CS281B/Stat241B (Spring 2006) Statistical Learning Theory **Lecture: 13**

More on AdaBoost, Risk bounds

Lecturer: Peter Bartlett *Scribe: Jason Wolfe*

1 AdaBoost discrete loss

The typical objective of classification is to, given a data set $(X_1, Y_1, ...X_n, Y_n)$, choose a classifier f that minimizes the expected discrete loss

$$L(f) = \mathbb{E}\, l(Y, f(X)) = \mathbb{E}\, \mathbf{1}[[f(X) \neq Y]]$$

on future data points. AdaBoost, on the other hand, chooses f to minimize the sample average $\hat{L}_\phi(f)$ where

$$L_\phi(f) = \mathbb{E}\, \phi(-Y f(X))$$

and $\phi(\alpha) = e^{-\alpha}$, which is convex.

Thus, AdaBoost involves two separate approximations to $L(f)$: it approximates the expected discrete loss L by the expected loss L_ϕ, and then further approximates this expectation by the sample average $\hat{L}_\phi$. These approximations can be justified theoretically, by demonstrating that choosing f^* to minimize $\hat{L}_\phi(f^*)$ will (under certain conditions) also tend to minimize $L(f^*)$.

We begin by discussing the relationship between L and L_ϕ. In particular, we assume that a f with small $L_\phi(f)$ has been chosen, and attempt to show that $L(f) = \mathbb{E}\, \mathbf{1}[[f(X) \neq Y]]$ will be small as well. Recall that for

$$H(\eta) = \inf_{\alpha \in \mathbb{R}} \left(\phi(\alpha)\eta + \phi(-\alpha)(1-\eta)\right) \quad \text{for } \eta \in [0, 1],$$

$$\Psi(\theta) = \phi(0) - H\left(\frac{1+\theta}{2}\right) = \phi(0) - \inf_\alpha \left(\frac{1+\theta}{2}\phi(\alpha) + \frac{1-\theta}{2}\phi(-\alpha)\right),$$

and ϕ *classification calibrated* (as is $\phi(\alpha) = e^{-\alpha}$), we have

$$\Psi(L(f) - L^*) \leq L_\phi(f) - L^*_\phi.$$

Here, the *Bayes risk* $L^* = \inf_g L(g)$ and the optimal ϕ risk $L^*_\phi = \inf_g L_\phi(g)$ where g ranges, e.g., over all measurable functions.

In the context of AdaBoost,

$$H(\eta) = \inf_\alpha \left(\eta e^{-\alpha} + (1-\eta)e^{\alpha}\right).$$

Differentiating and setting to zero yields the optimal

$$\alpha^* = \ln \sqrt{\frac{\eta}{1-\eta}};$$

this suggests that if we could choose $f(x)$ separately for each x, it would be a monotonically transformed version of conditional probability. Plugging this α^* into H yields

$$H(\eta) = 2\sqrt{\eta(1-\eta)},$$

which is concave downward and symmetric around 1/2. Then $\Psi(\theta)$ simplifies to

$$\Psi(\theta) = 1 - \sqrt{(1+\theta)(1-\theta)} = 1 - \sqrt{1-\theta^2}.$$

Finally, plugging this in to the original inequality yields

$$1 - \sqrt{1 - (L(f) - L^*)^2} \leq L_\phi(f) - L_\phi^*.$$

Examining the Taylor series of the left side about 0 shows that this is equivalent to

$$L(f) - L^* \leq \sqrt{2(L_\phi(f) - L_\phi^*)}$$

when the excess risk is sufficiently small. Thus, driving the excess ϕ risk to zero will drive the discrete loss to zero as well, which justifies AdaBoost's use of this particular convex loss function.

2 Relationship to logistic regression

It turns out that we can interpret the value of $F(x)$ (where F is the boosted classifier returned by AdaBoost) as a transformed estimate of $\mathbb{P}(Y = 1|X = x)$. Consider a logistic model where

$$\mathbb{P}(Y = 1|X = x) = \frac{1}{1 + e^{-2f(x)}} = \frac{e^{f(x)}}{e^{f(x)} + e^{-f(x)}},$$

a rescaled version of the logistic function. In this model, the log loss (negative log likelihood) takes the form

$$\begin{aligned} -\ln \prod_{i=1}^{n} \mathbb{P}(Y = y_i|X = x_i) &= -\sum_{y_i=1} \ln \frac{1}{1+e^{-2f(x)}} - \sum_{y_i=-1} \ln\left(1 - \frac{1}{1+e^{-2f(x)}}\right) \\ &= \sum_{y_i=1} \ln\left(1 + e^{-2f(x)}\right) + \sum_{y_i=-1} \ln\left(1 + e^{2f(x)}\right) \\ &= \sum_{i=1}^{n} \ln\left(1 + e^{-2y_i f(x_i)}\right). \end{aligned}$$

Thus, the maximum likelihood logistic regression solution attempts to minimize the sample average of $\phi(\alpha) = \ln\left(1 + e^{-2\alpha}\right)$. This is closely related to AdaBoost, which minimizes the sample average of $\phi(\alpha) = e^{-\alpha}$. To see the connection, note that the first few terms in the Taylor expansion of $\ln\left(1 + e^{-2\alpha}\right) + 1 - \ln 2$ about 0,

$$1 - \alpha + \frac{\alpha^2}{2} - \ldots,$$

are actually identical to the corresponding terms in the Taylor expansion of $e^{-\alpha}$ about 0.

While the two functions are very similar near zero, their asymptotic behavior is very different. In general we have that

$$\ln\left(1 + e^{-2\alpha}\right) \leq e^{-\alpha};$$

furthermore, the former grows linearly as α approaches $-\infty$, whereas the latter grows exponentially. Thus, we can view AdaBoost as approximating the maximum likelihood logistic regression solution, except with (sometimes exponentially) larger penalties for mistakes. A further similarity between the methods is that the α^* for $\phi(\alpha) = \ln\left(1 + e^{-2\alpha}\right)$ is the same as for AdaBoost (see Section 1).

3 History of AdaBoost

AdaBoost has its roots in early theoretical investigations of classification and weak PAC (probably approximately correct) learning. The issue being considered was roughly the following: assume a class F of $\{-1,+1\}$-valued functions is given such that the Bayes risk $L^* = 0$ (e.g., $\exists f \in F$ with $L(f) = L^* = 0$, $\mathbb{P}(Y = f(X)) = 1$). Given this strong assumption, when we can (computationally) find this optimal f given some sample data points?

More formally, recall the basic idea of PAC learning:

$$\mathbb{P}\{(X_1, Y_1, ...X_n, Y_n) : L(f_n) \leq \epsilon\} \geq 1 - \delta.$$

That is, there exist quantities ϵ and δ such that the expected loss of the selected function f_n (which depends on the data) exceeds ϵ with probability less than δ. Weak PAC learning is similar, except that we are only concerned with bounding the loss away from 1/2:

$$\mathbb{P}\left\{(X_1, Y_1, ...X_n, Y_n) : L(f_n) \leq \frac{1}{2} - \gamma\right\} \geq 1 - \delta,$$

where γ is typically polynomial in n.

Schapire was the first to show that these two formulations are in fact equivalent. His proof was constructive, in that it demonstrated a particular learning algorithm (AdaBoost) that was able to construct a strong PAC hypothesis from an ensemble of weak PAC hypotheses. Several earlier earlier algorithms had come close to this result, but had the drawback that γ had to be known in advance. AdaBoost was the first boosting algorithm to relax this restriction; in fact, its name comes from the fact that it is able to "adapt" to different values of γ.

While AdaBoost was developed as the solution to a theoretical problem under rather strong assumptions, it also happens to demonstrate very good empirical performance. A large number of other boosting algorithms have been proposed since, but the original AdaBoost remains one of the most popular variants (perhaps by virtue of its simplicity).

4 Introduction to uniform convergence of risk bounds

So far, we've looked at algorithms (including AdaBoost) that optimize over a set of training samples:

$$\min_{f \in F} \hat{L}(f) = \hat{\mathbb{E}}\, l(y, f(x)) = \frac{1}{n}\sum_{i=1}^{n} l(y_i, f(x_i)).$$

If the empirical minimizer is f^*, we are interested in bounding the true loss $L(f^*) = \mathbb{E}\, l(y, f^*(x))$ under this function. In particular, we hope that $\hat{L}(f^*)$ will converge to $L(f^*)$ as $n \to \infty$.

For the (trivial) case where our function class F contains only a single function, we can simply appeal to the law of large numbers. For example, in the case of discrete loss, the Chernoff bound gives an upper bound on $\mathbb{P}(|\hat{L}(f^*) - L(f^*)| > \epsilon)$ that shrinks exponentially in n for any given ϵ.

This argument, however, fails when F is not a singleton. We cannot simply apply the law of large numbers to each $f \in F$ and then argue that the desired property holds when minimizing over all of F. The problem is that we are considering $L(\text{argmin}_{f \in F}\hat{L}(f))$, where the inner part depends on the data. In particular, if F is such that for any n and data set there are functions $f \in F$ with small $\hat{L}(f)$ but large $L(f)$, then choosing an f that minimizes $\hat{L}(f)$ may not tend to minimize $L(f)$.