ECE595 / STAT598: Machine Learning I
Lecture 22 Is Learning Feasible?

Spring 2020
Stanley Chan

School of Electrical and Computer Engineering
Purdue University
Welcome to Part 3 of ECE595 / STAT598!

Here is what we have learned:

- Part 1: The machine learning pipeline
- Part 2: Classification methods

What are we going to do in Part 3?

- Now that we have a method, then what?
- Will it do well? How well?
- Will it fail? When?
- Complex model = better?
- More sample = better?
- Can every problem be solved by learning?
- When do you overfit?
- How to avoid overfit?
Outline

Today’s Lecture:

- What constitutes a learning problem?
  - Training and testing samples
  - Target and Hypothesis function
  - Learning Model
- Is learning feasible?
  - An example
  - The power of probability
- Training versus Testing
  - In-sample error
  - Out-sample error
  - Probability bound

Reference:

- Learning from Data, chapter 1.3
Dataset

Let us first talk about a dataset:

- **Input vectors:** $x_1, \ldots, x_N$
- **Labels:** $y_1, \ldots, y_N$
- **Training set:** $\mathcal{D}$
- **Target function $f$:** Maps $x_n$ to $y_n$
- **Target function is always unknown to you**

$$y_n = f(x_n)$$

Label \hspace{2cm} Input
Training and Testing Set

Let us first talk about a dataset:

- **In-sample**: Samples that are inside the training set
- **Out-sample**: Samples that are outside the training set
Hypothesis Function

- Hypothesis set: $\mathcal{H} = \{h_1, \ldots, h_M\}$: Possible decision boundaries
- Algorithm: Picks $h_m$ from $\mathcal{H}$
- Final hypothesis: $g$: The one you found
Learning Model

Unknown target function
\[ f : \mathcal{X} \rightarrow \mathcal{Y} \]

\[ \mathcal{D} = \{(x_1, y_1), \ldots, (x_N, y_N)\} \]

Hypothesis Set
\[ \mathcal{H} = \{h_1, \ldots, h_M\} \]

Training Data

Learning Algorithm
\[ g(x) \approx f(x) \]

Final Hypothesis

unknown input distribution
\[ p(x) \]

\[ x_1, \ldots, x_N \rightarrow x \]
Outline

Today’s Lecture:

- What constitutes a learning problem?
  - Training and testing samples
  - Target and Hypothesis function
  - Learning Model

- Is learning feasible?
  - An example
  - The power of probability

- Training versus Testing
  - In-sample error
  - Out-sample error
  - Probability bound
Is Learning Feasible?

**In-sample** and **Out-sample**:
- In-sample: Training Data
- Out-sample: Testing Data

When can we claim "learning is feasible"?
Suppose we have a training set $\mathcal{D}$, can we learn the target function $f$?
- "Learn" means: I use the data you give me to come up with an $f$
- "Successful" means: All in-samples are correctly predicted
- And all out-samples are also correctly predicted
- If YES, then we are in business.
- Learning is feasible!
- If NO, then we can go home and sleep.
- There is just no way to learn $f$ from $\mathcal{D}$. 
Example

- Let $\mathcal{X} = \{0, 1\}^3$
- Each $x \in \mathcal{X}$ is a binary vector
- E.g., $x = [0, 0, 1]^T$ or $x = [1, 0, 1]^T$
- How many possible vectors are there? $2^3 = 8$
- Call them $x_1, \ldots, x_8$
- There is a target function $f$
- $f$ maps every $x$ to a $y$
- $y \in \{+1, -1\}$
- E.g., $f([0, 0, 1]) = +1$, $f([0, 1, 1]) = -1$, etc.
- How many possible $f$’s?
- You can think of $f$ as a 8-bit vector
- E.g., $f = [+1, -1, -1, -1, +1, +1, +1, -1]$.
- So there are $2^8 = 256$ possible $f$’s.
Example

- We have 8 input vectors: $\mathcal{X} = \{x_1, \ldots, x_8\}$
- We have 256 hypotheses: $\mathcal{H} = \{h_1, \ldots, h_{256}\}$
- Is learning feasible?
- Give me a subset $\mathcal{D} \subset \mathcal{X}$, can I find a hypothesis $g \in \mathcal{H}$ such that $g = f$?
- Suppose here is what you are given: $\bigcirc = -1$, $\bullet = +1$. You know 6 out of 8. These are the training data.

<table>
<thead>
<tr>
<th>$x_n$</th>
<th>$y_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>$\bigcirc$</td>
</tr>
<tr>
<td>0 0 1</td>
<td>$\bullet$</td>
</tr>
<tr>
<td>0 1 0</td>
<td>$\bullet$</td>
</tr>
<tr>
<td>0 1 1</td>
<td>$\bigcirc$</td>
</tr>
<tr>
<td>1 0 0</td>
<td>$\bullet$</td>
</tr>
<tr>
<td>1 0 1</td>
<td>$\bigcirc$</td>
</tr>
<tr>
<td>1 1 0</td>
<td>?</td>
</tr>
<tr>
<td>1 1 1</td>
<td>?</td>
</tr>
</tbody>
</table>
### Possibility 1

<table>
<thead>
<tr>
<th>$x_n$</th>
<th>$y_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>○</td>
</tr>
<tr>
<td>0 0 1</td>
<td>●</td>
</tr>
<tr>
<td>0 1 0</td>
<td>●</td>
</tr>
<tr>
<td>0 1 1</td>
<td>○</td>
</tr>
<tr>
<td>1 0 0</td>
<td>●</td>
</tr>
<tr>
<td>1 0 1</td>
<td>○</td>
</tr>
<tr>
<td>1 1 0</td>
<td>○</td>
</tr>
<tr>
<td>1 1 1</td>
<td>○</td>
</tr>
</tbody>
</table>

- One 1's will give me ●; Others give me ○
- So the last two entries should be ○
### Possibility 2

<table>
<thead>
<tr>
<th>$x_n$</th>
<th>$y_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>●</td>
</tr>
<tr>
<td>0 0 1</td>
<td>●</td>
</tr>
<tr>
<td>0 1 0</td>
<td>●</td>
</tr>
<tr>
<td>0 1 1</td>
<td>○</td>
</tr>
<tr>
<td>1 0 0</td>
<td>●</td>
</tr>
<tr>
<td>1 0 1</td>
<td>○</td>
</tr>
<tr>
<td>1 1 0</td>
<td>○</td>
</tr>
<tr>
<td>1 1 1</td>
<td>●</td>
</tr>
</tbody>
</table>

- Odd numbers of 1’s give me ●
- Even numbers of 1’s give me ○
- So [1 1 0] should be ○
- So [1 1 1] should be ●
### All the Possibilities

<table>
<thead>
<tr>
<th>$x_n$</th>
<th>$y_n$</th>
<th>$g$</th>
<th>$f_1$</th>
<th>$f_2$</th>
<th>$f_3$</th>
<th>$f_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 0</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>0 0 1</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>0 1 0</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>0 1 1</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>1 0 0</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1 0 1</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>1 1 0</td>
<td>○/●</td>
<td>●</td>
<td>○</td>
<td>●</td>
<td>○</td>
<td>●</td>
</tr>
<tr>
<td>1 1 1</td>
<td>○/●</td>
<td>●</td>
<td>○</td>
<td>●</td>
<td>○</td>
<td>●</td>
</tr>
</tbody>
</table>

- $f_1, f_2, f_3, f_4$ are the only hypotheses you need to consider.
- You just don’t know which one out of the four to choose!
- You won’t do better the random guess.
- So you haven’t learned anything from the training data.
- Learning is infeasible.
The Power of Probability

Unknown target function
\[ f : \mathcal{X} \to \mathcal{Y} \]

\[ \mathcal{D} = \{(x_1, y_1), \ldots, (x_N, y_N)\} \]

Hypothesis Set
\[ \mathcal{H} = \{h_1, \ldots, h_M\} \]

Final Hypothesis
\[ g(x) \approx f(x) \]

Learning Algorithm
\[ \mathcal{A} \]

Training Data

unknown input distribution
\[ p(x) \]

\[ x_1, \ldots, x_N \rightarrow x \]
Outline

Today’s Lecture:
- What constitutes a learning problem?
  - Training and testing samples
  - Target and Hypothesis function
  - Learning Model
- Is learning feasible?
  - An example
  - The power of probability
- Training versus Testing
  - In-sample error
  - Out-sample error
  - Probability bound
In-Sample Error

- Let $x_n$ be a *training* sample
- $h$: Your hypothesis
- $f$: The unknown target function
- If $h(x_n) = f(x_n)$, then say training sample $x_n$ is correctly classified.
- This will give you the **in-sample error**

**Definition (In-sample Error / Training Error)**

Consider a training set $D = \{x_1, \ldots, x_N\}$, and a target function $f$. The **in-sample error** (or the training error) of a hypothesis function $h \in \mathcal{H}$ is the empirical average of $\{h(x_n) \neq f(x_n)\}$:

$$E_{in}(h) \overset{\text{def}}{=} \frac{1}{N} \sum_{n=1}^{N} \left[ h(x_n) \neq f(x_n) \right],$$

(1)

where $[\cdot] = 1$ if the statement inside the bracket is true, and $= 0$ if the statement is false.
Out-Sample Error

- Let \( x \) be a testing sample drawn from \( p(x) \)
- \( h \): Your hypothesis
- \( f \): The unknown target function
- If \( h(x) = f(x) \), then say testing sample \( x \) is correctly classified.
- Since \( x \sim p(x) \), you need to compute the probability of error, called the **out-sample error**

**Definition (Out-sample Error / Testing Error)**

Consider an input space \( \mathcal{X} \) containing elements \( x \) drawn from a distribution \( p_X(x) \), and a target function \( f \). The **out-sample error** (or the testing error) of a hypothesis function \( h \in \mathcal{H} \) is

\[
E_{out}(h) \overset{\text{def}}{=} \mathbb{P}[h(x) \neq f(x)],
\]

where \( \mathbb{P}[\cdot] \) measures the probability of the statement based on the distribution \( p_X(x) \).
In-sample VS Out-sample

In-Sample Error

\[ E_{\text{in}}(h) = \frac{1}{N} \sum_{n=1}^{N} \left[ h(x_n) \neq f(x_n) \right] \]

Out-Sample Error

\[ E_{\text{out}}(h) = \mathbb{P}[h(x) \neq f(x)] \]

\[ = \left[ h(x_n) \neq f(x_n) \right] \mathbb{P}\left\{ h(x_n) \neq f(x_n) \right\} \]

\[ = \left[ h(x_n) = f(x_n) \right] \left( 1 - \mathbb{P}\left\{ h(x_n) \neq f(x_n) \right\} \right) \]

\[ = \mathbb{E}\left\{ \left[ h(x_n) \neq f(x_n) \right] \right\} \]
The Role of $p(x)$

Learning is feasible if $x \sim p(x)$

$p(x)$ says: Training and testing are related

If training and testing are unrelated, then hopeless – the deterministic example shown previously

If you draw training and testing samples with different bias, then you will suffer
When Will $E_{\text{in}} = E_{\text{out}}$?

**Theorem (Hoeffding Inequality)**

Let $X_1, \ldots, X_N$ be a sequence of i.i.d. random variables such that $0 \leq X_n \leq 1$ and $\mathbb{E}[X_n] = \mu$. Then, for any $\epsilon > 0$,

$$\mathbb{P} \left[ \left| \frac{1}{N} \sum_{n=1}^{N} X_n - \mu \right| > \epsilon \right] \leq 2e^{-2\epsilon^2 N}. \quad (3)$$
When Will $E_{in} = E_{out}$?

- To us, the inequality can be stated as
  \[ \mathbb{P} \left[ |E_{in}(h) - E_{out}(h)| > \epsilon \right] \leq 2e^{-2\epsilon^2 N}. \]

- $N = \text{number of training samples}$
- $\epsilon = \text{tolerance level}$
- Hoeffding is applicable because $[h(x) \neq f(x)]$ is either 1 or 0.
Appendix