CSE 417T: Introduction to Machine Learning
Lecture 14: Three Learning Principles

Henry Chai
10/10/19
Three Learning Principles

- Occam’s Razor
- Sampling Bias
- Data Snooping
Occam’s Razor

- “It is futile to do with more things that which can be done with fewer"
Occam’s Razor

- "Nature operates in the shortest way possible"
- “An explanation of the data should be made as simple as possible, but no simpler”
- "When you hear hoofbeats, think of horses not zebras”
- The simplest model that fits the data is also the most plausible
Occam’s Razor

- "Nature operates in the shortest way possible"
- “An explanation of the data should be made as simple as possible, but no simpler”
- "When you hear hoofbeats, think of horses not zebras”
- The simplest model that fits the data is also the most plausible
Simple model

\[ E_{in} \quad E_{out} \]

Number of training points, \( n \)

Expected error

Complex model

\[ E_{in} \quad E_{out} \]

Number of training points, \( n \)

Expected error
Simple Models

- Simple hypotheses, $h$
  - Low order polynomials
  - Linear models with small weights
  - Easily described in words (or bits)

- Or...

- Simple hypothesis sets, $\mathcal{H}$
  - Low VC-dimension
  - Small number of hypotheses
  - Small number of free parameters
A hypothesis set of simple hypotheses must be small.

Suppose a hypothesis, $h$, can be described using $l$ bits.

Then the hypothesis set $\mathcal{H}$ that contains all such hypotheses is of size $2^l$. 
Case Study #1

- Suppose I tell you that I’ve found a 10\textsuperscript{th}-order polynomial that perfectly fits my dataset of 10 points.

- Should you believe that the true function is a 10\textsuperscript{th}-order polynomial?
Case Study #1

• Suppose I tell you that I’ve found a line that perfectly fits my dataset of 10 points.

• Should you believe that the true function is a line?
Axiom of Non-falsifiability

- If an experiment has no chance of falsifying a hypothesis, then the result of that experiment provides no evidence one way or the other for the hypothesis.
Counterpoint: Hickam’s Dictum

- “A man can have as many diseases as he damn well pleases”
Sampling Bias

• If the data is sampled in a biased way, learning will produce a similarly biased outcome
Which of the following conclusions are affected by sampling bias?

I am bad at school, all of my friends get better grades than me!

I am not a great teacher, I get really bad course reviews!

I am not very successful, just look at my peers' LinkedIn profiles!

I am really behind as a grad student, my labmates all have more publications than me!
Case Study #2

- A late election poll by the Chicago Daily Tribune had Dewey so far ahead of Truman that they ran the following headline the day after election day:
A late election poll by the Chicago Daily Tribune had Dewey so far ahead of Truman that they ran the following headline the day after election day:

\[ P\{|E_{in}(h) - E_{out}(h)| > \epsilon\} \leq 2e^{-2\epsilon^2n} \]
Case Study #3

Source: https://me.me/i/newsweek-madam-president-lIan-clintons-ejourney-wiitelluune-dewey-defeats-truman-8726228
Case Study #3

Case Study #3

Source: https://projects.fivethirtyeight.com/2016-election-forecast/
Case Study #3

Clintoon-Trump Probably Won’t Be The Next ‘Dewey Defeats Truman’

- “the biggest difference between 1948 and 2016 may be our polling methods... the way in which participants were selected”
- “don’t expect 'Clinton Defeats Trump’ to become one of the great blunders of our political history"

**Formal Setup**

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

- **Hypothesis Set**
  \[ \mathcal{H} \]

- **Probability Distribution**
  \[ \mathcal{P} \text{ on } \mathcal{X} \]

- **Training data**
  \[ \mathcal{D} = \{(x_1, y_1), \ldots, (x_n, y_n)\} \]

- **Learning Algorithm**
  \[ \mathcal{A} \]

- **Learned Hypothesis**
  \[ g: \mathcal{X} \rightarrow \mathcal{Y} \]
Training-Test Mismatch

- Suppose your training data comes from some distribution $\mathcal{P}$ and you know that the true distribution over all inputs is some other distribution $\mathcal{P}'$

- You can reweight or resample your training data to make it look like it came from the distribution $\mathcal{P}'$
  - Requires that you know $\mathcal{P}'$
  - Requires complete representation of the true distribution in the training distribution
Data Snooping

• If a data set has affected any step in the learning process, its ability to assess the outcome has been compromised

• Ask yourself: “If the data were different, could/would I have done something different?”
Case Study #4

• Given some dataset \( \mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\} \) where \( \mathcal{X} = \mathbb{R}^d \) and \( \mathcal{Y} = \mathbb{R} \), you first normalize the outputs (can be helpful for some learning algorithms)

\[
y'_i = \frac{y_i - \bar{y}}{\sqrt{V(y)}} \quad \text{where} \quad \bar{y} \text{ is the mean of } \{y_1, y_2, \ldots, y_n\} \text{ and } V(y) \text{ is the variance of } \{y_1, y_2, \ldots, y_n\}
\]

• Partition the dataset \( \mathcal{D}' = \{(x_1', y'_1), (x_2', y'_2), \ldots, (x_n', y'_n)\} \) into test and training data \( \mathcal{D}'_{\text{test}} \) and \( \mathcal{D}'_{\text{train}} \)

• \( \mathcal{D}'_{\text{test}} \) affected the training process via the normalization!
Case Study #4

- Given some dataset $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$ where $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \mathbb{R}$, you first partition the dataset into test and training data $\mathcal{D}_{test}$ and $\mathcal{D}_{train}$.

- Normalize the training data outputs, save the mean and variance and use those to normalize the test data outputs.

- $\mathcal{D}_{test}$ no longer affects $\mathcal{D}_{train}$ and therefore, no longer affects the training process.
Case Study #5

- Given some data you fit a linear model and find that the in-sample error is high.
- So you fit a quadratic model; turns out that makes the in-sample error even worse.
- You do some reading and find that others have tried SVMs on the same data with no success.
- Time to get serious: break out the neural networks!
Data Reuse

- Trying different models on the same data set will eventually lead to “success”
- Account for data reuse by computing the combined VC dimension of all models (including what others tried)
<table>
<thead>
<tr>
<th>First Half of the Course: Foundations</th>
<th>Second Half of the Course: Techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td>Theory, Proofs, Math, Probability, Boring Stuff, etc...</td>
<td>Random Forests! Support Vector Machines! Neural Networks! Yay!</td>
</tr>
</tbody>
</table>