EECS 545: Machine Learning
Lecture 14. Learning Theory

Honglak Lee and Michał Dereziński

02/23/2022

Outline

- PAC theory
- Generalization bounds for finite hypothesis classes
  - Zero training error case
  - Non-zero training error case

- VC Dimension

Overview

- **Probably approximately correct learning (PAC learning)**
  - framework for mathematical analysis of machine learning
  - proposed in 1984 by Leslie Valiant (Turing Award winner 2010)
- In this framework, the learner receives samples and must select a generalization function (**hypothesis**) from a certain class of possible functions.
- Goal: with **high probability** (the "probably" part), the selected function will have **low generalization error** (the "approximately correct" part).
- PAC theory introduces **computational complexity theory**.
  - E.g., How many samples do you need to guarantee “probably approximately correctness”
  - E.g., With $N(\epsilon, \delta)$ samples, empirical error is within ϵ of the generalization error with high probability ($\geq 1-\delta$).

Generalization bounds for zero training error learning

Finite hypothesis classes

- Consider the case where our hypothesis class is finite.
- Our learning algorithm selects a hypothesis *h* from *H*
  - based on a sample *D* of *n* i.i.d. examples $\{x^{(i)}, y^{(i)}\}$ from $P(x; y)$ denoted as $D \sim P^n$

Generalization error

- We will denote the 0/1 training data error (also called loss) of a hypothesis h as:

$$L_D(h) = \frac{1}{n} \sum_i \mathbf{1}\{h(x^{(i)}) \neq y^{(i)}\}$$

and we'll denote the generalization (true) error as

$$L_P(h) = \mathbf{E}_{(\mathbf{x},y)\sim P}[\mathbf{1}(h(\mathbf{x}) \neq y)]$$

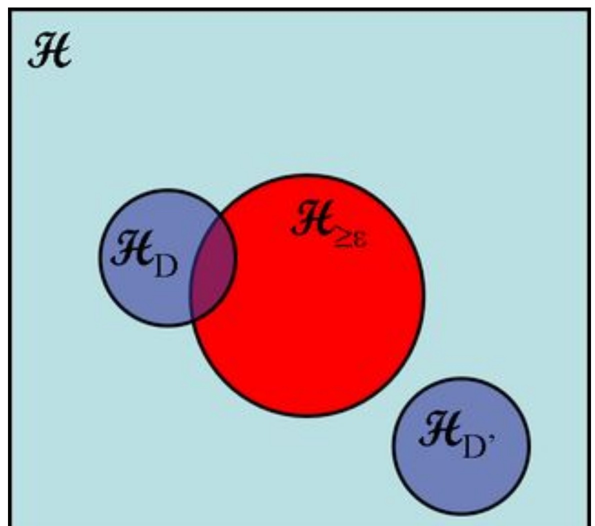
- The true error of the hypothesis we select based on low training error is most likely larger than the training error.

A bound for zero training error learning

- Suppose our algorithm finds a hypothesis consistent with the data, $L_D(h) = 0$.
- We are interested in the probability (over all possible training samples *D*) that *h* could have $L_P(h) \geq \epsilon$.
- Let's denote the set of hypothesis consistent with *D* as $H_D \subset H$ and the set of hypotheses that have true error greater than ϵ as $H_{\geq \epsilon} \subset H$.
- Note that H_D is a random subset of *H*, while $H_{\geq \epsilon}$ is fixed.

A bound for zero training error learning

- The problem for the learner occurs when the two overlap, since in the worst case, a consistent hypothesis can end up being from $H_{\geq\epsilon}$.
- Can we bound the difference in terms of the complexity of our hypothesis class *H*?



A bound for zero training error learning

- We're going to bound the probability that the overlap is non-empty.
- Consider a given hypothesis *h* that has true error greater than ϵ, i.e.: $h \in \mathcal{H}_{\geq \varepsilon}$
- The probability that it has made zero errors on *D*, $h \in \mathcal{H}_D$, is exponentially small in *n*:

$$Pr_{D \sim P^n}(L_D(h) = 0) \leq (1 - \varepsilon)^n$$

A bound for zero training error learning

- Now suppose that there are *K* hypotheses with true error greater than ϵ, i.e. $\mathcal{H}_{\geq\varepsilon} = \{h_1, \ldots, h_K\}$
- What is the probability that any of them is consistent with the data *D*?

$$Pr_{D\sim P^n}(\exists h \in \mathcal{H}_{\geq\varepsilon} : h \in \mathcal{H}_D)$$

$$= Pr_{D\sim P^n}(h_1 \in \mathcal{H}_D \vee \ldots \vee h_K \in \mathcal{H}_D)$$

  – where $\vee$ is the logical OR symbol.

A bound for zero training error learning

- To bound the probability of a union of events, we use the the union bound:

$$\textbf{Union Bound:} \quad P(A \cup B) \leq P(A) + P(B)$$

- Hence

$$Pr_{D \sim P^n}(\exists h \in \mathcal{H}_{\geq \varepsilon} : h \in \mathcal{H}_D) \leq K(1-\varepsilon)^n$$

- To make this bound useful, we will simplify it at the expense of further looseness.

A bound for zero training error learning

- Since we don't know *K* in general, we will upper-bound it by |*H*|.
- Since $(1-\varepsilon) \leq e^{-\varepsilon}$ for $\varepsilon \in [0,1]$, we will write:

$$
\begin{aligned}
Pr_{D\sim P^n}(\exists h \in \mathcal{H}_{\geq\varepsilon} : h \in \mathcal{H}_D) &\leq K(1-\varepsilon)^n \\
&\leq |\mathcal{H}|e^{-n\varepsilon}
\end{aligned}
$$

- Hence the probability that our algorithm will select a hypothesis with true error greater than ϵ given that it selected a hypothesis with zero training error is bounded by $|\mathcal{H}|e^{-n\varepsilon}$, which decreases exponentially with *n*.

What's the bound good for? (1)

- There are two ways to use the bound. One is to set the probability of failure δ, and the number of examples *n* and ask about the largest ϵ.

$$\delta = |\mathcal{H}|e^{-n\varepsilon} \rightarrow \varepsilon = \frac{\log|\mathcal{H}| + \log\frac{1}{\delta}}{n}$$

- So with prob. *1-δ* over the choice of training sample of size *n*, for any hypothesis *h* with zero training error, $L_P(h) \leq \frac{\log|\mathcal{H}| + \log\frac{1}{\delta}}{n}$

What's the bound good for? (2)

- The other way to use it is to fix δ and ϵ and ask how many examples *n* are needed to guarantee them (sample complexity).
- That is:

$$\delta = |\mathcal{H}|e^{-n\varepsilon} \rightarrow n = \frac{1}{\varepsilon}\left(\log|\mathcal{H}| + \log\frac{1}{\delta}\right)$$

PAC (Probably Approximately Correct) framework

- The PAC (Probably Approximately Correct) framework asks these questions about different hypothesis classes
- A class is **PAC-learnable** if the number of examples needed to learn a probably (with prob. δ), approximately (true error of at most ϵ) correct hypothesis is polynomial in parameters of the class
  - (e.g. depth of the tree, dimension of hyperplane, etc.)
  - as well as in ϵ and $1/\delta$ (log $1/\delta$)
  - for any distribution
- If the time complexity is also polynomial, the class is **efficiently PAC-learnable**

Generalizing for non-zero training error

- This is the simplest of generalization bounds but their form remains the same:
  - the difference in the true error and training error is bounded in terms of **complexity of the hypothesis class**.
- The limitation of the result so far:
  - it considers finite classes
  - even for finite classes, we have only considered hypotheses with zero training error (which may not exist).

Generalization bounds for non-zero training error learning

A bound for non-zero training error learning

- We will need a form of **Chernoff bound** for biased coins
- Suppose we have *n* i.i.d. examples $z^{(i)} \in \{0,1\}$
  - where $P(z^{(i)} = 1) = p$
- Let $\bar{z} = \frac{1}{n}\sum_i z^{(i)}$ be the proportion of *1*s in the sample.
- Then

$$Pr(p - \bar{z} \geq \varepsilon) \leq e^{-2n\varepsilon^2}$$

A bound for non-zero training error learning

- Given a hypothesis *h*, the probability that the difference between its true error and training error is greater than ϵ is bounded:

$$Pr_{D\sim P^n}(L_P(h) - L_D(h) \geq \varepsilon) \leq e^{-2n\varepsilon^2}$$

A bound for non-zero training error learning

$$Pr_{D\sim P^n}(L_P(h) - L_D(h) \geq \varepsilon) \leq e^{-2n\varepsilon^2}$$

- Now we need to bound the probability of observing the difference for all hypotheses, so that we can pick the one with lowest error safely. Again, we use the **union bound**:

$$Pr_{D\sim P^n}(\exists h \in \mathcal{H} : L_P(h) - L_D(h) \geq \varepsilon) \leq |\mathcal{H}|e^{-2n\varepsilon^2}$$

A bound for non-zero training error learning

- Setting $\delta = |\mathcal{H}| e^{-2n\varepsilon^2}$ and solving, we get: with prob. *1-δ* over the choice of training sample of size *n*, for all hypotheses in *H*:

$$L_P(h) - L_D(h) \leq \sqrt{\frac{\log|\mathcal{H}| + \log\frac{1}{\delta}}{2n}}$$

- In case of non-zero training error, the generalization bound decreases with the **square root of *1/n***, as opposed to with ***1/n***, which is much **slower**.

Complexity of hypothesis classes

- Consider the sizes of some simple hypothesis classes. The class *H* of all Boolean functions on *m* binary attributes $|\mathcal{H}| = 2^{2^m}$
  - Since we can choose any output for any of the 2^{2^m} possible inputs.
  - Clearly, this class is too rich, since $\log_2 |\mathcal{H}| = 2^m$.
- These bounds can be essentially tight,
  - i.e. there are pathological examples where we need that many samples.

Summary so far

- To learn from a small number of examples, we need a **small hypothesis class**.
- We have only considered finite hypothesis class.
- Finer-grained notions of complexity: VC-dimension.

Generalization bounds for infinite classes: VC dimension

Generalization bounds for infinite classes: VC dimension

- We will now consider the case where our hypothesis class is **infinite**, for example, hyperplanes of dimension *m*.

- As before, our learning algorithm selects a hypothesis *h* from *H*
  - based on a sample *D* of *n* i.i.d. examples $\{x^{(i)}, y^{(i)}\}$ from *P(x;y)*, denoted as $D \sim P^n$.

Bounding the complexity of hypothesis class

- We now consider the case of infinite hypothesis classes, for example, hyperplanes.
- To bound the error of a classifier from a class H using n examples, what matters is <u>not</u> the pure size of H, but its **richness**:
  - **The size of the largest dataset such that the model can perfectly classify every possible labeling of that dataset**

VC Dimension

- **Vapnik-Chervonenkis dimension** is one of the most fundamental measures of richness (or power or complexity or variance) of a hypothesis class studied in machine learning.

- VC theory provides such a general measure of complexity, and gives associated bounds on the optimism.

Main result

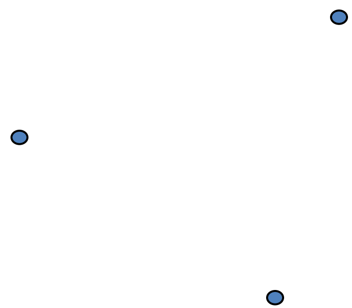
- A fundamental result by Vapnik and Chervonenkis is that the log *|H|* in the bounds can be replaced by a function *VC(H)*.
- **Theorem**: With probability *1-δ* over the choice of training sample of size *n*, for all hypotheses *h* in *H*:

$$L_P(h) - L_D(h) \le \sqrt{\frac{VC(\mathcal{H})(\log \frac{2n}{VC(\mathcal{H})} + 1) + \log \frac{4}{\delta}}{n}}$$

Shattering:

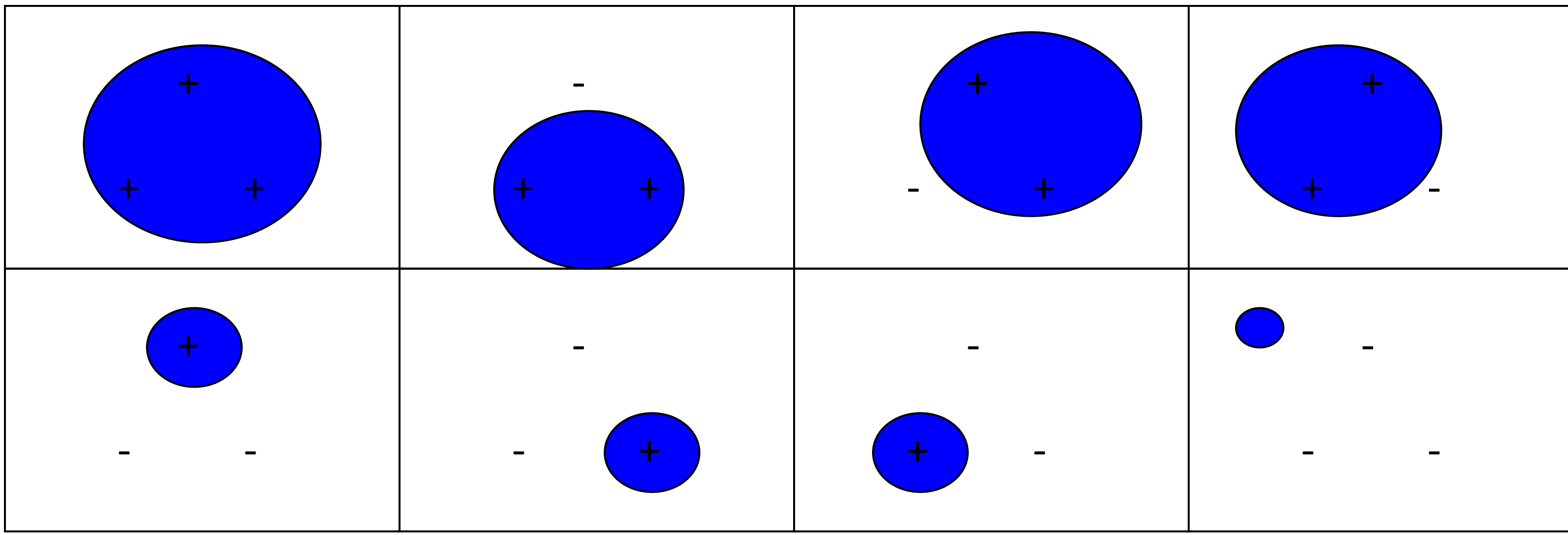
- A set of instances S is <u>shattered</u> by a hypothesis class H if for every <u>dichotomy</u> (every possible labeling) of S there is a consistent hypothesis in H
  - Intuition: The hypothesis class H is rich enough so that it can handle any possible labeling of dataset S.

Example: Shattering

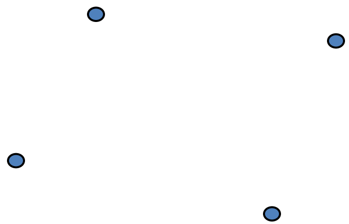


Is this set of points shattered by the hypothesis class *H* of all circles?

Example: Shattering



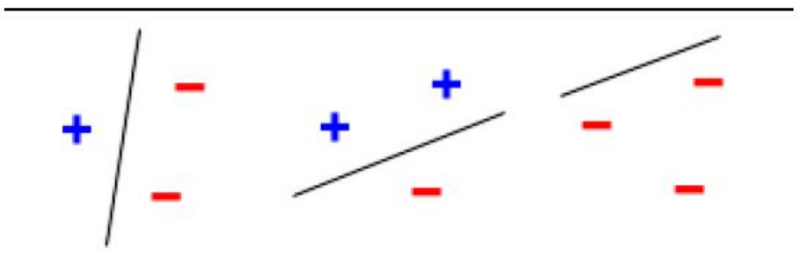
Is this set of points shattered by circles?



Shattering hyperplanes

- Given a set of three points in 2 dimensions, can the set of hyperplanes shatter it?

$$\mathcal{H} = \{h(\mathbf{x}) = sign(w_1 x_1 + w_2 x_2 + b)\}$$



VC dimension

- **Definition:** The VC-dimension of a class H over the input space X is the size of the largest finite set shattered by H.
- To show the VC-dimension of a class is v, we need to show:
  – There exists a set of size v shattered by H (usually easy)
  – There does not exists a set of size v+1 shattered by H (usually harder).

VC dimension

- What is the VC dimension of the previous examples?

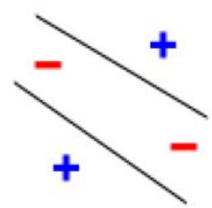
$$\mathcal{H} = \{h(\mathbf{x}) = sign(w_1x_1 + w_2x_2 + b)\}$$

VC dimension

- What is the VC dimension of the previous examples?

$$\mathcal{H} = \{h(\mathbf{x}) = sign(w_1x_1 + w_2x_2 + b)\}$$

- There exists a set of three points that can be shattered by H.
- There does not exist a set S of size 4 shattered by H!



VC dimension for hyperplanes

- In general, we can show the VC dimension of hyperplanes in m dimensions is m+1.

- However, in general, the number of parameters of a classifier is not necessarily it’s VC dimension.

Summary

- Given the VC dimension of the hypothesis class *VC(H)*, with probability *1-δ* over the choice of training sample of size *n*, for all hypotheses *h* in *H*:

$$L_P(h) - L_D(h) \leq \sqrt{\frac{VC(\mathcal{H})(\log \frac{2n}{VC(\mathcal{H})} + 1) + \log \frac{4}{\delta}}{n}}$$

- i.e., this gives a generalization error bound, as well as sample complexity.

Conclusion

- PAC theory provides a bound on generalization error and sample complexity

- The bound critically depends on the complexity of the hypothesis class.

- VC dimension can provide a measure for the complexity of the hypothesis class.

Quiz

https://forms.gle/9NvcsUsLcQ3CZeDQA