Hypothesis Testing and Computational Learning Theory

Doug Downey EECS 349 Fall 2010

With slides from Bryan Pardo, Tom Mitchell
Overview

• Hypothesis Testing: How do we know our learners are “good”?  
  – What does performance on test data imply/guarantee about future performance?

• Computational Learning Theory: Are there general laws that govern learning?  
  – Sample Complexity: How many training examples are needed to learn a successful hypothesis?  
  – Computational Complexity: How much computational effort is needed to learn a successful hypothesis?
Some terms

$X$ is the set of all possible instances

$C$ is the set of all possible concepts $c$

where $c : X \rightarrow \{0,1\}$

$H$ is the set of hypotheses considered by a learner, $H \subseteq C$

$L$ is the learner

$D$ is a probability distribution over $X$ that generates observed instances
The **true error** of hypothesis $h$, with respect to the target concept $c$ and observation distribution $D$ is the probability that $h$ will misclassify an instance drawn according to $D$

$$\text{error}_D \equiv P \left[ c(x) \neq h(x) \right]_{x \in D}$$

In a perfect world, we’d like the true error to be 0
Definition

• The **sample error** of hypothesis $h$, with respect to the target concept $c$ and sample $S$ is the proportion of $S$ that $h$ misclassifies:

\[
\text{error}_S(h) = \frac{1}{|S|} \sum_{x \in S} \delta(c(x), h(x))
\]

where $\delta(c(x), h(x))$ returns 1 iff $c(x) = h(x)$
Problems Estimating Error

1. **Bias**: If $S$ is training set, $error_S(h)$ is optimistically biased

\[ bias \equiv E[error_S(h)] - error_D(h) \]

For unbiased estimate, $h$ and $S$ must be chosen independently

2. **Variance**: Even with unbiased $S$, $error_S(h)$ may still vary from $error_D(h)$
Example on Independent Test Set

Hypothesis $h$ misclassifies 12 of the 40 examples in $S$

$$error_{S}(h) = \frac{12}{40} = 0.30$$

What is $error_{D}(h)$?
Estimators

Experiment:

1. choose sample $S$ of size $n$ according to distribution $\mathcal{D}$

2. measure $\text{error}_S(h)$

$\text{error}_S(h)$ is a random variable (i.e., result of an experiment)

$\text{error}_S(h)$ is an unbiased estimator for $\text{error}_D(h)$

Given observed $\text{error}_S(h)$ what can we conclude about $\text{error}_D(h)$?
Confidence Intervals

If

- $S$ contains $n$ examples, drawn independently of $h$ and each other
- $n \geq 30$ and $n \cdot \text{error}_S(h)$, $n \cdot (1 - \text{error}_S(h))$ each $> 5$

Then

- With approximately 95% probability, $\text{error}_D(h)$ lies in interval

$$
\text{error}_S(h) \pm 1.96 \sqrt{\frac{\text{error}_S(h)(1 - \text{error}_S(h))}{n}}
$$
Confidence Intervals

• Under same conditions...

• With approximately N% probability, $error_D(h)$ lies in interval

$$error_S(h) \pm z_N \sqrt{\frac{error_S(h)(1 - error_S(h))}{n}}$$

where

<table>
<thead>
<tr>
<th>N%</th>
<th>50%</th>
<th>68%</th>
<th>80%</th>
<th>90%</th>
<th>95%</th>
<th>98%</th>
<th>99%</th>
</tr>
</thead>
<tbody>
<tr>
<td>$z_N$</td>
<td>0.67</td>
<td>1.00</td>
<td>1.28</td>
<td>1.64</td>
<td>1.96</td>
<td>2.33</td>
<td>2.58</td>
</tr>
</tbody>
</table>
Life Skills

• “Convincing demonstration” that certain enhancements improve performance?

• Use online Fisher Exact or Chi Square tests to evaluate hypotheses, e.g:
  – http://people.ku.edu/~preacher/chisq/chisq.htm
Overview

• Hypothesis Testing: How do we know our learners are “good”? 
  – What does performance on test data imply/guarantee about future performance?

• Computational Learning Theory: Are there general laws that govern learning?
  – Sample Complexity: How many training examples are needed to learn a successful hypothesis?
  – Computational Complexity: How much computational effort is needed to learn a successful hypothesis?
Computational Learning Theory

• Are there general laws that govern learning?
  – *No Free Lunch Theorem:* The expected accuracy of *any* learning algorithm across all concepts is 50%.

• But can we still say something positive?
  – Yes.
  – *Probably Approximately Correct* (PAC) learning
The world isn’t perfect

• If we can’t provide every instance for training, a consistent hypothesis may have error on unobserved instances.

• How many training examples do we need to bound the likelihood of error to a reasonable level?
  When is our hypothesis Probably Approximately Correct (PAC)?
Definitions

• A hypothesis is consistent if it has zero error on training examples

• The **version space** \((VS_{H,T})\) is the set of all hypotheses consistent on training set \(T\) in our **hypothesis space** \(H\)
  
  – (reminder: hypothesis space is the set of concepts we’re considering, e.g. depth-2 decision trees)
Definition: $\varepsilon$-exhausted

**IN ENGLISH:**

The set of hypotheses consistent with the training data $T$ is $\varepsilon$-exhausted if, when you test them on the actual distribution of instances, all consistent hypotheses have error below $\varepsilon$.

**IN MATH:**

$$V S_{H,T} \text{ is } \varepsilon \text{-exhausted for concept } c$$

and sample distribution on $D$, if....

$$\forall \ h \in V S_{H,T} \ , \ error_D (h) < \varepsilon$$
A Theorem

If hypothesis space $H$ is finite, & training set $T$ contains $m$ independent randomly drawn examples of concept $c$

THEN, for any $0 \leq \varepsilon \leq 1$

$$P(VS_{H,T} \text{ is NOT } \varepsilon \text{-exhausted}) \leq |H|e^{-\varepsilon m}$$
Proof of Theorem

If hypothesis $h$ has true error $\varepsilon$, the probability of it getting a single random example right is:

$$P(h \text{ got } 1 \text{ example right}) = 1 - \varepsilon$$

Ergo the probability of $h$ getting $m$ examples right is:

$$P(h \text{ got } m \text{ examples right}) = (1 - \varepsilon)^m$$
Proof of Theorem

If there are $k$ hypotheses in $H$ with error at least $\varepsilon$, call the probability at least of those $k$ hypotheses got $m$ instances right $P(\text{at least one bad } h \text{ looks good})$.

This prob. is BOUNDED by $k (1-\varepsilon)^m$

$P(\text{at least one bad } h \text{ looks good}) \leq k (1-\varepsilon)^m$

"Union" bound
Proof of Theorem (continued)

Since \( k \leq |H| \), it follows that \( k(1-\varepsilon)^m \leq |H|(1-\varepsilon)^m \)

If \( 0 \leq \varepsilon \leq 1 \), then \((1 - \varepsilon) \leq e^{-\varepsilon}\)

Therefore. ..

\[ P(\text{at least one bad } h \text{ looks good }) \leq k(1-\varepsilon)^m \leq |H|(1-\varepsilon)^m \leq |H|e^{-\varepsilon m} \]

Proof complete!

We now have a bound on the likelihood that a hypothesis is consistent with the training data will have error \( \geq \varepsilon \)
Using the theorem

Let's rearrange to see how many training examples we need to set a bound $\delta$ on the likelihood our true error is $\varepsilon$.

\[ |H| e^{-\varepsilon m} \leq \delta \]

\[ \ln \left( |H| e^{-\varepsilon m} \right) \leq \ln(\delta) \]

\[ \ln (|H|) + \ln \left( e^{-\varepsilon m} \right) \leq \ln (\delta) \]

\[ \ln (|H|) - \varepsilon m \leq \ln (\delta) \]

\[ \ln (|H|) - \ln (\delta) \leq \varepsilon m \]

\[ \frac{1}{\varepsilon} \left( \ln (|H|) - \ln (\delta) \right) \leq m \]

\[ \frac{1}{\varepsilon} \left( \ln (|H|) + \ln \left( \frac{1}{\delta} \right) \right) \leq m \]
Probably Approximately Correct (PAC)

\[
\frac{1}{\varepsilon} \left( \ln \left( \left| H \right| \right) - \ln \left( \delta \right) \right) \leq m
\]

- \( \varepsilon \): The worst error we'll tolerate
- \( |H| \): Hypothesis space size
- \( \delta \): The likelihood a hypothesis consistent with the training data will have error \( \varepsilon \)
- \( m \): Number of training examples
Using the bound

\[ \frac{1}{\varepsilon} \left( \ln \left( |H| \right) - \ln \left( \delta \right) \right) \leq m \]

Plug in \( \varepsilon, \delta, \) and \( H \) to get a number of training examples \( m \) that will “guarantee” your learner will generate a hypothesis that is Probably Approximately Correct.

NOTE: This assumes that the concept is actually IN \( H, \) that \( H \) is finite, and that your training set is drawn using distribution \( D \)
Problems with PAC

- The PAC Learning framework has 2 disadvantages:
  1) It can lead to weak bounds
  2) Sample Complexity bound cannot be established for infinite hypothesis spaces

- We introduce the VC dimension for dealing with these problems
**Def:** A set of instances $S$ is shattered by hypothesis set $H$ iff for every possible concept $c$ on $S$ there exists a hypothesis $h$ in $H$ that is consistent with that concept.
Can a linear separator shatter this?

The ability of H to shatter a set of instances is a measure of its capacity to represent target concepts defined over those instances.
Can a quadratic separator shatter this?

This sounds like a homework problem....
Vapnik-Chervonenkis Dimension

**Def:** The Vapnik-Chervonenkis dimension, $\text{VC}(H)$ of hypothesis space $H$ defined over instance space $X$ is the size of the largest finite subset of $X$ shattered by $H$. If arbitrarily large finite sets can be shattered by $H$, then $\text{VC}(H)$ is infinite.
How many training examples needed?

- Upper bound on $m$ using $\text{VC}(H)$

$$m \geq \frac{1}{\varepsilon} \left( 4 \log_2 \left( \frac{2}{\delta} \right) + 8 \text{VC}(H) \log_2 \left( \frac{13}{\varepsilon} \right) \right)$$

- Lower bound on $m$ using $\text{VC}(C)$

$$\max \left[ \frac{1}{\varepsilon} \log(1/\delta), \frac{\text{VC}(C) - 1}{32 \varepsilon} \right]$$

There's more on this in the textbook.