Advanced Machine Learning & Perception

Instructor: Tony Jebara
Topic 8

• Beyond binary output...
• Based on T. Joachims’ slides
• Multi-Class SVM
• Structured Prediction
• Cutting Plane Algorithms
SVM Extensions

Classification

Feature/Kernel Selection

Transduction

Regression

Meta/Multi-Task Learning

Multi-Class / Structured
Multi-Class & Structured Output

- Support vector machines predict only a binary output
- Can SVMs handle multi-class labels??

$\{\text{male, female, child}\}$
Multi-Class & Structured Output

• Or, (almost any) structured output?
• For example: Natural Language Parsing
  Given a sequence of words $x$, predict the parse tree $y$. Dependencies from structural constraints, since $y$ has to be a tree.

The dog chased the cat

$x$

\[ \begin{array}{c}
  x \\
  \text{The dog chased the cat} \\
  \end{array} \quad \Rightarrow \quad \begin{array}{c}
  y \\
  \text{S} \\
  \text{NP} \\
  \text{VP} \\
  \text{Det} \quad \text{N} \\
  \text{V} \\
  \text{Det} \quad \text{N} \\
  \end{array} \]
Multi-Class & Structured Output

• Or, (almost any) structured output?
  For example: Protein Sequence Alignment
  Given two sequences \( x=(s,t) \), predict an alignment \( y \).
  Structural dependencies, since prediction has to be a valid global/local alignment.

\[
\begin{align*}
  s &= (\text{ABJLHBNJYAUGAI}) \\
  t &= (\text{BHJKBNYGU}) \\
  \text{AB-JLHBNJYAUGAI} \\
  \text{BHJK-BN-YGU}
\end{align*}
\]
Multi-Class & Structured Output

• Or, (almost any) structured output?
  For example: Information Retrieval
    Given a query $x$, predict a ranking $y$.
    Dependencies between results (e.g. avoid redundant hits)
    Loss function over rankings (e.g. AvgPrec)

Boosting
1. AdaBoost
2. Freund
3. Schapire
4. Kernel-Machines
5. Support Vector Machines
6. MadaBoost
7. ...

$X$ Boosting $\rightarrow$ $Y$
Multi-Class & Structured Output

- Or, (almost any) structured output?
- For Example, Noun-Phrase Co-reference
  Given a set of noun phrases $x$, predict a clustering $y$. Structural dependencies, since prediction has to be an equivalence relation.
  Correlation dependencies from interactions.

$x$

- The policeman fed the cat. He did not know that he was late.
- The cat is called Peter.

$y$

- The policeman fed the cat. He did not know that he was late.
- The cat is called Peter.
Multi-Class & Structured Output

- These problems are usually solved via maximum likelihood
- Or via Bayesian Networks and Graphical Models
- Problem: these methods are not discriminative!
- They learn $p(x,y)$ instead of $x \rightarrow y$ like an SVM...
- We will adapt the SVM approach to these domains...

The policeman fed the cat. He did not know that he was late.
The cat is called Peter.
Support Vector Machine

• Binary classification: \[ \left\{ (x_1, y_1), \ldots, (x_n, y_n) \right\} \rightarrow f(x) = w^T x + b \]

• Primal (P) and dual (D) give same solution

\[ \text{P: } \min_{w, b, \xi \geq 0} \frac{1}{2} \|w\|^2 + \frac{C}{n} \sum_{i=1}^{n} \xi_i \quad \text{s.t. } y_i \left( w^T x_i + b \right) \geq 1 - \xi_i \]

\[ \text{D: } \max_{\lambda} \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i, j=1}^{n} \lambda_i \lambda_j y_i y_j x_i^T x_j \quad \text{s.t. } 0 \leq \lambda_i \leq \frac{C}{n}, \sum_{i=1}^{n} \lambda_i y_i = 0 \]

• Primal (P) and dual (D) give same solution \[ w^* = \sum_{i=1}^{n} \lambda_i^* y_i x_i \]
Support Vector Machine & b=0

• Binary classification: \[ \{(x_1, y_1), \ldots, (x_n, y_n)\} \rightarrow f(x) = w^T x \]

- Hard Margin (separable)
  \[ \delta \]
- Soft Margin (training error)
  \[ \delta \]

\[ P: \quad \min_{w,b,\xi \geq 0} \frac{1}{2} \|w\|^2 + \frac{C}{n} \sum_{i=1}^{n} \xi_i \quad s.t. \quad y_i \left( w^T x_i \right) \geq 1 - \xi_i \]

\[ D: \quad \max_{\lambda} \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{n} \lambda_i \lambda_j y_i y_j x_i^T x_j \quad s.t. 0 \leq \lambda_i \leq \frac{C}{n} \]

• Solution through origin \[ w^* = \sum_{i=1}^{n} \lambda_i^* y_i x_i \] (or just pad x with 1)
Multi-Class & Structured Output

- View the problem as a list of all possible answers
- Approach: view as multi-class classification task
- Every complex output $y_i \in Y$ is one class
- Problems: Exponentially many classes!
  
  How to predict efficiently? How to learn efficiently?
  
  Potentially huge model! Manageable number of features?

The dog chased the cat

$y^1$

$y^2$

$y^k$
Multi-Class Output

• View the problem as a list of all possible answers
• Approach: view as multi-class classification task
• Every complex output $y_i \in \{1, \ldots, k\}$ is one of $K$ classes
• Enumerate many constraints (slow)...

\[
\left\{ (x_1, y_1), \ldots, (x_n, y_n) \right\} \rightarrow f(x) = \arg \max_{i \in \{1, \ldots, k\}} w_i^T x
\]

\[
\min_{w_1, \ldots, w_k, \xi \geq 0} \sum_{i=1}^{k} \|w_i\|^2 + \frac{C}{n} \sum_{i=1}^{n} \xi_i \\
\text{s.t. } \forall j \neq y_1 : \left( w_{y_1}^T x_1 \right) \geq \left( w_j^T x_1 \right) + 1 - \xi_1 \\
\text{s.t. } \ldots \\
\text{s.t. } \forall j \neq y_n : \left( w_{y_n}^T x_n \right) \geq \left( w_j^T x_n \right) + 1 - \xi_n
\]
Joint Feature Map

• Instead of solving for K different w’s, make 1 long w
• Replace each x with $\phi(x, y = i) = \left[0^T \ 0^T \ ... \ 0^T \ x^T \ 0^T \ ... \ 0^T\right]^T$
• Put the x vector in the i’th position
• The feature vectors is DK dimensional

$y_i \in \{1, \ldots, k\}$

$\left\{(x_1, y_1), \ldots, (x_n, y_n)\right\} \rightarrow f(x) = \text{arg max}_{y \in Y} \ w^T \phi(x, y)$

$$\min_{w, \xi \geq 0} \left\| w \right\|^2$$

$$\text{s.t.} \ \forall y \cup Y \setminus y_1 : w^T \phi(x_1, y_1) \geq w^T \phi(x_1, y) + 1$$

$$\text{s.t.} \ \ldots$$

$$\text{s.t.} \ \forall y \cup Y \setminus y_n : w^T \phi(x_n, y_n) \geq w^T \phi(x_n, y) + 1$$
Joint Feature Map

• Learn weight vector so that $w^T \phi(x_i, y)$ is max for correct $y$

$$\min_{w, \xi \geq 0} \|w\|^2$$

$$\text{s.t. } \forall y \cup Y \setminus y_1 : w^T \phi(x_1, y_1) \geq w^T \phi(x_1, y) + 1$$

$$\text{s.t. } \ldots$$

$$\text{s.t. } \forall y \cup Y \setminus y_n : w^T \phi(x_n, y_n) \geq w^T \phi(x_n, y) + 1$$

$$\bar{w}^T \Phi(x_1, y_1) \quad \bar{w}^T \Phi(x_2, y_2) \quad \bar{w}^T \Phi(x_3, y_3) \quad \bar{w}^T \Phi(x_n, y_n)$$
Joint Feature Map with Slack

\[
\begin{align*}
\min_{w, \xi \geq 0} & \quad \frac{1}{2} \|w\|^2 + \frac{C}{n} \sum_{i=1}^{n} \xi_i \\
\text{s.t.} & \quad \forall y \cup Y \setminus y_1 : w^T \phi(x_1, y_1) \geq w^T \phi(x_1, y) + 1 - \xi_1 \\
\text{s.t.} & \quad \ldots \\
\text{s.t.} & \quad \forall y \cup Y \setminus y_n : w^T \phi(x_n, y_n) \geq w^T \phi(x_n, y) + 1 - \xi_n
\end{align*}
\]

\[
\begin{align*}
\tilde{w}^T \phi(x_1, y_1) & \quad \tilde{w}^T \phi(x_2, y_2) & \quad \tilde{w}^T \phi(x_3, y_3) & \quad \ldots & \quad \tilde{w}^T \phi(x_n, y_n)
\end{align*}
\]

\[
\begin{align*}
(x_1, y_1) & \quad (x_2, y_2) & \quad (x_3, y_3) & \quad \ldots & \quad (x_n, y_n)
\end{align*}
\]
The label loss function

• Not all classes are created equal, why clear each by 1?

\[
\min_{w, \xi \geq 0} \frac{1}{2} \|w\|^2 + \frac{c}{n} \sum_{i=1}^{n} \xi_i \\
\text{s.t. } \forall y \cup Y \setminus y_1 : w^T \phi(x_1, y_1) \geq w^T \phi(x_1, y) + 1 - \xi_1 \\
\text{s.t. } ... \\
\text{s.t. } \forall y \cup Y \setminus y_n : w^T \phi(x_n, y_n) \geq w^T \phi(x_n, y) + 1 - \xi_n
\]

• Instead of a constant 1 value, clear some classes more

\[\Delta(y, y_1) = \text{Loss for predicting } y \text{ instead of } y_1\]

• For example, if \(y\) can be \{lion, tiger, cat\}

\[\Delta(\text{tiger}, \text{lion}) = \Delta(\text{lion}, \text{tiger}) = 1\]
\[\Delta(\text{cat}, \text{lion}) = \Delta(\text{lion}, \text{cat}) = 999\]
\[\Delta(\text{tiger}, \text{tiger}) = \Delta(\text{cat}, \text{cat}) = \Delta(\text{lion}, \text{lion}) = 0\]
Joint Feature Map with Any Loss

$$\min_{w, \xi \geq 0} \frac{1}{2} \|w\|^2 + \frac{c}{n} \sum_{i=1}^{n} \xi_i$$

s.t. \( \forall y \cup Y \setminus y_1 : w^T \Phi(x_1, y_1) \geq w^T \Phi(x_1, y) + \Delta(y, y_1) - \xi_1 \)

s.t. ...

s.t. \( \forall y \cup Y \setminus y_n : w^T \Phi(x_n, y_n) \geq w^T \Phi(x_n, y) + \Delta(y, y_n) - \xi_n \)
Joint Feature Map with Slack

- Loss function $\Delta$ measures match between target & prediction

$$
\begin{align*}
\min_{w, \xi \geq 0} & \frac{1}{2} \| w \|^2 + \frac{C}{n} \sum_{i=1}^{n} \xi_i \\
\text{s.t.} & \forall y \cup Y \setminus y_1 : w^T \phi(x_1, y_1) \geq w^T \phi(x_1, y) + \Delta(y, y_1) - \xi_1 \\
\text{s.t.} & \ldots \\
\text{s.t.} & \forall y \cup Y \setminus y_n : w^T \phi(x_n, y_n) \geq w^T \phi(x_n, y) + \Delta(y, y_n) - \xi_n
\end{align*}
$$

Lemma: The training loss is upper bounded by

$$
Err_S(h) = \frac{1}{n} \sum_{i=1}^{n} \Delta(y_i, h(x_i)) \leq \frac{1}{n} \sum_{i=1}^{n} \xi_i
$$
Generic Structural SVM (slow!)

- **Application Specific Design of Model**
  - **Loss function** $\Delta(y_i, y)$
  - **Representation** $\Phi(x, y)$
    
    ➔ Markov Random Fields [Lafferty et al. 01, Taskar et al. 04]

- **Prediction:**

  $\hat{y} = \arg\max_{y \in Y} \{ \bar{w}^T \Phi(x, y) \}$

- **Training:**

  $$\min_{\bar{w}, \xi \geq 0} \frac{1}{2} \bar{w}^T \bar{w} + \frac{C}{n} \sum_{i=1}^{n} \xi_i$$

  s.t. $\forall y \in Y \setminus y_1 : \bar{w}^T \Phi(x_1, y_1) \geq \bar{w}^T \Phi(x_1, y) + \Delta(y_1, y) - \xi_1$

  ...

  $\forall y \in Y \setminus y_n : \bar{w}^T \Phi(x_n, y_n) \geq \bar{w}^T \Phi(x_n, y) + \Delta(y_n, y) - \xi_n$

- **Applications:** Parsing, Sequence Alignment, Clustering, etc.
Reformulating the QP

**n-Slack Formulation:**

\[
\begin{align*}
\min_{\vec{w}, \xi} \quad & \frac{1}{2} \vec{w}^T \vec{w} + \frac{C}{n} \sum_{i=1}^{n} \xi_i \\
\text{s.t.} \quad & \forall y' \in Y : \vec{w}^T \Phi(x_1, y_1) - \vec{w}^T \Phi(x_1, y') \geq \Delta(y_1, y) - \xi_1 \\
& \ldots \\
& \forall y' \in Y : \vec{w}^T \Phi(x_n, y_n) - \vec{w}^T \Phi(x_n, y') \geq \Delta(y_n, y) - \xi_n
\end{align*}
\]
Reformulating the QP

n-Slack Formulation: \[ \begin{align*} \min_{\vec{w}, \xi} \quad & \frac{1}{2} \vec{w}^T \vec{w} + C \sum_{i=1}^{n} \xi_i \\ \text{s.t.} \quad & \forall y' \in Y : \vec{w}^T \Phi(x_1, y_1) - \vec{w}^T \Phi(x_1, y') \geq \Delta(y_1, y) - \xi_1 \\ & \vdots \\ & \forall y' \in Y : \vec{w}^T \Phi(x_n, y_n) - \vec{w}^T \Phi(x_n, y') \geq \Delta(y_n, y) - \xi_n \end{align*} \]

1-Slack Formulation: \[ \begin{align*} \min_{\vec{w}, \xi} \quad & \frac{1}{2} \vec{w}^T \vec{w} + C \xi \\ \text{s.t.} \quad & \forall y_1 \ldots y_n \in Y : \frac{1}{n} \sum_{i=1}^{n} \left[ \vec{w}^T \Phi(x_i, y_i) - \vec{w}^T \Phi(x_i, y'_i) \right] \geq \frac{1}{n} \sum_{i=1}^{n} [\Delta(y_i, y'_i)] - \xi \end{align*} \]
Comparing \( n \text{-Slack} \) & \( 1 \text{-Slack} \)

• Example: \( Y = \{A, B, C\} \) and \( y_1 = A, y_2 = A, y_3 = B, y_4 = C \)

\( n \text{-Slack} \rightarrow n(k-1) \) constraints

\[
\begin{align*}
y_1 &\geq B, y_1 \geq C \\
y_2 &\geq B, y_2 \geq C \\
y_3 &\geq A, y_3 \geq C \\
y_4 &\geq A, y_4 \geq B
\end{align*}
\]

\( 1 \text{-Slack} \rightarrow k^n \) constraints

\[
\begin{align*}
y_1 y_2 y_3 y_4 &\geq AAAA, AAAB, AAAC, AABA, \\
&\quad AABB, AACA, AACB, AACC, \\
&\quad ABAA, ABAB, ABAC, ABBA, \\
&\quad ABBB, ABBC, ABCA, ABCB, \\
&\quad ABCC, ACAA, ACAB, ACAC, \ldots
\end{align*}
\]

• Idea: we expect only a few constraints to be active
• Cutting-Plane: a greedy approach to QP
• Solve with only a few constraints at a time
• If solution violates come constraints, add them back in
• If we are smart about which ones to add, may not need \( k^n \)
1-Slack Cutting-Plane Algorithm

- **Input:** \((x_1, y_1), \ldots, (x_n, y_n), C, \epsilon\)
- \(S \leftarrow \emptyset, \bar{w} \leftarrow 0, \xi \leftarrow 0\)
- **REPEAT**
  - **FOR** \(i = 1, \ldots, n\)
    - Compute \(y_i' = \arg\max_{y \in Y} \{\Delta(y_i, y) + \bar{w}^T \Phi(x_i, y)\}\)
  - **ENDFOR**
  - **IF** \(\sum_{i=1}^{n} \left[\Delta(y_i, y_i') - \bar{w}^T[\Phi(x_i, y_i') - \Phi(x_i, y_i)]\right] > \xi + \epsilon\)
  - \(S \leftarrow S \cup \{\bar{w}^T \frac{1}{n} \sum_{i=1}^{n} [\Phi(x_i, y_i') - \Phi(x_i, y_i)] \geq \frac{1}{n} \sum_{i=1}^{n} \Delta(y_i, y_i') - \xi\}\)
  - **optimise** StructSVM over \(S\) to get \(w\) and \(\xi\)
  - **ENDIF**
- **UNTIL** solution has not changed during iteration \([\text{Jo06]}\) \([\text{JoFinYu08]}\)
Polynomial Sparsity Bound

Theorem: The cutting-plane algorithm finds a solution to the Structural SVM soft-margin optimization problem in the 1-slack formulation after adding at most

\[ \left\lfloor \log_2 \left( \frac{\Delta}{4R^2C} \right) \right\rfloor + \left\lfloor \frac{16R^2C}{\varepsilon} \right\rfloor \]

constraints to the working set \( S \), so that the primal constraints are feasible up to a precision \( \varepsilon \) and the objective on \( S \) is optimal. The loss has to be bounded \( 0 \leq \Delta(y_i,y) \leq \Delta \) and \( 2\|\Phi(x,y)\| \leq R \).

[Jo03] [Jo06] [TeoLeSmVi07] [JoFinYu08]
Joint Feature Map for Trees

- Weighted Context Free Grammar
  - Each rule (e.g. $S \rightarrow NP \ VP$) has a weight
  - Score of a tree is the sum of its weights
  - Find highest scoring tree $h(\vec{x}) = \arg\max_{y \in Y} \left[ \vec{w}^T \Phi(x, y) \right]$

\[
f : X \rightarrow Y \downarrow
\]

\[
\begin{align*}
\Phi(x, y) &= \begin{pmatrix}
1 & S \rightarrow NP \ VP \\
0 & S \rightarrow NP \\
2 & NP \rightarrow Det \ N \\
1 & VP \rightarrow V \ NP \\
\vdots & \\
0 & Det \rightarrow dog \\
2 & Det \rightarrow the \\
1 & N \rightarrow dog \\
1 & V \rightarrow chased \\
1 & N \rightarrow cat
\end{pmatrix}
\end{align*}
\]
Experiments: NLP

Implementation
- Incorporated modified version of Mark Johnson’s CKY parser
- Learned weighted CFG with $\epsilon = 0.01, C = 1$

Data
- Penn Treebank sentences of length at most 10 (start with POS)
- Train on Sections 2-22: 4098 sentences
- Test on Section 23: 163 sentences

<table>
<thead>
<tr>
<th>Method</th>
<th>Test Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc</td>
</tr>
<tr>
<td>PCFG with MLE</td>
<td>55.2</td>
</tr>
<tr>
<td>SVM with (1-$F_1$)-Loss</td>
<td>58.9</td>
</tr>
</tbody>
</table>

- more complex features [TaKlCoKoMa04]
Experiments of 1-Slack versus n-Slack Part-of-speech tagging on Penn Treebank ~36,000 examples, ~250,000 features in linear HMM model

[JoFinYu08]
StructSVM for Any Problem

General
- SVM-struct algorithm and implementation
  [http://svmlight.joachims.org](http://svmlight.joachims.org)
- Theory (e.g. training-time linear in n)

Application specific
- Loss function $\Delta(y_i, y)$
- Representation $\Phi(x, y)$
- Algorithms to compute
  \[
  \hat{y} = \arg\max_{y \in Y} \{ w^T \Phi(x_i, y) \}
  \]
  \[
  \hat{y} = \arg\max_{y \in Y} \{ \Delta(y_i, y) + w^T \Phi(x_i, y) \}
  \]

Properties
- General framework for discriminative learning
- Direct modeling, not reduction to classification/regression
- “Plug-and-play”
Struct SVM with Relative Margin

• Add relative margin constraints to struct SVM (ShiJeb09)
• Correct beats wrong labels but not by too much (relatively)

\[
\begin{align*}
\min_{w, \xi \geq 0} & \quad \frac{1}{2} \|w\|^2 + \frac{C}{n} \sum_{i=1}^{n} \xi_i \\
\text{s.t.} & \quad \forall y \cup Y \setminus y_1 : B \geq w^T \phi(x_1, y) - w^T \phi(x_1, y_1) \geq \Delta(y, y_1) - \xi_1 \\
\text{s.t.} & \quad \ldots \\
\text{s.t.} & \quad \forall y \cup Y \setminus y_n : B \geq w^T \phi(x_n, y) - w^T \phi(x_n, y_n) \geq \Delta(y, y_n) - \xi_n \\
\text{• Needs both} & \quad \arg \max_{y \in Y} w^T \phi(x, y) \text{ and } \arg \min_{y \in Y} w^T \phi(x, y)
\end{align*}
\]
Struct SVM with Relative Margin

• Similar bound holds for relative margin
• Maximum # of cuts is
  \[
  \max \left\{ \frac{2CR^2}{\varepsilon^2_B}, \frac{2n}{\varepsilon}, \frac{8CR^2}{\varepsilon^2} \right\}
  \]

• Try sequence learning problems for Hidden Markov Modeling
• Consider named entity recognition (NER) task
• Consider part-of-speech (POS) task

<table>
<thead>
<tr>
<th></th>
<th>NER</th>
<th>POS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRF</td>
<td>5.13 ± 0.28</td>
<td>11.34 ± 0.64</td>
</tr>
<tr>
<td>StructSVM</td>
<td>5.09 ± 0.32</td>
<td>11.14 ± 0.60</td>
</tr>
<tr>
<td>StructRMM</td>
<td><strong>5.05 ± 0.28</strong></td>
<td><strong>10.42 ± 0.47</strong></td>
</tr>
<tr>
<td>p-value</td>
<td>0.07</td>
<td>0.00</td>
</tr>
</tbody>
</table>