COS 511: Theoretical Machine Learning

Lecturer: Rob Schapire — Lecture 8
Scribe: Rafael Mendes de Oliveira — February 28, 2013

Last time, we talked about Hoeffding's inequality. That is, given $X_1, \ldots, X_m$ i.i.d., where $X_i \in [0,1]$, $1 \le i \le m$, and given $p = \mathbf{E}[X_i]$, if we define $\hat{p} = \frac{1}{m}\sum_{i=1}^{m} X_i$ (the empirical average), we have that

$$\Pr[|\hat{p} - p| \ge \epsilon] \le 2e^{-2\epsilon^2 m}.$$

Having this inequality, we asked the question: how fast does the training error converge to its true error?

And the answer was: if we fix the hypothesis h, and define $X_i = 1_{\{h(x_i) \neq y_i\}}$, $p = err(h)$ and $\hat{p} = e\hat{r}r(h)$, then we get the following bound:

$$\Pr[|err(h) - e\hat{r}r(h)| \ge \epsilon] \le 2e^{-2\epsilon^2 m}.$$

1 A Stronger Chernoff Bound

Today, we will show that:

$$\Pr[\hat{p} \ge p + \epsilon] \le \exp(-RE(p + \epsilon \mid\mid p)m) \tag{1}$$

where $RE(p + \epsilon \mid\mid p)$ is the relative entropy, or Kullback-Leibler divergence. Now, what is relative entropy? This concept comes from information theory, and we will define it below.

Suppose that Alice wants to send to Bob 1 letter of the alphabet. There are 26 letters, and therefore (in the naive way), we can encode each letter uniquely using a 5-bit string. If Alice and Bob are communicating in English, in this encoding Alice is using the same number of bits for common letters (such as A, E) as for uncommon letters (such as Q and Z). For this reason, this encoding turns out to be rather inefficient.

It turns out that if messages are coming from some distribution P where $P(x)$ is the probability of sending symbol x, then the best way to encode x is to use $\lg\left(\frac{1}{P(x)}\right)$ bits to encode symbol x. With this information, the expected size of the message is $\sum_x P(x)\lg\left(\frac{1}{P(x)}\right)$, which is equal to the entropy of the distribution P.

The argument above works when we know the distribution P. What if we assume that the real distribution is Q instead of P? (That is, we are assuming a wrong distribution.) Now, we will be using a wrong encoding, since we are encoding for Q rather than P. Then – by the argument above – the expected number of bits is

$$\sum_x P(x)\lg\left(\frac{1}{Q(x)}\right).$$

Hence, our deviation from the optimum is

$$\sum_x P(x)\lg\left(\frac{1}{Q(x)}\right) - \sum_x P(x)\lg\left(\frac{1}{P(x)}\right) = \sum_x P(x)\lg\left(\frac{P(x)}{Q(x)}\right) = RE(P \mid\mid Q)$$

And this is the definition of relative entropy. From now on, we will slightly change our definition of relative entropy to be $RE(P \mid\mid Q) = \sum_x P(x) \ln \left(\frac{P(x)}{Q(x)} \right)$ (this corresponds to the original definition up to multiplication by a constant). This new definition will be useful in our calculations later on.

Notice that $RE(P \mid\mid Q)$ is always nonnegative, since we "know" that $\sum_x P(x) \lg \left(\frac{1}{P(x)} \right)$ is the optimum. The relative entropy computes how far apart distribution Q is from P. When $P = (p, 1-p)$ and $Q = (q, 1-q)$ are distributions over just two items, we will often use the abbreviated notation $RE(p \mid\mid q)$ as shorthand for $RE((p, 1-p) \mid\mid (q, 1-q))$, where $p, q \in [0, 1]$.

Now, we are ready to prove (1).

Suppose $X \geq 0$ is a random variable and $t > 0$ is any positive number. Then Markov's inequality says that

$$\Pr[X \geq t] \leq \frac{\mathbf{E}[X]}{t}. \tag{2}$$

Proof: $\mathbf{E}[X] = \Pr[X \geq t] \cdot \mathbf{E}[X \mid X \geq t] + \Pr[X < t] \cdot \mathbf{E}[X \mid X < t] \geq \Pr[X \geq t] \cdot t + 0$, and this implies (2).

Markov's inequality, as it is, is very weak for our purposes. However, notice that if we have a monotonically increasing function f such that $f(p) \geq 0 \ \forall p$, we can apply Markov's inequality on it and get a better bound. So, we will pick any $\lambda > 0$.

Then, $\hat{p} \geq q \Leftrightarrow e^{\lambda m \hat{p}} \geq e^{\lambda m q}$ and thus $\Pr[\hat{p} \geq q] = \Pr[e^{\lambda m \hat{p}} \geq e^{\lambda m q}]$, and by Markov's inequality on the random variable $f(\hat{p}) = e^{\lambda m \hat{p}}$, we have:

$$\begin{aligned}
\Pr[\hat{p} \geq q] &= \Pr[e^{\lambda m \hat{p}} \geq e^{\lambda m q}] && \text{(by equivalence above)} \\
&\leq e^{-\lambda m q} \cdot \mathbf{E}[e^{\lambda m \hat{p}}] && \text{(by Markov)} \\
&= e^{-\lambda m q} \cdot \mathbf{E}\left[\exp\left(\lambda \sum_{i=1}^{m} X_i \right) \right] \\
&= e^{-\lambda m q} \cdot \mathbf{E}\left[\prod_{i=1}^{m} e^{\lambda X_i} \right] \\
&= e^{-\lambda m q} \cdot \prod_{i=1}^{m} \mathbf{E}\left[e^{\lambda X_i} \right] \\
&\leq e^{-\lambda m q} \cdot \prod_{i=1}^{m} \mathbf{E}\left[1 - X_i + e^{\lambda} \cdot X_i \right] && \text{(by convexity of function } e^{\lambda x} \text{ on } [0,1]) \\
&= e^{-\lambda m q} \cdot \prod_{i=1}^{m} (1 - p + e^{\lambda} \cdot p) && \text{(since } \mathbf{E}[X_i] = p) \\
&= \left(\frac{1 - p + e^{\lambda} \cdot p}{e^{\lambda q}} \right)^m .
\end{aligned}$$

Let $\phi(\lambda) = \ln[e^{-\lambda q}(1 - p + e^{\lambda} p)]$. Then, we have shown that $\forall \lambda > 0$:

$$\Pr[\hat{p} \geq q] \leq e^{\phi(\lambda) m}$$

and therefore, minimizing $\phi(\lambda)$, we get $\lambda_{min} = \ln\left(\frac{q(1-p)}{(1-q)p}\right)$, which implies $\phi(\lambda_{min}) = -RE(q \mid\mid p)$. Setting $q = p+\epsilon$, we get the desired inequality, and this proves (1). □

Note: to prove that $\Pr[\hat{p} \leq p-\epsilon] \leq \exp(-RE(p-\epsilon \mid\mid p)m)$, just plug into the inequality above $X_i' = 1 - X_i$ and proceed with the calculations.

Also, notice that this inequality is stronger than Hoeffdings' inequality. To derive the latter from (1), one just needs to show that $RE(q \mid\mid p) \geq 2(q-p)^2$, which one can do by using the Taylor expansion of $RE(q \mid\mid p)$.

2 McDiarmid's Inequality

We may want to show that, given a function $f(X_1, \ldots, X_m)$ on random variables X_i, its value is close to $\mathbf{E}[f(X_1, \ldots, X_m)]$. This may not be true in general, but if we assume that f is a function that does not change much if we change one of the X_i's while keeping the others fixed, then we can get a good bound on $\Pr[|f(X_1 \ldots, X_m) - \mathbf{E}[f(X_1, \ldots, X_m)]| \geq \epsilon]$. This gives us McDiarmid's inequality, which is stated as follows:

- assume that, for all $x_1, \ldots, x_m$ and x_i', there exists a constant c_i such that:

$$|f(x_1, \ldots, x_i, \ldots, x_m) - f(x_1, \ldots, x_i', \ldots, x_m)| \leq c_i$$

- and assume also that $X_1, \ldots, X_m$ are independent, not necessarily identically distributed.

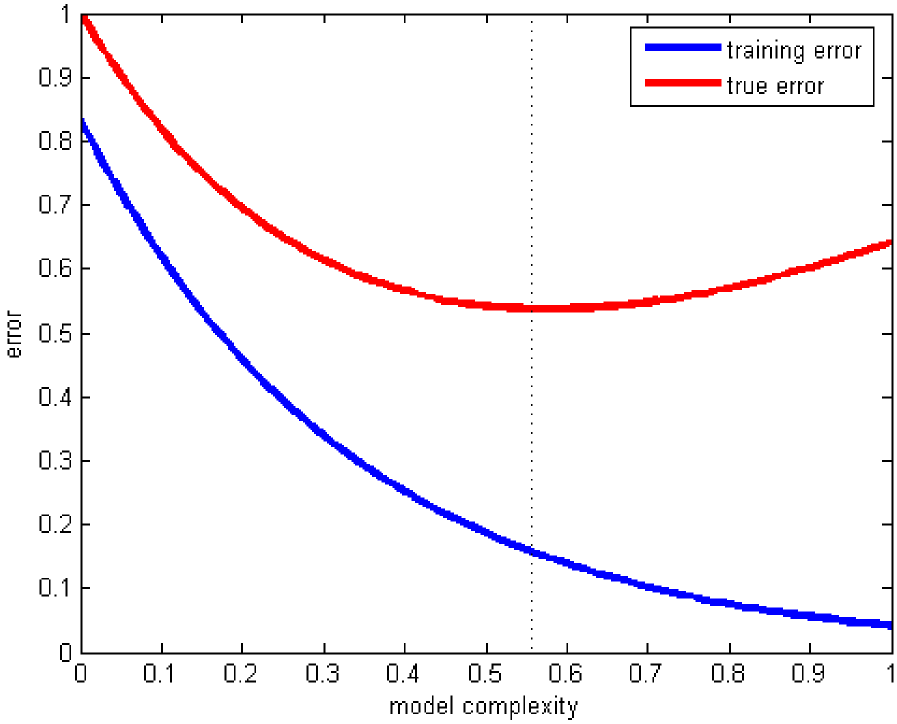
Then, we have that

$$\Pr[f(X_1 \ldots, X_m) \geq \mathbf{E}[f(X_1, \ldots, X_m)] + \epsilon] \leq \exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^m c_i^2}\right).$$

3 Overfitting

For a fixed hypothesis h, we know how to show that $|err_D(H) - \hat{err}_D(h)| \leq \epsilon$, but we want to show that $\forall\ h \in \mathcal{H} : |err_D(H) - \hat{err}_D(h)| \leq \epsilon$ with high probability. If $\mathcal{H}$ is finite, then given m examples, with probability $\geq 1-\delta$, we have $\forall\ h \in \mathcal{H} : |err_D(H) - \hat{err}_D(h)| \leq \epsilon$ if $m = O\left(\frac{\ln(|\mathcal{H}|) + \ln(1/\delta)}{\epsilon^2}\right)$. (This follows from the union bound combined with Hoeffding's inequality.) Or, written another way (using exercise from homework 1), with probability $\geq 1-\delta$, $\forall\ h \in \mathcal{H} : err_D(h) \leq \hat{err}_D(h) + O\left(\sqrt{\frac{|h| + \ln(1/\delta)}{m}}\right)$.

Notice that to guarantee this bound, m depends quadratically on $1/\epsilon$, as opposed to linear in $1/\epsilon$ from previous classes. This is necessary because of Hoeffding's inequality, since the dependency in Hoeffding's inequality is on $e^{-2\epsilon^2 m}$. This result that we just obtained implies that as we increase the complexity of our hypothesis, the upper bound on true error behaves as in the following graph. This is called overfitting. In this graph model complexity refers to $|h|$ and the error refers to ϵ.


training error
true error
error
model complexity
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1