Accelerated Training of Max-Margin Markov Networks with Kernels

Xinhua Zhang
University of Alberta
Alberta Innovates Centre for Machine Learning (AICML)

Joint work with
Ankan Saha (Univ. of Chicago) and **SVN Vishwanathan** (Purdue Univ)

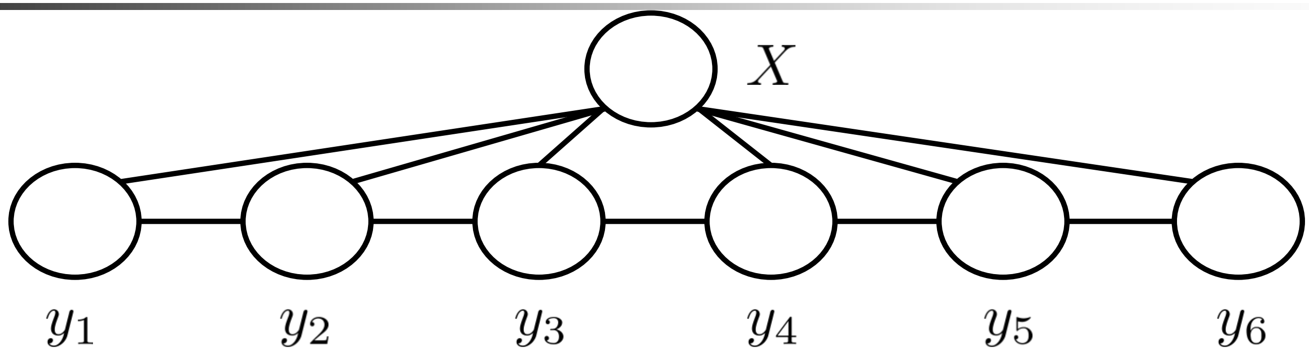
Outline

- Objective of max-margin Markov network (M^3N)
- Smoothing for M^3N
- Excessive gap technique in general, and problem for M^3N
- Bregman divergence for prox-function
  - Retain the accelerated rates $\frac{1}{k^2}$
  - Efficient computation by graphical model factorization
- Kernelization
- Conclusion

Outline

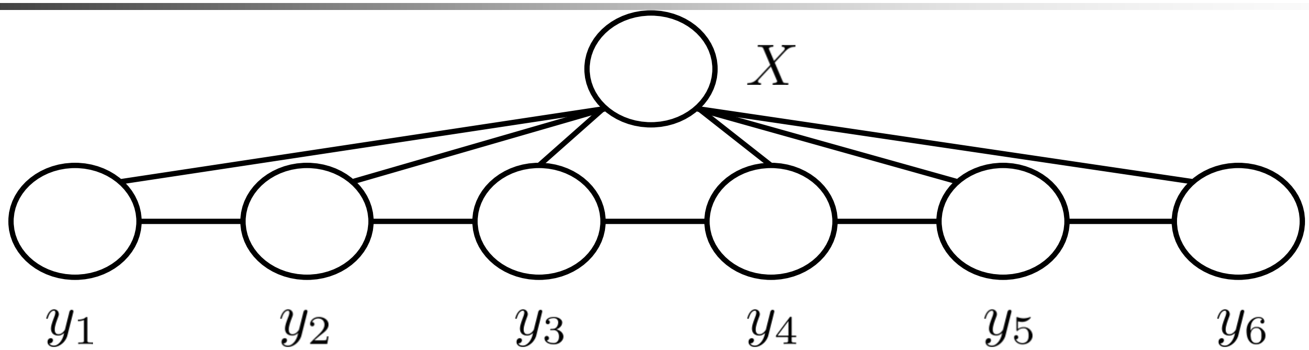
- Objective of max-margin Markov network (M^3N)
- Smoothing for M^3N
- Excessive gap technique in general, and problem for M^3N
- Bregman divergence for prox-function
  - Retain the accelerated rates $\frac{1}{k^2}$
  - Efficient computation by graphical model factorization
- Kernelization
- Conclusion

Structured output prediction



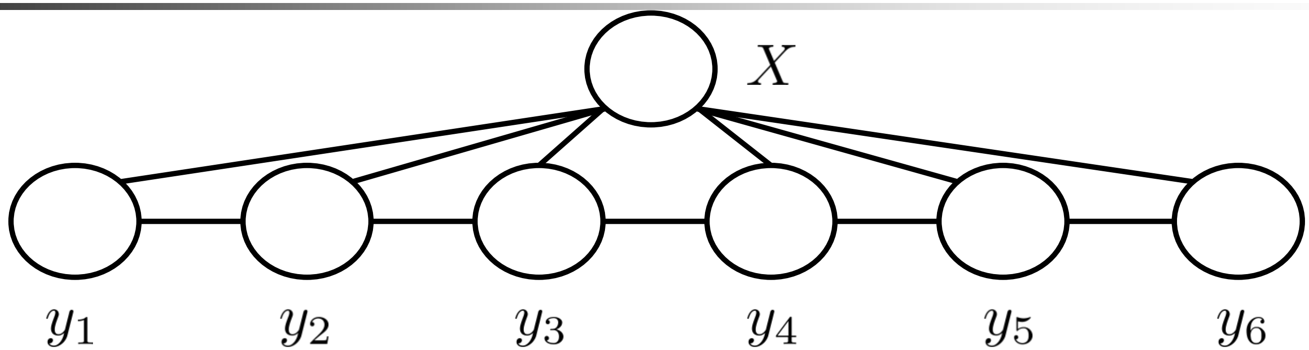
- Structured feature/label joint map: $\phi(\mathbf{x}^i, \mathbf{y}^i)$
- Linear discriminant: $\langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) \rangle$
- Structured label loss: $\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i)$ with $\ell(\mathbf{y}^i, \mathbf{y}^i; \mathbf{x}^i) = 0$
- Hinge loss
  - Loss augmented discriminant: $\overbrace{\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) + \langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}) \rangle}^{:=\Psi(\mathbf{y})} \ \forall\ \mathbf{y}$

Structured output prediction



- Structured feature/label joint map: $\phi(\mathbf{x}^i, \mathbf{y}^i)$
- Linear discriminant: $\langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) \rangle$
- Structured label loss: $\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i)$ with $\ell(\mathbf{y}^i, \mathbf{y}^i; \mathbf{x}^i) = 0$
- Hinge loss
  - Loss augmented discriminant $\overbrace{\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) + \langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}) \rangle}^{:= \Psi(\mathbf{y})} \ \forall\ \mathbf{y}$
  - Max over $\mathbf{y}$: $\max_{\mathbf{y} \in \mathcal{Y}} \{ \Psi(\mathbf{y}) - \Psi(\mathbf{y}_i) \}$

Structured output prediction



- Structured feature/label joint map: $\phi(\mathbf{x}^i, \mathbf{y}^i)$
- Linear discriminant: $\langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) \rangle$
- Structured label loss: $\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i)$ with $\ell(\mathbf{y}^i, \mathbf{y}^i; \mathbf{x}^i) = 0$
- Hinge loss:
  - Loss augmented discriminant $\overbrace{\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) + \langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}) \rangle}^{:= \Psi(\mathbf{y})} \;\; \forall\, \mathbf{y}$
  - Max over $\mathbf{y}$: $\max_{\mathbf{y} \in \mathcal{Y}} \left\{ \ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) - \langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) - \phi(\mathbf{x}^i, \mathbf{y}) \rangle \right\}$

Max-margin Markov networks and conditional random fields

- Structured feature/label joint map: $\phi(\mathbf{x}^i, \mathbf{y}^i)$
- Linear discriminant: $\langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) \rangle$
- Structured label loss: $\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i)$
- M^{3}N:

$$J(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n} \max_{\mathbf{y}\in\mathcal{Y}} \left\{ \ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) - \langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) - \phi(\mathbf{x}^i, \mathbf{y}) \rangle \right\}$$

- CRF:

$$J(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n} \log \sum_{\mathbf{y}\in\mathcal{Y}} \exp\left(\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) - \langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) - \phi(\mathbf{x}^i, \mathbf{y}) \rangle \right)$$

Max-margin Markov networks and conditional random fields

- Structured feature/label joint map: $\phi(\mathbf{x}^i, \mathbf{y}^i)$
- Linear discriminant: $\langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) \rangle$
- Structured label loss: $\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i)$
- M^{3}N:

$$J(\mathbf{w}) = \frac{\lambda}{2} \|\mathbf{w}\|_2^2 + \frac{1}{n} \sum_{i=1}^{n} \max_{\mathbf{y} \in \mathcal{Y}} \left\{ \ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) - \langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) - \phi(\mathbf{x}^i, \mathbf{y}) \rangle \right\}$$

- CRF:

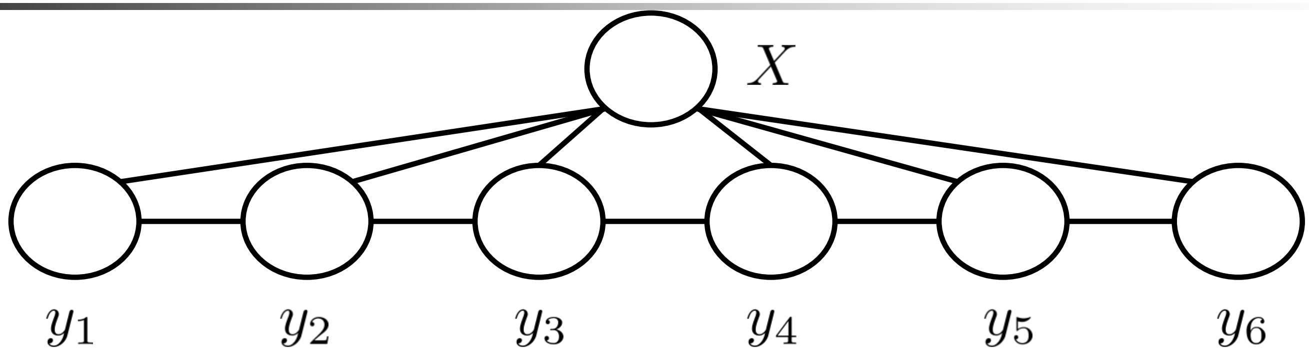
$$J(\mathbf{w}) = \frac{\lambda}{2} \|\mathbf{w}\|_2^2 + \frac{1}{n} \sum_{i=1}^{n} \log \sum_{\mathbf{y} \in \mathcal{Y}} \exp\left(\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) - \langle \mathbf{w}, \phi(\mathbf{x}^i, \mathbf{y}^i) - \phi(\mathbf{x}^i, \mathbf{y}) \rangle \right)$$

Max-Margin Markov Networks

- Major challenges
  - Large space of $\mathcal{Y}$, so need to (carefully) keep factorization
  - Loss is not smooth

$$J(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n} \underbrace{\max_{\mathbf{y}\in\mathcal{Y}} \left\{ \ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) - \left\langle \mathbf{w}, \boldsymbol{\phi}(\mathbf{x}^i, \mathbf{y}^i) - \boldsymbol{\phi}(\mathbf{x}^i, \mathbf{y}) \right\rangle \right\}}_{\text{not smooth}}$$

Factorization for structured output space $\mathcal{Y}$



- Factorization
  - Feature factorization: $\boldsymbol{\phi}(\mathbf{x}^i, \mathbf{y}) = \bigoplus_{c \in \mathcal{C}} \boldsymbol{\phi}(\mathbf{x}^i, y_c)$
  - Loss factorization

$$\ell(\mathbf{y}, \mathbf{y}^i; \mathbf{x}^i) = \sum_{c \in \mathcal{C}} \ell(y_c, y_c^i; \mathbf{x}^i)$$

  - Probability factorization

$$p(\mathbf{y}; \mathbf{x}) \propto \prod_{c \in \mathcal{C}} \exp\left(\psi_c(y_c, \mathbf{x})\right)$$

Non-smooth solvers: State of the art for M³N

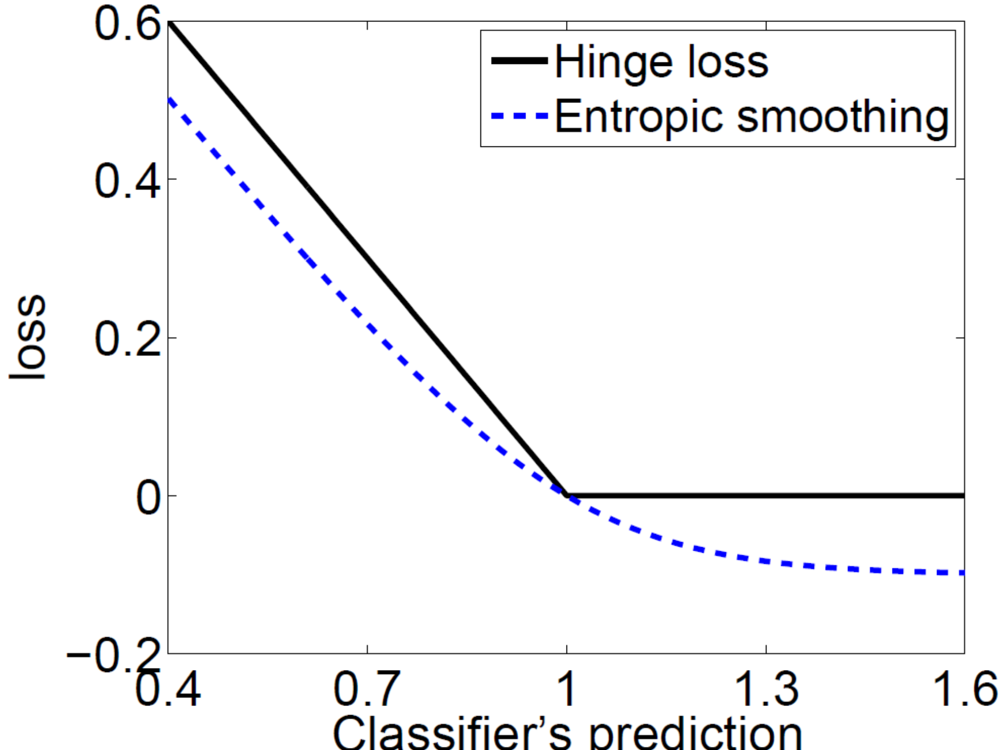
Algorithm	Rate of convergence
BMRM / SVM-Struct	$O\left(\frac{G^2 \log \lvert\mathcal{Y}\rvert}{\lambda\epsilon}\right)$ $\left(\lVert\phi(\mathbf{x}^i, y_c)\rVert \le G\right)$
Extragradient	
Exponentiated Gradient	
SMO	pd: $O\left(n \lvert\mathcal{Y}\rvert \log \frac{1}{\epsilon}\right)$ psd: $O\left(n \lvert\mathcal{Y}\rvert \frac{1}{\lambda\epsilon}\right)$
Our algorithm	$O\left(G\sqrt{\frac{\log \lvert\mathcal{Y}\rvert}{\lambda\epsilon}}\right)$

Outline

- Objective of max-margin Markov network (M^3N)
- Smoothing for M^3N
- Excessive gap technique in general, and problem for M^3N
- Bregman divergence for prox-function
  - Retain the accelerated rates $\frac{1}{k^2}$
  - Efficient computation by graphical model factorization
- Kernelization
- Conclusion

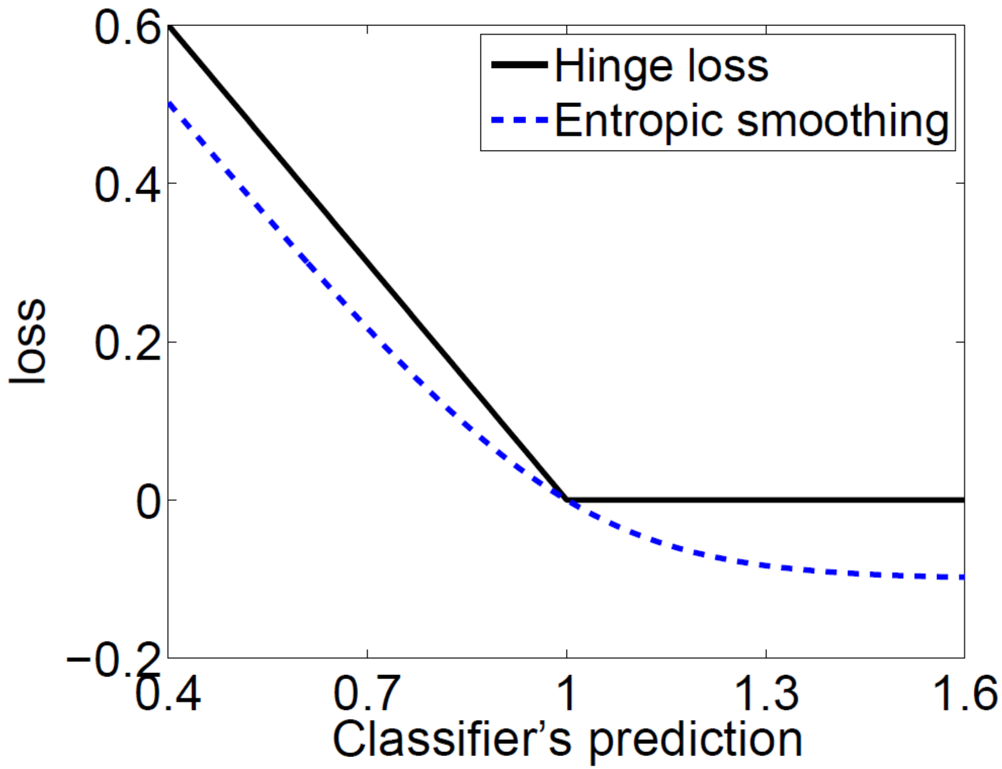
Intuition of smoothing

- Find a smooth and tight approximation of the non-smooth objectives



Intuition of smoothing

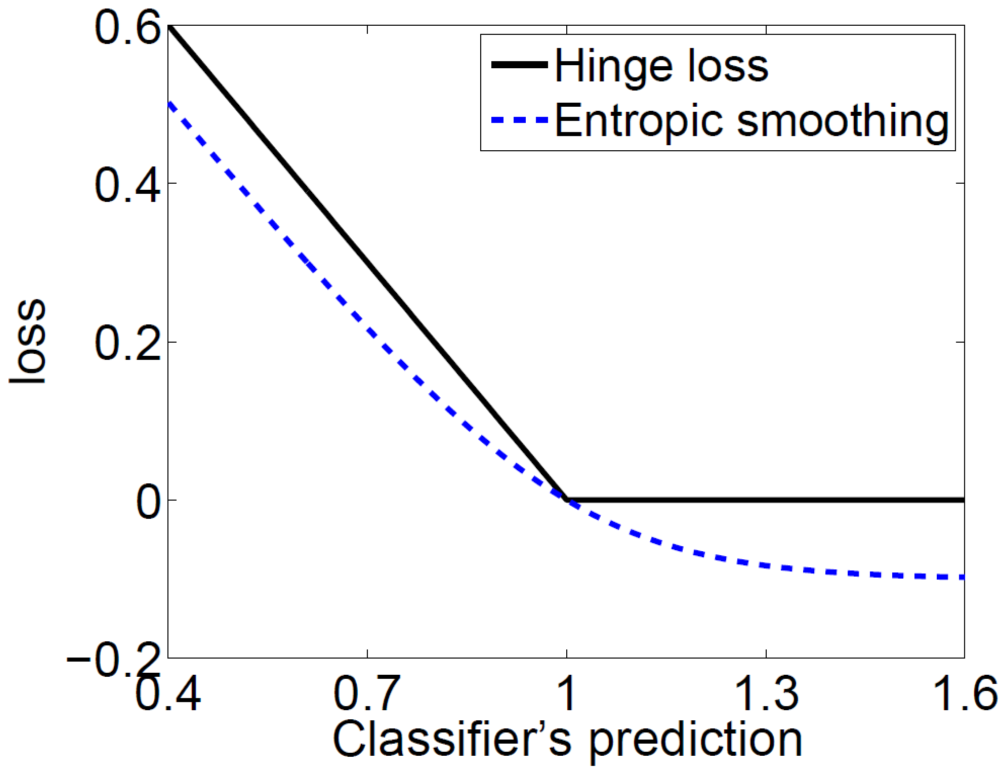
- Find a smooth and tight approximation of the non-smooth objectives



Q: general procedure for smoothing?

Intuition of smoothing

- Find a smooth and tight approximation of the non-smooth objectives



Q: general procedure for smoothing?

A: Fenchel conjugation

Key observation

- Loss for M³N has rich structure (though non-smooth)

$$\frac{1}{n}\sum_{i=1}^{n}\max_{\mathbf{y}\in\mathcal{Y}}\left\{\ell(\mathbf{y},\mathbf{y}^i;\mathbf{x}^i)+\underbrace{\left\langle \mathbf{w},\boldsymbol{\phi}(\mathbf{x}^i,\mathbf{y})-\boldsymbol{\phi}(\mathbf{x}^i,\mathbf{y}^i)\right\rangle}_{u^i_{\mathbf{y}}=\langle \mathbf{w},A_{i,\mathbf{y}}\rangle}\right\}$$

- Can be rewritten
  - A : a matrix stacking ϕ features

$$\frac{1}{n}\sum_{i=1}^{n}\max_{\mathbf{y}\in\mathcal{Y}}\left\{\ell(\mathbf{y},\mathbf{y}^i;\mathbf{x}^i)+u^i_{\mathbf{y}}\right\} \qquad \mathbf{u}=A^\top\mathbf{w}$$

Key observation

- Loss for M³N has rich structure (though non-smooth)

$$\frac{1}{n}\sum_{i=1}^{n}\max_{\mathbf{y}\in\mathcal{Y}}\left\{\ell(\mathbf{y},\mathbf{y}^i;\mathbf{x}^i)+\underbrace{\langle\mathbf{w},\boldsymbol{\phi}(\mathbf{x}^i,\mathbf{y})-\boldsymbol{\phi}(\mathbf{x}^i,\mathbf{y}^i)\rangle}_{u^i_{\mathbf{y}}=\langle\mathbf{w},A_{i,\mathbf{y}}\rangle}\right\}$$

- Can be rewritten
  - A : a matrix stacking ϕ features

$$\frac{1}{n}\sum_{i=1}^{n}\max_{\mathbf{y}\in\mathcal{Y}}\left\{\ell(\mathbf{y},\mathbf{y}^i;\mathbf{x}^i)+u^i_{\mathbf{y}}\right\} \qquad \mathbf{u}=A^\top\mathbf{w}$$

  - Further written as $g^\star(\mathbf{u})$

    where g is a convex function with a compact domain Q

Key observation

- Loss for M³N has rich structure (though non-smooth)

$$\frac{1}{n}\sum_{i=1}^{n}\max_{\mathbf{y}\in\mathcal{Y}}\left\{\ell(\mathbf{y},\mathbf{y}^i;\mathbf{x}^i)+\underbrace{\langle\mathbf{w},\boldsymbol{\phi}(\mathbf{x}^i,\mathbf{y})-\boldsymbol{\phi}(\mathbf{x}^i,\mathbf{y}^i)\rangle}_{u^i_{\mathbf{y}}=\langle\mathbf{w},A_{i,\mathbf{y}}\rangle}\right\}$$

- Empirical risk can be written as $g^\star(A^\top\mathbf{w})$
  - A : a matrix stacking $\boldsymbol{\phi}$ features
  - g : is a convex function with a compact domain Q

$$Q=\left\{\boldsymbol{\alpha}:\alpha^i_{\mathbf{y}}\geq 0,\ \text{and}\ \sum_{\mathbf{y}}\alpha^i_{\mathbf{y}}=\frac{1}{n},\ \forall\, i\right\}$$

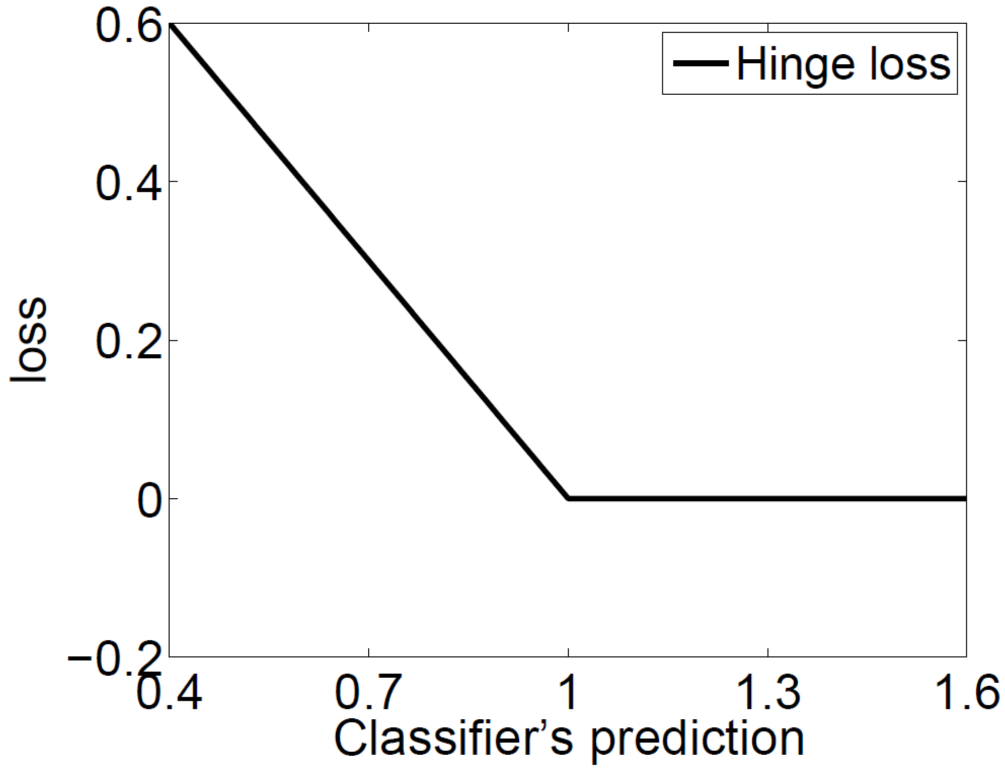
$$g(\boldsymbol{\alpha})=\begin{cases}-\sum_i\sum_{\mathbf{y}}\ell^i_{\mathbf{y}}\alpha^i_{\mathbf{y}} & \text{if } \boldsymbol{\alpha}\in Q\\ +\infty & \text{otherwise.}\end{cases}$$

Significance of this $g^\star(A^\top \mathbf{w})$ reformulation: smoothing

- It helps us to design a *tight* and *smooth* approximation
- Use a prox-function *d*
  - d is strongly convex with modulus 1 (wrt some norm on Q)
  - $\min_{\boldsymbol{\alpha} \in Q} d(\boldsymbol{\alpha}) = 0$, let $\mathcal{D} = \max_{\boldsymbol{\alpha} \in Q} d(\boldsymbol{\alpha})$
- Desirable properties
  - $(g + \mu d)^\star$ has Lipschitz continuous gradient (lcg)
  - $(g + \mu d)^\star - g^\star \in [-\mu \mathcal{D}, 0]$

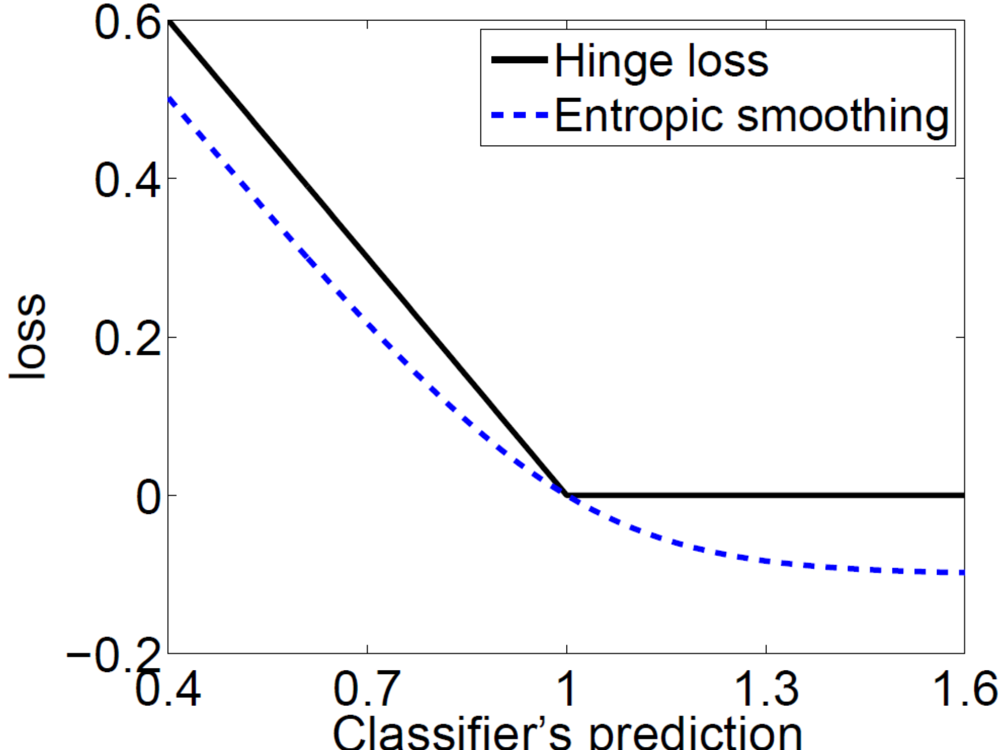
Example approximation: *tight* and *smooth*

- Example: hinge loss



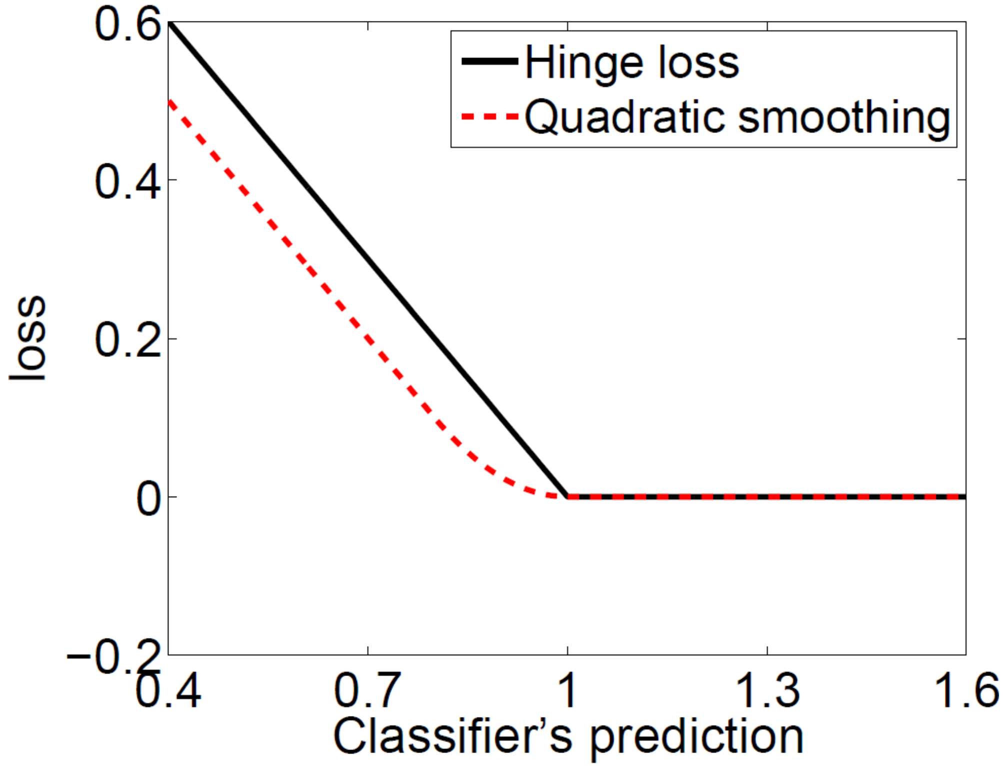
Example approximation: *tight* and *smooth*

- Example: hinge loss
- Entropic prox-function: logistic loss



Example approximation: *tight* and *smooth*

- Example: hinge loss
- Quadratic prox-function



Smoothing M3N into CRF

- M3N loss

$$g^\star(\mathbf{u}) = \frac{1}{n}\sum_{i=1}^{n}\max_{\mathbf{y}\in\mathcal{Y}}\left\{\ell(\mathbf{y},\mathbf{y}^i;\mathbf{x}^i) - u^i_{\mathbf{y}}\right\}$$

- Use entropic prox-function

$$d(\boldsymbol{\alpha}) = \sum_{i=1}^{n}\sum_{\mathbf{y}}\alpha^i_{\mathbf{y}}\log\alpha^i_{\mathbf{y}} + \log n + \log|\mathcal{Y}|\,,$$

then

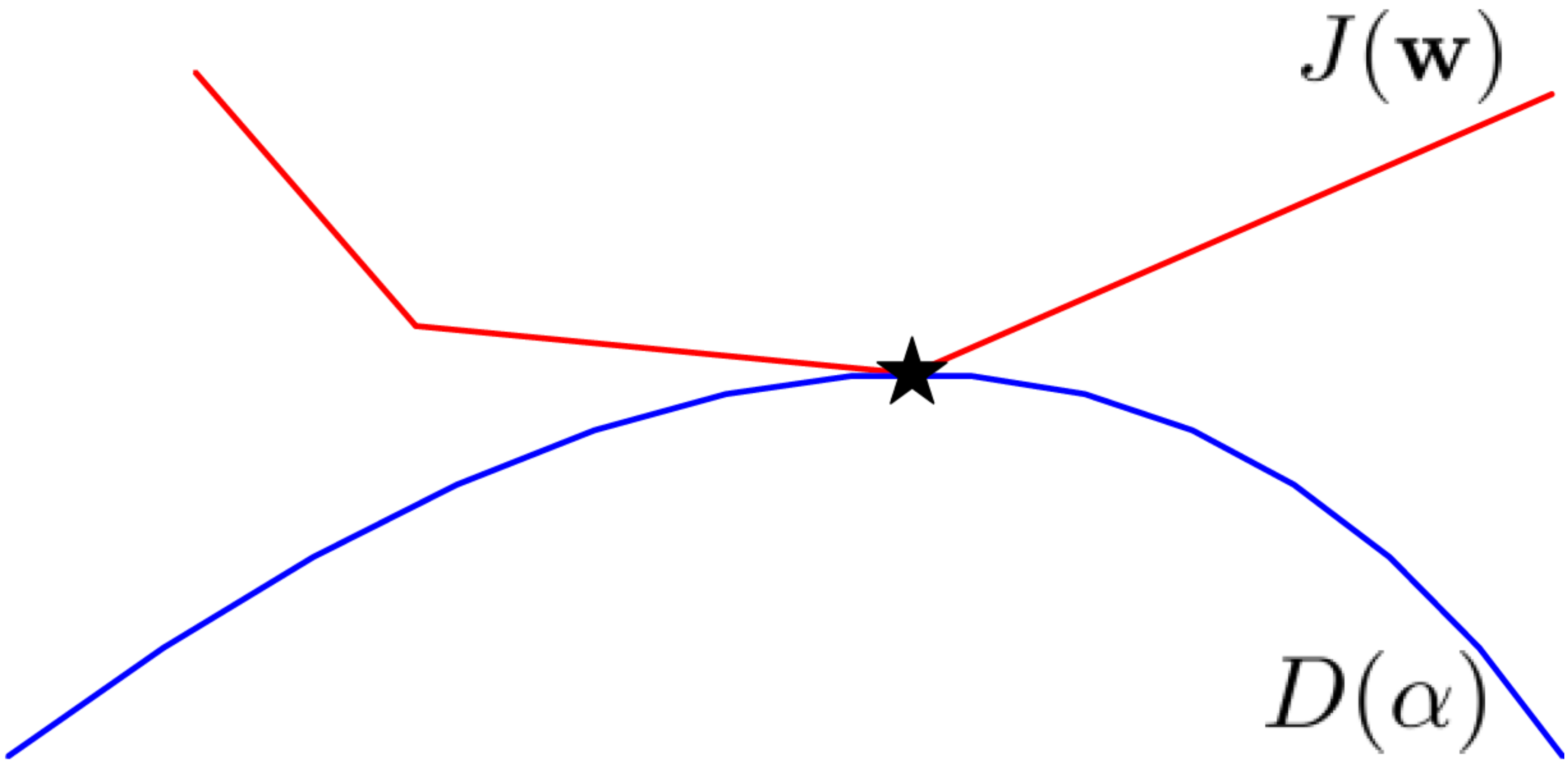
$$(g+\mu d)^\star(\mathbf{u}) = \frac{\mu}{n}\sum_{i=1}^{n}\log\sum_{\mathbf{y}\in\mathcal{Y}}\exp\left(\frac{u^i_{\mathbf{y}} + \ell^i_{\mathbf{y}}}{\mu}\right) - \mu\log|\mathcal{Y}|$$

CRF

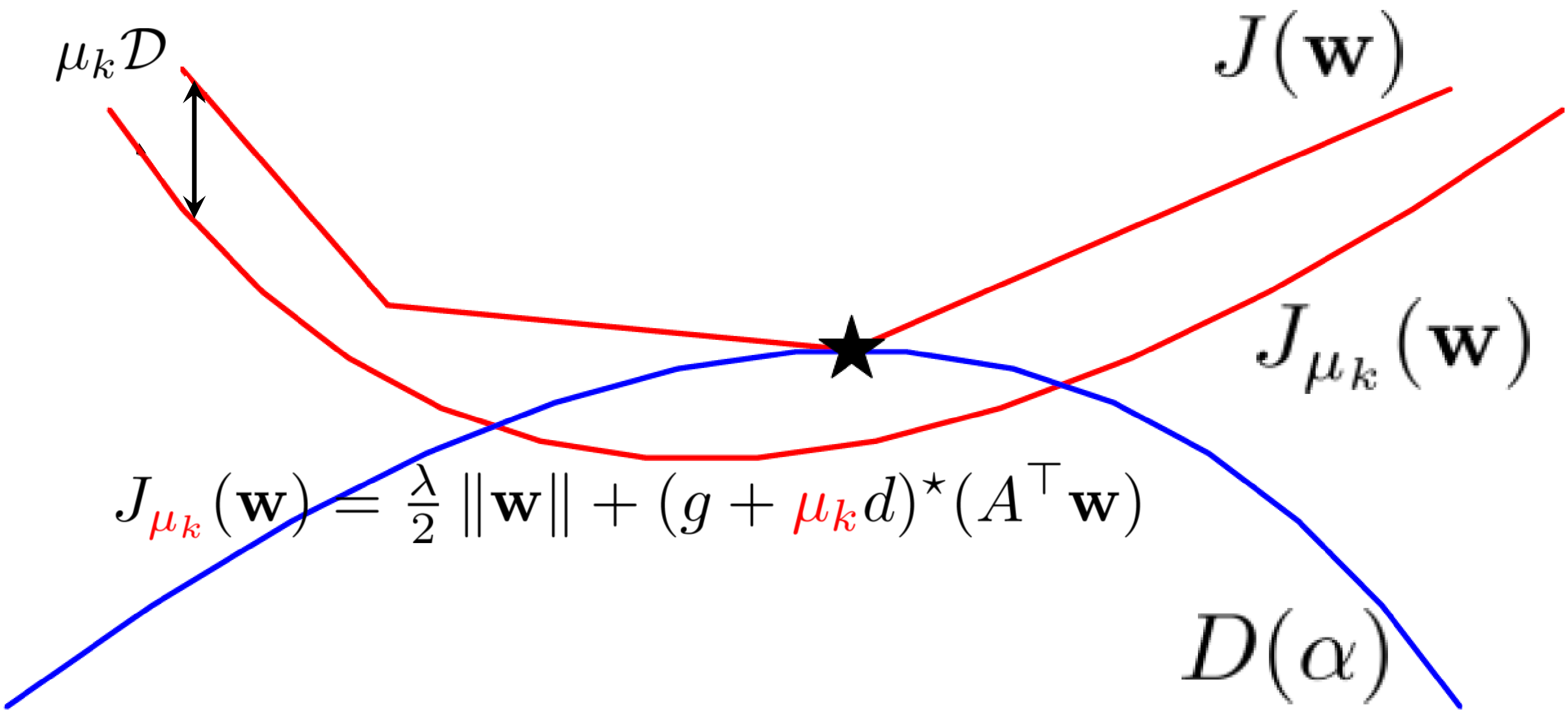
Outline

- Objective of max-margin Markov network (M^3N)
- Smoothing for M^3N
- Excessive gap technique in general, and problem for M^3N
- Bregman divergence for prox-function
  - Retain the accelerated rates $\frac{1}{k^2}$
  - Efficient computation by graphical model factorization
- Kernelization
- Conclusion

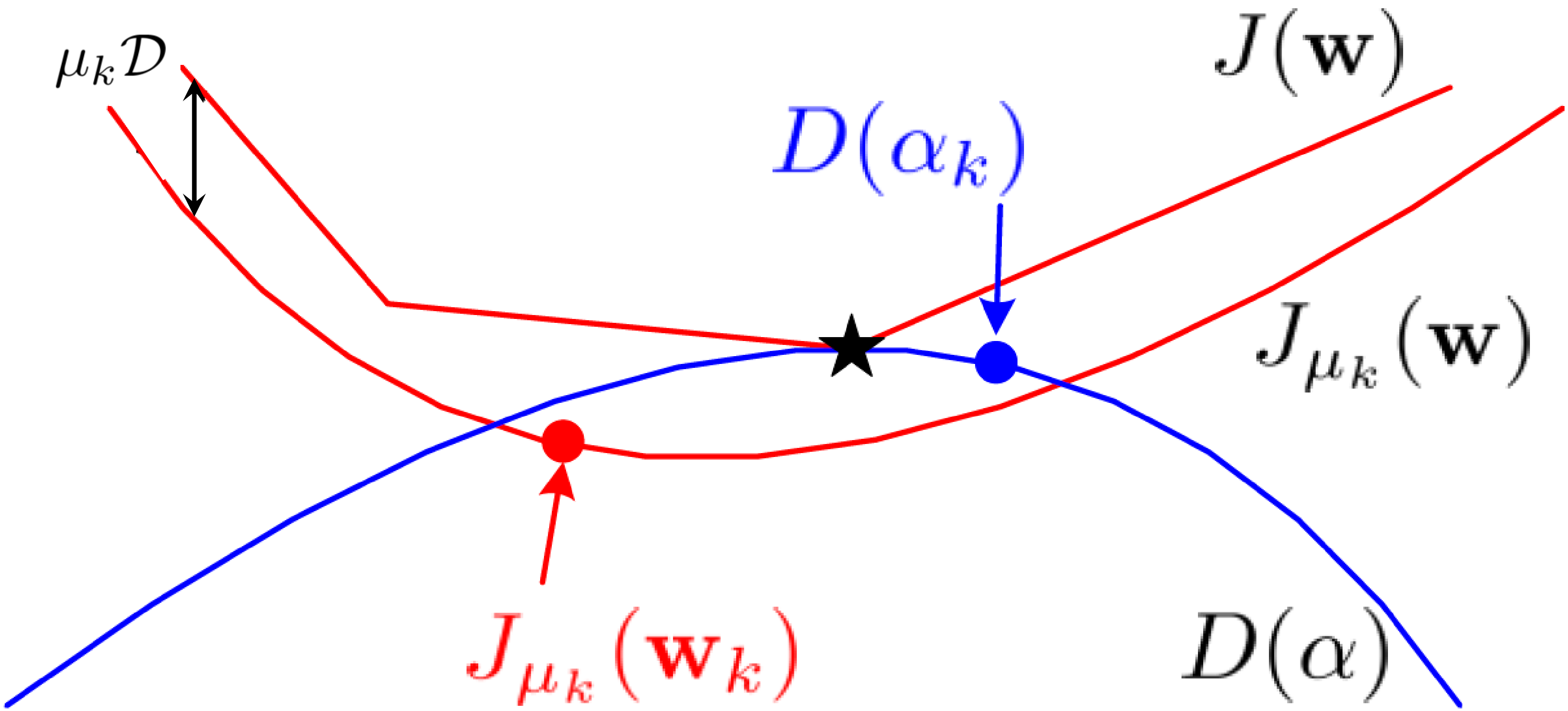
Excessive Gap Technique



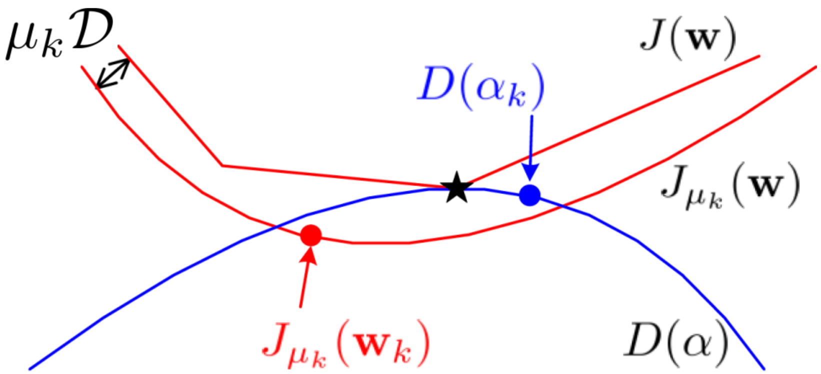
Excessive Gap Technique



Excessive Gap Technique



Key technical challenges



$$J(\mathbf{w}_k) - D(\boldsymbol{\alpha}_k) \leq \mu_k \mathcal{D}$$

- Key challenges of excessive gap minimization
  - Let μ_k approach 0 as rapidly as possible
  - Still allow $\mathbf{w}_k$ and $\boldsymbol{\alpha}_k$ to be updated efficiently

Rates of convergence

- Rates when using Euclidean prox-function $d(\boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{\alpha}\|^2$

$$J(\mathbf{w}_k) - D(\boldsymbol{\alpha}_k) \leq \frac{6\mathcal{D}}{(k+1)(k+2)} \frac{\|A\|^2}{\lambda}$$

- But, Euclidean prox-function does not work for M^3N
  - Key issue: cannot maintain factorization in the updates
  - Need to evaluate the smooth objective

$$(g + \mu d)^\star(A^\top \mathbf{w}) = \max_{\boldsymbol{\alpha} \in Q} \left\{ \left\langle A^\top \mathbf{w}_k, \boldsymbol{\alpha} \right\rangle - g(\boldsymbol{\alpha}) - \mu d(\boldsymbol{\alpha}) \right\}$$

  - Maximizer must factorize over the graphical model.
  - Intuition: arithmetic mean of two iid densities is not iid.

Outline

- Objective of max-margin Markov network (M^3N)
- Smoothing for M^3N
- Excessive gap technique in general, and problem for M^3N
- Bregman divergence for prox-function
  - Retain the accelerated rates $\frac{1}{k^2}$
  - Efficient computation by graphical model factorization
- Kernelization
- Conclusion

Using Bregman divergence prox-function

- We show Bregman divergence maintains factorization
  - Intuition: geometric mean of two iid densities is still iid
- We show same $\frac{1}{k^2}$ rates hold for Bregman divergence prox-function

$$Q = \left\{ \boldsymbol{\alpha} : \alpha_{\mathbf{y}}^i \geq 0, \text{ and } \sum_{\mathbf{y}} \alpha_{\mathbf{y}}^i = \frac{1}{n}, \ \forall\ i \right\}$$

$$d(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \sum_{\mathbf{y}} \alpha_{\mathbf{y}}^i \log \alpha_{\mathbf{y}}^i + \log n + \log |\mathcal{Y}|\,,$$

$$J(\mathbf{w}_k) - D(\boldsymbol{\alpha}_k) \leq \frac{6 \log |\mathcal{Y}|}{(k+1)(k+2)} \frac{\max_{i,\mathbf{y}} \|\boldsymbol{\phi}(\mathbf{x}_i, \mathbf{y})\|^2}{\lambda}$$

Comparison

- Resulting rates
  - Ours

$$\max_{i,\mathbf{y}} \|\boldsymbol{\phi}(\mathbf{x}_i, \mathbf{y})\| \sqrt{\frac{6\mathrm{KL}(\boldsymbol{\alpha}^* || \boldsymbol{\alpha}_0)}{\lambda \epsilon}}$$

  - (Collins et al, 2008)

$$\max_{i,\mathbf{y}} \|\boldsymbol{\phi}(\mathbf{x}_i, \mathbf{y})\|^2 \frac{\mathrm{KL}(\boldsymbol{\alpha}^* || \boldsymbol{\alpha}_0)}{\lambda \epsilon}$$

Outline

- Objective of max-margin Markov network (M^3N)
- Smoothing for M^3N
- Excessive gap technique in general, and problem for M^3N
- Bregman divergence for prox-function
  - Retain the accelerated rates $\frac{1}{k^2}$
  - Efficient computation by graphical model factorization
- Kernelization
- Conclusion

Kernelization

- $\mathbf{w}$ enters the objective only via inner products

$$\langle \mathbf{w}, \boldsymbol{\phi}(\mathbf{x}^i, \mathbf{y}) \rangle$$

- So kernelize on $\mathcal{X} \times \mathcal{Y}$
  - Further factorize the kernel onto $\mathcal{X} \times \{\mathcal{Y}_c\}_c$
- Key idea: implicitly represent $\mathbf{w}$ in terms of $\boldsymbol{\beta}$
  - Roughly speaking:

$$\mathbf{w} = \sum_{i=1}^{n} \sum_{\mathbf{y} \in \mathcal{Y}} \beta_{\mathbf{y}}^i \boldsymbol{\phi}(\mathbf{x}^i, \mathbf{y})$$

  - This $\beta_{\mathbf{y}}^i$ factorizes over the graphical model
  - Then $\langle \mathbf{w}, \boldsymbol{\phi}(\mathbf{x}^i, \mathbf{y}) \rangle$ can be computed by using kernels

Conclusion

- Excessive gap technique enjoys accelerated rates $\frac{1}{k^2}$
  - But only shown for Euclidean prox-function
- Euclidean prox-function is problematic for M^3N
  - Does not allow computations to factorize
- We extend prox-function to Bregman divergence
  - Efficient computation by graphical model factorization
  - Improved rates compared with state-of-the-art M^3N solvers
  - Admits kernelization