Optimization by Gradient Boosting

Gérard Biau and Benoît Cadre

Abstract Gradient boosting is a state-of-the-art prediction technique that sequentially produces a model in the form of linear combinations of elementary predictors—typically decision trees—by solving an infinite-dimensional convex optimization problem. We provide in the present paper a thorough analysis of two widespread versions of gradient boosting, and introduce a general framework for studying these algorithms from the point of view of functional optimization. We prove their convergence as the number of iterations tends to infinity and highlight the importance of having a strongly convex risk functional to minimize. We also present a reasonable statistical context ensuring consistency properties of the boosting predictors as the sample size grows. In our approach, the optimization procedures are run forever (that is, without resorting to an early stopping strategy), and statistical regularization is basically achieved via an appropriate L^2 penalization of the loss and strong convexity arguments.

1 Introduction

More than twenty years after the pioneering articles of Freund and Schapire [Schapire, 1990, Freund, 1995, Freund and Schapire, 1996, 1997], boosting is still one of the most powerful ideas introduced in statistics and machine learning. Freund and Schapire's AdaBoost algorithm and its numerous descendants have proven to be competitive in a variety of applications, and are still able to provide state-of-the-art decisions in difficult real-life problems. In addition, boosting procedures

Gérard Biau
Sorbonne Université, CNRS, LPSM, 4 place Jussieu, 75005 Paris, France e-mail: gerard.biau@sorbonne-universite.fr

Benoît Cadre
Univ Rennes, CNRS, IRMAR - UMR 6625, F-35000 Rennes, France e-mail: benoit.cadre@univ-rennes2.fr

are computationally fast and comfortable with both regression and classification problems. For surveys of various aspects of boosting algorithms, we refer to Meir and Rätsch [2003], Bühlmann and Hothorn [2007], and to the monographs by Hastie et al. [2009] and Bühlmann and van de Geer [2011]. These references point in particular to approaches related to boosting, for example Frank and Wolfe [1956] algorithm, Mallat and Zhang [1993] matching pursuit algorithm, and weak greedy algorithms of Temlyakov [2000].

In a nutshell, the basic idea of boosting is to combine the outputs of many "simple" predictors, in order to produce a powerful committee with performances improved over the single members. Historically, the first formulations of Freund and Schapire considered boosting as an iterative classification algorithm that is run for a fixed number of iterations, and, at each iteration, selects one of the base classifiers, assigns a weight to it, and outputs the weighted majority vote of the chosen classifiers. Later on, Breiman [1997, 1998, 1999, 2000, 2004] made in a series of papers and technical reports the breakthrough observation that AdaBoost is in fact a gradient-descent-type algorithm in a function space, thereby identifying boosting at the frontier of numerical optimization and statistical estimation. This connection was further emphasized by Friedman et al. [2000], who rederived AdaBoost as a method for fitting an additive model in a forward stagewise manner. Following this, Friedman [2001, 2002] developed a general statistical framework (both for regression and classification) that (i) yields a direct interpretation of boosting methods from the perspective of numerical optimization in a function space, and (ii) generalizes them by allowing optimization of an arbitrary loss function. The term "gradient boosting" was coined by the author, who paid a special attention to the case where the individual additive components are decision trees. At the same time, Mason et al. [1999, 2000] embraced a more mathematical approach, revealing boosting as a principle to optimize a convex risk in a function space, by iteratively choosing a weak learner that approximately points in the negative gradient direction.

This functional view of boosting has led to the development of algorithms in many areas of machine learning and computational statistics, beyond regression and classification. The history of boosting goes on today with algorithms such as XGBoost [Extreme Gradient Boosting, Chen and Guestrin, 2016], a tree boosting system widely recognized for its outstanding results in numerous data challenges. [An overview of its successes is given in the introductive section of the paper by Chen and Guestrin, 2016.] From a general point of view, XGBoost is but a scalable implementation of gradient boosting that contains various systems and algorithmic optimizations. Its mathematical principle is to perform a functional gradient-type descent in a space of decision trees, while regularizing the objective to avoid overfitting.

However, despite a long list of successes, much work remains to be done to clarify the mathematical forces driving gradient boosting algorithms. Many influential articles regard boosting with a statistical eye and study the somewhat idealized problem of empirical risk minimization with a convex loss [e.g., Blanchard et al., 2003, Lugosi and Vayatis, 2004]. These papers essentially concentrate on the statistical properties of the approach (that is, consistency and rates of convergence as the sample size grows) and often ignore the underlying optimization aspects. Other important articles, such as Bühlmann and Yu [2003], Mannor et al. [2003], Zhang and Yu [2005], Bickel et al. [2006], Bartlett and Traskin [2007] take advantage of the iterative principle of boosting, but mainly focus on regularization via early stopping (that is, stopping the boosting iterations at some point), without paying too much attention to the optimization side. Notable exceptions are the pioneering notes of Breiman cited above, and the original paper by Mason et al. [2000], who envision gradient boosting as an infinite-dimensional numerical optimization problem and pave the way for a more abstract analysis. All in all, there is to date no sound theory of gradient boosting in terms of numerical optimization. This state of affairs is a bit paradoxical, since optimization is certainly the most natural mathematical environment for gradient-descent-type algorithms.

In line with the above, our main objective in this article is to provide a thorough analysis of two widespread models of gradient boosting, due to Friedman [2001] and Mason et al. [2000]. We introduce in Section 2 a general framework for studying the algorithms from the point of view of functional optimization in an L^2 space, and prove in Section 3 their convergence as the number of iterations tends to infinity. Our results allow for a large choice of convex losses in the optimization problem (differentiable or not), while highlighting the importance of having a strongly convex risk functional to minimize. This point is interesting, since it provides some theoretical justification for adding a penalty term to the objective, as advocated for example in the XGBoost system of Chen and Guestrin [2016]. Thus, the main message of Section 3 is that, under appropriate conditions, the sequence of boosted iterates converges towards the minimizer of the empirical risk functional over the set of linear combinations of weak learners. However, this does not guarantee that the output of the algorithms (i.e., the boosting predictor) enjoys good statistical properties, as overfitting may kick in. For this reason, we present in Section 4 a reasonable framework ensuring consistency properties of the boosting predictors as the sample size grows. In our approach, the optimization procedures are run forever (that is, without resorting to an early stopping strategy), and statistical regularization is basically achieved via an appropriate L^2 penalization of the loss and strong convexity arguments. For clarity, most proofs are gathered in the Supplementary Material Document.

Before embarking on the analysis, we would like to stress that the present paper is theoretical in nature and that its main goal is to clarify/solidify some of the optimization ideas that are behind gradient boosting. In particular, we do not report experimental results, and refer to the specialized literature on (extreme) gradient boosting for discussions on the computational aspects and experiments with real-world data.

2 Gradient boosting

The purpose of this section is to describe the gradient boosting procedures that we analyze in the paper.

2.1 Mathematical context

We assume to be given a sample $\mathscr{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ of i.i.d. observations, where each pair (X_i, Y_i) takes values in $\mathscr{X} \times \mathscr{Y}$. Throughout, $\mathscr{X}$ is a Borel subset of $\mathbb{R}^d$, and $\mathscr{Y} \subset \mathbb{R}$ is either a finite set of labels (for classification) or a subset of $\mathbb{R}$ (for regression). The vector space $\mathbb{R}^d$ is endowed with the Euclidean norm $\|\cdot\|$.

Our goal is to construct a predictor $F : \mathscr{X} \to \mathbb{R}$ that assigns a response to each possible value of an independent random observation distributed as X_1. In the context of gradient boosting, this general problem is addressed by considering a class $\mathscr{F}$ of functions $f : \mathscr{X} \to \mathbb{R}$ (called the weak or base learners) and minimizing some empirical risk functional

$$C_n(F) = \frac{1}{n} \sum_{i=1}^{n} \psi(F(X_i), Y_i)$$

over the linear combinations of functions in $\mathscr{F}$. The function $\psi : \mathbb{R} \times \mathscr{Y} \to \mathbb{R}_+$, called the loss, is convex in its first argument and measures the cost incurred by predicting $F(X_i)$ when the answer is Y_i. For example, in the least squares regression problem, $\psi(x, y) = (y - x)^2$ and

$$C_n(F) = \frac{1}{n} \sum_{i=1}^{n} (Y_i - F(X_i))^2.$$

However, many other examples are possible, as we will see below. Let δ_z denote the Dirac measure at z, and let $\mu_n = (1/n) \sum_{i=1}^{n} \delta_{(X_i, Y_i)}$ be the empirical measure associated with the sample $\mathscr{D}_n$. Clearly,

$$C_n(F) = \mathbb{E}\psi(F(X), Y),$$

where (X, Y) denotes a random pair with distribution μ_n and the symbol $\mathbb{E}$ denotes the expectation with respect to μ_n. Naturally, the theoretical (i.e., population) version of C_n is

$$C(F) = \mathbb{E}\psi(F(X_1), Y_1),$$

where now the expectation is taken with respect to the distribution of (X_1, Y_1). It turns out that most of our subsequent developments are independent of the context, whether empirical or theoretical. Therefore, to unify the notation, we let throughout (X, Y) be a generic pair of random variables with distribution $\mu_{X,Y}$, keeping in mind that $\mu_{X,Y}$ may be the distribution of (X_1, Y_1) (theoretical risk), the standard empirical measure μ_n (empirical risk), or any smoothed version of μ_n.

We let μ_X be the distribution of X, $L^2(\mu_X)$ the vector space of all measurable functions $f : \mathscr{X} \to \mathbb{R}$ such that $\int |f|^2 \mathrm{d}\mu_X < \infty$, and denote by $\langle \cdot, \cdot \rangle_{\mu_X}$ and $\|\cdot\|_{\mu_X}$ the corresponding norm and scalar product. Thus, for now, our problem is to minimize the quantity

$$C(F) = \mathbb{E}\psi(F(X), Y)$$

over the linear combinations of functions in a given subset $\mathscr{F}$ of $L^2(\mu_X)$. A typical example for $\mathscr{F}$ is the collection of all binary decision trees in $\mathbb{R}^d$ using axis parallel cuts with k terminal nodes. In this case, each $f \in \mathscr{F}$ takes the form $f = \sum_{j=1}^k \beta_j \mathbb{1}_{A_j}$, where $(\beta_1, \dots, \beta_k) \in \mathbb{R}^k$ and $A_1, \dots, A_k$ is a tree-structured partition of $\mathbb{R}^d$ [Devroye et al., 1996, Chapter 20].

As noted earlier, we assume that, for each $y \in \mathscr{Y}$, the function $\psi(\cdot, y)$ is convex. In the framework we have in mind, the function ψ may take a variety of different forms, ranging from standard (regression or classification) losses to more involved penalized objectives. It may also be differentiable or not. Before discussing some examples in detail, we list assumptions that will be needed at some point. Throughout, we let $\xi(\cdot, y) = \partial_x^- \psi(\cdot, y)$ (left derivative) be a subgradient of the convex function $\psi(\cdot, y)$ (the choice of a specific subgradient $\xi(\cdot, y)$ has no impact on the results). In particular, for all $(x_1, x_2) \in \mathbb{R}^2$,

$$\psi(x_1, y) \geq \psi(x_2, y) + \xi(x_2, y)(x_1 - x_2). \tag{1}$$

Assumption $\mathbf{A_1}$

$\mathbf{A_1}$ One has $\mathbb{E}\psi(0, Y) < \infty$. In addition, for all $F \in L^2(\mu_X)$, there exists $\delta > 0$ such that

$$\sup_{G \in L^2(\mu_X) : \|G - F\|_{\mu_X} \leq \delta} \mathbb{E}|\partial_x^- \psi(G(X), Y)|^2 < \infty.$$

This assumption ensures that the convex functional C is locally bounded (in particular, $C(F) < \infty$ for all $F \in L^2(\mu_X)$, and C is continuous). Indeed, by inequality (1), for all $G \in L^2(\mu_X)$,

$$\psi(G(x), y) \leq \psi(0, y) + \xi(G(x), y)G(x).$$

Therefore, by Assumption $\mathbf{A_1}$ and the Cauchy-Schwarz inequality,

$$\mathbb{E}\psi(G(X), Y) \leq \mathbb{E}\psi(0, Y) + \big(\mathbb{E}\xi(G(X), Y)^2 \mathbb{E}G(X)^2\big)^{1/2},$$

so that C is locally bounded. Naturally, Assumption $\mathbf{A_1}$ is automatically satisfied for the choice $\mu_{X,Y} = \mu_n$.

Assumption $\mathbf{A_2}$

$\mathbf{A_2}$ There exists $\alpha > 0$ such that, for all $y \in \mathscr{Y}$, the function $\psi(\cdot, y)$ is α-strongly convex, i.e., for all $(x_1, x_2) \in \mathbb{R}^2$ and $t \in [0, 1]$,

$$\psi(tx_1 + (1-t)x_2, y) \leq t\psi(x_1, y) + (1-t)\psi(x_2, y) - \frac{\alpha}{2}t(1-t)(x_1 - x_2)^2.$$

This assumption will be used in most, but not all, results. Strong convexity will play an essential role in the statistical Section 4. We note that, under Assumption $\mathbf{A_2}$, for all $(x_1, x_2) \in \mathbb{R}^2$,

$$\psi(x_1, y) \geq \psi(x_2, y) + \xi(x_2, y)(x_1 - x_2) + \frac{\alpha}{2}(x_1 - x_2)^2, \tag{2}$$

which is of course an inequality tighter than (1). Furthermore, the α-strong convexity of $\psi(\cdot, y)$ implies the α-strong convexity of the risk functional C over $L^2(\mu_X)$.

In addition to Assumptions $\mathbf{A_1}$ and $\mathbf{A_2}$, we require the following:

$\mathbf{A_3}$ There exists a positive constant L such that, almost surely, for all $(x_1, x_2) \in \mathbb{R}^2$,

$$|\mathbb{E}(\xi(x_1, Y) - \xi(x_2, Y) \mid X)| \leq L|x_1 - x_2|.$$

(In the sequel, in order to lighten the text, we drop the "almost sure" wording wherever conditional expectations are involved.) However esoteric this assumption may seem, it is in fact mild and provide a framework that encompasses a large variety of familiar situations. In particular, Assumption $\mathbf{A_3}$ admits a stronger version $\mathbf{A_3'}$, which is useful as soon as the function ψ is continuously differentiable with respect to its first variable:

$\mathbf{A_3'}$ For all $y \in \mathscr{Y}$, the function $\psi(\cdot, y)$ is continuously differentiable, and there exists a positive constant L such that, for all $(x_1, x_2, y) \in \mathbb{R}^2 \times \mathscr{Y}$,

$$|\partial_x \psi(x_1, y) - \partial_x \psi(x_2, y)| \leq L|x_1 - x_2|.$$

Assumption $\mathbf{A_3'}$ implies $\mathbf{A_3}$, but the converse is not true. To see this, just note that, in the smooth situation $\mathbf{A_3'}$, we have $\xi(x, y) = \partial_x \psi(x, y)$. Therefore,

$$\mathbb{E}(\xi(x_1, Y) \mid X) = \int \partial_x \psi(x_1, Y) \mu_{Y|X}(\mathrm{d}y),$$

where $\mu_{Y|X}$ is the conditional distribution of Y given X. Assumption $\mathbf{A_3}$ (or $\mathbf{A_3'}$) plays a key role in controlling the decrease of the risk functional along the boosting iterations, as can be seen very clearly in Lemma 1 and Lemma 2. This type of Lipschitz hypotheses is classical in the optimization literature [e.g., Bubeck, 2015]. We also note that, in the context of $\mathbf{A_3'}$, the functional C is differentiable at any $F \in L^2(\mu_X)$ in the direction $G \in L^2(\mu_X)$, with differential

$$dC(F; G) = \langle \nabla C(F), G \rangle_{\mu_X},$$

where $\nabla C(F)(x) := \int \partial_x \psi(F(x), y) \mu_{Y|X=x}(\mathrm{d}y)$ is the gradient of C at F. However, Assumption $\mathbf{A_3}$ allows to deal with a larger variety of losses, including non-differentiable ones, as shown in the examples below.

2.2 Some examples

Each of the loss functions that we discuss in this subsection corresponds to a machine learning algorithm, as thoroughly explained in Bühlmann and Hothorn [2007, Section 3]. We refer to this article for more properties of these losses and for issues regarding their practical implementation.

- A first canonical example, in the regression setting, is to let $\psi(x,y) = (y-x)^2$ (squared error loss), which is 2-strongly convex in its first argument (Assumption $\mathbf{A_2}$) and satisfies Assumption $\mathbf{A_1}$ as soon as $\mathbb{E}Y^2 < \infty$. It also satisfies $\mathbf{A_3'}$, with $\partial_x \psi(x,y) = 2(x-y)$ and $L = 2$.
- Another example in regression is the loss $\psi(x,y) = |y-x|$ (absolute error loss), which is convex but not strongly convex in its first argument. Whenever strong convexity of the loss is required, a possible strategy is to regularize the objective via an L^2-type penalty, and take

  $$\psi(x,y) = |y-x| + \gamma x^2,$$

  where γ is a positive parameter (possibly function of the sample size n in the empirical setting). This loss is (2γ)-strongly convex in x and satisfies $\mathbf{A_1}$ and $\mathbf{A_2}$ whenever $\mathbb{E}|Y| < \infty$, with $\xi(x,y) = \operatorname{sgn}(x-y) + 2\gamma x$ (with $\operatorname{sgn}(u) = 2\mathbb{1}_{[u>0]} - 1$ for $u \neq 0$ and $\operatorname{sgn}(0) = 0$). On the other hand, the function $\psi(\cdot, y)$ is not differentiable at y, so that the smoothness Assumption $\mathbf{A_3'}$ is not satisfied. However,

  $$\begin{aligned}\mathbb{E}(\xi(x_1,Y) - \xi(x_2,Y) \mid X) &= \int (\operatorname{sgn}(x_1-y) - \operatorname{sgn}(x_2-y))\mu_{Y|X}(\mathrm{d}y) + 2\gamma(x_1-x_2)\\ &= \mu_{Y|X}((-\infty,x_1)) - \mu_{Y|X}((-\infty,x_2)) + 2\gamma(x_1-x_2)\\ &\quad - \mu_{Y|X}((x_1,\infty)) + \mu_{Y|X}((x_2,\infty)).\end{aligned}$$

  Thus, if we assume for example that $\mu_{Y|X}$ has a density (with respect to the Lebesgue measure) bounded by B, then

  $$|\mathbb{E}(\xi(x_1,Y) - \xi(x_2,Y) \mid X)| \leq 2(B+\gamma)|x_1 - x_2|,$$

  and Assumption $\mathbf{A_3}$ is therefore satisfied. Of course, in the empirical setting, assuming that $\mu_{Y|X}$ has a density precludes the use of the empirical measure μ_n for $\mu_{X,Y}$. A safe and simple alternative is to consider a smoothed version $\tilde{\mu}_n$ of μ_n [based, for example, on a kernel estimate; see Devroye and Györfi, 1985], and to minimize the functional

  $$C_n(F) = \int |y - F(x)|\tilde{\mu}_n(\mathrm{d}x,\mathrm{d}y) + \gamma \int F(x)^2 \tilde{\mu}_n(\mathrm{d}x)$$

  over the linear combinations of functions in $\mathscr{F}$.
- In the ±1-classification problem, the final classification rule is $+1$ if $F(x) > 0$ and -1 otherwise. Often, the function $\psi(x,y)$ has the form $\phi(yx)$, where $\phi : \mathbb{R} \to \mathbb{R}_+$ is convex. Classical losses include the choices $\phi(u) = \ln_2(1+e^{-u})$ (logit loss), $\phi(u) = e^{-u}$ (exponential loss), and $\phi(u) = \max(1-u, 0)$ (hinge loss). None of these losses is strongly convex, but here again, this can be repaired whenever needed by regularizing the problem via

  $$\psi(x,y) = \phi(yx) + \gamma x^2, \tag{3}$$

where $\gamma > 0$. It is for example easy to see that $\psi(x, y) = \ln_2(1 + e^{-yx}) + \gamma x^2$ satisfies Assumptions $\mathbf{A_1}$, $\mathbf{A_2}$, and $\mathbf{A_3'}$. This is also true for the penalized sigmoid loss $\psi(x, y) = (1 - \tanh(\beta yx)) + \gamma x^2$, where β is a positive parameter. In this case, $\psi(\cdot, y)$ is $2(\gamma - \beta^2)$-strongly convex as soon as $\beta < \sqrt{\gamma}$. Another interesting example in the classification setting is the loss $\psi(x, y) = \phi(yx) + \gamma x^2$, where

$$\phi(u) = \begin{cases} -u + 1 & \text{if } u \leq 0 \\ e^{-u} & \text{if } u > 0. \end{cases}$$

We leave it as an easy exercise to prove that Assumptions $\mathbf{A_1}$, $\mathbf{A_2}$, and $\mathbf{A_3'}$ are satisfied. Examples could be multiplied endlessly, but the point we wish to make is that our assumptions are mild and allow to consider a large variety of learning problems. We also emphasize that regularized objectives of the form (3) are typically in action in the Extreme Gradient Boosting system of Chen and Guestrin [2016].

2.3 Two algorithms

Let $\text{lin}(\mathscr{F})$ be the set of all linear combinations of functions in $\mathscr{F}$, our collection of base predictors in $L^2(\mu_X)$. So, each $F \in \text{lin}(\mathscr{F})$ has the form $F = \sum_{j=1}^J \beta_j f_j$, where $(\beta_1, \ldots, \beta_J) \in \mathbb{R}^J$ and $f_1, \ldots, f_J$ are elements of $\mathscr{F}$. Finding the infimum of the functional C over $\text{lin}(\mathscr{F})$ is a challenging infinite-dimensional optimization problem, which requires an algorithm. The core idea of the gradient boosting approach is to greedily locate the infimum by producing a combination of base predictors via a gradient-descent-type algorithm in $L^2(\mu_X)$. Focusing on the basics, this can be achieved by two related yet different strategies, which we examine in greater mathematical details below. Algorithm 1 appears in Mason et al. [2000], whereas Algorithm 2 is essentially due to Friedman [2001].

It is implicitly assumed throughout this paragraph that Assumption $\mathbf{A_1}$ is satisfied. We recall that under this assumption, the convex functional C is locally bounded and therefore continuous. Thus, in particular,

$$\inf_{F \in \text{lin}(\mathscr{F})} C(F) = \inf_{F \in \overline{\text{lin}(\mathscr{F})}} C(F),$$

where $\overline{\text{lin}(\mathscr{F})}$ is the closure of $\text{lin}(\mathscr{F})$ in $L^2(\mu_X)$. Loosely speaking, looking for the infimum of C over $\overline{\text{lin}(\mathscr{F})}$ is the same as looking for the infimum of C over all (finite) linear combinations of base functions in $\mathscr{F}$. We note in addition that if Assumption $\mathbf{A_2}$ is satisfied, then there exists a unique function $\bar{F} \in \overline{\text{lin}(\mathscr{F})}$ (which we call the boosting predictor) such that

$$C(\bar{F}) = \inf_{F \in \overline{\text{lin}(\mathscr{F})}} C(F). \tag{4}$$

Algorithm 1. In this approach, we consider a class $\mathscr{F}$ of functions $f : \mathscr{X} \to \mathbb{R}$ such that $0 \in \mathscr{F}$, $f \in \mathscr{F} \Leftrightarrow -f \in \mathscr{F}$, and $\|f\|_{\mu_X} = 1$ for $f \neq 0$. An example is the collection $\mathscr{F}$ of all ± 1-binary trees in $\mathbb{R}^d$ using axis parallel cuts with k terminal nodes (plus zero). Each nonzero $f \in \mathscr{F}$ takes the form $f = \sum_{j=1}^{k} \beta_j \mathbb{1}_{A_j}$, where $|\beta_j| = 1$ and $A_1, \ldots, A_k$ is a tree-structured partition of $\mathbb{R}^d$ [Devroye et al., 1996, Chapter 20]. The parameter k is a measure of the tree complexity. For example, trees with $k = d + 1$ are such that $\overline{\mathrm{lin}(\mathscr{F})} = L^2(\mu_X)$ [Breiman, 2000]. Thus, in this case,

$$\inf_{F \in \mathrm{lin}(\mathscr{F})} C(F) = \inf_{F \in L^2(\mu_X)} C(F).$$

Although interesting from the point of view of numerical optimization, this situation is however of little interest for statistical learning, as we will see in Section 4.

Suppose now that we have a function $F \in \mathrm{lin}(\mathscr{F})$ and wish to find a new $f \in \mathscr{F}$ to add to F so that the risk $C(F + wf)$ decreases at most, for some small value of w. Viewed in function space terms, we are looking for the direction $f \in \mathscr{F}$ such that $C(F + wf)$ most rapidly decreases. Assume for the moment, to simplify, that ψ is continuously differentiable in its first argument. Then the knee-jerk reaction is to take the opposite of the gradient of C at F, but since we are restricted to choosing our new function in $\mathscr{F}$, this will in general not be a possible choice. Thus, instead, we start from the approximate identity

$$C(F) - C(F + wf) \approx -w\langle \nabla C(F), f \rangle_{\mu_X} \tag{5}$$

and choose $f \in \mathscr{F}$ that maximizes $-\langle \nabla C(F), f \rangle_{\mu_X}$. For an arbitrary (i.e., not necessarily differentiable) ψ, we simply replace the gradient by a subgradient and choose $f \in \mathscr{F}$ that maximizes $-\mathbb{E}\xi(F(X), Y) f(X)$. This motivates the following iterative algorithm:

Gradient Boosting Algorithm 1

```
1: Require $(w_t)_t$ a sequence of positive real numbers.
2: Set $t = 0$ and start with $F_0 \in \mathscr{F}$.
3: Compute
       $f_{t+1} \in \arg\max_{f \in \mathscr{F}} -\mathbb{E}\xi(F_t(X), Y) f(X)$        (6)
   and let $F_{t+1} = F_t + w_{t+1} f_{t+1}$.
4: Take $t \leftarrow t + 1$ and go to step 3.
```

(Throughout the article, it is assumed to simplify that maximizers as in (6) exist. This requirement can be avoided, for example, by working with approximate ε_t-maximizers, as long as the quality of the approximation ε_t is controlled. This essentially adds technical terms to the equations, without adding much to the general picture.) We emphasize that the method performs a gradient-type descent in the function space $L^2(\mu_X)$. At each iteration, it chooses a base predictor to include in the combination. This predictor is chosen so as to maximally reduce the value of the risk functional. However, the main difference with a standard gradient descent is that Algorithm 1 forces the descent direction to belong to $\mathscr{F}$. To understand the rationale behind this principle, assume that ψ is continuously differentiable in its first argument. As we have seen earlier, in this case,

$$-\mathbb{E}\xi(F_t(X),Y)f(X) = -\langle \nabla C(F_t), f\rangle_{\mu_X},$$

and, for $\nabla C(F_t) \neq 0$,

$$\frac{-\nabla C(F_t)}{\|\nabla C(F_t)\|_{\mu_X}} = \arg\max_{F\in L^2(\mu_X):\|F\|_{\mu_X}=1} - \langle \nabla C(F_t), F\rangle_{\mu_X}.$$

Thus, at each step, Algorithm 1 mimics the computation of the negative gradient by restricting the search of the supremum to the class $\mathscr{F}$, i.e., by taking

$$f_{t+1} \in \arg\max_{f\in\mathscr{F}} - \langle \nabla C(F_t), f\rangle_{\mu_X},$$

which is exactly (6). In the empirical case (i.e., $\mu_{X,Y} = \mu_n$) this descent step takes the form

$$f_{t+1} \in \arg\max_{f\in\mathscr{F}} - \frac{1}{n}\sum_{i=1}^{n} \nabla C(F_t)(X_i)\cdot f(X_i).$$

Finding this optimum is a non-trivial computational problem, which necessitates a strategy. For example, in the spirit of the CART algorithm of Breiman et al. [1984], Chen and Guestrin [2016] use in the XGBoost package a greedy approach that starts from a single leaf and iteratively adds branches to the tree.

The sequence $(w_t)_t$ is the sequence of step sizes, which are allowed to change at every iteration and should be carefully chosen for convergence guarantees. It is also stressed that the algorithm is assumed to be run forever, i.e., stopping or not the iterations is not an issue at this stage of the analysis. As we will see in the next section, the algorithm is convergent under our assumptions (with an appropriate choice of the sequence $(w_t)_t$), in the sense that

$$\lim_{t\to\infty} C(F_t) = \inf_{F\in\text{lin}(\mathscr{F})} C(F).$$

Of course, in the empirical case, the statistical properties as $n \to \infty$ of the limit deserve a special treatment, connected with possible overfitting issues. This important discussion is postponed to Section 4.

Algorithm 2. The principle we used so far rests upon the simple Taylor-like identity (5), which encourages us to imitate the definition of the negative gradient in the class $\mathscr{F}$. Still starting from (5), there is however another strategy, maybe more natural, which consists in choosing f_{t+1} by a least squares approximation of $-\xi(F_t(X),Y)$. To follow this route, we modify a bit the collection of weak learners, and consider a class $\mathscr{P} \subset L^2(\mu_X)$ of functions $f : \mathscr{X} \to \mathbb{R}$ such that $f \in \mathscr{P} \Leftrightarrow -f \in \mathscr{P}$, and $af \in \mathscr{P}$ for all $(a,f) \in \mathbb{R}\times\mathscr{P}$ (in particular, $0 \in \mathscr{P}$, which is thus a cone of $L^2(\mu_X)$). Binary trees in $\mathbb{R}^d$ using axis parallel cuts with k terminal nodes are a good

example of a possible class $\mathscr{P}$. These base learners are of the form $f = \sum_{j=1}^{k} \beta_j \mathbb{1}_{A_j}$, where this time $(\beta_1, \ldots, \beta_k) \in \mathbb{R}^k$, without any normative constraint.

Given F_t, the idea of Algorithm 2 is to choose $f_{t+1} \in \mathscr{P}$ that minimizes the squared norm between $-\xi(F_t(X), Y)$ and $f_{t+1}(X)$, i.e., to let

$$f_{t+1} \in \arg\min_{f \in \mathscr{P}} \mathbb{E}(-\xi(F_t(X), Y) - f(X))^2,$$

or, equivalently,

$$f_{t+1} \in \arg\min_{f \in \mathscr{P}} \big(2\mathbb{E}\xi(F_t(X), Y) f(X) + \|f\|^2_{\mu_X}\big).$$

A more algorithmic format is shown below.

Gradient Boosting Algorithm 2

```
1: Require $\nu$ a positive real number.
2: Set $t = 0$ and start with $F_0 \in \mathscr{P}$.
3: Compute
       $f_{t+1} \in \arg\min_{f \in \mathscr{P}} \big(2\mathbb{E}\xi(F_t(X), Y) f(X) + \|f\|^2_{\mu_X}\big)$    (7)
   and let $F_{t+1} = F_t + \nu f_{t+1}$.
4: Take $t \leftarrow t + 1$ and go to step 3.
```

We note that, contrary to Algorithm 1, the step size ν is kept fixed during the iterations. We will see in the next section that choosing a small enough ν (depending in particular on the Lipschitz constant of Assumption $\mathbf{A_3}$) is sufficient to ensure the convergence of the algorithm. In the empirical setting, assuming that ψ is continuously differentiable in its first argument, the optimization step (7) reads

$$f_{t+1} \in \arg\min_{f \in \mathscr{P}} \frac{1}{n} \sum_{i=1}^{n} (-\nabla C(F_t)(X_i) - f(X_i))^2.$$

Therefore, in this context, the gradient boosting algorithm fits f_{t+1} to the negative gradient instances $-\nabla C(F_t)(X_i)$ via a least squares minimization. When $\psi(x, y) = (y - x)^2/2$, then $-\nabla C(F_t)(X_i) = Y_i - F_t(X_i)$, and the algorithm simply fits f_{t+1} to the residuals $Y_i - F_t(X_i)$ at step t, in the spirit of original boosting procedures. This observation is at the source of gradient boosting, which Algorithm 2 generalizes to a much larger variety of loss functions and to more abstract measures.

3 Convergence of the algorithms

This section is devoted to analyzing the convergence of the gradient boosting Algorithm 1 and Algorithm 2 as the number of iterations t tends to infinity. Despite its importance, no results (or only partial answers) have been reported so far on this question.

3.1 Algorithm 1

The convergence of this algorithm rests upon the choice of the step size sequence $(w_t)_t$, which needs to be carefully specified. We take $w_0 > 0$ arbitrarily and set

$$w_{t+1} = \min\big(w_t, -(2L)^{-1}\mathbb{E}\xi(F_t(X),Y)f_{t+1}(X)\big), \quad t \geq 0, \tag{8}$$

where L is the Lipschitz constant of Assumption $\mathbf{A_3}$. Clearly, the sequence $(w_t)_t$ is nonincreasing. It is also nonnegative. To see this, just note that, by definition,

$$f_{t+1} \in \arg\max_{f\in\mathscr{F}} - \mathbb{E}\xi(F_t(X),Y)f(X),$$

and thus, since $0 \in \mathscr{F}$, $-\mathbb{E}\xi(F_t(X),Y)f_{t+1}(X) \geq 0$. The main result of this section is encapsulated in the following theorem.

Theorem 1 *Assume that Assumptions* $\mathbf{A_1}$ *and* $\mathbf{A_3}$ *are satisfied, and let* $(F_t)_t$ *be defined by Algorithm 1 with* $(w_t)_t$ *as in (8). Then*

$$\lim_{t\to\infty} C(F_t) = \inf_{F\in\text{lin}(\mathscr{F})} C(F).$$

Proof See Supplementary Material Document. □

Observe that Theorem 1 holds without Assumption $\mathbf{A_2}$, i.e., there is no need here to assume that the function $\psi(x,y)$ is strongly convex in x. However, whenever Assumption $\mathbf{A_2}$ is satisfied, there exists as in (4) a unique boosting predictor $\bar{F} \in \overline{\text{lin}(\mathscr{F})}$ such that $C(\bar{F}) = \inf_{F\in\text{lin}(\mathscr{F})} C(F)$, and the theorem guarantees that $\lim_{t\to\infty} C(F_t) = C(\bar{F})$.

The proof of the theorem relies on the following lemma, which states that the sequence $(C(F_t))_t$ is nonincreasing. Since $C(F)$ is nonnegative for all F, we conclude that $C(F_t) \downarrow \inf_k C(F_k)$ as $t \to \infty$. This is the key argument to prove the convergence of $C(F_t)$ towards $\inf_{F\in\text{lin}(\mathscr{F})} C(F)$.

Lemma 1 *Assume that Assumptions* $\mathbf{A_1}$ *and* $\mathbf{A_3}$ *are satisfied. Then, for each* $t \geq 0$,

$$C(F_t) - C(F_{t+1}) \geq Lw_{t+1}^2.$$

In particular, $C(F_t) \downarrow \inf_k C(F_k)$ *as* $t \to \infty$, $\sum_{t\geq 1} w_t^2 < \infty$, *and* $\lim_{t\to\infty} w_t = 0$.

Proof Let $t \geq 0$. Recall that $F_{t+1} = F_t + w_{t+1}f_{t+1}$. If $f_{t+1} = 0$, then $w_{t+1} = 0$ and $F_{t+1} = F_t$, so that there is nothing to prove. Thus, in the remainder of the proof, it is assumed that f_{t+1} is different from zero and, in turn, that $\|f_{t+1}\|_{\mu_X} = 1$. Applying technical Lemma 1 of the Supplementary Material Document, we may write

$$\begin{aligned}
C(F_t) &\geq C(F_{t+1}) - w_{t+1}^2 L - w_{t+1}\mathbb{E}\xi(F_t(X),Y)f_{t+1}(X) \\
&\geq C(F_{t+1}) - w_{t+1}^2 L + 2Lw_{t+1}\min\big(w_t, -(2L)^{-1}\mathbb{E}\xi(F_t(X),Y)f_{t+1}(X)\big) \\
&= C(F_{t+1}) + Lw_{t+1}^2,
\end{aligned}$$

by definition (8) of the sequence $(w_t)_t$. □

Theorem 1 ensures that the risk of the boosting iterates gets closer and closer to the minimal risk as the number of iterations grows. It turns out that, whenever $\overline{\mathrm{lin}(\mathscr{F})} = L^2(\mu_X)$, under Assumption $\mathbf{A_2}$ and the smooth framework of Assumption $\mathbf{A'_3}$, the sequence $(F_t)_t$ itself approaches $\bar{F} = \arg\min_{F \in L^2(\mu_X)} C(F)$, as shown in Corollary 1 below. This corollary is an easy consequence of Theorem 1 and the strong convexity of C.

Corollary 1 *Assume that* $\overline{\mathrm{lin}(\mathscr{F})} = L^2(\mu_X)$. *Assume, in addition, that Assumptions* $\mathbf{A_1}$, $\mathbf{A_2}$, *and* $\mathbf{A'_3}$ *are satisfied, and let* $(F_t)_t$ *be defined by Algorithm 1 with* $(w_t)_t$ *as in (8). Then*

$$\lim_{t\to\infty} \|F_t - \bar{F}\|_{\mu_X} = 0,$$

where

$$\bar{F} = \arg\min_{F \in L^2(\mu_X)} C(F).$$

Proof By the α-strong convexity of C,

$$C(F_t) \geq C(\bar{F}) + \mathbb{E}\xi(\bar{F}, Y)(F_t - \bar{F}) + \frac{\alpha}{2}\|F_t - \bar{F}\|^2_{\mu_X},$$

which, under $\mathbf{A'_3}$, takes the more familiar form

$$C(F_t) \geq C(\bar{F}) + \langle \nabla C(\bar{F}), F_t - \bar{F}\rangle_{\mu_X} + \frac{\alpha}{2}\|F_t - \bar{F}\|^2_{\mu_X}.$$

But, since $\bar{F} = \arg\min_{F \in L^2(\mu_X)} C(F)$, we know that $\langle \nabla C(\bar{F}), F_t - \bar{F}\rangle_{\mu_X} = 0$. Thus,

$$C(F_t) - C(\bar{F}) \geq \frac{\alpha}{2}\|F_t - \bar{F}\|^2_{\mu_X},$$

and the conclusion follows from Theorem 1. □

We would like to close this subsection by stressing that Theorem 1 is not quantitative, in the sense that nothing is known about the speed of convergence when increasing the number of iterations of the algorithm. This is an open question, which unfortunately cannot be dealt with in the present article. In line with the remarks of a referee, we believe that the existing analyses for L_2Boosting [e.g., Bühlmann, 2006] and weak greedy algorithms [e.g., Temlyakov, 2000, Champion et al., 2014] could be a promising route to follow.

3.2 Algorithm 2

Recall that, in this context, each iteration picks an $f_{t+1} \in \mathscr{P}$ that satisfies

$$2\mathbb{E}\xi(F_t(X), Y) f_{t+1}(X) + \|f_{t+1}\|^2_{\mu_X} \leq 2\mathbb{E}\xi(F_t(X), Y) f(X) + \|f\|^2_{\mu_X} \quad \text{for all } f \in \mathscr{P}.$$

Theorem 2 *Assume that Assumptions* $\mathbf{A_1}$-$\mathbf{A_3}$ *are satisfied, and let* $(F_t)_t$ *be defined by Algorithm 2 with* $0 < \nu < 1/(2L)$*. Then*

$$\lim_{t\to\infty} C(F_t) = \inf_{F\in \text{lin}(\mathscr{P})} C(F).$$

Proof See Supplementary Material Document. □

The architecture of the proof is similar to that of Theorem 1. (Note however that this theorem requires the strong convexity Assumption $\mathbf{A_2}$). In particular, we need the following important lemma, which states that the risk of the iterates decreases at each step of the algorithm.

Lemma 2 *Assume that Assumptions* $\mathbf{A_1}$ *and* $\mathbf{A_3}$ *are satisfied, and let* $0 < \nu < 1/(2L)$*. Then, for each* $t \geq 0$,

$$C(F_t) - C(F_{t+1}) \geq \frac{\nu}{2}(1-2\nu L)\|f_{t+1}\|^2_{\mu_X}.$$

In particular, $C(F_t) \downarrow \inf_k C_k$ *as* $t\to\infty$, $\sum_{t\geq 1}\|f_t\|^2_{\mu_X} < \infty$, *and* $\lim_{t\to\infty}\|f_t\|_{\mu_X} = 0$.

Proof Let $t \geq 0$. Applying technical Lemma 1 of the Supplementary Material Document, we may write

$$\begin{aligned}
&C(F_t)\\
&\quad \geq C(F_{t+1}) - \nu^2 L\|f_{t+1}\|^2_{\mu_X} - \nu\mathbb{E}\xi(F_t(X),Y)f_{t+1}(X)\\
&\quad = C(F_{t+1}) - \nu^2 L\|f_{t+1}\|^2_{\mu_X} - \frac{\nu}{2}\big(2\mathbb{E}\xi(F_t(X),Y)f_{t+1}(X) + \|f_{t+1}\|^2_{\mu_X}\big) + \frac{\nu}{2}\|f_{t+1}\|^2_{\mu_X}.
\end{aligned}$$

Upon noting that $2\mathbb{E}\xi(F_t(X),Y)f_{t+1}(X) + \|f_{t+1}\|^2_{\mu_X} \leq 0$ (since $0\in\mathscr{P}$), we conclude that

$$C(F_t) \geq C(F_{t+1}) + \frac{\nu}{2}(1-2\nu L)\|f_{t+1}\|^2_{\mu_X}.$$

□

Remark 1 The parameter ν can be regarded as controlling the learning rate of the boosting procedure. The lower bound of Lemma 2 suggests the optimal value $\nu^\star = 1/(4L)$. In practice, ν is often chosen "small enough", which leads to a larger number of iterations (and thus more computing time) for the same training risk. All in all, both ν and the number of iterations control prediction risk and these parameters do not operate independently.

As in Algorithm 1, the sequence $(F_t)_t$ approaches $\bar{F} = \arg\min_{F\in L^2(\mu_X)} C(F)$, provided $\overline{\text{lin}(\mathscr{P})} = L^2(\mu_X)$ and $\mathbf{A}_3'$ is satisfied in place of $\mathbf{A_3}$. This is summarized in the following Corollary. Its proof is similar to the proof of Corollary 1 and is therefore omitted.

Corollary 2 *Assume that* $\overline{\text{lin}(\mathscr{P})} = L^2(\mu_X)$*. Assume, in addition, that Assumptions* $\mathbf{A_1}$, $\mathbf{A_2}$, *and* $\mathbf{A}_3'$ *are satisfied, and let* $(F_t)_t$ *be defined by Algorithm 2 with* $0 < \nu < 1/(2L)$*. Then*

$$\lim_{t\to\infty} \|F_t - \bar{F}\|_{\mu_X} = 0,$$

where

$$\bar{F} = \arg\min_{F\in L^2(\mu_X)} C(F).$$

Theorem 1/Corollary 1 and Theorem 2/Corollary 2 guarantee that, under appropriate assumptions, Algorithm 1 and Algorithm 2 converge towards the infimum of the risk functional. Given the unusual form of these algorithms, which have the flavor of gradient descents while being different, these results are all but obvious and cannot be deduced from general optimization principles. As far as we know, they are novel in the gradient boosting literature and extend our understanding of the approach.

Perhaps the most natural framework of Algorithm 1 and Algorithm 2 is when $\mu_{X,Y} = \mu_n$, the empirical measure. In this statistical context, both algorithms track the infimum of the empirical risk functional $C_n(F) = \frac{1}{n}\sum_{i=1}^n \psi(F(X_i), Y_i)$ over the linear combinations of weak learners in $\mathscr{F}$ (Algorithm 1) or in $\mathscr{P}$ (Algorithm 2). This task is achieved by sequentially constructing linear combinations of base learners, of the form $F_t = F_0 + \sum_{k=1}^t w_k f_k$ with $f_k \in \mathscr{F}$ for Algorithm 1, and $F_t = F_0 + \nu\sum_{k=1}^t f_k$ with $f_k \in \mathscr{P}$ for Algorithm 2. We stress that, in the empirical case, the boosted iterates F_t and their eventual limit $\bar{F}_n$ are measurable functions of the data set $\mathscr{D}_n$. That being said, Theorem 1 and Theorem 2 are numerical-analysis-type results, which do not provide information on the statistical properties of the boosting predictor $\bar{F}_n$. From this point of view, more or less catastrophic situations can happen, depending on the "size" of $\text{lin}(\mathscr{F})$ (Algorithm 1) or $\text{lin}(\mathscr{P})$ (Algorithm 2), which should not be neither too small (to catch complex decisions) nor excessively large (to avoid overfitting).

To be convinced of this, consider for example Algorithm 1 with $\psi(x, y) = (y-x)^2$ (least squares regression problem) and $\mathscr{F}$ = all binary trees with $d+1$ leaves. Denote by P_n the empirical measure based on the X_i only, $1 \leq i \leq n$. Then, by Theorem 1, $\lim_{t\to\infty} C_n(F_t) = C_n(\bar{F}_n)$, where

$$\bar{F}_n = \arg\min_{F\in L^2(P_n)} C_n(F).$$

Assume, to simplify, that all X_i are different. It is then easy to see that the boosting predictor $\bar{F}_n$ takes the value Y_i at each X_i and is arbitrarily defined elsewhere. Of course, in general, such a function $\bar{F}_n$ does not converge as $n \to \infty$ towards the regression function $F^\star(x) = \mathbb{E}(Y|X = x)$, and this is a typical situation where the gradient boosting algorithms overfit. The overfitting issue of boosting procedures has been recognized for a long time, and various approaches have been proposed to combat it, in particular via early stopping [that is, stopping the iterations before convergence; see, e.g., Bühlmann and Yu, 2003, Mannor et al., 2003, Zhang and Yu, 2005, Bickel et al., 2006, Bartlett and Traskin, 2007].

Nevertheless, the natural question we would like to answer is whether there exists a reasonable context in which the boosting predictors enjoy good statistical properties as the sample size grows, without resorting to any stopping strategy. The next section provides a positive response. The major constraint we face, imposed by the gradient-descent nature of the algorithms, is that we are required to perform a minimization over a vector space ($\mathrm{lin}(\mathscr{F})$ for Algorithm 1 and $\mathrm{lin}(\mathscr{P})$ for Algorithm 2). In particular, there is no question of imposing constraints on the coefficients of the linear combinations, which, for example, cannot reasonably be assumed to be bounded. As we will see, the trick is to carefully constrain the "complexity" of the vector spaces $\mathrm{lin}(\mathscr{F})$ or $\mathrm{lin}(\mathscr{P})$ in a manner compatible with the algorithms. The second message is the importance of having a strongly convex risk functional to minimize, which, in some way, restrict the norm of the sequence $(F_t)_{t\geq 0}$ of boosted iterates. As we have pointed out several times, if the loss function is not natively strongly convex in its first argument, then this type of regularization can be achieved by resorting to an L^2-type penalty.

4 Large sample properties

We consider in this section a functional minimization problem whose solution can be computed by gradient boosting and enjoys non-trivial statistical properties. The context and notation are similar to that of the previous sections, but must be slightly adapted to fit this new framework.

For simplicity, it will be assumed throughout that $\mathscr{X}$ is a compact subset of $\mathbb{R}^d$. We consider i.i.d. data $\mathscr{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ taking values in $\mathscr{X} \times \mathscr{Y}$, and let P_n be the empirical measure based on the X_i only, $1 \leq i \leq n$. We denote by P the common distribution of the X_i and assume that P has a density g with respect to the Lebesgue measure λ on $\mathbb{R}^d$, with

$$0 < \inf_{\mathscr{X}} g \leq \sup_{\mathscr{X}} g < \infty.$$

We concentrate on Algorithm 1 and take as weak learners a finite class $\mathscr{F}_n$ of simple functions on $\mathscr{X}$ with ±1 values, which may possibly vary with the sample size n. It is actually easy to verify that all subsequent results are valid for Algorithm 2 by letting $\mathscr{P}_n = \{\lambda f : f \in \mathscr{F}_n, \lambda \in \mathbb{R}\}$.

The typical example we have in mind for $\mathscr{F}_n$ is a finite class of binary trees using axis parallel cuts with k leaves. Of course, the parameter k has to be carefully chosen as a function of the sample size to guarantee consistency, as we will see below. The fact that the class $\mathscr{F}_n$ is supposed to be finite should not be too disturbing, since in practice the optimization step (6) is typically performed over a finite family of functions. This is for example the case when a CART-style top-down recursive partitioning is used to compute the minimum at each iteration of the algorithm. In this approach, the optimal tree in (6) is greedily searched for by passing from one level of node to the next one with cuts that are located between two data points. So, even though the collection $\mathscr{F}_n$ may be very large, it is nevertheless fair to assume that its cardinal is finite.

As before, it is assumed that the identically zero function belongs to $\mathscr{F}_n$. So, in this framework, we see that there exists a (large) integer $N = N(n) \geq 1$ and a partition of

$\mathscr{X}$ into measurable subsets A_j^n, $1 \leq j \leq N$, such that any $F \in \text{lin}(\mathscr{F}_n)$ takes the form $F = \sum_{j=1}^N \alpha_j \mathbb{1}_{A_j^n}$, where $(\alpha_1, \ldots, \alpha_N) \in \mathbb{R}^N$. To avoid pathological situations, we assume that there exists a positive sequence $(\nu_n)_n$ such that $\min_{1 \leq j \leq N} \lambda(A_j^n) \geq \nu_n$. Of course, it is supposed that $N \to \infty$ as n tends to infinity.

We let $\phi : \mathbb{R} \times \mathscr{Y} \to \mathbb{R}_+$ be a loss function, assumed to be convex in its first argument and to satisfy $\bar{\phi} := \sup_{y \in \mathscr{Y}} \phi(0, y) < \infty$. In line with the previous sections, we are interested in minimizing over $\text{lin}(\mathscr{F}_n)$ the empirical risk functional $C_n(F)$ defined by

$$C_n(F) = \frac{1}{n} \sum_{i=1}^n \psi(F(X_i), Y_i),$$

where $\psi(x, y) = \phi(x, y) + \gamma_n x^2$ and $(\gamma_n)_n$ is a sequence of positive parameters such that $\lim_{n \to \infty} \gamma_n = 0$. (Note that γ_n depends only on n and is therefore kept fixed during the iterations of the algorithm.) Put differently,

$$C_n(F) = A_n(F) + \gamma_n \|F\|_{P_n}^2, \tag{9}$$

where

$$A_n(F) = \frac{1}{n} \sum_{i=1}^n \phi(F(X_i), Y_i).$$

Assumption $\mathbf{A_1}$ is obviously satisfied (with $\mu_{X,Y} = \mu_n$, in the notation of Section 3), and the same is true for Assumption $\mathbf{A_2}$ by the α-strong convexity of the function $\psi(\cdot, y)$ for each fixed y, with α independent of y.

Remark 2 If the function $\phi(\cdot, y)$ is natively α-strongly convex with a parameter α independent of y, then we may consider the simplest problem of minimizing the functional $A_n(F)$. Indeed, in this case there is no need to resort to the $\gamma_n \|F\|_{P_n}^2$ penalty term since Lemma 3 of the Supplementary Material Document allows to bound $\|F\|_{P_n}^2$. As we have seen in Section 2, this is for example the case in the least squares problem, when $\phi(x, y) = (y - x)^2$. However, to keep a sufficient degree of generality, we will consider in the following the more general optimization problem (9).

Now, let

$$\bar{F}_n = \arg\min_{F \in \text{lin}(\mathscr{F}_n)} C_n(F).$$

We have learned in Theorem 1 that whenever Assumption $\mathbf{A_3}$ is satisfied, the boosted iterates $(F_t)_t$ of Algorithm 1 satisfy $\lim_{t \to \infty} C_n(F_t) = C_n(\bar{F}_n)$, i.e.,

$$\lim_{t \to \infty} \left(A_n(F_t) + \gamma_n \|F_t\|_{P_n}^2 \right) = A_n(\bar{F}_n) + \gamma_n \|\bar{F}_n\|_{P_n}^2.$$

For $F \in L^2(P)$, the population counterpart of $A_n(F)$ is the convex functional $A(F) := \mathbb{E}\phi(F(X_1), Y_1)$, which is assumed to be locally bounded, and thus continuous. Throughout, we denote by $F^\star$ a minimizer of $A(F)$ over $L^2(P)$, i.e.,

$$F^\star \in \arg\min_{F \in L^2(P)} A(F).$$

We have for example $F^\star(x) = \mathbb{E}(Y|X = x)$ in the regression problem with $\phi(x, y) = (y - x)^2$ and $F^\star(x) = \log(\frac{\eta(x)}{1-\eta(x)})$ in the classification problem with $\phi(x, y) = \log_2(1 + e^{-yx})$, where $\eta(x) = \mathbb{P}(Y = 1|X = x)$.

Our goal in this section is to investigate the large sample properties of $\bar{F}_n$, i.e., to analyze the statistical behavior of the boosting predictor $\bar{F}_n$ as $n \to \infty$. In particular, a sensible objective is to show that $A(\bar{F}_n)$ gets asymptotically close to the minimal risk $A(F^\star)$ as the sample size grows. This necessitates a proof, since all we know for now is that

$$A_n(\bar{F}_n) + \gamma_n \|\bar{F}_n\|^2_{P_n} - A(F^\star) = \inf_{F \in \text{lin}(\mathscr{F}_n)} \big(A_n(F) + \gamma_n \|F\|^2_{P_n} - A(F^\star)\big),$$

which is our starting point. The following assumption on ϕ will be needed in the analysis:

A$_4$ For all $p \geq 0$, there exists a constant $\zeta(p) > 0$ such that, for all $(x_1, x_2, y) \in \mathbb{R}^2 \times \mathscr{Y}$ with $\max(|x_1|, |x_2|) \leq p$,

$$|\phi(x_1, y) - \phi(x_2, y)| \leq \zeta(p)|x_1 - x_2|.$$

It is readily seen that all classical convex losses in regression and classification satisfy this local Lipschitz assumption. Finally, we let $A^n(x) = A^n_j$ whenever $x \in A^n_j$, and, for $E \subset \mathbb{R}^d$,

$$\text{diam}(E) = \sup_{x, x' \in E} \|x - x'\|.$$

Recall that $\bar{\phi} := \sup_{y \in \mathscr{Y}} \phi(0, y) < \infty$.

Theorem 3 *Assume that Assumptions* **A$_3$** *(with $\psi(x, y) = \phi(x, y) + \gamma_n x^2$) and* **A$_4$** *are satisfied, and that $F^\star$ is bounded. Assume, in addition, that* $\text{diam}(A^n(X)) \to 0$ *in probability as $n \to \infty$. Then, provided $\gamma_n \to 0$, $N \to \infty$, $\frac{\log N}{n v_n} \to 0$, and*

$$\frac{1}{\sqrt{n v_n \gamma_n}} \zeta\left(\sqrt{\frac{2\bar{\phi}}{v_n \gamma_n \inf_{\mathscr{X}} g}}\right) \to 0,$$

we have $\lim_{n\to\infty} \mathbb{E}A(\bar{F}_n) = A(F^\star)$.

Proof See Supplementary Material Document. □

The main message of this theorem is that, under appropriate conditions on the loss and provided the size of the weak learner classes are judiciously increased, gradient boosting does not overfit. In other words, in this framework, stopping the iterations is not necessary and the algorithms may be run indefinitely, without worrying about early stopping issues.

In line with Remark 2, we leave it as an exercise to prove that if the function $\phi(\cdot, y)$ is already α-strongly convex with a parameter α independent of y, then a similar result holds with the conditions $N \to \infty$, $\frac{\log N}{n v_n} \to 0$, and

$$\frac{1}{\sqrt{n v_n}} \zeta\left(\sqrt{\frac{a}{v_n \inf_{\mathscr{X}} g}}\right) \to 0,$$

where $a = \frac{2}{\alpha} \sup_{y \in \mathscr{Y}} |\xi(0, y)| + \sqrt{2\bar{\phi}/\alpha}$. In this case, we can take $\gamma_n = 0$ (i.e., no penalty) and resort to Lemma 3 of the Supplementary Material Document to bound the quantity $\|F\|^2_{P_n}$.

Next, we point out that the conditions of Theorem 3 are mild and cover a wide variety of losses and possible classes of weak learners. As an example, let $\mathscr{X} = [0,1]^d$ and take for $\mathscr{F}_n$ the set of all binary trees on $[0,1]^d$ with k_n leaves, where cuts are perpendicular to the axes and located at the middle of the cells. Although combinatorially rich, this family of trees is finite (see Figure 1 for an illustration in dimension $d = 2$).

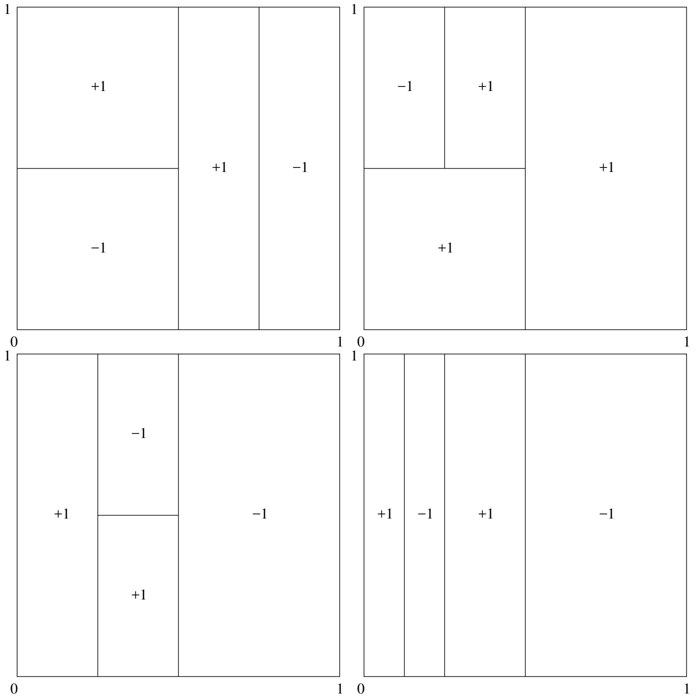

Fig. 1 Four examples of trees in the class $\mathscr{F}_n$, in dimension $d = 2$, with $k_n = 4$.

It is easy to verify that any $F \in \text{lin}(\mathscr{F}_n)$ takes the form $F = \sum_{j=1}^{N} \alpha_j \mathbb{1}_{A_j^n}$, where $N \leq 2^{dk_n}$ and the A_j^n, $1 \leq j \leq N$, form a regular grid over $[0,1]^d$. Thus, clearly, $v_n \geq 2^{-dk_n}$. In addition, considering for example the loss $\phi(x, y) = (y - x)^2$, we see that the conditions of Theorem 3 take the simple form

$$k_n \to \infty, \quad \frac{k_n 2^{dk_n}}{n} \to 0, \quad \text{and} \quad \frac{2^{dk_n}}{\sqrt{n}} \to 0.$$

Let us finally note that in the ±1-classification setting, each F defines a classifier g_F in a natural way, by

$$g_F(x) = \begin{cases} +1 & \text{if } F(x) > 0 \\ -1 & \text{otherwise,} \end{cases}$$

and the main concern is not the behavior of the theoretical risk $A(F)$ with respect to $A(F^\star)$, but rather the proximity between the probability of error $L(g_F) := \mathbb{P}(g_F(X) \neq Y)$ and the Bayes risk $L^\star := \inf_{g:\mathscr{X}\to\{-1,1\}} \mathbb{P}(g(X) \neq Y)$. For most classification losses [Zhang, 2004, Bartlett et al., 2006], the difference $L(g_F) - L^\star$ is small as long as $A(F) - A(F^\star)$ is. In our framework, we conclude that for such well-behaved losses, under the assumptions of Theorem 3,

$$\lim_{n\to\infty} \mathbb{E}L(g_{\bar{F}_n}) = L^\star.$$

Acknowledgements We greatly thank the Editors and two referees for valuable comments and insightful suggestions, which led to a substantial improvement of the paper.

References

P.L. Bartlett and M. Traskin. AdaBoost is consistent. *Journal of Machine Learning Research*, 8:2347–2368, 2007.

P.L. Bartlett, M.I. Jordan, and J.D. McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101:138–156, 2006.

P.J. Bickel, Y. Ritov, and A. Zakai. Some theory for generalized boosting algorithms. *Journal of Machine Learning Research*, 7:705–732, 2006.

G. Blanchard, G. Lugosi, and N. Vayatis. On the rate of convergence of regularized boosting classifiers. *Journal of Machine Learning Research*, 4:861–894, 2003.

L. Breiman. *Arcing the edge*. Technical Report 486, Statistics Department, University of California, Berkeley, 1997.

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

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

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

L. Breiman. Population theory for boosting ensembles. *The Annals of Statistics*, 32: 1–11, 2004.

L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. *Classification and Regression Trees*. Chapman & Hall/CRC Press, Boca Raton, 1984.

S. Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8:231–357, 2015.

P. Bühlmann. Boosting for high-dimensional linear models. *The Annals of Statistics*, 34:559–583, 2006.

P. Bühlmann and T. Hothorn. Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22:477–505, 2007.

P. Bühlmann and S. van de Geer. *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer, Berlin, 2011.

P. Bühlmann and B. Yu. Boosting with the L_2 loss: Regression and classification. *Journal of the American Statistical Association*, 98:324–339, 2003.

M. Champion, C. Cierco-Ayrolles, S. Gadat, and M. Vignes. Sparse regression and support recovery with L_2-boosting algorithms. *Journal of Statistical Planning and Inference*, 155:19–41, 2014.

T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM, New York, 2016.

L. Devroye and L. Györfi. *Nonparametric Density Estimation: The L_1 View*. Wiley, New York, 1985.

L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer, New York, 1996.

M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.

Y. Freund. Boosting a weak learning algorithm by majority. *Information and Computation*, 121:256–285, 1995.

Y. Freund and R.E. Schapire. Experiments with a new boosting algorithm. In S. Lorenza, editor, *Machine Learning: Proceedings of the Thirteenth International Conference on Machine Learning*, pages 148–156. Morgan Kaufmann Publishers, San Francisco, 1996.

Y. Freund and R.E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55: 119–139, 1997.

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

J.H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29:1189–1232, 2001.

J.H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38:367–378, 2002.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Second Edition*. Springer, New York, 2009.

G. Lugosi and N. Vayatis. On the Bayes-risk consistency of regularized boosting methods. *The Annals of Statistics*, 32:30–55, 2004.

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

S. Mannor, R. Meir, and T. Zhang. Greedy algorithms for classification — consistency, convergence rates, and adaptivity. *Journal of Machine Learning Research*, 4:713–742, 2003.

L. Mason, J. Baxter, P. Bartlett, and M. Frean. Boosting algorithms as gradient descent. In S.A. Solla, T.K. Leen, and K. Müller, editors, *Proceedings of the 12th International Conference on Neural Information Processing Systems*, pages 512–518. The MIT Press, Cambridge, MA, 1999.

L. Mason, J. Baxter, P. Bartlett, and M. Frean. Functional gradient techniques for combining hypotheses. In A.J. Smola, P.L. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 221–246. The MIT Press, Cambridge, MA, 2000.

R. Meir and G. Rätsch. An introduction to boosting and leveraging. In S. Mendelson and A.J. Smola, editors, *Advanced Lectures on Machine Learning: Machine Learning Summer School 2002*, pages 118–183. Springer, Berlin, 2003.

R.E. Schapire. The strength of weak learnability. *Machine Learning*, 5:197–227, 1990.

V.N. Temlyakov. Weak greedy algorithms. *Advances in Computational Mathematics*, 12:213–227, 2000.

T. Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. *The Annals of Statistics*, 32:56–85, 2004.

T. Zhang and B. Yu. Boosting with early stopping: Convergence and consistency. *The Annals of Statistics*, 33:1538–1579, 2005.