A simple setting…

- Classification
  - \( m \) data points
  - Finite number of possible hypothesis (e.g., dec. trees of depth \( d \))

- A learner finds a hypothesis \( h \) that is consistent with training data
  - Gets zero error in training: \( \text{error}_{\text{train}}(h) = 0 \)

- What is the probability that \( h \) has more than \( \varepsilon \) true error?
  - \( \text{error}_{\text{true}}(h) \geq \varepsilon \)
How likely is a bad hypothesis to get \( m \) data points right?

- Hypothesis \( h \) that is **consistent** with training data → got \( m \) i.i.d. points right
  - h "bad" if it gets all this data right, but has high true error
- Prob. \( h \) with \( \text{error}_{\text{true}}(h) \geq \varepsilon \) gets one data point right
  \[ P(h \text{ gets one point right}) \leq 1 - \varepsilon \]

- Prob. \( h \) with \( \text{error}_{\text{true}}(h) \geq \varepsilon \) gets \( m \) data points right
  \[ P(h \text{ gets } m \text{ i.i.d. points right}) \leq (1 - \varepsilon)^m \]
  - exponentially small (as \( m \) increases)

But there are many possible hypothesis that are consistent with training data
How likely is learner to pick a bad hypothesis

- Prob. \( h \) with \( \text{error}_{\text{true}}(h) \geq \varepsilon \) gets \( m \) data points right
  \[ P(h_{\text{bad consistent with data}}) \leq (1-\varepsilon)^m \]
- There are \( k \) hypothesis consistent with data
  - How likely is learner to pick a bad one?
  \[
P(\exists h \text{ that is bad and consistent with data})
  = P(h_1 \text{ bad consistent } \lor h_2 \text{ bad consistent } \lor \ldots \lor h_k \text{ bad consistent})
  \]

Union bound

- \( P(A \text{ or } B \text{ or } C \text{ or } D \text{ or } \ldots) \leq P(A) + P(B) + P(C) + \ldots \)
How likely is learner to pick a bad hypothesis

- Prob. $h$ with $\text{error}_{true}(h) \geq \varepsilon$ gets $m$ data points right
  \[ P(h \text{ bad, consistent}) \leq (1-\varepsilon)^m \]

- There are $k$ hypotheses consistent with data
  - How likely is learner to pick a bad one?
  \[ P(\exists \text{ bad } h \text{ consistent with data}) \leq k (1-\varepsilon)^m \leq |H| (1-\varepsilon)^m \leq |H| e^{-m\varepsilon} \]

Review: Generalization error in finite hypothesis spaces [Haussler ’88]

- **Theorem**: Hypothesis space $H$ finite, dataset $D$ with $m$ i.i.d. samples, $0 < \varepsilon < 1$: for any learned hypothesis $h$ that is consistent on the training data:
  \[ P(\text{error}_{true}(h) \geq \varepsilon) \leq |H| e^{-m\varepsilon} \]
Using a PAC bound

Typically, 2 use cases:
1. Pick \( \varepsilon \) and \( \delta \), give you \( m \)
2. Pick \( m \) and \( \delta \), give you \( \varepsilon \)

\[ P(\text{error}_{true}(h) > \varepsilon) \leq |H|e^{-me} \]

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

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

Review: Generalization error in finite hypothesis spaces [Haussler '88]

**Theorem:** Hypothesis space \( H \) finite, dataset \( D \) with \( m \) i.i.d. samples, \( 0 < \varepsilon < 1 \): for any learned hypothesis \( h \) that is consistent on the training data:

\[ P(\text{error}_{true}(h) > \varepsilon) \leq |H|e^{-me} \]

Even if \( h \) makes zero errors in training data, may make errors in test.
Limitations of Haussler ‘88 bound

\[ P(\text{error}_{\text{true}}(h) > \epsilon) \leq |H|e^{-m\epsilon} \]

- Consistent classifier

- Size of hypothesis space

    \[ |h| \leq |H| \]

What if our classifier does not have zero error on the training data?

- A learner with zero training errors may make mistakes in test set

- What about a learner with \( \text{error}_{\text{train}}(h) \) in training set?

    \[ \text{no longer assume } \text{error}_{\text{train}}(h) = 0 \]
Simpler question: What’s the expected error of a hypothesis?

- The error of a hypothesis is like estimating the parameter of a coin!

\[ P \left( \theta - \frac{1}{m} \sum_{i} x_i > \epsilon \right) \leq e^{-2m\epsilon^2} \]

Chernoff bound: for \( m \) i.i.d. coin flips, \( x_1, \ldots, x_m \), where \( x_i \in \{0,1\} \). For \( 0 < \epsilon < 1 \):

\[ \hat{\theta}_{\text{MLE}} = \frac{1}{m} \sum_{i} x_i \]

Using Chernoff bound to estimate error of a single hypothesis

\[ P \left( \theta - \frac{1}{m} \sum_{i} x_i > \epsilon \right) \leq e^{-2m\epsilon^2} \]
But we are comparing many hypothesis: **Union bound**

For each hypothesis $h_i$:

$$ P(\text{error}_{true}(h_i) - \text{error}_{train}(h_i) > \epsilon) \leq e^{-2m\epsilon^2} $$

What if I am comparing two hypothesis, $h_1$ and $h_2$?

$$ P(\text{error}_{true}(h_1) - \text{error}_{true}(h_2) > \epsilon) \leq 2e^{-2m\epsilon^2} $$

**Generalization bound for $|H|$ hypothesis**

- **Theorem:** Hypothesis space $H$ finite, dataset $D$ with $m$ i.i.d. samples, $0 < \epsilon < 1$ : for any learned hypothesis $h$:

$$ P(\text{error}_{true}(h) - \text{error}_{train}(h) > \epsilon) \leq |H|e^{-2m\epsilon^2} \leq \delta $$

with prob. $1-\delta$:

$$ \text{error}_{true}(h) - \text{error}_{train}(h) \leq \sqrt{\frac{\ln|H| + \ln\delta}{2m}} \leq \epsilon $$
PAC bound and Bias-Variance tradeoff

\[ P(\text{error}_\text{true}(h) - \text{error}_\text{train}(h) > \epsilon) \leq |H|e^{-2me^2} \]

or, after moving some terms around, with probability at least 1-\(\delta\):

\[ \text{error}_\text{true}(h) \leq \text{error}_\text{train}(h) + \sqrt{\frac{\ln |H| + \ln \frac{1}{\delta}}{2m}} \]

- Important: PAC bound holds for all \(h\), but doesn't guarantee that algorithm finds best \(h\)!!!
Boolean formulas with \( n \) binary features

\[
m \geq \frac{1}{2e^2} \left( \ln |H| + \ln \frac{1}{\delta} \right)
\]

\( H \): any Boolean formula
\( n \) binary attributes

\[
x_1 \land \ldots \land x_n
\]

\[
0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad \ldots
\]

\[
0 
\]

\[
\ln |H| = 2^n k!
\]

\[
\text{Huge, very big}
\]

\[
H_k = \text{Number of decision trees of depth } k
\]

\[
H_0 = 2
\]

\[
H_{k+1} = (\text{#choices of root attribute}) \times
\]

\[
(\text{# possible left subtrees}) \times
\]

\[
(\text{# possible right subtrees})
\]

\[
H_{k+1} = n \times H_k \times H_k
\]

\[
L_k = \log_2 H_k
\]

\[
L_0 = 1
\]

\[
L_{k+1} = \log_2 n + 2L_k
\]

\[
\text{So } L_k = (2^2 - 1) \left( 1 + \log_2 n \right) + 1
\]
PAC bound for decision trees of depth $k$

$$m \geq \frac{\ln 2}{2e^2} \left( (2^k - 1)(1 + \log_2 n) + 1 + \ln \frac{1}{\delta} \right)$$

- Bad!!!
  - Number of points is exponential in depth!

- But, for $m$ data points, decision tree can’t get too big…
  - no more than $m$ leaves

Number of leaves never more than number data points

Number of decision trees with $k$ leaves

$H_k = \text{Number of decision trees with } k \text{ leaves}$
$H_0 = 2$

$$H_{k+1} = n \sum_{i=1}^{k} H_i H_{k+1-i}$$

Loose bound:
$$H_k = n^{k-1} (k + 1)^{2k-1}$$

Reminder:
$$|\text{DTs depth } k| = 2 \cdot (2n)^{2^k - 1}$$
PAC bound for decision trees with $k$ leaves – Bias-Variance revisited

$$H_k = n^{k-1}(k + 1)^2k-1$$

$$error_{true}(h) \leq error_{train}(h) + \sqrt{\frac{\ln |Z| + \ln \frac{1}{\delta}}{2m}}$$

$$error_{true}(h) \leq error_{train}(h) + \sqrt{\frac{(k - 1) \ln n + (2k - 1) \ln (k + 1) + \ln \frac{1}{\delta}}{2m}}$$

Announcements

- **Midterm:**
  - Thursday Oct. 25th, Thursday 5-6:30pm, MM A14
  - All content up to, and including SVMs and Kernels
    - Not learning theory
  - any book, class notes, your printouts of class materials that are on the class website, including my annotated slides and relevant readings, and Andrew Moore’s tutorials. You cannot use materials brought by other students.
  - Calculators are not necessary.
  - No laptops, PDAs or cellphones.
What did we learn from decision trees?

- Bias-Variance tradeoff formalized

\[
\text{error}_\text{true}(h) \leq \text{error}_\text{train}(h) + \sqrt{\frac{(k - 1) \ln n + (2k - 1) \ln(k + 1) + \ln \frac{1}{\delta}}{2m}}
\]

- Moral of the story:
  Complexity of learning not measured in terms of size hypothesis space, but in maximum number of points that allows consistent classification
  - Complexity \( m \) – no bias, lots of variance
  - Lower than \( m \) – some bias, less variance

What about continuous hypothesis spaces?

- Continuous hypothesis space:
  - \(|H| = \infty\)
  - Infinite variance???

- As with decision trees, only care about the maximum number of points that can be classified exactly!
How many points can a linear boundary classify exactly? (1-D)

- - can do 2
  - + can do 3
  + - -
  + - -
  + +
  - +

How many points can a linear boundary classify exactly? (2-D)

- can do 3
  + - can't do 4
  + - -
  + - -
  + +
  + +
  - +
  - +

©2005-2007 Carlos Guestrin
How many points can a linear boundary classify exactly? (d-D)

\[ w_0 + \sum_{i=1}^{d} w_i x_i = d + 1 \]

PAC bound using VC dimension

- Number of training points that can be classified exactly is VC dimension!!

  Measures relevant size of hypothesis space, as with decision trees with k leaves

\[
\text{error}_{\text{true}}(h) \leq \text{error}_{\text{train}}(h) + \sqrt{\frac{2 \ln \frac{2m}{|H|} + 1}{m}} + \ln \frac{4}{\delta}
\]

only depend on VC(|H|)
not on |H|
Shattering a set of points

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.

VC 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 $VC(H) \equiv \infty$. 
PAC bound using VC dimension

- Number of training points that can be classified exactly is VC dimension!!!
  - Measures relevant size of hypothesis space, as with decision trees with k leaves
  - Bound for infinite dimension hypothesis spaces:

\[
\text{error}_{\text{true}}(h) \leq \text{error}_{\text{train}}(h) + \sqrt{\frac{VC(H) \left( \ln \frac{2m}{VC(H)} + 1 \right) + \ln \frac{4}{\delta}}{2n}}
\]

Examples of VC dimension

- Linear classifiers:
  - \( VC(H) = d+1 \), for \( d \) features plus constant term \( b \)

- Neural networks
  - \( VC(H) = \#\text{parameters} \)
  - Local minima means NNs will probably not find best parameters

- 1-Nearest neighbor?
Another VC dim. example -

What can we shatter?

- What’s the VC dim. of decision stumps in 2d?

Another VC dim. example -

What can’t we shatter?

- What’s the VC dim. of decision stumps in 2d?
What you need to know

- Finite hypothesis space
  - Derive results
  - Counting number of hypothesis
  - Mistakes on Training data

- Complexity of the classifier depends on number of points that can be classified exactly
  - Finite case – decision trees
  - Infinite case – VC dimension

- Bias-Variance tradeoff in learning theory
- Remember: will your algorithm find best classifier?
What you have learned thus far

- Learning is function approximation
- Point estimation
- Regression
- Naive Bayes
- Logistic regression
- Bias-Variance tradeoff
- Neural nets
- Decision trees
- Cross validation
- Boosting
- Instance-based learning
- SVMs
- Kernel trick
- PAC learning
- VC dimension
- Margin bounds
- Mistake bounds

Review material in terms of...

- Types of learning problems
- Hypothesis spaces
  - what they can represent
- Loss functions
- Optimization algorithms
This is a very incomplete view!!!