COS 511: Theoretical Machine Learning

Lecturer: Rob Schapire
Scribe: José Simões Ferreira

Lecture #10
March 06, 2013

In the last lecture the concept of Rademacher complexity was introduced, with the goal of showing that for all f in a family of functions $\mathcal{F}$ we have $\hat{\mathrm{E}}_S[f] \approx \mathrm{E}[f]$. Let us summarize the definitions of interest:

$$\mathcal{F} \text{ family of functions } f : Z \to [0,1]$$

$$S = \langle z_1, \ldots, z_m \rangle$$

$$\hat{R}_S(\mathcal{F}) = \mathrm{E}_\sigma \left[\sup_{f \in \mathcal{F}} \frac{1}{m} \sum_{i=1}^{m} \sigma_i f(z_i) \right]$$

$$R_m(\mathcal{F}) = \mathrm{E}_S \left[\hat{R}_S(\mathcal{F}) \right].$$

We also began proving the following theorem:

Theorem. *With probability at least* $1 - \delta$ *and* $\forall f \in \mathcal{F}$*:*

$$E[f] \leq \hat{E}_S[f] + 2R_m(\mathcal{F}) + \mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}} \right)$$

$$E[f] \leq \hat{E}_S[f] + 2\hat{R}_S(\mathcal{F}) + \mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}} \right).$$

Which we now prove in full.

Proof. Let us define:

$$\Phi(S) = \sup_{f \in \mathcal{F}} \left(\mathrm{E}[f] - \hat{\mathrm{E}}_S[f] \right)$$

$$\mathrm{E}[f] = \mathrm{E}_{z \sim D}[f(z)]$$

$$\hat{\mathrm{E}}_S[f] = \frac{1}{m} \sum_{i=1}^{m} f(z_i)$$

Step 1 $\Phi(S) \leq \mathrm{E}_S[\Phi(S)] + \mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}} \right)$

This was proven last lecture and follows from McDiarmid's inequality.

Step 2 $\mathrm{E}_S[\Phi(S)] \leq \mathrm{E}_{S,S'} \left[\sup_{f \in \mathcal{F}} \left(\hat{\mathrm{E}}_{S'}[f] - \hat{\mathrm{E}}_S[f] \right) \right]$

This was also shown last lecture. We also considered generating new samples T, T' by flipping a coin, i.e. running through $i = 1, \ldots, m$ we flip a coin, swapping z_i with z_i' if heads, and doing nothing otherwise. We then claimed that the distributions thus generated are distributed the same as S and S', and we noted

$$\hat{\mathrm{E}}_{S'}[f] - \hat{\mathrm{E}}_S[f] = \frac{1}{m} \sum_i \left(f(z_i') - f(z_i) \right)$$

which means we can write

$$\hat{\mathrm{E}}_{T'}[f] - \hat{\mathrm{E}}_{T}[f] = \frac{1}{m}\sum_i \sigma_i \left(f(z_i') - f(z_i)\right)$$

which is written in terms of Rademacher random variables, σ_i. We now proceed with the proof.

Step 3 We first claim

$$\mathrm{E}_{S,S'}\left[\sup_{f\in\mathcal{F}}\left(\hat{\mathrm{E}}_{S'}[f] - \hat{\mathrm{E}}_{S}[f]\right)\right] = \mathrm{E}_{S,S',\sigma}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{m}\sum_i \left(f(z_i') - f(z_i)\right)\sigma_i\right)\right].$$

To see this, note that the right hand side is effectively the same expectation as the left hand side, but with respect to T and T', which are identically distributed to S and S'. Now we can write

$$\begin{aligned}\mathrm{E}_{S,S',\sigma}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{m}\sum_i \left(f(z_i') - f(z_i)\right)\sigma_i\right)\right] &\leq \mathrm{E}_{S,S',\sigma}\left[\sup_{f\in\mathcal{F}}\frac{1}{m}\sum_i \sigma_i f(z_i')\right] \\ &\quad + \mathrm{E}_{S,S',\sigma}\left[\sup_{f\in\mathcal{F}}\frac{1}{m}\sum_i (-\sigma_i) f(z_i)\right]\end{aligned}$$

where we are just maximizing over the sums separately. We now note two points:

1. The random variable $-\sigma_i$ has the same distribution as σ_i;
2. The expectation over S is irrelevant in the first term, since the term inside the expectation does not depend on S. Similarly, the expectation over S' is irrelevant in the second term.

Therefore

$$\begin{aligned}\mathrm{E}_{S,S',\sigma}\left[\sup_{f\in\mathcal{F}}\frac{1}{m}\sum_i \sigma_i f(z_i')\right] &= \mathrm{E}_{S',\sigma}\left[\sup_{f\in\mathcal{F}}\frac{1}{m}\sum_i \sigma_i f(z_i')\right] \\ &= \mathrm{E}_{S'}\left[\mathrm{E}_{\sigma}\left[\sup_{f\in\mathcal{F}}\frac{1}{m}\sum_i \sigma_i f(z_i')\right]\right] \\ &= R_m(\mathcal{F})\end{aligned}$$

and, similarly

$$\mathrm{E}_{S,S',\sigma}\left[\sup_{f\in\mathcal{F}}\frac{1}{m}\sum_i (-\sigma_i) f(z_i)\right] = R_m(\mathcal{F}).$$

Step 4 We have thus shown

$$\mathrm{E}_{S,S'}\left[\sup_{f\in\mathcal{F}}\left(\hat{\mathrm{E}}_{S'}\left[f\right]-\hat{\mathrm{E}}_{S}\left[f\right]\right)\right]\leq 2R_m(\mathcal{F}).$$

Chaining our results together, we obtain

$$\Phi(S)=\sup_{f\in\mathcal{F}}\left(\mathrm{E}\left[f\right]-\hat{\mathrm{E}}_{S}\left[f\right]\right)\leq 2R_m(\mathcal{F})+\mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}}\right).$$

We conclude that with probability at least $1-\delta$ and $\forall f\in\mathcal{F}$

$$\mathrm{E}\left[f\right]-\hat{\mathrm{E}}_{S}\left[f\right]\leq\Phi(S)\leq 2R_m(\mathcal{F})+\mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}}\right).$$

Therefore, with probability at least $1-\delta$ and $\forall f\in\mathcal{F}$

$$\mathrm{E}\left[f\right]\leq\hat{\mathrm{E}}_{S}\left[f\right]+2R_m(\mathcal{F})+\mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}}\right).$$

This is one of the results we were seeking. Proving the result for $\hat{R}_S(\mathcal{F})$ is just a matter of applying McDiarmid's inequality to obtain, with probability at least $1-\delta$

$$\hat{R}_S(\mathcal{F})\leq R_m(\mathcal{F})+\mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}}\right).$$

□

1 Motivation

The original motivation behind the theorem above was to obtain a relationship between generalization error and training error. We want to be able to say that, with probability at least $1-\delta$, $\forall h\in\mathcal{H}$

$$err(h)\leq\hat{err}(h)+\text{ small term.}$$

We note that $err(h)$ is evocative of $\mathrm{E}\left[f\right]$ and $\hat{err}(h)$ is evocative of $\hat{\mathrm{E}}_S\left[f\right]$, which appear in our theorem. Let us write

$$err(h)=\Pr_{(x,y)\sim D}\left[h(x)\neq y\right]=\mathrm{E}_{(x,y)\sim D}\left[\mathbf{1}\{h(x)\neq y\}\right]$$

$$\hat{err}(h)=\frac{1}{m}\sum_i\mathbf{1}\{h(x_i)\neq y_i\}=\hat{\mathrm{E}}_S\left[\mathbf{1}\{h(x)\neq y\}\right]$$

as per our definitions. We see that, to fit our definition, we must work with functions f which are indicator functions. Let us define

$$Z=X\times\{-1,+1\}$$

and for $h\in\mathcal{H}$:

$$f_h(x,y)=\mathbf{1}\{h(x)\neq y\}.$$

Now we can write:

$$\mathrm{E}_{(x,y)\sim D}\left[\mathbf{1}\{h(x) \neq y\}\right] = \mathrm{E}\left[f_h\right]$$

$$\hat{\mathrm{E}}_S\left[\mathbf{1}\{h(x) \neq y\}\right] = \hat{\mathrm{E}}_S\left[f_h\right]$$

$$\mathcal{F}_{\mathcal{H}} = \{f_h : h \in \mathcal{H}\}.$$

This allows us to use our theorem to state that:

With probability $\geq 1 - \delta$
$\forall h \in \mathcal{H}$

$$err(h) \leq e\hat{r}r(h) + 2R_m(\mathcal{F}_{\mathcal{H}}) + \mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}}\right)$$

$$err(h) \leq e\hat{r}r(h) + 2\hat{R}_S(\mathcal{F}_{\mathcal{H}}) + \mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}}\right).$$

We want to write the above in terms of the Rademacher complexity of $\mathcal{H}$, which we can do by looking at the definition of Rademacher complexity. We have

$$\hat{R}_S(\mathcal{F}_{\mathcal{H}}) = \mathrm{E}_\sigma\left[\sup_{f_h \in \mathcal{F}_{\mathcal{H}}} \frac{1}{m}\sum_i \sigma_i f_h(x_i, y_i)\right].$$

Now, our functions f_h are just indicator functions and can be written $f_h(x_i, y_i) = \frac{1-y_i h(x_i)}{2}$. Further, we are indexing each function by a function $h \in \mathcal{H}$. Therefore, we can just index the supremum with $h \in \mathcal{H}$ instead of $f_h \in \mathcal{F}_{\mathcal{H}}$. Writing this out gives

$$\begin{aligned}\hat{R}_S(\mathcal{F}_{\mathcal{H}}) &= \mathrm{E}_\sigma\left[\sup_{h\in\mathcal{H}} \frac{1}{m}\sum_i \sigma_i\left(\frac{1-y_i h(x_i)}{2}\right)\right] \\ &= \frac{1}{2}\mathrm{E}_\sigma\left[\frac{1}{m}\sum_i \sigma_i + \sup_{h\in\mathcal{H}} \frac{1}{m}\sum_i (-y_i\sigma_i)h(x_i)\right].\end{aligned}$$

Because σ_i is a Rademacher random variable, its expectation is just 0. For the second term, we note that because the sample S is fixed, the y_i's are fixed, and therefore the term $-y_i\sigma_i$ is distributed the same as σ_i. Hence, we conclude

$$\hat{R}_S(\mathcal{F}_{\mathcal{H}}) = 0 + \frac{1}{2}\mathrm{E}_\sigma\left[\sup_{h\in\mathcal{H}} \frac{1}{m}\sum_i \sigma_i h(x_i)\right] = \frac{1}{2}\hat{R}_S(\mathcal{H}).$$

We have therefore shown

$$err(h) \leq e\hat{r}r(h) + R_m(\mathcal{H}) + \mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}}\right)$$

$$err(h) \leq e\hat{r}r(h) + \hat{R}_S(\mathcal{H}) + \mathcal{O}\left(\sqrt{\frac{\ln 1/\delta}{m}}\right).$$

2 Obtaining other bounds

It was alluded to in class that obtaining the above bounds in terms of Rademacher complexity subsumes other bounds previously shown, which can be demonstrated with an example. We first state a simple theorem (a slightly weaker version of this theorem will be proved in a later homework assignment).

Theorem. *For* $|\mathcal{H}| < \infty$*:*

$$\hat{R}_S(\mathcal{H}) \leq \sqrt{\frac{2 \ln |\mathcal{H}|}{m}}.$$

Now consider again the definition of empirical Rademacher complexity:

$$\hat{R}_S(\mathcal{H}) = \mathrm{E}_\sigma \left[\sup_{h \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(x_i) \right].$$

We see that it only depends on how the hypothesis behaves on the fixed set S. We therefore have a finite set of behaviors on the set.

Define $\mathcal{H}' \subseteq \mathcal{H}$, where $\mathcal{H}'$ is composed of one representative from $\mathcal{H}$ for each possible labeling of the sample set S by $\mathcal{H}$. Therefore

$$|\mathcal{H}'| = |\Pi_{\mathcal{H}}(S)| \leq \Pi_{\mathcal{H}}(m).$$

Since the complexity only depends on the behaviors on S, we claim

$$\hat{R}_S(\mathcal{H}) = \mathrm{E}_\sigma \left[\sup_{h \in \mathcal{H}'} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(x_i) \right] = \hat{R}_S(\mathcal{H}').$$

We can now use the theorem stated above to write

$$\hat{R}_S(\mathcal{H}') \leq \sqrt{\frac{2 \ln |\Pi_{\mathcal{H}}(S)|}{m}}.$$

Finally, we recall that after proving Sauer's lemma, we showed $\Pi_{\mathcal{H}}(m) \leq \left(\frac{em}{d}\right)^d$, for $m \geq d \geq 1$. Therefore

$$\hat{R}_S(\mathcal{H}) \leq \sqrt{\frac{2d \ln \left(\frac{em}{d}\right)}{m}}.$$

We have thus used the Rademacher complexity results to get an upper bound for the case of infinite $|\mathcal{H}|$ in terms of VC-dimension.

3 Boosting

Up until this point, the PAC learning model we have been considering requires that we be able to learn to arbitrary accuracy. Thus, the problem we have been dealing with is:

Strong learning $\mathcal{C}$ is *strongly* PAC-learnable if

$\exists$ algorithm A
$\forall$ distributions $\mathcal{D}$
$\forall c \in \mathcal{C}$
$\forall \epsilon > 0$
$\forall \delta > 0$
A, given $m = poly\,(1/\epsilon, 1/\delta, \ldots)$ examples, computes h such that

$$\Pr\left[err(h) \leq \epsilon\right] \geq 1 - \delta.$$

But what if we can only find an algorithm that gives slightly better than an even chance of error (e.g. 40%)? Could we use it to develop a better algorithm, iteratively improving our solution to arbitrary accuracy? We want to consider the following problem:

Weak learning $\mathcal{C}$ is *weakly* PAC-learnable if

$\exists \gamma > 0$
$\exists$ algorithm A
$\forall$ distributions $\mathcal{D}$
$\forall c \in \mathcal{C}$
$\forall \delta > 0$
A, given $m = poly\,(1/\epsilon, 1/\delta, \ldots)$ examples, computes h such that

$$\Pr\left[err(h) \leq \frac{1}{2} - \gamma\right] \geq 1 - \delta.$$

We note that in this problem we no longer require arbitrary accuracy, but only that the algorithm picked be able to do slightly better than random guessing, with high probability. The natural question that arises is whether weak learning is equivalent to strong learning.

Consider first the simpler case of a fixed distribution $\mathcal{D}$. In this case, the answer to our question is no, which we can illustrate through a simple example.

Example: For fixed $\mathcal{D}$, define
$X = \{0,1\}^n \cup \{z\}$
$\mathcal{D}$ picks z with probability $1/4$ and with probability $3/4$ picks uniformly from $\{0,1\}^n$
$\mathcal{C} = \{$ all concepts over X $\}$.

In a training sample, we expect to see z with high probability, and therefore z will be correctly learned by the algorithm. However, the remaining points are exponential in m, so that with only $poly(1/\epsilon, 1/\delta, \ldots)$ number of examples, we are unlikely to do much better than even chance on the rest of the domain. We therefore expect the error to be given roughly by

$$err(h) \approx \frac{1}{2} \cdot \frac{3}{4} + 0 \cdot \frac{1}{4} = \frac{3}{8}$$

in which case $\mathcal{C}$ is weakly learnable, but not strongly learnable.

We wish to prove that in the general case of an arbitrary distribution the following theorem holds:

Theorem. *Strong and weak learning are equivalent under the PAC learning model.*

The way we will reach this result is by developing a *boosting algorithm* which constructs a strong learning algorithm from a weak learning algorithm.

3.1 The boosting problem

The challenge faced by the boosting algorithm can be defined by the following problem.

Boosting problem *Given:*
$(x_1, y_1), \ldots, (x_m, y_m)$ with $y_i \in \{-1, +1\}$
access to a weak learner A:
$\forall$ distributions D
given examples from D
computes h such that

$$\Pr\left[err_D(h) \leq \frac{1}{2} - \gamma\right] \geq 1 - \delta$$

Goal: find H such that with high probability $err_{\mathcal{D}}(H) \leq \epsilon$ for any fixed ϵ.

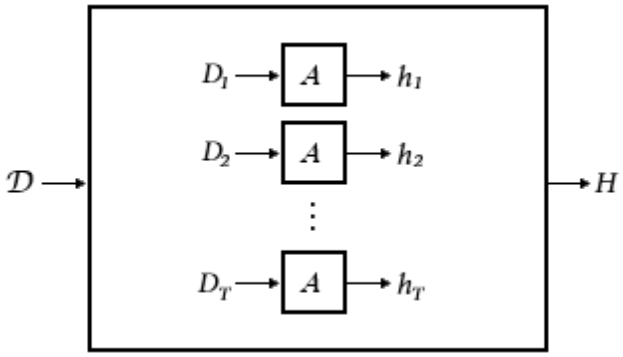

Figure 1: Schematic representation of boosting algorithm.

The main idea behind the boosting algorithm is to produce a number of different distributions D from $\mathcal{D}$, using the sample provided. This is necessary because running A on the same sample alone will not, in general, be enough to produce an arbitrarily accurate hypothesis (certainly so if A is deterministic). A boosting algorithm will therefore run as follows:

Boosting algorithm
for $t = 1, \ldots, T$
run A on D_t to get weak hypothesis $h_t : X \to \{-1, +1\}$
$\epsilon_t = err_{D_t}(h_t) = \frac{1}{2} - \gamma_t$, where $\gamma_t \geq \gamma$
end
output H, where H is a combination of the weak hypotheses $h_1, \ldots, h_T$.

In the above, the distributions D_t are distributions on the indices $1, \ldots, m$, and may vary from round to round. It is by adjusting these distributions that the boosting algorithm will be able to achieve high accuracy. Intuitively, we want to pick the distributions D_t such that, on each round, they provide us with more information about the points in the sample

that are "hard" to learn. The boosting algorithm can be seen schematically in Figure 1

Let us define: $D_t(i) = D_t(x_i, y_i)$. We pick the distribution as follows:

$$\forall i : D_1(i) = \frac{1}{m}$$

$$D_{t+1}(i) = \frac{D_t(i)}{Z_t} \cdot \begin{cases} e^{\alpha_t} & \text{if } h_t(x_i) \neq y_i \\ e^{-\alpha_t} & \text{if } h_t(x_i) = y_i \end{cases}$$

where $\alpha_t > 0$.

Intuitively, all our examples are considered equally in the first round of boosting. Going forward, if an example is misclassified, its weight in the next round will increase, while the weights of the correctly classified examples will decrease, so that the classifier will focus on the examples which have proven harder to classify correctly.