The Rate of Convergence of AdaBoost

Indraneel Mukherjee
Cynthia Rudin
Rob Schapire
Top 10 algorithms in data mining

Xindong Wu · Vipin Kumar · J. Ross Quinlan · Joydeep Ghosh · Qiang Yang · Hiroshi Motoda · Geoffrey J. McLachlan · Angus Ng · Bing Liu · Philip S. Yu · Zhi-Hua Zhou · Michael Steinbach · David J. Hand · Dan Steinberg

Received: 9 July 2007 / Revised: 28 September 2007 / Accepted: 8 October 2007
Published online: 4 December 2007
© Springer-Verlag London Limited 2007

Abstract This paper presents the top 10 data mining algorithms identified by the IEEE International Conference on Data Mining (ICDM) in December 2006: C4.5, k-Means, SVM, Apriori, EM, PageRank, AdaBoost, kNN, Naive Bayes, and CART. These top 10 algorithms are among the most influential data mining algorithms in the research community. With each algorithm, we provide a description of the algorithm, discuss the impact of the algorithm, and review current and further research on the algorithm. These 10 algorithms cover classification,
AdaBoost (Freund and Schapire 97)

An Empirical Comparison of Supervised Learning Algorithms

Rich Caruana  
Alexandru Niculescu-Mizil  
Department of Computer Science, Cornell University, Ithaca, NY 14853 USA

With excellent performance on all eight metrics, calibrated boosted trees were the best learning algorithm overall. Random forests are close second, followed by uncalibrated bagged trees, calibrated SVMs, and uncalibrated neural nets. The models that performed poorest were naive bayes, logistic regression, decision trees, and boosted stumps. Although some methods clearly perform better or worse than other methods on average, there is significant variability across the problems and metrics. Even the best models sometimes perform poorly, and models with poor average
Basic properties of AdaBoost’s convergence are still not fully understood.
Basic properties of AdaBoost’s convergence are still not fully understood.

We address one of these basic properties: convergence rates with no assumptions.
• AdaBoost is known for its ability to combine “weak classifiers” into a “strong” classifier

• AdaBoost iteratively minimizes “exponential loss”

  (Breiman 99, Frean and Downs, 1998; Friedman et al., 2000; Friedman, 2001; Mason et al., 2000; Onoda et al., 1998; Ratsch et al., 2001; Schapire and Singer, 1999)
• AdaBoost is known for its ability to combine “weak classifiers” into a “strong” classifier

• AdaBoost iteratively minimizes “exponential loss”

(Breiman 99, Frean and Downs, 1998; Friedman et al., 2000; Friedman, 2001; Mason et al., 2000; Onoda et al., 1998; Ratsch et al., 2001; Schapire and Singer, 1999)

Examples: \{((x_i, y_i)) \}_{i=1,...,m}, \text{ with each } (x_i, y_i) \in \mathcal{X} \times \{-1,1\}

Hypotheses: \( H = \{h_1,...,h_N\}, \text{ where } h_j : \mathcal{X} \to [-1,1] \)
• AdaBoost is known for its ability to combine “weak classifiers” into a “strong” classifier

• AdaBoost iteratively minimizes “exponential loss” (Breiman 99, Frean and Downs, 1998; Friedman et al., 2000; Friedman, 2001; Mason et al., 2000; Onoda et al., 1998; Ratsch et al., 2001; Schapire and Singer, 1999)

Examples: \(\{(x_i, y_i)\}_{i=1,...,m}\), with each \((x_i, y_i) \in \mathcal{X} \times \{-1,1\}\)

Hypotheses: \(H = \{h_1, ..., h_N\}\), where \(h_j : \mathcal{X} \rightarrow [-1,1]\)

Combination: \(F(x) = \lambda_1 h_1(x) + \ldots + \lambda_N h_N(x)\)
• AdaBoost is known for its ability to combine “weak classifiers” into a “strong” classifier

• AdaBoost iteratively minimizes “exponential loss” (Breiman 99, Frean and Downs, 1998; Friedman et al., 2000; Friedman, 2001; Mason et al., 2000; Onoda et al., 1998; Ratsch et al., 2001; Schapire and Singer, 1999)

Examples: \(\{(x_i, y_i)\}_{i=1,...,m}\), with each \((x_i, y_i) \in \mathcal{X} \times \{-1,1\}\)

Hypotheses: \(H = \{h_1,...,h_N\}\), where \(h_j: \mathcal{X} \to [-1,1]\)

Combination: \(F(x) = \lambda_1 h_1(x) + \ldots + \lambda_N h_N(x)\)

\[
\text{misclassif. error} \quad \leq \quad \text{exponential loss} \\
\frac{1}{m} \sum_{i=1}^{m} 1_{[y_iF(x_i) \leq 0]} \quad \leq \quad \frac{1}{m} \sum_{i=1}^{m} \exp(-y_iF(x_i))
\]
AdaBoost is known for its ability to combine "weak classifiers" into a "strong" classifier. AdaBoost iteratively minimizes "exponential loss" (Breiman 99, Frean and Downs, 1998; Friedman et al., 2000; Friedman, 2001; Mason et al., 2000; Onoda et al., 1998; Ratsch et al., 2001; Schapire and Singer, 1999)

Examples: \( \{(x_i, y_i)\}_{i=1,...,m} \), with each \((x_i, y_i) \in \mathcal{X} \times \{-1, 1\}\)

Hypotheses: \( H = \{h_1, ..., h_N\} \), where \( h_j : \mathcal{X} \rightarrow [-1, 1] \)

Exponential loss:

\[
L(\lambda) = \frac{1}{m} \sum_{i=1}^{m} \exp \left( -\sum_{j=1}^{N} \lambda_j y_i h_j(x_i) \right)
\]
Examples: \( \{(x_i, y_i)\}_{i=1,...,m} \), with each \((x_i, y_i) \in \mathcal{X} \times \{-1, 1\}\)

Hypotheses: \( H = \{h_1, ..., h_N\} \), where \( h_j : \mathcal{X} \rightarrow [-1, 1] \)

Exponential loss:

\[
L(\lambda) = \frac{1}{m} \sum_{i=1}^{m} \exp \left( -\sum_{j=1}^{N} \lambda_j y_i h_j(x_i) \right)
\]
Examples: \( \{(x_i, y_i)\}_{i=1,...,m} \), with each \((x_i, y_i) \in X \times \{-1,1\}\)

Hypotheses: \( H = \{h_1, ..., h_N\} \), where \( h_j : X \rightarrow [-1,1] \)

Exponential loss:

\[
L(\lambda) = \frac{1}{m} \sum_{i=1}^{m} \exp \left( -\sum_{j=1}^{N} \lambda_j y_i h_j(x_i) \right)
\]
Known:

- AdaBoost converges asymptotically to the minimum of the exponential loss (Collins et al 2002, Zhang and Yu 2005)
Known:

- AdaBoost converges asymptotically to the minimum of the exponential loss (Collins et al 2002, Zhang and Yu 2005)
- Convergence rates under assumptions:
  - “weak learning” assumption holds, hypotheses are better than random guessing (Freund and Schapire 1997, Schapire and Singer 1999)
  - assume that a finite minimizer exists (Rätsch et al 2002, many classic results)
Known:

- AdaBoost converges asymptotically to the minimum of the exponential loss (Collins et al 2002, Zhang and Yu 2005)
- Convergence rates under assumptions:
  - “weak learning” assumption holds, hypotheses are better than random guessing (Freund and Schapire 1997, Schapire and Singer 1999)
  - assume that a finite minimizer exists (Rätsch et al 2002, many classic results)
- Conjectured by Schapire (2010) that fast convergence rates hold without any assumptions.
Known:

• AdaBoost converges asymptotically to the minimum of the exponential loss (Collins et al 2002, Zhang and Yu 2005)

• Convergence rates under assumptions:
  • “weak learning” assumption holds, hypotheses are better than random guessing (Freund and Schapire 1997, Schapire and Singer 1999)
  • assume that a finite minimizer exists (Rätsch et al 2002, many classic results)

• Conjectured by Schapire (2010) that fast convergence rates hold without any assumptions.

• Convergence rate is relevant for consistency of AdaBoost (Bartlett and Traskin 2007).
Outline

• **Convergence Rate 1: Convergence to a target loss**
  “Can we get within $\epsilon$ of a ‘reference’ solution?”

• **Convergence Rate 2: Convergence to optimal loss**
  “Can we get within $\epsilon$ of an optimal solution?”
Main Messages

• Usual approaches assume a finite minimizer
  – Much more challenging not to assume this!

• Separated two different modes of analysis
  – comparison to reference, comparison to optimal
  – different rates of convergence are possible in each

• Analysis of convergence rates often ignore the “constants”
  – we show they can be extremely large in the worst case
• Convergence Rate 1: Convergence to a target loss
  “Can we get within $\epsilon$ of a “reference” solution?”

Based on a conjecture that says...
"At iteration $t$, $L(\lambda^t)$ will be at most $\epsilon$ more than that of any parameter vector of $\ell_1$-norm bounded by $B$ in a number of rounds that is at most a polynomial in $\log N, m, B$, and $1/\epsilon$."
"At iteration $t$, $L(\lambda^t)$ will be at most $\epsilon$ more than that of any parameter vector of $\ell_1$-norm bounded by $B$ in a number of rounds that is at most a polynomial in $\log N, m, B$, and $1/\epsilon$. "
"At iteration $t$, $L(\lambda^t)$ will be at most $\varepsilon$ more than that of any parameter vector of $\ell_1$-norm bounded by $B$ in a number of rounds that is at most a polynomial in $\log N, m, B$, and $1/\varepsilon$."

radius $B$
"At iteration $t$, $L(\lambda^t)$ will be at most $\epsilon$ more than that of any parameter vector of $\ell_1$-norm bounded by $B$ in a number of rounds that is at most a polynomial in $\log N, m, B, \text{ and } 1/\epsilon$.\"
"At iteration $t$, $L(\lambda^t)$ will be at most $\epsilon$ more than that of any parameter vector of $\ell_1$-norm bounded by $B$ in a number of rounds that is at most a polynomial in $\log N, m, B$, and $1/\epsilon$. "
This happens at:

\[ t \leq poly \left( \log N, m, B, \frac{1}{\epsilon} \right) \]
This happens at:

\[ t \leq \text{poly}\left(\log N, m, B, \frac{1}{\epsilon}\right) \]
This happens at:

\[ t \leq \text{poly}\left(\log N, m, B, \frac{1}{\epsilon}\right) \]
Theorem 1: For any $\lambda^* \in \mathbb{R}^N$, AdaBoost achieves loss at most $L(\lambda^*) + \epsilon$ in at most $13 \| \lambda^* \|_1^6 \epsilon^{-5}$ rounds.
Theorem 1: For any $\lambda^* \in \mathbb{R}^N$, AdaBoost achieves loss at most $L(\lambda^*) + \epsilon$ in at most $13 \| \lambda^* \|_1^6 \epsilon^{-5}$ rounds.

\[ poly\left( \log N, m, B, \frac{1}{\epsilon} \right) \]
Theorem 1: For any $\lambda^* \in \mathbb{R}^N$, AdaBoost achieves loss at most $L(\lambda^*) + \epsilon$ in at most $13 \| \lambda^* \|_1^6 \epsilon^{-5}$ rounds.

$$poly\left(\log N, m, B, \frac{1}{\epsilon}\right)$$

Best known previous result is that it takes at most order $e^{1/\epsilon^2}$ rounds (Bickel et al).
Theorem 1: For any $\lambda^* \in \mathbb{R}^N$, AdaBoost achieves loss at most $L(\lambda^*) + \epsilon$ in at most $13 \| \lambda^* \|_1^6 \epsilon^{-5}$ rounds.

- Dependence on $\| \lambda^* \|_1$ is necessary for many datasets.

Lemma: There are simple datasets for which the number of rounds required to achieve loss $L^*$ is at least (roughly) the norm of the smallest solution achieving loss $L^*$.
Theorem 1: For any $\lambda^* \in \mathbb{R}^N$, AdaBoost achieves loss at most $L(\lambda^*) + \epsilon$ in at most $13 \| \lambda^* \|_1^6 \epsilon^{-5}$ rounds.

- Dependence on $\| \lambda^* \|_1$ is necessary for many datasets.

Lemma: There are simple datasets for which the number of rounds required to achieve loss $L^*$ is at least

$$\inf \left\{ \| \lambda \|_1 : L(\lambda) \leq L^* \right\} / 2 \ln m$$
Theorem 1: For any \( \lambda^* \in \mathbb{R}^N \), AdaBoost achieves loss at most \( L(\lambda^*) + \epsilon \) in at most \( 13 \| \lambda^* \|_1^6 \epsilon^{-5} \) rounds.

• Dependence on \( \| \lambda^* \|_1 \) is necessary for many datasets.

Lemma: There are simple datasets for which the number of rounds required to achieve loss \( L^* \) is at least

\[
\inf \left\{ \| \lambda \|_1 : L(\lambda) \leq L^* \right\} / 2 \ln m
\]

Lemma: There are simple datasets for which the norm of the smallest solution achieving loss \( L^* \) is exponential in the number of examples.
Theorem 1: For any $\lambda^* \in \mathbb{R}^N$, AdaBoost achieves loss at most $L(\lambda^*) + \epsilon$ in at most $13 \||\lambda^*\|_1^6 \epsilon^{-5}$ rounds.

- Dependence on $\||\lambda^*\|_1$ is necessary for many datasets.

**Lemma:** There are simple datasets for which the number of rounds required to achieve loss $L^*$ is at least

$$\inf \left\{ ||\lambda||_1 : L(\lambda) \leq L^* \right\} / 2 \ln m$$

**Lemma:** There are simple datasets for which

$$\inf \left\{ ||\lambda||_1 : L(\lambda) \leq \frac{2}{m} + \epsilon \right\} \geq \left( 2^{m-2} - 1 \right) \ln(1 / (3\epsilon))$$
Theorem 1: For any $\lambda^* \in \mathbb{R}^N$, AdaBoost achieves loss at most $L(\lambda^*) + \epsilon$ in at most $13 \| \lambda^* \|_1^6 \epsilon^{-5}$ rounds.
Theorem 1: For any $\lambda^* \in \mathbb{R}^N$, AdaBoost achieves loss at most $L(\lambda^*) + \epsilon$ in at most $13 \| \lambda^* \|_1^6 \epsilon^{-5}$ rounds.

Conjecture: AdaBoost achieves loss at most $L(\lambda^*) + \epsilon$ in at most $O(B^2 / \epsilon)$ rounds.
Rate on a Simple Dataset (Log scale)

Number of rounds:

- 10
- 100
- 1000
- 10000
- 1e+05

Loss (Optimal Loss):

- 3e-06
- 3e-05
- 3e-04
- 3e-03
- 3e-02

Graph shows the rate on a simple dataset with a logarithmic scale.
Outline

• Convergence Rate 1: Convergence to a target loss
  “Can we get within $\epsilon$ of a “reference” solution?”

• Convergence Rate 2: Convergence to optimal loss
  “Can we get within $\epsilon$ of an optimal solution?”
Theorem 2: AdaBoost reaches within $\epsilon$ of the optimal loss in at most $C / \epsilon$ rounds, where $C$ only depends on the data.
Theorem 2: AdaBoost reaches within $\epsilon$ of the optimal loss in at most $C/\epsilon$ rounds, where $C$ only depends on the data.

- Better dependence on $\epsilon$ than Theorem 1, actually optimal.
- Doesn’t depend on the size of the best solution within a ball
- Can’t be used to prove the conjecture because in some cases $C>2^m$. (Mostly it’s much smaller.)
Theorem 2: AdaBoost reaches within $\epsilon$ of the optimal loss in at most $C / \epsilon$ rounds, where $C$ only depends on the data.

- Main tool is the "decomposition lemma"
  - Says that examples fall into 2 categories,
    - Zero loss set $Z$
    - Finite margin set $F$.
  - Similar approach taken independently by (Telgarsky, 2011)
Theorem 2: AdaBoost reaches within $\epsilon$ of the optimal loss in at most $C / \epsilon$ rounds, where $C$ only depends on the data.
Theorem 2: AdaBoost reaches within $\epsilon$ of the optimal loss in at most $C / \epsilon$ rounds, where $C$ only depends on the data.
Theorem 2: AdaBoost reaches within $\epsilon$ of the optimal loss in at most $C / \epsilon$ rounds, where $C$ only depends on the data.
Decomposition Lemma

For any dataset, there exists a partition of the training examples into $Z$ and $F$ s.t. these hold simultaneously:

1.) For some $\gamma > 0$, there exists vector $\eta^+$, $\| \eta^+ \|_1 = 1$ such that:

$$\forall i \in Z, \sum_j \eta_j^+ y_i h_j(x_i) \geq \gamma,$$  
(Margins are at least gamma in $Z$)

$$\forall i \in F, \sum_j \eta_j^+ y_i h_j(x_i) = 0,$$  
(Examples in $F$ have zero margins)

2.) The optimal loss considering only examples in $F$ is achieved by some finite $\eta^*$. 
For any dataset, there exists a partition of the training examples into $Z$ and $F$ s.t. these hold simultaneously:

1.) For some $\gamma > 0$, there exists vector $\eta^+$, $\|\eta^+\|_1=1$ such that:

$$\forall i \in Z, \sum_j \eta_j^+ y_i h_j(x_i) \geq \gamma,$$  
(Margins are at least gamma in $Z$)

$$\forall i \in F, \sum_j \eta_j^+ y_i h_j(x_i) = 0,$$  
(Examples in $F$ have zero margins)

2.) The optimal loss considering only examples in $F$ is achieved by some finite $\eta^*$.
Decomposition Lemma

For any dataset, there exists a partition of the training examples into $Z$ and $F$ s.t. these hold simultaneously:

1.) For some $\gamma > 0$, there exists vector $\eta^+$, $\|\eta^+\|_1 = 1$ such that:

$$\forall i \in Z, \sum_j \eta_j^+ y_i h_j(x_i) \geq \gamma,$$  \hspace{1cm} (Margins are at least gamma in $Z$)

$$\forall i \in F, \sum_j \eta_j^+ y_i h_j(x_i) = 0,$$  \hspace{1cm} (Examples in $F$ have zero margins)

2.) The optimal loss considering only examples in $F$ is achieved by some finite $\eta^*$. 
We provide a conjecture about dependence on $m$.

Conjecture: If hypotheses are $\{-1,0,1\}$-valued, AdaBoost converges to within $\epsilon$ of the optimal loss within

$$2^{O(m \ln m)} \epsilon^{-1+o(1)}$$ rounds.

This would give optimal dependence on $m$ and $\epsilon$ simultaneously.
To summarize

• Two rate bounds, one depends on the size of the best solution within a ball and has dependence $\epsilon^{-5}$.
• The other depends only on $C/\epsilon$ but $C$ can be doubly exponential in $m$.
• Many lower bounds and conjectures in the paper.
To summarize

• Two rate bounds, one depends on the size of the best solution within a ball and has dependence $\epsilon^{-5}$.

• The other depends only on $C/\epsilon$ but $C$ can be doubly exponential in $m$.

• Many lower bounds and conjectures in the paper.

Thank you