# On the Convergence of Boosting Procedures

**Tong Zhang** TZHANG@WATSON.IBM.COM
IBM T.J. Watson Research Center, Yorktown Heights, NY

**Bin Yu** BINYU@STAT.BERKELEY.EDU
Department of Statistics, University of California at Berkeley, CA

## Abstract

A boosting algorithm seeks to minimize empirically a loss function in a greedy fashion. The resulted estimator takes an additive function form and is built iteratively by applying a base estimator (or learner) to updated samples depending on the previous iterations. This paper studies convergence of boosting when it is carried out over the linear span of a family of basis functions. For general loss functions, we prove the convergence of boosting's greedy optimization to the infinimum of the loss function over the linear span. As a side product, these results reveal the importance of restricting the greedy search step sizes, as known in practice through the works of Friedman and others.

## 1. Introduction

In this paper we consider boosting algorithms for classification and regression. These algorithms present one of the major progresses in machine learning. In their original version, the computational aspect is explicitly specified as part of the estimator/algorithm. That is, the empirical minimization of an appropriate loss function is carried out in a greedy fashion. Boosting algorithms construct composite estimators using often simple base estimators through the greedy fitting procedure. For the squared loss function, they were often referred in the signal processing community as *matching pursuit* (Mallat & Zhang, 1993). More recently, it was noticed that the Adaboost method proposed in the Machine Learning community (Freund & Schapire, 1997) could also be regarded as stage-wise fitting of additive models under the exponential loss function (Breiman, 1998; Breiman, 1999; Friedman et al., 2000; Mason et al., 2000; Schapire & Singer, 1999). In this paper, we use the term *boosting* to indicate a greedy stage-wise procedure to minimize a certain loss function empirically. And the abstract formulation will be presented in Section 2.

In spite of significant practical interests in boosting procedures, their convergence properties are not yet fully understood. Previous studies have been focused on special loss functions. Specifically, Mallat and Zhang proved the convergence of matching pursuit in (Mallat & Zhang, 1993). in (Breiman, 2000), Breiman obtained a convergence result of boosting with the exponential loss function for $\pm 1$-trees (under some smoothness assumptions on the underlying distribution). In (Collins et al., 2002), a Bregman divergence based analysis was given. A convergence result was also obtained in (Mason et al., 2000) for a gradient descent version of boosting.

None of these studies can provide any information on the numerical rate of convergence for the optimization. The question of numerical rate of convergence has been studied when one works with the 1-norm regularized version of boosting where we assume that the optimization is performed in the convex hull of the basis functions. Specifically, for function estimation under the least squares loss, the convergence of greedy algorithm in convex hull was studied in (Jones, 1992; Lee et al., 1996). For general loss functions, the convergence of greedy algorithms (again, the optimization is restricted to the convex hull) was recently studied in (Zhang, 2003a).[1] In this paper, we apply the same underlying idea to the standard boosting procedure where we do not limit the optimization to the convex hull of the basis functions. The resulting bound provides information on the rate of convergence for the optimization. Our analysis reproduces and generalizes previous convergence results such as that of matching pursuit and Adaboost. An interesting observation of our analysis is the important role of small step-size in the convergence of boosting procedures. This provides

[1] Also see (Rätsch et al., 2001) for a related study, but their analysis depends on quantities that can ill-behave.

*Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)*, Washington DC, 2003.

some theoretical justification for Friedman's empirical observation (Friedman, 2001) that using small step-sizes almost always helps in boosting procedures.

Due to the limitation of space, we will only include proofs for the two main theorems.

## 2. Abstract Boosting Procedure

We now describe the basics to define the boosting procedure that we will analyze in this paper. A similar set-up can be found in (Mason et al., 2000). The main difference is that they use a gradient descent rule in their boosting procedure while here we use approximate minimization.

Let $S$ be a set of real-valued functions, and define

$$\mathrm{span}(S) = \left\{ \sum_{j=1}^{m} w^j f^j : f^j \in S, w^j \in R, m \in Z^+ \right\},$$

which forms a linear function space. For all $f \in \mathrm{span}(S)$, we can define its 1-norm with respect to the basis $S$ as:

$$\|f\|_1 = \inf \left\{ \|w\|_1; f = \sum_{j=1}^{m} w^j f^j : f^j \in S, m \in Z^+ \right\}. \quad (1)$$

We want to find a function $\bar{f} \in \mathrm{span}(S)$ that approximately solves the following optimization problem:

$$\inf_{f \in \mathrm{span}(S)} A(f), \quad (2)$$

where $A$ is a convex function of $f$ defined on $\mathrm{span}(S)$. Note that the optimal value may not be achieved by any $f \in \mathrm{span}(S)$. Also for certain formulations (such as logistic regression), it is possible that the optimal value is $-\infty$.

The abstract form of greedy-boosting procedure (with restricted step size) considered in this paper is given by the following algorithm:

**Algorithm 2.1** *(Greedy Boosting)*

Pick $f_0 \in \mathrm{span}(S)$
**for** $k = 0, 1, 2, \ldots$
  Select a closed subset $\Lambda_k \subset R$ such that $0 \in \Lambda_k$ and $\Lambda_k = -\Lambda_k$
  Find $\bar{\alpha}_k \in \Lambda_k$ and $\bar{g}_k \in S$ to approximately minimize: $(\alpha_k, g_k) \to A(f_k + \alpha_k g_k)$ $(*)$
  Let $f_{k+1} = f_k + \bar{\alpha}_k \bar{g}_k$
**end**

**Remark 2.1** *The approximate minimization of* $(*)$ *in Algorithm 2.1 should be interpreted as finding* $\bar{\alpha}_k \in \Lambda_k$ *and* $\bar{g}_k \in S$ *such that*

$$A(f_k + \bar{\alpha}_k \bar{g}_k) \leq \inf_{\alpha_k \in \Lambda_k, g_k \in S} A(f_k + \alpha_k g_k) + \epsilon_k, \quad (3)$$

*where* $\epsilon_k \geq 0$ *is a sequence of non-negative numbers that converges to zero.*

**Remark 2.2** *Our convergence analysis allows the choice of* $\Lambda_k$ *to depend on the previous steps of the algorithm. However, the most interesting* $\Lambda_k$ *for the purpose of this paper will be independent of previous steps of the algorithm:*

- $\Lambda_k = R$
- $\sup \Lambda_k = \tilde{h}_k$ *where* $\tilde{h}_k \geq 0$ *and* $\tilde{h}_k \to 0$.

*As we will see later, the restriction of* $\alpha_k$ *to the subset* $\Lambda_k \subset R$ *is useful in the convergence analysis.*

## 3. Assumptions

For all $f \in \mathrm{span}(S)$ and $g \in S$, we define a real-valued function $A_{f,g}(\cdot)$ as:

$$A_{f,g}(h) = A(f + hg).$$

**Definition 3.1** *Let* $A(f)$ *be a function of* $f$ *defined on* $\mathrm{span}(S)$. *Denote by* $\mathrm{span}(S)'$ *the dual space of* $\mathrm{span}(S)$ *(that is, the space of real valued linear functionals on* $\mathrm{span}(S)$*). We say that* $A$ *is differentiable with gradient* $\nabla A \in \mathrm{span}(S)'$ *if for all* $f, g \in \mathrm{span}(S)$*:*

$$\lim_{h \to 0} \frac{1}{h}(A(f + hg) - A(f)) = \nabla A(f)^T g,$$

*where* $\nabla A(f)^T g$ *denotes the value of linear functional* $\nabla A(f)$ *at* $g$.

For reference, we shall state the following assumption which is required in our analysis:

**Assumption 3.1** *Let* $A(f)$ *be a convex function of* $f$ *defined on* $\mathrm{span}(S)$, *which satisfies the following conditions*

1. $A(f)$ *is differentiable with gradient* $\nabla A$.
2. *For all* $f \in \mathrm{span}(S)$ *and* $g \in S$: $A_{f,g}(h)$ *(as a real function of* $h$*) is second order differentiable and the second derivative satisfies:*

$$A''_{f,g}(0) \leq M(\|f\|_1), \quad (4)$$

*where* $M(\cdot)$ *is a non-decreasing real-valued function.*

The $M(\cdot)$ function will appear in the convergence analysis. Although our analysis can handle unbounded $M(\cdot)$, the most interesting boosting examples have bounded $M(\cdot)$ (as we will show shortly).

For statistical estimation problems such as classification and regression with a covariate or predictor variable $X$ and a real response variable $Y$ having a joint distribution, we are interested in the following form of $A(f)$ in (2):

$$A(f) = \psi(E_{X,Y}\phi(f(X), Y)), \tag{5}$$

where $\phi(f, y)$ is a loss function that is convex in $f$. $\psi$ is a monotonic increasing auxiliary function which is introduced so that $A(f)$ is convex and $M(\cdot)$ behaves nicely (e.g. bounded). The behavior of Algorithm 2.1 is not affected by the choice of $\psi$ as long as $\epsilon_k$ in (3) is appropriately redefined. We may thus always take $\psi(u) = u$, but choosing other auxiliary functions can be convenient for certain problems in our analysis since the resulted formulation has bounded $M(\cdot)$ function (see examples given below). We have also used $E_{X,Y}$ to indicate the expectation with respect to the joint distribution of $(X, Y)$.

An important application of boosting is binary classification. In this case, it is very natural for us to use a set of basis functions that satisfy the following conditions:

$$\sup_{g \in S, x} |g(x)| \leq 1, \qquad y = \pm 1. \tag{6}$$

For certain loss functions (such as least squares), this condition can be relaxed. In the classification literature, $\phi(f, y)$ usually has a form $\phi(fy)$. The following examples include commonly used loss functions. They show that for a typical boosting loss function $\phi$, there exists a constant $M$ such that $\sup_a M(a) \leq M$.

### 3.1. Logistic Regression

This is the traditional loss function used in statistics, which is given by

$$\phi(f, y) = \ln(1 + \exp(-fy)), \quad \psi(u) = u.$$

We assume that the basis functions satisfy the condition

$$\sup_{g \in S, x} |g(x)| \leq 1, \qquad y = \pm 1.$$

It can be verified that $A(f)$ is convex differentiable. We also have

$$A''_{f,g}(0) = E_{X,Y} \frac{g(X)^2 Y^2}{(1 + e^{f(X)Y})(1 + e^{-f(X)Y})} \leq 1/4.$$

### 3.2. Exponential Loss

This loss function is used in the AdaBoost algorithm, which is the original boosting procedure for classification problems. It is given by

$$\phi(f, y) = \exp(-fy), \quad \psi(u) = \ln u.$$

Again we assume that the basis functions satisfy the condition

$$\sup_{g \in S, x} |g(x)| \leq 1, \qquad y = \pm 1.$$

In this case, it is also not difficult to verify that $A(f)$ is convex differentiable. Hence we also have

$$A''_{f,g}(0) \leq \frac{E_{X,Y} g(X)^2 Y^2 \exp(-f(X)Y)}{E_{X,Y} \exp(-f(X)Y)} \leq 1.$$

### 3.3. Least Squares

The least squares formulation has been widely studied in regression, but can also be applied to classification problems (Bühlmann & Yu, 2003; Friedman, 2001). Greedy boosting-like procedure for least squares was firstly proposed in the signal processing community, where it was called *matching pursuit* (Mallat & Zhang, 1993). The loss function is given by

$$\phi(f, y) = \frac{1}{2}(f - y)^2, \quad \psi(u) = u.$$

We impose the following weaker condition on the basis functions

$$\sup_{g \in S} E_X g(X)^2 \leq 1, \quad E_Y Y^2 < \infty.$$

It is clear that $A(f)$ is convex differentiable, and the second derivative is bounded as

$$A''_{f,g}(0) = E_X g(X)^2 \leq 1.$$

### 3.4. Modified Least Squares

For classification problems, we may consider the following modified version of the least squares loss which has a better approximation property (Zhang, 2003b):

$$\phi(f, y) = \frac{1}{2} \max(1 - fy, 0)^2, \quad \psi(u) = u.$$

Since this loss is for classification problems, we impose the following condition

$$\sup_{g \in S} E_X g(X)^2 \leq 1, \quad y = \pm 1.$$

It is clear that $A(f)$ is convex differentiable, and we have the following bound for the second derivative

$$A''_{f,g}(0) \leq E_X g(X)^2 \leq 1.$$

### 3.5. $p$-norm boosting

$p$-norm loss can be interesting both for regression and classification. In this paper we will only consider the case with $p \geq 2$:

$$\phi(f,y) = |f-y|^p, \quad \psi(u) = \frac{1}{2(p-1)}u^{2/p}.$$

We impose the following condition

$$\sup_{g\in S} E_X|g(X)|^p \leq 1, \quad E_Y|Y|^p < \infty.$$

It can be shown that

$$A''_{f,g}(h) \leq E_{X,Y}^{2/p}|g(X)|^p \leq 1.$$

## 4. Convergence Analysis

In this section, we consider the convergence behavior of $f_k$ obtained from the greedy boosting procedure as $k$ increases.

Given an arbitrary fixed reference function $\bar{f} \in \text{span}(S)$ with the representation

$$\bar{f} = \sum_j \bar{w}^j \bar{f}_j, \quad \bar{f}_j \in S, \tag{7}$$

we would like to compare $A(f_k)$ to $A(\bar{f})$. Since $\bar{f}$ is arbitrary, such a comparison will be used to obtain a bound on the numerical convergence rate.

Given any finite subset $S' \subset S$ such that $S' \supset \{\bar{f}_j\}$, we can represent $\bar{f}$ as

$$\bar{f} = \sum_{g\in S'} \bar{w}^g_{S'} g,$$

where $\bar{w}^g_{S'} = \bar{w}^j$ when $g = \bar{f}_j$ for some $j$, and $\bar{w}^g_{S'} = 0$ when $g \notin \{\bar{f}_j\}$. A quantity that will appear in our analysis is $\|\bar{w}_{S'}\|_1 = \sum_{g\in S'} |\bar{w}^g_{S'}|$. Since $\|\bar{w}_{S'}\|_1 = \|\bar{w}\|_1$, without any confusion, we will still denote $\bar{w}_{S'}$ by $\bar{w}$ with the convention that $\bar{w}^g = 0$ for all $g \notin \{\bar{f}_j\}$.

Given this reference function $\bar{f}$, we consider a representation of $f_k$ as a linear combination of a finite number of functions $S_k \subset S$, where $S_k \supset \{\bar{f}_j\}$ to be chosen later,

$$f_k = \sum_{g\in S_k} \beta^g_k f^g_k. \tag{8}$$

With this representation, we define

$$\Delta W_k = \|\bar{w} - \beta_k\|_1 = \sum_{g\in S_k} |\bar{w}^g - \beta^g_k|.$$

In our convergence analysis, we will specify convergence bounds in terms of $\|\bar{w}\|_1$ and a sequence of non-decreasing numbers $s_k$ satisfying the following condition:

$$s_k = \|f_0\|_1 + \sum_{i=0}^{k-1} h_i, \qquad |\bar{\alpha}_k| \leq h_k \in \Lambda_k, \tag{9}$$

where $\{\bar{\alpha}_k\}$ are the step-sizes in (3) that are computed in the boosting algorithm.

Using the definition of 1-norm for $\bar{f}, f_0 \in \text{span}(S)$ in (1). It is clear that $\forall \epsilon > 0$, we can choose a finite subset $S_k \subset S$, vector $\beta_k$ and vector $\bar{w}$ such that

$$\|\beta_k\|_1 = \sum_{g\in S_k} |\beta^g_k| \leq s_k + \epsilon/2, \qquad \|\bar{w}\|_1 \leq \|\bar{f}\|_1 + \epsilon/2.$$

It follows that with appropriate representation, the following inequality holds for all $\epsilon > 0$:

$$\Delta W_k \leq s_k + \|\bar{f}\|_1 + \epsilon. \tag{10}$$

### 4.1. One-step analysis

The purpose of this section is to show that $A(f_{k+1}) - A(\bar{f})$ decreases from $A(f_k) - A(\bar{f})$ by a reasonable quantity. Cascading this analysis leads to a numerical rate of convergence for the boosting procedure.

The basic idea is to upper bound the minimum of a set of numbers by an appropriately chosen weighted average of these numbers. This proof technique, which we shall call "averaging method", was used in (Jones, 1992; Lee et al., 1996; Zhang, 2003a) for analyses of greedy type algorithms.

For $h_k$ that satisfies (9), the symmetry of $\Lambda_k$ implies $h_k \text{sign}(\bar{w}^g - \beta^g_k) \in \Lambda_k$. Therefore the approximate minimization step (3) implies that for all $g \in S_k$, we have

$$A(f_{k+1}) \leq A(f_k + h_k s^g g) + \epsilon_k, \quad s^g = \text{sign}(\bar{w}^g - \beta^g_k).$$

Now multiply the above inequality by $|\bar{w}^g - \beta^g_k|$, and sum over $g \in S_k$, we obtain

$$\begin{aligned} &\Delta W_k(A(f_{k+1}) - \epsilon_k) \\ &\leq \sum_{g\in S_k} |\beta^g_k - \bar{w}^g| A(f_k + h_k s^g g) =: B(h_k). \end{aligned} \tag{11}$$

We only need to upper bound $B(h_k)$, which in turn gives an upper bound on $A(f_{k+1})$.

We recall a simple but important property of a convex function that follows directly from the definition of convexity of $A(f)$ as a function of $f$: $\forall f_1, f_2$

$$A(f_2) \geq A(f_1) + \nabla A(f_1)^T(f_2 - f_1). \tag{12}$$

We are now ready to prove the following one-step convergence bound, which is the main result of this section.

**Lemma 4.1** *Assume that $A(f)$ satisfies Assumption 3.1. Consider $h_k$ and $s_k$ that satisfy (9). Let $\bar{f}$ be an arbitrary function in* $\mathrm{span}(S)$, *and define*

$$\Delta A(f_k) = \max(0, A(f_k) - A(\bar{f})) \tag{13}$$

$$\bar{\epsilon}_k = \frac{h_k^2}{2} M(s_{k+1}) + \epsilon_k. \tag{14}$$

*Then after $k$-steps, the following bound holds for $f_{k+1}$ obtained from Algorithm 2.1:*

$$\Delta A(f_{k+1}) \leq \left(1 - \frac{h_k}{s_k + \|\bar{f}\|_1}\right) \Delta A(f_k) + \bar{\epsilon}_k. \tag{15}$$

*Proof Sketch.* From $0 \in \Lambda_k$ and (3), it is easy to see that if $A(f_k) - A(\bar{f}) < 0$, then $A(f_{k+1}) - A(\bar{f}) \leq \bar{\epsilon}_k$, which implies (15). Hence the lemma holds in this case. Therefore in the following, we assume that $A(f_k) - A(\bar{f}) \geq 0$.

Using Taylor expansion, we can bound each term on the right hand side of (11) as:

$$\begin{aligned}
&A(f_k + h_k s^g g) \\
\leq &A(f_k) + h_k s^g \nabla A(f_k)^T g + \frac{h_k^2}{2} \sup_{\xi \in [0,1]} A''_{f_k, g}(\xi h_k s^g) \\
\leq &A(f_k) + h_k s^g \nabla A(f_k)^T g + \frac{h_k^2}{2} M(\|f_k\|_1 + h_k),
\end{aligned}$$

where the second inequality follows from Assumption 3.1. Taking a weighted average, we have:

$$\begin{aligned}
B(h_k) \leq &\sum_{g \in S_k} |\beta_k^g - \bar{w}^g| \left[A(f_k) + \nabla A(f_k)^T h_k s^g g \right. \\
&\left. + \frac{h_k^2}{2} M(\|f_k\|_1 + h_k)\right] \\
= &\Delta W_k A(f_k) + h_k \nabla A(f_k)^T (\bar{f} - f_k) \\
&+ \frac{h_k^2}{2} \Delta W_k M(\|f_k\|_1 + h_k) \\
\leq &\Delta W_k A(f_k) + h_k [A(\bar{f}) - A(f_k)] \\
&+ \frac{h_k^2}{2} \Delta W_k M(\|f_k\|_1 + h_k).
\end{aligned}$$

The last inequality follows from (12). Now using (11) and the bound $\|f_k\|_1 + h_k \leq s_{k+1}$, we obtain

$$\begin{aligned}
&(A(f_{k+1}) - A(\bar{f})) - \epsilon_k \\
\leq &\left(1 - \frac{h_k}{\Delta W_k}\right) (A(f_k) - A(\bar{f})) + \frac{h_k^2}{2} M(s_{k+1}).
\end{aligned}$$

Now replace $\Delta W_k$ by the right hand side of (10) with $\epsilon \to 0$, we obtain the lemma. □

### 4.2. Multi-step analysis

This section provides a convergence bound for the boosting algorithm (2.1) by applying Lemma 4.1 repeatedly. The main result is given by the following lemma.

**Lemma 4.2** *Under the assumptions of Lemma 4.1, we have*

$$\Delta A(f_k) \leq \frac{\|f_0\|_1 + \|\bar{f}\|_1}{s_k + \|\bar{f}\|_1} \Delta A(f_0) + \sum_{j=1}^{k} \frac{s_j + \|\bar{f}\|_1}{s_k + \|\bar{f}\|_1} \bar{\epsilon}_{j-1}. \tag{16}$$

*Proof Sketch.* Note that $\forall a \geq 0$,

$$\begin{aligned}
\prod_{\ell=j}^{k} \left(1 - \frac{h_\ell}{s_\ell + a}\right) \leq &\exp\left(-\int_{s_j}^{s_{k+1}} \frac{1}{v + a} dv\right) \\
= &\frac{s_j + a}{s_{k+1} + a}.
\end{aligned}$$

By recursively applying (15), and using the above inequality, we obtain

$$\begin{aligned}
\Delta A(f_{k+1}) \leq &\prod_{\ell=0}^{k} \left(1 - \frac{h_\ell}{s_\ell + \|\bar{f}\|_1}\right) \Delta A(f_0) \\
&+ \sum_{j=0}^{k} \prod_{\ell=j+1}^{k} \left(1 - \frac{h_\ell}{s_\ell + \|\bar{f}\|_1}\right) \bar{\epsilon}_j \\
\leq &\frac{s_0 + \|\bar{f}\|_1}{s_{k+1} + \|\bar{f}\|_1} \Delta A(f_0) + \sum_{j=0}^{k} \frac{s_{j+1} + \|\bar{f}\|_1}{s_{k+1} + \|\bar{f}\|_1} \bar{\epsilon}_j.
\end{aligned}$$

□

The above result gives a quantitative bound on the convergence of $A(f_k)$ to the value $A(\bar{f})$ of an arbitrary reference function $\bar{f} \in \mathrm{span}(S)$. We can see that the numerical rate or speed of convergence of $A(f_k)$ to $A(\bar{f})$ depends on $\|\bar{f}\|_1$. Specifically, it follows from the above bound that

$$\begin{aligned}
A(f_{k+1}) \leq &A(\bar{f}) \left\{1 - \frac{s_0 + \|\bar{f}\|_1}{s_{k+1} + \|\bar{f}\|_1}\right\} + \frac{s_0 + \|\bar{f}\|_1}{s_{k+1} + \|\bar{f}\|_1} A(f_0) \\
&+ \sum_{j=0}^{k} \frac{s_{j+1} + \|\bar{f}\|_1}{s_{k+1} + \|\bar{f}\|_1} \bar{\epsilon}_j.
\end{aligned} \tag{17}$$

To our knowledge, this is the first convergence bound for greedy boosting procedures with quantitative numerical convergence rate information. Previous analyses, including matching pursuit for least squares, Breiman's analysis of the exponential loss, as well as the analysis of gradient boosting in (Mason et al.,

2000), were all limiting results without any information on the numerical rate of convergence. The key conceptual difference here is that we do not compare to the optimal value directly, but instead, to the value of an arbitrary $\bar{f} \in \text{span}(S)$ so that $\|\bar{f}\|_1$ can be used to measure the convergence rate. This approach is also crucial for problems where $A(\cdot)$ can take $-\infty$ as its infinimum for which a direct comparison will clearly fail (for example, Breiman's exponential loss analysis requires smoothness assumptions to prevent this $-\infty$ infinimum value).

A general limiting convergence theorem follows directly from the above lemma. Due to the space limitation, we skip the proof.

**Theorem 4.1** *Assume that* $\sum_{j=0}^{\infty} \bar{\epsilon}_j < \infty$ *and* $\sum_{j=0}^{\infty} h_j = \infty$, *then we have the following optimization convergence result for the greedy boosting algorithm (2.1):* $\lim_{k\to\infty} A(f_k) = \inf_{f\in\text{span}(S)} A(f)$.

**Corollary 4.1** *For loss functions in Section 3, we have* $\sup_a M(a) < \infty$. *Therefore as long as there exist* $h_j$ *in (9) and* $\epsilon_j$ *in (3) such that* $\sum_{j=0}^{\infty} h_j = \infty$, $\sum_{j=0}^{\infty} h_j^2 < \infty$, *and* $\sum_{j=0}^{\infty} \epsilon_j < \infty$, *we have the following convergence result for the greedy boosting procedure:* $\lim_{k\to\infty} A(f_k) = \inf_{f\in\text{span}(S)} A(f)$.

## 5. Examples of Convergence Analysis

We now illustrate our convergence analysis with a few examples. In the discussion below, we focus on the crucial small step size condition which is implicit in the unrestricted step-size case, but explicit in the restricted step-size case.

### 5.1. Unrestricted step-size

In this case, we let $\Lambda_k = R$ for all $k$ so that the size of $\bar{\alpha}_k$ in the boosting algorithm is unrestricted. For simplicity, we will only consider the case that $\sup_a M(a)$ is upper bounded by a constant $M$.

Interestingly enough, although the size of $\bar{\alpha}_k$ is not restricted in the boosting algorithm itself, for certain formulations the inequality $\sum_j \bar{\alpha}_j^2 < \infty$ still holds. Theorem 4.1 can then be applied to show the convergence of such boosting procedures. For convenience, we will impose the following additional assumption for the step size $\bar{\alpha}_k$ in Algorithm 2.1:

$$A(f_k + \bar{\alpha}_k \bar{g}_k) = \inf_{\alpha_k \in R} A(f_k + \alpha_k \bar{g}_k), \qquad (18)$$

which means that given the selected basis function $\bar{g}_k$, the corresponding $\bar{\alpha}_k$ is chosen to be the exact minimizer. Due to the space limitation, we skip the proof.

**Lemma 5.1** *Assume that* $\bar{\alpha}_k$ *satisfies (18). If* $\exists c > 0$ *such that* $\inf_k \inf_{\xi\in(0,1)} A''_{(1-\xi)f_k+\xi f_{k+1},\, \bar{g}_k}(0) \geq c$, *then* $\sum_{j=0}^{k} \bar{\alpha}_j^2 \leq 2c^{-1}[A(f_0) - A(f_{k+1})]$.

By combining Lemma 5.1 and Corollary 4.1, we obtain

**Corollary 5.1** *Assume that* $\sup_a M(a) < +\infty$ *and* $\epsilon_j$ *in (3) satisfies* $\sum_{j=0}^{\infty} \epsilon_j < \infty$. *Assume also that in Algorithm 2.1, we let* $\Lambda_k = R$ *and let* $\bar{\alpha}_k$ *satisfy (18). If* $\inf_k \inf_{\xi\in(0,1)} A''_{(1-\xi)f_k+\xi f_{k+1},\, \bar{g}_k}(0) > 0$, *then* $\lim_{k\to\infty} A(f_k) = \inf_{f\in\text{span}(S)} A(f)$.

#### Least squares loss

The convergence of unrestricted step-size boosting using the least squares loss (matching pursuit) was studied in (Mallat & Zhang, 1993). Since a scaling of the basis function does not change the algorithm, without loss of generality, we can assume that $E_X g(X)^2 = 1$ for all $g \in S$ (assume $S$ does not contain function 0). In this case, it is easy to check that $\forall g \in S$:

$$A''_{f,g}(0) = E_X g(X)^2 = 1.$$

Therefore the conditions in (5.1) are satisfied as long as $\sum_{j=0}^{\infty} \epsilon_j < \infty$. This shows that the matching pursuit procedure converges, that is,

$$\lim_{k\to\infty} A(f_k) = \inf_{f\in\text{span}(S)} A(f).$$

We would like to point out that for matching pursuit, the inequality in (5.1) can be replaced by the following equality $\sum_{j=0}^{k} \bar{\alpha}_j^2 = 2[A(f_0) - A(f_{k+1})]$, which was referred to as "energy conservation" in (Mallat & Zhang, 1993), and was used there to prove the convergence.

#### Exponential loss

The convergence behavior of boosting with the exponential loss was previously studied by Breiman (Breiman, 2000) for $\pm 1$-trees under the assumption $\inf_x P(Y = 1|x)P(Y = -1|x) > 0$. Using exact computation, Breiman obtained an equality similar to the matching pursuit energy conservation equation. As part of the convergence analysis, the equality was used to show $\sum_{j=0}^{\infty} \bar{\alpha}_j^2 < \infty$.

The following lemma shows that under a more general condition, the convergence of unrestricted boosting with exponential loss follows directly from Corollary 5.1. This result generalizes that of (Breiman, 2000). Due to the space limitation, proof will be skipped.

**Lemma 5.2** *Assume that*

$$\inf_{g\in S} E_X |g(X)| \sqrt{P(Y = 1|X)P(Y = -1|X)} > 0.$$

*If $\bar{\alpha}_k$ satisfies (18), then*

$$\inf_k \inf_{\xi\in(0,1)} A''_{(1-\xi)f_k+\xi f_{k+1},\,\bar{g}_k}(0) > 0.$$

*Hence* $\sum_j \bar{\alpha}_j^2 < \infty$.

### 5.2. Restricted step-size

Although unrestricted step-size boosting procedures can be successful in certain cases, for general problems, they may fail. In such cases, the crucial condition of $\sum_{j=0}^{\infty} \bar{\alpha}_j^2 < \infty$, as required in the proof of Corollary 5.1, can be violated.

Intuitively, the difficulty associated with large $\bar{\alpha}_j$ is due to the potential problem of large oscillation in that a greedy-step may search for a sub-optimal direction, which needs to be corrected later on. If a large step is taken toward the sub-optimal direction, then many more additional steps have to be taken to correct the mistake. If the additional steps are also large, then we may over correct and go to some other sub-optimal directions. In general it becomes difficult to keep track of the overall effect.

The large oscillation problem can be avoid by restricting the step size when we compute $\bar{\alpha}_j$. This idea was advocated by Friedman, who discovered empirically that taking small step size helps (Friedman, 2001). In our analysis, we can restrict the search region so that Corollary 4.1 is automatically satisfied. Since we believe this is an important case which applies for general loss functions, we shall explicitly state the corresponding convergence result below.

**Corollary 5.2** *Consider loss functions in Section 3, where* $\sup_a M(a) < +\infty$. *Pick any sequence of positive numbers* $h_j$ $(j \geq 0)$ *such that* $\sum_{j=0}^{\infty} h_j = \infty$, $\sum_{j=0}^{\infty} h_j^2 < \infty$. *If we choose* $\Lambda_k$ *in Algorithm 2.1 such that* $h_k = \sup \Lambda_k$, *and* $\epsilon_j$ *in (3) such that* $\sum_{j=0}^{\infty} \epsilon_j < \infty$, *then*

$$\lim_{k\to\infty} A(f_k) = \inf_{f\in \mathrm{span}(S)} A(f).$$

Note that the above result requires that the step size $h_j$ to be small $(\sum_{j=0}^{\infty} h_j^2 < \infty)$, but also not too small $(\sum_{j=0}^{\infty} h_j = \infty)$. As discussed above, the first condition prevents large oscillation. The second condition is needed to ensure that $f_k$ can cover the whole space $\mathrm{span}(S)$.

### 5.3. AdaBoost for large margin separable problems

The original idea of Adaboost (boosting with the exponential loss function) is developed under the assumption that the weak learning algorithm can always make reasonable progress at each round. Under some appropriate measurement of progress, it was shown in (Freund & Schapire, 1997) that the expected classification error decreases exponentially. The result was later extended in (Schapire et al., 1998) using the concept of margin. In this section, we go beyond the limiting convergence results in the last section and use the bound given by Lemma 4.2 to provide a numerical convergence rate for AdaBoost under a large margin separable condition to be stated below.

Given a real-valued classification function $p(x)$, we consider the following discrete prediction rule:

$$y = \begin{cases} 1 & \text{if } p(x) \geq 0, \\ -1 & \text{if } p(x) < 0. \end{cases} \tag{19}$$

Its classification error (for simplicity, we ignore the point $p(x) = 0$, which is assumed to occur rarely) is given by

$$L_\gamma(p(x), y) = \begin{cases} 1 & \text{if } p(x)y \leq \gamma, \\ 0 & \text{if } p(x)y > \gamma \end{cases}$$

with $\gamma = 0$. In general, we may consider $\gamma \geq 0$ and the parameter $\gamma \geq 0$ is often referred to as margin, and we shall call the corresponding error function $L_\gamma$ margin error.

In (Schapire et al., 1998) the authors proved that under appropriate assumptions on the base learner, the expected margin error $L_\gamma$ with a positive margin $\gamma > 0$ also decreases exponentially. It follows that regularity assumptions of weak learning for Adaboost imply the following margin condition: $\exists \gamma > 0$ such that $\inf_{f\in\mathrm{span}(S),\|f\|_1=1} L_\gamma(f, y) = 0$, which in turn implies the following inequality: $\forall s > 0$,

$$\inf_{f\in\mathrm{span}(S),\|f\|_1=1} E_{X,Y} \exp(-sf(X)Y) \leq \exp(-\gamma s). \tag{20}$$

We now show that under (20), the expected margin errors (with small margin) from Algorithm 2.1 may decrease exponentially.

$$f_0 = 0, \quad \sup \Lambda_k \leq h_k, \quad \epsilon_k \leq h_k^2/2.$$

Note that this implies that $\bar{\epsilon}_k \leq h_k^2$ for all $k$.

Now applying (17) with $\bar{f} = sf$ for any $s > 0$ and let $f$ approach the minimum in (20), we obtain (recall $\|f\|_1 = 1$)

$$A(f_k) \leq -s\gamma \frac{s_k}{s_k + s} + \sum_{j=1}^{k} \frac{s_j + s}{s_k + s} \bar{\epsilon}_{j-1} \leq -s\gamma \frac{s_k}{s_k + s} + \sum_{j=0}^{k-1} h_j^2.$$

Now let $s \to \infty$, we have

$$A(f_k) \leq -\gamma s_k + \sum_{j=0}^{k-1} h_j^2.$$

Assume we pick a constant $h < \gamma$ and let $h_k = h$, then

$$E_{X,Y} \exp(-f_k(X)Y) \leq \exp(-kh(\gamma - h)),$$

which implies that the margin error decreases exponentially for all margin less than $\gamma - h$. We shall point out that this requires a prior knowledge of $\gamma$. If we don't know $\gamma$, then we can let the step size $h_k$ decrease sufficiently slowly so that asymptotically the margin error decreases slightly slower than exponential.

## 6. Conclusion

In this paper, we studied a general version of boosting procedure given in Algorithm 2.1. The convergence behavior of this algorithm is studied using the so-called averaging technique, which were previously used to analyze greedy algorithms for optimization problems defined in the convex hull of a set of basis functions. Specifically, this technique is applied to problems defined on the whole linear space spanned by the basis functions. We derived an estimate of the numerical convergence rate and established conditions that ensures the convergence of Algorithm 2.1. Our results generalize those in previous studies such as the matching pursuit analysis in (Mallat & Zhang, 1993) and the convergence analysis of Adaboost by Breiman (Breiman, 2000). Furthermore, our analysis shows the importance of using small-step size in boosting procedures, which provides theoretical insights into Friedman's empirical observation (Friedman, 2001).

## References


Breiman, L. (1998). Arcing classifiers. *The Annals of Statistics, 26*, 801–849. with discussion.

Breiman, L. (1999). Prediction games and arcing algorithms. *Neural Computation, 11*, 1493–1517.

Breiman, L. (2000). *Some infinity theory for predictor ensembles* (Technical Report 577). Statistics Department, University of California, Berkeley.

Bühlmann, P., & Yu, B. (2003). Boosting with the $L_2$ loss: Regression and classification. *Journal of American Statistical Association.* to appear.

Collins, M., Schapire, R. E., & Singer, Y. (2002). Logistic regression, adaboost and bregman distances. *Machine Learning, 48*, 253–285.

Freund, Y., & Schapire, R. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci., 55*, 119–139.

Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting. *The Annals of Statistics, 28*, 337–407. With discussion.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Ann. Statist., 29*, 1189–1232.

Jones, L. (1992). A simple lemma on greedy approximation in Hilbert space and convergence rates for projection pursuit regression and neural network training. *Ann. Statist., 20*, 608–613.

Lee, W., Bartlett, P., & Williamson, R. (1996). Efficient agnostic learning of neural networks with bounded fan-in. *IEEE Transactions on Information Theory, 42*, 2118–2132.

Mallat, S., & Zhang, Z. (1993). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing, 41*, 3397–3415.

Mason, L., Bartlett, P., Baxter, J., & Frean, M. (2000). Functional gradient techniques for combining hypotheses. In B. S. A. Smola and D. Schuurmans (Eds.), *Advances in large margin classifiers.* MIT Press.

Rätsch, G., Mika, S., & Warmuth, M. (2001). *On the convergence of leveraging*NeuroCOLT2 Technical Report NC-TR-01-098). Royal Holloway College, London. A short version appeared in NIPS 14, MIT press, 2002.

Schapire, R., Freund, Y., Bartlett, P., & Lee, W. (1998). Boosting the margin: a new explanation for the effectiveness of voting methods. *Ann. Statist., 26*, 1651–1686.

Schapire, R. E., & Singer, Y. (1999). Improved boosting algorithms using confidence-rated predictions. *Machine Learning, 37*, 297–336.

Zhang, T. (2003a). Sequential greedy approximation for certain convex optimization problems. *IEEE Transaction on Information Theory, 49*, 682–691.

Zhang, T. (2003b). Statistical behavior and consistency of classification methods based on convex risk minimization. *The Annals of Statitics.* to appear.