Semi-supervised Learning via Regularized Boosting Working on Multiple Semi-Supervised Assumptions (TPAMI '11)

Written by: Ke Chen, Shihai Wang

Presented by: Mike Izbicki
Why not RegBoost?

- Transductive vs inductive
- Uses 3 semi-supervised assumptions
  - Smoothness
  - Low density separation
  - Manifold
- Boosting
  - Converts any weak supervised algorithm into strong semi-supervised algorithm
Transduction: learn labels only for unlabeled inputs

Induction: learn labels for any future input
SSA 1: Smoothness

• If two points are close together, their labels should be consistent
SSA 1: Smoothness

- If two points are close together, their labels should be consistent
SSA 2: Low density separation

- The decision boundary probably lies in a low density region
SSA 2: Low density separation

- The decision boundary probably lies in a low density region
SSA 3: Manifold

- High dimensional data lies on a low dimensional manifold
SSA 3: Manifold

- High dimensional data lies on a low dimensional manifold
Effects of assumptions

- **a** = data
- **b** = smooth
- **c** = smooth + manifold
- **d** = smooth + manifold + low density
“Margin Cost” Boosting Framework

- A point is classified by taking the weighted sum of many weak classifiers

- Steps:
  - Assign “pseudo-labels” to unlabeled data
  - Add new weight and classifier to minimize “cost”

\[
F_t(x) = F_{t-1}(x) + w_t \cdot f_t(x)
\]

\[
\text{minimize } \langle \nabla C(F), f \rangle
\]
The RegBoost Cost Functional

**MarginBoost**

\[
C(F) = \sum_{x_i \in L} \alpha_i C[y_i F(x_i)] + \sum_{x_i \in U} \alpha_i C[\|F(x_i)\|]
\]

**RegBoost**

\[
C(F) = \sum_{i \in S} \left\{ \frac{1}{|L|} I_{i,L} \alpha_i C[\hat{y}_i F(x_i)] + \frac{1}{|U|} I_{i,U} \beta_i |N(i)|^{-1} \sum_{j \in N(i)} \omega_{ij} C[\hat{y}_j F(x_i)] \right\}
\]

\[
\beta_i = \lambda[p(x_i)]
\]

\[
\omega_{ij} = \exp\left( -\frac{\|x_i - x_j\|^2}{2\sigma^2} \right)
\]
Skipping the proof...
## Empirical Comparison

<table>
<thead>
<tr>
<th>Data Set</th>
<th>3 Nearest Neighbors</th>
<th>Naïve Bayes</th>
<th>C4.5 Decision Tree</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ADAB</td>
<td>ASMB</td>
<td>SEMIB</td>
</tr>
<tr>
<td>AUS</td>
<td>42.7±4.0</td>
<td>39.9±4.0</td>
<td>36.8±4.9</td>
</tr>
<tr>
<td>BUPA</td>
<td>47.0±6.8</td>
<td>45.2±9.3</td>
<td>45.5±7.1</td>
</tr>
<tr>
<td>GC</td>
<td>37.4±3.7</td>
<td>37.1±2.7</td>
<td>31.0±4.9</td>
</tr>
<tr>
<td>HMS</td>
<td>39.2±12.1</td>
<td>36.6±8.0</td>
<td>31.3±6.4</td>
</tr>
<tr>
<td>HDC</td>
<td>52.5±4.2</td>
<td>46.2±11.1</td>
<td>49.2±9.3</td>
</tr>
<tr>
<td>HEP</td>
<td>37.1±10.3</td>
<td>36.5±11.2</td>
<td>33.6±3.5</td>
</tr>
<tr>
<td>HC</td>
<td>33.3±13.4</td>
<td>32.2±8.1</td>
<td>34.3±6.1</td>
</tr>
<tr>
<td>ION</td>
<td>29.6±10.1</td>
<td>27.6±7.3</td>
<td>35.0±4.1</td>
</tr>
<tr>
<td>KVK</td>
<td>32.5±3.1</td>
<td>35.8±2.7</td>
<td>35.7±2.9</td>
</tr>
<tr>
<td>MM</td>
<td>28.4±4.3</td>
<td>27.0±2.6</td>
<td>26.4±4.7</td>
</tr>
<tr>
<td>PID</td>
<td>36.1±4.2</td>
<td>35.6±6.0</td>
<td>35.4±3.6</td>
</tr>
<tr>
<td>VOTE</td>
<td>12.9±4.6</td>
<td>12.1±3.9</td>
<td>13.6±5.7</td>
</tr>
<tr>
<td>WDBC</td>
<td>12.9±7.9</td>
<td>12.3±7.0</td>
<td>19.9±11.3</td>
</tr>
</tbody>
</table>

Experimental setup not *easily* reproducible!
Does not consider run time!
## Feature Comparison

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Year</th>
<th>Inductive</th>
<th>Smoothness</th>
<th>Low density</th>
<th>Manifold</th>
<th>Boosting</th>
</tr>
</thead>
<tbody>
<tr>
<td>RegBoost</td>
<td>2011</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>SemiBoost</td>
<td>2009</td>
<td>Y</td>
<td>?</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>RegBoost</td>
<td>2008</td>
<td>Y</td>
<td>Y</td>
<td>?</td>
<td>?</td>
<td>Y</td>
</tr>
<tr>
<td>Cluster Kernels</td>
<td>2003</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>ASSEMBLE</td>
<td>2002</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>MarginBoost</td>
<td>2002</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
</tr>
<tr>
<td>Markov Random Walks</td>
<td>2001</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>TSVM</td>
<td>1999</td>
<td>Y</td>
<td>Y</td>
<td>Y</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>Graph Mincuts</td>
<td>1998</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td>N</td>
</tr>
<tr>
<td>AdaBoost</td>
<td>1995</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>Y</td>
</tr>
</tbody>
</table>
My thoughts on the future

- Accounting for natural ordering of labels
  - Also, probability distributions
- Using multiple sources of supervision
  - Semi-supervised learning vs. recommendation engines
  - Help solve the “cold start” problem
- Online algorithms
My project: beer classification

- Has many of the problems mentioned
- Labels provided by volunteers!
  - Look, smell, mouthfeel, taste, overall
  - Rank each as 1-5
- Unlabeled data
  - beeradvocate.com
  - 75,000 beers in database
  - Pro rank, ABV, style, brewer
  - etc...
Some “problems” with my dataset

- Classes have a natural ordering, since they are real valued
  - Current assumptions do not account for this
- Many inputs have no natural ordering
  - How does this affect the semi-supervised assumptions?
- Multiple sources of supervision
  - Compare to recommendation engines
  - “Cold start problem”
- Can do both active and semi-supervised learning at the same time