ICML 2009 Tutorial
Survey of Boosting
from an Optimization Perspective

Part I: Entropy Regularized LPBoost
Part II: Boosting from an Optimization Perspective

Manfred K. Warmuth - UCSC
S.V.N. Vishwanathan - Purdue & Microsoft Research

Updated: August 15, 2009
1. Introduction to Boosting

2. What is Boosting?

3. LPBoost

4. Entropy Regularized LPBoost

5. Overview of Boosting algorithms

6. Conclusion and Open Problems
Outline

1. Introduction to Boosting
2. What is Boosting?
3. LPBoost
4. Entropy Regularized LPBoost
5. Overview of Boosting algorithms
6. Conclusion and Open Problems
Setup for Boosting

[Giants of field: Schapire, Freund]

- examples: 11 apples
- -1 if artificial
  + 1 if natural
- goal:
  classification
Setup for Boosting

-1/+1 examples
weight $d_n \approx$ size
Weak hypotheses

- weak hypotheses: decision stumps on two features
- goal: find convex combination of weak hypotheses that classifies all
Boosting: 1st iteration

First hypothesis:
- error: $\frac{1}{11}$
- edge: $\frac{9}{11}$

edge = 1 − 2 error
low error = high edge
Update after 1st

Misclassified examples
- increased weights

After update
- edge of hypothesis decreased
Before 2nd iteration

Hard examples

- high weight
Boosting: 2nd hypothesis

Pick hypotheses with high edge
Update after 2nd

After update
- edges of all past hypotheses should be small
3rd hypothesis
Update after 3rd
4th hypothesis
Update after 4th
Final convex combination of all hypotheses

Decision: $\sum_{t=1}^{T} w_t h^t(x) \geq 0$?

Positive total weight - Negative total weight
Protocol of Boosting

- Maintain distribution on $N \pm 1$ labeled examples
- At iteration $t = 1, \ldots, T$:
  - Receive “weak” hypothesis $h^t$ of high edge
  - Update $d^{t-1}$ to $d^t$ more weights on “hard” examples
- Output convex combination of the weak hypotheses
  \[
  \sum_{t=1}^{T} w_t h^t(x)
  \]

Two sets of weights:
- distribution on $d$ on examples
- distribution on $w$ on hypotheses
Edge vs. margin

Edge of a hypothesis $h$ for a distribution $d$ on the examples

$$
\sum_{n=1}^{N} \frac{y_n h(x_n)}{d_n} d \in \mathcal{P}^N
$$

average goodness of hypothesis

Margin of example $n$ for current hypothesis weighting $w$

$$
\sum_{t=1}^{T} \frac{y_n h^t(x_n)}{w_t} w \in \mathcal{P}^T
$$

average goodness of example
Edge vs. margin

Edge of a hypothesis $h$ for a distribution $d$ on the examples

$$\sum_{n=1}^{N} d_n \left( \sum_{n=1}^{N} y_n h(x_n) \right) \quad d \in \mathcal{P}^N$$

average goodness of hypothesis

Margin of example $n$ for current hypothesis weighting $w$

$$\sum_{t=1}^{T} w_t \left( \sum_{t=1}^{T} y_n h^t(x_n) \right) \quad w \in \mathcal{P}^T$$

average goodness of example
Objectives

Edge
- Edges of past hypotheses should be small after update
- Minimize maximum edge of past hypotheses

Margin
- Choose convex combination of weak hypotheses that maximizes the minimum margin

Which margin?
- SVM: 2-norm (weights on examples)
- Boosting: 1-norm (weights on base hypotheses)

Connection between objectives?
Edge vs. margin

\[
\min_{d \in S^N} \max_{q=1,2,...,t-1} \sum_{n=1}^{N} y_n h^q(x_n)d_n = \max_{w \in S^{t-1}} \min_{n=1,2,...,N} \sum_{q=1}^{t-1} y_n h^q(x_n) w_q
\]

- \text{edge of hypothesis q}
- \text{margin of example n}

Linear Programming duality
Boosting as zero-sum-game

Rock, Paper, Scissors game

\[
\begin{array}{ccc}
  & R & P & S \\
R & 0 & 1 & -1 \\
P & -1 & 0 & 1 \\
S & 1 & -1 & 0 \\
\end{array}
\]

Row player minimizes
Column player maximizes

payoff \( = \ d^T U \ w \)
\( = \ \sum_{i,j} d_i u_{i,j} w_j \)

Single row is pure strategy of row player and \( d \) is mixed strategy

Single column is pure strategy of column player and \( w \) is mixed strategy
Optimum strategy

R  P  S
\[
\begin{array}{ccc}
  w_1 & w_2 & w_3 \\
  .33 & .33 & .33 \\
\end{array}
\]

\[
\begin{array}{cccc}
  d_1 & .33 & 0 & 1 & -1 \\
  d_2 & .33 & -1 & 0 & 1 \\
  d_3 & .33 & 1 & -1 & 0 \\
\end{array}
\]

- Min-max theorem:

\[
\min_d \max_w d^T U w = \min_d \max_j d^T U e_j
\]

\[
= \max_w \min_d d^T U w = \max_i \min_w e_i^T U w
\]

\[
= \text{value of the game (0 in example)}
\]

\[e_j\] is pure strategy
Connection to Boosting?

- Rows are the examples
- Columns the weak hypothesis
- $U_{i,j} = h^j(x_i)y_i$
- Row sum: margin of example
- Column sum: edge of weak hypothesis
- Value of game:
  \[
  \min \max \text{ edge} = \max \min \text{ margin}
  \]

Van Neumann’s Minimax Theorem
### Weak hypothesis = column of game matrix $U$

<table>
<thead>
<tr>
<th>examples $x_n$</th>
<th>labels $y_n$</th>
<th>1st stump $h^1(x_n)$</th>
<th>$U_{*,1} = u_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td></td>
</tr>
</tbody>
</table>
Edges/margins

\[
\begin{array}{cccc}
R & P & S \\
& w_1 & w_2 & w_3 & \text{margin} \\
& .33 & .33 & .33 \\
\end{array}
\]

\[
\begin{array}{cccccc}
R & d_1 & .33 & 0 & 1 & 1 & 0 \\
P & d_2 & .33 & -1 & 0 & 1 & 0 & \text{min} \\
S & d_3 & .33 & 1 & -1 & -1 & 0 \\
\end{array}
\]

\[
\begin{array}{ccc}
\text{edge} & 0 & 0 & 0 \\
\text{max} & \\
\end{array}
\]

Value of game 0
New column added: boosting

<table>
<thead>
<tr>
<th>R</th>
<th>P</th>
<th>S</th>
<th>(w_1)</th>
<th>(w_2)</th>
<th>(w_3)</th>
<th>(w_4)</th>
<th>margin</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>.44</td>
<td>0</td>
<td>.22</td>
<td>.33</td>
<td></td>
</tr>
</tbody>
</table>

| R | \(d_1\) | .22 | 0 | 1 | -1 | 1 | .11 |
| P | \(d_2\) | .33 | -1 | 0 | 1 | 1 | .11 | min |
| S | \(d_3\) | .44 | 1 | -1 | 0 | -1 | .11 |

Value of game **increases** from 0 to .11
Row added: on-line learning

<table>
<thead>
<tr>
<th></th>
<th>R</th>
<th>P</th>
<th>S</th>
<th>edge</th>
</tr>
</thead>
<tbody>
<tr>
<td>w_1</td>
<td>.33</td>
<td>w_2</td>
<td>.44</td>
<td>-.11</td>
</tr>
<tr>
<td>w_3</td>
<td>.22</td>
<td></td>
<td></td>
<td>-.11</td>
</tr>
<tr>
<td>R</td>
<td>d_1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>P</td>
<td>d_2</td>
<td>.22</td>
<td>-1</td>
<td>0</td>
</tr>
<tr>
<td>S</td>
<td>d_3</td>
<td>.44</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td></td>
<td>d_4</td>
<td>.33</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>

Value of game decreases from 0 to -.11
**Boosting: maximize margin incrementally**

<table>
<thead>
<tr>
<th></th>
<th>$d_1^1$</th>
<th>$d_1^2$</th>
<th>$d_1^3$</th>
<th>$d_2^1$</th>
<th>$d_2^2$</th>
<th>$d_2^3$</th>
<th>$d_3^1$</th>
<th>$d_3^2$</th>
<th>$d_3^3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_1^1$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>$d_1^2$</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$d_1^3$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

In each iteration solve optimization problem to update $d$

- Column player / oracle provides new hypothesis
- Boosting is column generation method in $d$ domain and coordinate/gradient descent in $w$ domain
Outline

1. Introduction to Boosting
2. What is Boosting?
3. LPBoost
4. Entropy Regularized LPBoost
5. Overview of Boosting algorithms
6. Conclusion and Open Problems
What is Boosting?

Boosting = greedy method for increasing margin

Converges to optimum margin w.r.t. all hypotheses

Want small number of iterations
Assumption on next weak hypothesis

For current weighting of examples, oracle returns hypothesis of edge $\geq g$

Goal

- For given $\epsilon$, produce convex combination of weak hypotheses with margin $\geq g - \epsilon$
- Number of iterations $O\left(\frac{\log N}{\epsilon^2}\right)$
Min max thm for the inseparable case

Slack variables in $\mathbf{w}$ domain = capping in $\mathbf{d}$ domain

$$\max_{\mathbf{w} \in S^t, \psi \geq 0} \min_{n=1,2,...,N} \left( \sum_{q=1}^{t} u_n^q w_q + \psi_n \right) - \frac{1}{\nu} \sum_{n=1}^{N} \psi_n$$

margin of example $n$

$$= \min_{\mathbf{d} \in S^N, \mathbf{d} \leq \frac{1}{\nu} \mathbf{1}} \max_{q=1,2,...,t} u^q \cdot \mathbf{d}$$

edge of hypothesis $q$

Notation: $u_n^q = y_n h^q(x_n)$
Outline

1. Introduction to Boosting
2. What is Boosting?
3. LPBoost
4. Entropy Regularized LPBoost
5. Overview of Boosting algorithms
6. Conclusion and Open Problems
Choose distribution that minimizes the maximum edge via LP

\[
\min_{\sum_n d_n = 1, d \leq \frac{1}{\nu}} \max_{q=1,2,\ldots,t} \mathbf{u}^q \cdot \mathbf{d} \\
\]

- All weight is put on examples with minimum soft margin
- **Brittle**: iteration bound can be linear in \( N \) on carefully constructed artificial data sets

[WGR07]
LPBoost may require $\Omega(N)$ iterations

<table>
<thead>
<tr>
<th>$\alpha_1$</th>
<th>$\alpha_2$</th>
<th>$\alpha_3$</th>
<th>$\alpha_4$</th>
<th>$\alpha_5$</th>
<th>margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

| $d_1$ | .125 | +1 | -.95 | -.93 | -.91 | -.99 |
| $d_2$ | .125 | +1 | -.95 | -.93 | -.91 | -.99 |
| $d_3$ | .125 | +1 | -.95 | -.93 | -.91 | -.99 |
| $d_4$ | .125 | +1 | -.95 | -.93 | -.91 | -.99 |
| $d_5$ | .125 | -.98 | +1 | -.93 | -.91 | +.99 |
| $d_6$ | .125 | -.97 | -.96 | +1 | -.91 | +.99 |
| $d_7$ | .125 | -.97 | -.95 | -.94 | +1 | +.99 |
| $d_8$ | .125 | -.97 | -.95 | -.93 | -.92 | +.99 |

| edge | .0137 | -.7075 | -.6900 | -.6725 | .0000 |

| value | -1 |

Warmuth (UCSC)
LPBoost may require $\Omega(N)$ iterations

<table>
<thead>
<tr>
<th></th>
<th>$\alpha_1$</th>
<th>$\alpha_2$</th>
<th>$\alpha_3$</th>
<th>$\alpha_4$</th>
<th>$\alpha_5$</th>
<th>margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_1$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
<td>-.99</td>
</tr>
<tr>
<td>$d_2$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
<td>-.99</td>
</tr>
<tr>
<td>$d_3$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
<td>-.99</td>
</tr>
<tr>
<td>$d_4$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
<td>-.99</td>
</tr>
<tr>
<td>$d_5$</td>
<td>1</td>
<td>-.98</td>
<td>+1</td>
<td>-.93</td>
<td>-.91</td>
<td>+.99</td>
</tr>
<tr>
<td>$d_6$</td>
<td>0</td>
<td>-.97</td>
<td>-.96</td>
<td>+1</td>
<td>-.91</td>
<td>+.99</td>
</tr>
<tr>
<td>$d_7$</td>
<td>0</td>
<td>-.97</td>
<td>-.95</td>
<td>-.94</td>
<td>+1</td>
<td>+.99</td>
</tr>
<tr>
<td>$d_8$</td>
<td>0</td>
<td>-.97</td>
<td>-.95</td>
<td>-.93</td>
<td>-.92</td>
<td>+.99</td>
</tr>
<tr>
<td>edge</td>
<td>value</td>
<td>-.98</td>
<td>1</td>
<td>-.93</td>
<td>-.91</td>
<td>.99</td>
</tr>
</tbody>
</table>
LPBoost may require $\Omega(N)$ iterations

<table>
<thead>
<tr>
<th>$\alpha_1$</th>
<th>$\alpha_2$</th>
<th>$\alpha_3$</th>
<th>$\alpha_4$</th>
<th>$\alpha_5$</th>
<th>margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$d_1$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_2$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_3$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_4$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_5$</td>
<td>0</td>
<td>-.98</td>
<td>+1</td>
<td>-.93</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_6$</td>
<td>1</td>
<td>-.97</td>
<td>-.96</td>
<td>+1</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_7$</td>
<td>0</td>
<td>-.97</td>
<td>-.95</td>
<td>-.94</td>
<td>+1</td>
</tr>
<tr>
<td>$d_8$</td>
<td>0</td>
<td>-.97</td>
<td>-.95</td>
<td>-.93</td>
<td>-.92</td>
</tr>
<tr>
<td>edge</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>value</td>
<td>-1</td>
<td>-.98</td>
<td>-.96</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Warmuth (UCSC)  ICML '09 Boosting Tutorial
### LPBoost may require $\Omega(N)$ iterations

<table>
<thead>
<tr>
<th>$\alpha_1$</th>
<th>$\alpha_2$</th>
<th>$\alpha_3$</th>
<th>$\alpha_4$</th>
<th>$\alpha_5$</th>
<th>margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

$\begin{align*}
\text{edge} & & & & & \\
\text{value} & -1 & -.98 & -.96 & -.94 & 1 & .99
\end{align*}$
LPBoost may require $\Omega(N)$ iterations.

<table>
<thead>
<tr>
<th>$\alpha_1$</th>
<th>$\alpha_2$</th>
<th>$\alpha_3$</th>
<th>$\alpha_4$</th>
<th>$\alpha_5$</th>
<th>margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$d_1$</th>
<th>0</th>
<th>+1</th>
<th>-.95</th>
<th>-.93</th>
<th>-.91</th>
<th>-.99</th>
<th>-.91</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_2$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
<td>-.99</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_3$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
<td>-.99</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_4$</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
<td>-.99</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_5$</td>
<td>0</td>
<td>-.98</td>
<td>+1</td>
<td>-.93</td>
<td>-.91</td>
<td>+.99</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_6$</td>
<td>0</td>
<td>-.97</td>
<td>-.96</td>
<td>+1</td>
<td>-.91</td>
<td>+.99</td>
<td>-.91</td>
</tr>
<tr>
<td>$d_7$</td>
<td>0</td>
<td>-.97</td>
<td>-.95</td>
<td>-.94</td>
<td>+1</td>
<td>+.99</td>
<td>1</td>
</tr>
<tr>
<td>$d_8$</td>
<td>1</td>
<td>-.97</td>
<td>-.95</td>
<td>-.93</td>
<td>-.92</td>
<td>+.99</td>
<td>-.92</td>
</tr>
</tbody>
</table>

| edge   | -.97 | -.95 | -.94 | -.92 | .99 |
| value  | -1   | -.98 | -.96 | -.94 | -.92 |
**LPBoost may require \( \Omega(N) \) iterations**

<table>
<thead>
<tr>
<th>( \alpha_1 )</th>
<th>( \alpha_2 )</th>
<th>( \alpha_3 )</th>
<th>( \alpha_4 )</th>
<th>( \alpha_5 )</th>
<th>margin</th>
</tr>
</thead>
<tbody>
<tr>
<td>.5</td>
<td>.0026</td>
<td>0</td>
<td>0</td>
<td>.4975</td>
<td></td>
</tr>
<tr>
<td>( d_1 )</td>
<td>0.4974</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.99</td>
</tr>
<tr>
<td>( d_2 )</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
</tr>
<tr>
<td>( d_3 )</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
</tr>
<tr>
<td>( d_4 )</td>
<td>0</td>
<td>+1</td>
<td>-.95</td>
<td>-.93</td>
<td>-.91</td>
</tr>
<tr>
<td>( d_5 )</td>
<td>0</td>
<td>-.98</td>
<td>+1</td>
<td>-.93</td>
<td>-.91</td>
</tr>
<tr>
<td>( d_6 )</td>
<td>.4898</td>
<td>-.97</td>
<td>-.96</td>
<td>+1</td>
<td>-.91</td>
</tr>
<tr>
<td>( d_7 )</td>
<td>0</td>
<td>-.97</td>
<td>-.95</td>
<td>-.94</td>
<td>+1</td>
</tr>
<tr>
<td>( d_8 )</td>
<td>.0127</td>
<td>-.97</td>
<td>-.95</td>
<td>-.93</td>
<td>-.92</td>
</tr>
<tr>
<td><strong>edge</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>value</strong></td>
<td>-1</td>
<td>-.98</td>
<td>-.96</td>
<td>-.94</td>
<td>-.92</td>
</tr>
</tbody>
</table>
Outline

1. Introduction to Boosting
2. What is Boosting?
3. LPBoost
4. Entropy Regularized LPBoost
5. Overview of Boosting algorithms
6. Conclusion and Open Problems
Entropy Regularized LPBoost

\[
\min_{\sum_n d_n = 1, d \leq \frac{1}{\nu}} \max_{q=1,2,\ldots,t} u^q \cdot d + \frac{1}{\eta} \Delta(d, d^0)
\]

- \(d_n = \frac{\exp^{-\eta \text{ soft margin of example } n}}{Z} \) "soft min"

- Form of weights first in \(\nu\)-Arc algorithm \([\text{RSS+00}]\)
- Regularization in \(d\) domain makes problem strongly convex
- Gradient of dual Lipschitz continuous in \(w\) \([\text{e.g. HL93, RW97}]\)
The effect of entropy regularization

Different distribution on the examples

LPBoost: lots of zeros / brittle

ERLPLBoost: smoother
Overview of Boosting algorithms

Outline

1. Introduction to Boosting
2. What is Boosting?
3. LPBoost
4. Entropy Regularized LPBoost
5. Overview of Boosting algorithms
6. Conclusion and Open Problems
AdaBoost

\[ d_n^t := \frac{d_{n}^{t-1} \exp(-w_t u_n^t)}{\sum_{n'} d_{n'}^{t-1} \exp(-w_t u_{n'}^t)}, \]

where \( w_t \) s.t. \( \sum_{n'} d_{n'}^{t-1} \exp(-w u_{n'}^t) \) is minimized

- Easy to implement
- Adjusts distribution so that edge of last hypothesis is zero
- Gets within half of the optimal hard margin \( \text{[RSD07]} \)
  but only in the limit
# Corrective versus totally corrective

Processing **last** hypothesis versus **all** past hypotheses

<table>
<thead>
<tr>
<th>Corrective</th>
<th>Totally Corrective</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdaBoost</td>
<td>LPBoost</td>
</tr>
<tr>
<td>LogitBoost</td>
<td>TotalBoost</td>
</tr>
<tr>
<td>AdaBoost*</td>
<td>SoftBoost</td>
</tr>
<tr>
<td>SS, Colt08</td>
<td>ERLLPBoost</td>
</tr>
</tbody>
</table>
From AdaBoost to ERLPBoost

AdaBoost

<table>
<thead>
<tr>
<th>Primal:</th>
<th>Dual:</th>
</tr>
</thead>
</table>
| \[
\min_d \Delta(d, d^{t-1}) \\
\text{s.t.} \quad d \cdot u^{t-1} = 0, \quad \|d\|_1 = 1
\] | \[
\max_w -\ln \sum_n d_n^{t-1} \exp(u_n^{t-1} w_{t-1}) \\
\text{s.t.} \quad w \geq 0
\] |

Achieves half of optimum hard margin in the limit

AdaBoost*

<table>
<thead>
<tr>
<th>Primal:</th>
<th>Dual:</th>
</tr>
</thead>
</table>
| \[
\min_d \Delta(d, d^{t-1}) \\
\text{s.t.} \quad d \cdot u^{t-1} \leq \gamma_{t-1}, \quad \|d\|_1 = 1
\] | \[
\max_w -\ln \sum_n d_n^{t-1} \exp(u_n^{t-1} w_{t-1}) - \gamma_{t-1} \|w\|_1 \\
\text{s.t.} \quad w \geq 0
\] |

where edge bound $\gamma_t$ is adjusted downward by a heuristic

Good iteration bound for reaching optimum hard margin
Overview of Boosting algorithms

**SoftBoost**

Primal:

\[
\min_d \Delta(d, d^0) \\
\text{s.t.} \quad \|d\|_1 = 1, \quad d \leq \frac{1}{\nu} 1 \\
d \cdot u^q \leq \gamma_{t-1}, \\
1 \leq q \leq t - 1
\]

Dual:

\[
\min_{w, \psi} - \ln \sum_n d_n^0 \exp(-\eta \sum_{q=1}^{t-1} u_n^q w_q) \\
- \eta \psi_n) - \frac{1}{\nu} \|\psi\|_1 - \gamma_{t-1} \|w\|_1 \\
\text{s.t.} \quad w \geq 0, \quad \psi \geq 0
\]

where edge bound \(\gamma_{t-1}\) is adjusted downward by a heuristic

**Good iteration bound for reaching soft margin**

**ERLPBoost**

Primal:

\[
\min_{d, \gamma} \gamma + \frac{1}{\eta} \Delta(d, d^0) \\
\text{s.t.} \quad \|d\|_1 = 1, \quad d \leq \frac{1}{\nu} 1 \\
d \cdot u^q \leq \gamma, \\
1 \leq q \leq t - 1
\]

Dual:

\[
\min_{w, \psi} - \frac{1}{\eta} \ln \sum_n d_n^0 \exp(-\eta \sum_{q=1}^{t-1} u_n^q w_q) \\
- \eta \psi_n) - \frac{1}{\nu} \|\psi\|_1 \\
\text{s.t.} \quad w \geq 0, \quad \|w\|_1 = 1, \quad \psi \geq 0
\]

where for the iteration bound \(\eta\) is fixed to \(\max\left(\frac{2}{\epsilon} \ln \frac{N}{\nu}, \frac{1}{2}\right)\)

**Good iteration bound for reaching soft margin**
## Iteration bounds

<table>
<thead>
<tr>
<th>Corrective</th>
<th>Totally Corrective</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdaBoost</td>
<td>LPBoost</td>
</tr>
<tr>
<td>LogitBoost</td>
<td>TotalBoost</td>
</tr>
<tr>
<td>AdaBoost*</td>
<td>SoftBoost</td>
</tr>
<tr>
<td>SS, Colt08</td>
<td>ERLPBoost</td>
</tr>
</tbody>
</table>

- **Strong oracle**: returns hypothesis with maximum edge
- **Weak oracle**: returns hypothesis with edge $\geq g$

- In $O(\frac{\log N}{\epsilon^2})$ iterations, within $\epsilon$ of maximum soft margin for strong oracle or within $\epsilon$ of $g$ for weak oracle.

- **Ditto for hard margin case**

- In $O(\frac{\log N}{g^2})$ iterations consistency with weak oracle.
Overview of Boosting algorithms

Synopsis

- LPBoost often unstable
- For safety, add relative entropy regularization
- Corrective algs
  - Sometimes easy to code
  - Fast per iteration
- Totally corrective algs
  - Smaller number of iterations
  - Nevertheless faster overall time

**Weak** versus **strong** oracle makes a big difference in practice
$O\left(\frac{\log N}{\epsilon^2}\right)$ iteration bounds

**Good**
- Bound is major design tool
- Any reasonable Boosting algorithm should have this bound

**Bad**
- Bound is weak
  - $\epsilon = .01 \quad N \leq 1.2 \times 10^5$
  - $\epsilon = .001 \quad N \leq 1.7 \times 10^7$

- Why are totally corrective algorithms much better in practice?
Overview of Boosting algorithms

Lower bounds on the number of iterations

- Majority of $\Omega\left(\frac{\log N}{g^2}\right)$ hypotheses for achieving consistency with weak oracle of guarantee $g$ [Fr95]

- Later: $\Omega\left(\frac{1}{\epsilon^2}\right)$ iteration bound for getting within $\epsilon$ of hard margin with strong oracle
Outline

1. Introduction to Boosting
2. What is Boosting?
3. LPBoost
4. Entropy Regularized LPBoost
5. Overview of Boosting algorithms
6. Conclusion and Open Problems
Conclusion

- Adding relative entropy regularization of LPBoost leads to good boosting alg.
- Boosting is instantiation of MaxEnt and MinxEnt principles \[\text{[Jaines 57, Kullback 59]}\]
- Relative entropy regularization smoothens one-norm regularization

Open

- When hypotheses have one-sided error then \(O\left(\frac{\log N}{\epsilon}\right)\) iterations suffice \[\text{[As00, HW03]}\]
- Does ERLPBoost have \(O\left(\frac{\log N}{\epsilon}\right)\) bound when hypotheses one-sided?
- Strengthen general lower bound to \(\Omega\left(\frac{\log N}{\epsilon^2}\right)\)
- Compare ours with Freund’s algorithms that don’t just cap, but forget examples
Acknowledgement

- Rob Schapire and Yoav Freund for pioneering Boosting
- Gunnar Rätsch for bringing in optimization
- Karen Glocer for helping with figures and plots