Multi-Task Discriminative Estimation for Generative Models and Probabilities

Tony Jebara
Columbia University

May 25, 2010

1. Margin Constraints in Generative Learning
2. Maximum Entropy Discrimination
3. Feature & Kernel Sparsity
4. Multi-Task Margin Constraints
5. Sequential Quadratic Programming
6. Relative Margin Constraints
7. Conclusions

Generative Learning

- Given $\mathcal{D} = (\mathbf{x}_t, y_t)_{t=1}^{T}$ sampled *iid* from unknown $P(\mathbf{x}, y)$
- Find rule producing $\hat{y}$ from $\mathbf{x}$ with low error

Generative Learning

- Given $\mathcal{D} = (\mathbf{x}_t, y_t)_{t=1}^T$ sampled *iid* from unknown $P(\mathbf{x}, y)$
- Find rule producing $\hat{y}$ from $\mathbf{x}$ with low error
- Generative Bayesian approach:
    - Assume $p(\mathbf{x}, y|\Theta)$ and $p(\Theta)$
    - Get $p(\Theta|\mathcal{D}) \propto \prod_{t=1}^T p(\mathbf{x}_t, y_t|\Theta)p(\Theta)$
    - Predict $\hat{y} = \arg\max_y \int_\Theta p(\mathbf{x}, y|\Theta)p(\Theta|\mathcal{D})$

Generative Learning

- Given $\mathcal{D} = (\mathbf{x}_t, y_t)_{t=1}^T$ sampled *iid* from unknown $P(\mathbf{x}, y)$
- Find rule producing $\hat{y}$ from $\mathbf{x}$ with low error
- Generative Bayesian approach:
    - Assume $p(\mathbf{x}, y|\Theta)$ and $p(\Theta)$
    - Get $p(\Theta|\mathcal{D}) \propto \prod_{t=1}^T p(\mathbf{x}_t, y_t|\Theta)p(\Theta)$
    - Predict $\hat{y} = \arg\max_y \int_\Theta p(\mathbf{x}, y|\Theta)p(\Theta|\mathcal{D})$
- Conditional Bayesian approach:
    - Assume $p(y|\mathbf{x}, \Theta)$ and $p(\Theta)$
    - Get $p(\Theta|\mathcal{D}) \propto \prod_{t=1}^T p(y_t|\mathbf{x}_t, \Theta)p(\Theta)$
    - Predict $\hat{y} = \arg\max_y \int_\Theta p(y|\mathbf{x}, \Theta)p(\Theta|\mathcal{D})$

Generative Learning

- Given $\mathcal{D} = (\mathbf{x}_t, y_t)_{t=1}^{T}$ sampled *iid* from unknown $P(\mathbf{x}, y)$
- Find rule producing $\hat{y}$ from $\mathbf{x}$ with low error
- Generative Bayesian approach:
  - Assume $p(\mathbf{x}, y|\Theta)$ and $p(\Theta)$
  - Get $p(\Theta|\mathcal{D}) \propto \prod_{t=1}^{T} p(\mathbf{x}_t, y_t|\Theta)p(\Theta)$
  - Predict $\hat{y} = \arg\max_y \int_\Theta p(\mathbf{x}, y|\Theta)p(\Theta|\mathcal{D})$
- Conditional Bayesian approach:
  - Assume $p(y|\mathbf{x}, \Theta)$ and $p(\Theta)$
  - Get $p(\Theta|\mathcal{D}) \propto \prod_{t=1}^{T} p(y_t|\mathbf{x}_t, \Theta)p(\Theta)$
  - Predict $\hat{y} = \arg\max_y \int_\Theta p(y|\mathbf{x}, \Theta)p(\Theta|\mathcal{D})$
- Problem: high train & test error if assumptions were wrong!
- Solution: add margin constraints so correct y_t wins by γ

Discriminating with Margin Constraints

Conditional Bayes, $\hat{y} = \arg\max_y \int_\Theta q(\Theta) \quad p(y|\mathbf{x}, \Theta)$ via

$$\min_{q(\Theta)} \mathcal{KL}\left(q(\Theta) \| \frac{1}{Z} \prod_{t=1}^{T} p(y_t|\mathbf{x}_t, \Theta) p(\Theta)\right)$$

Discriminating with Margin Constraints

Conditional Bayes, $\hat{y} = \arg\max_y \int_\Theta q(\Theta) \quad p(y|\mathbf{x}, \Theta)$ via

$$\min_{q(\Theta)} \mathcal{KL}\left(q(\Theta) \| \frac{1}{Z} \prod_{t=1}^{T} p(y_t|\mathbf{x}_t, \Theta) p(\Theta)\right)$$

$$s.t. \int_\Theta q(\Theta) \quad p(y_t|\mathbf{x}_t, \Theta) \geq \max_{y \neq y_t} \int_\Theta q(\Theta) \quad p(y|\mathbf{x}_t, \Theta) + \gamma \qquad \forall t$$

Discriminating with Margin Constraints

Log Conditional Bayes, $\hat{y} = \arg\max_y \int_\Theta q(\Theta) \ln p(y|\mathbf{x}, \Theta)$ via

$$\min_{q(\Theta)} \mathcal{KL}\left(q(\Theta) \| \frac{1}{Z} \prod_{t=1}^{T} p(y_t|\mathbf{x}_t, \Theta) p(\Theta)\right)$$

$$s.t. \int_\Theta q(\Theta) \ln p(y_t|\mathbf{x}_t, \Theta) \geq \max_{y \neq y_t} \int_\Theta q(\Theta) \ln p(y|\mathbf{x}_t, \Theta) + \gamma \qquad \forall t$$

Discriminating with Slackened Margin Constraints

Log Conditional Bayes, $\hat{y} = \arg\max_y \int_\Theta q(\Theta) \ln p(y|\mathbf{x}, \Theta)$ via

$$\min_{q(\Theta)} \mathcal{KL}\left(q(\Theta) \| \frac{1}{Z} \prod_{t=1}^{T} p(y_t|\mathbf{x}_t, \Theta) p(\Theta)\right) + C \sum_{t=1}^{T} \xi_t$$

$$s.t. \int_\Theta q(\Theta) \ln p(y_t|\mathbf{x}_t, \Theta) \geq \max_{y \neq y_t} \int_\Theta q(\Theta) \ln p(y|\mathbf{x}_t, \Theta) + \gamma - \xi_t \;\; \forall t$$

Discriminating with Slackened Margin Constraints

Log Conditional Bayes, $\hat{y} = \arg\max_y \int_\Theta q(\Theta) \ln p(y|\mathbf{x}, \Theta)$ via

$$\min_{q(\Theta)} \mathcal{KL}\left(q(\Theta) \| \frac{1}{Z} \prod_{t=1}^{T} p(y_t|\mathbf{x}_t, \Theta) p(\Theta) \right) + C \sum_{t=1}^{T} \xi_t$$

$$s.t. \int_\Theta q(\Theta) \ln p(y_t|\mathbf{x}_t, \Theta) \geq \max_{y \neq y_t} \int_\Theta q(\Theta) \ln p(y|\mathbf{x}_t, \Theta) + \gamma - \xi_t \;\; \forall t$$

Easy, Maximum Entropy Discrimination (Jaakkola Meila Jebara 99)
Slack allows some misclassification of training data

Primal and Dual MED

Primal

$$\min_{q(\Theta)} \mathcal{KL}\left(q(\Theta)\|\hat{p}(\Theta)\right) + C\sum_{t=1}^{T}\xi_t$$

$$s.t. \int_{\Theta} q(\Theta)\ln p(y_t|\mathbf{x}_t,\Theta) \geq \max_{y\neq y_t}\int_{\Theta} q(\Theta)\ln p(y|\mathbf{x}_t,\Theta) + \gamma - \xi_t \;\; \forall t$$

Dual

$$\max_{\lambda\in[0,C]} -\ln\int_{\Theta}\hat{p}(\Theta)\prod_{t=1}^{T}\left(\frac{p(y_t|\mathbf{x}_t,\Theta)}{\max_{y\neq y_t} p(y|\mathbf{x}_t,\Theta)}\right)^{\lambda_t}\exp(-\gamma\lambda_t)$$

$$q(\Theta) \;=\; \frac{1}{Z(\lambda)}\hat{p}(\Theta)\prod_{t=1}^{T}\left(\frac{p(y_t|\mathbf{x}_t,\Theta)}{\max_{y\neq y_t} p(y|\mathbf{x}_t,\Theta)}\right)^{\lambda_t}\exp(-\gamma\lambda_t)$$

MED for Exponential Family

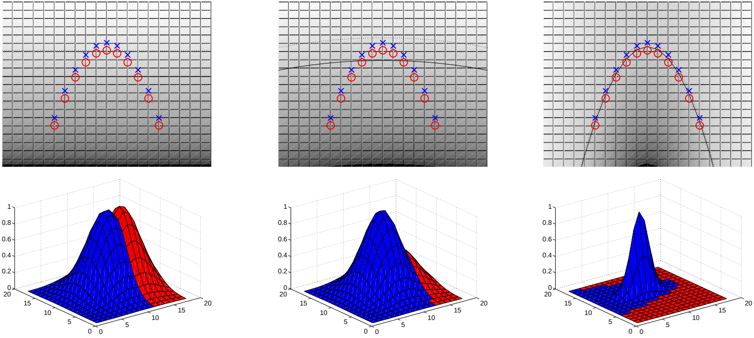
- Consider binary case $y \in \{\pm 1\}$
- Discriminant as ratio of class-conditional exponential families
- Set $p(y|\mathbf{x}, \Theta) \propto e^{\mathbf{x}^\top \theta_y - K(\theta_y) + by/2}$ and $\Theta = \{\theta_+, \theta_-, b\}$
- Set $\hat{p}(\Theta) \propto e^{\theta_{+1}^\top \chi - \mathcal{K}(\chi)} e^{\theta_{-1}^\top \chi - \mathcal{K}(\chi)} \mathcal{N}(b|0, \infty)$ conjugate prior
- MED dual is a convex program (Jebara 04)

$$\max_{\substack{\lambda \in [0, C] \\ \sum_i y_i \lambda_i = 0}} \gamma \sum_i \lambda_i - \mathcal{K}(\chi + \sum_i y_i \lambda_i \mathbf{x}_i) - \mathcal{K}(\chi - \sum_i y_i \lambda_i \mathbf{x}_i)$$

- Solvable in $\mathcal{O}(T^3)$ via e.g. ellipsoid method

MED for Exponential Family

- Discriminant as ratio of class-conditional exponential families
- Set $p(y|\mathbf{x}, \Theta) \propto \mathcal{N}(\mathbf{x}|\mu_y, \Sigma_y)e^{by/2}$ as Gaussians with parameters given by $\Theta = \{\mu_{+1}, \Sigma_{+1}, \mu_{-1}, \Sigma_{-1}, b\}$



(a) MED Initialization (b) MED Intermediate (c) MED Converged

Figure: Discriminative constraints for Gaussians.

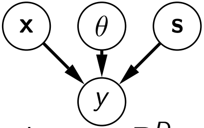
MED for Support Vector Machines

- Set $p(y|\mathbf{x}, \Theta) \propto \exp(y/2(\mathbf{x}^\top \theta + b))$ and $\Theta = \{\theta, b\}$
- Set $\hat{p}(\Theta) = \mathcal{N}(b|0, \infty)\mathcal{N}(\theta|\mathbf{0}, \mathbf{I})$
- MED dual produces support vector machine optimization

$$\max_{\substack{\lambda \in [0, C] \\ \sum_i y_i \lambda_i = 0}} \gamma \sum_i \lambda_i - \frac{1}{2} \sum_{ij} \lambda_i \lambda_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j$$

- MED prediction becomes the same
  $\hat{y} = \mathrm{sign}\left(\sum_t y_t \lambda_t \mathbf{x}_t^\top \mathbf{x} + b\right)$
- Solvable in $\mathcal{O}(T^3)$ with quadratic programming
- Faster solution to ϵ accuracy with e.g. Pegasos

MED for Feature Selection

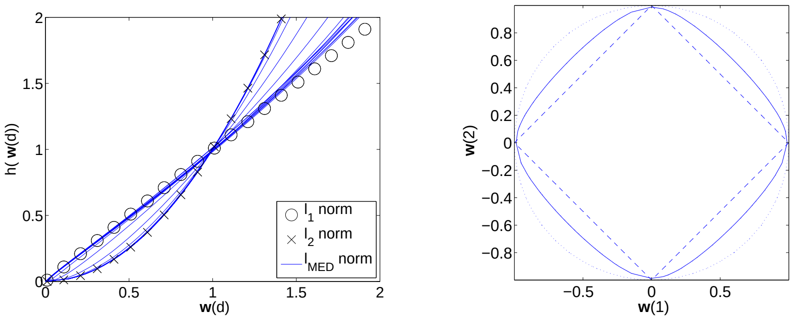


- Model $\Theta = \{\theta, b, \mathbf{s}\}$ where $\mathbf{s} \in \mathbb{B}^D$ sparsifies $\theta \in \mathbb{R}^D$
- Set $p(y|\mathbf{x}, \Theta, \mathbf{s}) \propto \exp(y/2(\sum_d \mathbf{s}(d)\mathbf{x}(d)\theta(d) + b))$
- Set $\hat{p}(\Theta) = \mathcal{N}(b|0, \infty)\mathcal{N}(\theta|\mathbf{0}, \mathbf{I}) \prod_d \rho^{\mathbf{s}(d)}(1-\rho)^{1-\mathbf{s}(d)}$
- Parameter ρ (or $\alpha = \frac{1-\rho}{\rho}$) is prior % of non-sparse features
- MED dual is

$$\max_{\substack{\lambda \in [0,C] \\ \sum_i y_i \lambda_i = 0}} \gamma \sum_i \lambda_i - \sum_{d=1}^{D} \ln \left(\alpha + e^{\frac{1}{2}\sum_{ij} y_i y_j \lambda_i \lambda_j \mathbf{x}_i(d) \mathbf{x}_j(d)} \right)$$

- MED prediction is $\hat{y} = \operatorname{sign}\left(\sum_{di} y_i \lambda_i \mathbf{x}_i(d)\mathbf{x}(d)\hat{\mathbf{s}}(d) + b\right)$
  where $\hat{\mathbf{s}}(d) = \left(1 + \alpha \exp(-\frac{1}{2}\sum_{ij} y_i y_j \lambda_i \lambda_j \mathbf{x}_i(d)\mathbf{x}_j(d))\right)^{-1}$

MED for Feature Selection



(a) One dimensional plot (b) Two dimensional contour plot

Figure: (a) Various norms on the weight vector shown as α varies from $\alpha = 0$ which mimics an ℓ_2 norm to α large which mimics an ℓ_1 norm. (b) a two-dimensional contour plot of the ℓ_1 penalty (dot-dash line), the ℓ_2 penalty (dotted line) and the ℓ_{MED} penalty with $\alpha = 2$ (solid line).

MED for Feature Selection

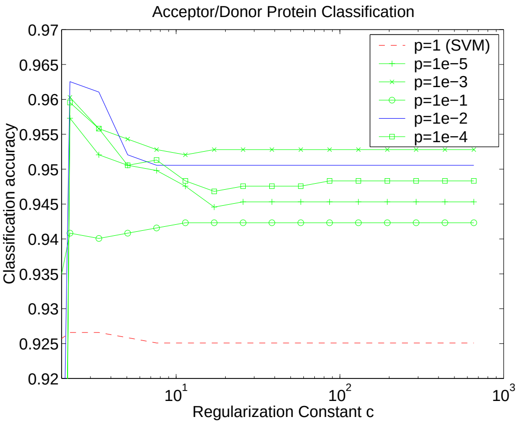
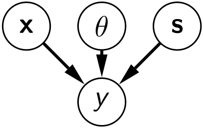


Figure: Acceptor/donor sequence classification accuracy.

MED for Kernel Selection

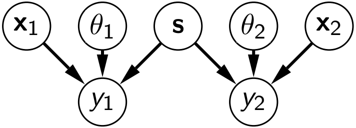


- Select from $\phi_1(\mathbf{x}), \ldots, \phi_D(\mathbf{x})$ mappings to Hilbert space
- Model $\Theta = \{\theta_1, \ldots, \theta_D, b, \mathbf{s}\}$ where $\mathbf{s} \in \mathbb{B}^D$ selects $\theta_d \in \mathcal{H}$
- Set $p(y|\mathbf{x}, \Theta, \mathbf{s}) \propto \exp(y/2(\sum_d \mathbf{s}(d)\theta_d^\top \phi_d(\mathbf{x}) + b))$
- MED dual is

$$\max_{\substack{\lambda \in [0,C] \\ \sum_i y_i \lambda_i = 0}} \gamma \sum_i \lambda_i - \sum_{d=1}^{D} \ln \left(\alpha + e^{\frac{1}{2} \sum_{ij} y_i y_j \lambda_i \lambda_j k_d(\mathbf{x}_i, \mathbf{x}_j)} \right)$$

- MED prediction is $\hat{y} = \mathrm{sign}\left(\sum_{di} y_i \lambda_i \hat{\mathbf{s}}(d) k_d(\mathbf{x}, \mathbf{x}_i) + b\right)$ where $\hat{\mathbf{s}}(d) = \left(1 + \alpha \exp(-\frac{1}{2} \sum_{ij} y_i y_j \lambda_i \lambda_j k_d(\mathbf{x}, \mathbf{x}_i))\right)^{-1}$

Multi-Task Margin Constraints for Kernel Selection



- Have M models $\Theta = \{\Theta_1, \ldots, \Theta_M, \mathbf{s}\}$ and sparsifier $\mathbf{s} \in \mathbb{B}^D$
- Each model is $\Theta_m = \{\theta_{m1}, \ldots, \theta_{mD}, b_m\}$
- Set $p(y|\mathbf{x}, \Theta_m, \mathbf{s}) \propto \exp(y/2(\sum_d \mathbf{s}(d)\theta_{md}^\top \phi_d(\mathbf{x}) + b_m))$
- MED dual is

$$\max_{\substack{\lambda \in [0,C] \\ \sum_i y_{mi}\lambda_{mi}=0}} \gamma \sum_{mi} \lambda_{mi} - \sum_{d=1}^{D} \ln\left(\alpha + e^{\frac{1}{2}\sum_{mij} y_{mi}y_{mj}\lambda_{mi}\lambda_{mj}k_d(\mathbf{x}_{mi},\mathbf{x}_{mj})}\right)$$

- Task m predicts $\hat{y} = \mathrm{sign}\left(\sum_{di} y_{mi}\lambda_{mi}\hat{\mathbf{s}}(d)k_d(\mathbf{x}, \mathbf{x}_{mi}) + b_m\right)$

$$\hat{\mathbf{s}}(d) = \left(1 + \alpha \exp(-\tfrac{1}{2}\sum_m \sum_{ij} y_{mi}y_{mj}\lambda_{mi}\lambda_{mj}k_d(\mathbf{x}_{mi}, \mathbf{x}_{mj}))\right)^{-1}$$

Multi-Task Margin Constraints for Kernel Selection

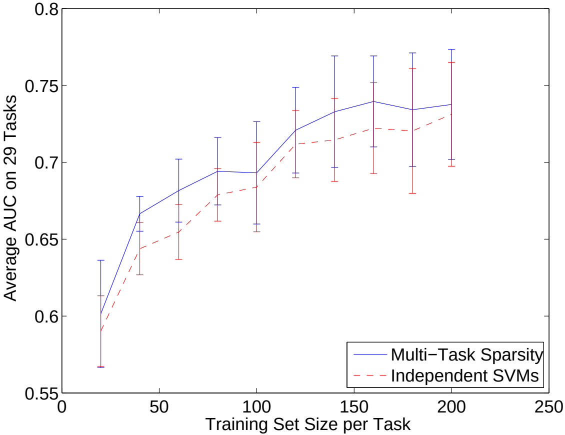


Figure: Feature and RBF kernel selection on the Landmine dataset. Values for C and α obtained by cross-validation on held out data.

Sequential Quadratic Programming

- How to optimize MED when it's not a QP? For example, $\max_{\lambda \in [0,C], \sum_i y_i \lambda_i = 0} \gamma \sum_i \lambda_i - \sum_{d=1}^{D} \ln\left(\alpha + e^{\lambda^\top H \lambda}\right)$
- Lower bound $-\ln$ terms with a quadratic, get sequential QP

Theorem (Jebara 09)

For any $\mathbf{v}$*, the term* $-\ln\left(\alpha + e^{\mathbf{u}^\top \mathbf{u}}\right)$ *is lower bounded by*

$$-\ln\left(\alpha + e^{\mathbf{v}^\top \mathbf{v}}\right) - 2\frac{\mathbf{v}^\top(\mathbf{u}-\mathbf{v})}{\alpha e^{-\mathbf{v}^\top \mathbf{v}} + 1} - (\mathbf{u}-\mathbf{v})^\top \left(\mathcal{G}\mathbf{v}\mathbf{v}^\top + I\right)(\mathbf{u}-\mathbf{v})$$

when $\mathcal{G} \geq \frac{\tanh(\frac{1}{2}\ln(\alpha \exp(-\mathbf{v}^\top \mathbf{v})))}{\ln(\alpha \exp(-\mathbf{v}^\top \mathbf{v}))}$.

Sequential Quadratic Programming

0	Input: dataset $\mathcal{D}$, $C > 0$, $\alpha \geq 0$, $0 < \epsilon < 1$
1	Initialize Lagrange multipliers to zero, $\lambda = \mathbf{0}$.
2	Store $\tilde{\lambda} = \lambda$.
3	Apply bound on all $-\ln\left(\alpha + e^{\lambda^\top H\lambda}\right)$ terms at $\tilde{\lambda}$.
4	Solve resulting (fast) SVM problem to get λ.
5	If $\|\lambda - \tilde{\lambda}\| > \epsilon\|\lambda\|$ go to 2.
6	Output λ.

Theorem (Jebara 09)

SQP achieves $(1-\epsilon)J(\lambda^*)$ within $\left\lceil \frac{\log(1/\epsilon)}{\log\left(\min\left(1+\frac{1}{\alpha},2\right)\right)} \right\rceil$ iterations.

Sparse multi-task is a constant factor more work than SVMs.

Sequential Quadratic Programming

Proof.

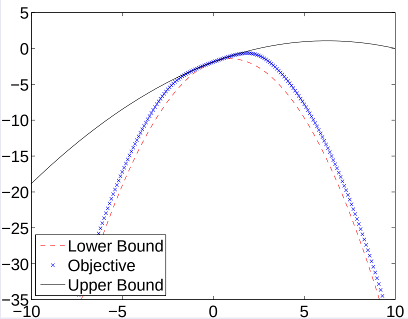


Figure: Quadratic bounding sandwich. Compare upper and lower bound curvatures to bound maximum # of iterations.

□

Relative Margin Constraints

- Why stop with just γ margin constraints?
- Limit spread between correct to weakest by β constraint
- Relative margin machines (Shivaswamy & Jebara 08)

$$\min_{q(\Theta)} \mathcal{KL}\left(q(\Theta)\|\hat{p}(\Theta)\right)$$

$$s.t. \int_{\Theta} q(\Theta) \ln p(y_t|\mathbf{x}_t,\Theta) \geq \max_{y \neq y_t} \int_{\Theta} q(\Theta) \ln p(y|\mathbf{x}_t,\Theta) + \gamma \;\; \forall t$$

$$s.t. \int_{\Theta} q(\Theta) \ln p(y_t|\mathbf{x}_t,\Theta) \leq \min_{y \neq y_t} \int_{\Theta} q(\Theta) \ln p(y|\mathbf{x}_t,\Theta) + \beta \;\; \forall t$$

- The SVM optimization (with slack omitted) then becomes $\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2$ *subject to* $\beta \geq y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq \gamma$

Relative Margin Constraints

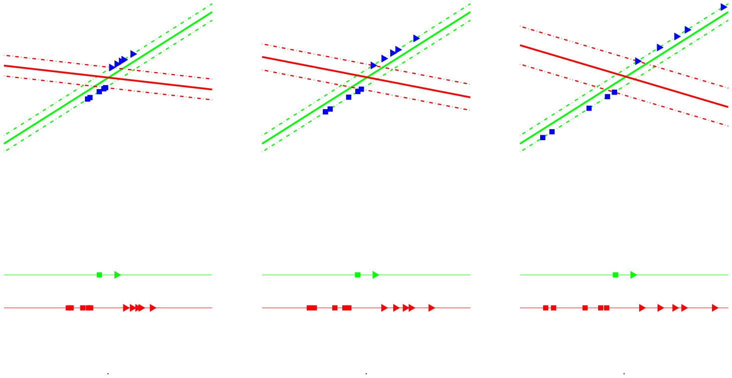


Figure: Top: As the data is scaled, the SVM (red or dark shade) deviates from the RMM (green or light shade). Bottom: The projections of the examples (that is $\mathbf{w}^\top \mathbf{x} + b$) on the real line for the SVM and the RMM.

Relative Margin Constraints in Binary Classification

Dataset	SVM	KFDA	Σ-SVM	RMM (C=D)	RMM
banana	10.5±0.4	10.8±0.5	10.5±0.4	**10.4±0.4**	**10.4±0.4***
b.cancer	**25.3±4.6***	26.6±4.8	28.8±4.6	25.9±4.5	27.2±4.8
diabetes	**23.1±1.7**	**23.2±1.8**	24.2±1.9	**23.1±1.7**	**23.0±1.7***
f.solar	**32.3±1.8**	33.1±1.6	34.6±2.0	**32.3±1.8***	33.1±2.5
german	**23.4±2.2**	24.1±2.4	25.9±2.4	**23.4±2.1**	**23.2±2.2***
heart	15.5±3.3	15.7±3.2	19.9±3.6	15.4±3.3	**15.2±3.1***
image	**3.0±0.6**	3.1±0.6	3.3±0.7	3.0±0.6	**2.9±0.7**
ringnorm	1.5±0.1	**1.5±0.1**	1.5±0.1	**1.5±0.1**	**1.5±0.1***
splice	10.9±0.7	10.6±0.7	10.8±0.6	10.8±0.6	10.8±0.6
thyroid	4.7±2.1	**4.2±2.1**	4.5±2.1	**4.2±1.8***	**4.2±2.2**
titanic	22.3±1.1	**22.0±1.3***	23.1±2.2	22.3±1.1	**22.2±1.3**
twonorm	2.4±0.1	2.4±0.2	2.5±0.2	2.4±0.1	**2.3±0.1***
waveform	9.9±0.4	9.9±0.4	10.5±0.5	10.0±0.4	**9.7±0.4***

Relative Margin Constraints in Structured Prediction

MED extended to structured prediction (Zhu & Xing 09)
Add relative margin to structured prediction (Shivaswamy & J 09)
Multi-class classification error

Kernel	StructSVM	StructRMM	p-value
Poly 1	3.78 ± 0.54	3.85 ± 0.62	0.55
Poly 2	2.11 ± 0.43	$\mathbf{1.46} \pm \mathbf{0.34}$	0.00
Ploy 3	1.73 ± 0.37	$\mathbf{1.24} \pm \mathbf{0.43}$	0.00
Poly 4	1.55 ± 0.45	$\mathbf{1.18} \pm \mathbf{0.43}$	0.00

Sequence label error (Named Entity Rec. & Part of Speech)

	CRF	StructSVM	StructRMM	p-value
NER	5.13 ± 0.28	5.09 ± 0.32	$\mathbf{5.05} \pm \mathbf{0.28}$	0.07
POS	11.34 ± 0.64	11.14 ± 0.60	$\mathbf{10.42} \pm \mathbf{0.47}$	0.00

Conclusions

- Add margin constraints in generative learning
- Leads to Maximum Entropy Discrimination
- Natural tool for multi-task feature and kernel sparsity
- Optimizations via sequential quadratic programming
- Only constant time more work than SVM
- Relative margin constraints yield further improvements
- Relative margin is a little generative in nature...