Lecture 3: The Bias vs Complexity Trade-Off

CSE 427: Machine Learning

Md Zahidul Hasan

Lecturer, Computer Science
BRAC University

Spring 2023

There's no universal learner

Up until now, we have tried learning with inductive bias. We have chosen a hypothesis set based on some prior knowledge about the concept we are trying to learn. Hence, we have a bias towards the chosen hypothesis set that is captured by the **approximation error**. If we cast a narrow net, our hypothesis could be very biased. If we cast e wide net, our hypothesis could have negligible bias but huge complexity and thus succumb to overfitting. Can we learn a concept without any prior assumption (without a hypothesis set)? Is there a universal learner that can learn any concept with zero prior knowledge? The answer is negative and we can see why. In most applications, we only have access to a fraction of the data. Without any inductive bias, our learned could return a hypothesis that incurs a heavy loss in the long run. This is why it is said that there is no such thing as a universal learner or no free lunch.

The No Free Lunch Theorem

No-Free-Lunch

Let $\mathcal{X}$ be a domain and we are trying to learn binary classification. Let $\mathcal{A}$ be an algorithm that tries to learn from a sample S of size $m \leq \frac{|\mathcal{X}|}{2}$. Then there exists a distribution $\mathcal{D}$ on $\mathcal{X} \times \{0,1\}$ so that:

1. There exists a hypothesis $h : \mathcal{X} \to \{0,1\}$ with $R_D(h) = 0$.
2. Yet, there is a good chance (at least $\frac{1}{7}$) that our learning algorithm returns a hypothesis $A(S)$ that makes frequent mistakes (error rate greater than $\frac{1}{8}$). In short,

$$Pr[R_D(A(S)) > \tfrac{1}{8}] \geq \tfrac{1}{7}$$

Proof: Since, $m \leq \frac{|\mathcal{X}|}{2}$, our algorithm doesn't have access to the majority of the data in the domain. And since, it's a binary classification problem, the algorithm could very well be misinformed by the data.

Let, C be a subset of $\mathcal{X}$ with size $2m \leq |X|$. How many binary classification functions can we define from C to $\{0,1\}$? $T = 2^{2m}$.

The No Free Lunch Theorem contd.

For each of these functions $f_1, f_2, \cdots, f_T$, we can define a domain so that $R_{D_i}(f_i) = 0$. But, we will show that our learning algorithm will return a hypothesis that makes a lot of mistakes on average if we use a sample of size m generated from this distribution. The distributions are defined as follows:

$$\mathcal{D}_i(\{x, y\}) = \begin{cases} \frac{1}{|C|}, & \text{if } f_i(x) = y \\ 0, & \text{otherwise} \end{cases}$$

In the first part of the proof we will show that there exists a distribution $\mathcal{D}_i$ so that if we derive a sample according to that distribution, our algorithm will err in 25% cases on average. In other words,

$$\max_{1 \leq i \leq T} \mathbb{E}_{S \sim \mathcal{D}_i^m}[R_{D_i}(A(S))] \geq \frac{1}{4}.$$

The No Free Lunch Theorem contd.

How many different m size samples can we derive from C? That's $k = (2m)^m$. So,

$\max_{1 \leq i \leq T} \mathbb{E}_{S \sim \mathcal{D}_i^m}[R_{D_i}(A(S))]$

$= \max_{1 \leq i \leq T} \frac{1}{k} \sum_{j=1}^{k} R_{D_i}(A(S_j^i))$

And since the maximum is greater than the average, we get,

$\geq \frac{1}{T} \sum_{i=1}^{T} \frac{1}{k} \sum_{j=1}^{k} R_{D_i}(A(S_j^i))$

$= \frac{1}{k} \sum_{j=1}^{k} \frac{1}{T} \sum_{i=1}^{T} R_{D_i}(A(S_j^i))$

$= \frac{1}{k} \sum_{j=1}^{k} \frac{1}{T} \sum_{i=1}^{T} \frac{1}{2m} \sum_{x \in C} \mathbb{I}\{A(S_j^i)(x) \neq f_i(x)\}$

Let's say that, the instances of C that are used in creating sample S_j^i are contained in the set A, and the rest are contained in B. So,

$\geq \frac{1}{k} \sum_{j=1}^{k} \frac{1}{T} \sum_{i=1}^{T} \frac{1}{2m} \sum_{x \in B} \mathbb{I}\{A(S_j^i)(x) \neq f_i(x)\}$

That's because the number of mismatches in B is smaller than the number of mismatches in C. That's obvious because B is a smaller set.

$\geq \frac{1}{k} \sum_{j=1}^{k} \frac{1}{T} \sum_{i=1}^{T} \frac{1}{2|B|} \sum_{x \in B} \mathbb{I}\{A(S_j^i)(x) \neq f_i(x)\}$

The No Free Lunch Theorem contd.

And notice that, A will have at most m unique instances from C. So B will have at least m instances. Therefore, $|B| \geq m$.
$= \frac{1}{k} \sum_{j=1}^{k} \frac{1}{2|B|} \sum_{x \in B} \frac{1}{T} \sum_{i=1}^{T} \mathbb{I}\{A(S_j^i)(x) \neq f_i(x)\}$
But of the T functions, exactly $\frac{T}{2}$ of them will disagree with $A(S_j^i)$ on x.
$= \frac{1}{k} \sum_{j=1}^{k} \frac{1}{2|B|} \sum_{x \in B} \frac{1}{T} \frac{T}{2}$
$= \frac{1}{4}$

Lemma

Let X be a random variable that takes values in $[0, 1]$. Assume that, $\mathbb{E}[X] = \mu$. Then for any $a \in (0, 1)$, $\mathbb{E}[X > a] \geq \frac{\mu - a}{1 - a}$.
Proof: DIY using Markov's Inequality.

With the lemma given above, we can easily show that,
$Pr[R_{\mathcal{D}(A(S))} > \frac{1}{8}] \geq \frac{\mathbb{E}[R_{\mathcal{D}}(A(S))] - \frac{1}{8}}{1 - \frac{1}{8}} \geq \frac{\frac{1}{4} - \frac{1}{8}}{1 - \frac{1}{8}} = \frac{1}{7}$.

The Bias-Complexity Trade-off

In the no-free-lunch theorem, we have seen that our ERM algorithm considered a hypothesis class H which contains all $T = (2m)^m$ functions possible from $\mathcal{X}$ to $\{0, 1\}$. So, there is no bias in our choice of H. We basically chose the richest hypothesis class and that lead to poor performance by our ERM hypothesis. Even though it minimized the **approximation error** (0 to be more specific), the **estimation error** was high. So, we don't want that. But we also don't want to drift away from finding the best hypothesis and hence increase the bias (approximation error). So, this dilemma is known as the bias vs complexity trade-off in machine learning. This is what we are going to learn in this course. How to select the best and simplest model without being too biased.