FoSML: Lecture #10

Recall our result from last time, which we now prove.

Proof of Proposition: By the union bound,

$$\mathbb{P}\left(\sup_{k \geq 1} \left| R(h_{S_1,k}^{ERM}) - \hat{R}_{S_2}(h_{S_1,k}^{ERM}) \right| > \varepsilon + \sqrt{\frac{\log k}{\alpha m}} \right)$$

$$= \sum_{k=1}^{\infty} \mathbb{P}\left(\left| R(h_{S_1,k}^{ERM}) - \hat{R}_{S_2}(h_{S_1,k}^{ERM}) \right| > \varepsilon + \sqrt{\frac{\log k}{\alpha m}} \right)$$

$$= \sum_{k=1}^{\infty} \mathbb{E}\left(\mathbb{P}\left(\left| R(h_{S_1,k}^{ERM}) - \hat{R}_{S_2}(h_{S_1,k}^{ERM}) \right| > \varepsilon + \sqrt{\frac{\log k}{\alpha m}} \;\middle|\; S_1 \right) \right) \quad (\star)$$

By conditioning on S_1, we fix $h_{S_1,k}^{ERM}$. Moreover, S_2 is independent from S_1

Then by Hoeffding,

$$\mathbb{P}\left(\left| R(h_{S_1,k}^{ERM}) - \hat{R}_{S_2}(h_{S_1,k}^{ERM}) \right| > \varepsilon + \sqrt{\frac{\log k}{\alpha m}} \;\middle|\; S_1 \right)$$

$$\leq 2\exp\left(-2\alpha m \left(\varepsilon + \sqrt{\frac{\log k}{\alpha m}}\right)^2\right)$$

$$\leq 2\exp\left(-2\alpha m \varepsilon^2 m - 2\log k\right)$$

$$= \frac{2\exp\left(-2\alpha m \varepsilon^2\right)}{k^2}$$

Now, summing over k in $(\star)$ yields

$$\mathbb{P}\left(\sup_{k\geq 1} \left|R(h_{s_1,k}^{ERM}) - \hat{R}_{s_2}(h_{s_1,k}^{ERM})\right| > \varepsilon + \sqrt{\frac{\log k}{2m}}\right)$$

$$\leq \frac{\pi^2}{3} \cdot \exp(-2\alpha m \varepsilon^2)$$

$$\leq 4\exp(-2\alpha m \varepsilon^2). \blacksquare$$

- Our main theoretical result for cross validation is to show that the estimator learned from cross validation is not much worse than what you get from SRM. This is encouraging, because SRM is great in theory, but hard to implement in practice. CV is much easier to implement practically.

Theorem (CV compared to SRM): For any $\delta > 0$, the following holds with probability exceeding $1-\delta$:

$$R(h_s^{cv}) - R(h_{s_1}^{SRM}) \leq 2\sqrt{\frac{\log(\max\{k(h_s^{cv}), k(h_{s_1}^{SRM})\})}{\alpha m}} + 2\sqrt{\frac{\log(4/\delta)}{2\alpha m}}.$$

Proof: Note that

$$R(h_s^{cv}) \leq \hat{R}_{s_2}(h_s^{cv}) + \sqrt{\frac{\log(k(h_s^{cv}))}{\alpha m}} + \sqrt{\frac{\log(4/\delta)}{2\alpha m}} \quad \text{holds with}$$

probability exceeding $1-\delta$ by our above proposition (just solve for ε in $\delta = 4\exp(-2\alpha m \varepsilon^2)$). Then since h_s^{cv} minimizes the empirical error on S_2, we have

$$\hat{R}_{S_2}(h_S^{CV}) + \sqrt{\frac{\log(k(h_S^{CV}))}{\alpha m}} + 2\sqrt{\frac{\log(4/\delta)}{2\alpha m}}$$

$$\leq \hat{R}_{S_2}(h_{S_1}^{SRM}) + \sqrt{\frac{\log(k(h_S^{CV}))}{\alpha m}} + 2\sqrt{\frac{\log(4/\delta)}{2\alpha m}}$$

$h_{S_1}^{SRM}$ by learning bound for SRM

$$\leq R(h_{S_1}^{SRM}) + \sqrt{\frac{\log(k(h_{S_1}^{SRM}))}{\alpha m}} + \sqrt{\frac{\log(k(h_S^{CV}))}{\alpha m}} + 2\sqrt{\frac{\log(4/\delta)}{2\alpha m}}$$

$$\leq R(h_{S_1}^{SRM}) + 2\sqrt{\frac{\log\{\max\{k(h_S^{CV}), k(h_{S_1}^{SRM})\}\}}{\alpha m}} + 2\sqrt{\frac{\log(4/\delta)}{2\alpha m}}. \blacksquare$$

- So, as long as $(1-\alpha)$ is large enough so that SRM on $(1-\alpha)m$ points (i.e. on S_1) gives a good result, while α is large enough so that the $\alpha^{-1/2}$ penalties on the RHS don't hurt too much, we get a good result/bound for CV.

- In practice, we may not have m large enough to balance these two conditions, so we use n-fold cross-validation, in which parameters for an algorithm are selected by randomly partitioning the data into training and validation, finding parameters error via cross validation, then re-partitioning the data. ↑ n

- The more times this split is done (i.e. the more folds), the more costly it is. So, it is often the case that 5-fold cross validation is used

- SRM says "pick a predictor that fits the data well, while also being simple." This is an example of regularization, a ubiquitous method in statistics and machine learning.

- Let $\mathcal{H} = \bigcup_{\gamma > 0} \mathcal{H}_\gamma$ be a nested decomposition of hypothesis classes. For example,

$\mathcal{H}_\gamma = \{ x \mapsto w \cdot \Phi(x) \mid \|w\| \leq \gamma \}$ for some function Φ mapping into a high dimensional space (this is closely related to kernel methods, which we shall discuss later)

- Here, we are looking at an uncountable sum, since $\gamma \in (0, \infty)$, but we can still consider SRM: $\underset{\gamma > 0,\, h \in \mathcal{H}_\gamma}{\arg\min} \left(\hat{R}_S(h) + \tilde{R}_m(\mathcal{H}_\gamma) + \sqrt{\frac{\log \gamma}{m}} \right)$.

- We can consider more general penalties for complexity than $\tilde{R}_m(\mathcal{H}_\gamma) + \sqrt{\frac{\log \gamma}{m}}$.

- Indeed, let $R : \mathcal{H} \to \mathbb{R}$ be s.t. $\underset{\gamma > 0,\, h \in \mathcal{H}_\gamma}{\arg\min} \hat{R}_S(h) + \underbrace{\text{pen}(\gamma, m)}_{\text{arbitrary penalty}}$

$$= \underset{h \in \mathcal{H}}{\arg\min} \hat{R}_S(h) + \lambda R(h)$$

for some $\lambda > 0$. The second formulation should be familiar to those familiar with signal processing, and we will look at several such regularized problems towards the end of the course.

- One does have to take care with how to solve regularized statistical learning problems in practice, since the problems are often non-convex, and thus hard to solve computationally.