Beyond Binary Classification

Hang Li
Microsoft Research Asia
hangli@microsoft.com
Introduction
# Learning Methods for Classification

<table>
<thead>
<tr>
<th>Method</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>K Near Neighbor</td>
<td>Multi-Class</td>
</tr>
<tr>
<td>Naïve Bayes</td>
<td>Multi-Class</td>
</tr>
<tr>
<td>Decision Tree / Decision List</td>
<td>Multi-Class</td>
</tr>
<tr>
<td>Maximum Entropy / Logistic Regression</td>
<td>Multi-Class</td>
</tr>
<tr>
<td>Support Vector Machines</td>
<td>Binary Class</td>
</tr>
<tr>
<td>Ada Boost</td>
<td>Binary Class</td>
</tr>
</tbody>
</table>
From Binary Classification to More Complicated Predictions
This Talk: Survey on Learning Methods for Multi-Class Classification, Structure Prediction, and Ranking, Using SVM Approach

From Studies with Yinhua Hu, Yunbo Cao, Dijun Luo, Jun Xu, Tie-Yan Liu, and others
Talk Outline

• Introduction
• Multi-Class Classification
• Learning for structure Prediction
• Learning to Rank
• Summary
Multi-Class Classification
Linear SVM

\[
\min_{w, b} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{N} \xi_i \\
y_i (w \cdot x_i + b) \geq 1 - \xi_i \quad i = 1, \ldots, N \\
\xi_i \geq 0
\]

\[
f(x) = \text{sign} \left( \sum_{i=1}^{N} \alpha_i^* y_i (x_i \cdot x) + b^* \right)
\]
Multi Class Classification Problem

- Input space: $X \subseteq R^d$
- Output space: $Y = \{1, 2, \ldots, K\}$
- Prediction function: $f : X \rightarrow Y$
- Learning
  - Input: $S = \{(x_i, y_i), x_i \in X, y_i \in Y, i = 1, \ldots N\}$
  - Output: $f(x; \hat{w})$
Methods for Multi Class Classification

• Multi-Class
  

• Error Correcting Output Code (ECOC)
  

• Hierarchy
Multi-Class SVM
Cramer & Singer (2001)

• Model:
\[ \forall k, \langle w_k, x \rangle \]

• Prediction:
\[ \arg \max_k \langle w_k, x \rangle \]
Multi-Class SVM

\[
\min_{w, \xi} \frac{1}{2} \sum_{k=1}^{K} \| w_k \|^2 + C \sum_{i=1}^{N} \xi_i
\]

\[\forall i, \forall y \in Y \setminus y_i, \; \langle w_{y_i}, x_i \rangle - \langle w_y, x_i \rangle \geq 1 - \xi_i\]

\[\forall i, \xi_i \geq 0\]
Error Correcting Output Code
Dietterich & Bakiri (1995)

• Encoding
  \( \mathbf{M} \) is matrix of size \( K \times L \) over \( \{-1, 0, +1\} \)

• Base classifier construction

  \( L \) binary classifiers \( h_1(x), \cdots, h_L(x) \)

  \( \mathbf{M}_{k,l} = +1 \)

  \( \mathbf{M}_{k,l} = -1 \)

• Decoding

  \( \mathbf{h}(x) = [h_1(x), \cdots, h_L(x),] \)

  \( \arg \min_k D(\mathbf{M}_k, \mathbf{h}(x)) \)
Encoding

- Coding Matrix
  - One vs rest
  - One vs one
  - Random code

\[
\begin{pmatrix}
+1 & -1 & -1 \\
-1 & +1 & -1 \\
-1 & -1 & +1
\end{pmatrix}
\]

\[
\begin{pmatrix}
+1 & +1 & 0 \\
-1 & 0 & +1 \\
0 & -1 & -1
\end{pmatrix}
\]
Combining ECOC and Multi Class SVM in Single Framework

- Coding matrix (one vs rest) and decoding metric (dot product) are given
- Binary classifier: SVM
- Simultaneously training binary SVM classifiers is equivalent to Multi Class SVM
Using Framework of ECOC

- **Classification**
  
  \[ f(x) = \arg \max_y F(x, y) \]
  
  \[ F(x, y) = \langle M_y, \cdot(h(x) \rangle \]
  
  \[ h(x) = [h_1(x), \ldots, h_L(x)] \]

- **Loss Function**

  \[ \left( 1 - F(x, y'; w) - \max_{y \neq y'} F(y, y'; w) \right)_+ \]
Learning

• Training data

\[ S = \{(x_1, y_1), \ldots, (x_N, y_N)\} \]

• Regularized total loss

\[
L(S; w) = \sum_{i=1}^{N} \left[ 1 - \left( \sum_{l=1}^{L} M_{y_i, l} \langle w_l, x_i \rangle - \max_{y \neq y_i} \sum_{l=1}^{L} M_{y, l} \langle w_l, x_i \rangle \right) \right]_+ + \lambda \sum_{l=1}^{L} \| w_l \|^2
\]

• SVM

\[
\min_{w, \xi} \frac{1}{2} \sum_{l=1}^{L} \| w_l \|^2 + C \sum_{i=1}^{N} \xi_i
\]

\[ \forall i, \forall y \in Y \setminus y_i, \quad \sum_{l=1}^{L} M_{y_i, l} \langle w_l, x_i \rangle - \sum_{l=1}^{L} M_{y, l} \langle w_l, x_i \rangle \geq 1 - \xi_i \]

\[ \xi_i \geq 0 \]
ECOC: One vs. Rest

\[
\min_{w, \xi} \frac{1}{2} \sum_{k=1}^{K} \| w_k \|^2 + C \sum_{i=1}^{N} \xi_i
\]

\[
\forall i, \forall y \in Y \setminus y_i, \quad \langle w_{y_i} - w_y, x_i \rangle \geq \frac{1}{2} (1 - \xi_i)
\]

\[
\xi_i \geq 0
\]
Structure Prediction
Example of Structure Prediction Term Extraction

\[ x: \text{The schedule is subject to change without notice} \]

\[ y: \text{The schedule is subject to change without notice} \]

\[ y': \text{The schedule is subject to change without notice} \]
Methods for Structure Prediction

• SVM

• Hidden Markov Model
• Conditional Random Fields
Discriminative Approach to Structure Prediction

- Input space: $X$, output space: $Y$
- Prediction function: $f : X \rightarrow Y$
- Discriminate function: $F : X \times Y \rightarrow R$
- Predict: $f(x; w) = \arg \max_{y \in Y} F(x, y; w)$
- Linear function: $F(x, y; w) = \langle w, \Psi(x, y) \rangle$
  \[
  \Psi(x, y) \in \mathbb{R}^d
  \]
Discriminative Approach

• Prediction:
  – Input: \((x, Y)\)
  – Output: \(f(x; \hat{w}) = \arg \max_{y \in Y} F(x, y; \hat{w})\)

• Learning
  – Input: \(S = \{(x_i, y_i, Y_i), i = 1, \cdots, N\}\)
  – Output: \(f(x; \hat{w})\)
Notes

- Space $Y$ is large
- $F$ is function of both $x$ and $y$
- Outputs $y$’s are interdependent
- Number of outputs $y$’s is exponential
- Dynamic programming algorithm must exist for computing $F(x,y;w)$
SVM Model

\[
\min_{w,\xi} \frac{1}{2} \| w \|^2 + \frac{C}{n} \sum_{i=1}^{N} \xi_i \\
\forall i, \forall y \in Y_i \setminus y_i : \langle w, \Psi(x_i, y_i) \rangle - \langle w, \Psi(x_i, y) \rangle \geq 1 - \xi_i \\
\xi_i \geq 0
\]
Loss Function

• General loss function

\[ L \left( F(x, y; w) - \max_{y' \in Y \setminus y} F(x, y'; w) \right) \]

• Hinge loss function

\[ L \left( 1 - \left[ F(x, y; w) - \max_{y' \in Y \setminus y} F(x, y'; w) \right]_+ \right) \]
Ranking
Example of Ranking: Information Retrieval

\[ D = \{d_1, d_2, \ldots, d_n\} \]

documents (information)

query (or question)

\[ q \]

\[ f(q, d, D) \]

relevance scores for ranking

\[ d_1 \sim f(q, d_1) \]
\[ d_2 \sim f(q, d_2) \]
\[ \vdots \]
\[ d_n \sim f(q, d_n) \]
Methods for Ranking

• Point-wise Ranking Methods
  – A. Sahshu, A. Levin, Ranking with Large Margin Principle: Two Approaches, NIPS’03

• Pair-wise Ranking Methods
  – Yunbo Cao, Jun Xu, Tie-Yan Liu, Hang Li, Yalou Huang, Hsiao-Wuen Hon, Adapting Ranking SVM to Document Retrieval, Proc. of SIGIR’06.
Point-wise Ranking SVM
Shashua and Levin (2003)

- Input space: $X$, output space: $Y$ with order
- Ranking function $f : X \rightarrow Y$
- Ranking: by $f(x; w)$

$$f(x) = \arg \min_{k \in \{1, \ldots, K\}} (w \cdot x + b_k < 0)$$

$$b_K = \infty$$
Point-wise Ranking SVM

\[
\min_{w, b, \xi} \frac{1}{2} \| w \|^2 + C \sum_i \sum_j (\xi_i^j + \xi_i^{*j+1}) \\
\langle w, x_i^j \rangle - b_j \leq -1 + \xi_i^j \\
\langle w, x_i^{j+1} \rangle - b_j \geq 1 - \xi_i^{*j+1} \\
\xi_i^j \geq 0, \xi_i^{*j} \geq 0 \\
j = 1, \ldots, k - 1, \quad i = 1, \ldots, i_j
(Pair-wise) Ranking SVM

- Input space: $X$
- Ranking function $f : X \rightarrow R$
- Ranking: $x_i \succ x_j \iff f(x_i; w) > f(x_j; w)$
- Linear ranking function: $f(x; w) = \langle w, x \rangle$

$$\langle w, x^{(1)} - x^{(2)} \rangle > 0 \iff f(x^{(1)}; w) > f(x^{(2)}; w)$$

- Transforming to binary classification:

$$(\tilde{x}^{(1)} - \tilde{x}^{(2)}, z), \quad z = \begin{cases} +1 & y^{(1)} \succ y^{(2)} \\ -1 & y^{(2)} \succ y^{(1)} \end{cases}$$
Pair-wise Ranking SVM

\[
\min_{w, \xi} \frac{1}{2} \| w \|^2 + C \xi_i \\
\]

\[
z_i \langle w, x^{(1)}_i - x^{(2)}_i \rangle \geq 1 - \xi_i \\
\xi_i \geq 0
\]

\[
\min_w \sum_{i=1}^l \left[ 1 - z_i \langle w, x^{(1)}_i - x^{(2)}_i \rangle \right] + \lambda \| w \|^2
\]
Pair-wise Ranking SVM

- **Learning**
  - Input: \( S = \{(x^{(1)}_i > x^{(2)}_i)\}_{i=1}^m \)
  - Output: \( f(x; \hat{w}) \)
Direct Application of Ranking SVM to Information Retrieval

- Query document pair $\rightarrow$ feature vector
- Combining instance pairs from all queries
Applying Ranking SVM to Document Retrieval

- Cost sensitiveness: negative effects of making errors on top
  
  \[d: \text{definitely relevant}, \ p: \text{partially relevant}, \ n: \text{not relevant}\]
  
  ranking 1: \( p \ d \ p \ n \ n \ n \ n \)
  
  ranking 2: \( d \ p \ n \ p \ n \ n \ n \)
  
- Query normalization: number of instance pairs varies according to query
  
  \[q1: \ d \ p \ p \ n \ n \ n \ n \]
  
  \[q2: \ d \ d \ p \ p \ p \ n \ n \ n \ n \ n \ n \ n \]
  
  \[\text{q1 pairs: } 2*(d, p) + 4*(d, n) + 8*(p, n) = 14\]
  
  \[\text{q2 pairs: } 6*(d, p) + 10*(d, n) + 15*(p, n) = 31\]
Rank Pair Distinction
Query Normalization
Ranking SVM for IR

**Loss Function**

\[
\min_{\bar{w}} L(\bar{w}) = \sum_{i=1}^{l} \tau_{k(i)} \mu_{q(i)} \left[ 1 - z_i \langle \bar{w}, \bar{x}_i^{(1)} - \bar{x}_i^{(2)} \rangle \right] + \lambda \| \bar{w} \|^2
\]
Summary
Summary

• Multi-Class Classification
  Combining ECOC and Multi-Class SVM in Single Framework
• Structure Prediction
  Transforming into Binary Classification
• Ranking
  Transforming into Binary Classification
Thank You