CS281B/Stat241B (Spring 2008) Statistical Learning Theory **Lecture: 20**

Model Selection

Lecturer: Peter Bartlett *Scribe: Christopher Hundt*

1 Model Selection

1.1 Setup

Consider a prediction problem P on $\mathcal{X} \times \mathcal{Y}$: given examples $(X_1, Y_1), \ldots, (X_n, Y_n)$, the goal is to choose a prediction function f from some class F to minimize the prediction error $R(f) := \mathbb{E}[l(Y, f(X))]$. We attempt to approximate this choice by choosing the Empirical Risk Minimizer (ERM), defined as that function $\hat{f}$ which minimizes $\hat{R}(f)$ over F.

1.2 Approximation error versus estimation error

When choosing the class F in which to look for a prediction function, we consider the excess risk that choosing the ERM from a given class F has over the Bayes risk:

$$R(\hat{f}) - R^* = \left(R(\hat{f}) - \inf_{f \in F} R(f) \right) + \left(\inf_{f \in F} R(f) - R^* \right)$$

In the above decomposition, we call the first term the *estimation error* and the second term the *approximation error*. There is a tradeoff between these two terms: a richer class F tends to cause a greater estimation error but a smaller approximation error. So the balancing act that model selection attempts is to ensure that

1. F is large enough to make the approximation error small, and
2. F is small enough to make the estimation error small.

One way to try to choose a function class is to start with an F with zero approximation error and split it into a hierarchy of increasingly complex subsets.

Example. The following function classes have zero approximation error, for example, in the setting of minimizing expected squared error on a compact subset of $\Re^n$.

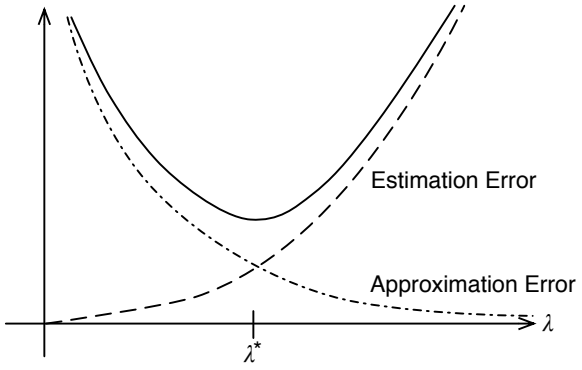
- Regression trees without size constraints
- span(G) for a suitably large base class G
- RKHSs where the kernel matrix K_n has full rank for all distinct values $x_1, \ldots, x_n$

1.3 Splitting a function class

Suppose we have a function class F. We could split it into

$$F = \bigcup_{\lambda \in I} F_\lambda$$

where I is some index set and $F_\lambda \subseteq F$ for all λ. Frequently the splitting is increasing: $\lambda \geq \lambda' \implies F_{\lambda'} \subset F_\lambda$. Then, as λ increases, the complexity of the class increases, meaning that the estimation error goes up and the approximation error goes down. The ideal choice of λ minimizes the sum of the two error terms. In the diagram below, the solid line shows the sum of the error terms, and λ^* is the optimal choice of λ:



1.4 Regularization

Of course, to draw a graph like the one above for a real set of function classes would require knowing the approximation error and estimation error for $\hat{f}_\lambda$ as functions of λ. Then we could optimize

$$R(\hat{f}_\lambda) = \hat{R}(\hat{f}_\lambda) + \left(R(\hat{f}_\lambda) - \hat{R}(\hat{f}_\lambda) \right),$$

Where $\hat{f}_\lambda$ is the ERM within F_λ. Since in general we don't know the correction term in parentheses above, we attempt to find an approximate solution by using regularization: we pick a function $C_n(\lambda)$ which we hope will approximate $R(\hat{f}_\lambda) - \hat{R}(\hat{f}_\lambda)$. Then, conceptually, we choose $f = \hat{f}_{\hat{\lambda}}$, where

$$\hat{\lambda} = \operatorname*{argmin}_\lambda \left(\hat{R}(\hat{f}_\lambda) + C_n(\lambda) \right).$$

This would require finding $\hat{f}_\lambda$ for all the (possibly infinitely many) values of λ. So we often reduce the problem to a single optimization, reformulating it as

$$\hat{f}_{\hat{k}} = \operatorname*{argmin}_{f \in F} \left(\hat{R}(f) + C_n(\lambda(f)) \right),$$

where $\lambda(f) = \min\{\lambda : f \in F_\lambda\}$.

Frequently, $C_n(\lambda)$ will be a (high-probability) bound on $R(\hat{f}_\lambda) - \hat{R}(\hat{f}_\lambda)$, giving a near-optimality guarantee in lieu of a promise of the ideal choice of f.

Example: Structural Risk Minimization. If $F \subseteq \{-1, 1\}^{\mathcal{X}}$ and the loss function is 0-1 loss then, with high probability,

$$\sup_{f \in F_\lambda} |R(f) - \hat{R}(f)| \leq C_n(\lambda),$$

where

$$C_n(\lambda) = c\sqrt{\frac{d_{\mathrm{VC}}(F_\lambda)}{n}} \qquad \text{or} \qquad C_n(\lambda) = c'\sqrt{\frac{d_{\mathrm{VC}}(F_\lambda)\log n}{n}}.$$

1.5 Oracle Inequality

Suppose the model class F is split countably into $F_1, F_2, \ldots$ and we have a risk estimator $r_{n,k}$ that, for some choice of b and c, satisfies

$$P\left(R(\hat{f}_k) > r_{n,k} + \varepsilon\right) \le ce^{-2bn\varepsilon^2} \text{ for all } n,\ k, \text{ and } \varepsilon. \tag{1}$$

Let

$$C_n(k) = r_{n,k} - \hat{R}(\hat{f}_k) + \sqrt{\frac{\log k}{bn}}. \tag{2}$$

Then choosing $\hat{k}$ to minimize $\hat{R}(\hat{f}_k) + C_n(k)$ will be the same as picking

$$\hat{k} = \operatorname*{argmin}_k \left(r_{n,k} + \sqrt{\frac{\log k}{bn}} \right).$$

Theorem 1.1. If (1) is satisfied and $C_n(k)$ is defined as in (2) then

$$\mathbb{E}\left[R(\hat{f}_{\hat{k}})\right] - R^* \le \inf_k \left(\inf_{f \in F_k} (R(f) - R^*) + \mathbb{E}\left[C_n(k)\right] \right) + \sqrt{\frac{\log(2ce)}{2bn}}.$$

Example: Hold-out set. Suppose

$$(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n), (X_1', Y_1'), (X_2', Y_2'), \ldots, (X_{bn}', Y_{bn}')$$

are i.i.d. and we use the (X_i, Y_i) as the training data and the (X_i', Y_i') as the test data for a model selection problem. Define

$$r_{n,k} = \frac{1}{bn} \sum_{i=1}^{bn} 1_{\hat{f}_k(X_i') \neq Y_i'},$$

i.e., the empirical risk of $\hat{f}$ on the test set. Then, since the indicators summed above are i.i.d. in $[0,1]$, each with expectation $R(\hat{f}_k)$,

$$P\left(R(\hat{f}_k) > r_{n,k} + \varepsilon\right) \le e^{-2bn\varepsilon^2},$$

by Hoeffding, satisfying (1) with $c = 1$. Then Theorem 1.1 implies

$$\mathbb{E}\left[R(\hat{f}_{\hat{k}})\right] - R^* \le \min_k \left(\inf_{f \in F_k} (R(f) - R^*) + \mathbb{E}\left[r_{n,k} - \hat{R}(\hat{f}_k)\right] + \sqrt{\frac{\log k}{bn}} \right) + \frac{1}{\sqrt{bn}}.$$

That is, if you define $\hat{f}_k$ as the ERM of the training set for each k and then choose $\hat{k}$ to minimize the empirical risk of $\hat{f}_k$ on the test set, penalized by $(\log k/bn)^{1/2}$, then the above risk bound applies to $\hat{f}_{\hat{k}}$.

Also note that although the analysis in this example may appear to extend to cross-validation, additional difficulties arise in finding error bounds in that case due in part to fact that the final classifier is trained on the entire data set, including the data used to validate the choice of $\hat{k}$.

Example: Structural Risk Minimization. With $F \subseteq \{-1,1\}^{\mathcal{X}}$ and 0-1 loss,

$$r_{n,k} = \hat{R}(\hat{f}_k) + \sqrt{\frac{d_{\mathrm{VC}}(F_k)\log(2n+1) + \log 4}{n}}$$

satisfies (1) with $b = 1/2$ and $c = 1$, so

$$\mathbb{E}\left[R(\hat{f}_{\hat{k}})\right] - R^* \leq \min_k \left(\inf_{f \in F_k} (R(f) - R^*) + \sqrt{\frac{d_{\mathrm{VC}}(F_k)\log(2n+1) + \log 4}{n}} + \sqrt{\frac{2\log k}{n}} \right) + \frac{1}{\sqrt{n}}.$$

Example: Rademacher Averages. Recall

$$P\left(\sup_{f \in F} \left(R(f) - \hat{R}(f) \right) > \hat{R}_n(F) + \varepsilon \right) \leq e^{-2cn\varepsilon^2}.$$

Then letting

$$r_{n,k} = \hat{R}(\hat{f}_k) + \hat{R}_n(F_k)$$

gives

$$\mathbb{E}\left[R(\hat{f}_{\hat{k}})\right] - R^* \leq \min_k \left(\inf_{f \in F_k} (R(f) - R^*) + R_n(F_k) + \sqrt{\frac{\log k}{n}} \right) + \frac{1}{\sqrt{n}}.$$

Note that local Rademacher averages can be used to get a better bound than the one provided by Theorem 1.1. See *Local Rademacher Complexities*, by Bartlett, Bousquet, and Mendelson, for more.