30 August 2021

CS492(F). Comp. Learning Theory — Introduction.

1. Course overview.

(1) As described in the syllabus, the goal of this course is to study standard and, in a sense, old-style mathematical theories that attempt to explain why machine-learning algorithms work and provide principles behind the designs of those algorithms.

(2) To get a sense about what we will do (in this course), let us recall the basics of ML. In an overly-simplified view on ML, an ML algorithm tries to solve the following optimization problem:

$$\operatorname*{argmin}_{h \in \mathcal{H}} R(h), \quad R(h) = \mathbb{E}_{(x,y)\sim D}[L(h(x), y)] \quad \cdots (1)$$

Here $\mathcal{H}$ is a set of functions from $\mathcal{X}$ to $\mathcal{Y}$, D is an unknown probability distribution on input-output pairs, and L is a loss function that measures how well $h(x)$ approximates y.

Unfortunately, we cannot solve the opt. pb. in (1). One reason is that we don't know D and cannot compute $R(h)$. We only have a finite number of (independent) samples $S = ((x_1, y_1), \cdots, (x_m, y_m))$ from D.

As a result, an ML alg. typically optimizes a proxy to the ideal goal $R(h)$ that is, $\hat{R}_S(h)$ built from the samples S.

For instance, the algo. may solve the following problem:

$$\underset{h \in \mathcal{H}}{\operatorname{argmin}} \ \hat{R}_S(h) \quad , \quad \hat{R}_S(h) = \frac{1}{m} \sum_{i=1}^{m} L(h(x_i), y_i) \qquad \cdots (2)$$

(3). The situation just explained gives rise to two natural questions.

① Is it ok to use $\hat{R}_S$, instead of R? Can we say formally that a good solution to the pb (2) with $\hat{R}_S$ is also a good solution to the pb (1)?

② How can we solve the pb. (2) efficiently?

(4). Partial answers to ① are results on sample complexity and generalization bounds, which we will look at in the course. Here is an example of such a result for regression (Thm 11.8):

probability over the samples S.

Thm 11.8. Under some condition, $\exists\, c_1, c_2, c_3$ s.t for all $\delta > 0$,

$$\mathbb{P}\left[\forall h \in \mathcal{H}.\ R(h) \le \hat{R}_S(h) + c_1 \sqrt{\frac{\log c_2 m}{m}} + c_3 \sqrt{\frac{\log \frac{1}{\delta}}{2m}} \right] \ge 1 - \delta.$$

upper bound.

happens with high probability.

Due to the randomness of S, we cannot give an upper bound of R using $\hat{R}_S$. We can only give a high-probability bound. Note that as m gets larger, the upper bound becomes tighter. Intuitively, this happens because as we have more samples, $\hat{R}_S(h)$ becomes a better sample estimate of $R(h)$.

Another thing to note is the use of universal quantification over all $h \in \mathcal{H}$. Thus, the above bound is often called uniform generalisation bound. The uniformity ensures that the bound holds for all optimisers of the proxy $\hat{R}_S(h)$. Also, it is the reason that we have $C_1\sqrt{\frac{\log C_2 m}{m}}$ in the upper bound. This summand is closely related to the complexity of $\mathcal{H}$. By the way, the last summand $C_3\sqrt{\frac{\log \frac{1}{\delta}}{2m}}$ is related to the error caused by estimating $R(h)$ (for a single h) using samples (i.e., $\hat{R}_S(h)$).

(5) For the question ②, we will learn mathematical tools, such as kernel and convex duality, which allow us to design good algorithms for optimising $\hat{R}_S(h)$. For instance, consider two hyp. sets $\mathcal{H}_1$ and $\mathcal{H}_2$:

① $\mathcal{H}_1 = \{ h : \mathbb{R}^I \to \mathbb{R} \mid h(x) = \langle w, \Phi(x) \rangle \text{ for some } w \in \mathbb{R}^n \}$

where $\Phi : \mathbb{R}^I \to \mathbb{R}^n$

② $\mathcal{H}_2 = \{ h : \mathbb{R}^I \to \mathbb{R} \mid h(x) = \langle \alpha, (y_i K(x, x_i))_{i=1\ldots m} \rangle$ for some $\alpha \in \mathbb{R}^m \}$.

where $(x_1, y_1), \ldots, (x_m, y_m)$ are samples in S, and $K : \mathbb{R}^I \times \mathbb{R}^I \to \mathbb{R}$ is a binary function.

Note the prominent role played by samples in the case ②, and contrast it with the absence of those samples in ①. A learnt function h in ② remembers (explicitly) samples used in learning, while a learnt function h in ① doesn't. Despite this difference,

tools that we will learn show deep connection between $\mathcal{H}_1$ and $\mathcal{H}_2$ in the context of learning. In so doing, they will enable us to solve optimization problems from learning in multiple different ways.

(6). Why does someone want to study these complex and old-fashioned mathematical tools, instead of cool fashionable theories on deep learning? My personal view is that these tools are basic, and by studying them, one can gain deeper insights into key ingredients of ML algorithms.

One such example is regularization. That is, the induction of, for instance, $\lambda\|w\|_2^2$ term in the optimization objective

$$\text{argmin} \quad \lambda\|w\|_2^2 + \cdots$$

Tools that we will learn give some partial answers on where these regularization terms come from. In some cases, they are included to improve the accuracy of $\widehat{R}_S(h)$ with respect to $R(h)$. In other cases, they come from constraints. When an optimization algorithm is designed via convex duality (such as Lagrangian and Fenchel duality).

2. Plan

(1) If time permits, we will cover Chap2 - Chap6 and Chap11 - Chap12 of the textbook "Foundations of Machine Learning" (2nd edition).

(2) If time permits, we will have a Q&A session when we finish a chapter. We will discuss questions posed by students and solve some exercise problems together.

(3). See the course webpage for the detailed tentative schedule.

3. Logistics.

(1) Evaluation

2-3 homework assignments (40%).

Final exam (40%).

Critical survey (20%).

(2) Critical Survey

① Pick a paper in COLT18-21, and write an in-depth survey on the topic or problem studied in the paper, including the results of the paper itself.

② Two submissions.

Survey proposal (29 Oct, max 1 page excluding bib., 5%)

Survey article (3 Dec, max 4 pages excluding bib., 15%)

(3) Teaching Staffs.

Lecturer ... Hongseok Yang (hongseok00@gmail.com, office hour .... 6:00pm - 7:00pm, Monday)

TA ... Sangho Lim (lim.sang@kaist.ac.kr).

(4) Course webpage: https://github.com/hongseok-yang/CLT21

4. Honour code.

(1) We adopt a very strict policy for handling the violation of honour code. If a student is found to cheat by copying solutions of homework questions or exam questions from his or her peers or other sources, she or he will get F automatically.

(2) Plagiarism is also a violation of honour code. Copying texts from textbooks, papers, wikipedia and other sources is an instance of plagiarism. You will have to rephrase those texts in your own words, and mention the sources of the texts explicitly by including appropriate citations. Ideally, though, your reports have to be based mostly on your own phrases and sentences, not such borrowed rephrased texts.

5. Final remarks.

(1) I am not an expert on the theories of machine learning. So, I might make mistakes, and my understanding and explanations might be inaccurate sometimes. If you feel that something is strange or incorrect, please say so by email, or to KLMS, or during my lectures.

(2) The textbook is good, but contains various typos or errors. So, when you read the textbook, if you encounter such errors, try to find out what the authors really want to say and to fix those errors.