Caltech

Structured Perceptrons & Structural SVMs

4/6/2017

CS 159: Advanced Topics in Machine Learning

Recall: Sequence Prediction

POS Tags:
Det, Noun, Verb, Adj, Adv, Prep

$\underbrace{\hspace{6em}}_{L=6}$

- Input: $x = (x^1,\ldots,x^M)$
- Predict: $y = (y^1,\ldots,y^M)$
  - Each y^i one of L labels.

- x = "Fish Sleep"
- y = (N, V)

- x = "The Dog Ate My Homework"
- y = (D, N, V, D, N)

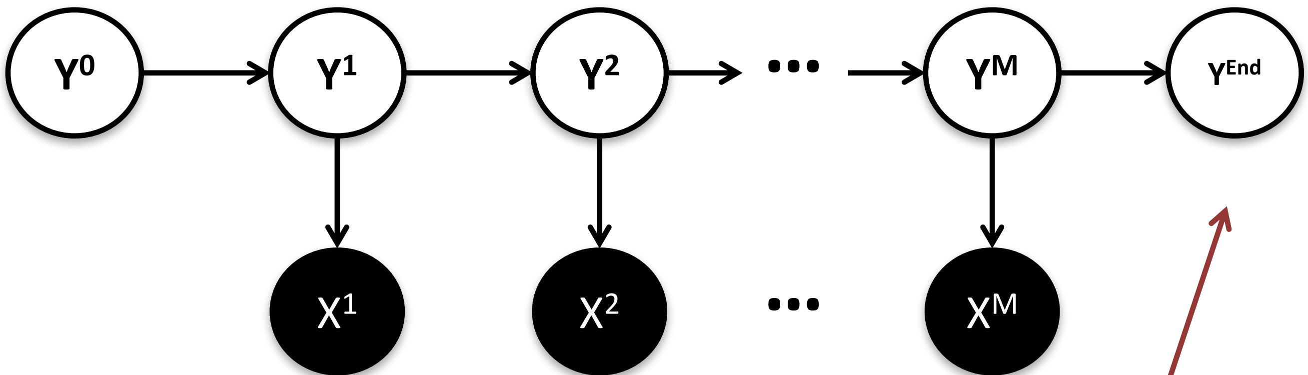
- x = "The Fox Jumped Over The Fence"
- y = (D, N, V, P, D, N)

Recall: 1^{st} Order HMM

- $x = (x^1,x^2,x^4,x^4,\ldots,x^M)$ (sequence of words)
- $y = (y^1,y^2,y^3,y^4,\ldots,y^M)$ (sequence of POS tags)

- $P(x^j|y^j)$ Probability of state y^j generating x^j
- $P(y^{j+1}|y^j)$ Probability of state y^j transitioning to y^{j+1}
- $P(y^1|y^0)$ y^0 is defined to be the Start state
- $P(End|y^M)$ Prior probability of y^M being the final state
  - Not always used

HMM Graphical Model Representation



Optional

$$P\left(x,y\right) = P(End \mid y^{M}) \prod_{i=1}^{M} P(y^{i} \mid y^{i-1}) \prod_{i=1}^{M} P(x^{i} \mid y^{i})$$

Most Common Prediction Problem

- Given input sentence, predict POS Tag seq.

$$h(x) = \operatorname{argmax}_y P(y|x) = \operatorname{argmax} \log P(y|x)$$

$$\log P(y|x) = \sum_i \left[\log P\left(y^j \middle| x^j\right) + \log P(x^j|x^{j-1})\right]$$

- Solve using Viterbi
  – Special case of max product algorithm

Simple Example

- x = “Fish Sleep”
- y = (N,V)

$$F(\mathrm{y},\mathrm{x}) \equiv \log P(y|x) = \sum_i \left[\log P\left(y^j \middle| x^j\right) + \log P(x^j|x^{j-1})\right]$$

Log P(y^j|x^j)

Log P	P(*|N)	P(*|V)
P(Fish|*)	2	1
P(Sleep|*)	1	0

Log P(x^j|x^{j-1})

Log P	P(N|*)	P(V|*)
P(*|N)	-2	1
P(*|V)	2	-2
P(*|Start)	1	-1

New Notation

$$F(y,x) \equiv \sum_{j=1}^{M} \left[w^T \varphi^j(y^j, y^{j-1} \mid x) \right]$$

$$w = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \qquad \varphi^j(a,b \mid x) = \begin{bmatrix} \varphi_1^j(a \mid x) \\ \varphi_2(a,b) \end{bmatrix}$$

- "Unary Features"
- "Pairwise Transition Features"

$$\varphi_1^j(a \mid x) = \begin{bmatrix} 1_{\left[(a=Noun)\wedge\left(x^j='Fish'\right)\right]} \\ 1_{\left[(a=Noun)\wedge\left(x^j='Sleep'\right)\right]} \\ 1_{\left[(a=Verb)\wedge\left(x^j='Fish'\right)\right]} \\ 1_{\left[(a=Verb)\wedge\left(x^j='Sleep'\right)\right]} \end{bmatrix}$$

$$\varphi_2(a,b) = \begin{bmatrix} 1_{\left[(a=Noun)\wedge(b=Start)\right]} \\ 1_{\left[(a=Noun)\wedge(b=Noun)\right]} \\ 1_{\left[(a=Noun)\wedge(b=Verb)\right]} \\ 1_{\left[(a=Verb)\wedge(b=Start)\right]} \\ 1_{\left[(a=Verb)\wedge(b=Noun)\right]} \\ 1_{\left[(a=Verb)\wedge(b=Verb)\right]} \end{bmatrix}$$

New Notation

Duplicate word features for each label.

Noun Class Features

$$\varphi_1^1(Noun \mid \text{"Fish Sleep"}) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\varphi_1^2(Noun \mid \text{"Fish Sleep"}) = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Verb Class Features

$$\varphi_1^1(Verb \mid \text{"Fish Sleep"}) = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\varphi_1^2(Verb \mid \text{"Fish Sleep"}) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\varphi_1^j(a \mid x) = \begin{bmatrix} 1_{\left[(a=Noun)\wedge\left(x^j='Fish'\right)\right]} \\ 1_{\left[(a=Noun)\wedge\left(x^j='Sleep'\right)\right]} \\ 1_{\left[(a=Verb)\wedge\left(x^j='Fish'\right)\right]} \\ 1_{\left[(a=Verb)\wedge\left(x^j='Sleep'\right)\right]} \end{bmatrix}$$

$$\varphi_1^j(a \mid x) = \begin{bmatrix} 1_{[a=1]}\phi_1(x^j) \\ \vdots \\ 1_{[a=L]}\phi_1(x^j) \end{bmatrix}$$

New Notation

One feature for every transition.

$$\varphi_1^j(a \mid x) = \begin{bmatrix} 1_{[(a=Noun)\wedge(x^j='Fish')]} \\ 1_{[(a=Noun)\wedge(x^j='Sleep')]} \\ 1_{[(a=Verb)\wedge(x^j='Fish')]} \\ 1_{[(a=Verb)\wedge(x^j='Sleep')]} \end{bmatrix}$$

$$\varphi_2(a,b) = \begin{bmatrix} 1_{[(a=Noun)\wedge(b=Start)]} \\ 1_{[(a=Noun)\wedge(b=Noun)]} \\ 1_{[(a=Noun)\wedge(b=Verb)]} \\ 1_{[(a=Verb)\wedge(b=Start)]} \\ 1_{[(a=Verb)\wedge(b=Noun)]} \\ 1_{[(a=Verb)\wedge(b=Verb)]} \end{bmatrix}$$

$$\varphi_2(Noun, Start) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\varphi_2(Verb, Start) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\varphi_2(Verb, Noun) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$F(y,x) \equiv \sum_{j=1}^{M} \left[w^T \varphi^j (y^j, y^{j-1} \mid x) \right]$$

$$w = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$$

$$\varphi^j(a,b \mid x) = \begin{bmatrix} \varphi_1^j(a \mid x) \\ \varphi_2^j(a,b) \end{bmatrix}$$

$$w_1 = \begin{bmatrix} 2 \\ 1 \\ 1 \\ 0 \end{bmatrix} \quad \varphi_1^j(a \mid x) = \begin{bmatrix} 1_{\left[(a=Noun)\wedge\left(x^j='Fish'\right)\right]} \\ 1_{\left[(a=Noun)\wedge\left(x^j='Sleep'\right)\right]} \\ 1_{\left[(a=Verb)\wedge\left(x^j='Fish'\right)\right]} \\ 1_{\left[(a=Verb)\wedge\left(x^j='Sleep'\right)\right]} \end{bmatrix}$$

Old Notation:

	P(*|N)	P(*|V)
P(Fish|*)	2	1
P(Sleep|*)	1	0

Old Notation:

	P(N|*)	P(V|*)
P(*|N)	-2	1
P(*|V)	2	-2
P(*|Start)	1	-1

$$w_2 = \begin{bmatrix} 1 \\ -2 \\ 2 \\ -1 \\ 1 \\ -2 \end{bmatrix} \quad \varphi_2(a,b) = \begin{bmatrix} 1_{\left[(a=Noun)\wedge(b=Start)\right]} \\ 1_{\left[(a=Noun)\wedge(b=Noun)\right]} \\ 1_{\left[(a=Noun)\wedge(b=Verb)\right]} \\ 1_{\left[(a=Verb)\wedge(b=Start)\right]} \\ 1_{\left[(a=Verb)\wedge(b=Noun)\right]} \\ 1_{\left[(a=Verb)\wedge(b=Verb)\right]} \end{bmatrix}$$

Recap: 1st Order Sequential Model

POS Tags:
Det, Noun, Verb, Adj, Adv, Prep

$\underbrace{\hspace{6em}}_{L=6}$

- Input: $x = (x^1,\ldots,x^M)$
- Predict: $y = (y^1,\ldots,y^M)$
  - Each y^i one of L labels.
- Linear Model w.r.t. pairwise features $\phi^j(a,b|x)$:

Encodes Structure

- Prediction via maximizing F:

$$h(x) = \text{argmax}_y F(y,x) = \text{argmax}_y w^T \Psi(y,x)$$

x = "Fish Sleep"   y = (N,V)

$$w_1 = \begin{bmatrix} 2 \\ 1 \\ 1 \\ 0 \end{bmatrix} \quad \varphi_1^j(a \mid x) = \begin{bmatrix} 1_{[(a=Noun)\wedge(x^j='Fish')]} \\ 1_{[(a=Noun)\wedge(x^j='Sleep')]} \\ 1_{[(a=Verb)\wedge(x^j='Fish')]} \\ 1_{[(a=Verb)\wedge(x^j='Sleep')]} \end{bmatrix} \quad w_2 = \begin{bmatrix} 1 \\ -2 \\ 2 \\ -1 \\ 1 \\ -2 \end{bmatrix} \quad \varphi_2^j(a,b) = \begin{bmatrix} 1_{[(a=Noun)\wedge(b=Start)]} \\ 1_{[(a=Noun)\wedge(b=Noun)]} \\ 1_{[(a=Noun)\wedge(b=Verb)]} \\ 1_{[(a=Verb)\wedge(b=Start)]} \\ 1_{[(a=Verb)\wedge(b=Noun)]} \\ 1_{[(a=Verb)\wedge(b=Verb)]} \end{bmatrix}$$

$$F(y=(N,V), x=\text{"Fish Sleep"}) = w_1^T \varphi_1^1(N,x) + w_2^T \varphi_2(N,Start) + w_1^T \varphi_1^2(V,x) + w_2^T \varphi_2(V,N)$$
$$= w_{1,1} + w_{2,1} + w_{1,4} + w_{2,5} = 2 + 1 + 0 + 1 = 4$$

Prediction: $\underset{y}{\operatorname{argmax}} F(y,x)$

y	F(y,x)
(N,N)	2+1+1-2 = 2
(N,V)	2+1+0+1 = 4
(V,N)	1-1+1+2 = 3
(V,V)	1-1+0-2 = -2

Why New Notation?

- Easier to reason about:
  - Computing predictions
  - Learning (linear model!)
  - Extensions (just generalize ϕ)

$$\varphi_1^j(a \mid x) = \begin{bmatrix} 1_{[(a=Noun)\wedge(x^j='Fish')]} \\ 1_{[(a=Noun)\wedge(x^j='Sleep')]} \\ 1_{[(a=Verb)\wedge(x^j='Fish')]} \\ 1_{[(a=Verb)\wedge(x^j='Sleep')]} \end{bmatrix}$$

$$\varphi_2(a,b) = \begin{bmatrix} 1_{[(a=Noun)\wedge(b=Start)]} \\ 1_{[(a=Noun)\wedge(b=Noun)]} \\ 1_{[(a=Noun)\wedge(b=Verb)]} \\ 1_{[(a=Verb)\wedge(b=Start)]} \\ 1_{[(a=Verb)\wedge(b=Noun)]} \\ 1_{[(a=Verb)\wedge(b=Verb)]} \end{bmatrix}$$

$$w = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \qquad \varphi^j(a,b \mid x) = \begin{bmatrix} \varphi_1^j(a \mid x) \\ \varphi_2(a,b) \end{bmatrix}$$

Generalizes Multiclass

- Stack weight vectors for each class:

$$F(y, x) \equiv w^T \Psi(y, x)$$

$$w = \begin{bmatrix} w_1 \\ w_1 \\ \vdots \\ w_K \end{bmatrix} \qquad \Psi(y, x) = \begin{bmatrix} 1_{[y=1]} x \\ 1_{[y=2]} x \\ \vdots \\ 1_{[y=K]} x \end{bmatrix}$$

$$h(x) = \mathrm{argmax}_y w^T \Psi(y, x) = \mathrm{argmax}_y w_y^T x$$

Learning for Structured Prediction

Perceptron Learning Algorithm
(Linear Classification Model)

- $w^1 = 0$
- For t = 1 ….
  - Receive example (x,y)
  - If $h(x|w^t) = y$
    - $w^{t+1} = w^t$
  - Else
    - $w^{t+1} = w^t + yx$

$$h(x) = \mathrm{sign}(w^T x)$$

Training Set:

$$S = \{(x_i, y_i)\}_{i=1}^{N}$$

$$y \in \{+1, -1\}$$

Go through training set in arbitrary order (e.g., randomly)

Structured Perceptron
(Linear Classification Model)

- $w^1 = 0$
- For t = 1 ....
  - Receive example (x,y)
  - If $h(x|w^t) = y$
    - $w^{t+1} = w^t$
  - Else
    - $w^{t+1} = w^t + \Psi(y,x)$

Only thing that changes!

$$h(x) = \text{argmax}_{y'} w^T \Psi(y', x)$$

Training Set:

$S = \{(x_i, y_i)\}$

y_i structured

Go through training set in arbitrary order (e.g., randomly)

Structured Perceptron

Method	Error rate/%	Numits
Perc, avg, cc=0	2.93	10
Perc, noavg, cc=0	3.68	20
Perc, avg, cc=5	3.03	6
Perc, noavg, cc=5	4.04	17
ME, cc=0	3.4	100
ME, cc=5	3.28	200

Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms
Michael Collins, EMNLP 2002
http://www.cs.columbia.edu/~mcollins/papers/tagperc.pdf

Limitations of Perceptron

- Not all mistakes are created equal
  - One POS tag wrong as bad as five!
  - Even worse for more complicated problems

Comparison

Method	HMM	CRF	Perceptron	SVM
Error	9.36	5.17	5.94	5.08

Large Margin Methods for Structured and Interdependent Output Variables
Ionnis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, Yasemin Altun
Journal of Machine Learning Research, Volume 6, Pages 1453-1484

Hamming Loss

- Hamming Loss: $$\ell(y,x,F) = \sum_{j=1}^{M} 1_{\left[h(x)^j \neq y^j\right]}$$

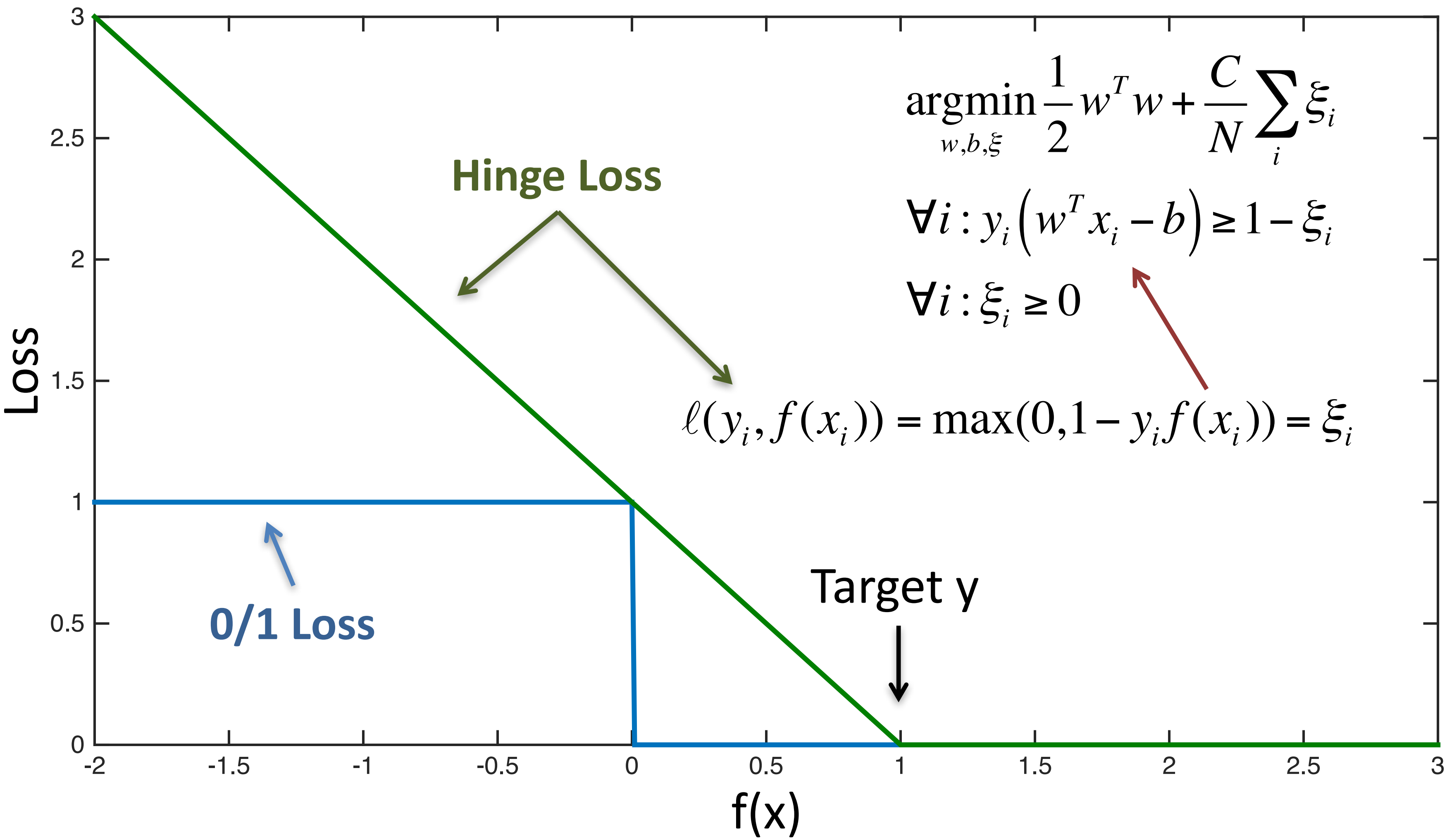
- True y = (D,N,V,D,N)
  - y' = (D,N,V,N,N)
  - y" = (V,D,N,V,V)

y" has much worse hamming loss
(loss of 5 vs loss of 1)

(But not continuous!)

Need to define continuous surrogate of Hinge Loss!

Original Hinge Loss
(Support Vector Machine)

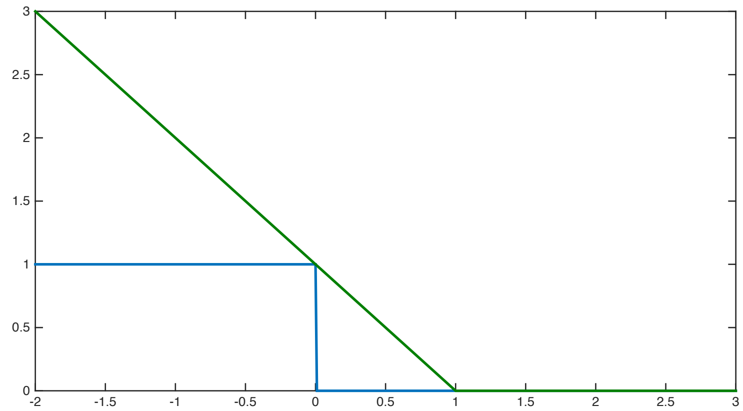


Property of Hinge Loss

$$\underset{w,b,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i : y_i \left(w^T x_i - b \right) \geq 1 - \xi_i$$

$$\forall i : \xi_i \geq 0$$



$h(x) = \underset{y \in \{-1,+1\}}{\operatorname{argmax}}\, y f(x) = sign(f(x)) \quad \rightarrow \quad \xi_i \geq 1_{[h(x_i) \neq y_i]}$
$\ell(y_i, f(x_i)) = \max(0, 1 - y_i f(x_i)) = \xi_i$

Hinge loss = continuous upper bound on 0/1 loss

Hamming Hinge Loss
(Structural SVM)

$$\operatorname*{argmin}_{w,\xi} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

Sometimes normalize by M

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i \qquad \forall i : \xi_i \geq 0$$

$$h(x) = \operatorname*{argmax}_y F(y, x) \quad \rightarrow \quad F(y_i, x_i) - F(h(x_i), x_i) \leq 0$$

Learned Predictor

$$\rightarrow \quad \xi_i \geq \sum_j 1_{\left[h(x_i)^j \neq y_i^j\right]}$$

$$\ell(y_i, x_i, F) = \max_{y'} \left\{ \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \left(F(y_i, x_i) - F(y', x_i)\right) \right\} = \xi_i$$

Continuous upper bound on Hamming Loss!

Hamming Hinge Loss

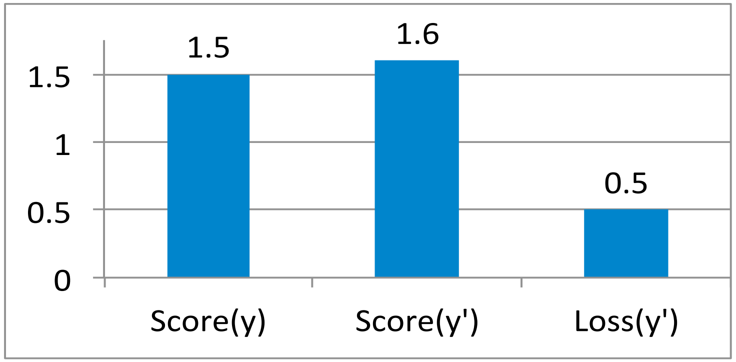
$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : \underbrace{F(y_i, x_i)}_{\text{Score}(y_i)} - \underbrace{F(y', x_i)}_{\text{Score}(y')} \geq \underbrace{\frac{1}{M_i} \sum_j 1_{\left[y'^j \neq y_i^j\right]}}_{\text{Loss}(y')} \underbrace{- \xi_i}_{\text{Slack}} \qquad \forall i : \xi_i \geq 0$$

Suppose for incorrect $y' = h(x_i)$:

Then:

$$\boxed{\xi_i = 0.6 \geq 0.5 = \frac{1}{M_i} \sum_j 1_{\left[h(x_i)^j \neq y_i^j\right]}}$$



Slack variable upper bounds Hamming Loss!

Structural SVM

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

Sometimes normalize by M

"Slack"

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i \qquad \forall i : \xi_i \geq 0$$

Consider: $y' = \underset{y}{\operatorname{argmax}} F(y, x)$ ➔ $F(y_i, x_i) - F(y', x_i) \leq 0$

Prediction of Learned Model

Slack is continuous upper bound on Hamming Loss!

$y' \neq y_i$ ➔ $\xi_i \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]}$

$y' = y_i$ ➔ $\xi_i \geq 0$

Reduction to Independent Multiclass

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i \qquad \forall i : \xi_i \geq 0$$

Suppose: $F(y,x) \equiv \sum_{j=1}^{M} \left[w^T \varphi^j (y^j \mid x) \right]$

$$\varphi^j (y^j \mid x) = \begin{bmatrix} 1_{\left[y^j = 1\right]} \phi_1(x^j) \\ \vdots \\ 1_{\left[y^j = L\right]} \phi_1(x^j) \end{bmatrix}$$

No pairwise features.

Stack features $\varphi_1(x^j)$ L times

$$\forall i, j, a : w_{y_i^j}^T \phi_1(x^j) - w_a^T \phi_1(x^j) \geq 1 - \xi_{ij}$$

Decompose constraints to multiclass hinge loss per token!

Example 1

$$\operatorname*{argmin}_{w,\xi} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i \qquad \forall i : \xi_i \geq 0$$

x_i = "Fish Sleep"

y_i = (N,V)

$\xi_i = 0$

y'	F(y',x_i)	F(y_i,x_i) – F(y',x_i)	Loss
(N,N)	2	2	1
(N,V)	4	0	0
(V,N)	1	3	2
(V,V)	1	3	1

Example 2

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i \qquad \forall i : \xi_i \geq 0$$

x_i = “Fish Sleep”

y_i = (N,V)

$\xi_i = 2$

y’	$F(y',x_i)$	$F(y_i,x_i) - F(y',x_i)$	Loss
(N,N)	4	-1	1
(N,V)	3	0	0
(V,N)	0	3	2
(V,V)	1	2	1

Example 3

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j \right]} - \xi_i \qquad \forall i : \xi_i \geq 0$$

x_i = “Fish Sleep”

y_i = (N,V)

$\xi_i = 1$

y’	F(y’,x_i)	F(y_i,x_i) – F(y’,x_i)	Loss
(N,N)	2	2	1
(N,V)	4	0	0
(V,N)	3	1	2
(V,V)	1	3	1

When is Slack Positive?

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i \qquad \forall i : \xi_i \geq 0$$

- Whenever margin not big enough!

$$\xi_i > 0 \iff \exists y' : \; F(y_i, x_i) - F(y', x_i) < \sum_j 1_{\left[y'^j \neq y_i^j\right]}$$

$$\xi_i = \max_{y'} \left\{ \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \left(F(y_i, x_i) - F(y', x_i) \right) \right\} = \ell(y_i, x_i, F)$$

Verify that above definition ≥0

Structural SVM

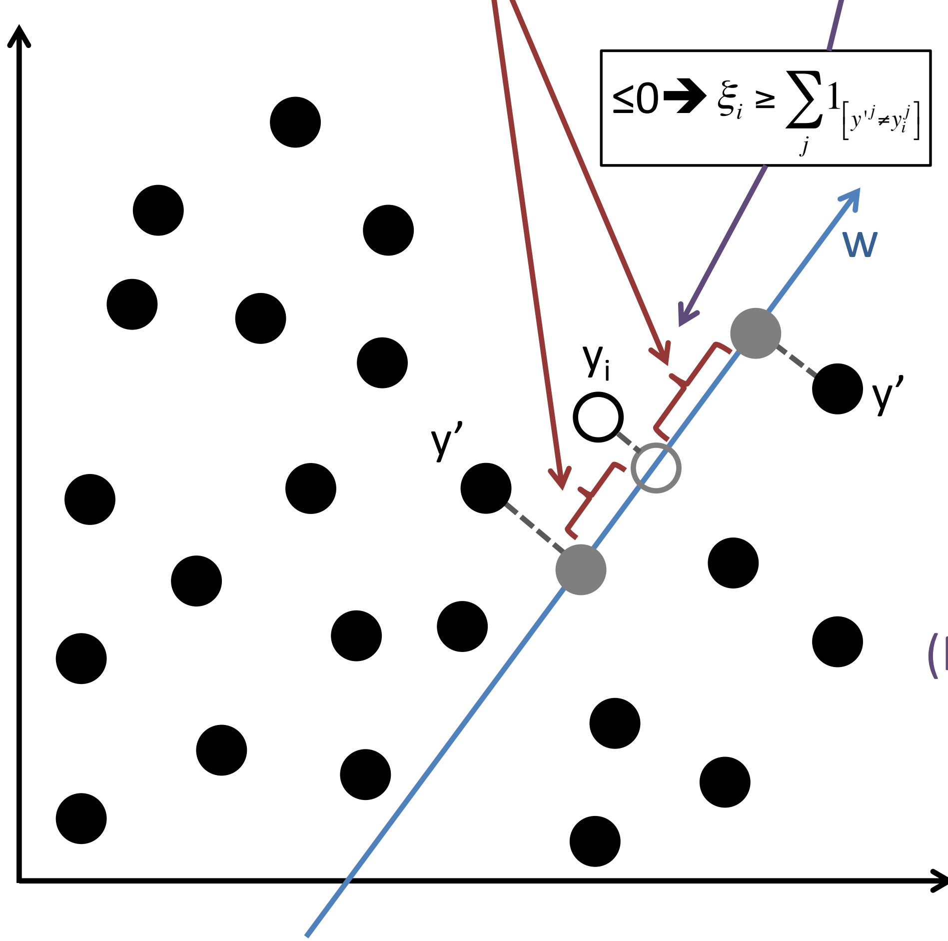
Geometric Interpretation

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i \qquad \forall i : \xi_i \geq 0$$

$$\leq 0 \Rightarrow \xi_i \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]}$$

$$F(y,x) = w^T \underbrace{\Psi(y,x)}_{\text{High Dimensional Point}}$$



Size of Margin

vs

Size of Margin Violations

(C controls trade-off)

(Margin scaled by Hamming Loss)

Structural SVM Training

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\underbrace{\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i}_{\text{Often Exponentially Many!}} \qquad \forall i : \xi_i \geq 0$$

- Strictly convex optimization problem
  - Same form as standard SVM optimization
  - Easy right?
- **Intractable # of constraints!**

Structural SVM Training

$$\forall y': \quad F(y_i, x_i) \geq F(y', x_i) + \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i$$

- The trick is to not enumerate all constraints.
- Only solve the SVM objective over a small subset of constraints (**working set**).
  - Efficient!
- But some constraints might be violated.

Example

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i \qquad \forall i : \xi_i \geq 0$$

x_i = "Fish Sleep"

y_i = (N,V)

$\xi_i = 0$

y'	$F(y',x_i)$	$F(y_i,x_i) - F(y',x_i)$	Loss
(N,N)	2	2	1
(N,V)	4	0	0
(V,N)	3	1	2
(V,V)	1	3	1

Approximate Hinge Loss

- Choose tolerate ε>0:

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i - \varepsilon \qquad \forall i : \xi_i \geq 0$$

Consider:	$y' = \underset{y}{\operatorname{argmax}} F(y, x)$ Prediction of Learned Model	➔	$F(y_i, x_i) - F(y', x_i) \leq 0$
Slack is continuous upper bound on Hamming Loss - ε!	$y' \neq y_i$	➔	$\xi_i \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \varepsilon$
	$y' = y_i$	➔	$\xi_i \geq 0$

Example

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i - \varepsilon \qquad \forall i : \xi_i \geq 0$$

x_i = “Fish Sleep”

y_i = (N,V)

$\xi_i = 0$

$\varepsilon = 1$

y’	F(y’,x_i)	F(y_i,x_i) – F(y’,x_i)	Loss	
(N,N)	2	2	1	✓
(N,V)	4	0	0	✓
(V,N)	3	1	2	✗
(V,V)	1	3	1	✓

Structural SVM Training

- **STEP 0:** Specify tolerance ε
- **STEP 1:** Solve SVM objective function using only working set of constraints **W** (initially empty). The trained model is w.
- **STEP 2:** Using w, find the y' whose constraint is most violated.
- **STEP 3:** If constraint is violated by more than ε, add it to **W**.

Constraint Violation Formula:

$$\left(\frac{1}{M_i}\sum_j 1_{\left[y'^j \neq y_i^j\right]} + \xi_i\right) - \left(F(y_i, x_i) - F(y', x_i)\right) \geq \varepsilon$$

- **Repeat STEP 1-3** until no additional constraints are added. Return most recent model w trained in STEP 1.

*This is known as a "cutting plane" method.

Example

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

Choose ε=0.1

$$\forall i, y' \in W_i : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i - \varepsilon \qquad \forall i : \xi_i \geq 0$$

Init: $W_i = \varnothing$

Solve: $\xi_i = 0$

x_i = "Fish Sleep"

y_i = (N,V)

y'	$F(y',x_i)$	$F(y_i,x_i) - F(y',x_i)$	Loss	Viol.
(N,N)	0	0	1	1
(N,V)	0	0	0	0
(V,N)	0	0	2	2
(V,V)	0	0	1	1

Constraint Violation: Loss – Slack – (F(y,x)-F(y',x)) = Viol

Example

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

Choose ε=0.1

$$\forall i, y' \in W_i : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i - \varepsilon \qquad \forall i : \xi_i \geq 0$$

Update: $W_i = \{(V,N)\}$

Solve: $\xi_i = 0$

x_i = "Fish Sleep"

y_i = (N,V)

y'	$F(y',x_i)$	$F(y_i,x_i) - F(y',x_i)$	Loss	Viol.	
(N,N)	0	0	1	1	✗
(N,V)	0	0	0	0	✓
(V,N)	0	0	2	2	✗
(V,V)	0	0	1	1	✗

Constraint Violation: Loss – Slack – (F(y,x)-F(y',x)) = Viol

Example

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

Choose ε=0.1

$$\forall i, y' \in W_i : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i - \varepsilon \qquad \forall i : \xi_i \geq 0$$

Update: $W_i = \{(V,N)\}$

Solve: $\xi_i = 0.5$

x_i = "Fish Sleep"

y_i = (N,V)

y'	$F(y',x_i)$	$F(y_i,x_i) - F(y',x_i)$	Loss	Viol.
(N,N)	0.7	0.2	1	0.2
(N,V)	0.9	0	0	0
(V,N)	-0.6	1.5	2	0
(V,V)	0	0.9	1	0.4

Constraint Violation: Loss – Slack – (F(y,x)-F(y',x)) = Viol

Example

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

Choose ε=0.1

$$\forall i, y' \in W_i : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i - \varepsilon \qquad \forall i : \xi_i \geq 0$$

Update: $W_i = \{(V,N),(N,N)\}$

Solve: $\xi_i = 0.5$

x_i = "Fish Sleep"

y_i = (N,V)

y'	$F(y',x_i)$	$F(y_i,x_i) - F(y',x_i)$	Loss	Viol.
(N,N)	0.7	0.2	1	0.2
(N,V)	0.9	0	0	0
(V,N)	-0.6	1.5	2	0
(V,V)	0	0.9	1	0.4

Constraint Violation: Loss – Slack – (F(y,x)-F(y',x)) = Viol

Example

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' \in W_i : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i - \varepsilon \qquad \forall i : \xi_i \geq 0$$

Choose ε=0.1

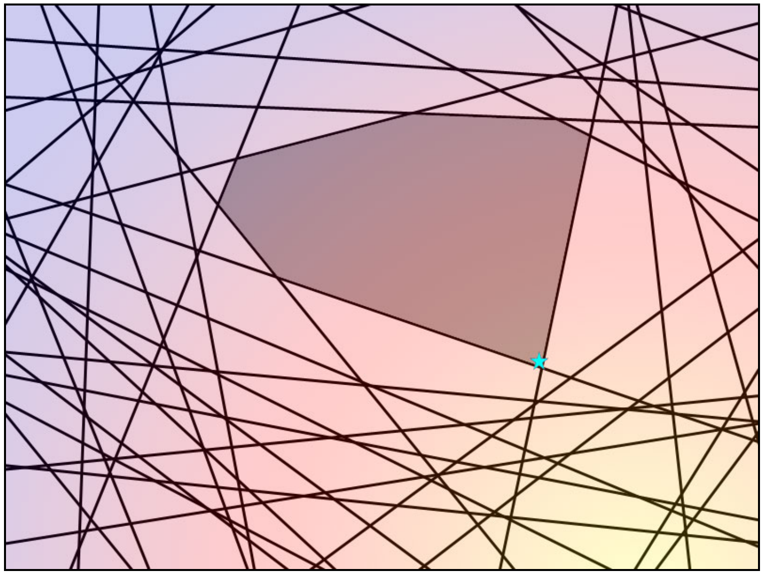
Solve: $\xi_i = 0.55$

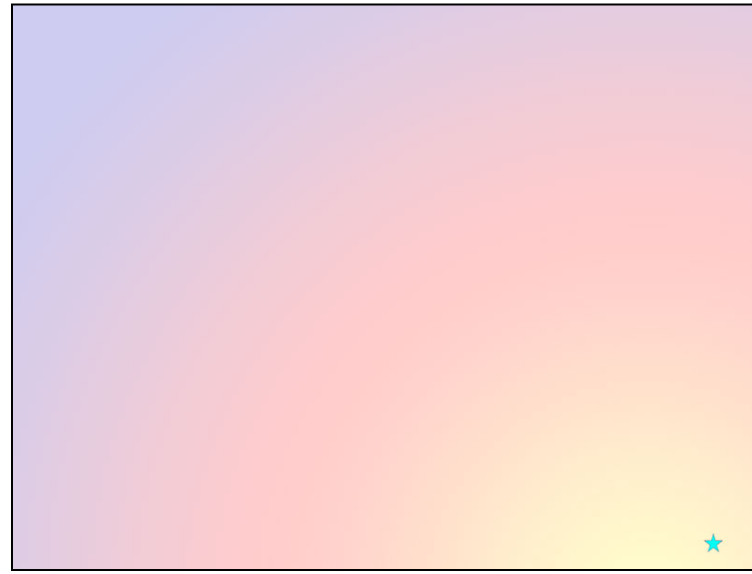
No constraint is violated by more than ε

			Loss	Viol.
			1	0
			0	0
			2	0
(V,V)	-0.05	0.95	1	0.05

Constraint Violation: Loss – Slack – (F(y,x)-F(y’,x)) = Viol

Geometric Example





Naïve SVM Problem

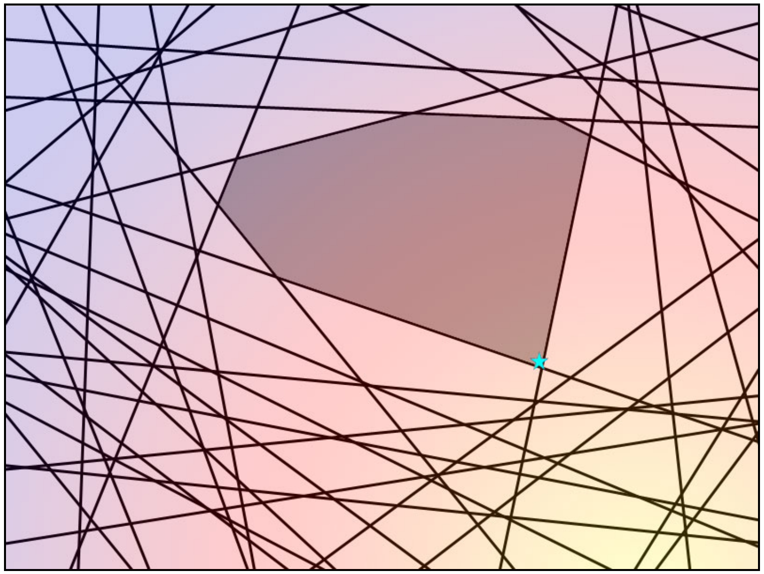
- Exponential constraints
- Most are dominated by a small set of "important" constraints

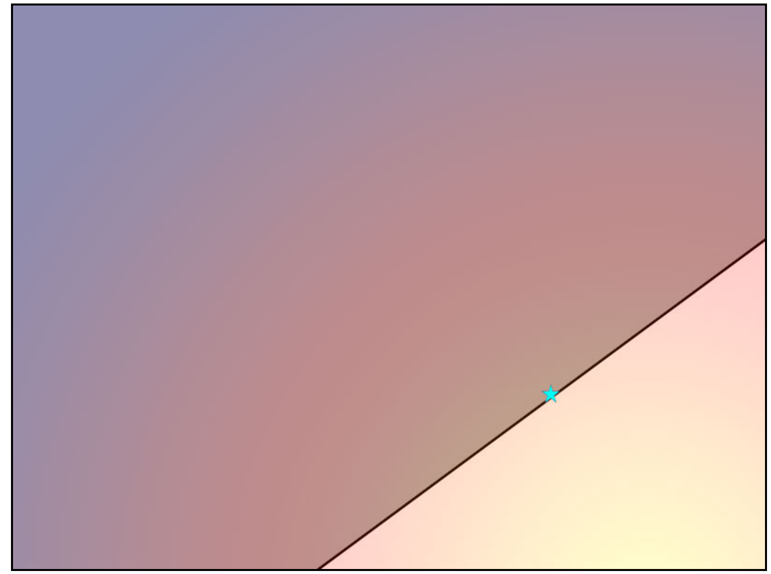
Structural SVM Approach

- Repeatedly finds the next most violated constraint…
- …until set of constraints is a good approximation.

*This is known as a "cutting plane" method.

Geometric Example





Naïve SVM Problem

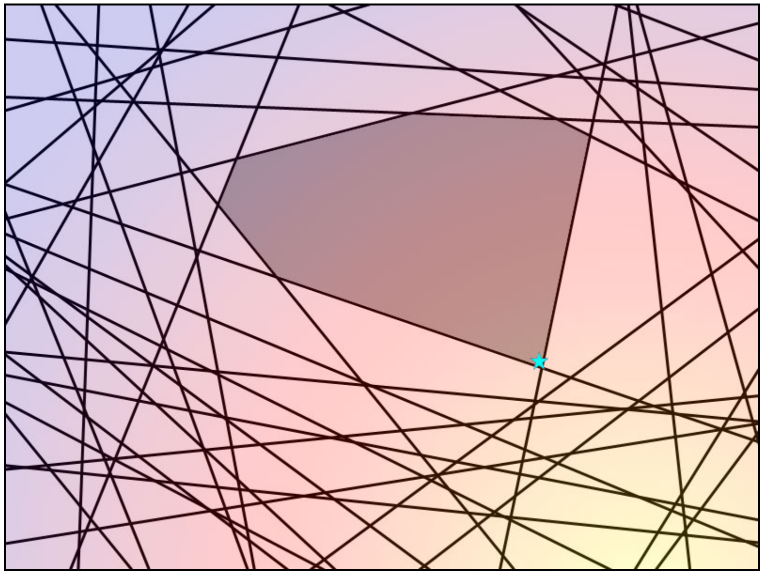
- Exponential constraints
- Most are dominated by a small set of "important" constraints

Structural SVM Approach

- Repeatedly finds the next most violated constraint…
- …until set of constraints is a good approximation.

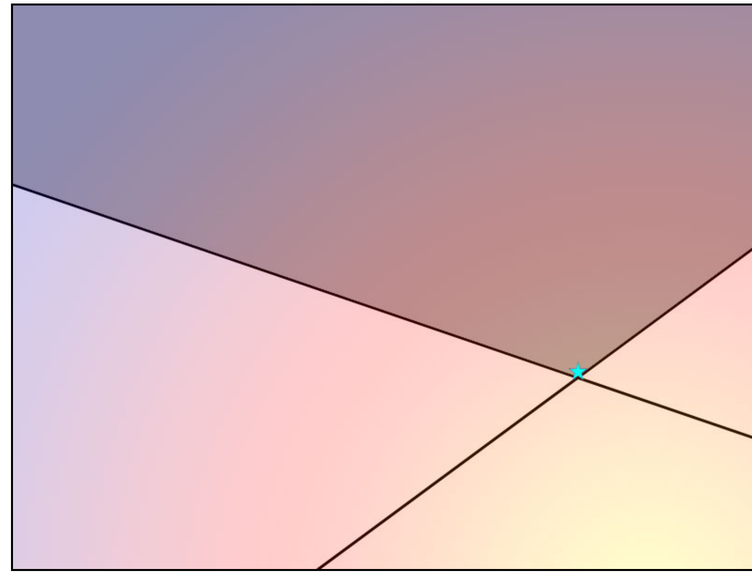
*This is known as a "cutting plane" method.

Geometric Example



Naïve SVM Problem

- Exponential constraints
- Most are dominated by a small set of "important" constraints

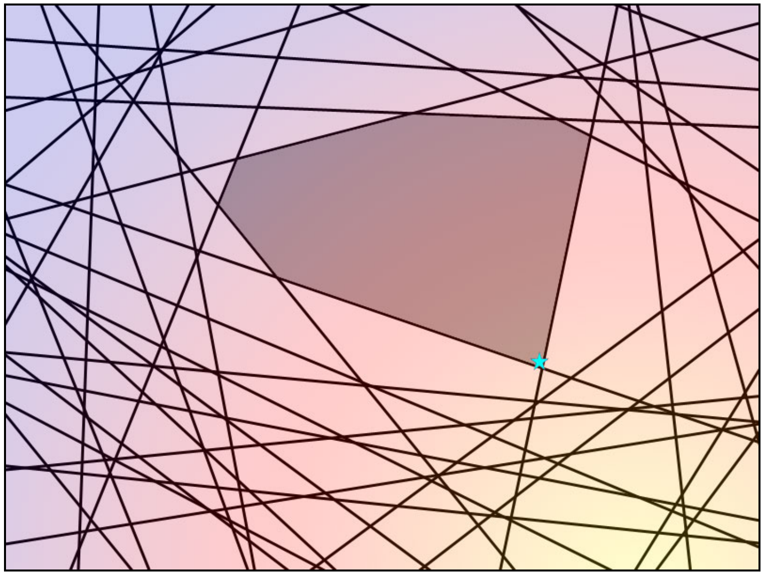


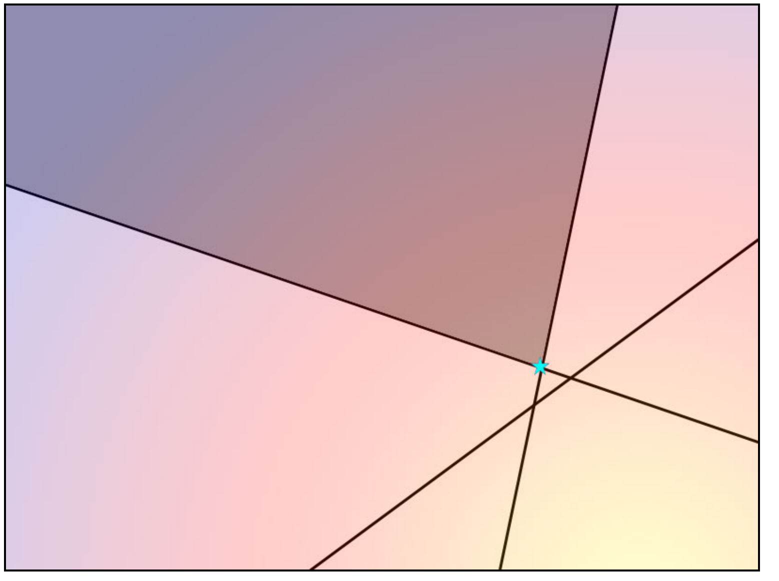
Structural SVM Approach

- Repeatedly finds the next most violated constraint…
- …until set of constraints is a good approximation.

*This is known as a "cutting plane" method.

Geometric Example





Naïve SVM Problem

- Exponential constraints
- Most are dominated by a small set of "important" constraints

Structural SVM Approach

- Repeatedly finds the next most violated constraint…
- …until set of constraints is a good approximation.

*This is known as a "cutting plane" method.

Linear Convergence Rate

- Guarantee for any ε>0:

$$\underset{w,\xi}{\operatorname{argmin}} \frac{1}{2} w^T w + \frac{C}{N} \sum_i \xi_i$$

$$\forall i, y' : F(y_i, x_i) - F(y', x_i) \geq \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i - \varepsilon$$

$$\forall i : \xi_i \geq 0$$

- Terminates after #iterations: $O\left(\frac{1}{\varepsilon}\right)$

Proof found in:
http://www.cs.cornell.edu/people/tj/publications/joachims_etal_09a.pdf

Finding Most Violated Constraint

- A constraint is violated when:

$$F(y',x_i) - F(y_i,x_i) + \sum_j 1_{\left[y'^j \neq y_i^j\right]} - \xi_i > 0$$

- Finding most violated constraint reduces to

$$\underset{y'}{\operatorname{argmax}}\, F(y',x_i) + \sum_j 1_{\left[y'^j \neq y_i^j\right]}$$

"Loss augmented inference"

- Highly related to prediction:

$$\underset{y}{\operatorname{argmax}}\, F(y,x_i)$$

"Augmented" Scoring Function

$$F(y, x_i) \equiv \sum_{j=1}^{M} \left[w^T \varphi^j (y^j, y^{j-1} \mid x_i) \right]$$

Goal:

$$\operatorname*{argmax}_{y'} F(y', x_i) + \sum_j 1_{\left[y'^j \neq y_i^j\right]}$$

Solve Using Viterbi!

$$\tilde{F}(y, x_i, y_i) \equiv \sum_{j=1}^{M} \left[\tilde{w}^T \tilde{\varphi}^j (y^j, y^{j-1} \mid x_i, y_i) \right]$$

$$\tilde{\varphi}^j (a, b \mid x_i, y_i) = \begin{bmatrix} \varphi^j (a, b \mid x_i) \\ 1_{\left[a \neq y_i^j\right]} \end{bmatrix}$$

Additional Unary Feature!

$$\tilde{w} = \begin{bmatrix} w \\ 1 \end{bmatrix}$$

Goal: $\operatorname*{argmax}_{y'} \tilde{F}(y', x_i, y_i)$

Structural SVM Recipe

- Feature map: $\Psi(y, x)$
- Inference: $h(x) = \mathrm{argmax}_y F(y, x) \equiv w^T \Psi(y, x)$
- Loss function: . $\Delta_i\,(y)$
- Loss-augmented: $\mathrm{argmax}_y w^T \Psi(y, x)$+. $\Delta_i\,(y)$

  (most violated constraint)