Multiclass

Jordan Boyd-Graber
University of Colorado Boulder

LECTURE 13

Slides adapted from Rob Schapire and Fei Xia
Motivation

- Binary and Multi-class: problems and classifiers
- Solving Multi-class problems with binary classifiers
  - One-vs-all
  - All pairs
  - Error correcting codes
Classification Problems

- Natural binary
  - Spam classification (spam vs. ham)
  - Segmentation (same or different)
  - Coreference

- However, many are multiclass
  - Topic classification
  - Part of speech tagging
  - Scene classification
Classification Problems

• Natural binary
  ○ Spam classification (spam vs. ham)
  ○ Segmentation (same or different)
  ○ Coreference

• However, many are multiclass
  ○ Topic classification
  ○ Part of speech tagging
  ○ Scene classification
Classifiers

- Some are directly multi-class (naïve Bayes, logistic regression, KNN)
- Other classifiers are basically binary
Classifiers

- Some are directly multi-class (naïve Bayes, logistic regression, KNN)
- Other classifiers are basically binary
  - SVM
  - Perceptron
  - Boosting
## Reduction

### Multiclass Data

<table>
<thead>
<tr>
<th>name</th>
<th>age</th>
<th>sex</th>
<th>color</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cindy</td>
<td>5</td>
<td>F</td>
<td>☢️</td>
</tr>
<tr>
<td>Marcia</td>
<td>15</td>
<td>F</td>
<td>🔴</td>
</tr>
<tr>
<td>Bobby</td>
<td>6</td>
<td>M</td>
<td>💧</td>
</tr>
<tr>
<td>Jan</td>
<td>12</td>
<td>F</td>
<td>🟢</td>
</tr>
<tr>
<td>Peter</td>
<td>13</td>
<td>M</td>
<td>🟡</td>
</tr>
</tbody>
</table>
Reduction

Multiclass Data

\[
\langle \text{name}=\text{Cindy}, \text{age}=5, \text{sex}=\text{F}\rangle,
\langle \text{name}=\text{Marcia}, \text{age}=15, \text{sex}=\text{F}\rangle,
\langle \text{name}=\text{Bobby}, \text{age}=6, \text{sex}=\text{M}\rangle,
\langle \text{name}=\text{Jan}, \text{age}=12, \text{sex}=\text{F}\rangle,
\langle \text{name}=\text{Peter}, \text{age}=13, \text{sex}=\text{M}\rangle,
\]

Binary Classifier

\[
(x_1, +), (x_2, -), (x_3, +), \ldots \rightarrow A \rightarrow h
\]

\[
h(x) \in \{+, -\}
\]
Reduction

Multiclass Data

\[ \langle \text{name=\textit{Cindy}, \text{age}=5, \text{sex=F}} \rangle, \]
\[ \langle \text{name=\textit{Marcia}, \text{age}=15, \text{sex=F}} \rangle, \]
\[ \langle \text{name=\textit{Bobby}, \text{age}=6, \text{sex=M}} \rangle, \]
\[ \langle \text{name=\textit{Jan}, \text{age}=12, \text{sex=F}} \rangle, \]
\[ \langle \text{name=\textit{Peter}, \text{age}=13, \text{sex=M}} \rangle, \]

Binary Classifier

\[ (x_1, +), (x_2, -), (x_3, +), \ldots \]

\[ h(x) \in \{+, -\} \]

Goal: Multiclass Classifier
One-Against-All

- Break $k$-class problem into $k$ binary problems and solve separately
- Combine predictions: evaluate all $h$’s, hope exactly one is $+$ (otherwise, take highest confidence)
One-Against-All

- Break $k$-class problem into $k$ binary problems and solve separately
- Combine predictions: evaluate all $h$’s, hope exactly one is + (otherwise, take highest confidence)
- Incorrect prediction if only one is wrong
All-Pairs (Friedman; Hastie & Tibshirani)

<table>
<thead>
<tr>
<th></th>
<th>x₁</th>
<th>x₂</th>
<th>x₃</th>
<th>x₄</th>
<th>x₅</th>
<th>h₁</th>
<th>h₂</th>
<th>h₃</th>
<th>h₄</th>
<th>h₅</th>
<th>h₆</th>
</tr>
</thead>
<tbody>
<tr>
<td>x₁</td>
<td>-</td>
<td>x₁</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x₂</td>
<td>-</td>
<td></td>
<td>x₂</td>
<td>-</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x₃</td>
<td>-</td>
<td>x₂</td>
<td>+</td>
<td></td>
<td>x₃</td>
<td>-</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x₄</td>
<td>-</td>
<td>-</td>
<td>x₃</td>
<td>+</td>
<td>x₄</td>
<td>+</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x₅</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- One binary problem for each pair of classes
- Take class with most positives and least negatives
- Faster and more accurate than one-against-all
Time Comparison

Assume training time is $O(m^\alpha)$ and test time is $O(c_t)$

<table>
<thead>
<tr>
<th></th>
<th>Training</th>
<th>Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVA</td>
<td>$O(km^\alpha)$</td>
<td>$O(kc_t)$</td>
</tr>
<tr>
<td>All-pairs</td>
<td>$O(k^2 (\frac{m}{k})^\alpha)$</td>
<td>$O(k^2 c_t)$</td>
</tr>
</tbody>
</table>
Assume training time is $O(m^\alpha)$ and test time is $O(c_t)$

<table>
<thead>
<tr>
<th></th>
<th>Training</th>
<th>Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVA</td>
<td>$O(km^\alpha)$</td>
<td>$O(kc_t)$</td>
</tr>
<tr>
<td>All-pairs</td>
<td>$O(k^2 (\frac{m}{k})^\alpha)$</td>
<td>$O(k^2 c_t)$</td>
</tr>
</tbody>
</table>

OVA better for testing time, all-pairs better for training. (All-pairs usually better for performance.)
Error Correcting Output Codes (Dietterich & Bakiri)

- Reduce to binary using “coding” matrix

\[
\begin{array}{c|ccccc}
M & 1 & 2 & 3 & 4 & 5 \\
\hline
\text{Green} & + & - & + & - & + \\
\text{Yellow} & - & - & + & + & + \\
\text{Red} & + & + & - & - & - \\
\text{Blue} & + & + & + & + & - \\
\end{array}
\]
Error Correcting Output Codes (Dietterich & Bakiri)

- Reduce to binary using “coding” matrix
- Train classifier for each bit

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>$x_2$</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$x_3$</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>-</td>
</tr>
<tr>
<td>$x_4$</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>$x_5$</td>
<td>+</td>
<td>-</td>
<td>+</td>
<td>-</td>
<td>+</td>
</tr>
</tbody>
</table>

$h_1 \downarrow \ h_2 \downarrow \ h_3 \downarrow \ h_4 \downarrow \ h_5 \downarrow$
Error Correcting Output Codes (Dietterich & Bakiri)

- Reduce to binary using “coding” matrix
- Train classifier for each bit

Choose closest row of coding matrix to predict
ECOC

- If rows of $M$ are far apart, will be robust to error
- Much faster if $k$ is large
- Disadvantage: binary problems may be unnatural
That’s it for classification!

- You can implement multiple forms of classification
- Derive theoretical bounds for many classification tasks
- Today is bridge to the future: classification foundation of other ML tasks