Lecture 8

Learning Theory

“No Free Lunch” Theorems

\[ \text{Acc}_D(L) = \text{Generalization accuracy of learner } L \]
\[ = \text{Accuracy of } L \text{ on non-training examples} \]
\[ F = \text{Set of all possible concepts, } y = f(x) \]

**Theorem:** For any learner \( L \), \[ \frac{1}{n} \sum_{i=1}^{n} \text{Acc}_D(L_i) = \frac{1}{2} \]
*(given any distribution \( D \) over \( x \) and training set size \( n \))

**Proof sketch:** Given any training set \( S \):

For every concept \( f \) where \( \text{Acc}_D(L) = \frac{1}{2} + \delta \),
there is a concept \( f' \) where \( \text{Acc}_D(L) = \frac{1}{2} - \delta \).
\[ \forall x \in S, f'(x) = f(x) = y. \forall x \not\in S, f'(x) = f(x). \]

**Corollary:** For any two learners \( L_1, L_2 \):

If \( \exists \) learning problem s.t. \( \text{Acc}_D(L_1) > \text{Acc}_D(L_2) \)
Then \( \exists \) learning problem s.t. \( \text{Acc}_D(L_2) > \text{Acc}_D(L_1) \)

What Does This Mean in Practice?

- Don’t expect your favorite learner to always be best
- Try different approaches and compare
- But how could (say) a multilayer perceptron be less accurate than a single-layer one?

Bias and Variance

- Bias-variance decomposition is key tool for understanding learning algorithms
- Helps explain why simple learners can outperform powerful ones
- Helps explain why model ensembles outperform single models
- Helps understand & avoid overfitting
- Standard decomposition for squared loss
- Can be generalized to zero-one loss

Definitions

- Given training set: \( \{ (x_1, t_1), \ldots, (x_n, t_n) \} \)
- Learner induces model: \( y = f(x) \)
- Loss measures quality of learner’s predictions
  - Squared loss: \( L(t, y) = (t - y)^2 \)
  - Absolute loss: \( L(t, y) = |t - y| \)
  - Zero-one loss: \( L(t, y) = 0 \) if \( y = t \), 1 otherwise
  - Etc.
- Loss = Bias + Variance + Noise
  (This lecture: ignore noise; see paper)
**Bias**

![Bias Diagram](image)

**Variance**

![Variance Diagram](image)

**Decomposition for squared loss**

\[
(t - y)^2 = (t - \bar{y} + \bar{y} - y)^2 = (t - \bar{y})^2 + (\bar{y} - y)^2 + 2(t - \bar{y})(\bar{y} - y)
\]

\[
E[(t - y)^2] = E[(t - \bar{y})^2] + E[(\bar{y} - y)^2] + 2E[(t - \bar{y})(\bar{y} - y)]
\]

- **Exp. loss** = **Bias** + **Variance**
- (Expectations are over training sets)

**How to generalize this to other lossfuncs?**

\[
E[(t - y)^2] = (t - \bar{y})^2 + E[(\bar{y} - y)^2]
\]

\[
(a - b)^2 \rightarrow L(a, b)
\]

\[
E[(t - y)^2] \rightarrow E[L(t, y)]
\] (Exp. loss)

\[
(t - \bar{y})^2 \rightarrow L(t, \bar{y})
\] (Bias)

\[
E[(\bar{y} - y)^2] \rightarrow E[L(\bar{y}, y)]
\] (Variance)

**But what should \( \bar{y} \) be?**

Define **Main Prediction**

Prediction with min average loss relative to all predictions

\[
\bar{y}_L = \arg\min \ E[L(y, y')]
\]

- Squared loss: \( \bar{y} = \text{Mean} \)
- Absolute loss: \( \bar{y} = \text{Median} \)
- Zero-one loss: \( \bar{y} = \text{Mode} \)

**Generalized definitions**

- **Bias** = Loss incurred by main prediction = \( L(t, \bar{y}) \)
- **Variance** = Average loss incurred by prediction relative to main prediction = \( E[L(y, \bar{y})] \)

These definitions have all the required properties.

For zero-one loss:

\[
\text{Bias} = \begin{cases} 
0 & \text{if main prediction is correct} \\
1 & \text{otherwise} 
\end{cases}
\]

\[
\text{Variance} = \text{Prob}(\text{Prediction} \neq \text{Main pred}) = P(y \neq \bar{y})
\]
Can we decompose zero-one loss into these?
Assume two-class problem.

\[
\text{Bias} = 0 \implies \text{Loss} = \text{Bias} + \text{Variance}
\]
\[
\text{Loss} = P(y \neq t)
\]
\[
\text{Bias} = 0 \implies y = t
\]

\[
\text{Bias} = 1 \implies \text{Loss} = \text{Bias} - \text{Variance}
\]
\[
\text{Loss} = P(y \neq t) = 1 - P(y = t) = 1 - P(y \neq \bar{y})
\]
because if \( y \neq \bar{y} \) then \( y = t \implies y \neq \bar{y} \).

Increasing variance can reduce loss!

Can we generalize this further?

\[
\text{Loss} = \text{Bias} + c \cdot \text{Variance}
\]
where \( c = 1 \) if Bias = 0, otherwise see below

- Applies to:
  - Squared loss: \( c = 1 \)
  - Two-class problems: \( c = -1 \)
  - Multiclass problems: \( c = -P(y = t | y \neq \bar{y}) \)
  - Variable cost: \( c = -L(t, \bar{y})/L(\bar{y}, t) \)

Metric loss functions

- What about loss functions where decomposition does not apply?
- For any metric loss function:
  \[
  \text{Loss} \leq \text{Bias} + \text{Variance}
  \]
  \[
  \text{Loss} \geq \max \{ \text{Bias} - \text{Var}, \text{Var} - \text{Bias} \}
  \]

\[
\begin{array}{c|c|c}
\text{Bias} & \text{Var} & \text{Var} \\
\hline
0 & & \\
\end{array}
\]

Possible values of loss

Loss

PAC Learning

- Overfitting happens because training error is bad estimate of generalization error
- Can we infer something about generalization error from training error?
- Overfitting happens when the learner doesn’t see “enough” examples
- Can we estimate how many examples are enough?

Problem Setting

Given:
- Set of instances \( X \)
- Set of hypotheses \( H \)
- Set of possible target concepts \( C \)
- Training instances generated by a fixed, unknown probability distribution \( D \) over \( X \)

Learner observes sequence \( D \) of training examples \( (x, c(x)) \), for some target concept \( c \in C \)
- Instances \( x \) are drawn from distribution \( D \)
- Teacher provides target value \( c(x) \) for each

Learner must output a hypothesis \( h \) estimating \( c \)
- \( h \) is evaluated by its performance on subsequent instances drawn according to \( D \)

Note: probabilistic instances, noise-free classifications
True Error of a Hypothesis

Definition: The true error (denoted $\text{error}_T(h)$) of hypothesis $h$ with respect to target concept $c$ and distribution $D$ is the probability that $h$ will misclassify an instance drawn at random according to $D$.

$$\text{error}_T(h) = \mathbb{P}_{x \sim D}[c(x) \neq h(x)]$$

Version Spaces

Version Space $V_{S,H,D}$:
Subset of hypotheses in $H$ consistent with training data $D$

$$V_{S,H,D} = \{ h \in H | c(x) = h(x), \forall x \in S \}$$

($\epsilon = \text{training error}, \text{error} = \text{true error}$)

Interesting! This bounds the probability that any consistent learner will output a hypothesis $h$ with $\text{error}(h) \geq \epsilon$

If we want this probability to be at most $\delta$

$$|H|e^{-\epsilon m} \leq \delta$$

then

$$m \geq \frac{1}{\epsilon}(\ln |H| + \ln(1/\delta))$$

Two Notions of Error

Training error of hypothesis $h$ with respect to target concept $c$
- How often $h(x) \neq c(x)$ over training instances

True error of hypothesis $h$ with respect to $c$
- How often $h(x) \neq c(x)$ over future random instances

Our concern:
- Can we bound the true error of $h$ given the training error of $h$?
- First consider when training error of $h$ is zero

How Many Examples Are Enough?

Theorem:
If the hypothesis space $H$ is finite, and $D$ is a sequence of $m \geq 1$ independent random examples of some target concept $c$, then for any $0 \leq \epsilon \leq 1$, the probability that $V_{S,H,D}$ contains a hypothesis with error greater than $\epsilon$ is less than $|H|e^{-\epsilon m}$

Proof sketch:
Prob(1 hyp. w/ error > $\epsilon$ consistent w/ 1 ex.) < $1 - \epsilon \leq e^{-\epsilon}$
Prob(1 hyp. w/ error > $\epsilon$ consistent with m exs.) < $e^{-\epsilon m}$
Prob(1 of $|H|$ hyps. consistent with m exs.) < $|H|e^{-\epsilon m}$

Learning Conjunctions

How many examples are sufficient to ensure with probability at least $(1 - \delta)$ that every $h$ in $V_{S,H,D}$ satisfies $\text{error}_T(h) \leq \epsilon$?

Use our theorem:

$$m \geq \frac{1}{\epsilon}(\ln |H| + \ln(1/\delta))$$

Suppose $H$ contains conjunctions of constraints on up to $n$ Boolean attributes (i.e., $n$ literals). Then $|H| = 3^n$, and

$$m \geq \frac{1}{\epsilon}(\ln 3^n + \ln(1/\delta))$$

$$\geq \frac{1}{\epsilon}(n \ln 3 + \ln(1/\delta))$$
How About PlayTennis?
1 attribute with 3 values (outlook)
9 attributes with 2 values (temp, humidity, wind, etc.)
Language: Conjunctive features or null concept
\(|H| = 4 \times 3^9 + 1 = 78733\)
\(m \geq \frac{1}{\epsilon} (\ln 78733 + \ln(1/\delta))\)
If we want to ensure that with probability 90%,
\(V_S\) contains only hypotheses with error \(\leq 10\%\),
then it is sufficient to have \(m\) examples, where
\(m \geq \frac{1}{0.1} (\ln 78733 + \ln(1/0.05)) = 143\)
(# examples in domain: \(3 \times 2^9 = 1536\))

PAC Learning
Consider a class \(C\) of possible target concepts defined over
a set of instances \(X\) of length \(n\), and a learner \(L\) using
hypothesis space \(H\).

**Definition:** \(C\) is PAC-learnable by \(L\) using \(H\) iff
for all \(\epsilon < \delta < 1/2\), distributions \(D\) over \(X\), \(\epsilon\) such that
any hypothesis \(h \in H\) such that
error \(h \leq \epsilon\), in time that is polynomial in \(1/\epsilon, 1/\delta, n\) and size(\(C\)).

Agnostic Learning
So far, assumed \(e \in H\)
Agnostic learning setting: don’t assume \(e \in H\)
• What can we say in this case?
  • Hoeffding bounds:
    \(Pr(\text{error}(\hat{h}) > \text{error}(h) + \epsilon) \leq e^{-2m\epsilon^2}\)
  • For hypothesis space \(H\):
    \(Pr(\text{error}(h) > \text{error}(\hat{h}) + \epsilon) \leq |H|e^{-2m\epsilon^2}\)
• What is the sample complexity in this case?
  \(m \geq \frac{1}{2\epsilon^2} (\ln |H| + \ln(1/\delta))\)

VC Dimension
• What about hypotheses with numeric parameters?
• Solution: Use VC dimension instead of \(\ln |H|\)

Shattering a Set of Instances
*Definition:* a **dichotomy** of a set \(S\) is a partition
of \(S\) into two disjoint subsets.

*Definition:* a set of instances \(S\) is **shattered** by
hypothesis space \(H\) if and only if for every
dichotomy of \(S\) there exists some hypothesis in \(H\)
consistent with this dichotomy.

Three Instances Shattered
Instance space \(X\)
The Vapnik-Chervonenkis Dimension

Definition: The Vapnik-Chervonenkis dimension, $VC(H)$, 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 of $X$ can be shattered by $H$, then $VC(H) = \infty$.

VC Dim. of Linear Decision Surfaces

VC dim. of hyperplane in $d$-dimensional space is $d + 1$

Sample Complexity from VC Dimension

How many randomly drawn examples suffice to guarantee error of at most $\epsilon$ with probability at least $(1 - \delta)$?

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

Support Vector Machines

- Many different hyperplanes can separate positive and negative examples
- Choose hyperplane with maximum margin
- Margin: Min. distance between plane and example
- Bound on VC dimension decreases with margin
- Support vectors: Examples that determine the plane
- $E[error(h)] \leq \frac{E[\#support vectors]}{\#training vectors} + 1$
- Noisy data: use slack variables
- Avoids overfitting even in very high-dimensional spaces (e.g., text)
- Non-linear: augment data with derived features

Support Vector Machines

Learning Theory: Summary

- “No free lunch” theorems
- Bias and variance
- PAC learning
- VC dimension
- Support vector machines