

## Results for Linear Kernel and Gaussian Kernel

The **mean (standard deviation)** of the **accuracy.** Bold numbers: **best** or **not significantly worse** than the best

| D | Linear Kernel | | | | Gaussian Kernel | | | |
|---|---|---|---|---|---|---|---|---|
| | $AL^{0\text{-}1}$ | WW | CS | LLW | $AL^{0\text{-}1}$ | WW | CS | LLW |
| (1) | **96.3** (3.1) | **96.0** (2.6) | **96.3** (2.4) | 79.7 (5.5) | **96.7** (2.4) | **96.4** (2.4) | **96.2** (2.3) | 95.4 (2.1) |
| (2) | **62.5** (6.0) | **62.2** (3.6) | **62.5** (3.9) | 52.8 (4.6) | **69.5** (4.2) | 66.8 (4.3) | **69.4** (4.8) | **69.2** (4.4) |
| (3) | **58.8** (2.0) | **59.1** (1.9) | 56.6 (2.0) | 57.7 (1.7) | 63.3 (1.8) | 64.2 (2.0) | 64.2 (1.9) | **64.7** (2.1) |
| (4) | **86.2** (2.2) | 85.7 (2.5) | **85.8** (2.3) | 74.1 (3.3) | **86.0** (2.7) | 84.9 (2.4) | **85.6** (2.4) | **86.0** (2.5) |
| (5) | **78.8** (2.2) | **78.8** (1.7) | **78.4** (2.3) | 69.8 (3.7) | **84.3** (2.5) | **84.4** (2.6) | 83.8 (2.3) | **84.4** (2.6) |
| (6) | 94.9 (0.7) | 94.9 (0.8) | **95.2** (0.8) | 75.8 (1.5) | **96.5** (0.6) | **96.6** (0.5) | 96.3 (0.6) | 96.4 (0.5) |
| (7) | 84.9 (0.7) | **85.4** (0.7) | 84.7 (0.7) | 74.9 (0.9) | **91.9** (0.5) | **92.0** (0.6) | **91.9** (0.5) | **91.9** (0.4) |
| (8) | **96.6** (0.6) | 96.5 (0.7) | 96.3 (0.6) | 76.2 (2.2) | 98.7 (0.4) | 98.8 (0.4) | 98.8 (0.3) | **98.9** (0.3) |
| (9) | 96.0 (0.5) | 96.1 (0.5) | **96.3** (0.5) | 92.5 (0.8) | **96.8** (0.5) | 96.6 (0.4) | 96.7 (0.4) | 96.6 (0.4) |
| (10) | **74.1** (3.3) | 72.0 (3.8) | 71.3 (4.3) | 34.0 (6.4) | 83.6 (3.8) | 83.8 (3.4) | **85.0** (3.9) | 83.2 (4.2) |
| (11) | **85.5** (2.9) | **85.9** (2.7) | **85.4** (3.3) | 79.8 (5.6) | **86.0** (3.1) | **85.3** (2.9) | 85.5 (3.3) | 84.4 (2.7) |
| (12) | **64.4** (7.1) | 59.7 (7.8) | **66.3** (6.9) | 58.3 (8.1) | **68.4** (8.6) | **68.1** (6.5) | **66.6** (8.9) | **68.0** (7.2) |
| avg | 81.59 | 81.02 | 81.25 | 68.80 | 85.14 | 84.82 | 85.00 | 84.93 |
| #b | 9 | 6 | 8 | 0 | 9 | 6 | 6 | 7 |

### Linear Kernel

$AL^{01}$: slight benefit

LLW: poor perf.

### Gauss. Kernel

LLW: improved perf.

$AL^{01}$: maintain benefit

# Multiclass Zero-One Classification

| | Fisher Consistent? | Perform well in low dimensional feature? |
|---|---|---|
| 1. The SVM WW Model (Weston et.al., 2002)<br>Relative Margin Model | ✗ | ✓ |
| 2. The SVM CS Model (Crammer and Singer, 1999)<br>Relative Margin Model | ✗ | ✓ |
| 3. The SVM LLW Model (Lee et.al., 2004)<br>Absolute Margin Model | ✓ | ✗ |
| 4. The $AL^{0\text{-}1}$ (Adversarial Surrogate Loss)<br>Relative Margin Model | ✓ | ✓ |

# Other results

General Multiclass Classification

## General Multiclass Classification

1. Zero-One Loss Metric
2. Ordinal Classification with the Absolute Loss Metric
3. Ordinal Classification with the Squared Loss Metric
4. Weighted Multiclass Loss Metrics
5. Classification with Abstention / Reject Option

# Performance-Aligned Graphical Models

Based on:

**Rizal Fathony**, Ashkan Rezaei, Mohammad Bashiri, Xinhua Zhang, Brian D. Ziebart. "*Distributionally Robust Graphical Models*". Advances in Neural Information Processing Systems 31 (NIPS), 2018

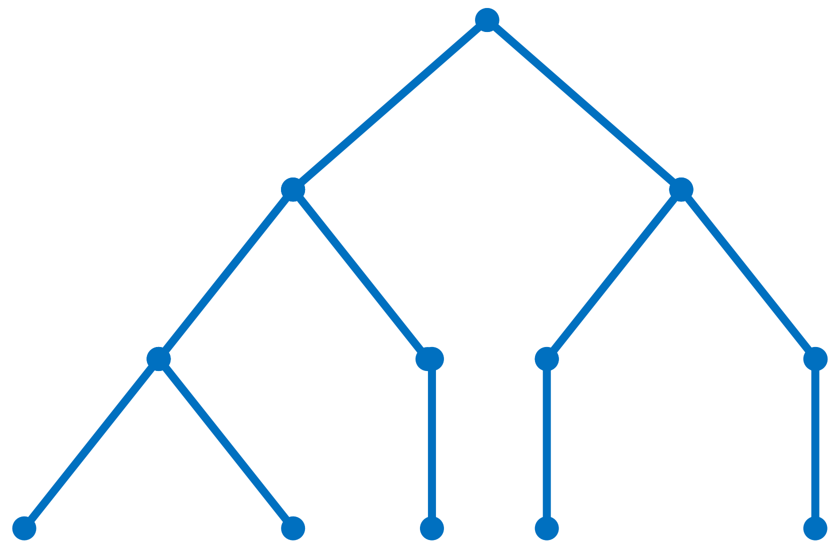
# Conditional Graphical Models

## Some Popular Graphical Structure in Structured Prediction

### Chain Structure

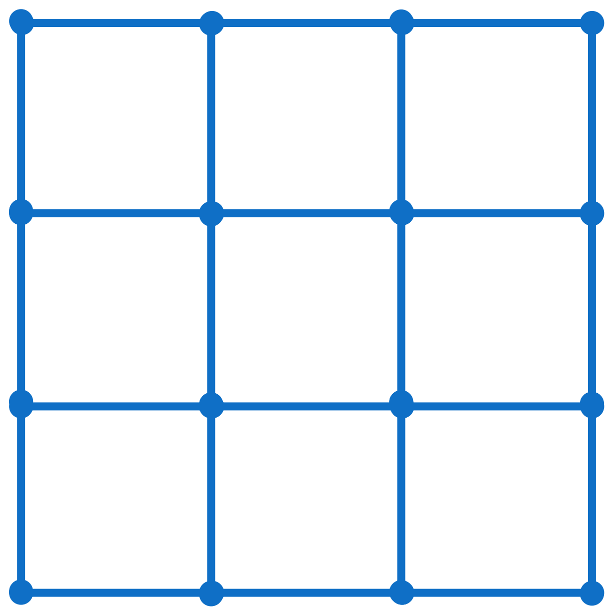
Activity Prediction, Sequence Tagging, NLP tasks: e.g. Named Entity Recognition

### Tree Structure



Parse Tree-Based NLP tasks:
Semantic Role Labeling
and Sentiment Analysis

### Lattice Structure



Computer Vision Tasks:
e.g. Image Segmentation

# Previous Approaches for Conditional Graphical Models

## 1 Conditional Random Fields (CRF)

(Lafferty et. al., 2001)

✓ **Fisher Consistent**

Produce Bayes optimal prediction in ideal case.

✗ **No easy mechanism to incorporate customized loss/performance metrics**

The algorithm optimized the conditional likelihood.
Loss/performance metric-based prediction
can be performed after learning process.

## 2 Structured SVM (SSVM)

(Tsochantaridis et. al., 2005)

✗ **No Fisher consistency guarantee**

Based on Multiclass SVM-CS.
Not consistent for distribution with no majority label.

✓ **Align with the loss/performance metrics**

The algorithm accept customized loss/performance
metric in its optimization objective.

# Adversarial Graphical Models (AGM)

## Primal:

$$\min_{\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \max_{\check{P}(\check{\mathbf{y}}|\mathbf{x})} \mathbb{E}_{\mathbf{X}\sim\tilde{P};\hat{\mathbf{Y}}|\mathbf{X}\sim\hat{P};\check{\mathbf{Y}}|\mathbf{X}\sim\check{P}} \left[ loss(\hat{\mathbf{Y}}, \check{\mathbf{Y}}) \right] \; s.t.: \mathbb{E}_{\mathbf{X}\sim\tilde{P};\check{\mathbf{Y}}|\mathbf{X}\sim\check{P}} \left[ \Phi(\mathbf{X}, \check{\mathbf{Y}}) \right] = \tilde{\Phi}$$

- Feature function $\Phi(\mathbf{X}, \mathbf{Y})$ is additively decomposed over cliques, $\Phi(\mathbf{x}, \mathbf{y}) = \sum_c \phi(\mathrm{x}, \mathrm{y}_c)$
- The loss metric is additively decomposed over each $y_i$ variables, $\mathrm{loss}(\hat{\boldsymbol{y}}, \check{\boldsymbol{y}}) = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{loss}(\hat{\mathrm{y}}_\mathrm{i}, \check{\mathrm{y}}_\mathrm{i})$
- Focus on pairwise graphical models: interactions between label = edges in graphs

## Dual:

$$\min_{\theta_e, \theta_v} \mathbb{E}_{\mathbf{X},\mathbf{Y}\sim\tilde{P}} \max_{\check{P}(\check{\mathbf{y}}|\mathbf{x})} \min_{\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \sum_{\hat{\mathbf{y}},\check{\mathbf{y}}} \hat{P}(\hat{\mathbf{y}}|\mathbf{x}) \check{P}(\check{\mathbf{y}}|\mathbf{x}) \Big[ \textstyle\sum_i^n \mathrm{loss}(\hat{y}_i, \check{y}_i)$$

$$+ \theta_e \cdot \textstyle\sum_{(i,j)\in E} \left[ \phi(\mathbf{x}, \check{y}_i, \check{y}_j) - \phi(\mathbf{x}, y_i, y_j) \right] + \theta_v \cdot \sum_i^n \left[ \phi(\mathbf{x}, \check{y}_i) - \phi(\mathbf{x}, y_i) \right] \Big]$$

$\theta_e$: Lagrange multipliers for constraints with edge features
$\theta_v$: Lagrange multipliers for constraints with node features

size:
$k^n \times k^n$

Intractable
for modestly-sized $n$

# AGM | Marginal Formulation

## Dual:

$$\min_{\theta_e,\theta_v} \mathbb{E}_{\mathbf{X},\mathbf{Y}\sim\tilde{P}} \max_{\check{P}(\check{\mathbf{y}}|\mathbf{x})} \min_{\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \sum_{\hat{\mathbf{y}},\check{\mathbf{y}}} \hat{P}(\hat{\mathbf{y}}|\mathbf{x})\check{P}(\check{\mathbf{y}}|\mathbf{x}) \Big[ \textstyle\sum_i^n \text{loss}(\hat{y}_i, \check{y}_i)$$
$$+ \theta_e \cdot \textstyle\sum_{(i,j)\in E} [\phi(\mathbf{x}, \check{y}_i, \check{y}_j) - \phi(\mathbf{x}, y_i, y_j)] + \theta_v \cdot \textstyle\sum_i^n [\phi(\mathbf{x}, \check{y}_i) - \phi(\mathbf{x}, y_i)] \Big]$$

## Dual | Marginal Formulation:

$$\min_{\theta_e,\theta_v} \mathbb{E}_{\mathbf{X},\mathbf{Y}\sim\tilde{P}} \max_{\check{P}(\check{\mathbf{y}}|\mathbf{x})} \min_{\hat{P}(\hat{\mathbf{y}}|\mathbf{x})} \Big[ \textstyle\sum_i^n \sum_{\hat{y}_i,\check{y}_i} \hat{P}(\hat{y}_i|\mathbf{x})\check{P}(\check{y}_i|\mathbf{x}) loss(\hat{y}_i, \check{y}_i)$$
$$+ \textstyle\sum_{(i,j)\in E} \sum_{\check{y}_i,\check{y}_j} \check{P}(\check{y}_i, \check{y}_j|\mathbf{x}) [\theta_e \cdot \phi(\mathbf{x}, \check{y}_i, \check{y}_j)] - \sum_{(i,j)\in E} \theta_e \cdot \phi(\mathbf{x}, y_i, y_j)$$
$$+ \textstyle\sum_i^n \sum_{\check{y}_i} \check{P}(\check{y}_i|\mathbf{x}) [\theta_v \cdot \phi(\mathbf{x}, \check{y}_i)] - \sum_i^n \theta_v \cdot \phi(\mathbf{x}, y_i) \Big],$$

Similar to CRF and SSVM:
General Graphical Models:
Intractable

Focus:
Graphs with low tree-width,
e.g.: chain, tree, simple loops.
Tractable optimization

The objective depends on $\hat{P}(\hat{\mathbf{y}}|\mathbf{x})$ only through its node marginal probability $\hat{P}(\hat{y}_i|\mathbf{x})$
The objective depends on $\check{P}(\check{\mathbf{y}}|\mathbf{x})$ only through its node and edge marginal probability $\check{P}(\check{y}_i|\mathbf{x})$ and $\check{P}(\check{y}_i, \check{y}_j|\mathbf{x})$

# AGM | Optimization

## Matrix Notation (Tree Structure AGM):

$$\min_{\theta_e,\theta_v} \mathbb{E}_{\mathbf{X},\mathbf{Y}\sim\tilde{P}} \max_{\mathbf{Q}} \min_{\mathbf{p}} \sum_i^n \Big[\mathbf{p}_i \mathbf{L}_i(\mathbf{Q}_{pt(i);i}^{\mathrm{T}}\mathbf{1}) + \Big\langle \mathbf{Q}_{pt(i);i} - \mathbf{Z}_{pt(i);i}, \textstyle\sum_l \theta_e^{(l)} \mathbf{W}_{pt(i);i;l} \Big\rangle$$
$$+ (\mathbf{Q}_{pt(i);i}^{\mathrm{T}}\mathbf{1} - \mathbf{z}_i)^{\mathrm{T}}(\textstyle\sum_l \theta_v^{(l)} \mathbf{w}_{i;l})\Big]$$
$$\text{subject to: } \mathbf{Q}_{pt(pt(i));pt(i)}^{\mathrm{T}}\mathbf{1} = \mathbf{Q}_{pt(i);i}\mathbf{1}, \ \forall i \in \{1,\ldots,n\},$$

## Optimization Techniques:

- Stochastic (sub)-gradient descent
  (outer optimization for $\theta_e$ and $\theta_v$)
- Dual decomposition (inner $\mathbf{Q}$ optimization)
- Discrete optimal transport solver (recovering $\mathbf{Q}$)
- Closed-form solution (inner $\mathbf{p}$ optimization)

## Runtime (for a single subgradient update):

- Depends on the loss metric used
- For the additive zero-one loss (Hamming loss)

$O(nlk \log k + nk^2)$

$k$: # classes, $n$: # nodes,

$l$: # iterations in dual decomposition

| CRF | SSVM |
|---|---|
| $O(nk^2)$ | $O(nk^2)$ |

## General graphs low tree-width

$O\big(nlwk^{(w+1)} \log k + nk^{2(w+1)}\big)$

$n$: # cliques, $w$: treewidth of the graph

# AGM | Consistency

If the loss function is additive

## AGM is consistent

when $f$ is optimized over all measurable functions on the input space

## AGM is also consistent

when $f$ is optimized over a restricted set of functions:

all measurable function that are additive over the edge and node potentials.

# AGM | Experiments (1)

## Facial Emotion Intensity Prediction (Chain Structure, Labels with Ordinal Category)

- Each node: 3 class classification: *neutral = 1< increasing = 2 < apex = 3*
- 167 sequences
- Ordinal loss metrics: zero-one loss, absolute loss, and squared loss
- Weighted and unweighted. Weights reflect the focus of prediction (e.g. focus more on latest nodes)

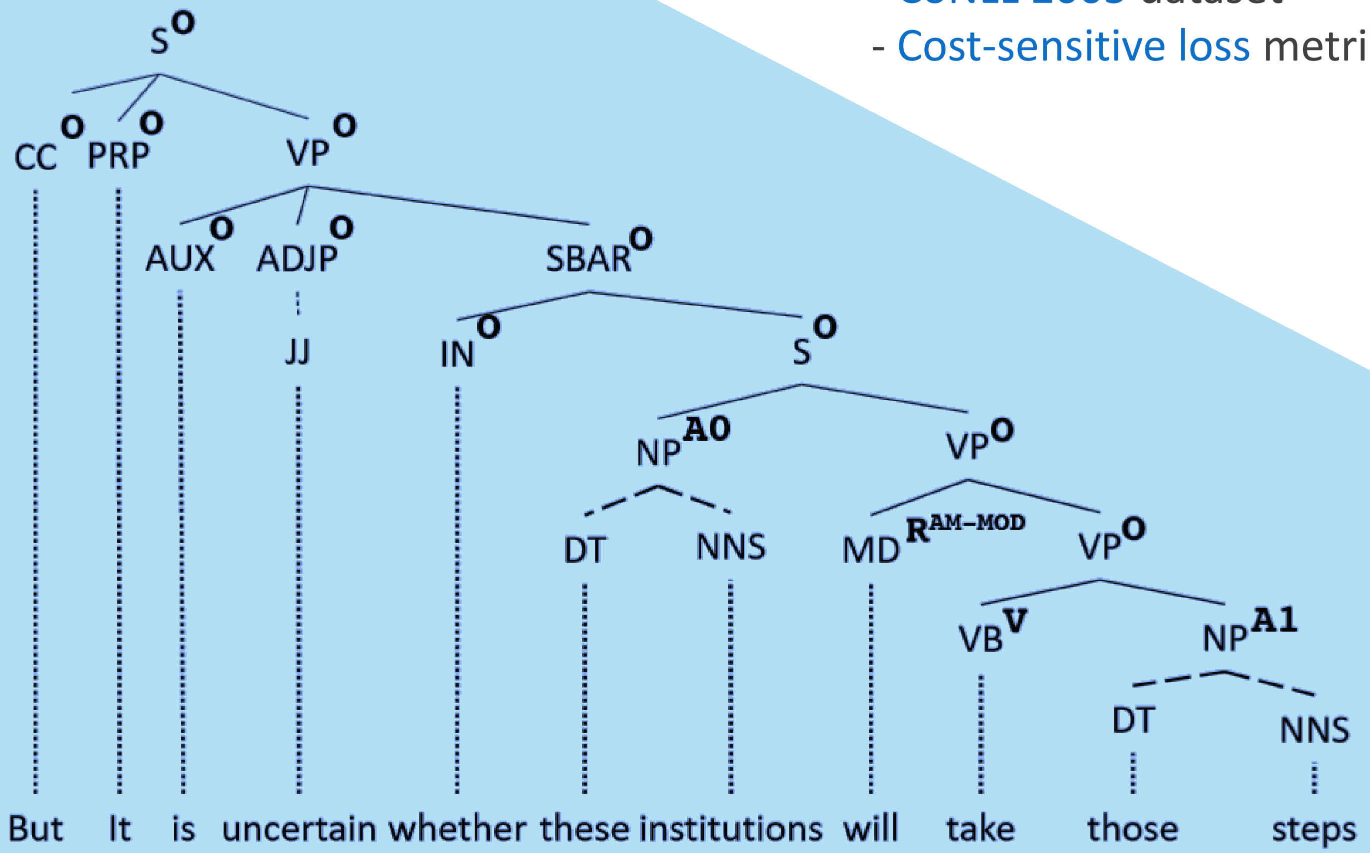
Results: The **mean (standard deviation)** of the average **loss metrics.**
Bold numbers: **best** or **not significantly worse** than the best

| Loss metrics | AGM | CRF | SSVM |
|---|---|---|---|
| zero-one, unweighted | 0.34 | **0.32** | 0.37 |
| absolute, unweighted | **0.33** | 0.34 | 0.40 |
| quadratic, unweighted | **0.38** | **0.38** | 0.40 |
| zero-one, weighted | **0.28** | 0.32 | 0.29 |
| absolute, weighted | **0.29** | 0.36 | **0.29** |
| quadratic, weighted | 0.36 | 0.40 | **0.33** |
| average | 0.33 | 0.35 | 0.35 |
| # bold | 4 | 2 | 2 |

# AGM | Experiments (2)

## Semantic Role Labeling (Tree Structure)

- Predict label of each node given known parse tree.
- CoNLL 2005 dataset
- Cost-sensitive loss metric is used reflect the importance of each label




### Results:

Table 2: The average loss metrics for the semantic role labeling task.

| Loss metrics | AGM | CRF | SSVM |
|---|---|---|---|
| cost-sensitive loss | 0.14 | 0.19 | 0.14 |

# Conditional Graphical Models

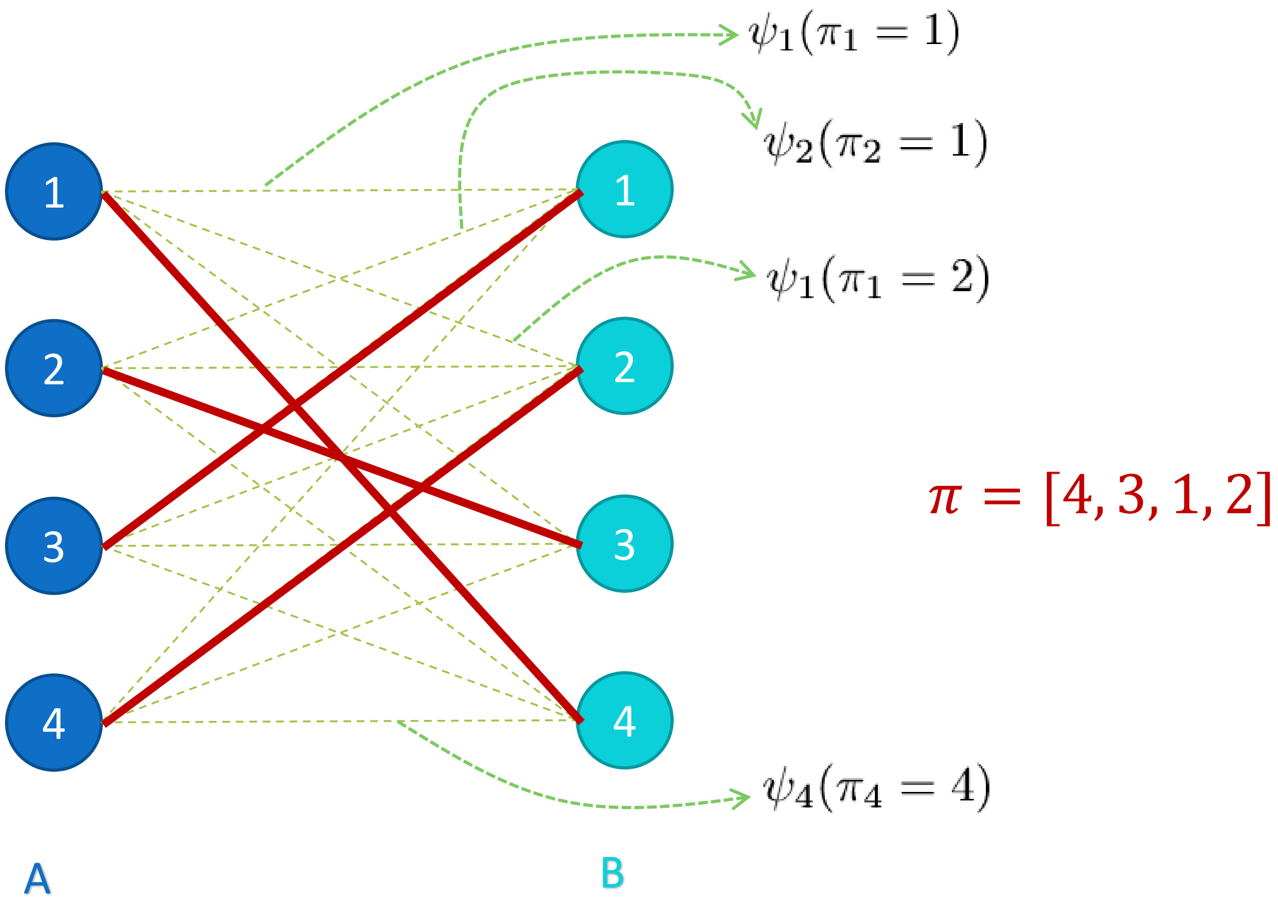
| | Performance-Aligned? | Consistent? |
|---|---|---|
| Conditional Random Field (CRF)<br>(Lafferty et. al., 2001) | ✗ | ✓ |
| Structured SVM<br>(Tsochantaridis et. al., 2005) | ✓ | ✗ |
| Adversarial Graphical Models<br>(our approach) | ✓ | ✓ |

# Bipartite Matching in Graphs

Based on:

**Rizal Fathony***, Sima Behpour*, Xinhua Zhang, Brian D. Ziebart. “*Efficient and Consistent Adversarial Bipartite Matching*”. International Conference on Machine Learning (ICML), 2018.

# Bipartite Matching Task




Maximum weighted bipartite matching:

$$\max_{\pi \in \Pi} \psi(\pi) = \max_{\pi \in \Pi} \sum_i \psi_i(\pi_i)$$

Machine learning task:

Learn the appropriate weights $\psi_i(\cdot)$

Objective:

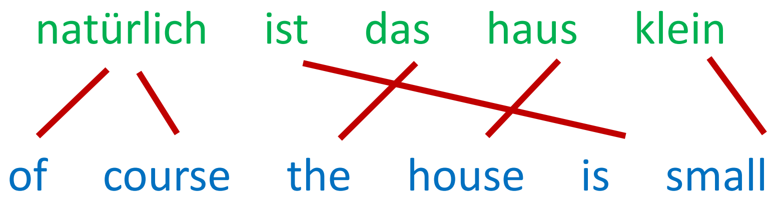
Minimize a loss metric, e.g., the Hamming loss

$$\text{loss}_{\text{Ham}}(\pi, \pi') = \sum_{i=1}^{n} 1(\pi'_i \neq \pi_i)$$

# Learning Bipartite Matching | Applications

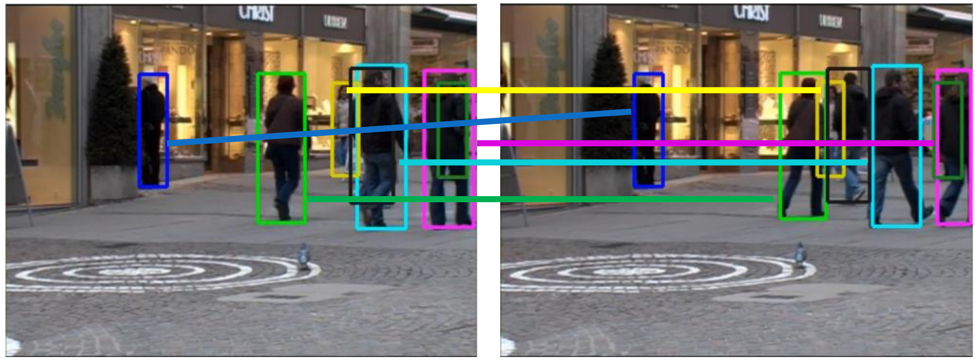
## ❶ Word alignment

(Taskar et. al., 2005; Pado & Lapta, 2006; Mac-Cartney et. al., 2008)



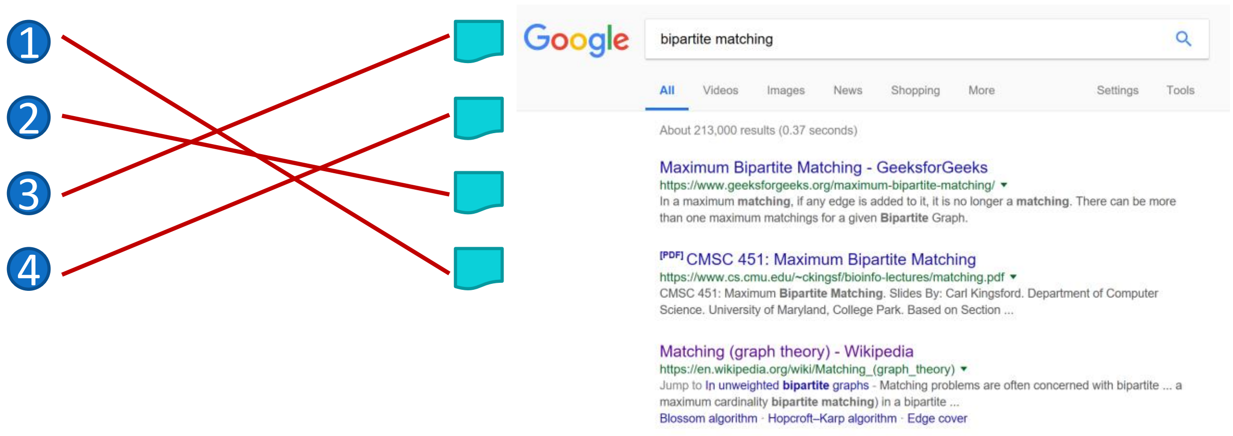

## ❷ Correspondence between images

(Belongie et. al., 2002; Dellaert et. al., 2003)



## ❸ Learning to rank documents

(Dwork et. al., 2001; Le & Smola, 2007)




A non-bipartite matching task can be converted to a bipartite matching problem

# Previous Approaches for Bipartite Matching

## 1 CRF (Petterson et. al., 2009; Volkovs & Zemel, 2012)

$$P_\psi(\pi) = \frac{1}{Z_\psi} \exp\left(\sum_{i=1}^{n} \psi_i(\pi_i)\right)$$

$$Z_\psi = \sum_{\pi} \prod_{i=1}^{n} \exp\left(\psi_i(\pi_i)\right) = \text{perm}(\mathbf{M})$$

$$\text{where} M_{i,j} = \exp\left(\psi_i(j)\right)$$

**✓ Fisher Consistent**

Produce Bayes optimal prediction in ideal case

**✗ Computationally intractable**

Normalization term requires matrix permanent computation (a #P-hard problem).

Approximation is needed for modestly sized problems.

## 2 Structured SVM (Tsochantaridis et. al., 2005)

solved using constraint generation

$$\min_{\psi} \mathbb{E}_{\pi \sim \tilde{P}} \left[\max_{\pi'} \{\text{loss}(\pi, \pi') + \psi(\pi')\} - \psi(\pi)\right]$$

$\tilde{P}$ is the empirical distribution

**✓ Computationally Efficient**

Hungarian algorithm for computing the maximum violated constraints

**✗ No Fisher consistency guarantee**

Based on Multiclass SVM-CS

Not consistent for distribution with no majority label

# Adversarial Bipartite Matching (our approach)

Primal:

$$\min_{\hat{P}(\hat{\pi}|x)} \max_{\check{P}(\check{\pi}|x)} \mathbb{E}_{x\sim\tilde{P};\hat{\pi}|x\sim\hat{P};\check{\pi}|x\sim\check{P}}[\text{loss}(\hat{\pi},\check{\pi})]$$

$$\text{s.t. } \mathbb{E}_{x\sim\tilde{P};\check{\pi}|x\sim\check{P}}\left[\sum_{i=1}^{n}\phi_i(x,\check{\pi}_i)\right] = \mathbb{E}_{(x,\pi)\sim\tilde{P}}\left[\sum_{i=1}^{n}\phi_i(x,\pi_i)\right]$$

Augmented Hamming loss matrix for $n = 3$ permutations

| | $\check{\pi}=123$ | $\check{\pi}=132$ | $\check{\pi}=213$ | $\check{\pi}=231$ | $\check{\pi}=312$ | $\check{\pi}=321$ |
|---|---|---|---|---|---|---|
| $\hat{\pi}=123$ | $0+\delta_{123}$ | $2+\delta_{132}$ | $2+\delta_{213}$ | $3+\delta_{231}$ | $3+\delta_{312}$ | $2+\delta_{321}$ |
| $\hat{\pi}=132$ | $2+\delta_{123}$ | $0+\delta_{132}$ | $3+\delta_{213}$ | $2+\delta_{231}$ | $2+\delta_{312}$ | $3+\delta_{321}$ |
| $\hat{\pi}=213$ | $2+\delta_{123}$ | $3+\delta_{132}$ | $0+\delta_{213}$ | $2+\delta_{231}$ | $2+\delta_{312}$ | $3+\delta_{321}$ |
| $\hat{\pi}=231$ | $3+\delta_{123}$ | $2+\delta_{132}$ | $2+\delta_{213}$ | $0+\delta_{231}$ | $3+\delta_{312}$ | $2+\delta_{321}$ |
| $\hat{\pi}=312$ | $3+\delta_{123}$ | $2+\delta_{132}$ | $2+\delta_{213}$ | $3+\delta_{231}$ | $0+\delta_{312}$ | $2+\delta_{321}$ |
| $\hat{\pi}=321$ | $2+\delta_{123}$ | $3+\delta_{132}$ | $3+\delta_{213}$ | $2+\delta_{231}$ | $2+\delta_{312}$ | $0+\delta_{321}$ |

Dual:

$$\min_{\theta} \mathbb{E}_{x,\pi\sim\tilde{P}} \min_{\hat{P}(\hat{\pi}|x)} \max_{\check{P}(\check{\pi}|x)} \mathbb{E}_{\substack{\hat{\pi}|x\sim\hat{P} \\ \check{\pi}|x\sim\check{P}}}\left[\text{loss}(\hat{\pi},\check{\pi}) + \theta\cdot\sum_{i=1}^{n}(\phi_i(x,\check{\pi}_i) - \phi_i(x,\pi_i))\right]$$

Hamming loss

Lagrangian term $\delta$

size:

$n! \times n!$

<u>Intractable</u>

for modestly-sized $n$

# Polytope of the Permutation Mixtures

**Dual:**

$$\min_{\theta} \mathbb{E}_{(x,\pi)\sim\tilde{P}} \min_{\hat{P}(\hat{\pi}|x)} \max_{\check{P}(\check{\pi}|x)} \mathbb{E}_{\hat{\pi}|x\sim\hat{P};\check{\pi}|x\sim\check{P}} \left[ \sum_{i=1}^{n} I(\pi'_i \neq \pi_i) + \theta \cdot \sum_{i=1}^{n} (\phi_i(x,\check{\pi}_i) - \phi_i(x,\pi_i)) \right]$$

## Marginal Distribution Matrices:

**Predictor**

$\mathbf{P} =$

| | 1 | 2 | 3 |
|---|---|---|---|
| $\hat{\pi}_1$ | $p_{1,1}$ | $p_{1,2}$ | $p_{1,3}$ |
| $\hat{\pi}_2$ | $p_{2,1}$ | $p_{2,2}$ | $p_{2,3}$ |
| $\hat{\pi}_3$ | $p_{3,1}$ | $p_{3,2}$ | $p_{3,3}$ |

$p_{i,j} = \hat{P}(\hat{\pi}_i = j)$

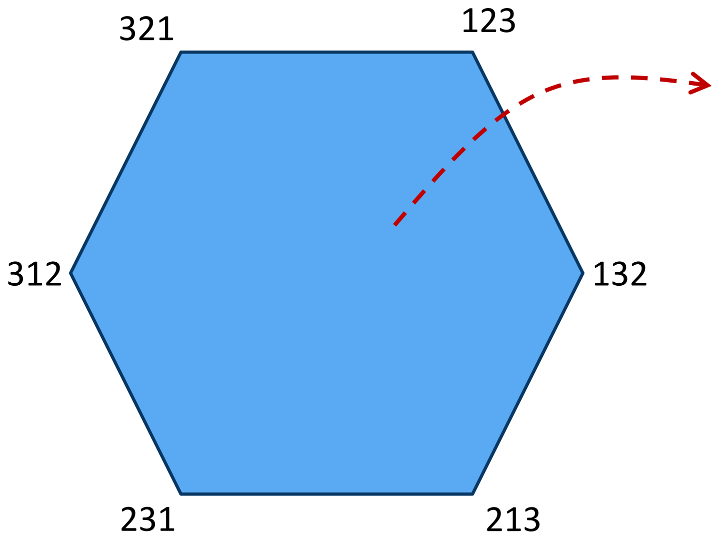
**Adversary**

$\mathbf{Q} =$

| | 1 | 2 | 3 |
|---|---|---|---|
| $\check{\pi}_1$ | $q_{1,1}$ | $q_{1,2}$ | $q_{1,3}$ |
| $\check{\pi}_2$ | $q_{2,1}$ | $q_{2,2}$ | $q_{2,3}$ |
| $\check{\pi}_3$ | $q_{3,1}$ | $q_{3,2}$ | $q_{3,3}$ |

$q_{i,j} = \check{P}\,(\check{\pi}_i = j)$

## Birkhoff – Von Neumann theorem:




convex polytope whose points are doubly stochastic matrices

$$\mathbf{P1} = \mathbf{P}^{\top}\mathbf{1} = \mathbf{Q1} = \mathbf{Q}^{\top}\mathbf{1} = \mathbf{1}$$

reduce the space of optimization: from $O(n!)$ to $O(n^2)$

# Marginal Distribution Formulation

## Dual:

$$\min_{\theta} \mathbb{E}_{(x,\pi)\sim\tilde{P}} \min_{\hat{P}(\hat{\pi}|x)} \max_{\check{P}(\check{\pi}|x)} \mathbb{E}_{\hat{\pi}|x\sim\hat{P};\check{\pi}|x\sim\check{P}} \left[ \sum_{i=1}^{n} I(\pi_i' \neq \pi_i) + \theta \cdot \sum_{i=1}^{n} (\phi_i(x,\check{\pi}_i) - \phi_i(x,\pi_i)) \right]$$

## Marginal Formulation:

Rearrange the optimization order and add regularization and smoothing penalties

$$\max_{\mathbf{Q}\geq\mathbf{0}} \min_{\theta} \frac{1}{m} \sum_{i=1}^{m} \min_{\mathbf{P}_i\geq\mathbf{0}} \left[ \langle \mathbf{Q}_i - \mathbf{Y}_i, \textstyle\sum_k \theta_k \mathbf{X}_{i,k} \rangle - \langle \mathbf{P}_i, \mathbf{Q}_i \rangle + \frac{\mu}{2}\|\mathbf{P}_i\|_F^2 - \frac{\mu}{2}\|\mathbf{Q}_i\|_F^2 \right] + \frac{\lambda}{2}\|\theta\|_2^2$$

$$\text{s.t.}: \mathbf{P}_i\mathbf{1} = \mathbf{P}_i^\top\mathbf{1} = \mathbf{Q}_i\mathbf{1} = \mathbf{Q}_i^\top\mathbf{1} = \mathbf{1}, \quad \forall i$$

## Optimization Techniques Used:

- Outer (Q) : projected Quasi-Newton (Schmidt, et.al., 2009)
- Inner ($\theta$) : closed-form solution
- Inner (P) : projection to doubly-stochastic matrix
- Projection to doubly-stochastic matrix : ADMM

# Consistency

## Empirical Risk Perspective of Adversarial Bipartite Matching

$$\min_{\theta} \mathbb{E}_{\substack{x \sim P \\ \pi | x \sim \tilde{P}}} \left[ AL^{\text{perm}}_{f_\theta}(x, \pi) \right]$$

$$\text{where: } AL^{\text{perm}}_{f_\theta}(x, \pi) \triangleq \min_{\hat{P}(\hat{\pi}|x)} \max_{\check{P}(\check{\pi}|x)} \mathbb{E}_{\substack{\hat{\pi}|x \sim \hat{P} \\ \check{\pi}|x \sim \check{P}}} \left[ \text{loss}(\hat{\pi}, \check{\pi}) + f_\theta(x, \check{\pi}) - f_\theta(x, \pi) \right]$$

## $\text{AL}^{\text{perm}}$ is consistent

when $f$ is optimized over all measurable functions on the input space $(x, \pi)$

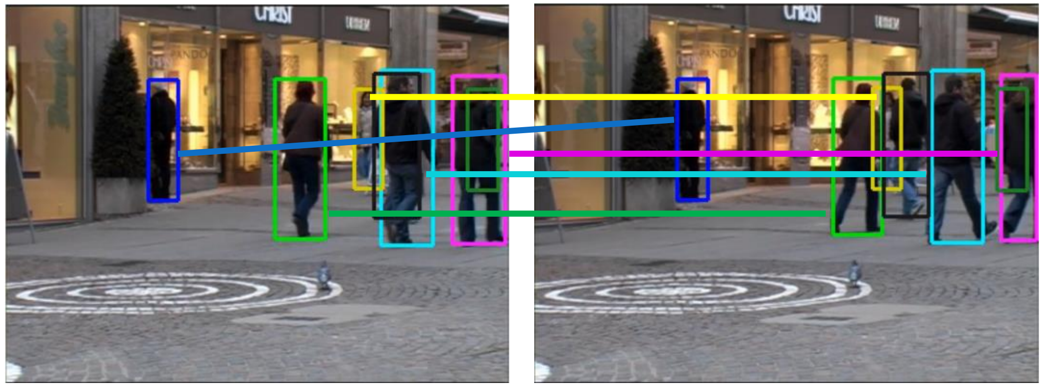
## $\text{AL}^{\text{perm}}$ is also consistent

$f$ is optimized over a restricted set of functions: $f(x, \pi) = \sum_i g_i(x, \pi_i)$

when $g$ is allowed to be optimized over all measurable functions on the individual input space $(x, \pi_i)$

# Experiments

## Application: Video Tracking



## Public Benchmark Datasets

*Table 3.* Dataset properties

| DATASET | # ELEMENTS | # EXAMPLES |
|---|---|---|
| TUD-CAMPUS | 12 | 70 |
| TUD-STADTMITTE | 16 | 178 |
| ETH-SUNNYDAY | 18 | 353 |
| ETH-BAHNHOF | 34 | 999 |
| ETH-PEDCROSS2 | 30 | 836 |

## Empirical runtime (until convergence)

*Table 5.* Running time (in seconds) of the model for various number of elements $n$ with fixed number of samples ($m = 50$)

| DATASET | # ELEMENTS | | ADV MARG. | | SSVM |
|---|---|---|---|---|---|
| CAMPUS | 12 | 1.0 | 1.96 | 1.0 | 0.22 |
| STADTMITTE | 16 | 1.3 | 2.46 | 1.2 | 0.25 |
| SUNNYDAY | 18 | 1.5 | 2.75 | 1.4 | 0.15 |
| PEDCROSS2 | 30 | 2.5 | 8.18 | 4.2 | 0.26 |
| BAHNHOF | 34 | 2.8 | 9.79 | 5.0 | 0.31 |

relative: 12=1.0 relative: 1.96=1.0

Adversarial. Marginal Formulation:
grows (roughly) quadratically in $n$

CRF: impractical even for $n = 20$
(Petterson et. al., 2009)

# Experiment Results

Table 1: The mean and standard deviation (in parenthesis) of the average accuracy (1 - the average Hamming loss) for the adversarial bipartite matching model compared with Structured-SVM.

| Training/ Testing | Adv. Bipartite Matching | Structured SVM |
|---|---|---|
| Campus/ Stadtmitte | 0.662 (0.08) | 0.662 (0.08) |
| Stadtmitte/ Campus | 0.667 (0.11) | 0.660 (0.12) |
| Bahnhof/ Sunnyday | **0.754** (0.10) | 0.729 (0.15) |
| Pedcross2/ Sunnyday | **0.750** (0.10) | 0.736 (0.13) |
| Sunnyday/ Bahnhof | **0.751** (0.18) | 0.739 (0.20) |
| Pedcross2/ Bahnhof | **0.763** (0.16) | 0.731 (0.21) |
| Bahnhof/ Pedcross2 | **0.714** (0.16) | 0.701 (0.18) |
| Sunnyday/ Pedcross2 | **0.712** (0.17) | 0.700 (0.18) |

6 pairs of dataset

significantly outperforms SSVM

2 pairs of dataset

competitive with SSVM

# Bipartite Matching in Graphs

| | Efficient? | Consistent? | Perform well? |
|---|---|---|---|
| Conditional Random Field (CRF) (Petterson et. al., 2009; Volkovs & Zemel, 2012) | ✗ | ✓ | ? |
| Structured SVM (Tsochantaridis et. al., 2005) | ✓ | ✗ | – |
| Adversarial Bipartite Matching (our approach) | ✓ | ✓ | ✓ |

# Conclusion

# Robust Adversarial Learning Algorithms

- ✓ Align better with the loss/performance metric
  (by incorporating the metric into its learning objective)
- ✓ Provide Fisher consistency guarantee
- ✓ Computationally efficient
- ✓ Perform well in practice

# Future Directions

# Future Directions (1)

## 1. Fairness in Machine Learning

Important issues in automated decision using ML algorithms.

Requires the algorithm to produce fair prediction.

Our formulation only enforces constraints on the adversary.

$$\min_{\hat{P}(\hat{Y}|\mathbf{X})} \max_{\check{P}(\check{Y}|\mathbf{X})} \mathbb{E}_{\mathbf{X},Y\sim\tilde{P};\hat{Y}|\mathbf{X}\sim\hat{P};\check{Y}|\mathbf{X}\sim\check{P}} \left[\text{loss}(\hat{Y},\check{Y})\right]$$

$$\text{s.t. } \mathbb{E}_{\mathbf{X}\sim\tilde{P};\check{Y}|\mathbf{X}\sim\check{P}}[\phi(\mathbf{X},\check{Y})] = \mathbb{E}_{\mathbf{X},Y\sim\tilde{P}}[\phi(\mathbf{X},Y)]$$

Add fairness constraints to the predictor?

## 2. Statistical Theory of Loss Functions

In multiclass classification problem, both $AL^{0\text{-}1}$ and SVM-LLW are Fisher consistent. However, their performances are quite different.

Is there any stronger statistical guarantee that can separate the high-performing Fisher consistent algorithm from the low-performing ones?

# Future Directions (2)

## 3. Structured Prediction & Graphical Models

More complex graphical structures are popular in some applications, e.g. computer vision.

Exact learning algorithms for AGM in this case may be intractable.

Can we develop learning algorithms for general graphical models?

What kind of approximation algorithms can be applicable?

## 4. Deep Learning

Deep learning has been successfully applied to many prediction problems.

Most of deep learning architectures are not designed to optimize customized loss metrics.

How can the robust adversarial learning approach help designing deep learning architectures?

# Collaborators

AL
Anqi Liu

KA
Kaiser Asif

BZ
Prof. Brian Ziebart

MB
Mohammad Bashiri

SB
Sima Behpour

XZ
Prof. Xinhua Zhang

AR
Ashkan Rezaei

WX
Wei Xing

# Publications

- **Consistent Robust Adversarial Prediction for General Multiclass Classification**
  **Rizal Fathony**, Kaiser Asif, Anqi Liu, Mohammad Bashiri, Wei Xing, Sima Behpour, Xinhua Zhang, Brian D. Ziebart.
  Submitted to JMLR.

- **Distributionally Robust Graphical Models**
  **Rizal Fathony**, Ashkan Rezaei, Mohammad Bashiri, Xinhua Zhang, Brian D. Ziebart.
  Advances in Neural Information Processing Systems 31 (NeurIPS), 2018.

- **Efficient and Consistent Adversarial Bipartite Matching**
  **Rizal Fathony***, Sima Behpour*, Xinhua Zhang, Brian D. Ziebart.
  International Conference on Machine Learning (ICML), 2018.

- **Adversarial Surrogate Losses for Ordinal Regression**
  **Rizal Fathony**, Mohammad Bashiri, Brian D. Ziebart.
  Advances in Neural Information Processing Systems 30 (NIPS), 2017.

- **Adversarial Multiclass Classification: A Risk Minimization Perspective**
  **Rizal Fathony**, Anqi Liu, Kaiser Asif, Brian D. Ziebart.
  Advances in Neural Information Processing Systems 29 (NIPS), 2016.

- **Kernel Robust Bias-Aware Prediction under Covariate Shift**
  Anqi Liu, **Rizal Fathony**, Brian D. Ziebart. ArXiv Preprints, 2016.

# Thank You