CS281B/Stat241B (Spring 2006) Statistical Learning Theory **Lecture: 24**

Pseudo-dimension and Model Selection

Lecturer: Peter Bartlett *Scribe: Slav Petrov*

1 Estimates for the Pseudo-Dimension

Continued from last lecture.

Estimates for the pseudo-dimension are easy to obtain.

The pseudo-dimension is a straight froward extension of the VC-dimension to real valued functions. It is not limited to functions with values $\{0, 1\}$.

Recall the definition from last class:

Definition. (*Pseudo-dimension*) The pseudo-dimension of $\mathcal{F} \subseteq \mathbb{R}^{\mathcal{X}}$ is:

$$d_P(\mathcal{F}) = d_{VC}(\{(x, y) \mapsto \text{sign}(f(x) - y) : f \in \mathcal{F}\}).$$

Hence, if we can compute functions from the class

$$F = \{x \mapsto f(\theta, x) : \theta \in \mathbb{R}^d\}$$

in no more than t arithmetic operations, then we can compute

$$d_P(\mathcal{F}) = d_{VC}(\{(x, y) \mapsto \text{sign}(f(x) - y) : f \in \mathcal{F}\})$$

in $O(t)$ steps. Therefore $d_P(F) = O(dt)$.

2 Model Selection

Consider a prediction problem: P on $(\mathcal{X} \times \mathcal{Y})$.

We see examples $(X_1, Y_1), (X_2, Y_2), \ldots (X_n, Y_n)$ and we want to choose a prediction rule

$$f \in \mathcal{F}, \text{ s.t. } L(f) = \mathbb{E}\left[\ell(Y, f(X))\right]$$

is near minimal.

We would like $\mathcal{F}$ to be a rich set (unless we know something in advance about the underlying distribution). The problem is that, the richer $\mathcal{F}$, the harder it is to find f.

We can choose the Empirical Risk Minimizer (ERM): $\hat{f} \in \mathcal{F}$, such that $\hat{L}(f)$ is minimal.

2.1 Approximation-estimation trade-off

$$L(\hat{f}) - L^* = \underbrace{\left(L(\hat{f}) - \inf_{f\in\mathcal{F}} L(f)\right)}_{\text{estimation}} + \underbrace{\left(\inf_{f\in\mathcal{F}} L(f) - L^*\right)}_{\text{approximation}}$$

The bigger the class $\mathcal{F}$, the better the approximation and the worse the estimation.

When doing model selection, we would like to take a big class $\mathcal{F}$ and split it up in a hierarchy, from which we can choose a model of appropriate complexity. Typically this is done by regularization.

Split $\mathcal{F}$ into

$$\mathcal{F} = \bigcup_{k=1}^{\infty} \mathcal{F}_k \quad \text{or} \quad F = \bigcup_{\lambda>0} \mathcal{F}_\lambda,$$

where the $\mathcal{F}_\lambda$ are balls of radius λ in some Reproducing Kernel Hilbert Space (RKHS).

Example.

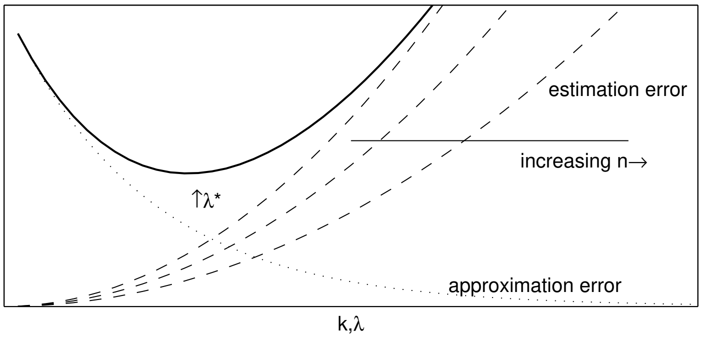
Consider the quadratic loss function $\ell(Y, \hat{Y}) = (Y - \hat{Y})^2$. Let $\mathcal{X}$ be a compact subset of $\mathbb{R}^d$. $L^* = L(f^*)$ with smooth $f^* : \mathbb{R}^d \to [-B, B]$.

Then the following classes have zero approximation error.

- 2-layer neural nets: $\sum \alpha_i \sigma_i(x)$.
- Regression trees (no constraints on size).
- Certain infinite dimensional RKHSs (it suffices if for distinct $x_1, x_2, \dots x_n$, the kernel matrix K_n has full rank).

Assume a class with zero approximation error.

We need to use a regularization/complexity penalty to compensate for this overfitting effect.



Regularization:

Pick $f \in \mathcal{F}$ to minimize $\hat{L}(f) + C_n(f)$, e.g. $\hat{L}(f) + C_n(f) = \hat{L}(f) + C_n(\lambda(f))$.

One can view this as doing two optimizations: For each k choose $\hat{f}_k = \operatorname{argmin}_{f\in\mathcal{F}} \hat{L}(f)$. Then choose $\hat{k} = \operatorname{argmin}_k \hat{L}(\hat{f}_k) + C_n(k)$. We want $L(\hat{f}_{\hat{k}}) - L^*$ close to

$$\min_k L(\hat{f}_k) - L^* = \min_k \left(\left(L(\hat{f}_k) - \inf_{f\in\mathcal{F}_k} L(f) \right) + \left(\inf_{f\in\mathcal{F}_k} L(f) - L^* \right) \right)$$

To be continued in the next lecture.