Learning Theory

PAC-learning, VC Dimension and Margin-based Bounds (cont.)

Machine Learning – 10701/15781

Carlos Guestrin

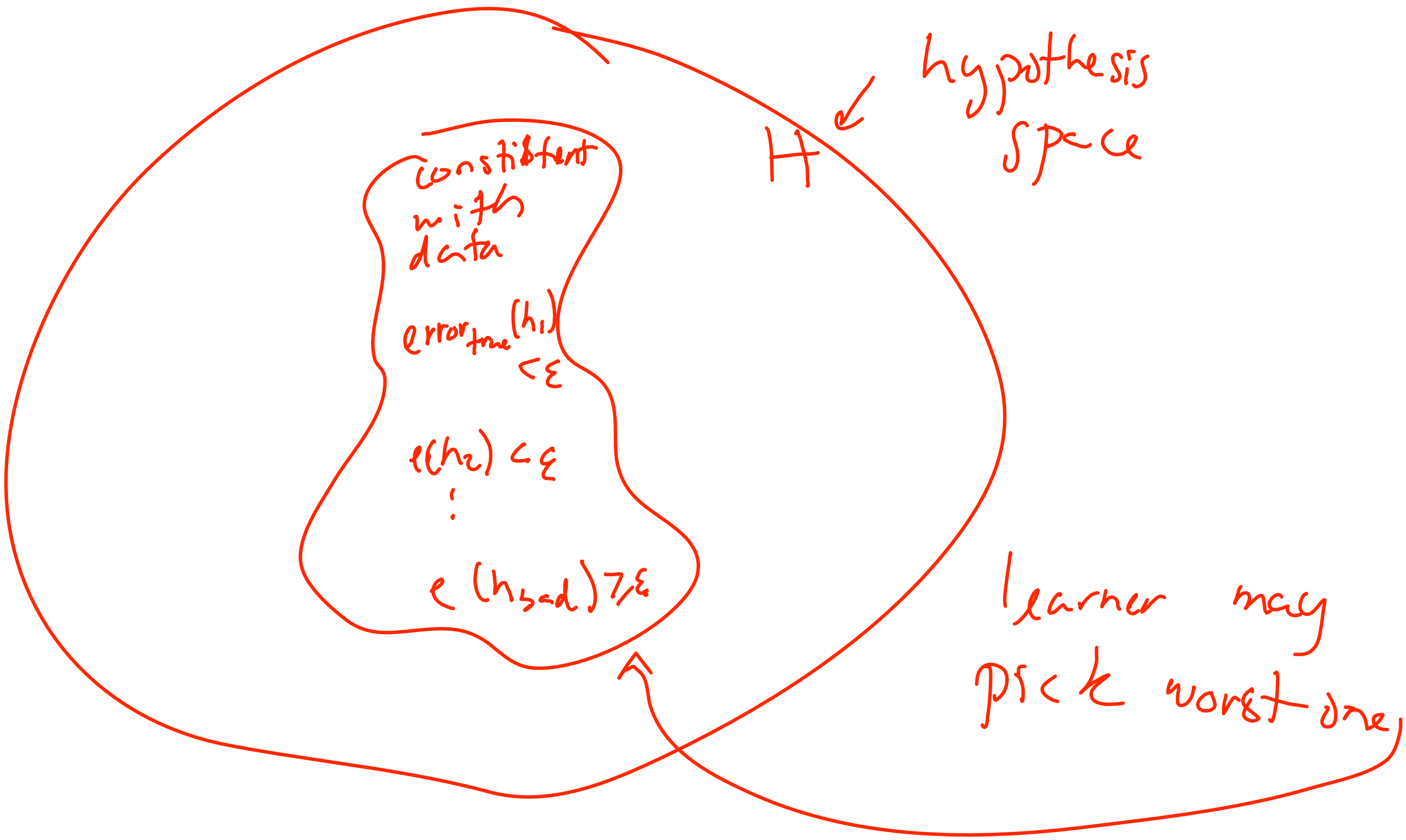
Carnegie Mellon University

March 5th, 2007

A simple setting…

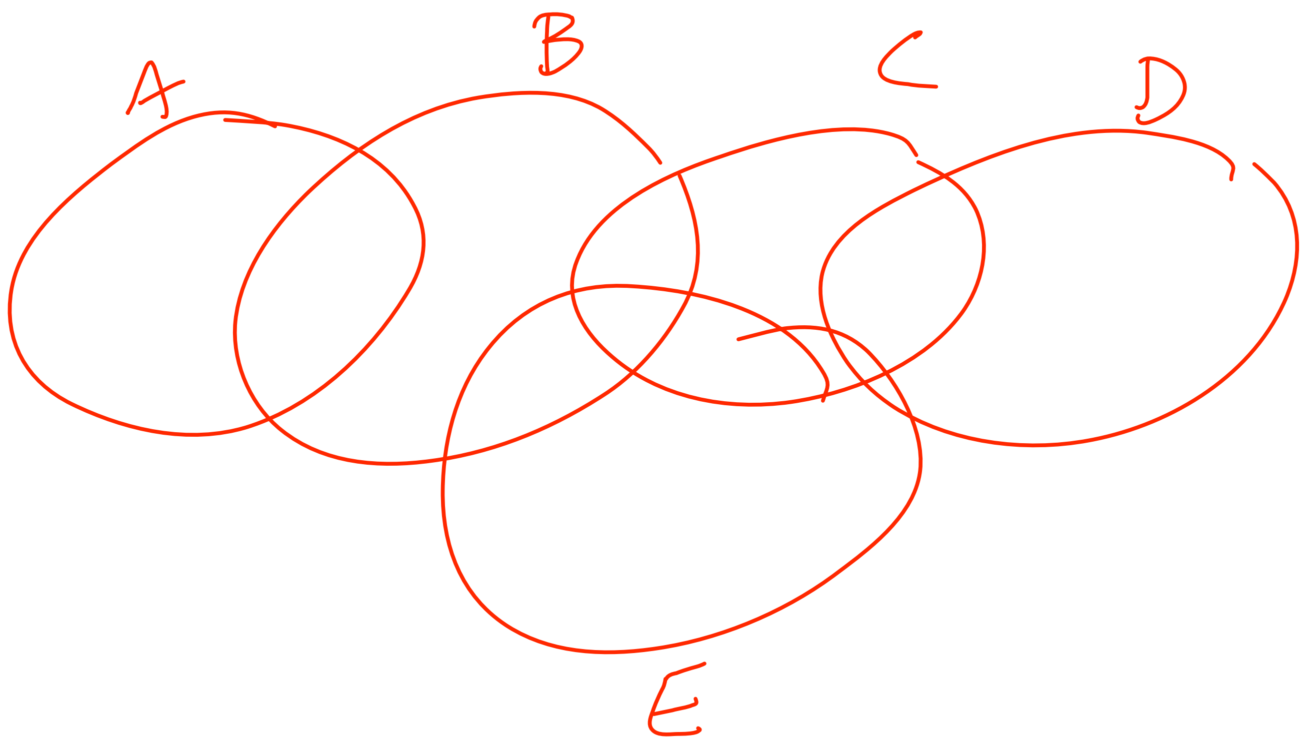
- Classification
  - m data points
  - **Finite** number of possible hypothesis (e.g., dec. trees of depth d) on categorical data
- A learner finds a hypothesis *h* that is **consistent** with training data
  - Gets zero error in training – $\text{error}_{train}(h) = 0$
- What is the probability that *h* has more than ε true error?
  - $\text{error}_{true}(h) > \varepsilon$

But there are many possible hypothesis that are consistent with training data



Union bound

- P(A or B or C or D or …) $\leq P(A) + P(B) + P(C) + \dots$



How likely is learner to pick a bad hypothesis

$(1-\varepsilon)^m \leq (e^{-\varepsilon})^m = e^{-\varepsilon m}$

- Prob. h_i with $\text{error}_{\text{true}}(h_i) \geq \varepsilon$ gets m data points right

$P(e_t(h_i) \geq \varepsilon$ & consistent with m datapoints$) \leq (1-\varepsilon)^m$

- There are k hypothesis consistent with data
  - How likely is learner to pick a bad one?

$P(e_t(h_1) \geq \varepsilon$ & h_1 consistent with m $\vee$ $e_t(h_2) \geq \varepsilon$ & consistent $\vee \ldots \vee e_t(h_k) \geq \varepsilon$ & consistent$)$

$\leq \sum_i P(e_t(h_i) \geq \varepsilon$ & consistent with m data points$)$

$\leq k(1-\varepsilon)^m$

$\leq |H|(1-\varepsilon)^m \leq |H| e^{-\varepsilon m}$!

$\boxed{1-\varepsilon \leq e^{-\varepsilon} \quad \varepsilon \geq 0}$

simplify eqns.

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

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

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

as $m \rightarrow$ increases $\Rightarrow$ Prob. make a bad decision decrease exponentially fast

as $|H| \rightarrow$ increases $\Rightarrow$ Chances of making a bad decision increase linearly with $|H|$

Using a PAC bound

I want: $\text{error}_{true}(h) \leq \varepsilon$
guarantee with high prob.
guarantee with prob. $\geq 1-\delta$

PAC: Probably Approximately Correct

- Typically, 2 use cases: $P(\text{error}_{true}(h) > \epsilon) \leq |H|e^{-m\epsilon}$
  - 1: Pick ε and δ, give you *m*
  - 2: Pick m and δ, give you ε

1: e.g., $\varepsilon \leq 0.1$

$1-\delta \geq 0.95$ I am right

$\delta \geq |H| e^{-m\varepsilon}$

$\ln \delta \geq \ln |H| - m\varepsilon$

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

\# points you need

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:

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

Even if *h* makes zero errors in training data, may make errors in test

Limitations of Haussler '88 bound

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

- Consistent classifier

- Size of hypothesis space

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 $error_{train}(h)$ in training set?

Simpler question: What's the expected error of a hypothesis?

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

- Chernoff bound: for *m* i.i.d. coin flips, $x_1,\ldots,x_m$, where $x_i \in \{0,1\}$. For $0<\varepsilon<1$:

$$\boxed{P\left(\theta - \frac{1}{m}\sum_i x_i > \epsilon\right) \leq e^{-2m\epsilon^2}}$$

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:

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

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

Generalization bound for |H| hypothesis

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

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

PAC bound and Bias-Variance tradeoff

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

or, after moving some terms around,
with probability at least 1-δ:

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

- **Important: PAC bound holds for all *h*,**

but doesn't guarantee that algorithm finds best *h*!!!

What about the size of the hypothesis space?

$$m \geq \frac{1}{2\epsilon^2}\left(\ln|H| + \ln\frac{1}{\delta}\right)$$

- How large is the hypothesis space?

Boolean formulas with *n* binary features

$$m \geq \frac{1}{2\epsilon^2}\left(\ln|H| + \ln\frac{1}{\delta}\right)$$

Number of decision trees of depth k

$$m \geq \frac{1}{2\epsilon^2}\left(\ln|H| + \ln\frac{1}{\delta}\right)$$

Recursive solution

Given *n* attributes

H_k = Number of decision trees of depth k

H_0 =2

H_{k+1} = (#choices of root attribute) *

(# possible left subtrees) *

(# possible right subtrees)

= n * H_k * H_k

Write $L_k = \log_2 H_k$

$L_0 = 1$

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

So $L_k = (2^k-1)(1+\log_2 n) + 1$

PAC bound for decision trees of depth k

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

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

- But, for *m* data points, decision tree can't get too big…

Number of leaves never more than number data points

Number of decision trees with k leaves

$$m \geq \frac{1}{2\epsilon^2}\left(\ln|H| + \ln\frac{1}{\delta}\right)$$

H_k = Number of decision trees with k 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:

$$|\text{DTs depth } k| = 2 * (2n)^{2^k-1}$$

PAC bound for decision trees with k leaves – Bias-Variance revisited

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

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

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

Announcements

- Midterm on Wednesday
  - Open book and notes, no other material
  - Bring a calculator
  - No laptops, PDAs or cellphones

What did we learn from decision trees?

- Bias-Variance tradeoff formalized

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

- 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?

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

- Continuous hypothesis space:
  - |H| = 1
  - 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)

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

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

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**

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

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:**

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

Examples of VC dimension

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

- Linear classifiers:
  - VC(H) = d+1, for *d* features plus constant term *b*
- Neural networks
  - VC(H) = #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?

Big Picture

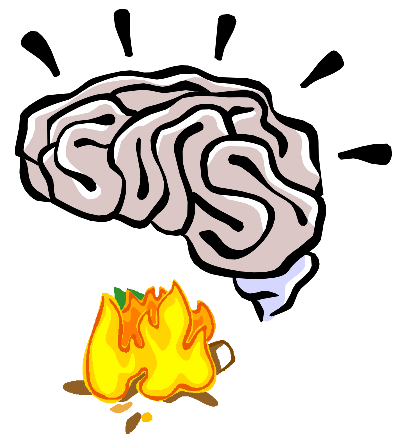
Machine Learning – 10701/15781

Carlos Guestrin
Carnegie Mellon University

March 5th, 2007

What you have learned thus far

- Learning is function approximation
- Point estimation
- Regression
- Naïve 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
- Loss functions
- Optimization algorithms

BIG PICTURE

(a few points of comparison)

learning task: DE, Cl, Reg

loss function: LL, Mrg, RMS

DE	density estimation
Cl	Classification
Reg	Regression
LL	Log-loss/MLE
Mrg	Margin-based
RMS	Squared error

Boosting
Cl, exp-loss

Naïve Bayes
DE, LL

SVM regression
Reg, Mrg

Logistic regression
DE, LL

SVMs
Cl, Mrg

kernel regression
Reg, RMS

Instance-based Learning
DE,Cl,Reg

Neural Nets
DE,Cl,Reg,RMS

Decision trees
DE,Cl,Reg

linear regression
Reg, RMS

This is a very incomplete view!!!