ESE 680-004: Learning and Control **Fall 2019**

Lecture 15: Most of Statistical Learning theory

Lecturer: Nikolai Matni *Scribes: Shuo Li*

1 A Simple Problem

Before diving into statistical learning theory, we first setup a problem our learner might be faced with. The book used papaya example to illustrate its idea, while we would use a robotics example. Since analysis in the book mainly focuses on binary classification problem, we would also start from a binary classification problem in robotics.

Suppose that we want to train a learner to detect anomaly of an vehicle engine. We have features like engine's sound, weight, temperature and outside weather. After doing some experiments, we collected some training data of different engine states with corresponding features. The goal is to get a model which could accurately detect anomaly basing on such features using training data we have.

In the example above, the statistical learning theory aims to formally measure how well our model could perform on data we have not seen before, or generalize to testing data.

2 The Statistical Learning Framework

In this part, we set up a formal model on which we build our analysis.

1. **The model's input:**
   (a) **Domain set:** An arbitrary set, $\mathcal{X}$. This is the set of objects that we may wish to label. For instance, in the engine anomaly detection case, the domain set is all combinations of different features, including sound, weight, temperature and weather. These features could be real values or logical integers.
   (b) **Label Set**: The set of possible labels, $\mathcal{Y}$. For our current discussion, we will restrict the label set to be a two-element set, i.e.$\{-1, 1\}$ or $\{0, 1\}$. In our case, we define our label set as $\{0, 1\}$ where 0 means normal state, 1 means anomaly.
   In other problems, including multi-classification or regression, the label set could also be sets of logical integers or real values.
   (c) **Training Set:** $S = \{(x_1, y_1) \ldots, (x_m, y_m)\}$ is a finite sequence of pairs in $\mathcal{X} \times \mathcal{Y}$: that is, a sequence of labeled domain points. In our engine case, the training set is the (features, engine states) training dataset we collected through experiments.
2. **The Model's Output:** The learner is requested to output a prediction rule, $f : \mathcal{X} \to \mathcal{Y}$. This function is also called a predictor, a hypothesis, or a classifier (or regressor in regression problems). The predictor can be used to predict the label of new domain points. In our case, the predictor is the one trained upon training set. We are going to use this model to detect anomaly. We use $A(S)$ to denote the hypothesis (predictor) that a learning algorithm, A, returns upon receiving the training sequence S.
3. **The Generative Model in Nature:** We assume that the instances are generated by some probability distribution. Using Bayesian Rule, we can write the generative probability as $p(x, y) = p(x) * p(y|x)$, where $x \in \mathcal{R}^4$ is the feature vector, y is the engine state. Let us denote that probability over $\mathcal{X}$ by $\mathcal{D}$.

It is important to note that we do not assume that the learner knows anything about this distribution. Actually, our predictor is expected to learn this probability through training data. We denote the generative model as $g : \mathcal{X} \to \mathcal{Y}$, and that $y_i = f(x_i)$ for all i. In our engine case, g is the function that project our input features onto different engine states.

4. **Measures of success:** We define the error of a classifier (predictor) to be the probability that it does not predict the correct label on a random data point generated by the aforementioned underlying distribution. In our engine case, the measure of success is the probability that the prediction is correct basing on input features using the trained predictor.

   Formally, given a domain subset, $A \subset \mathcal{X}$, the probability distribution $\mathbb{D}$, assigns a number, $\mathcal{D}(A)$, which determines how likely it is to observe a point. In many cases, we refer to A as an event and express it using a function $\pi : \mathcal{X} \to \{0, 1\}$, namely, $A = \{x \in \mathcal{X} : \pi(x) = 1\}$. In that case, we also use the notation $\mathbb{P}_{\mathbf{x}\sim\mathcal{D}}[\pi(x)]$ to express $\mathcal{D}(A)$.

   We define the error of a prediction rule: $h : \mathcal{X} \to \mathcal{Y}$, to be

   $$Loss_{\mathcal{D}}(f(x), y) \equiv \mathbb{P}_{x\sim\mathcal{D}}[h(x) \neq f(x)] \equiv \mathcal{D}(\{x : f(x) \neq y\}) \tag{1}$$

   That is, the error of such h is the probability of randomly choosing an example x for which $h(x) \neq f(x)$. The subscript $(\mathcal{D}, f)$ indicates that the error is measured with respect to the probability distribution $\mathcal{D}$ and the correct labeling function f.

5. **A Note About the Information Available to the Learner:** The learner is blind to the underlying distribution $\mathcal{D}$ over the world and to the labeling function f. In our engine example, we have no knowledge about how to bind engine states with its sound, weight, temperate and outside weather. The only way the learner can interact with the environment is through collected training dataset.

3 Empirical Risk Minimization

As mentioned earlier, a learning algorithm receives as input a training set S, sampled from an unknown distribution $\mathcal{D}$ and labeled by some target function g, and should output a predictor $f_S : \mathcal{X} \to \mathcal{Y}$. The goal of the predictor is to find f_S that could achieve the lowest prediction error with respect to the unknown $\mathcal{D}$ and g.

However, since the learner does not know what $\mathcal{D}$ and g are, the **empirical error** or **empirical risk** which could be calculated by the learner is often used as the proxy of the true error. The error on training error is as

$$Loss_S(f_S(x), y) \equiv \frac{|\{i \in [m] : f_S(x_i) \neq y_i\}|}{m} \tag{2}$$

where $[m] = \{1, \ldots, m\}$.

Since the training dataset is the snapshot of the world that is available to the learner, it makes sense to search for a solution that works well on that data. The learning paradigm - coming up with a predictor h that minimizes $L_S(f_S(x), y)$ - is called *Empirical Risk Minimization* or ERM for short.

Note that, in classification case, the empirical risk is the error rate on the training dataset. In regression problems, the empirical risks could be formalized as the Mean Squared Error loss on training dataset, i.e.

$$L_S(f_S(x), y) = \frac{1}{m}\sum_{i=1}^{m} \|f_S(x_i) - y_i\|^2$$

4 True Risk V.S. Empirical Risk: Overfitting

Although the empirical risk minimization algorithm seems reasonable, the fact that the training dataset might be non representative of the true distribution may cause the predictor learned upon the training

dataset to overfit. That is, the predictor might perform well or perfectly (i.e. achieve zero empirical loss on training dataset), but poorly on the true distribution.

For instance, consider the following predictor:

$$f_S(x) = \begin{cases} y_i & \text{if } \exists i \in [m] \text{ } s.t. \text{ } x_i = x \\ 0 & \text{otherwise} \end{cases}$$

Clearly, no matter what the sample is, $L_S(h_S) = 0$, and therefore this predictor may be chosen by an ERM algorithm (it is one of the empirical-minimum-cost predictor; no predictor can have smaller error). On the other hand, since the predictor almost always predict 0 for any x, the true error rate of this predictor is approximately 1/2. Thus, $L_{\mathcal{D}} = 1/2$. We have found a predictor whose performance on the training dataset is excellent yet its performance on the true distribution is very poor (as poor as random predictor). This phenomenon is called *overfitting*.

Formally, we could also decouple the true risk as follows:

$$L_{\mathcal{D}}(f_S(x), y) = L_{\mathcal{D}}(f_S(x), y) - L_{\mathcal{S}}(f_S(x), y) + L_{\mathcal{S}}(f_S(x), y)$$

We define the term $L_{\mathcal{D}}(f_S(x), y) - L_{\mathcal{S}}(f_S(x), y)$ as *generalization risk*, while the second term is actually *empirical risk*.

5 The Independently and Identically Distributed (i.i.d) Assumption

Clearly, any guarantee on the error with respect to the underlying true distribution, $\mathcal{D}$, for an algorithm that has access only to a sample S should depend on the relationship between $\mathcal{D}$ and S. The common assumption in statistical machine learning is that the training sample S is generated by sampling points from the distribution $\mathcal{D}$ independently of each other. Formally, we define i.i.d assumption as

- **The i.i.d assumption:** The examples in the training set are independently and identically distributed according to the distribution $\mathcal{D}$. That is, every x_i in S is freshly sampled according to $\mathcal{D}$ and then labeled according to the generative function, f. We denote this assumption by $S \sim \mathcal{D}^m$ where m is the size of S, and $\mathcal{D}^m$ denotes the probability over m-tuples induced by applying $\mathcal{D}$ to pick each element of the tuple independently of the other members of the tuple.

By using the i.i.d assumption, we could get that

$$\mathbb{E}[L_S(f_S(x), y)] = L_{\mathcal{D}}(f_S(x), y)$$

In other words, the expectation of empirical loss equals to the true loss. As a result, concentration bounds may help to compute the bound on generalization risk.

Actually, because of the i.i.d assumption, we can say **Learning theories tries to formalize the intuition *if there's enough data, this (empirical risk) should approximate the true risk.***

6 Hoeffding's Inequality and McDiarmid's Inequality

6.1 Hoeffding's Inequality

Hoeffding's inequality is a powerful technique—perhaps the most important inequality in learning theory—for bounding the probability that sums of bounded random variables are too large or too small.

Hoeffding's Inequality: For a random variable X, with $\mathbb{E}(X) = 0$ and $a \leq X \leq b$,then for $s > 0$

$$\mathbb{E}(e^{sX}) \leq e^{s^2(b-a)^2}/8$$

6.2 McDiarmid's Inequality [1]

Let$Z_1, \ldots, Z_n$ be independent random variables taking values in $\mathcal{Z}$. Suppose that $F : \mathcal{Z}^n \to \mathbb{R}$ satisfies

$$\sup_{(z_1,\ldots,z_n),\hat{z}_i} |F(z_1, \ldots, z_n) - F(z_1, \ldots, z_{i-1}, \hat{z}_i, z_{i+1}, \ldots, z_n)| \leq c_i$$

Then,

$$\mathbb{P}[F(Z_1, \ldots, Z_n) - \mathbb{E}[F] \geq t] \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^n c_i^2}\right) \tag{3}$$

Hoeffding's Inequality is a special case of McDiarmid's Inequality. We require that the random variables $Z_1, ..., Z_n$ are i.i.d, and $Z_i \in [0, 1]$. Then, we define $f(Z_1, \ldots, Z_n) = \frac{1}{m}\sum_{i=1}^n Z_i$. Note that because the Z_i's are constrained to be either 0 or 1, changing one of these values will change the value of $f(Z_1, \ldots, Z_n)$ by at most $\frac{1}{m}$. So, we set $c_i = \frac{1}{m}$ in McDiarmid's Inequality to get the required result.

7 Generalization Bound

In this section, we want to get the upper bound of generalization. Specifically, given a distribution $\mathcal{X}$, a finite collection of possible prediction functions $\mathcal{F}$, and a sample set sampled from $\mathcal{X}$, we want to get the upper true loss bound of a given predictor $f \in \mathcal{F}$ over distribution $\mathcal{X}$, which we denote as $R[f]$.

7.1 Post-hoc Validation

Our idea is to get the generalization bound using McDiarmid's inequality. Firstly, we assume that losses are bounded, i.e. given a function (predictor) $f \in \mathcal{F}$,

$$|Loss_{\mathcal{S}}(f_S(x), y)| \leq B \text{ for all}(x, y)$$

Then, we define variable $Z_i = Loss_S(f_S(X_i), Y_i)$, and rewrite empirical loss as

$$F(z_1, \ldots, z_n) = -\frac{1}{n}\sum_{i=1}^n Loss_S(f_S(x_i), y_i) = -R_S[f]$$

Note that, true loss equals to the expectation of empirical loss, i.e.

$$\mathbb{E}[R_S(f)] = R[f]$$

Also, because the loss is bounded, we could use McDiarmid's Inequality to get the upper bound of generalization, i.e. let $c_i \equiv 2B/n$, $t = \epsilon$, we could get

$$\mathbb{P}[R[f] - R_S[f] \geq \epsilon] \leq \exp\left(-\frac{n\epsilon^2}{2B^2}\right) \tag{4}$$

By inverting 4, we could get that with probability at least $1 - \delta$ we have

$$R[f] \leq R_S[F] + B\sqrt{\frac{2\log(1/\delta)}{n}} \tag{5}$$

7.2 Uniform Convergence

By far, we are able to get the generalization bound for a given function (predictor) $f \in \mathcal{F}$. In general, we want to evaluate the generalization bound of an algorithm $\mathcal{A}$ over a sample set S. In this case, the variables $Z_i = loss(f_S(X_i), Y_i)$ are no longer independent (**because predictors would use the same samples**), which does not satisfy the assumption of McDiarmid's Inequality.

Therefore, instead of evaluating a specific function f_S output by algorithm $\mathcal{A}$, i.e. $A(S) = f_S$, we instead want to proof that the generalization upper bound over the entire hypothesis space is small, i.e. we want to bound

$$\mathbb{P}[\sup_{f \in \mathcal{F}}(R[f] - R_S[f]) \geq \epsilon]$$

or equivalently,

$$\mathbb{P}[\exists f \in \mathcal{F} : R[f] - R_S[f] \geq \epsilon]$$

7.2.1 Finite Function Class

We first consider the uniform convergence for finite function class. Suppose $\mathcal{F}$ is a finite collection of possible prediction functions, and assume that for all $f \in \mathcal{F}$, $x \in \mathcal{X}$, and $y \in Y$, $|\ell(f(x), y)| \leq B$. Using Ineq. 4, we could get that for a given function f,

$$\mathbb{P}[R[f] - R_S[f] \geq \epsilon] \leq \exp\left(-\frac{n\epsilon^2}{2B^2}\right)$$

By union bound, we could extend above inequality to the whole function space, i.e.

$$\mathbb{P}[\exists f\mathcal{F} : R[f] - R_S[f] \geq \epsilon] \leq |\mathcal{F}|\mathbb{P}[R[f] - R_S[f]\epsilon] \tag{6}$$

$$\leq |\mathcal{F}| \exp\left(-\frac{n\epsilon^2}{2B^2}\right) \tag{7}$$

where $|\mathcal{F}|$ is the number of possible functions. By inverting above inequality, we could get

$$R[f] \leq R_S[f] + B\sqrt{\frac{2\log(|\mathcal{F}/\delta|)}{2n}} \ \forall f \in \mathcal{F}$$

7.2.2 Infinite Function Class

We start from threshold functions to study the generalization bound for infinite function class. Firstly, in our example, the threshold function is defined as

$$F := f_a : f_a(x) = \mathbb{I}(x < a), a \in \mathbb{R}$$

Obviously, $|\mathcal{F}| = \infty$, where in fact $\mathcal{F}$ is uncountably infinite. Now consider a random variable $x \sim p(x)$, and assume that $X \in (-\infty, a_*\}$, for some unknown a_*. Define the loss function to be

$$loss_S(f_a(x), x) = 1 - f_a(x) = \begin{cases} 0 & \text{if } x < a \\ 1 & \text{otherwise} \end{cases}$$

Then, $R[f_a] = \mathbb{E}_x[1 - f_a(x)] = \mathbb{P}[x \in [a, a_*]]$, and $R[f_{a_*}] = 0$. The empirical risk is given by $R_S[f_a] = \frac{1}{n}\sum_{i=1}^{n} 1 - f_a(x_i)$. We then sample $S = (x_1, \ldots, x_n)$ and set $\hat{a} = max_i x_i$. Note that by construction, $\hat{a}$ is an empirical risk minimizer achieving $R_S[f_{\hat{a}}] = 0$.

We want to bound $\mathbb{P}[R[f_{\hat{a}}] - R_S[f_{\hat{a}}] \geq \epsilon] = \mathbb{P}[R[f_{\hat{a}} \geq \epsilon]|R_S[f_{\hat{a}}] = 0]$, i.e. probability of drawing $S = (x_1, \ldots, x_n)$ s.t. $R_S[f_{\hat{a}}] = 0$ but $R[f_{\hat{a}}]$ is large. A demonstration of the error probability is as Fig.1.

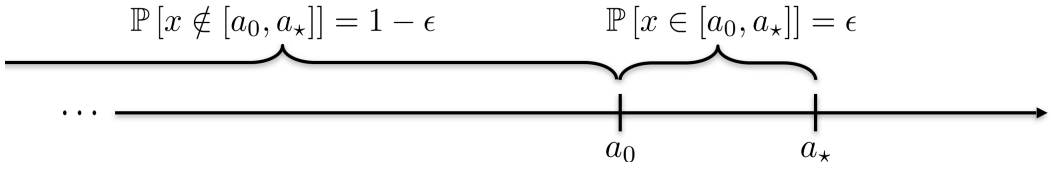

Figure 1: Demonstration of the error probability

Recall that we set $\hat{a} = \max_i x_i$. Therefore, the only way to achieve $R_S[f_{\hat{a}}] = 0$ and $R[f_{\hat{a}}] \geq \epsilon$ is if $x_i \notin [a_0, a_*]$ for all $i = 1, \ldots, n$

$$\begin{aligned}\mathbb{P}[R[f_{\hat{a}}] \geq \epsilon] &= \mathbb{P}[x_i \notin [a_0, a_*] \ \forall i = 1, \ldots, n] \\ &= (\mathbb{P}[x \notin [a_0, a_*]])^n \ \text{since} x_i \overset{i.i.d}{\sim} p(x) \\ &= (1-\epsilon)^n \leq e^{-\epsilon n}\end{aligned}$$

By inverting above inequality, we could get that

$$R[f_{\hat{a}}] = \mathbb{P}[x \in [\hat{a}, a_*]] \leq \frac{\log(1/\delta)}{n} \tag{8}$$

This example tells two things. First, it is the "expressiveness" of $\mathcal{F}$ that matters, not its cardinality. Specifically, the cardinality of threshold function is infinite, but the generalization bound is finite. Second, if the function class $\mathcal{F}$ and data satisfy a realizability assumption, i.e. $\exists f_* \in \mathcal{F}$ such that $R[f_*] = 0$, then we get fast rates of $\mathcal{O}(1/n)$ as opposed to $\mathcal{O}(1/\sqrt{n})$.

8 Covering Numbers

In the last section, we make the conclusion that it is the "expressiveness" of function class $\mathcal{F}$ that matters, not its cardinality. Actaully, there are many ways to meassure expressivity of function classes. Among them, covering numbers [2], VC-dimension [3], and Rademacher/Gaussian complexities [4] are most common. Firstly, we briefly define the above definitions.

Covering Numbers Let $A \subset \mathbb{R}^m$ be a set of vectors. We say that A is $r - covered$ by a set A', with respect to the Euclidean metric, if for all $\mathbf{a} \in A$ there exists $a' \in A'$ with $\|\mathbf{a} - \mathbf{a}'\| \leq r$. We define by $\mathcal{N}(r, A)$ the cardinality of the smallest A' that $r - covers$ A.

Example: Suppose that $A \subset \mathbb{R}^m$, let $c = \max_{a \in A} \|\mathbf{a}\|$, and assume that A lies in a d-dimensional subspace of $\mathbb{R}^m$. Then, $\mathcal{N}(r, A) \leq (2c\sqrt{d}/d)^d$. To see this, let $\mathbf{v}_1, \ldots, \mathbf{v}_d$ be an orthonormal basis of the subspace. Then, any $\mathbf{a} \in A$ can be written as $\mathbf{a} = \sum_{i=1}^d \alpha_i \mathbf{v}_i$, with $\|\alpha\|_\infty \leq \|\alpha\|_2 = \|\mathbf{a}\|_2 \leq c$. Let $\epsilon \in \mathbb{R}$ and consider the set

$$A' = \{\sum_{i=1}^d \alpha_i \mathbf{v}_i : \forall i, \alpha_i' \in \{-c, -c+\epsilon, -c+2\epsilon, \ldots, c\}\}.$$

Given $\mathbf{a} \in A$, s.t. $\mathbf{a} = \sum_{i=1}^d \alpha_i \mathbf{v}_i$ with $\|\alpha\|_\infty \leq c$, there exists $a' \in A'$ such that

$$\|\mathbf{a} - \mathbf{a}'\|^2 = \|\sum_i (\alpha_i' - \alpha_i)\mathbf{v}_i\|^2 \leq \epsilon^2 \sum_i \|\mathbf{v}_i\|^2 \leq \epsilon^2 d.$$

Choose $\epsilon = r/\sqrt{d}$; then $\|\mathbf{a} - \mathbf{a}'\| \leq r$ and therefore A' is an r-cover of A. Hence,

$$\mathcal{N}(r, A) \leq |A'| = (\frac{2c}{\epsilon})^d = (\frac{2c\sqrt{d}}{r})^d$$

VC-dimension The VC-dimension of a hypothesis class $\mathcal{H}$ denoted $VCdim(\mathbf{H})$, is the maximal size of a set $C \subset \mathcal{X}$ that can be shattered by $\mathcal{H}$. if $\mathcal{H}$ can shatter sets of arbitrarily large size we say that $\mathcal{H}$ has infinite VC-dimension.

Rademacher/Gaussian complexity Let μ be a probability distribution on a set $\mathcal{X}$ and suppose that $X_1, \ldots, X_n$ are independently samples selected according to μ. Let $\mathcal{F}$ be a class of functions mapping from $\mathcal{X}$ to $\mathbb{R}$. Define the maximum discrepency of $\mathcal{F}$ as the random variable

$$\hat{D}_n(\mathcal{F}) = \sup_{f \in \mathcal{F}} (\frac{2}{n} \sum_{i=1}^{n/2} f(X_i) - \frac{2}{n} \sum_{i=n/2+1}^{n} f(X_i))$$

Denote the expected maximum discrepency of $\mathcal{F}$ by $D_n(\mathcal{F}) = \mathbf{E}\hat{D}_n(\mathcal{F})$.

Define the random variable

$$\hat{R}_n(\mathcal{F}) = \mathbf{E}[\sup_{f\in\mathcal{F}} |\frac{2}{n}\sum_{i=1}^{n} \sigma_i f(X_i)||X_1, \ldots, X_n],$$

where $\sigma_1, \ldots, \sigma_n$ are independent uniform ± 1-valued random variables. Then the Rademacher complexity of $\mathcal{F}$ is $R_n(\mathcal{F}) = \mathbf{E}\hat{R}_n(\mathcal{F}))$. Similarly, define the random variable

$$\hat{G}_n(\mathcal{F}) = \mathbf{E}[\sup_{f\in\mathbf{F}} |\frac{2}{n}\sum_{i=1}^{n} g_i f(X_i)||X_1, \ldots, X_n],$$

where $g_1, \ldots, g_n$ are independent Gaussian $\mathcal{N}(0,1)$ random variables. The Gaussian complexity of $\mathcal{F}$ is $G_n(\mathcal{F}) = \mathbf{E}(\mathcal{F})$.

In our lecture note, we will talk about **covering number**.

8.1 Covering Number Lemma

Covering Number Lemma: Let $Q : \mathcal{F} \times Z \to \mathbb{R}$ be L-Lipschitz with respect to the first argument:

$$|Q(f,z) - Q(f',z)| \leq L\|f - f'\|, \ \forall z \in Z, f, f'\mathcal{F} \tag{9}$$

Let S be a $\frac{\epsilon}{2L} - net$ for $\mathcal{F}$, and suppose that

$$\mathbb{P}[Q(f,z) - \mathbb{E}_z[Q(f,z)] \geq \epsilon] \leq \delta \ \forall f \in S \tag{10}$$

Then

$$\mathbb{P}[\exists f \in \mathcal{F} : Q(f,z) - \mathbb{E}_z[Q(f,z)] \geq 2\epsilon] \leq \delta\mathcal{N}(\mathcal{F}, \frac{\epsilon}{2L}) \tag{11}$$

Proof:

$$\begin{aligned} Q(f,z) - \mathbb{E}_z[Q(f,z)] &= Q(s,z) - \mathbb{E}_z[Q(s,z)] + Q(f,z) - Q(s,z) - \mathbb{E}_z[Q(f,z) - Q(s,z)] \\ &\leq Q(s,z) - \mathbb{E}_z[Q(s,z)] + L\|f - s\| + L\|f - s\| \\ &\leq Q(s,z) - \mathbb{E}_z[Q(s,z)] + \epsilon \end{aligned}$$

The first inequality follows because if event A implies B, then $P(A) \leq P(B)$. Then we get

$$\begin{aligned} &\mathbb{P}[\exists f \in \mathcal{F} : Q(f,z) - \mathbb{E}_z[Q(f,z)] \geq 2\epsilon] \\ &\leq \mathbb{P}[\exists s \in \mathcal{S} : Q(s,z) - \mathbb{E}_z[Q(s,z)] \geq \epsilon] \\ &\leq \delta\mathcal{N}(\mathcal{F}, \frac{\epsilon}{2L}) \end{aligned}$$

The last inequality follows by the union bound over the $\mathcal{N}(\mathcal{F}, \frac{\epsilon}{2L})$ elements of the $\frac{\epsilon}{2L}$-net.

8.2 Uniform Convergence using Covering Number

Next, we get a generalization bound using covering number lemma. First, we set $Q(f,z) = \ell(f(x), y)$, and assume that the loss function is L-Lipschiz with respect to $f \in \mathcal{F}$, i.e. $|\ell(f_1(x), y) - \ell(f_2(x), y)| \leq L\|f_1 - f_2\|$. Further, we assume that $|\text{loss}(f(x), y)| \leq B$, then using McDiarmid, for fixed $f \in S$, we have

$$\mathbb{P}[R[f] - R_S[f] \geq \epsilon] \leq \exp(-\frac{n\epsilon^2}{2B^2}) \tag{12}$$

Applying the covering number lemma, we see that

$$\mathbb{P}[\exists f \in \mathcal{F} : R[f] - R_S[f] \geq 2\epsilon] \geq \mathcal{N}(\mathcal{F}, \frac{\epsilon}{2L}) \exp(-\frac{n\epsilon^2}{2B^2}) \tag{13}$$

Inverting, we conclude that with probability at least $1 - \delta$ we have that

$$\mathbb{R}[h] \leq R_S[h] + 2B\sqrt{\frac{2\log(\mathcal{N}(\mathcal{F}, \frac{\epsilon}{2L})/\delta)}{n}} \tag{14}$$

8.3 Linear Classifier Example

From the above example, we could make the following conclusions: (i) $\mathcal{N}(\mathcal{F}, \frac{\epsilon}{2L})$ captures "expressivity", and shows that the smoother the loss functions generalize better; (ii) Suppose we want to estimate the risk of the best linear classifier $\mathcal{F} = \{f(x) = w^T x : \|w\| \leq M\}$ (iii) Suppose the loss is L-Lipschitz, then it is 2LM-bounded; (iv)if $x, w \in \mathbb{R}^d$, then $\mathcal{N}(\mathcal{F}, \epsilon) \leq (\frac{4M}{\epsilon})^d$; (v) We conclude, using our lemma, that with probability at least $1 - \delta$, $R[w] \leq R_S[w] + \epsilon \ \forall \|w\| \leq W$ if

$$n \geq \frac{2dL^2M^2}{\epsilon^2} \log(\frac{8ML}{\epsilon}) + \frac{2B^2M^2}{\epsilon^2} \log(1/\delta);$$

(vi)Bound can be improved, but uniform convergence always needs $n \geq \Omega(\frac{d}{\epsilon^2})$.

References

[1] Colin McDiarmid. *On the method of bounded differences*, page 148–188. London Mathematical Society Lecture Note Series. Cambridge University Press, 1989.

[2] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms.* Cambridge University Press, New York, NY, USA, 2014.

[3] Vladimir N. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, Berlin, Heidelberg, 1995.

[4] Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *J. Mach. Learn. Res.*, 3:463–482, March 2003.