Machine Learning Theory (CS 6783)

Lecture 4: Rademacher Complexity and Finite Lemma

1 Back to Penny Matching, a Betting Version

We saw that via Cover's result, one could predict outcomes of arbitrary sequences and compete against any set of predictions $\mathcal{F} \subseteq \{\pm 1\}^n$ in hindsight with an additive factor of the Rademcher complexity of this set given by

$$\mathcal{R}_n(\mathcal{F}) = \frac{1}{n}\mathbb{E}_\epsilon\left[\sup_{f\in\mathcal{F}}\sum_{t=1}^{n} f_t\epsilon_t\right]$$

This set if interesting enough, it ensures that our payoff is never too negative against any adversary. But say we wanted to ensure that we dont make much lesser than what Shannon's machine would have done against an arbitrary opponent. It turns out that Shannon's machine was a 7 state, finite state machine. While we dont have access to this machine, one strategy would be to build a system that can compete with arbitrary 7 state automaton. Can we do this? Well we could define the corresponding ϕ to have the minimum over all such machines in hindsight, but the key bottleneck is that stability of such a ϕ is not guaranteed. However, if you recall the linear betting game, we did not need any such stability. Hence if we instead consider a betting version of the game where we try to match pennies and also place a bet of desired amount on the outcome, then it turns out we can use arbitrary ϕ. So lets consider this fancier version. Say $\mathcal{F} \subset \bigcup_{t=0}^{n-1}\{\pm 1\}^t \mapsto \mathbb{R}$. That is, each $f \in \mathcal{F}$ can take as input any sequence of up to length $n-1$ of binary labels and outputs a real number. The real number that is output should be viewed as follows, its magnitude is the amount the strategy f is suggesting we bet and the sign represents whether we bet on heads or tails. Now given this, we can use

$$\phi(y_1,\ldots,y_n) = \min_{f\in\mathcal{F}}\frac{1}{n}\sum_{t=1}^{n} -f(y_1,\ldots,y_{t-1})\cdot y_t + C_n(\mathcal{F})$$

That is, can we do as well as the best strategy for betting amongst the set of strategies $\mathcal{F}$ with an additional slack of $C_n(\mathcal{F})$. Since this is the betting game, we don't need stability any more. To answer the question of what is the smallest $C_n(\mathcal{F})$ needed to ensure existence of a successful learning algorithm we simply use the lemma for linear betting which just tells us that we need to ensure that $\mathbb{E}_\epsilon[\phi(\epsilon)] \geq 0$. Using this we immediately conclude that the optimal $C_n(\mathcal{F})$ is given by

$$C_n(\mathcal{F}) = -\mathbb{E}_\epsilon\left[\min_{f\in\mathcal{F}}\frac{1}{n}\sum_{t=1}^{n} -f(\epsilon_1,\ldots,\epsilon_{t-1})\cdot\epsilon_t\right] = \mathbb{E}_\epsilon\left[\max_{f\in\mathcal{F}}\frac{1}{n}\sum_{t=1}^{n}\epsilon_t f(\epsilon_1,\ldots,\epsilon_{t-1})\right]$$

Through this course we will use the short hand $x_{1:t}$ to represent the sequence $x_1,\ldots,x_t$. The above inspires us to define a new Rademacher complexity like term which is called sequential Rademacher

complexity given by

$$\mathcal{R}_n^{sq}(\mathcal{F}) = \frac{1}{n}\mathbb{E}_\epsilon\left[\sup_{f\in\mathcal{F}}\sum_{t=1}^{n}\epsilon_t f(\epsilon_{1:t-1})\right]$$

This is a strict generalization of Rademacher complexity since we if $\mathcal{F}$ consists of only function that only consider length of input and not the actual bits, we recover the classical Rademacher complexity.

2 Massart's Finite Lemma

Now a first key result we prove is that the sequential Rademacher complexity of a finite class of strategies can be bounded by order $O(\sqrt{\frac{\log|\mathcal{F}|}{n}})$. We will in fact prove a more precise lemma below (that will be useful later) that will imply our goal.

Lemma 1. *For any set $\mathcal{F} \subset \bigcup_{t=0}^{n-1}\{\pm 1\}^t \mapsto \mathbb{R}$:*

$$\mathcal{R}_n^{sq}(\mathcal{F}) = \frac{1}{n}\mathbb{E}_\epsilon\left[\max_{f\in\mathcal{F}}\sum_{t=1}^{n}\epsilon_t f(\epsilon_{1:t-1})\right] \le \frac{1}{n}\sqrt{2\max_{\epsilon\in\{\pm1\}^n}\max_{f\in\mathcal{F}}\left(\sum_{t=1}^{n} f(\epsilon_{1:t-1})^2\right)\log|\mathcal{F}|}$$

Proof.

$$\begin{aligned}
\max_{f\in\mathcal{F}}\sum_{t=1}^{n}\epsilon_t f(\epsilon_{1:t-1}) &= \frac{1}{\lambda}\log\left(\max_{f\in\mathcal{F}}\exp\left(\lambda\sum_{t=1}^{n}\epsilon_t f(\epsilon_{1:t-1})\right)\right)\\
&\le \frac{1}{\lambda}\log\left(\sum_{f\in\mathcal{F}}\exp\left(\lambda\sum_{t=1}^{n}\epsilon_t f(\epsilon_{1:t-1})\right)\right)\\
&= \frac{1}{\lambda}\log\left(\sum_{f\in\mathcal{F}}\prod_{t=1}^{n}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\right)
\end{aligned}$$

Taking expectation w.r.t. Rademacher random variables,

$$\mathbb{E}_\epsilon\left[\max_{f\in\mathcal{F}}\sum_{t=1}^{n}\epsilon_t f(\epsilon_{1:t-1})\right] \le \frac{1}{\lambda}\mathbb{E}_\epsilon\left[\log\left(\sum_{f\in\mathcal{F}}\prod_{t=1}^{n}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\right)\right]$$

Since log is a concave function, by Jensen's inequality, Expected log is upper bounded by log of expectation and so:

$$\begin{aligned}
&\leq \frac{1}{\lambda}\log\left(\mathbb{E}_\epsilon\left[\sum_{f\in\mathcal{F}}\prod_{t=1}^{n}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\right]\right)\\
&= \frac{1}{\lambda}\log\left(\sum_{f\in\mathcal{F}}\mathbb{E}_\epsilon\left[\prod_{t=1}^{n}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\right]\right)\\
&\leq \frac{1}{\lambda}\log\left(|\mathcal{F}|\times\max_{f\in\mathcal{F}}\mathbb{E}_\epsilon\left[\prod_{t=1}^{n}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\right]\right)\\
&= \frac{1}{\lambda}\log\left(|\mathcal{F}|\right)+\frac{1}{\lambda}\log\left(\max_{f\in\mathcal{F}}\mathbb{E}_\epsilon\left[\prod_{t=1}^{n}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\right]\right) \qquad (1)
\end{aligned}$$

Now for any fixed $f\in\mathcal{F}$, we would like to bound $\mathbb{E}_\epsilon\left[\prod_{t=1}^{n}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\right]$. To this end, note that:

$$\begin{aligned}
\mathbb{E}_\epsilon\left[\prod_{t=1}^{n}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\right] &= \mathbb{E}_\epsilon\left[\prod_{t=1}^{n-1}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\times\mathbb{E}_{\epsilon_n}\left[\exp\left(\lambda\epsilon_n f(\epsilon_{1:n-1})\right)\right]\right]\\
&= \mathbb{E}_\epsilon\left[\prod_{t=1}^{n-1}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\times\frac{\exp\left(\lambda f(\epsilon_{1:n-1})\right)+\exp\left(-\lambda f(\epsilon_{1:n-1})\right)}{2}\right]
\end{aligned}$$

Using the fact that for any x, $\frac{e^x+e^{-x}}{2}\leq e^{x^2/2}$,

$$\begin{aligned}
&\leq \mathbb{E}_\epsilon\left[\prod_{t=1}^{n-1}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\times\exp\left(\frac{\lambda^2 f(\epsilon_{1:n-1})^2}{2}\right)\right]\\
&\leq \mathbb{E}_\epsilon\left[\prod_{t=1}^{n-1}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\times\exp\left(\frac{\lambda^2 f(\epsilon_{1:n-1})^2}{2}\right)\right]\\
&\leq \mathbb{E}_\epsilon\left[\prod_{t=1}^{n-1}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\times\max_{\epsilon_{n-1}}\exp\left(\frac{\lambda^2 f(\epsilon_{1:n-1})^2}{2}\right)\right]\\
&= \mathbb{E}_\epsilon\left[\prod_{t=1}^{n-2}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\times\mathbb{E}_{\epsilon_{n-1}}\left[\exp\left(\lambda\epsilon_{n-1} f(\epsilon_{1:n-2})\right)\right]\times\max_{\epsilon_{n-1}}\exp\left(\frac{\lambda^2 f(\epsilon_{1:n-1})^2}{2}\right)\right]\\
&\leq \mathbb{E}_\epsilon\left[\prod_{t=1}^{n-2}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\times\exp\left(\frac{\lambda^2 f(\epsilon_{1:n-2})^2}{2}\right)\times\max_{\epsilon_{n-1}}\exp\left(\frac{\lambda^2 f(\epsilon_{1:n-1})^2}{2}\right)\right]\\
&\leq \mathbb{E}_\epsilon\left[\prod_{t=1}^{n-2}\exp\left(\lambda\epsilon_t f(\epsilon_{1:t-1})\right)\times\max_{\epsilon_{n-2},\epsilon_{n-1}}\left(\exp\left(\frac{\lambda^2 f(\epsilon_{1:n-2})^2}{2}\right)\times\exp\left(\frac{\lambda^2 f(\epsilon_{1:n-1})^2}{2}\right)\right)\right]
\end{aligned}$$

Proceeding similarly we get,

$$\begin{aligned}
&\leq \max_{\epsilon_1,\ldots,\epsilon_{n-1}} \prod_{t=1}^{n} \exp\left(\frac{\lambda^2 f(\epsilon_{1:t-1})^2}{2}\right) \\
&= \exp\left(\frac{\lambda^2 \max_{\epsilon_1,\ldots,\epsilon_{n-1}} \sum_{t=1}^{n} f(\epsilon_{1:t-1})^2}{2}\right)
\end{aligned}$$

Plugging this back in Equation 1 we get,

$$\begin{aligned}
\mathbb{E}_\epsilon\left[\max_{f\in\mathcal{F}} \sum_{t=1}^{n} \epsilon_t f(\epsilon_{1:t-1})\right] &\leq \frac{1}{\lambda}\log(|\mathcal{F}|) + \frac{1}{\lambda}\log\left(\exp\left(\frac{\lambda^2 \max_{f\in\mathcal{F}} \max_{\epsilon_1,\ldots,\epsilon_{n-1}} \sum_{t=1}^{n} f(\epsilon_{1:t-1})^2}{2}\right)\right) \\
&= \frac{1}{\lambda}\log(|\mathcal{F}|) + \frac{\lambda \max_{f\in\mathcal{F}} \max_{\epsilon_1,\ldots,\epsilon_{n-1}} \sum_{t=1}^{n} f(\epsilon_{1:t-1})^2}{2}
\end{aligned}$$

Choosing $\lambda = \sqrt{\frac{2\log|\mathcal{F}|}{\max_{f\in\mathcal{F}} \max_{\epsilon}\left(\sum_{t=1}^{n} f(\epsilon_{1:t-1})^2\right)}}$ completes the proof. □

To apply the lemma, lets try to complete with Shannon's machine or rather all possible K-state finite automaton. There are $(2K)^{2K}$ such machines in total. Let $\mathcal{F}$ represent the strategies of all such automaton. Note that $\mathcal{F}$'s will only bet one dollar on outcomes of games. Now we can do as well as Shannon's machine or any other such machine (although we might bet more money that just one dollar) with an additive factor of just

$$\frac{1}{n}\sqrt{2 \max_{\epsilon\in\{\pm1\}^n} \max_{f\in\mathcal{F}} \left(\sum_{t=1}^{n} f(\epsilon_{1:t-1})^2\right) \log|\mathcal{F}|} = \sqrt{\frac{2\log|\mathcal{F}|}{n}} = \sqrt{\frac{4K\log(2K)}{n}}$$