12 Attaining the Best Possible Accuracy

In this last part of the book, we study a number of advanced theoretical topics with a continuing focus on fundamental properties and limitations of boosting and AdaBoost, as well as techniques and principles for the design of improved algorithms.

We begin, in this chapter, with a return to the central problem of understanding AdaBoost's ability to generalize. Previously, in chapters 4 and 5, we provided analyses of AdaBoost's generalization error where, as in our study of boosting generally, we took as our starting point the weak learning assumption, that is, the premise that the classifiers generated by the weak learning algorithm are reliably better than random guessing. Naively, this assumption did indeed seem weak, but we have now come to see that its consequences are actually quite strong. Not only does it imply that eventually boosting will perfectly fit any training set, but the results of chapters 4 and 5 show that it also implies that the generalization error can be driven arbitrarily close to zero with sufficient training data. This is an excellent property—the very one that defines boosting.

On the other hand, we know that it cannot always be possible to attain perfect generalization accuracy. Typically, we expect real data to be corrupted with some form of noise, randomness, or mislabeling that makes it impossible to perfectly predict the labels of nearly all test examples, even with unlimited training and computation. Instead, we are faced with a fundamental limit on how much the test error can be minimized due to intrinsic randomness in the data itself. This minimum possible error rate is called the *Bayes error*.

Thus, our earlier analyses superficially appear to be inapplicable when the Bayes error is strictly positive. However, this is not necessarily the case. Even if the weak learning assumption does not hold so that the weighted errors of the weak hypotheses are converging to $\frac{1}{2}$, these analyses can still be applied, depending as they do on the edges of all the weak hypotheses. Moreover, in practice the weak learning assumption may in fact continue to hold, even when perfect generalization is unachievable. This is because the weak hypothesis space typically is not fixed, but grows in complexity with the size of the training set; for instance, this happens "automatically" when using decision trees as base classifiers since the generated trees will usually be bigger if trained with more data. This presents the usual delicate balance between complexity and fit to the data, but one that leaves open

the possibility, according to our analysis, for very good generalization, as is often seen in practice.

Nevertheless, these analyses do not explicitly provide absolute guarantees on the performance of AdaBoost relative to the optimal Bayes error (other than when it is zero). In other words, they do not specify conditions under which AdaBoost's generalization error will necessarily converge to the best possible error rate; rather, they provide generalization bounds which are in terms of statistics that can be measured only after training is complete.

In this chapter, we give an alternative analysis in which we prove that a slight variation of AdaBoost does indeed produce a combined classifier whose accuracy is very close to the optimum attainable by any classifier whatsoever, provided the base classifiers are sufficiently but not overly expressive, and provided the training set is sufficiently large. In this sense, the algorithm is said to be *universally consistent*. (Note that this notion of consistency is entirely unrelated to and distinct from the one studied, for instance, in section 2.2.5.)

This analysis pulls together many of the topics studied earlier in this book, particularly the view of AdaBoost as an algorithm for minimizing exponential loss. The analysis shows first that AdaBoost quickly minimizes the true expected exponential loss relative to the minimum possible, and then shows how this directly implies good classification accuracy compared to the Bayes optimal.

Although these results are strong, they are limited by their underlying assumptions, especially with regard to the expressiveness of the base hypotheses. To emphasize this point, we also give a simple example in which minimization of exponential loss provably fails to generate a classifier close to the Bayes optimal, even when the noise affecting the data is of a particularly simple form.

12.1 Optimality in Classification and Risk Minimization

We begin with a discussion of optimality in classification and its relation to minimization of exponential loss. We return to the simple problem of binary classification with $\mathcal{X}$ denoting the instance space, and the set of possible labels consisting only of $\mathcal{Y} = \{-1, +1\}$. We let $\mathcal{D}$ denote the true distribution over labeled pairs in $\mathcal{X} \times \mathcal{Y}$. Unless specified otherwise, in this chapter probabilities and expectations denoted $\mathbf{Pr}[\cdot]$ and $\mathbf{E}[\cdot]$ are with respect to a random pair (x, y) generated according to $\mathcal{D}$.

In general, for such a random pair, the label y will not necessarily be determined by the instance x. In other words, the conditional probability that x is labeled positive, denoted

$$\pi(x) \doteq \mathbf{Pr}[y = +1 \mid x], \tag{12.1}$$

need not be equal to 0 or 1. When $\pi(x) \in (0, 1)$, it becomes inherently impossible to predict y perfectly from x, even with full knowledge of $\mathcal{D}$. Nevertheless, we can still characterize

the best that is possible in minimizing the chance of an incorrect prediction. In particular, if y is predicted to be $+1$, then the probability of being incorrect is $1-\pi(x)$; and if y is predicted to be -1, then an error occurs with probability $\pi(x)$. Thus, to minimize the chance of a mistake, we should predict using the rule

$$h_{\text{opt}}(x) = \begin{cases} +1 & \text{if } \pi(x) > \frac{1}{2} \\ -1 & \text{if } \pi(x) < \frac{1}{2}. \end{cases}$$

(It makes no difference how we predict if $\pi(x) = \frac{1}{2}$.) This rule is called the *Bayes optimal classifier*. Its error, called the *Bayes (optimal) error*, is exactly

$$\text{err}^* \doteq \text{err}(h_{\text{opt}}) = \mathbf{E}[\min\{\pi(x), 1-\pi(x)\}] .$$

This is the minimum error achievable by *any* classifier, regardless of any considerations of learning or computation. (Here, as usual, $\text{err}(h)$ denotes the generalization error of a classifier h as in equation (2.3).)

Thus, the best we can hope for in a learning procedure is that its error will converge to the Bayes error. The purpose of this chapter is to give general conditions under which AdaBoost has this property.

As seen in section 7.1, AdaBoost can be interpreted as an algorithm for minimizing exponential loss. That is, given a training set $S = \langle (x_1, y_1), \ldots, (x_m, y_m) \rangle$, AdaBoost minimizes the *empirical risk* (or loss)

$$\widehat{\text{risk}}(F) \doteq \frac{1}{m} \sum_{i=1}^{m} e^{-y_i F(x_i)}$$

over all linear combinations F of base classifiers in the given space $\mathcal{H}$. (We assume an exhaustive weak learner that, on every round, returns the best weak hypothesis.) The empirical risk can itself be viewed as an estimate or proxy for the *true risk*, that is, the expected loss with respect to the true distribution $\mathcal{D}$:

$$\text{risk}(F) \doteq \mathbf{E}\left[e^{-yF(x)}\right]. \tag{12.2}$$

As seen in section 7.5.3, this expectation can be broken down using marginalization as

$$\mathbf{E}\left[\mathbf{E}\left[e^{-yF(x)} \mid x\right]\right] = \mathbf{E}\left[\pi(x)e^{-F(x)} + (1-\pi(x))e^{F(x)}\right], \tag{12.3}$$

where the outer expectations are only with respect to x, and the inner expectation on the left is with respect to y conditioned on x. As with classification error, we can now compute the minimum possible value of this risk by optimizing on each instance x separately. This can be done by setting to zero the first derivative of the expression inside the expectation (taken with respect to $F(x)$). Doing so gives the optimal predictor

$$F_{\text{opt}}(x) = \frac{1}{2} \ln \left(\frac{\pi(x)}{1 - \pi(x)} \right) \tag{12.4}$$

where we allow this function to include $\pm\infty$ in its range in case $\pi(x)$ is 0 or 1. With respect to exponential loss, this is the optimal predictor over *all* real-valued functions F, not only those that are linear combinations of the base classifiers. Plugging back into equation (12.3) gives the *optimal (exponential) risk*

$$\text{risk}^* \doteq \text{risk}(F_{\text{opt}}) = 2\mathbf{E}\left[\sqrt{\pi(x)(1 - \pi(x))}\right].$$

Note that

$$\text{sign}(F_{\text{opt}}(x)) = \begin{cases} +1 & \text{if } \pi(x) > \frac{1}{2} \\ -1 & \text{if } \pi(x) < \frac{1}{2}. \end{cases}$$

Thus, the sign of F_{opt}, the minimizer of the exponential risk, is exactly equal to the Bayes optimal classifier h_{opt} (ignoring the case $\pi(x) = \frac{1}{2}$). This means that if we can minimize the exponential loss—not only on the training set, but also over the entire distribution—then we can trivially convert it into a classifier that is optimal with respect to classification accuracy.

Of course, finding F_{opt} exactly is sure to be infeasible since we are working only with a finite training sample from $\mathcal{D}$, and also because our learning algorithms are restricted to use functions F of a particular form. Nevertheless, we will see that it is sufficient to find a function F whose risk is *close* to optimal. That is, if F's true risk is close to risk*, then the generalization error of $\text{sign}(F)$ will also be close to the Bayes error. This is the first part of our analysis.

In the second part, we bound the risk of the predictor F generated by AdaBoost relative to the optimal risk, thus also obtaining a bound on the generalization error of its combined classifier $H = \text{sign}(F)$ relative to the Bayes error. (Here, we are using $f(g)$ as shorthand for the function obtained by composing f with g.)

Beginning with the first part of the analysis, the next theorem shows generally that closeness to the optimal risk also implies closeness to the Bayes error.

Theorem 12.1 In the notation above, suppose the function $F : \mathcal{X} \to \mathbb{R}$ is such that

$$\text{risk}(F) \leq \text{risk}^* + \varepsilon. \tag{12.5}$$

Let $h(x) = \text{sign}(F(x))$ if $F(x) \neq 0$, and let $h(x)$ be chosen arbitrarily from $\{-1, +1\}$ otherwise. Then

$$\text{err}(h) \leq \text{err}^* + \sqrt{2\varepsilon - \varepsilon^2} \leq \text{err}^* + \sqrt{2\varepsilon}.$$

Proof Let us focus first on a single instance $x \in \mathcal{X}$. Let $v(x)$ denote the conditional probability that h misclassifies x relative to the conditional probability of h_{opt} doing the same. That is,

$$\nu(x) \doteq \mathbf{Pr}[h(x) \neq y \mid x] - \mathbf{Pr}\big[h_{\text{opt}}(x) \neq y \mid x\big].$$

Our eventual goal is to bound

$$\mathbf{E}[\nu(x)] = \text{err}(h) - \text{err}(h_{\text{opt}}) = \text{err}(h) - \text{err}^*.$$

Clearly, $\nu(x) = 0$ if $h(x) = h_{\text{opt}}(x)$. Otherwise, suppose $h_{\text{opt}}(x) = -1$ (so that $\pi(x) \leq \frac{1}{2}$) but $h(x) = +1$. Then we can compute directly that

$$\nu(x) = (1 - \pi(x)) - \pi(x) = 1 - 2\pi(x).$$

Similarly, $\nu(x) = 2\pi(x) - 1$ if $h_{\text{opt}}(x) = +1$ and $h(x) = -1$. Thus, in general,

$$\nu(x) = \begin{cases} 0 & \text{if } h(x) = h_{\text{opt}}(x) \\ |1 - 2\pi(x)| & \text{else.} \end{cases} \tag{12.6}$$

Likewise, let $\rho(x)$ be the corresponding quantity for the risk:

$$\rho(x) \doteq \mathbf{E}\big[e^{-yF(x)} \mid x\big] - \mathbf{E}\big[e^{-yF_{\text{opt}}(x)} \mid x\big].$$

This quantity is always nonnegative since the risk is minimized pointwise for every x by F_{opt}. By assumption,

$$\mathbf{E}[\rho(x)] = \text{risk}(F) - \text{risk}(F_{\text{opt}}) = \text{risk}(F) - \text{risk}^* \leq \varepsilon.$$

If $h(x) = +1$ but $h_{\text{opt}}(x) = -1$, then $F(x) \geq 0$ but $\pi(x) \leq \frac{1}{2}$. Under these circumstances, the conditional risk

$$\mathbf{E}\big[e^{-yF(x)} \mid x\big] = \pi(x)e^{-F(x)} + (1 - \pi(x))e^{F(x)}, \tag{12.7}$$

as a function of $F(x)$, is convex with a single minimum at $F_{\text{opt}}(x) \leq 0$. This means that its minimum on the restricted range $F(x) \geq 0$ is realized at the point closest to $F_{\text{opt}}(x)$, namely, $F(x) = 0$. Thus, equation (12.7) is at least 1 in this case. A symmetric argument shows that the same holds when $h(x) = -1$ but $h_{\text{opt}}(x) = +1$. Therefore, by equation (12.4),

$$\rho(x) \geq \begin{cases} 0 & \text{if } h(x) = h_{\text{opt}}(x) \\ 1 - 2\sqrt{\pi(x)(1 - \pi(x))} & \text{else.} \end{cases} \tag{12.8}$$

Let $\phi : [0, 1] \to [0, 1]$ be defined by

$$\phi(z) \doteq 1 - \sqrt{1 - z^2}.$$

Then equations (12.6) and (12.8) imply that

$$\rho(x) \geq \phi(\nu(x)) \tag{12.9}$$

for all x. This is because if $h(x) = h_{\text{opt}}(x)$, then $\phi(\nu(x)) = \phi(0) = 0 \leq \rho(x)$. And if $h(x) \neq h_{\text{opt}}(x)$, then

$$\phi(\nu(x)) = 1 - \sqrt{1 - |1 - 2\pi(x)|^2} = 1 - 2\sqrt{\pi(x)(1 - \pi(x))} \leq \rho(x).$$

It can be verified (by taking derivatives) that ϕ is convex. Thus, by equation (12.9) and Jensen's inequality (equation (A.4)),

$$\mathbf{E}[\rho(x)] \geq \mathbf{E}[\phi(\nu(x))] \geq \phi\left(\mathbf{E}[\nu(x)]\right).$$

Since ϕ is strictly increasing, it has a well-defined inverse that is also increasing, namely,

$$\phi^{-1}(z) = \sqrt{2z - z^2}. \tag{12.10}$$

Pulling everything together gives

$$\begin{aligned}
\operatorname{err}(h) - \operatorname{err}(h_{\text{opt}}) &= \mathbf{E}[\nu(x)] \\
&\leq \phi^{-1}\left(\mathbf{E}[\rho(x)]\right) \\
&= \phi^{-1}\left(\operatorname{risk}(F) - \operatorname{risk}(F_{\text{opt}})\right) \\
&\leq \phi^{-1}(\varepsilon) = \sqrt{2\varepsilon - \varepsilon^2}.
\end{aligned}$$

■

12.2 Approaching the Optimal Risk

Theorem 12.1 shows that we can find a classifier that is close in accuracy to the Bayes optimal if we can approximately minimize the expected exponential loss relative to the best possible among all real-valued functions. We know that AdaBoost minimizes exponential loss; specifically, in section 8.2 we proved that AdaBoost asymptotically (that is, in the limit of a large number of rounds) minimizes the exponential loss on the training set relative to the best linear combination of base classifiers. Unfortunately, this is inadequate for our current purposes because, to apply theorem 12.1 to AdaBoost, we will need to extend this analysis along several dimensions: First, we will need nonasymptotic results that give explicit rates of convergence (unlike the analysis of section 8.2); second, we now need to analyze the true, rather than the empirical, risk; and third, we now require convergence to the optimal among *all* functions, not just those that are linear combinations of base classifiers.

12.2.1 Expressiveness of the Base Hypotheses

We will eventually need to address all of these, but we start with the last point, which regards the expressiveness of the weak hypotheses in the space $\mathcal{H}$. Let us denote the *span* of $\mathcal{H}$, that is, the set of all linear combinations of weak hypotheses in $\mathcal{H}$, by

$$\operatorname{span}(\mathcal{H}) \doteq \left\{ F : x \mapsto \sum_{t=1}^{T} \alpha_t h_t(x) \;\middle|\; \alpha_1, \ldots, \alpha_T \in \mathbb{R};\, h_1, \ldots, h_T \in \mathcal{H};\, T \geq 1 \right\}.$$

For simplicity, we assume $\mathcal{H}$ consists only of binary classifiers with range $\{-1, +1\}$, and we also assume $\mathcal{H}$ is closed under negation so that $-h \in \mathcal{H}$ whenever $h \in \mathcal{H}$.

To apply theorem 12.1 to AdaBoost, the algorithm must at least have the potential opportunity to choose a function F whose true risk is close to the best possible. Since such algorithms output functions only in the span of $\mathcal{H}$, this means that we must assume that there exist functions in span($\mathcal{H}$) which have close to minimum risk. In other words, for any $\varepsilon > 0$, we need to assume that there exists some F in span($\mathcal{H}$) which satisfies equation (12.5). This is equivalent to assuming that

$$\inf_{F \in \text{span}(\mathcal{H})} \text{risk}(F) = \text{risk}^*. \tag{12.11}$$

This is our strongest and most important assumption. In section 12.3, we will explore what happens when it does not hold.

If F_{opt} is actually in span($\mathcal{H}$), then equation (12.11) clearly holds. However, here we are making the slightly weaker assumption that F_{opt}'s risk can only be approached, not necessarily attained, by functions in span($\mathcal{H}$). This assumption can be relaxed a bit further by assuming that the smallest risk of functions in span($\mathcal{H}$) is close to, rather than equal to, the optimal (so that equation (12.11) holds only approximately). Our analysis can be applied in this case, yielding asymptotic error bounds that will be correspondingly close to, but different from, the Bayes error.

To simplify the analysis, we regard $\mathcal{H}$ as a fixed space. However, as noted earlier, larger training sets sometimes warrant richer hypothesis spaces. Our analysis will be applicable in this case as well, and will quantify how quickly the hypotheses can increase in complexity as a function of the number of training examples while still admitting convergence to the Bayes optimal.

12.2.2 Proof Overview

Our goal is to show that risk(F_T), the true risk of the function generated by AdaBoost after T rounds, converges to the optimal risk, $\text{risk}^* = \text{risk}(F_{\text{opt}})$. Since F_{opt} may not itself belong to the span of $\mathcal{H}$, we instead focus on comparing F_T's risk with that of a fixed *reference function* $\check{F}$ that is in the span. This will be sufficient for our purposes since, by equation (12.11), $\check{F}$ can itself be chosen to have risk arbitrarily close to risk*.

Our analysis will require that we take into account the norm, or overall magnitude, of the weights defining functions in the span of $\mathcal{H}$, especially the reference functions. If F is in span($\mathcal{H}$), then it can be written in the form

$$F(x) = \sum_{t=1}^{T} \alpha_t h_t(x).$$

We define its *norm*, written $|F|$, to be

$$\sum_{t=1}^{T} |\alpha_t|. \tag{12.12}$$

If the function F can be written in more than one way as a linear combination of base hypotheses, then we define the norm to be the minimum (or infimum) value of equation (12.12) among all such equivalent representations.

Equation (12.11) then implies that there exist, for each $B > 0$, reference functions $\check{F}_B$ in the span of $\mathcal{H}$ such that $|\check{F}_B| < B$, and such that

$$\text{risk}(\check{F}_B) \to \text{risk}^* \tag{12.13}$$

as $B \to \infty$. Thus, if we can show that the function F_T produced by AdaBoost has risk close to that of $\check{F}_B$, then this will also imply risk close to optimal, for an appropriately large choice of B.

An annoyance of working with exponential loss is its unboundedness, that is, the property that $e^{-yF(x)}$ can be unboundedly large. This is particularly a problem when trying to relate the exponential loss on the training set to its true expectation, since a random variable with a very large range is also likely to have high variance, making the estimation of its expectation infeasible from a small sample. This difficulty is reflected, for instance, by Hoeffding's inequality (theorem 2.1), which requires that the random variables be bounded. (An extreme example illustrating the problem is a lottery ticket that pays a million dollars with probability 10^{-6}, and otherwise results in the loss of one dollar. Its expected value is very close to zero, but any sample of reasonable size will almost certainly consist only of losing tickets with an empirical average of -1. The variance of this random variable is about 10^6.)

To sidestep this problem, we will restrict the range of the functions generated by AdaBoost by "clamping" them within a fixed range, thus also limiting the magnitude of the exponential loss. Specifically, for $C > 0$, let us define the function

$$\text{clamp}_C(z) \doteq \begin{cases} C & \text{if } z \geq C \\ z & \text{if } -C \leq z \leq C \\ -C & \text{if } z \leq -C, \end{cases}$$

which simply clamps its argument to the range $[-C, C]$. Next, let us define $\overline{F}_T$ to be the clamped version of F_T:

$$\overline{F}_T(x) \doteq \text{clamp}_C(F_T(x)).$$

Note that the classifications induced by $\overline{F}_T$ are the same as for F_T since

$$\text{sign}(\overline{F}_T(x)) = \text{sign}(F_T(x))$$

always. Therefore, if $\text{sign}(\overline{F}_T)$ converges to the Bayes optimal, then $\text{sign}(F_T)$ does as well. By theorem 12.1, this means that it is sufficient to show that the risk of $\overline{F}_T$ converges to the optimal risk. This, in turn, can be proved using the fact that, on the one hand, $\overline{F}_T$ is bounded, so its empirical risk is close to its true risk; and, on the other hand, the empirical risk of $\overline{F}_T$ is not much worse than that of F_T, which is minimized by the learning algorithm.

So we can now summarize our entire argument in four parts. We will show each of the following, where we use the notation $\lesssim$ to indicate informal, approximate inequality:

1. The empirical exponential loss of the function F_T generated by AdaBoost, an algorithm that provably minimizes this loss, rapidly converges to a value not much worse than that of the reference function $\check{F}_B$; that is,

   $$\widehat{\text{risk}}(F_T) \lesssim \widehat{\text{risk}}(\check{F}_B).$$

2. Clamping does not significantly increase risk, so that

   $$\widehat{\text{risk}}(\overline{F}_T) \lesssim \widehat{\text{risk}}(F_T).$$

3. The empirical risk of the clamped versions of all functions of the form generated by AdaBoost will be close to their true risk, so that

   $$\text{risk}(\overline{F}_T) \lesssim \widehat{\text{risk}}(\overline{F}_T).$$

   This is essentially a uniform-convergence result of the sort seen in chapters 2 and 4.

4. Similarly, the empirical risk of the fixed reference function $\check{F}_B$ will be close to its true risk, so that

   $$\widehat{\text{risk}}(\check{F}_B) \lesssim \text{risk}(\check{F}_B).$$

Combining all four parts along with equation (12.13) will allow us to conclude that

$$\text{risk}(\overline{F}_T) \lesssim \widehat{\text{risk}}(\overline{F}_T) \lesssim \widehat{\text{risk}}(F_T) \lesssim \widehat{\text{risk}}(\check{F}_B) \lesssim \text{risk}(\check{F}_B) \lesssim \text{risk}^*,$$

so that, by theorem 12.1, the error of the corresponding classifier $\text{sign}(\overline{F}_T) = \text{sign}(F_T)$ is also close to the Bayes optimal.

12.2.3 Formal Proofs

In more precise terms, we prove the following theorem which provides a bound on AdaBoost's risk in terms of the risk of the reference function, the number of rounds T, the number of training examples m, and the complexity of the base hypothesis space $\mathcal{H}$ as measured by its VC-dimension d (see section 2.2.3). Note that both the reference function $\check{F}_B$ and the clamping parameter C are used only for the sake of the mathematical argument, and need not be known by the algorithm.

Theorem 12.2 Suppose AdaBoost is run on m random examples from distribution $\mathcal{D}$ for T rounds, producing output F_T, using an exhaustive weak learner and a negation-closed

base hypothesis space $\mathcal{H}$ of VC-dimension d. Let $\check{F}_B$ be a reference function as above. Then for a suitable choice of C defining $\overline{F}_T = \text{clamp}_C(F_T)$, with probability at least $1 - \delta$,

$$\text{risk}(\overline{F}_T) \leq \text{risk}(\check{F}_B) + \frac{2B^{6/5}}{T^{1/5}} + 2\left(\frac{32}{m}\left((T+1)\ln\left(\frac{me}{T+1}\right) + dT\ln\left(\frac{me}{d}\right) + \ln\left(\frac{16}{\delta}\right)\right)\right)^{1/4} + e^B\sqrt{\frac{\ln(4/\delta)}{m}}. \tag{12.14}$$

As shown in the next corollary, this immediately implies convergence to the Bayes optimal as the sample size m gets large, for a suitable number of rounds T. Here, for the moment we add subscripts or superscripts, as in F_T^m, B_m, T_m, etc., to emphasize explicitly the dependence on m. Also, as used in the corollary, an infinite sequence of random variables $X_1, X_2, \ldots$ is said to *converge almost surely* (or *with probability* 1) to some other random variable X, written $X_m \xrightarrow{a.s.} X$, if

$$\mathbf{Pr}\left[\lim_{m\to\infty} X_m = X\right] = 1. \tag{12.15}$$

Corollary 12.3 If, under the conditions of theorem 12.2, we run AdaBoost for $T = T_m = \theta(m^a)$ rounds, where a is any constant in (0, 1), then as $m \to \infty$,

$$\text{risk}(\overline{F}_{T_m}^m) \xrightarrow{a.s.} \text{risk}^*, \tag{12.16}$$

and therefore,

$$\text{err}(H_m) \xrightarrow{a.s.} \text{err}^* \tag{12.17}$$

where $H_m(x) = \text{sign}(F_{T_m}^m(x)) = \text{sign}(\overline{F}_{T_m}^m(x))$.

Proof Before proving the corollary, we make some general remarks concerning the convergence of random variables. Almost sure convergence, as defined in equation (12.15), is equivalent to the condition that for all $\varepsilon > 0$, with probability 1, all of the X_m's come within ε of X, for m sufficiently large; that is,

$$\mathbf{Pr}[\exists n \geq 1, \forall m \geq n : |X_m - X| < \varepsilon] = 1. \tag{12.18}$$

A commonly used tool for proving such convergence is the *Borel-Cantelli lemma*, which states that if $e_1, e_2, \ldots$ is a sequence of events for which

$$\sum_{m=1}^{\infty} \mathbf{Pr}[e_m \text{ does } \textit{not} \text{ hold}] < \infty,$$

then

$$\mathbf{Pr}[\exists n \geq 1, \forall m \geq n : e_m \text{ holds}] = 1.$$

In other words, with probability 1, all of the events e_m hold for m sufficiently large, provided that the sum of the individual probabilities of the events not holding converges to any finite value. Thus, setting e_m to the event that $|X_m - X| < \varepsilon$, we see that to prove equation (12.18), it suffices to show that

$$\sum_{m=1}^{\infty} \mathbf{Pr}[|X_m - X| \geq \varepsilon] < \infty. \tag{12.19}$$

And therefore, to show $X_m \xrightarrow{a.s.} X$, it suffices to show that equation (12.19) holds for all $\varepsilon > 0$. We will apply this technique shortly.

To prove the corollary, we set $B = B_m = (\ln m)/4$, and $\delta = \delta_m = 1/m^2$. With these choices, for every $\varepsilon > 0$, we can choose m so large that

1. the excess risk appearing in equation (12.14)—that is, the amount by which risk($\overline{F}^m_{T_m}$) can exceed risk($\check{F}_{B_m}$)—is smaller than $\varepsilon/2$;
2. risk($\check{F}_{B_m}$) is within $\varepsilon/2$ of risk*.

Together, these imply that for m sufficiently large, the probability that

$$\text{risk}(\overline{F}^m_{T_m}) < \text{risk}^* + \varepsilon$$

is at least $1 - \delta_m$. Since risk$^* \leq$ risk($\overline{F}^m_{T_m}$) always, and since $\sum_{m=1}^{\infty} \delta_m < \infty$, the Borel-Cantelli lemma now implies, by the argument above, that risk($\overline{F}^m_{T_m}$) converges almost surely to risk*, proving equation (12.16). From this, equation (12.17) now follows by a direct application of theorem 12.1. ∎

These results can be generalized to the case in which the complexity of the base hypotheses depends on the number of training examples m simply by regarding the VC-dimension d as a (not too fast-growing) function of m, and adjusting T appropriately.

12.2.4 Bounding How Fast AdaBoost Minimizes Empirical Risk

We now prove theorem 12.2 following the four-part outline given above. We begin with part 1, in which we bound the rate at which AdaBoost minimizes the exponential loss.

Lemma 12.4 After T rounds, the exponential loss of the function F_T generated by AdaBoost satisfies

$$\widehat{\text{risk}}(F_T) \leq \widehat{\text{risk}}(\check{F}_B) + \frac{2B^{6/5}}{T^{1/5}}.$$

Proof We adopt the notation of both algorithms 1.1 (p. 5) and 7.1 (p. 178). Our approach will be to focus on three key quantities, how they relate to one another, and how they evolve over time. The first of these is

$$R_t \doteq \ln\left(\widehat{\text{risk}}(F_t)\right) - \ln\left(\widehat{\text{risk}}(\check{F}_B)\right), \tag{12.20}$$

that is, the difference between the logarithm of the exponential loss attained by AdaBoost after T rounds, and that of the reference function $\check{F}_B$. Our aim is to show that R_t gets small quickly. Note that R_t never increases.

The second quantity of interest is

$$S_t \doteq B + \sum_{t'=1}^{t} \alpha_{t'}, \tag{12.21}$$

which will provide an upper bound on the norms $|\check{F}_B| + |F_t|$. Here and throughout, we assume without loss of generality that the α_t's are all nonnegative (or equivalently, that $\epsilon_t \leq \frac{1}{2}$ for all t), so that S_t never decreases.

And the third quantity that we focus on is the edge $\gamma_t \doteq \frac{1}{2} - \epsilon_t$.

Roughly speaking, our first claim shows that if AdaBoost's exponential loss is large relative to its associated norm, then the edge γ_t must also be large.

Claim 12.5 For $t \geq 1$,

$$R_{t-1} \leq 2\gamma_t S_{t-1}.$$

Proof As usual, D_t is the distribution computed by AdaBoost on round t. Thus, in the present notation,

$$D_t(i) = \frac{\exp\left(-y_i F_{t-1}(x_i)\right)}{m \cdot \widehat{\text{risk}}(F_{t-1})}. \tag{12.22}$$

Let us also define the analogous distribution $\check{D}$ for $\check{F}_B$:

$$\check{D}(i) \doteq \frac{\exp\left(-y_i \check{F}_B(x_i)\right)}{m \cdot \widehat{\text{risk}}(\check{F}_B)}.$$

Since relative entropy, as defined in equations (6.11) and (8.6) and discussed in section 8.1.2, is never negative, we have

$$\begin{aligned} 0 &\leq \text{RE}\left(D_t \;\|\; \check{D}\right) \\ &= \sum_{i=1}^{m} D_t(i) \ln\left(\frac{D_t(i)}{\check{D}(i)}\right) \end{aligned}$$

$$= \ln\left(\widehat{\text{risk}}(\check{F}_B)\right) - \ln\left(\widehat{\text{risk}}(F_{t-1})\right) - \sum_{i=1}^{m} D_t(i) y_i F_{t-1}(x_i) + \sum_{i=1}^{m} D_t(i) y_i \check{F}_B(x_i).$$

That is,

$$R_{t-1} \leq -\sum_{i=1}^{m} D_t(i) y_i F_{t-1}(x_i) + \sum_{i=1}^{m} D_t(i) y_i \check{F}_B(x_i). \tag{12.23}$$

To prove the claim, we bound the two terms on the right.

We have that

$$\begin{aligned} 2\gamma_t &= (1-\epsilon_t) - \epsilon_t \\ &= \sum_{i=1}^{m} D_t(i) y_i h_t(x_i) \\ &= \max_{h \in \mathcal{H}} \sum_{i=1}^{m} D_t(i) y_i h(x_i), \end{aligned}$$

where the last equality uses our assumptions that the weak learner is exhaustive, and that $\mathcal{H}$ is closed under negation. Thus,

$$\begin{aligned} \left|\sum_{i=1}^{m} D_t(i) y_i F_{t-1}(x_i)\right| &= \left|\sum_{i=1}^{m} D_t(i) y_i \sum_{t'=1}^{t-1} \alpha_{t'} h_{t'}(x_i)\right| \\ &= \left|\sum_{t'=1}^{t-1} \alpha_{t'} \sum_{i=1}^{m} D_t(i) y_i h_{t'}(x_i)\right| \\ &\leq \left(\sum_{t'=1}^{t-1} \alpha_{t'}\right) \max_{h \in \mathcal{H}} \left|\sum_{i=1}^{m} D_t(i) y_i h(x_i)\right| \\ &= 2\gamma_t \cdot \sum_{t'=1}^{t-1} \alpha_{t'}. \end{aligned} \tag{12.24}$$

Furthermore, we can write $\check{F}_B$ in the form

$$\check{F}_B(x) = \sum_{j=1}^{n} b_j \hat{h}_j(x)$$

where

$$\sum_{j=1}^{n} |b_j| \leq B, \tag{12.25}$$

and $\hat{h}_1, \ldots, \hat{h}_n$ are in $\mathcal{H}$. Then by a similar argument,

$$\left| \sum_{i=1}^{m} D_t(i) y_i \check{F}_B(x_i) \right| \leq 2\gamma_t \cdot B. \tag{12.26}$$

Combining equations (12.23), (12.24), and (12.26), together with the definition of S_{t-1} in equation (12.21), yields

$$R_{t-1} \leq \left| \sum_{i=1}^{m} D_t(i) y_i F_{t-1}(x_i) \right| + \left| \sum_{i=1}^{m} D_t(i) y_i \check{F}_B(x_i) \right| \leq 2\gamma_t S_{t-1},$$

as claimed. ∎

Next, let us define

$$\Delta R_t \doteq R_{t-1} - R_t, \tag{12.27}$$

$$\Delta S_t \doteq S_t - S_{t-1},$$

the amounts by which R_t *decreases* and S_t *increases* on round t. Note that these are both nonnegative. The next claim shows how these are related and, specifically, how their ratio is controlled by the edge γ_t.

Claim 12.6 For $t \geq 1$,

$$\frac{\Delta R_t}{\Delta S_t} \geq \gamma_t.$$

Proof We can compute ΔR_t exactly as follows:

$$\begin{aligned}
\Delta R_t &= \ln\left(\widehat{\text{risk}}(F_{t-1})\right) - \ln\left(\widehat{\text{risk}}(F_t)\right) \\
&= -\ln\left(\frac{\frac{1}{m}\sum_{i=1}^{m} \exp\left(-y_i F_t(x_i)\right)}{\widehat{\text{risk}}(F_{t-1})}\right) \\
&= -\ln\left(\frac{\frac{1}{m}\sum_{i=1}^{m} \exp\left(-y_i (F_{t-1}(x_i) + \alpha_t h_t(x_i))\right)}{\widehat{\text{risk}}(F_{t-1})}\right) \\
&= -\ln\left(\sum_{i=1}^{m} D_t(i) \exp\left(-\alpha_t y_i h_t(x_i)\right)\right) \\
&= -\tfrac{1}{2}\ln(1 - 4\gamma_t^2),
\end{aligned} \tag{12.28}$$

where the last equality uses equation (3.9) from the analysis of AdaBoost's training error given in theorem 3.1.

We can also obtain an exact expression for ΔS_t from the definition of α_t given in algorithm 1.1:

$$\Delta S_t = \alpha_t = \frac{1}{2} \ln \left(\frac{1+2\gamma_t}{1-2\gamma_t} \right).$$

Combining yields

$$\frac{\Delta R_t}{\Delta S_t} = \frac{-\ln(1-4\gamma_t^2)}{\ln\left(\frac{1+2\gamma_t}{1-2\gamma_t}\right)} \doteq \Upsilon(\gamma_t)$$

where Υ is the same function encountered in section 5.4.1 and defined in equation (5.32). The claim now follows from the fact that $\Upsilon(\gamma) \geq \gamma$ for all $0 \leq \gamma \leq \frac{1}{2}$ (see figure 5.4 (p. 113)). ∎

Together, these claims imply that the quantity $R_t^2 S_t$ never increases, as we show next, which will allow us in turn to relate R_t and S_t directly.

Claim 12.7 For $t \geq 1$, if $R_t \geq 0$, then

$$R_t^2 S_t \leq R_{t-1}^2 S_{t-1}.$$

Proof Combining claims 12.5 and 12.6 gives

$$\frac{2\Delta R_t}{R_{t-1}} \geq \frac{\Delta S_t}{S_{t-1}}. \tag{12.29}$$

Thus,

$$\begin{aligned} R_t^2 S_t &= (R_{t-1} - \Delta R_t)^2 (S_{t-1} + \Delta S_t) \\ &= R_{t-1}^2 S_{t-1} \left(1 - \frac{\Delta R_t}{R_{t-1}}\right)^2 \left(1 + \frac{\Delta S_t}{S_{t-1}}\right) \\ &\leq R_{t-1}^2 S_{t-1} \cdot \exp\left(-\frac{2\Delta R_t}{R_{t-1}} + \frac{\Delta S_t}{S_{t-1}}\right) \qquad (12.30) \\ &\leq R_{t-1}^2 S_{t-1} \qquad (12.31) \end{aligned}$$

where equation (12.30) uses $1 + x \leq e^x$ for all $x \in \mathbb{R}$, and equation (12.31) follows from equation (12.29). ∎

Applying claim 12.7 repeatedly yields (when $R_{t-1} \geq 0$)

$$R_{t-1}^2 S_{t-1} \leq R_0^2 S_0 \leq B^3, \tag{12.32}$$

since $S_0 = B$, and

$$R_0 = -\ln\left(\widehat{\text{risk}}(\check{F}_B)\right) \leq |\check{F}_B| \leq B.$$

Combining equations (12.28) and (12.32), along with claim 12.5, now implies that

$$\Delta R_t = -\frac{1}{2}\ln(1-4\gamma_t^2) \geq 2\gamma_t^2 \geq \frac{1}{2}\left(\frac{R_{t-1}}{S_{t-1}}\right)^2 \geq \frac{1}{2}\left(\frac{R_{t-1}}{B^3/R_{t-1}^2}\right)^2 = \frac{R_{t-1}^6}{2B^6}. \tag{12.33}$$

This shows that if the relative loss is large, then the progress that is made in reducing it will be large as well. The next and last claim shows how this implies an inductive bound on R_t:

Claim 12.8 Let $c = 1/(2B^6)$. If $R_t > 0$, then

$$\frac{1}{R_t^5} \geq \frac{1}{R_{t-1}^5} + 5c. \tag{12.34}$$

Proof Multiplying both sides by R_{t-1}^5 and rearranging terms, equation (12.34) can be rewritten as

$$\left(\frac{R_{t-1}}{R_t}\right)^5 \geq 1 + 5cR_{t-1}^5. \tag{12.35}$$

We have that

$$\begin{aligned}\left(\frac{R_t}{R_{t-1}}\right)^5\left(1+5cR_{t-1}^5\right) &= \left(1-\frac{\Delta R_t}{R_{t-1}}\right)^5\left(1+5cR_{t-1}^5\right)\\ &\leq \exp\left(-\frac{5\Delta R_t}{R_{t-1}} + 5cR_{t-1}^5\right) &(12.36)\\ &\leq 1, &(12.37)\end{aligned}$$

where equation (12.36) uses $1 + x \leq e^x$ for all x, and equation (12.37) follows from equation (12.33). This implies equation (12.35) and the claim. ∎

We can now prove lemma 12.4. If either $R_T \leq 0$ or $T \leq B^6$, then the lemma holds trivially (since $\widehat{\text{risk}}(F_T) \leq 1$), so we assume $R_T > 0$ and $T > B^6$ in what follows. Repeatedly applying claim 12.8 yields

$$\frac{1}{R_T^5} \geq \frac{1}{R_0^5} + 5cT \geq 5cT.$$

Thus,

$$R_T \leq \left(\frac{2B^6}{5T}\right)^{1/5} \leq \frac{B^{6/5}}{T^{1/5}}.$$

That is,

$$\begin{aligned}\widehat{\text{risk}}(F_T) &\le \widehat{\text{risk}}(\check{F}_B) \cdot \exp\left(\frac{B^{6/5}}{T^{1/5}}\right) \\ &\le \widehat{\text{risk}}(\check{F}_B) \cdot \left(1 + \frac{2B^{6/5}}{T^{1/5}}\right) \\ &\le \widehat{\text{risk}}(\check{F}_B) + \frac{2B^{6/5}}{T^{1/5}}\end{aligned}$$

since $e^x \le 1 + 2x$ for $x \in [0, 1]$, and since $\widehat{\text{risk}}(\check{F}_B) \le 1$. ∎

12.2.5 Bounding the Effect of Clamping

Moving on to part 2 of the proof, we show next that the degradation in exponential loss caused by clamping is limited.

Lemma 12.9 For any $F : \mathcal{X} \to \mathbb{R}$ and $C > 0$, let $\overline{F}(x) \doteq \text{clamp}_C(F(x))$. Then $\widehat{\text{risk}}(\overline{F}) \le \widehat{\text{risk}}(F) + e^{-C}$.

Proof Let (x, y) be any labeled pair. If $yF(x) \le C$, then

$$y\overline{F}(x) = \text{clamp}_C(yF(x)) \ge yF(x),$$

so $e^{-y\overline{F}(x)} \le e^{-yF(x)}$. Otherwise, if $yF(x) > C$, then $y\overline{F}(x) = C$, so $e^{-y\overline{F}(x)} = e^{-C}$. In either case, we conclude that

$$e^{-y\overline{F}(x)} \le e^{-yF(x)} + e^{-C}.$$

Therefore,

$$\frac{1}{m}\sum_{i=1}^{m} e^{-y_i\overline{F}(x_i)} \le \frac{1}{m}\sum_{i=1}^{m} e^{-y_iF(x_i)} + e^{-C}$$

as claimed. ∎

12.2.6 Relating Empirical and True Risks

For part 3, we relate the empirical risk to the true risk for all clamped functions of the form produced by AdaBoost. Let $\text{span}_T(\mathcal{H})$ be the subset of $\text{span}(\mathcal{H})$ consisting of all linear combinations of *exactly* T base hypotheses:

$$\text{span}_T(\mathcal{H}) \doteq \left\{ F : x \mapsto \sum_{t=1}^{T} \alpha_t h_t(x) \;\middle|\; \alpha_1, \ldots, \alpha_T \in \mathbb{R}; h_1, \ldots, h_T \in \mathcal{H} \right\}.$$

We wish to show that

$$\text{risk}(\text{clamp}_C(F)) \lesssim \widehat{\text{risk}}(\text{clamp}_C(F)) \tag{12.38}$$

uniformly for all F in $\text{span}_T(\mathcal{H})$, and so in particular for F_T generated by AdaBoost. We prove this in two steps. First, we use techniques developed in chapters 2 and 4 to show that the empirical probability of choosing an example (x, y) for which $yF(x) \leq \theta$ will very likely be close to its true probability, for all F in $\text{span}_T(\mathcal{H})$ and all real θ. We then apply this result to show equation (12.38).

Below, $\mathbf{Pr}_{\mathcal{D}}[\cdot]$ and $\mathbf{E}_{\mathcal{D}}[\cdot]$ denote true probability and expectation, and $\mathbf{Pr}_S[\cdot]$ and $\mathbf{E}_S[\cdot]$ denote empirical probability and expectation.

Lemma 12.10 Assume $m \geq \max\{d, T+1\}$. Then with probability at least $1-\delta$, for all $F \in \text{span}_T(\mathcal{H})$ and for all $\theta \in \mathbb{R}$,

$$\mathbf{Pr}_{\mathcal{D}}[yF(x) \leq \theta] \leq \mathbf{Pr}_S[yF(x) \leq \theta] + \varepsilon \tag{12.39}$$

where

$$\varepsilon = \sqrt{\frac{32}{m}\left((T+1)\ln\left(\frac{me}{T+1}\right) + dT\ln\left(\frac{me}{d}\right) + \ln\left(\frac{8}{\delta}\right)\right)}. \tag{12.40}$$

Proof We apply the general-purpose uniform-convergence results outlined in section 2.2. For each $F \in \text{span}_T(\mathcal{H})$ and each $\theta \in \mathbb{R}$, let us define the subset $A_{F,\theta}$ of $\mathcal{Z} \doteq \mathcal{X} \times \{-1, +1\}$ to be

$$A_{F,\theta} \doteq \{(x, y) \in \mathcal{Z} : yF(x) \leq \theta\}.$$

Let $\mathcal{A}$ be the set of all such subsets:

$$\mathcal{A} \doteq \left\{A_{F,\theta} : F \in \text{span}_T(\mathcal{H}), \theta \in \mathbb{R}\right\}.$$

Proving the lemma then is equivalent to showing that

$$\mathbf{Pr}_{\mathcal{D}}[(x, y) \in A] \leq \mathbf{Pr}_S[(x, y) \in A] + \varepsilon$$

for all $A \in \mathcal{A}$, with high probability. Theorem 2.6 provides a direct means of proving this. To apply the theorem, we need to count the number of "in-out behaviors" induced by sets $A \in \mathcal{A}$, that is, we need to bound the size of

$$\Pi_{\mathcal{A}}(S) \doteq \{\{(x_1, y_1), \ldots, (x_m, y_m)\} \cap A : A \in \mathcal{A}\}$$

for any finite sample $S = \langle (x_1, y_1), \ldots, (x_m, y_m)\rangle$.

Suppose that $\theta \in \mathbb{R}$ and that F is a function of the form

$$F(x) = \sum_{t=1}^{T} \alpha_t h_t(x). \tag{12.41}$$

Then clearly an example (x, y) is in $A_{F,\theta}$ if and only if $yF(x) \leq \theta$, that is, if and only if $G_{F,\theta}(x, y) = -1$ where

$$G_{F,\theta}(x, y) \doteq \text{sign}\,(yF(x) - \theta) = \text{sign}\left(\sum_{t=1}^{T} \alpha_t y h_t(x) - \theta\right). \tag{12.42}$$

(For this proof, we temporarily redefine $\text{sign}(0) \doteq -1$.) This means that each induced subset

$$\{(x_1, y_1), \ldots, (x_m, y_m)\} \cap A_{F,\theta}$$

is in exact one-to-one correspondence with the dichotomies induced by the space $\mathcal{G}$ of all functions $G_{F,\theta}$ of the form given in equation (12.42). (Recall that a dichotomy refers to the behavior, or labeling, induced by a function on the sample S—see section 2.2.3.) Thus, there must be the same number of subsets in $\Pi_{\mathcal{A}}(S)$ as dichotomies on S induced by functions in $\mathcal{G}$. Therefore, we focus now on counting the latter.

Similar to the proof of lemma 4.2, let us fix $h_1, \ldots, h_T$ and define the $(T+1)$-dimensional vectors

$$\mathbf{x}'_i = \langle y_i h_1(x_i), \ldots, y_i h_T(x_i); -1\rangle.$$

Then for any function F as in equation (12.41) and any θ, there must exist a linear threshold function σ on $\mathbb{R}^{T+1}$ such that $G_{F,\theta}(x_i, y_i) = \sigma(\mathbf{x}'_i)$ for all i. (Specifically, the coefficients defining σ are $\langle \alpha_1, \ldots, \alpha_T; \theta\rangle$, whose inner product with $\mathbf{x}'_i$ is exactly $y_i F(x_i) - \theta$.) Lemma 4.1 shows that the class Σ_{T+1} of all such linear threshold functions has VC-dimension $T+1$, which means, by Sauer's lemma (lemma 2.4) and equation (2.12), that the number of dichotomies induced by Σ_{T+1} on the m points $\mathbf{x}'_1, \ldots, \mathbf{x}'_m$ (and thus by $\mathcal{G}$ on S when $h_1, \ldots, h_T$ are fixed) is at most

$$\left(\frac{me}{T+1}\right)^{T+1}.$$

Since $\mathcal{H}$ has VC-dimension d, the number of behaviors of base classifiers $h \in \mathcal{H}$ on S is at most $(me/d)^d$. Therefore, by the same argument used in the proof of lemma 4.5, the number of dichotomies induced by $\mathcal{G}$, and thus $|\Pi_{\mathcal{A}}(S)|$, is at most

$$\left(\frac{me}{T+1}\right)^{T+1}\left(\frac{me}{d}\right)^{dT}.$$

Therefore, this is also a bound on $\Pi_{\mathcal{A}}(m)$, the largest value of $|\Pi_{\mathcal{A}}(S)|$ on any sample S of size m.

Plugging into theorem 2.6 now gives the claimed result. ∎

We can now prove equation (12.38).

Lemma 12.11 Let $C > 0$, and assume $m \geq \max\{d, T+1\}$. With probability at least $1-\delta$, for all $F \in \text{span}_T(\mathcal{H})$,

$$\text{risk}(\text{clamp}_C(F)) \leq \widehat{\text{risk}}(\text{clamp}_C(F)) + e^C \cdot \varepsilon$$

where ε is as in equation (12.40).

Proof We assume equation (12.39) holds for all $F \in \text{span}_T(\mathcal{H})$ and all $\theta \in \mathbb{R}$. By lemma 12.10, this will be so with probability at least $1-\delta$. Let $\overline{F}(x) \doteq \text{clamp}_C(F(x))$.

Mapping equation (12.39) to the loss function of interest, we claim first that

$$\mathbf{Pr}_{\mathcal{D}}\left[e^{-y\overline{F}(x)} \geq \theta\right] \leq \mathbf{Pr}_S\left[e^{-y\overline{F}(x)} \geq \theta\right] + \varepsilon \tag{12.43}$$

for all θ. For if $e^{-C} \leq \theta \leq e^C$, then $e^{-y\overline{F}(x)} \geq \theta$ if and only if $yF(x) \leq \ln\theta$, so that equation (12.43) follows from equation (12.39). If $\theta > e^C$, then both the true and the empirical probabilities appearing in equation (12.43) are equal to zero; likewise, if $\theta < e^{-C}$, then they are both equal to 1. In either case, equation (12.43) holds trivially.

It is known that the expected value of any random variable X with range $[0, M]$ can be computed by integrating the complement of its cumulative distribution function. That is,

$$\mathbf{E}[X] = \int_0^M \mathbf{Pr}[X \geq \theta]\, d\theta.$$

Thus, applying equation (12.43) and the fact that $e^{-y\overline{F}(x)}$ cannot exceed e^C gives

$$\begin{aligned}
\text{risk}(\overline{F}) &= \mathbf{E}_{\mathcal{D}}\left[e^{-y\overline{F}(x)}\right] \\
&= \int_0^{e^C} \mathbf{Pr}_{\mathcal{D}}\left[e^{-y\overline{F}(x)} \geq \theta\right] d\theta \\
&\leq \int_0^{e^C} \left(\mathbf{Pr}_S\left[e^{-y\overline{F}(x)} \geq \theta\right] + \varepsilon\right) d\theta \\
&= \mathbf{E}_S\left[e^{-y\overline{F}(x)}\right] + e^C \cdot \varepsilon \\
&= \widehat{\text{risk}}(\overline{F}) + e^C \cdot \varepsilon,
\end{aligned}$$

as claimed. ∎

Part 4 of the proof is comparatively simple since we only need to show that the single function $\check{F}_B$ is likely to have empirical risk close to its true risk.

Lemma 12.12 With probability at least $1-\delta$,

$$\widehat{\text{risk}}(\check{F}_B) \leq \text{risk}(\check{F}_B) + e^B\sqrt{\frac{\ln(2/\delta)}{m}}.$$

Proof Consider the random variables $\exp(-y_i\check{F}_B(x_i) - B)$, whose average is $e^{-B}\,\widehat{\text{risk}}(\check{F}_B)$, and whose expectation is $e^{-B}\,\text{risk}(\check{F}_B)$. Because $|\check{F}_B| \leq B$ and the hypotheses in $\mathcal{H}$ are binary, $|y_i\check{F}_B(x_i)| \leq B$, so that these random variables are in $[0, 1]$. Applying Hoeffding's inequality (theorem 2.1) now gives

$$e^{-B}\,\widehat{\text{risk}}(\check{F}_B) \leq e^{-B}\,\text{risk}(\check{F}_B) + \sqrt{\frac{\ln(2/\delta)}{m}}$$

with probability at least $1-\delta$. ∎

12.2.7 Finishing the Proof

We can now complete the proof of theorem 12.2 by combining lemmas 12.4 and 12.9 (applied to F_T) as well as lemmas 12.11 and 12.12. Together with the union bound, these give, with probability at least $1-2\delta$, that

$$\text{risk}(\overline{F}_T) \leq \text{risk}(\check{F}_B) + B\sqrt{\frac{\ln T}{T}} + e^{-C} + e^C \cdot \varepsilon + e^B\sqrt{\frac{\ln(2/\delta)}{m}} \tag{12.44}$$

where ε is as in equation (12.40). Replacing δ with $\delta/2$, and choosing C to minimize equation (12.44), completes the proof of theorem 12.2.

12.2.8 Comparison to Margin-Based Bounds

As the amount of training data increases, the foregoing shows that the classification accuracy of AdaBoost converges to optimality, provided the base hypotheses possess the right degree of expressiveness. This guarantee is absolute, in contrast to the generalization-error bounds given in section 5.2, which are in terms of the margins as measured on the dataset *following* training. Moreover, the current analysis does not depend on the weak learning assumption, and so is applicable even if the edges of the weak hypotheses are rapidly approaching zero.

On the other hand, the analysis given in this chapter, as in chapter 4, requires that the number of rounds T be controlled and kept significantly smaller than the training set size m (but also large enough for the algorithm to approach minimal exponential loss). In other words, the analysis predicts overfitting if the algorithm is run for too long. In this way, the analysis fails to explain the cases in which AdaBoost manages to avoid overfitting, unlike the margins analysis whose bounds are entirely independent of T.

In short, the margins theory seems to better capture AdaBoost's behavior when the weak learning assumption holds, for instance, when using a reasonably strong base learner, like a decision-tree algorithm, that does indeed generate base hypotheses that are consistently

and significantly better than random. In this case, by the results of section 5.4.1, we can expect large margins and an accompanying resistance to overfitting. When, due to noise or randomness in the data, the weak learning assumption does not hold without an inordinate blowup in the complexity of the base hypotheses, the current analysis shows that boosting can still be used—though in a mode requiring somewhat greater control—to deliver results comparable to the best possible.

The analysis of the generalization error given in this chapter was based on minimization of exponential loss. On the other hand, in section 7.3 we saw that this property alone is not sufficient to guarantee good generalization, and that any analysis must also take into account *how* the algorithm minimizes loss, as is done in the margins-based analysis of AdaBoost. These results are not in contradiction. On the contrary, the current analysis is very much based on the manner in which AdaBoost is able to generate a predictor with nearly minimal exponential loss by combining a relatively small number of base hypotheses.

12.3 How Minimizing Risk Can Lead to Poor Accuracy

Corollary 12.3 depends crucially on the key assumption that the minimum exponential loss can be realized or approached by linear combinations of base hypotheses as stated formally in equation (12.11). When this assumption does not hold, AdaBoost may produce a combined classifier whose performance is extremely poor relative to the Bayes optimal. This is true even though the base hypotheses may be rich enough to represent the Bayes optimal classifier as a linear threshold function, and even with unlimited training data, and even if the noise affecting the data is of a very simple form. Moreover, the difficulty applies to any algorithm that minimizes exponential loss, including AdaBoost.

To see this, we construct a simple example of a distribution $\mathcal{D}$ over labeled pairs, and a base hypothesis space $\mathcal{H}$ for which the linear combination of base hypotheses with minimum exponential loss induces a classifier with accuracy as bad as random guessing, even though the Bayes optimal classifier can be represented by just such a linear combination.

12.3.1 A Construction Using Confidence-Rated Hypotheses

In this construction, the instance space $\mathcal{X}$ consists of just three instances: the "large-margin" example, x_{lm}; the "puller," x_{pu}; and the "penalizer," x_{pe}. (The meaning of the names will become apparent later.) To generate a labeled example (x, y) according to $\mathcal{D}$, we first randomly choose x to be equal to x_{lm} with probability $\frac{1}{4}$; x_{pu} with probability $\frac{1}{4}$; and x_{pe} with probability $\frac{1}{2}$. The label y is chosen independently of x to be $+1$ with probability $1-\eta$, and -1 with probability η, where $0 < \eta < \frac{1}{2}$ is the fixed noise rate. Thus, it is as if the "true" label of each example, which in this case is always $+1$, is flipped to its opposite value -1 with probability η prior to being observed by the learner. Such a *uniform noise* model, which affects the true labels of all examples with equal probability, is perhaps the simplest possible model of noise.

The hypothesis space $\mathcal{H}$ consists of just two hypotheses: $\hbar_1$ and $\hbar_2$. Here, as in chapter 9, we allow these to be real-valued or confidence-rated. Later, we show how the same construction can be modified for binary classifiers. The hypotheses $\hbar_1$ and $\hbar_2$ are defined as follows:

x	$\hbar_1(x)$	$\hbar_2(x)$
x_{lm}	1	0
x_{pe}	c	$-\frac{1}{5}$
x_{pu}	c	1

where $c > 0$ is a small constant to be chosen later. In fact, our argument will hold for all sufficiently small (but positive) values of c. The hypotheses in $\mathcal{H}$ can be plotted geometrically as in figure 12.1.

Note that the Bayes optimal classifier for the distribution $\mathcal{D}$ predicts that all instances are positive, incurring a Bayes error rate of exactly η. This classifier can be represented as the sign of a (trivial) linear combination of base hypotheses, namely, $\text{sign}(\hbar_1(x))$.

In minimizing exponential loss, we aim to find a linear combination of $\hbar_1$ and $\hbar_2$,

$$F_{\boldsymbol{\lambda}}(x) \doteq \lambda_1 \hbar_1(x) + \lambda_2 \hbar_2(x),$$

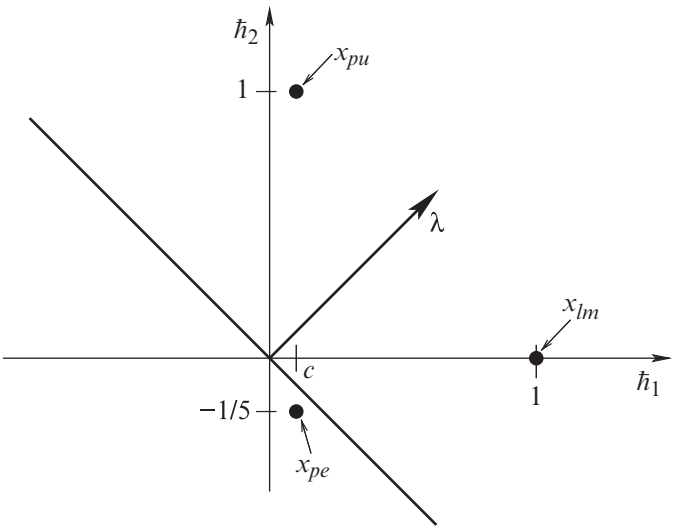

Figure 12.1
A plot of the hypotheses $\hbar_1$ and $\hbar_2$ on the instances x_{lm}, x_{pe}, and x_{pu}. Each instance x is represented by the point $\langle \hbar_1(x), \hbar_2(x) \rangle$. The vector $\boldsymbol{\lambda}$ schematically depicts the coefficients on $\hbar_1$ and $\hbar_2$ obtained by minimizing the exponential loss. The line perpendicular to $\boldsymbol{\lambda}$ represents the resulting decision boundary, which, in this case, predicts x_{pe} to be negative, and the other two instances to be positive.

that minimizes risk($F_{\boldsymbol{\lambda}}$) as defined in equation (12.2). We consider an ideal situation in which the true risk with respect to $\mathcal{D}$ is minimized directly, as will be the case in the limit of a very large training set for an algorithm like AdaBoost (if run for enough rounds). Our aim now is to show that the resulting classifier sign($F_{\boldsymbol{\lambda}}$) will have very poor accuracy.

Let us define

$$K(z) \doteq (1-\eta)e^{-z} + \eta e^{z}. \tag{12.45}$$

Then by construction of $\mathcal{D}$ and $\mathcal{H}$, we can write out the risk of $F_{\boldsymbol{\lambda}}$ explicitly as

$$L(\boldsymbol{\lambda}, c) \doteq \text{risk}(F_{\boldsymbol{\lambda}}) = \tfrac{1}{4}K(\lambda_1) + \tfrac{1}{2}K\left(c\lambda_1 - \tfrac{1}{5}\lambda_2\right) + \tfrac{1}{4}K\left(c\lambda_1 + \lambda_2\right), \tag{12.46}$$

where the three terms on the right correspond, respectively, to the expected loss associated with x_{lm}, x_{pe}, and x_{pu}. With c fixed, the vector $\boldsymbol{\lambda}$ is chosen to minimize this expression. Intuitively, when c is small, λ_1 is controlled almost entirely by x_{lm}, while λ_2 is controlled by the other two instances. In particular, the puller will tend to pull λ_2 in a strongly positive direction since it turns out that $\hbar_2$'s higher-confidence prediction on the puller more than offsets the higher weight assigned to the penalizer under the distribution $\mathcal{D}$. As a result, the penalizer will be predicted negative, as seen in figure 12.1. If this happens, then the overall error of the resulting classifier will be at least $\frac{1}{2}$ because of the penalizer's large weight under $\mathcal{D}$.

More formally, we prove the following:

Theorem 12.13 Given the construction described above, let $\boldsymbol{\lambda}^*(c)$ be any value of $\boldsymbol{\lambda}$ that minimizes the exponential loss $L(\boldsymbol{\lambda}, c)$. Then for any sufficiently small value of $c > 0$, the classification error of sign($F_{\boldsymbol{\lambda}^*(c)}$) is at least $\frac{1}{2}$. On the other hand, for some other choice of $\boldsymbol{\lambda}$, the classification error of sign($F_{\boldsymbol{\lambda}}$) is equal to the Bayes error rate of η.

Proof Because $K(z)$ is convex and unbounded as z tends to $\pm\infty$, and because $L(\boldsymbol{\lambda}^*(c), c) \leq L(\mathbf{0}, c) = 1$, it can be argued that the vectors $\boldsymbol{\lambda}^*(c)$, for all $c \in [0, 1]$, must all lie in a bounded subset of $\mathbb{R}^2$; without loss of generality, this subset is also closed, and therefore compact.

When $c = 0$,

$$L(\boldsymbol{\lambda}, 0) = \tfrac{1}{4}K(\lambda_1) + \tfrac{1}{2}K\left(-\tfrac{1}{5}\lambda_2\right) + \tfrac{1}{4}K(\lambda_2).$$

By the results of section 9.2.1, the minimizer $\boldsymbol{\lambda}^*(0)$ of this expression is unique. Moreover, because its derivative with respect to λ_2, $\partial L(\boldsymbol{\lambda}, 0)/\partial\lambda_2$, is strictly negative when $\lambda_2 = 0$, and because L (as a function only of λ_2) is convex, the minimizing value $\lambda_2^*(0)$ must be strictly positive.

We claim that $\boldsymbol{\lambda}^*(c)$ converges to $\boldsymbol{\lambda}^*(0)$ as c converges to 0 (from the right). If it does not, then there exist $\varepsilon > 0$ and a sequence $c_1, c_2, \ldots$ such that $0 < c_n < 1/n$, and

$$\|\boldsymbol{\lambda}^*(c_n) - \boldsymbol{\lambda}^*(0)\| > \varepsilon \tag{12.47}$$

for all n. Because the $\boldsymbol{\lambda}^*(c_n)$'s lie in a compact space, the sequence must have a convergent subsequence; without loss of generality, let that subsequence be the entire sequence so that $\boldsymbol{\lambda}^*(c_n) \to \tilde{\boldsymbol{\lambda}}$ for some $\tilde{\boldsymbol{\lambda}}$. By definition of $\boldsymbol{\lambda}^*(c_n)$ as a minimizer,

$$L(\boldsymbol{\lambda}^*(c_n), c_n) \leq L(\boldsymbol{\lambda}^*(0), c_n)$$

for all n. Taking limits, this implies by the continuity of L that

$$L(\tilde{\boldsymbol{\lambda}}, 0) = \lim_{n\to\infty} L(\boldsymbol{\lambda}^*(c_n), c_n) \leq \lim_{n\to\infty} L(\boldsymbol{\lambda}^*(0), c_n) = L(\boldsymbol{\lambda}^*(0), 0).$$

But because $\boldsymbol{\lambda}^*(0)$ is the unique minimizer of $L(\boldsymbol{\lambda}, 0)$, this means that $\tilde{\boldsymbol{\lambda}} = \boldsymbol{\lambda}^*(0)$, which contradicts equation (12.47). Therefore, $\boldsymbol{\lambda}^*(c) \to \boldsymbol{\lambda}^*(0)$, as claimed.

For any c, the resulting prediction on the penalizer x_{pe} will be the sign of

$$F_{\boldsymbol{\lambda}^*(c)}(x_{pe}) = c\lambda_1^*(c) - \tfrac{1}{5}\lambda_2^*(c),$$

which, in the limit $c \to 0$, is equal to $-\frac{1}{5}\lambda_2^*(0) < 0$ by the arguments above. Thus, for c sufficiently small, x_{pe} will be predicted to be negative, giving an overall error with respect to $\mathcal{D}$ of at least $\frac{1}{2}$. This is as bad as random guessing, and much worse than the Bayes error of η, which, as observed earlier, is realized by a trivial combination of the two base hypotheses. ∎

12.3.2 A Modified Construction Using Binary Classifiers

This construction can be modified so that all of the weak hypotheses are in fact binary classifiers with range $\{-1, +1\}$. In other words, for some distribution $\mathcal{D}$ over examples, and for some space of binary base classifiers, minimizing exponential loss over linear combinations of classifiers from this hypothesis space results in a classifier with accuracy as poor as random guessing, despite the existence of another linear combination of these same base classifiers whose performance matches the Bayes optimal.

To show this, we now represent instances by binary vectors $\mathbf{x}$ in $\mathcal{X} \doteq \{-1, +1\}^N$, where $N \doteq 2n + 11$, and where $n > 0$ will be chosen shortly. The base classifiers in $\mathcal{H}$ are each identified with a component of $\mathbf{x}$; that is, for each component j, there is a base classifier $\hbar_j$ for which $\hbar_j(\mathbf{x}) = x_j$ for every instance $\mathbf{x}$.

We will find it convenient to decompose every instance $\mathbf{x}$ into its first $2n + 1$ components, denoted $\mathbf{x}^{[1]}$, and its remaining 10 components, denoted $\mathbf{x}^{[2]}$. Thus, $\mathbf{x} = \langle \mathbf{x}^{[1]}; \mathbf{x}^{[2]} \rangle$ where $\mathbf{x}^{[1]} \in \{-1, +1\}^{2n+1}$ and $\mathbf{x}^{[2]} \in \{-1, +1\}^{10}$. Roughly speaking, this decomposition will correspond to the two base hypotheses $\hbar_1$ and $\hbar_2$ used in the construction of section 12.3.1.

Let $\mathcal{S}_k^p$ denote the set of p-dimensional binary vectors whose components add up to exactly k:

$$\mathcal{S}_k^p \doteq \left\{ \mathbf{u} \in \{-1, +1\}^p : \sum_{j=1}^{p} u_j = k \right\}.$$

For instance, $\mathcal{S}_p^p$ consists only of the all $+1$'s vector, while $\mathcal{S}_0^p$ consists of all p-dimensional vectors with an exactly equal number of $+1$'s and -1's.

The distribution $\mathcal{D}$ can now be described in terms of these sets. Specifically, a random instance $\mathbf{x} = \langle \mathbf{x}^{[1]}; \mathbf{x}^{[2]} \rangle$ is generated under $\mathcal{D}$ as follows:

- With probability $\frac{1}{4}$, a "large-margin" instance is chosen by selecting $\mathbf{x}^{[1]}$ uniformly at random from $\mathcal{S}_{2n+1}^{2n+1}$ and $\mathbf{x}^{[2]}$ uniformly from $\mathcal{S}_0^{10}$.
- With probability $\frac{1}{2}$, a "penalizer" instance is chosen with $\mathbf{x}^{[1]}$ selected uniformly from $\mathcal{S}_1^{2n+1}$ and $\mathbf{x}^{[2]}$ from $\mathcal{S}_{-2}^{10}$.
- With probability $\frac{1}{4}$, a "puller" instance is chosen with $\mathbf{x}^{[1]}$ selected uniformly from $\mathcal{S}_1^{2n+1}$ and $\mathbf{x}^{[2]}$ from $\mathcal{S}_{10}^{10}$.

The label y is selected just as before to be $+1$ with probability $1 - \eta$ and -1 otherwise. Thus, as before, the Bayes error is η, and now the Bayes optimal classifier can be represented by the majority vote of the components of $\mathbf{x}^{[1]}$.

As was done for instances, we also decompose every weight vector $\boldsymbol{\lambda} \in \mathbb{R}^N$ into $\langle \boldsymbol{\lambda}^{[1]}; \boldsymbol{\lambda}^{[2]} \rangle$, where $\boldsymbol{\lambda}^{[1]} \in \mathbb{R}^{2n+1}$ and $\boldsymbol{\lambda}^{[2]} \in \mathbb{R}^{10}$. A linear combination of weak classifiers thus has the form

$$F_{\boldsymbol{\lambda}}(\mathbf{x}) \doteq \sum_{j=1}^{N} \lambda_j x_j = \sum_{j=1}^{2n+1} \lambda_j^{[1]} x_j^{[1]} + \sum_{j=1}^{10} \lambda_j^{[2]} x_j^{[2]}. \tag{12.48}$$

Its risk with respect to $\mathcal{D}$ is

$$\begin{aligned} \text{risk}(F_{\boldsymbol{\lambda}}) &\doteq \mathbf{E}_{\mathcal{D}}[\exp(-y F_{\boldsymbol{\lambda}}(\mathbf{x}))] \\ &= \sum_{\mathbf{x}, y} \mathcal{D}(\mathbf{x}, y) \exp\left(-y \left(\sum_{j=1}^{2n+1} \lambda_j^{[1]} x_j^{[1]} + \sum_{j=1}^{10} \lambda_j^{[2]} x_j^{[2]} \right) \right) \end{aligned} \tag{12.49}$$

where the outer sum is over all labeled pairs $(\mathbf{x}, y)$ in $\mathcal{X} \times \{-1, +1\}$.

We claim that when this risk is minimized, all of the $\lambda_j^{[1]}$'s are necessarily equal to one another, as are all of the $\lambda_j^{[2]}$'s. Suppose this is not the case, and that $\boldsymbol{\lambda} = \langle \boldsymbol{\lambda}^{[1]}; \boldsymbol{\lambda}^{[2]} \rangle$ minimizes equation (12.49) with $\lambda_1^{[1]} \neq \lambda_2^{[1]}$. Holding all of the other parameters $\lambda_3^{[1]}, \lambda_4^{[1]}, \ldots, \lambda_{2n+1}^{[1]}$, and $\boldsymbol{\lambda}^{[2]}$ fixed, and treating these as constants, we see that every term appearing in the sum in equation (12.49) has the form

$$a \exp\left(b_1 \lambda_1^{[1]} + b_2 \lambda_2^{[1]} \right)$$

for some $b_1, b_2 \in \{-1, +1\}$, and some $a \geq 0$. Combining terms with the same exponent, the risk, as a function of $\lambda_1^{[1]}$ and $\lambda_2^{[1]}$, thus must have the form

$$A e^{\lambda_1^{[1]} - \lambda_2^{[1]}} + A' e^{\lambda_2^{[1]} - \lambda_1^{[1]}} + B e^{\lambda_1^{[1]} + \lambda_2^{[1]}} + C e^{-\lambda_1^{[1]} - \lambda_2^{[1]}} \tag{12.50}$$

where A, A', B, and C are nonnegative, and do not depend on $\lambda_1^{[1]}$ or $\lambda_2^{[1]}$; in fact, by the manner in which the distribution $\mathcal{D}$ was constructed, all four of these must be strictly positive. Moreover, because of the natural symmetry of the distribution, the probability of a labeled example $(\mathbf{x}, y)$ under $\mathcal{D}$ is unchanged by swapping the values of $x_1^{[1]}$ and $x_2^{[1]}$. This implies that $A = A'$. But then replacing $\lambda_1^{[1]}$ and $\lambda_2^{[1]}$ with their average $(\lambda_1^{[1]} + \lambda_2^{[1]})/2$ in equation (12.50) leads to a strictly smaller risk since $\lambda_1^{[1]} \neq \lambda_2^{[1]}$ (and since $e^z + e^{-z}$ is minimized uniquely when $z = 0$). This is a contradiction.

By similar arguments, at the minimizer of the risk, $\lambda_1^{[1]} = \lambda_j^{[1]}$ and $\lambda_1^{[2]} = \lambda_j^{[2]}$ for every component j; that is,

$$\lambda_1^{[1]} = \lambda_2^{[1]} = \cdots = \lambda_{2n+1}^{[1]} = \lambda^{[1]}$$

and

$$\lambda_1^{[2]} = \lambda_2^{[2]} = \cdots = \lambda_{10}^{[2]} = \lambda^{[2]}$$

for some common values $\lambda^{[1]}$ and $\lambda^{[2]}$. Thus, henceforth, we need consider vectors $\boldsymbol{\lambda}^{[1]}$ and $\boldsymbol{\lambda}^{[2]}$ of only this form.

Note that if $\mathbf{x}^{[1]} \in \mathcal{S}_{k_1}^{2n+1}$ and $\mathbf{x}^{[2]} \in \mathcal{S}_{k_2}^{10}$, then by equation (12.48),

$$F_{\boldsymbol{\lambda}}(\mathbf{x}) = \lambda^{[1]} k_1 + \lambda^{[2]} k_2.$$

Thus, by the construction of $\mathcal{D}$, equation (12.49) now simplifies to

$$\tfrac{1}{4} K\left((2n+1)\lambda^{[1]}\right) + \tfrac{1}{2} K\left(\lambda^{[1]} - 2\lambda^{[2]}\right) + \tfrac{1}{4} K\left(\lambda^{[1]} + 10\lambda^{[2]}\right) \tag{12.51}$$

where the three terms correspond to large-margin, penalizer, and puller instances, respectively, and where K was defined in equation (12.45). If we now define

$$\tilde{\lambda}_1 \doteq (2n+1)\lambda^{[1]},$$

$$\tilde{\lambda}_2 \doteq 10\lambda^{[2]},$$

$$\tilde{c} \doteq \frac{1}{2n+1},$$

then equation (12.51) can be written

$$\tfrac{1}{4} K\left(\tilde{\lambda}_1\right) + \tfrac{1}{2} K\left(\tilde{c}\tilde{\lambda}_1 - \tfrac{1}{5}\tilde{\lambda}_2\right) + \tfrac{1}{4} K\left(\tilde{c}\tilde{\lambda}_1 + \tilde{\lambda}_2\right),$$

which has the identical form as equation (12.46), the risk for the construction of section 12.3.1. In other words, we have reduced the minimization problem involving binary classifiers to our previous, simpler construction involving real-valued base hypotheses. Thus, we can now proceed exactly as before to show that for n sufficiently large (so that $\tilde{c}$ is sufficiently small), all of the penalizer examples will be classified -1 by the classifier induced by minimizing the risk, which therefore will have generalization error at least $\frac{1}{2}$.

So we conclude that AdaBoost's classification error can be much worse than optimal if the weak hypothesis space is not adequately expressive. In addition, in section 7.5.3, we described a technique for estimating the conditional probability of an instance being positive or negative. As pointed out in that section, this method relies on essentially the same assumption of expressiveness as given in equation (12.11). The example given above shows that this assumption is indispensable, and that the technique can fail badly without it. With suitable modifications, the same argument can be applied to logistic regression as well (see exercise 12.9).

Experiments based on the construction above are reported in section 14.4.

12.3.3 The Difficulty of Uniform Noise

In the preceding example, we utilized the simple uniform-noise model in which all labels of all instances are corrupted with the same probability $\eta > 0$. The results show that even a small positive value of η will cause the generalization error to be as bad as random guessing, despite the fact that with no noise ($\eta = 0$), an algorithm like AdaBoost will provably generate a classifier with perfect generalization accuracy (given enough training data). So from $\eta = 0$ to $\eta > 0$, the generalization error jumps abruptly from 0 to 50%.

Although contrived, this suggests that AdaBoost may be quite susceptible to such uniform noise. Indeed, experiments have shown this to be the case. For instance, in one empirical study, boosting was compared with bagging (another method of generating and combining base classifiers—see section 5.5) using a decision-tree algorithm as base learner. Among nine real-world benchmark datasets, boosting outperformed bagging significantly on five, while bagging did not beat boosting on even one (on the other four, there was no statistically significant difference). However, when artificial uniform-noise was added at a rate of 10%, the results were reversed: bagging performed better than boosting on six of the datasets, while boosting did better on just one (with a statistical tie occurring on the other two).

While we expect any algorithm to do worse on noisy data, these results show that the degradation in performance for boosting is faster than for other algorithms. Intuitively, this poor performance seems to be a consequence of AdaBoost's deliberate concentration on "hard" examples, a propensity that leads the algorithm to pile ever more weight onto the corrupted examples in a futile effort to match the noisy labels. An example is shown in figure 12.2. This was seen also in section 10.3, where this tendency was exploited beneficially as a means of identifying outliers.

The example constructed above suggests that a second factor affecting performance on noisy data may be an inability to represent the function minimizing the exponential loss using a linear combination of base classifiers of limited complexity.

Although AdaBoost degrades disappointingly with the addition of uniform noise in such semi-artificial experiments, it has also been observed that AdaBoost performs quite well on a wide array of real-world datasets. Such data is almost never "clean," having been corrupted

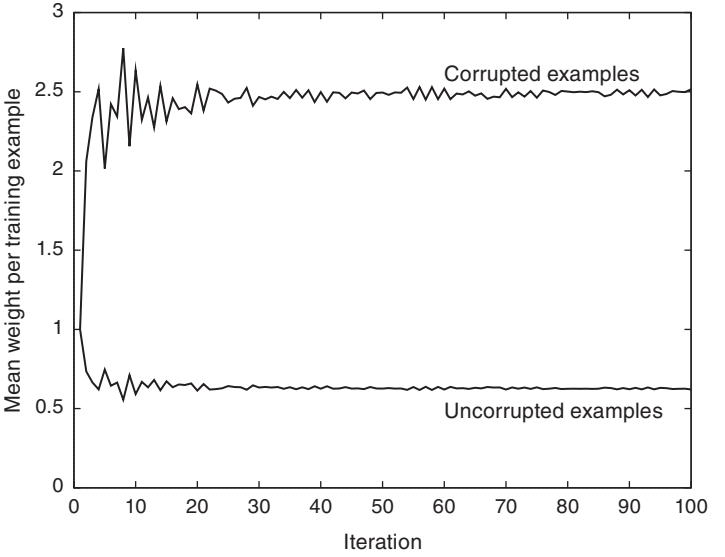

Figure 12.2
In this experiment, prior to training, 20% of the 2800 examples comprising this benchmark dataset were selected at random and their labels artificially corrupted. AdaBoost was then run using a decision-tree algorithm as the base learner. The graph shows the average weight, on each round, placed on the corrupted examples, compared with the weight placed on those that were left uncorrupted. (Copyright ©2000 Kluwer Academic Publishers (now Springer). Reprinted from figure 9 of [68] with permission from Springer Science and Business Media, LLC.)

in one way or another by measurement or recording errors, mislabelings, deletions, and so on. This paradox suggests that perhaps *uniform* noise is a poor model of the real-world influences that lead to the corruption of data. Perhaps, in real datasets, noise does not affect all instances equally, but instead affects instances close to the boundary that separates positives from negatives more strongly than those that are far from this boundary. Indeed, as discussed in section 7.5.1, logistic regression, a close relative of AdaBoost, posits just such a noise model, further hinting at the poor fit between such methods and uniform noise.

On the other hand, there may be an opportunity here to substantially improve AdaBoost's ability to handle noise. Indeed, a number of such algorithms have been suggested. Of those that are provably resistant to uniform noise, most are based on the construction of a very different kind of combined classifier; rather than assembling a final classifier that computes a (weighted) majority vote of the base classifiers, these methods instead construct a *branching program*, a computational structure that is much like a decision tree (see section 1.3), but in which two or more outgoing edges can lead to the same node. Thus, rather than forming a tree, the graph structure of a branching program forms a directed acyclic graph. A full description of such methods is beyond the scope of this book.

An alternative approach for making boosting resistant to noise and outliers will emerge from the theoretical study of optimal boosting given in chapter 13.

Summary

In this chapter, we have identified conditions under which AdaBoost provably converges to the best possible accuracy. This was proved using the algorithm's ability to minimize exponential loss, together with a proof that nearly minimal exponential loss implies nearly optimal classification accuracy. But we also saw in this chapter that AdaBoost's performance can be very poor when the weak hypotheses are insufficiently expressive, even with effectively unlimited training data. The uniform noise assumed in this example, though perhaps not entirely realistic, seems to be a problem for boosting, both theoretically and empirically.

Bibliographic Notes

Results on the consistency of AdaBoost and its variants have been studied under various conditions by a number of authors, including Breiman [38], Mannor, Meir, and Zhang [164], Jiang [128], Lugosi and Vayatis [161], Zhang [235], Zhang and Yu [236], and Bickel, Ritov, and Zakai [20]. The development given in sections 12.1 and 12.2 directly follows the proof of Bartlett and Traskin [14], with some modifications, the most significant being the improved rate of convergence given in lemma 12.4 which is due to Mukherjee, Rudin, and Schapire [172]. Theorem 12.1 was essentially proved by Zhang [235] and, in the slightly more refined form given here, by Bartlett, Jordan, and McAuliffe [12].

The example and proof given in section 12.3.1 are due to Long and Servedio [159], with some modifications and simplifications. Figure 12.1 is adapted from their paper as well. Their results show further that boosting with exponential loss will fail even when using certain forms of regularization, or when boosting is stopped early after only a limited number of rounds. The example in section 12.3.2 was inspired by one that they had used in their paper, but only experimentally and without proof.

Most of the works from the literature mentioned up to this point are applicable to broad and general classes of loss functions, not just exponential loss as presented here.

The experiments mentioned in section 12.3.3 that compare boosting and bagging with noisy data were reported by Dietterich [68]. Figure 12.2 was reprinted, with permission, from this work. See also Maclin and Opitz [162].

Algorithms for boosting in the presence of noise are given by Kalai and Servedio [129], and by Long and Servedio [157, 158]. These utilize an approach to boosting originally due to Mansour and McAllester [165], based on a branching-program representation. Other practical and theoretical research on boosting with various kinds of noise include [9, 17, 106, 141, 143, 186, 210].

Some of the exercises in this chapter are based on material from [12, 38, 159, 161, 172, 236].

Exercises

12.1 Regarding the proof of theorem 12.1, verify that:

a. ϕ is convex.

b. ϕ is strictly increasing.

c. $\phi^{-1}(z)$ is as given in equation (12.10), and is increasing.

12.2 This exercise generalizes theorem 12.1. Let $\ell : \mathbb{R} \to \mathbb{R}_+$ be a margin-based loss function with the following properties: (1) ℓ is convex; and (2) the derivative ℓ' of ℓ exists at 0 and is negative, that is, $\ell'(0) < 0$. Note that these properties together imply that ℓ is decreasing on $(-\infty, 0]$.

We use the notation in section 12.1, but redefine certain key quantities in terms of ℓ. In particular, h_{opt} and err^* are exactly as before, but for $F : \mathcal{X} \to \mathbb{R}$, we redefine

$$\text{risk}(F) \doteq \mathbf{E}[\ell(yF(x))]$$

and

$$\text{risk}^* \doteq \inf_F \text{risk}(F),$$

where the infimum is taken over all possible functions F. Further, for $p \in [0, 1]$ and $z \in \mathbb{R}$, let

$$C(p, z) \doteq p\ell(z) + (1 - p)\ell(-z),$$

and let $C_{\min}(p) \doteq \inf_{z \in \mathbb{R}} C(p, z)$.

As in theorem 12.1, we now let $F : \mathcal{X} \to \mathbb{R}$ be a given, fixed function, and let h be a corresponding thresholded classifier. Finally, we redefine $\rho(x) \doteq C(\pi(x), F(x)) - C_{\min}(\pi(x))$.

a. Show that

$$\rho(x) \geq \begin{cases} 0 & \text{if } h(x) = h_{\text{opt}}(x) \\ \ell(0) - C_{\min}(\pi(x)) & \text{else.} \end{cases}$$

b. For $r \in [-1, +1]$, we redefine

$$\phi(r) \doteq \ell(0) - C_{\min}\left(\frac{1+r}{2}\right).$$

Prove that ϕ has the following properties:

i. $\phi(r) = \phi(-r)$ for $r \in [-1, +1]$.

ii. ϕ is convex. [*Hint:* First prove and then apply the fact that if $\mathcal{F}$ is a family of convex, real-valued functions, then the function g defined by $g(x) = \sup_{f \in \mathcal{F}} f(x)$ is also convex.]

iii. $\phi(0) = 0$ and $\phi(r) > 0$ for $r \neq 0$. [*Hint:* For fixed $r \neq 0$, consider the values of $C((1+r)/2, z)$ in a small neighborhood of $z = 0$.]

iv. ϕ is strictly increasing on $[0, 1]$.

c. Prove that

$$\phi(\text{err}(h) - \text{err}^*) \leq \text{risk}(F) - \text{risk}^*.$$

d. Let $F_1, F_2, \ldots$ be a sequence of functions, and $h_1, h_2, \ldots$ a corresponding sequence of thresholded classifiers (that is, $h_n(x) = \text{sign}(F_n(x))$ whenever $F_n(x) \neq 0$). Prove that, as $n \to \infty$, if $\text{risk}(F_n) \to \text{risk}^*$, then $\text{err}(h_n) \to \text{err}^*$.

12.3 We continue exercise 12.2.

a. Suppose the loss $\ell(z) = \ln(1 + e^{-z})$. Show that

$$\phi(r) = \text{RE}_b\left(\frac{1+r}{2} \;\|\; \frac{1}{2}\right).$$

Also, show that if $\text{risk}(F) \leq \text{risk}^* + \varepsilon$, then

$$\text{err}(\text{sign}(F)) \leq \text{err}^* + \sqrt{2\varepsilon}.$$

b. Compute $\phi(r)$ for each of the following loss functions. Express your answers in as simple a form as possible.

i. $\ell(z) = (1 - z)^2$.

ii. $\ell(z) = (\max\{1 - z, 0\})^2$.

iii. $\ell(z) = \max\{1 - z, 0\}$.

Exercises 12.4 and 12.5 outline alternative methods for obtaining rates of convergence of the exponential loss to its minimum for two different variants of AdaBoost. Aside from the changes described below, we adopt the setup and notation of section 12.2. In particular, $\check{F}_B$ is a reference function with $|\check{F}_B| < B$.

12.4 *AdaBoost.S* is the same as AdaBoost, except that at the end of each round, the current combination of weak hypotheses is *scaled back*, that is, multiplied by a scalar in $[0, 1]$ if doing so will further reduce the exponential loss. Pseudocode is shown as algorithm 12.1, using the formulation of AdaBoost as a greedy algorithm for minimizing exponential loss

Algorithm 12.1
AdaBoost.S, a modified version of AdaBoost

Given: $(x_1, y_1), \ldots, (x_m, y_m)$ where $x_i \in \mathcal{X}$, $y_i \in \{-1, +1\}$.
Initialize $F_0 \equiv 0$.
For $t = 1, \ldots, T$:

- Choose $h_t \in \mathcal{H}$, $\alpha_t \in \mathbb{R}$ to minimize

$$\frac{1}{m} \sum_{i=1}^{m} \exp(-y_i (F_{t-1}(x_i) + \alpha_t h_t(x_i)))$$

  (over all choices of α_t and h_t).

- Update:

$$\tilde{F}_t = F_{t-1} + \alpha_t h_t$$

  and scale back:

$$F_t = s_t \tilde{F}_t$$

  where $s_t \in [0, 1]$ minimizes

$$\frac{1}{m} \sum_{i=1}^{m} \exp(-y_i s_t \tilde{F}_t).$$

Output F_T.

as presented in section 7.1. The code is largely the same as in algorithm 7.1 (p. 178), maintaining a combination F_t of weak hypotheses, and greedily choosing α_t and h_t on each round to effect the greatest drop in the empirical exponential loss. However, at the end of the round, after creating the new combination $\tilde{F}_t = F_{t-1} + \alpha_t h_t$, the result is multiplied by the value s_t in $[0, 1]$ that causes the greatest decrease in the exponential loss.

Below, D_t, R_t, and ΔR_t are defined as in equations (12.22), (12.20), and (12.27), but with F_t as redefined above.

a. Prove that

$$\sum_{i=1}^{m} D_t(i) y_i F_{t-1}(x_i) \geq 0.$$

[*Hint:* Consider the first derivative of $\widehat{\text{risk}}(s \tilde{F}_{t-1})$ when viewed as a function of s.]

b. Prove that if $R_{t-1} \geq 0$, then

$$\Delta R_t \geq \frac{R_{t-1}^2}{2B^2}.$$

[*Hint:* Prove an upper bound on R_{t-1} and a lower bound on ΔR_t, both in terms of γ_t (appropriately redefined for AdaBoost.S).]

c. Prove that if $R_t > 0$, then

$$\frac{1}{R_t} \geq \frac{1}{R_{t-1}} + \frac{1}{2B^2}.$$

d. Finally, show that

$$\widehat{\text{risk}}(F_T) \leq \widehat{\text{risk}}(\check{F}_B) + \frac{4B^2}{T}$$

(a much better bound than the one given for AdaBoost in lemma 12.4).

12.5 Consider a variant of AdaBoost that is the same as algorithm 7.1 (p. 178) except that α_t is restricted to the set $[-c_t, c_t]$; that is, on each round, $F_t = F_{t-1} + \alpha_t h_t$ where α_t and h_t are chosen together to greedily minimize the exponential loss over all choices of $h_t \in \mathcal{H}$ and over all choices of α_t in the restricted set $[-c_t, c_t]$ (rather than over all $\alpha_t \in \mathbb{R}$, as in algorithm 7.1). Here, $c_1, c_2, \ldots$ is a prespecified, nonincreasing sequence of positive numbers for which we assume that $\sum_{t=1}^{\infty} c_t = \infty$, but $\sum_{t=1}^{\infty} c_t^2 < \infty$. (For instance, $c_t = t^{-a}$, where $\frac{1}{2} < a \leq 1$, satisfies these conditions.) We assume $B > c_1$.

In what follows, R_t and D_t are as defined in equations (12.20) and (12.22) (with F_t as redefined above). However, we here redefine $S_t \doteq B + \sum_{t'=1}^{t} c_{t'}$.

a. For any $\alpha \in \mathbb{R}$ and $h \in \mathcal{H}$, use Taylor's theorem (theorem A.1) to show that

$$\ln\left(\widehat{\text{risk}}(F_{t-1} + \alpha h)\right) \leq \ln\left(\widehat{\text{risk}}(F_{t-1})\right) - \alpha \sum_{i=1}^{m} D_t(i) y_i h(x_i) + \frac{\alpha^2}{2}.$$

b. Let $\hat{h}_1, \ldots, \hat{h}_n \in \mathcal{H}$, and let $w_1, \ldots, w_n \in \mathbb{R}$ with $\sum_{j=1}^{n} |w_j| = 1$. Show that

$$\ln\left(\widehat{\text{risk}}(F_t)\right) \leq \ln\left(\widehat{\text{risk}}(F_{t-1})\right) - c_t \sum_{i=1}^{m} \sum_{j=1}^{n} w_j D_t(i) y_i \hat{h}_j(x_i) + \frac{c_t^2}{2}.$$

[*Hint:* Prove upper and lower bounds on $\sum_{j=1}^{n} |w_j| \ \ln\left(\widehat{\text{risk}}(F_{t-1} + c_t \text{sign}(w_j)\hat{h}_j)\right)$.]

c. On a particular round t, show that there exist a finite set of hypotheses $\hat{h}_1, \ldots, \hat{h}_n \in \mathcal{H}$, and real numbers $a_1, \ldots, a_n$ and $b_1, \ldots, b_n$ such that F_{t-1} and $\check{F}_B$ can be written in the form

$$F_{t-1}(x) = \sum_{j=1}^{n} a_j \hat{h}_j(x) \text{ and } \check{F}_B(x) = \sum_{j=1}^{n} b_j \hat{h}_j(x),$$

and for which

$$\sum_{j=1}^{n} \big(|a_j| + |b_j|\big) \leq S_{t-1}.$$

(Keep in mind that $\mathcal{H}$ need not be finite.)

d. By setting $w_j = (b_j - a_j)/W$ in part (b), where $W \doteq \sum_{j=1}^{n} |b_j - a_j|$, show that

$$R_t \leq R_{t-1}\left(1 - \frac{c_t}{S_{t-1}}\right) + \frac{c_t^2}{2}.$$

[*Hint:* Use equation (12.23).]

e. Show that

$$1 - \frac{c_t}{S_{t-1}} \leq \frac{S_{t-1}}{S_t}.$$

f. Show that

$$R_T \leq \frac{B^2}{S_T} + \frac{1}{2}\sum_{t=1}^{T} \frac{S_t}{S_T} \cdot c_t^2.$$

g. Let $\sigma(1), \sigma(2), \ldots$ be a sequence of positive integers such that $1 \leq \sigma(t) \leq t$ for all t, and as $t \to \infty$, $\sigma(t) \to \infty$ but $S_{\sigma(t)}/S_t \to 0$. Show that such a sequence must exist.

h. Show that

$$R_T \leq \frac{B^2}{S_T} + \frac{1}{2}\left[\frac{S_{\sigma(T)}}{S_T}\sum_{t=1}^{\sigma(T)} c_t^2 + \sum_{t=\sigma(T)+1}^{T} c_t^2\right], \tag{12.52}$$

and that the right-hand side of this inequality approaches 0 as $T \to \infty$. This shows that $\lim_{T\to\infty} \widehat{\text{risk}}(F_T) \leq \widehat{\text{risk}}(\check{F}_B)$, with rates of convergence in terms of B and the c_t's that can be obtained using equation (12.52).

12.6 Rather than using AdaBoost for a bounded number of rounds, consider applying regularization to the exponential loss. To be specific, given the setup and assumptions of theorem 12.2, and for $B > 0$, let $\hat{F}_B$ be any function which minimizes $\widehat{\text{risk}}(F)$ over all $F \in \text{span}(\mathcal{H})$ with $|F| \leq B$. (For simplicity, assume such a minimizing function exists.) As usual, $\check{F}_B$ is a reference function from this same space.

a. Prove that, with probability at least $1 - \delta$,

$$\text{risk}(\hat{F}_B) \leq \text{risk}(\breve{F}_B) + O\left(e^B \cdot B \cdot \sqrt{\frac{d \ln(m/d)}{m}} + e^B \sqrt{\frac{\ln(1/\delta)}{m}}\right).$$

[*Hint:* Use the techniques of section 5.3.]

b. Conclude that $\text{risk}(\hat{F}_B)$ converges almost surely to risk^* as $m \to \infty$, for an appropriate choice of B as a function of m.

12.7 Let the domain $\mathcal{X} = [0, 1]^n$, and let us assume that the conditional probability function π given in equation (12.1) is Lipschitz, meaning that for some constant $k > 0$, and for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$,

$$\left|\pi(\mathbf{x}) - \pi(\mathbf{x}')\right| \leq k \ \|\mathbf{x} - \mathbf{x}'\|_2.$$

Let $\mathcal{H}$ be the space of all decision trees with at most cn internal nodes where each test at each node is of the form $x_j \leq v$, for some $j \in \{1, \ldots, n\}$ and some $v \in \mathbb{R}$. Here, $c > 0$ is an absolute constant of your choosing (not dependent on n, k, or π).

a. Show that equation (12.11) holds in this case.

b. Show that the VC-dimension of $\mathcal{H}$ is upper bounded by a polynomial in n.

12.8 Verify the following details, which were omitted from the proof of theorem 12.13:

a. The vectors $\boldsymbol{\lambda}^*(c)$, for all $c \in [0, 1]$, are included in some compact subset of $\mathbb{R}^2$.

b. The minimizer of $\boldsymbol{\lambda}^*(0)$ is unique.

c. The partial derivative $\partial L(\boldsymbol{\lambda}, 0)/\partial \lambda_2$ is strictly negative when $\lambda_2 = 0$.

d. $\lambda_2^*(0) > 0$.

12.9 Let $\ell : \mathbb{R} \to \mathbb{R}_+$ be a margin-based loss function satisfying exactly the same properties described at the beginning of exercise 12.2. Note that these properties imply that ℓ is continuous. Suppose that in the construction of section 12.3.1 exponential loss is replaced by ℓ. In particular, this means redefining

$$K(z) \doteq (1 - \eta)\ell(z) + \eta\ell(-z).$$

In this exercise, we will see how to modify theorem 12.13 to prove a more general result that holds when any loss function ℓ with the stated properties is minimized in place of exponential loss.

a. Prove that $\lim_{s \to -\infty} \ell(s) = \infty$. [*Hint:* Use equation (A.3).]

b. Show that there exists a compact set $C \subseteq \mathbb{R}^2$ such that if $\boldsymbol{\lambda}$ minimizes $L(\boldsymbol{\lambda}, c)$ for any $c \in [0, 1]$, then $\boldsymbol{\lambda} \in C$.

c. Show that if ℓ is strictly convex, then $L(\boldsymbol{\lambda}, 0)$ has a unique minimum. Also, give an example showing that the minimum of $L(\boldsymbol{\lambda}, 0)$ need *not* be unique without this additional

assumption. (You should not assume ℓ is strictly convex in the remaining parts of this exercise.)

d. Let $M \subseteq \mathbb{R}^2$ be the set of *all* minima of $L(\boldsymbol{\lambda}, 0)$. Show that if $\boldsymbol{\lambda} \in M$, then $\lambda_2 > 0$.

e. Let $\boldsymbol{\lambda}^*(c)$ be as in theorem 12.13 (but for loss ℓ), and let $c_1, c_2, \ldots$ be any sequence converging to 0. Prove that if the sequence $\boldsymbol{\lambda}^*(c_1), \boldsymbol{\lambda}^*(c_2), \ldots$ converges, then its limit is in M.

f. Show that there exists $c_0 > 0$ such that for all $c \in (0, c_0]$, $F_{\boldsymbol{\lambda}^*(c)}(x_{pe}) < 0$.

12.10 Throughout this exercise, assume that instances $\mathbf{x}$ are binary vectors in $\{-1, +1\}^N$. We consider weighted combinations of the components of such vectors which now include a constant term. In other words, weight vectors now have the form $\boldsymbol{\lambda} = \langle \lambda_0, \lambda_1, \ldots, \lambda_N \rangle \in \mathbb{R}^{N+1}$, and (re)define the combination

$$F_{\boldsymbol{\lambda}}(\mathbf{x}) \doteq \lambda_0 + \sum_{j=1}^{N} \lambda_j x_j.$$

a. Suppose the weak learner produces confidence-rated decision stumps of the form

$$h(\mathbf{x}) = \begin{cases} c_+ & \text{if } x_j = +1 \\ c_- & \text{if } x_j = -1 \end{cases}$$

for some $c_+, c_- \in \mathbb{R}$ and some index $j \in \{1, \ldots, N\}$. Show that if $h_1, \ldots, h_T$ all have this form, and $\alpha_1, \ldots, \alpha_T \in \mathbb{R}$, then there exists $\boldsymbol{\lambda} \in \mathbb{R}^{N+1}$ for which $\sum_{t=1}^{T} \alpha_t h_t(\mathbf{x}) = F_{\boldsymbol{\lambda}}(\mathbf{x})$ for all $\mathbf{x} \in \{-1, +1\}^N$.

b. Suppose $\mathcal{D}$ is the distribution in section 12.3.2, and that $\boldsymbol{\lambda} \in \mathbb{R}^{N+1}$ minimizes the risk $\mathbf{E}_{\mathcal{D}}\left[e^{-yF_{\boldsymbol{\lambda}}(\mathbf{x})}\right]$. What will be the classification error (with respect to $\mathcal{D}$) of the induced classifier, $\text{sign}(F_{\boldsymbol{\lambda}})$? How does this compare to the Bayes optimal?

c. For any noise rate $\eta \in \left(0, \frac{1}{2}\right)$, show how to construct a modified distribution $\mathcal{D}$ so that if $\boldsymbol{\lambda}$ minimizes the risk (with respect to this new distribution), then the induced classifier, $\text{sign}(F_{\boldsymbol{\lambda}})$, will have classification error at least $\frac{1}{2}$, even though the Bayes error can be achieved by some other combination of the same form.