Lecture 3a: Empirical Risk Minimization

Introduction to Learning and Analysis of Big Data

A more general approach

- We saw the learning algorithms **Memorize** and k-**Nearest Neighbor**.
- We will now discuss a more **general approach** to the design of learning algorithms.
- We use the same assumptions:
  - Examples $\mathcal{X}$
  - labels $\mathcal{Y}$
  - Distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$
  - A learning algorithm gets $S \sim \mathcal{D}^m$ and outputs $\hat{h}_S : \mathcal{X} \to \mathcal{Y}$.
  - $\mathcal{D}$ is unknown to the learning algorithm.

Choosing a prediction rule

- If the algorithm knew $\mathcal{D}$, it could find the optimal prediction rule.
  - The Bayes-optimal predictor.
- Since S is a random sample from $\mathcal{D}$, it should be "similar" to $\mathcal{D}$.
- **Idea:** find a prediction rule that works well on S.
- The error of prediction rule h on S of size m:

$$\mathrm{err}(h, S) := \frac{1}{m} \sum_{i=1}^{m} \mathbb{I}[h(x_i) \neq y_i].$$

  (also called the **empirical risk**)
- **Empirical Risk Minimization** (ERM):
  Choose a prediction rule that minimizes $\mathrm{err}(h, S)$.
- Both **Memorize** and **Nearest Neighbor** are ERM algorithms.
  - But not k-**Nearest-Neighbors**.

Overfitting

- Problem: Empirical risk minimization can fail miserably.
- Example: The **Memorize** algorithm.
  - If the training sample is of size m,
  - and there are N examples (customers) distributed uniformly, $N \gg m$,
  - and there are two labels (drinks),
  - then $\mathrm{err}(\hat{h}_S, S) = 0$, but $\mathrm{err}(\hat{h}_S, \mathcal{D})$ will be very large.
- **Overfitting**: When the error on the training sample is low, but the error on the distribution is large.
- Can another ERM algorithm avoid this issue?

The **No Free Lunch** theorem

- Recall: $\mathcal{X}$ - examples, $\mathcal{Y} = \{0, 1\}$ - binary labels.

Theorem

For **any** *learning algorithm, if* $m \leq |\mathcal{X}|/2$, *there exists a distribution* $\mathcal{D}$ *over* $\mathcal{X} \times \{0, 1\}$ *such that*

- *There exists a prediction rule* $f : \mathcal{X} \to \{0, 1\}$ *with* $\mathrm{err}(f, \mathcal{D}) = 0$, *but*
- *With a probability of at least* $1/7$ *over random samples* $S \sim \mathcal{D}^m$,

$$\mathrm{err}(\hat{h}_S, \mathcal{D}) \geq 1/8.$$

- Proof idea:
  - Assume some algorithm $\mathcal{A}$;
  - choose a uniform distribution over $2m$ examples;
  - Set the true labels such that $\mathcal{A}$ would guess the wrong label on a large proportion of the examples it didn't observe.

Introducting **inductive bias**

- By the **No Free Lunch** theorem, no learning algorithm gets a low error on **all** distributions, unless it observes almost all possible examples.
- A common solution: **assume something** about the learning problem.
- Examples:
  - The coffee shop: The waiter got a hint that all customers with the same hairstyle like the same drink.
  - Identifying documents about economics: Assume that there is a small number of words that determine whether a document is about economics or not.
  - Identifying people in photos: Assume that photos of the same person are similar in a specific feature representation.
- **Inductive bias**: Restricting/directing the learning algorithms using external knowledge/assumptions about the learning problem.

Example: Learning dosage safety

- Learning problem: which medicine dosages are safe?
- $\mathcal{X} = [0, 100]$ (dosage), $\mathcal{Y} = \{0, 1\}$ (causes side effects?)
- A possible training sample:

- ERM without inductive bias might return the following rule:

Here $\mathrm{err}(\hat{h}_S, S) = 0$. What is $\mathrm{err}(\hat{h}_S, \mathcal{D})$?

- Inductive bias: Limit the ERM algorithm to return only functions that describe **thresholds** on the line:

$$\forall x \in \mathcal{X}, f_a(x) := \mathbb{I}[x \geq a].$$

- Now the algorithm will return something like this:

Again, $\mathrm{err}(\hat{h}_S, S) = 0$. What is $\mathrm{err}(\hat{h}_S, \mathcal{D})$ this time?

Inductive bias

- Recall:
  - Empirical Risk Minimization (ERM): Choose a prediction rule that minimizes $\mathrm{err}(h, S)$.
  - Inductive bias: Choose the prediction rule from a restricted set of functions called a **hypothesis class**: $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$.

ERM with a hypothesis class $\mathcal{H}$

Given a training sample $S \sim \mathcal{D}^m$, output $\hat{h}_S$ such that

$$\hat{h}_S \in \operatorname*{argmin}_{h \in \mathcal{H}} \mathrm{err}(h, S).$$

- We will show (later in the course):
  Limiting the ERM to a "simple" $\mathcal{H}$ can prevent overfitting.
- "Simple" for a finite class = small.
- But there are also "simple" infinite classes.

The Bias-Complexity tradeoff

- Sources of prediction error in an ERM algorithm (agnostic setting):
  - Perhaps the rules in $\mathcal{H}$ are not very good for $\mathcal{D}$.

$$\textbf{Approximation error}: \quad \mathrm{err}_{\mathrm{app}} := \inf_{h \in \mathcal{H}} \mathrm{err}(h, \mathcal{D})$$

  - Perhaps the error of the rule the ERM alg. selected is far from the best.

$$\textbf{Estimation error}: \quad \mathrm{err}_{\mathrm{est}} := \mathrm{err}(\hat{h}_S, \mathcal{D}) - \inf_{h \in \mathcal{H}} \mathrm{err}(h, \mathcal{D}).$$

- Total error: $\mathrm{err}(\hat{h}_S, \mathcal{D}) = \mathrm{err}_{\mathrm{app}} + \mathrm{err}_{\mathrm{est}}$.
- Fix sample size, make $\mathcal{H}$ richer (larger):
  - Approximation error gets smaller (lower **bias**),
  - Estimation error gets larger (higher **statistical complexity**).
- There is a trade-off between the two kinds of error.

The Bias-Complexity tradeoff

- Recall **Overfitting**: When estimation error is large. Can happen if $\mathcal{H}$ is too rich (large).
  - Symptoms: training error (error on S) is low, but true error is high.
- **Underfitting**: When approximation error is large.
  - Symptoms: training error is high.
- Best of both worlds: if we know of a small $\mathcal{H}$ which is suitable for our problem.
- E.g. when looking for safe medicine dosages, choose $\mathcal{H}$ to be the set of threshold functions: $x \mapsto \mathbb{I}[x \geq a]$.

- Selecting $\mathcal{H}$ can represent "world-knowledge" that helps learning.