CUHK STAT6050: Statistical Learning Theory Spring 2022

Lecture 1: Introduction to Statistical Learning Theory

Lecturer: Ben Dai

"*There is Nothing More Practical Than A Good Theory.*" — Kurt Lewin

1 Overview

In [Von Luxburg and Schölkopf, 2011]: *"Statistical learning theory is regarded as one of the most beautifully developed branches of artificial intelligence. It provides the theoretical basis for many of today's machine learning algorithms. The theory helps to explore what permits to draw valid conclusions from empirical data."*

This course mainly focuses on the subset of statistical learning theory which is highly related to supervised statistical methodologies. Following are some specific purposes:

- (Justification). Theoretical analysis of machine learning methods with a large-scale dataset, the methods can be arbitrary, from parametric models to deep neural networks. For example, asymptotically show that Method A is better than Method B; or we try to find conditions in which Method A is better; or if a method is the best one. (**Asymptotics**; **excess risk bound**; **Consistency**; **Convergence rate**; **Minimax rate**.)
- (Explore new methods). Most machine learning methods are motivated by { **intuition** | **numerical studies** | **theory** }. Statistical learning theory is one of the most important ways to motivate a useful method. For example, new surrogate losses in classification (**Fisher/excess risk consistency**), random forest (**bias-variance trade-off**), local smoothing (**nonparametric statistics**), Debiased Lasso, ...

2 Statistical Modeling: The Two Cultures [Breiman, 2001]

In this section, we will follow Leo Breiman's "Two Cultures" of statistical modeling [Breiman, 2001], to motivate the framework of learning theory on machine learning methods.

2.1 From MLE to ERM

Given the notations in Table 1, we recall the procedure of maximum likelihood estimation (MLE), as indicated in the upper panel of the following figure. The main idea of MLE is:

Inputs: Training data $\mathscr{D}_n$.

Table 1: Notations in supervised learning

Dataset

$\mathscr{D}_n = (\mathbf{X}_i, \mathbf{Y}_i)_{i=1,\cdots,n}$	$\triangleq$	Training set with n samples, where $(\mathbf{X}_i, \mathbf{Y}_i) \overset{d}{=} (\boldsymbol{X}, \boldsymbol{Y})(i = 1, \cdots, n)$ are i.i.d. random samples on a probability space with the probability measure $\mathbb{P}$.
$\mathbf{X}$	$\triangleq$	Features or inputs of a sample. $\mathbf{X} \in \mathscr{X} \subset \mathscr{R}^d$ is a d-length (random) vector.
$\mathbf{Y}$	$\triangleq$	Response or outcome of a sample. $\mathbf{Y} \in \mathscr{Y} \subset \mathscr{R}^K$ is a K-length vector.

Learning paradigm

$f(\cdot)$	$\triangleq$	A decision function. $f : \mathscr{X} \to \mathscr{R}^K; \mathbf{x} \to f(\mathbf{x})$ maps inputs (feature) space to the outcome-space, say the decision function is $f(x)$ given a sample $\mathbf{X} = \mathbf{x}$.
$l(\cdot, \cdot)$	$\triangleq$	A loss function. $l : \mathscr{Y} \times \mathscr{R}^K \to \mathscr{R}; \big(\mathbf{y}, f(\mathbf{x})\big) \to l\big(\mathbf{y}, f(\mathbf{x})\big)$ measure the discrepancy between the true outcome and the decision function.
$R(f)$	$\triangleq$	The risk of the decision function. $R(f) \triangleq \mathbb{E}\Big(l\big(\mathbf{Y}, f(\boldsymbol{X})\big)\Big)$

Step 1. Introduce parameters $\boldsymbol{\theta}$ that index the (conditional) probability distribution $\mathbb{P}_{\mathbf{Y}|\mathbf{X}}$ within a parametric family $\{p_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y}) | \boldsymbol{\theta} \in \Theta\}$.

Step 2. Establish the log-likelihood function based on the training data $\mathscr{D}_n$, and the optimal parameter is obtained by:

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta} \in \Theta} \frac{1}{n} \sum_{i=1}^{m} \log p_{\boldsymbol{\theta}}(\mathbf{x}_i, \mathbf{y}_i)$$

Step 3. Given a problem or evaluation loss, make prediction via the estimated density $p_{\widehat{\boldsymbol{\theta}}}(\mathbf{x}, \mathbf{y})$. (We already obtained the conditional density, we basically know everything about $\mathbf{Y}|\mathbf{X} = \mathbf{x}$). For example, for regression problem, $\mathbf{Y}$ is predicted as $\widehat{\mathbb{E}}(\mathbf{Y}|\mathbf{X} = \mathbf{x})$, where $\widehat{\mathbb{E}}$ is the expectation based on the estimated conditional density $\widehat{p}_{\widehat{\boldsymbol{\theta}}}$.

Example 2.1 (Logistic regression)**.** *For binary classification, $Y \in \{-1, +1\}$ indicates the binary label for the input $\mathbf{X}$, we apply **Steps 1-2** to estimate the conditional probability $p_{\widehat{\boldsymbol{\theta}}}(Y = 1|\mathbf{X} = \mathbf{x})$, and the predicted label is provided as $\widehat{Y} = \operatorname{sgn}\big(p_{\widehat{\boldsymbol{\theta}}}\big(Y = 1|\mathbf{X} = \mathbf{x}\big) - 1/2\big)$.*

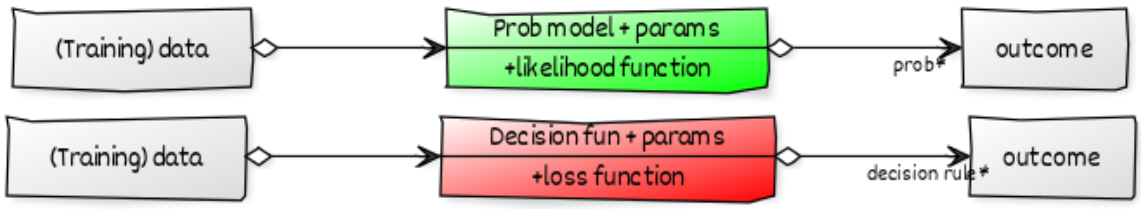

Figure 1: The differences between maximum likelihood estimation (MLE) and empirical risk minimization (ERM).

MLE is one of the most important (probably the single most important) tool in "the data modeling culture". Yet, it has some potential drawbacks as mentioned in [Breiman, 2001]:

- "*Assume that the data are generated by the following model: ...*"
- "*The conclusions are about the model's mechanism, and not about nature's mechanism.*"
- "*If the model is a poor emulation of nature, the conclusions may be wrong.*"
- Readers can find more detailed discussion in Section 6 in [Breiman, 2001].

Remark 2.2 (Over-estimation in MLE). MLE aims to summarize of all the information from data (something like *likelihood principle*). As we can see, MLE estimates parameters of the density function (again once we know the density we know everything). Yet, for a prediction problem, it may be too aggressive, since we just want to know a decision rule instead of the whole density function. In **Example 1.1**, the sign of $p\big(Y=1|\mathbf{X}=\mathbf{x}\big)-1/2$ is completely enough to make prediction, thus there is no need to estimate the exact conditional density. We usually refer this case as "*over-estimation*".

2.2 The problem comes first

[Breiman, 2001] further points out the key idea of "The Algorithmic Modeling Culture" is *the problem comes first*. To be more specific and practical, if we check a Kaggle data competition, the primary goal is to *predict* the outcome of new instances based on a *training dataset* to minimize (maximize) a *pre-given evaluation loss* (metric). Hence, the **Steps** will be "upside down".

Inputs: Training data $\mathscr{D}_n$; loss function $l(Y,\widehat{Y})$ for evaluation.

Step 1. Based on the problem (loss), construct a decision function. For example, for binary classification, the decision rule is given as $\widehat{Y}=\operatorname{sgn}(f_{\boldsymbol{\theta}}(\mathbf{X}))$.

Step 2. Obtain the optimal decision function via Empirical Risk Minimization (ERM), which directly minimizes the given loss function:

$$\widehat{\boldsymbol{\theta}}=\underset{\boldsymbol{\theta}\in\Theta}{\arg\min}\,\frac{1}{n}\sum_{i=1}^{n}l\big(\mathbf{y}_i,f_{\boldsymbol{\theta}}(\mathbf{x}_i)\big)$$

Step 3. Given a new instance $\mathbf{x}$, the predicted outcome is provided by the decision rule based on the estimated decision function $\widehat{f}_{\widehat{\boldsymbol{\theta}}}(\mathbf{x})$. For example, $\widehat{Y}=\operatorname{sgn}(\widehat{f}_{\widehat{\boldsymbol{\theta}}}(\mathbf{x}))$ for binary classification.

Remark 2.3 (Differences between MLE and ERM). (i) MLE estimates the density, yet ERM only estimates the prediction function w.r.t a specific loss. (ii) The decision rule is specified after (before) the estimation for MLE (ERM).

It is understandable why ERM would outperform MLE, since ERM is more specific for a task with a pre-given loss function. Back to Section 1, we aim to quantify the "goodness" of a machine learning method. To this end, we will focus on ERM, and introduce more concepts that intensely interested in statistical learning theory.

3 Framework

The content of this section is:

- Define a **risk** function to measure the performance of a decision function.
- Define the **Bayes rule** and a **excess risk** to measure "efficiency" of a decision function.

A **risk** function (or **generalization error**) is introduced to measure predictive performance. Given a decision function f, its predictive performance is computed as

$$R(f) = \mathbb{E}\Big(l\big(\mathbf{Y}, f(\mathbf{X})\big)\Big).$$

Note that the expectation is taken w.r.t. both $\mathbf{X}$ and $\mathbf{Y}$. Given a testing dataset $\mathscr{T}_m = (\mathbf{X}_j^{\text{te}}, \mathbf{Y}_j^{\text{te}})_{j=1,\cdots,m}$, the risk function is empirically evaluated as an averaged loss:

$$\widehat{R}_m(f) = \frac{1}{m}\sum_{j=1}^{m} l\big(\mathbf{y}_j^{\text{te}}, f(\mathbf{x}_j^{\text{te}})\big).$$

The risk function can be used to check the performance of a decision function, yet we want to further investigate its "efficiency". To this end, we first introduce the best decision function, namely Bayes decision function (rule), then compute the discrepancy to measure "efficiency".

Definition 3.1 (Bayes decision rule)**.** A Bayes (decision) rule is defined as the smallest risk achievable by any measurable decision function, that is,

$$f^* = \arg\min R(f),$$

where the minimum is taken over all possible measurable functions.

We illustrate the risk function and its Bayes rule by following two examples.

Lemma 3.2 (Mis-classification error)**.** *The mis-classification error (MCE) in binary classification ($Y \in \{-1,+1\}$) is defined as:*

$$R(f) = \mathbb{P}\Big(Y \neq \operatorname{sgn}(f(\mathbf{X}))\Big) = \mathbb{E}\Big(\mathbf{1}\big(Y \neq \operatorname{sgn}(f(\mathbf{X}))\big)\Big) = \mathbb{E}\Big(\mathbf{1}\big(Yf(\mathbf{X}) \leq 0\big)\Big),$$

and f^* is a Bayes rule iff

$$\operatorname{sgn}\big(f^*(\mathbf{x})\big) = \operatorname{sgn}\big(\mathbb{P}_{Y|\mathbf{X}}(Y=1|\mathbf{X}=\mathbf{x}) - 1/2\big).$$

Remark 3.3. f^* in binary classification is *non-identifiable*.

Lemma 3.4 (Mean squared error)**.** *The mean squared error (MSE) in (multi-outcome) regression* ($\mathbf{Y} \in \mathscr{R}^K$) *is defined as:*

$$R(f) = \mathbb{E}\Big(\big(\mathbf{Y} - f(\mathbf{X})\big)^2 \Big),$$

and the Bayes rule is defined as:

$$f^*(\mathbf{x}) = \mathbb{E}\big(\mathbf{Y}|\mathbf{X} = \mathbf{x}\big).$$

Once the Bayes rule is obtained, we can define the best risk as $R^* = R(f^*)$, which is the best performance you can achieve. To measure "efficiency", an **excess risk** (regret) is introduced:

$$\mathscr{E}(f) = R(f) - R^*.$$

Note that $\mathscr{E}(f) \geq 0$, since $R(f) \geq R^*$. Now, we want to check the performance and efficiency of our finite-sample estimator via ERM.

Before that, we would like to point out a probabilistic perspective of ERM. Note that our final goal is to find a minimizer of the risk function in a population level

$$\min_f R(f) = \min_f \mathbb{E}\big(l(\mathbf{Y}, f(\mathbf{X}))\big).$$

Two issues are likely to stand out. (i) We have no idea about calculating the expectation, since we don't want to make any assumption on data distribution. (ii) The minimum is taken over all measurable functions, which is infeasible to optimize.

To address (i), the strategy of ERM is to replace the population mean by the empirical average on a training dataset. This is a key of "*learning from data*": good performance in training set yields good performance in testing set or in population. The assumption of this framework is training set and testing test are i.i.d. samples[1]. To address (ii), we introduce a candidate class $\mathscr{F}$, usually a function space index by some parameters, yet it can be a general functional space as in nonparametric methods.

Now, the formulation of ERM is given as:

$$\widehat{f}_n = \arg\min_{f \in \mathscr{F}} \frac{1}{n} \sum_{i=1}^{n} l\big(\mathbf{y}_i, f(\mathbf{x}_i)\big), \tag{1}$$

where $\widehat{f}_n$ is the final estimator we obtained from the training set. Then, we tend to quantify the performance of $\widehat{f}_n$:

$$\mathscr{E}(\widehat{f}_n) = R(\widehat{f}_n) - R^*.$$

Remark 3.5. $\mathscr{E}(\widehat{f}_n)$ is random, and its randomness is caused by $\widehat{f}_n$, which is estimated from random samples in the training set $\mathscr{D}_n$.

To measure the performance based on the random criteria, we introduce three concepts:

[1] One may check "transfer learning" when training set and testing set have different distributions.

Definition 3.6 (Excess risk consistency)**.** $\widehat{f}_n$ is excess risk consistency w.r.t. the risk $R(\cdot)$ if

$$R(\widehat{f}_n) \xrightarrow{\mathbb{P}} R^*, \quad \text{as} \quad n \to \infty.$$

Definition 3.7 (Convergence rate)**.** Suppose that $\delta_n \to 0$, and $\widehat{f}_n$ satisfies that

$$\mathscr{E}(\widehat{f}_n) = R(\widehat{f}_n) - R^* = O_P(\delta_n),$$

then δ_n is the convergence rate of $\mathscr{E}(\widehat{f}_n)$.

Definition 3.8 (Probabilistic bound)**.** For any $\varepsilon > 0$, there exists $N_0(\varepsilon)$, for $n > N_0(\varepsilon)$

$$\mathbb{P}\big(\mathscr{E}(\widehat{f}_n) \geq \delta_n'(\varepsilon)\big) \leq \varepsilon,$$

provided that some $\delta_n'(\varepsilon) \to 0$, as $n \to \infty$.

Lemma 3.9. *Probabilistic bound $\Longrightarrow$ Convergence rate $\Longrightarrow$ Excess risk consistency.*

Lemma 3.10. *Suppose that $\mathscr{E}(\widehat{f}_n) = O_P(\delta_n)$, and $\delta_n = o(\omega_n)$ then $\mathscr{E}(\widehat{f}_n) = o_P(\omega_n)$.*

Example 3.11 (Toy example)**.** ***Data.*** *Suppose $(Y_1, \cdots, Y_n)$ is a sequence of i.i.d. random samples with $\mathbb{E}(Y_i) = \mu = 0$ and $\mathbf{Var}(Y_i) = \sigma = 1$.* ***Risk.*** $R(\theta) = \mathbb{E}l(Y, \theta) = \mathbb{E}\big((Y-\theta)^2\big)$.

Bayes decision function: $\theta^* = \mathbb{E}(Y) = \mu$.
Empirical estimator: $\widehat{\theta} = \bar{Y} = \frac{1}{n}\sum_{i=1}^n Y_i$ *is a function of* $(Y_1, \cdots, Y_n)$
Then, the excess risk is

$$\mathscr{E}(\widehat{\theta}) = R(\widehat{\theta}) - R^* = \mathbb{E}\big((Y-\widehat{\theta})^2\big) - \mathbb{E}\big((Y-\mu)^2\big) = \mathbb{E}(\widehat{\theta}^2) = \widehat{\theta}^2.$$

Note that the expectation is taken w.r.t. Y, which is independent with $(Y_1, \cdots, Y_n)$.

- *Probabilistic bound. For any $\delta > 0$,*

$$\mathbb{P}\big(\mathscr{E}(\widehat{\theta}) \geq \delta^2\big) = \mathbb{P}\big(\widehat{\theta}^2 \geq \delta^2\big) = \mathbb{P}\big(|\widehat{\theta}| \geq \delta\big) \leq \frac{1}{\sqrt{n}\delta},$$

  where the last inequality follows from the Chebyshev's inequality. Alternatively, we can say, for any $\varepsilon > 0$,

$$\mathbb{P}\big(\mathscr{E}(\widehat{\theta}) \geq \frac{1}{\varepsilon^2 n}\big) \leq \varepsilon.$$

- *Convergence rate and excess risk consistency.*

$$\mathscr{E}(\widehat{\theta}) = O_P(1/n).$$

Remark 3.12 (Strong/weak convergence). In MLE, we typically consider the asymptotics or convergence rate of $\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*\|_2$ or $\|\widehat{f}_n - f^*\|_{\mathscr{F}}$, which can be regarded as "strong convergence". For ERM, we consider the convergence of $R(\widehat{f}_n) - R^*$, which can be regarded as "weak convergence".

References

[Breiman, 2001] Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, 16(3):199–231.

[Von Luxburg and Schölkopf, 2011] Von Luxburg, U. and Schölkopf, B. (2011). Statistical learning theory: Models, concepts, and results. In *Handbook of the History of Logic*, volume 10, pages 651–706. Elsevier.