3.1 A Bound on AdaBoost's Training Error

We begin by proving a fundamental bound on AdaBoost's training error. In proving this main theorem, we make no assumptions about the training set and how it was generated, nor about the weak learner. The theorem simply gives a bound on the training error in terms of the error rates of the weak hypotheses.

In the simple version of AdaBoost shown as algorithm 1.1 (p. 5), D_1 is initialized to the uniform distribution over the training set. Here, however, we give a slightly more general proof applicable to an arbitrary initialization of D_1. The resulting proof provides an upper bound on the *weighted* fraction of examples misclassified by H, where each example i is weighted by $D_1(i)$. A bound on the ordinary, unweighted training error, when D_1 is initialized as in algorithm 1.1, follows immediately as a special case.

Theorem 3.1 Given the notation of algorithm 1.1, let $\gamma_t \doteq \frac{1}{2} - \epsilon_t$, and let D_1 be an arbitrary initial distribution over the training set. Then the weighted training error of the combined classifier H with respect to D_1 is bounded as

$$\mathbf{Pr}_{i\sim D_1}[H(x_i) \neq y_i] \leq \prod_{t=1}^{T} \sqrt{1-4\gamma_t^2} \leq \exp\left(-2\sum_{t=1}^{T}\gamma_t^2\right).$$

Note that because $\epsilon_t = \frac{1}{2} - \gamma_t$, the *edge* γ_t measures how much better than the random-guessing error rate of $\frac{1}{2}$ is the error rate of the t-th weak classifier h_t. As an illustration of the theorem, suppose all of the γ_t's are at least 10% so that no h_t has error rate above 40%. Then the theorem implies that the training error of the combined classifier is at most

$$\left(\sqrt{1-4(0.1)^2}\right)^T \approx (0.98)^T.$$

In other words, the training error drops *exponentially fast* as a function of the number of base classifiers combined. More discussion of this property follows below.

Here is the informal idea behind the theorem: On every round, AdaBoost increases the weights (under distribution D_t) of the misclassified examples. Moreover, because the final classifier H is a (weighted) majority vote of the weak classifiers, if some example is misclassified by H, then it must have been misclassified by most of the weak classifiers as well. This means that it must have had its weight increased on many rounds, so that its weight under the final distribution D_{T+1} must be large. However, because D_{T+1} is a distribution (with weights that sum to 1), there can be only a few examples with large weights, that is, where H makes an incorrect prediction. Therefore, the training error of H must be small.

We now give a formal argument.

Proof Let

$$F(x) \doteq \sum_{t=1}^{T} \alpha_t h_t(x). \tag{3.1}$$

Unraveling the recurrence in algorithm 1.1 that defines D_{t+1} in terms of D_t gives

$$\begin{aligned} D_{T+1}(i) &= D_1(i) \times \frac{e^{-y_i \alpha_1 h_1(x_i)}}{Z_1} \times \cdots \times \frac{e^{-y_i \alpha_T h_T(x_i)}}{Z_T} \\ &= \frac{D_1(i) \exp\left(-y_i \sum_{t=1}^{T} \alpha_t h_t(x_i)\right)}{\prod_{t=1}^{T} Z_t} \\ &= \frac{D_1(i) \exp\left(-y_i F(x_i)\right)}{\prod_{t=1}^{T} Z_t}. \end{aligned} \tag{3.2}$$

Since $H(x) = \text{sign}(F(x))$, if $H(x) \neq y$, then $yF(x) \leq 0$, which implies that $e^{-yF(x)} \geq 1$. That is, $\mathbf{1}\{H(x) \neq y\} \leq e^{-yF(x)}$. Therefore, the (weighted) training error is

$$\begin{aligned} \mathbf{Pr}_{i \sim D_1}[H(x_i) \neq y_i] &= \sum_{i=1}^{m} D_1(i)\ \mathbf{1}\{H(x_i) \neq y_i\} \\ &\leq \sum_{i=1}^{m} D_1(i) \exp\left(-y_i F(x_i)\right) && (3.3) \\ &= \sum_{i=1}^{m} D_{T+1}(i) \prod_{t=1}^{T} Z_t && (3.4) \\ &= \prod_{t=1}^{T} Z_t && (3.5) \end{aligned}$$

where equation (3.4) uses equation (3.2), and equation (3.5) uses the fact that D_{T+1} is a distribution (which sums to 1). Finally, by our choice of α_t, we have that

$$\begin{aligned} Z_t &= \sum_{i=1}^{m} D_t(i) e^{-\alpha_t y_i h_t(x_i)} \\ &= \sum_{i: y_i = h_t(x_i)} D_t(i) e^{-\alpha_t} + \sum_{i: y_i \neq h_t(x_i)} D_t(i) e^{\alpha_t} && (3.6) \\ &= e^{-\alpha_t}\ (1 - \epsilon_t) + e^{\alpha_t}\ \epsilon_t && (3.7) \\ &= e^{-\alpha_t}\ \left(\frac{1}{2} + \gamma_t\right) + e^{\alpha_t}\ \left(\frac{1}{2} - \gamma_t\right) && (3.8) \\ &= \sqrt{1 - 4\gamma_t^2}. && (3.9) \end{aligned}$$

Here, equation (3.6) uses the fact that both y_i and $h_t(x_i)$ are $\{-1, +1\}$-valued; equation (3.7) follows from the definition of ϵ_t; and equation (3.9) uses the definition of α_t, which, as we will discuss shortly, was chosen specifically to minimize equation (3.7).

Plugging into equation (3.5) gives the first bound of the theorem. For the second bound, we simply apply the approximation $1 + x \leq e^x$ for all real x. ∎

From the proof, it is apparent where AdaBoost's choice of α_t comes from: The proof shows that the training error is upper bounded by $\prod_t Z_t$. To minimize this expression, we can minimize each Z_t separately. Expanding Z_t gives equation (3.7), which can be minimized over choices of α_t using simple calculus giving the choice of α_t used in algorithm 1.1. Note that α_t is being chosen *greedily* on each round t without consideration of how that choice will affect future rounds.

As discussed above, theorem 3.1 assures a rapid drop in training error when each weak classifier is assumed to have error bounded away from $\frac{1}{2}$. This assumption, that $\epsilon_t \leq \frac{1}{2} - \gamma$ for some $\gamma > 0$ on every round t, is a slight relaxation of the empirical γ-weak learning assumption, as discussed in section 2.3.3. When this condition holds, theorem 3.1 implies that the combined classifier will have training error at most

$$\left(\sqrt{1 - 4\gamma^2}\right)^T \leq e^{-2\gamma^2 T},$$

an exponentially decreasing function of T for any $\gamma > 0$. Although the bound on training error is easier to understand in light of the weak-learnability condition, it is important to remember that AdaBoost and its analysis do *not* require this condition. AdaBoost, being adaptive, does not need to assume an a priori lower bound on the γ_t's, and the analysis takes into account all of the γ_t's. If some γ_t's are large, then the progress (in terms of reducing the bound on the training error) will be that much greater.

Although the bound implies an exponential drop in training error, the bound itself is nevertheless rather loose. For instance, figure 3.1 shows a plot of the training error of the combined classifier compared to the theoretical upper bound as a function of the number of rounds of boosting for the heart-disease dataset described in section 1.2.3. The figure also shows the training errors ϵ_t of the base classifiers h_t with respect to the distributions D_t on which they were trained.

3.2 A Sufficient Condition for Weak Learnability

The assumption of empirical γ-weak learnability is fundamental to the study of boosting, and theorem 3.1 proves that this assumption is sufficient to ensure that AdaBoost will drive down the training error very quickly. But when does this assumption actually hold? Is it possible that this assumption is actually vacuous, in other words, that there are no natural situations in which it holds? What's more, our formulation of weak learnability is somewhat cumbersome, depending as it does on the weighted training error of base hypotheses with respect to virtually any distribution over the training set.