Midterm Exam Review
  +
Structured Perceptron
  +
Structured SVM
Reminders

• Midterm Exam
  – Thu, Oct. 17 at 6:30pm – 8:00pm

• Homework 3: Structured SVM
  – Out: Sat, Sep. 28
  – Due: Sat, Oct. 12 at 11:59pm
MIDTERM EXAM LOGISTICS
Midterm Exam

• **Time / Location**
  – **Time:** Evening Exam
    *Thu, Oct. 17 at 6:30pm – 8:00pm*
  – **Room:** Hamburg Hall A301
  – **Seats:** There will be **assigned seats.** Please arrive early to find yours.
  – Please watch Piazza carefully for announcements

• **Logistics**
  – Covered material: Lecture 1 – Lecture 13
  – Format of questions:
    • Multiple choice
    • True / False (with justification)
    • Derivations
    • Short answers
    • Interpreting figures
    • Implementing algorithms on paper
  – No electronic devices
  – You are allowed to **bring** one 8½ x 11 sheet of notes (front and back)
Midterm Exam

• **Advice (for during the exam)**
  – Solve the easy problems first
    (e.g. multiple choice before derivations)
    • if a problem seems extremely complicated you’re likely missing something
  – Don’t leave any answer blank!
  – If you make an assumption, write it down
  – If you look at a question and don’t know the answer:
    • we probably haven’t told you the answer
    • but we’ve told you enough to work it out
    • imagine arguing for some answer and see if you like it
Topics for Midterm Exam

• Search-Based Structured Prediction
  – Reductions to Binary Classification
  – Learning to Search
  – RNN-LMs
  – seq2seq models

• Graphical Model Representation
  – Directed GMs vs. Undirected GMs vs. Factor Graphs
  – Bayesian Networks vs. Markov Random Fields vs. Conditional Random Fields

• Graphical Model Learning
  – Fully observed Bayesian Network learning
  – Fully observed MRF learning
  – Fully observed CRF learning
  – Parameterization of a GM
  – Neural potential functions

• Exact Inference
  – Three inference problems:
    (1) marginals
    (2) partition function
    (3) most probably assignment
  – Variable Elimination
  – Belief Propagation (sum-product and max-product)
  – MAP Inference via MILP
SAMPLE QUESTIONS
Sample Questions

Learning to Search

Suppose you are training a seq2seq model for supervised POS Tagging.
• Let the inputs to the encoder be $e_1, e_2, e_3, \ldots$
• Let the inputs to the decoder be $d_1, d_2, d_3, \ldots$
• Let the outputs of the decoder be $o_1, o_2, o_3, \ldots$

1. (1 point) **Short Answer**: Describe in words what the inputs to the encoder would be. Assume you are training with Teacher Forcing.

2. (1 point) **Short Answer**: Describe in words what the inputs of the decoder would be. Assume you are training with Teacher Forcing.

3. (1 point) **Short Answer**: Describe in words what the outputs of the decoder would be. Assume you are training with Teacher Forcing.
Sample Questions

Learning to Search

Suppose you are training a seq2seq model for supervised POS Tagging.
• Let the inputs to the encoder be \( e_1, e_2, e_3, \ldots \)
• Let the inputs to the decoder be \( d_1, d_2, d_3, \ldots \)
• Let the outputs of the decoder be \( o_1, o_2, o_3, \ldots \)

4. (1 point) **Short Answer**: Describe in words what the inputs to the encoder would be. Assume you are training with Scheduled Sampling. *(If your answer is the same as for Teacher Forcing, simply write “same”.)*

5. (1 point) **Short Answer**: Describe in words what the inputs of the decoder would be. Assume you are training with Scheduled Sampling. *(If your answer is the same as for Teacher Forcing, simply write “same”.)*

6. (1 point) **Short Answer**: Describe in words what the outputs of the decoder would be. Assume you are training with Scheduled Sampling. *(If your answer is the same as for Teacher Forcing, simply write “same”.)*
Sample Questions

6 Factor Graphs

Figure 4: A factor graph over three binary random variables $A$, $B$, $C$, i.e. sampled values $a$, $b$, $c$ from the random variables are in $\{0, 1\}$. Assume the factors are named $\psi_A(a)$, $\psi_{A,B}(a,b)$, $\psi_{A,B,C}(a,b,c)$, and $\psi_C(c)$.

1. (2 points) **Short answer:** Consider the factor graph in Figure 4. Using the given factor names, write the partition function $Z$ that ensures the joint probability distribution $p(a, b, c)$ sums-to-one.
2. (2 points) **Short answer:** Using the given factor names, write the joint probability mass function \( p(a, b, c) \) defined by the factor graph shown in Figure 4. *You may include the term \( Z \) directly in your answer—no need to copy it from above.*
Sample Questions

6 Factor Graphs

Figure 4: A factor graph over three binary random variables $A$, $B$, $C$, i.e. sampled values $a$, $b$, $c$ from the random variables are in $\{0, 1\}$. Assume the factors are named $\psi_A(a)$, $\psi_{A,B}(a,b)$, $\psi_{A,B,C}(a,b,c)$, and $\psi_C(c)$.

3. (2 points) **Drawing:** Suppose we have a joint probability distribution that factorizes as below:

$$p(w, x, y, z) \propto \psi_X(x)\psi_{X,Y}(x, y)\psi_{X,Y,Z}(x, y, z)\psi_{W,Z}(w, z)\psi_{Y,Z}(y, z)$$

where $\propto$ denotes *proportional to*. Draw the factor graph corresponding to this factorization of the joint distribution.
6 Factor Graphs

Figure 4: A factor graph over three binary random variables A, B, C, i.e. sampled values a, b, c from the random variables are in \{0, 1\}. Assume the factors are named $\psi_A(a)$, $\psi_{A,B}(a, b)$, $\psi_{A,B,C}(a, b, c)$, and $\psi_C(c)$.

3. (2 points) **Drawing:** Suppose we have a joint probability distribution that factorizes as below:

$$p(w, x, y, z) \propto \psi_X(x)\psi_{X,Y}(x, y)\psi_{X,Y,Z}(x, y, z)\psi_{W,Z}(w, z)\psi_{Y,Z}(y, z)$$

where $\propto$ denotes proportional to. Draw the factor graph corresponding to this factorization of the joint distribution.
7 Inference in Graphical Models

Consider yet another factor graph consisting of two random variables $Q \in \{\text{red}, \text{green}, \text{blue}\}$, $R \in \{\text{pencil}, \text{crayon}\}$. Suppose we have the following factors:

<table>
<thead>
<tr>
<th>$Q$</th>
<th>$\psi_Q(q)$</th>
<th>$Q$</th>
<th>$R$</th>
<th>$\psi_{Q,R}(q,r)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>3</td>
<td>red</td>
<td>pencil</td>
<td>2</td>
</tr>
<tr>
<td>green</td>
<td>1</td>
<td>green</td>
<td>pencil</td>
<td>1</td>
</tr>
<tr>
<td>blue</td>
<td>2</td>
<td>green</td>
<td>crayon</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>blue</td>
<td>pencil</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>blue</td>
<td>crayon</td>
<td>1</td>
</tr>
</tbody>
</table>

1. (2 points) **Short answer:** Draw a table containing all values of the function $s(q,r) = \psi_Q(q)\psi_{Q,R}(q,r)$. **You may use the integer abbreviations:** red=1, green=2, blue=3, pencil=1, crayon=2.
2. (2 points) **Numerical answer:** What is the value of the partition function $Z$ for the joint distribution $p(q, r)$?

<table>
<thead>
<tr>
<th>Q</th>
<th>$\psi_q(q)$</th>
<th>$\psi_{q,r}(q,r)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>3</td>
<td>red pencil 2</td>
</tr>
<tr>
<td>green</td>
<td>1</td>
<td>red crayon 2</td>
</tr>
<tr>
<td>blue</td>
<td>2</td>
<td>green pencil 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>green crayon 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>blue pencil 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>blue crayon 1</td>
</tr>
</tbody>
</table>
Sample Questions

7 Inference in Graphical Models

Consider yet another factor graph consisting of two random variables $Q \in \{\text{red, green, blue}\}$, $R \in \{\text{pencil, crayon}\}$. Suppose we have the following factors:

<table>
<thead>
<tr>
<th>Q</th>
<th>$\psi_Q(q)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>3</td>
</tr>
<tr>
<td>green</td>
<td>1</td>
</tr>
<tr>
<td>blue</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Q</th>
<th>R</th>
<th>$\psi_{Q,R}(q,r)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>pencil</td>
<td>2</td>
</tr>
<tr>
<td>red</td>
<td>crayon</td>
<td>2</td>
</tr>
<tr>
<td>green</td>
<td>pencil</td>
<td>1</td>
</tr>
<tr>
<td>green</td>
<td>crayon</td>
<td>3</td>
</tr>
<tr>
<td>blue</td>
<td>pencil</td>
<td>4</td>
</tr>
<tr>
<td>blue</td>
<td>crayon</td>
<td>1</td>
</tr>
</tbody>
</table>

3. (2 points) **Numerical answer:** What is the value of the joint probability $P(Q = \text{green}, R = \text{crayon})$? You may leave your answer in the form of an unsimplified fraction—no calculator necessary.
7 Inference in Graphical Models

Consider yet another factor graph consisting of two random variables $Q \in \{\text{red, green, blue}\}$, $R \in \{\text{pencil, crayon}\}$. Suppose we have the following factors:

<table>
<thead>
<tr>
<th>Q</th>
<th>$\psi_Q(q)$</th>
<th>$\psi_{Q,R}(q,r)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>3</td>
<td>red pencil 2</td>
</tr>
<tr>
<td>green</td>
<td>1</td>
<td>green pencil 1</td>
</tr>
<tr>
<td>blue</td>
<td>2</td>
<td>green crayon 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>blue pencil 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>blue crayon 1</td>
</tr>
</tbody>
</table>

4. (2 points) **Numerical answer:** What is the value of the marginal probability $P(Q = \text{green})$? You may leave your answer in the form of an unsimplified fraction—no calculator necessary.
Sample Questions

7 Inference in Graphical Models

Consider yet another factor graph consisting of two random variables $Q \in \{\text{red}, \text{green}, \text{blue}\}$, $R \in \{\text{pencil}, \text{crayon}\}$. Suppose we have the following factors:

<table>
<thead>
<tr>
<th>$Q$</th>
<th>$\psi_Q(q)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>3</td>
</tr>
<tr>
<td>green</td>
<td>1</td>
</tr>
<tr>
<td>blue</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$Q$</th>
<th>$R$</th>
<th>$\psi_{Q,R}(q,r)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>pencil</td>
<td>2</td>
</tr>
<tr>
<td>red</td>
<td>crayon</td>
<td>2</td>
</tr>
<tr>
<td>green</td>
<td>pencil</td>
<td>1</td>
</tr>
<tr>
<td>green</td>
<td>crayon</td>
<td>3</td>
</tr>
<tr>
<td>blue</td>
<td>pencil</td>
<td>4</td>
</tr>
<tr>
<td>blue</td>
<td>crayon</td>
<td>1</td>
</tr>
</tbody>
</table>

5. (2 points) **Short answer**: Suppose you run the Variable Elimination algorithm to eliminate the variable $Q$, resulting in a new factor graph with just one factor $m(r)$. Draw a table containing the values of this new factor.
7. Inference in Graphical Models

Consider yet another factor graph consisting of two random variables $Q \in \{\text{red, green, blue}\}$, $R \in \{\text{pencil, crayon}\}$. Suppose we have the following factors:

<table>
<thead>
<tr>
<th>$Q$</th>
<th>$\psi_Q(q)$</th>
<th>$\psi_{Q,R}(q,r)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>3</td>
<td>red pencil 2</td>
</tr>
<tr>
<td>green</td>
<td>1</td>
<td>red crayon 2</td>
</tr>
<tr>
<td>blue</td>
<td>2</td>
<td>green pencil 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>green crayon 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>blue pencil 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>blue crayon 1</td>
</tr>
</tbody>
</table>

6. (2 points) **Numerical answer:** What is the value of the marginal probability $P(R = \text{crayon})$? You may leave your answer in the form of an unsimplified fraction—no calculator necessary.
Sample Questions

1. (1 point) **Drawing:** Suppose you are running the Variable Elimination algorithm. The first variable you eliminate is B. Draw the factor graph that results after you have eliminated variable B.

![Factor Graph](image)

*Figure 4: A factor graph over three binary random variables A, B, C, i.e. sampled values a, b, c from the random variables are in \{0, 1\}. Assume the factors are named $\psi_A(a)$, $\psi_{A,B}(a,b)$, $\psi_{A,B,C}(a,b,c)$, and $\psi_C(c)$.***
Sample Questions

2. (1 point) **Numerical Answer:** Suppose you are running the Belief Propagation algorithm? How many messages are required to send a message from $f_{ABC}$ to $C$?

Figure 4: A factor graph over three binary random variables $A$, $B$, $C$, i.e. sampled values $a$, $b$, $c$ from the random variables are in $\{0, 1\}$. Assume the factors are named $\psi_A(a)$, $\psi_{A,B}(a, b)$, $\psi_{A,B,C}(a, b, c)$, and $\psi_C(c)$. 
1. (1 point) Is there a Bayesian Network that would convert to the factor graph shown above? Is yes, draw an example of such a Bayesian Network. If not, explain why not.
Q&A
MAP INFERENCE AS MATHEMATICAL PROGRAMMING
Exact Inference

1. Data

\[ \mathcal{D} = \{ \mathbf{x}^{(n)} \}_{n=1}^{N} \]

2. Model

\[ p(\mathbf{x} | \theta) = \frac{1}{Z(\theta)} \prod_{C \in \mathcal{C}} \psi_C(\mathbf{x}_C) \]

3. Objective

\[ \ell(\theta; \mathcal{D}) = \sum_{n=1}^{N} \log p(\mathbf{x}^{(n)} | \theta) \]

4. Learning

\[ \theta^* = \arg \max_{\theta} \ell(\theta; \mathcal{D}) \]

5. Inference

1. Marginal Inference

\[ p(\mathbf{x}_C) = \sum_{\mathbf{x}' : \mathbf{x}'_C = \mathbf{x}_C} p(\mathbf{x}' | \theta) \]

2. Partition Function

\[ Z(\theta) = \sum_{\mathbf{x}} \prod_{C \in \mathcal{C}} \psi_C(\mathbf{x}_C) \]

3. MAP Inference

\[ \hat{\mathbf{x}} = \arg \max_{\mathbf{x}} p(\mathbf{x} | \theta) \]
5. Inference

Three Tasks: (All three are NP-Hard in the general case)

1. Marginal Inference
Compute marginals of variables and cliques

\[ p(x_i) = \sum_{x' : x'_i = x_i} p(x' \mid \theta) \]

\[ p(x_C) = \sum_{x' : x'_C = x_C} p(x' \mid \theta) \]

2. Partition Function
Compute the normalization constant

\[ Z(\theta) = \sum_x \prod_{C \in \mathcal{C}} \psi_C(x_C) \]

3. MAP Inference
Compute variable assignment with highest probability

\[ \hat{x} = \arg\max_x p(x \mid \theta) \]
5. Inference

Three Tasks:

1. **Marginal Inference**
   Compute marginals of variables and cliques
   
   \[
   p(x_i) = \sum_{x' : x'_i = x_i} p(x' | \theta) \quad \text{and} \quad p(x_C) = \sum_{x' : x'_C = x_C} p(x' | \theta)
   \]

2. **Partition Function**
   Compute the normalization constant
   
   \[
   Z(\theta) = \sum_x \prod_{C \in C} \psi_C(x_C)
   \]

3. **MAP Inference** *(NP-Hard in the general case)*
   Compute variable assignment with highest probability
   
   \[
   \hat{x} = \arg\max_x p(x | \theta)
   \]
Suppose we want to predict the highest likelihood structure $y$, given observations $x$ and parameters $w$.

\[
\hat{y} = \arg\max_y \log p_w(y|x) = \arg\max_y \sum_j w^T f_{\text{node}}(x_j, y_j) + \sum_{j,k} w^T f_{\text{edge}}(x_{jk}, y_j, y_k)
\]
MAP Inference

Suppose we want to predict the highest likelihood structure $y$, given observations $x$ and parameters $w$.

$$\hat{y} = \arg\max_y \log p_w(y|x)$$

$$= \arg\max_y \sum_j w^T f_{\text{node}}(x_j, y_j) + \sum_{j,k} w^T f_{\text{edge}}(x_{jk}, y_j, y_k)$$

Idea:
1. Reformulate the problem as an integer linear program (ILP) – note that this is just going to be a new way of writing down the problem: $y \rightarrow z$
2. Then remove the integer constraints (i.e. solve the linear program (LP) relaxation)

Lemma: (Wainwright et al., 2002) If there is a unique MAP assignment, the LP relaxation of the ILP above is guaranteed to have an integer solution, which is exactly the MAP solution!
Integer Linear Programming

Whiteboard

– MAP Inference for a Binary Pairwise MRF as an ILP
– Question: What if we have non-binary variables?
Image Segmentation

\[
p_\theta(y \mid x) = \frac{1}{Z(\theta, x)} \exp\left\{ \sum_c \theta_c f_c(x, y_c) \right\}
\]

- Jointly segmenting/annotating images
- Image-image matching, image-text matching

- Problem:
  - Given structure (feature), learning \( \theta \)
  - Learning sparse, interpretable, \textbf{predictive} structures/features
Dependency parsing of Sentences

Challenge:
Structured outputs, and globally constrained to be a valid tree
OCR example

Sequential structure
Linear-chain CRF for OCR

\[ P(y | x) = \frac{1}{Z(x)} \prod_i \phi(x_i, y_i) \prod_i \phi(y_i, y_{i+1}) \]

\[ \phi(x_i, y_i) = \exp\{\sum_\alpha w_\alpha f_\alpha(x_i, y_i)\} \]

\[ \phi(y_i, y_{i+1}) = \exp\{\sum_\beta w_\beta f_\beta(y_i, y_{i+1})\} \]

\[ f_\beta(y, y') = I(y='z', y='a') \]

\[ f_\alpha(x, y) = I(x_p=1, y='z') \]

* Lafferty et al. 01
$y \Rightarrow z$ map for linear chain structures

OCR example:  $y = 'ABABB'$;

$z$'s are the indicator variables for the corresponding classes (alphabet)

<table>
<thead>
<tr>
<th></th>
<th>$z_1(m)$</th>
<th>$z_2(m)$</th>
<th>$z_3(m)$</th>
<th>$z_4(m)$</th>
<th>$z_5(m)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Z</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$z_{12}(m, n)$</th>
<th>$z_{23}(m, n)$</th>
<th>$z_{34}(m, n)$</th>
<th>$z_{45}(m, n)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0 1 0</td>
<td>0 0 0</td>
<td>0 1 0</td>
<td>0 0 0</td>
</tr>
<tr>
<td>B</td>
<td>0 0 0</td>
<td>1 0 0</td>
<td>0 0 0</td>
<td>0 1 0</td>
</tr>
<tr>
<td>Z</td>
<td>0 0 0 0</td>
<td>0 0 0 0</td>
<td>0 0 0 0</td>
<td>0 0 0 0</td>
</tr>
</tbody>
</table>
Rewriting the maximization function in terms of indicator variables:

\[
\max_z \sum_{j,m} z_j(m) \left[ w^T f_{\text{node}}(x_j, m) \right] + \sum_{j,k,m,n} z_{jk}(m,n) \left[ w^T f_{\text{edge}}(x_{jk}, m, n) \right]
\]

\[z_k(n)\]
\[
\begin{array}{cccc}
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
\end{array}
\]

\[z_j(m)\]
\[
\begin{array}{cccc}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 \\
\end{array}
\]

\[z_{jk}(m,n)\]

\[z_j(m) \geq 0; \ z_{jk}(m,n) \geq 0; \]

normalization \[\sum_m z_j(m) = 1\]

agreement \[\sum_n z_{jk}(m,n) = z_j(m)\]

integer \[z_j(m) \in \mathbb{Z}, \ z_{jk}(m,n) \in \mathbb{Z}\]
Rewriting the maximization function in terms of indicator variables:

\[
\max_z \sum_{j,m} z_j(m) \left[ w^T f_{\text{node}}(x_j, m) \right] + \sum_{j,k,m,n} z_{jk}(m, n) \left[ w^T f_{\text{edge}}(x_{jk}, m, n) \right]
\]

\[
\begin{align*}
z_k(n) & \begin{array}{c} 0 \quad 1 \quad 0 \quad 0 \end{array} \\
z_j(m) & \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\
z_{jk}(m, n) & \begin{array}{cccccc} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\end{align*}
\]

\[z_j(m) \geq 0; \; z_{jk}(m, n) \geq 0;\]

- Normalization: \[\sum_m z_j(m) = 1\]
- Agreement: \[\sum_n z_{jk}(m, n) = z_j(m)\]

\[
\max_{Az=b} (F^T w) \quad \text{subject to} \quad (F^T w) \quad \text{subject to} \quad A z = b
\]
MAP Inference

Suppose we want to predict the highest likelihood structure $y$, given observations $x$ and parameters $w$.

$$
\hat{y} = \arg\max_{y} \log p_w(y|x)
= \arg\max_{y} \sum_{j} w^T f_{\text{node}}(x_j, y_j) + \sum_{j,k} w^T f_{\text{edge}}(x_{jk}, y_j, y_k)
$$

Idea:
1. Reformulate the problem as an integer linear program (ILP) – note that this is just going to be a new way of writing down the problem: $y \rightarrow z$
2. Then remove the integer constraints (i.e. solve the linear program (LP) relaxation)

Lemma: (Wainwright et al., 2002) If there is a unique MAP assignment, the LP relaxation of the ILP above is guaranteed to have an integer solution, which is exactly the MAP solution!
STRUCTURED PERCEPTRON
Structured Perceptron

Whiteboard

– Multiclass Perceptron
– Structured Perceptron
– Structured Perceptron with Averaging
– Definition: Margin for Structured Outputs
– Mistake Bound for Structured Perceptron
Structured Perceptron

Mistake Bound:

**Definition 1** Let $\overline{\text{GEN}}(x_i) = \text{GEN}(x_i) - \{y_i\}$. In other words $\overline{\text{GEN}}(x_i)$ is the set of incorrect candidates for an example $x_i$. We will say that a training sequence $(x_i, y_i)$ for $i = 1 \ldots n$ is separable with margin $\delta > 0$ if there exists some vector $U$ with $||U|| = 1$ such that

$$\forall i, \forall z \in \overline{\text{GEN}}(x_i), \ U \cdot \Phi(x_i, y_i) - U \cdot \Phi(x_i, z) \geq \delta \ (3)$$

($||U||$ is the 2-norm of $U$, i.e., $||U|| = \sqrt{\sum_s U_s^2}$.)

**Theorem 1** For any training sequence $(x_i, y_i)$ which is separable with margin $\delta$, then for the perceptron algorithm in figure 2

$$\text{Number of mistakes} \leq \frac{R^2}{\delta^2}$$

where $R$ is a constant such that $\forall i, \forall z \in \overline{\text{GEN}}(x_i), ||\Phi(x_i, y_i) - \Phi(x_i, z)|| \leq R$.

from Collins (2002)
Structured Perceptron

• Results from Collins (2002) on two sequence tagging problems
  • Metrics:
    – **F-measure**: higher is better
    – **Error**: lower is better
  • Comparison of...
    – Structured Perceptron with and without averaging
    – Maximum entropy Markov model (**MEMM**)
  • Takeaways:
    – incredibly **easy to implement**
    – typically **blazing fast**

Table from Collins (2002)
aka. Max-Margin Markov Networks (M³Ns)

STRUCTURED SVM
Structured Perceptron

**Whiteboard**

- Warmup: Binary SVM
- Warmup: Binary SVM Hinge Loss
- Structured Large Margin
- Structured Hinge Loss
- Gradient of Structured Hinge Loss
- SGD for Structured SVM
- Loss Augmented MAP Inference
Max vs “Soft-Max” Margin

- SVMs:
  \[
  \min_w k \|w\|^2 - \sum_i \left( w^T f_i(y^i) - \max_y \left( w^T f_i(y) + \ell_i(y) \right) \right)
  \]
  Hard (Penalized) Margin

- Maxent:
  \[
  \min_w k \|w\|^2 - \sum_i \left( w^T f_i(y^i) - \log \sum_y \exp \left( w^T f_i(y) \right) \right)
  \]
  Soft Margin

- Very similar! Both try to make the true score better than a function of the other scores.
  - The SVM tries to beat the augmented runner-up
  - The maxent classifier tries to beat the “soft-max”
Hinge Loss

Consider the per-instance SVM objective:

$$\min_w k \|w\|^2 - \sum_i \left( \mathbf{w}^\top f_i(y^i) - \max_y \left[ \mathbf{w}^\top f_i(y) + \ell_i(y) \right] \right)$$

This is called the “hinge loss”

- Upper bounds zero-one loss
- Unlike maxent / log loss, you stop gaining objective once the true label wins by enough
- You can start from here and derive the SVM objective

$$\mathbf{w}^\top f_i(y^i) - \max_{y \neq y^i} \mathbf{w}^\top f_i(y)$$
Max (Conditional) Likelihood

Estimation

\[
\text{maximize}_w \sum_{x \in D} \log P_w(t(x) \mid x)
\]

Classification

\[
\text{arg max}_y w^\top f(x, y)
\]

\[
\log P_w(y \mid x) = w^\top f(x, y) - \log Z_w(x)
\]

Don’t need to learn entire distribution!
Results: Handwriting Recognition

Length: ~8 chars
Letter: 16x8 pixels
10-fold Train/Test
5000/50000 letters
600/6000 words

Models:
- Multiclass-SVMs*
- CRFs
- $M^3$ nets

* Crammer & Singer 01

![Graph showing error reduction over linear CRFs and multiclass SVMs]

45% error reduction over linear CRFs
33% error reduction over multiclass SVMs
Results: Hypertext Classification

- WebKB dataset
  - Four CS department websites: 1300 pages/3500 links
  - Classify each page: faculty, course, student, project, other
  - Train on three universities/test on fourth
- Inference: loopy belief propagation
- Learning: relaxed dual

![Graphical representation of the WebKB dataset]

<table>
<thead>
<tr>
<th>Method</th>
<th>Test Error</th>
<th>Error Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVMs</td>
<td>53%</td>
<td>38%</td>
</tr>
<tr>
<td>RMNs</td>
<td>38%</td>
<td></td>
</tr>
<tr>
<td>M^3Ns</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Taskar et al 02*
Named Entity Recognition

- Locate and classify named entities in sentences:
  - 4 categories: organization, person, location, misc.
  - e.g. "U.N. official Richard Butler heads for Baghdad".
- CoNLL 03 data set (200K words train, 50K words test)

$$y_i = \text{org/per/loc/misc/none}$$

$$f(y_i, x) = [...,$$

$$I(y_i=\text{org}, x_i=\text{"U.N."}),$$
$$I(y_i=\text{per}, x_i=\text{capitalized}),$$
$$I(y_i=\text{loc}, x_i=\text{known city}),$$
$$..., ]$$

32% error reduction over CRFs
Associative Markov networks

\[ P(y \mid x) \propto \prod_i \phi_i(y_i, x_i) \prod_{ij} \phi_{ij}(y_i, y_j, x_{ij}) = \exp \{ w^\top f(x, y) \} \]

**Point features**
- spin-images, point height

**Edge features**
- length of edge, edge orientation

“associative“ restriction

\[ \phi_{ij}(y_i, y_j) = \begin{cases} 1 & \text{if } k = 1 \\ \phi_{ij}(K, K) & \text{otherwise} \end{cases} \]

\[ \phi_{ij}(1, 1) = 1 \]

Bonus
\[ \phi_{ij}(k, k) \geq 1 \]
Max-margin AMNs results

Label: ground, building, tree, shrub
Training: 30 thousand points  Testing: 3 million points
Segmentation results

Hand labeled 180K test points

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>68%</td>
</tr>
<tr>
<td>V-SVM</td>
<td>73%</td>
</tr>
<tr>
<td>M³N</td>
<td>93%</td>
</tr>
</tbody>
</table>