Boosting under high noise.
Adaboost is sensitive to label noise

- Letter / Irvine Database
- Focus on a binary problem: \{F,I,J\} vs. other letters.

<table>
<thead>
<tr>
<th>Label Noise</th>
<th>Adaboost</th>
<th>Logitboost</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>0.8% ±0.2%</td>
<td>0.8% ±0.1%</td>
</tr>
<tr>
<td>20%</td>
<td>33.3% ±0.7%</td>
<td>31.6% ±0.6%</td>
</tr>
</tbody>
</table>

- Boosting puts too much weight on outliers.
- Need to give up on outliers.
Robustboost - A new boosting algorithm

<table>
<thead>
<tr>
<th>Label Noise</th>
<th>Adaboost</th>
<th>Logitboost</th>
<th>Robustboost</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>0.8% ±0.2%</td>
<td>0.8% ±0.1%</td>
<td>2.9% ±0.2%</td>
</tr>
<tr>
<td>20%</td>
<td>33.3% ±0.7%</td>
<td>31.6% ±0.6%</td>
<td>22.2% ±0.8%</td>
</tr>
</tbody>
</table>

error with respect to original (noiseless) labels

| 20%       | 22.1% ±1.2%   | 19.4% ±1.3%   | 3.7% ±0.4%    |
Approximating mistake loss with convex functions

Adaboost = $e^{-y(w \cdot x)}$

Hinge-Loss

Logitboost (logistic regression)

Brownboost

0-1 loss

Mistakes

Margin

Correct
Label noise and convex loss functions

- Algorithms for learning a classifier based on minimizing a convex loss function: perceptron, Adaboost, Logitboost, Logistic regression, soft margins SVM.

- Work well when data is linearly separable.

- Can get into trouble when not linearly separable.

- **Problem**: Convex loss functions are a poor approximation for classification error.

- **But**: No known efficient algorithms for minimizing a non-convex loss function.
Random label noise defeats any convex loss function

[Servedio, Long 2010]
Considering one symmetric half

[Servedio, Long 2010]
Theorem: for any convex loss function there exists a linearly separable distribution such that when independent label noise is added, the linear classifier that minimizes the loss function has very high classification error.

[Servedio, Long 2010]
Boost by majority, Brownboost,

• Target error set at start.
• Defines how many boosting iterations are needed
• The loss function depends on the time-to-finish.
• Close to end - give up on examples with large negative margins.
\[ \psi_{Ada}(s) = w_{Ada}(s) = e^{-s} \]

\[ \psi_{Logit}(s) = \ln(1 + e^{-s}) \]

\[ w_{Logit}(s) = \frac{1}{1 + e^{s}} \]
Experimental Results on Long/Servedio synthetic example
Adaboost on Long/Servedio
LogitBoost on Long/Servedio
Robustboost on Long/Servedio
Experimental Results on real-world data
Robustboost - A new boosting algorithm

<table>
<thead>
<tr>
<th>Label Noise</th>
<th>Adaboost</th>
<th>Logitboost</th>
<th>Robustboost</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>0.8% ±0.2%</td>
<td>0.8% ±0.1%</td>
<td>2.9% ±0.2%</td>
</tr>
<tr>
<td>20%</td>
<td>33.3% ±0.7%</td>
<td>31.6% ±0.6%</td>
<td>22.2 ±0.8%</td>
</tr>
</tbody>
</table>

error with respect to original (noiseless) labels

| 20%         | 22.1% ±1.2% | 19.4% ±1.3%  | 3.7% ±0.4%  |
Logitboost
0% Noise
Logitboost
20% Noise
Robustboost
20% Noise
JBoost V2.0

New in Version 2.0!

The following are the new features of JBoost 2.0:

- RobustBoost support added -- a new boosting algorithm that is resistant to label noise.
- A new visualization tool -- the score visualizer
- Support for stopping and restarting the boosting process while eliminating those examples with small weight from the restarted process.
- JBoost no longer supports Multi-class problems internally, but now offers a wrapper script.

Overview

JBoost is an easy to use and modify tool for boosting classification. JBoost includes state-of-the-art algorithms and can be used by researchers to quickly implement new boosting algorithms. JBoost also includes a set of easy to use scripts so that machine learning novices can quickly learn and utilize the power of boosting.

Some of the algorithms currently implemented include AdaBoost, LogitBoost, BoosTexter and RobustBoost. These algorithms are wrapped inside of an implementation of alternating decision trees (ADTrees), which allows for easy visualization of the final classifier, even for high dimensional data. Each of the algorithms comes with a set of options that allows for customization to your dataset.

To learn more, download JBoost or read the documentation.