Computational Learning Theory

[read Chapter 7]
[Suggested exercises: 7.1, 7.2, 7.5, 7.8]

- Computational learning theory
- Setting 1: learner poses queries to teacher
- Setting 2: teacher chooses examples
- Setting 3: randomly generated instances, labeled by teacher
- Probably approximately correct (PAC) learning
- Vapnik-Chervonenkis Dimension
- Mistake bounds

Computational Learning Theory

What general laws constrain inductive learning?

We seek theory to relate:

- Probability of successful learning
- Number of training examples
- Complexity of hypothesis space
- Accuracy to which target function is approximated
- Manner in which training examples presented

Training Examples for EnjoySport

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

What is the general concept?

Representing Hypotheses

Many possible representations

Here, h is conjunction of constraints on attributes

Each constraint can be

- a specfic value (e.g., $Water = Warm$)
- don't care (e.g., "$Water =?$")
- no value allowed (e.g., "Water=$\emptyset$")

For example,

$$\begin{array}{cccccc} \text{Sky} & \text{AirTemp} & \text{Humid} & \text{Wind} & \text{Water} & \text{Forecst} \\ \langle Sunny & ? & ? & Strong & ? & Same\rangle \end{array}$$

Prototypical Concept Learning Task

- **Given:**
  - Instances X: Possible days, each described by the attributes *Sky, AirTemp, Humidity, Wind, Water, Forecast*
  - Target function c: $EnjoySport : X \rightarrow \{0, 1\}$
  - Hypotheses H: Conjunctions of literals. E.g.

$$\langle ?, Cold, High, ?, ?, ? \rangle.$$

  - Training examples D: Positive and negative examples of the target function

$$\langle x_1, c(x_1) \rangle, \ldots \langle x_m, c(x_m) \rangle$$

- **Determine:**
  - A hypothesis h in H such that $h(x) = c(x)$ for all x in D?
  - A hypothesis h in H such that $h(x) = c(x)$ for all x in X?

Version Spaces

A hypothesis h is **consistent** with a set of training examples D of target concept c if and only if $h(x) = c(x)$ for each training example $\langle x, c(x) \rangle$ in D.

$$Consistent(h, D) \equiv (\forall \langle x, c(x) \rangle \in D)\ h(x) = c(x)$$

The **version space**, $VS_{H,D}$, with respect to hypothesis space H and training examples D, is the subset of hypotheses from H consistent with all training examples in D.

$$VS_{H,D} \equiv \{h \in H | Consistent(h, D)\}$$

Sample Complexity

How many training examples are sufficient to learn the target concept?

1. If learner proposes instances, as queries to teacher
   - Learner proposes instance x, teacher provides $c(x)$
2. If teacher (who knows c) provides training examples
   - teacher provides sequence of examples of form $\langle x, c(x) \rangle$
3. If some random process (e.g., nature) proposes instances
   - instance x generated randomly, teacher provides $c(x)$

Sample Complexity: 1

Learner proposes instance x, teacher provides $c(x)$ (assume c is in learner's hypothesis space H)

Optimal query strategy: play 20 questions

- pick instance x such that half of hypotheses in VS classify x positive, half classify x negative
- When this is possible, need $\lceil \log_2 |H| \rceil$ queries to learn c
- when not possible, need even more

Sample Complexity: 2

Teacher (who knows c) provides training examples (assume c is in learner's hypothesis space H)

Optimal teaching strategy: depends on H used by learner

Consider the case $H =$ conjunctions of up to n boolean literals and their negations

> e.g., $(AirTemp = Warm) \wedge (Wind = Strong)$, where $AirTemp, Wind, \ldots$ each have 2 possible values.

- if n possible boolean attributes in H, $n+1$ examples suffice
- why?

Sample Complexity: 3

Given:

- set of instances X
- set of hypotheses H
- set of possible target concepts C
- training instances generated by a fixed, unknown probability distribution $\mathcal{D}$ over X

Learner observes a sequence D of training examples of form $\langle x, c(x) \rangle$, for some target concept $c \in C$

- instances x are drawn from distribution $\mathcal{D}$
- teacher provides target value $c(x)$ for each

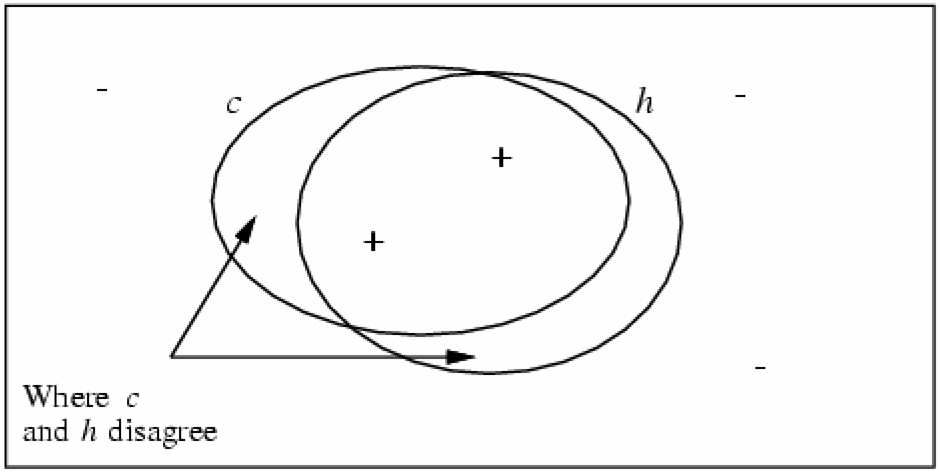
Learner must output a hypothesis h estimating c

- h is evaluated by its performance on subsequent instances drawn according to $\mathcal{D}$

Note: randomly drawn instances, noise-free classifications

True Error of a Hypothesis

Instance space X



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

$$error_{\mathcal{D}}(h) \equiv \Pr_{x \in \mathcal{D}}[c(x) \neq h(x)]$$

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

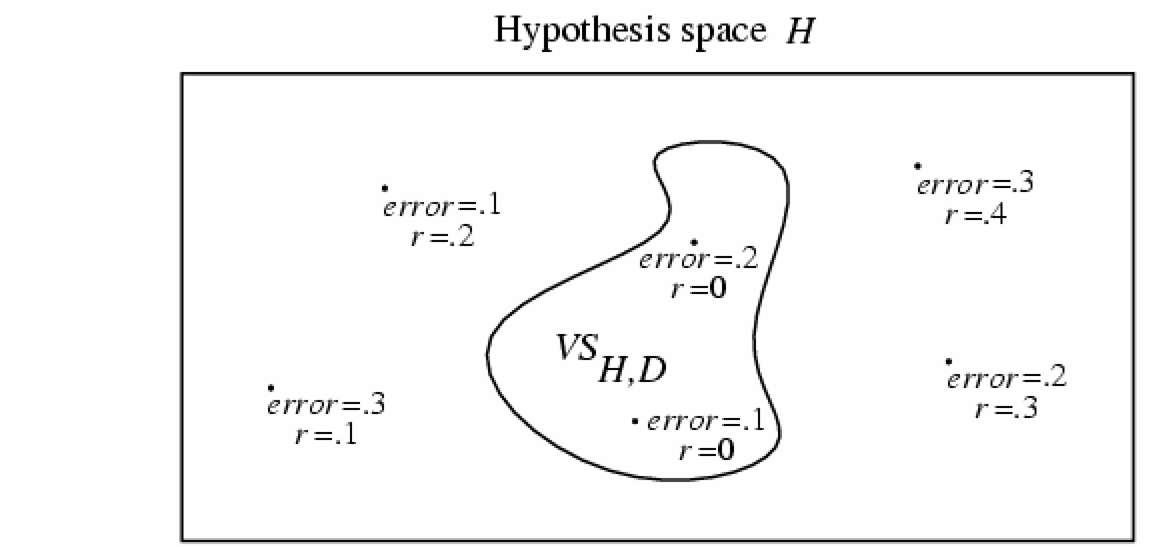
True error of hypothesis h with respect to c

- 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 (i.e., $h \in VS_{H,D}$)

Exhausting the Version Space



(r = training error, $error$ = true error)

Definition: The version space $VS_{H,D}$ is said to be ϵ-**exhausted** with respect to c and $\mathcal{D}$, if every hypothesis h in $VS_{H,D}$ has true error less than ϵ with respect to c and $\mathcal{D}$.

$$(\forall h \in VS_{H,D})\ error_{\mathcal{D}}(h) < \epsilon$$

How many examples will ϵ-exhaust the VS?

Theorem: [Haussler, 1988].

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 the version space with respect to H and D is not ϵ-exhausted (with respect to c) is less than

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

Any learner that outputs a hypothesis consistent with all training examples

Interesting! This bounds the probability that <u>any consistent learner</u> will output a hypothesis h with $error(h) \geq \epsilon$

If we want to this probability to be below δ

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

then

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

Learning Conjunctions of Boolean Literals

How many examples are sufficient to assure with probability at least $(1-\delta)$ that

every h in $VS_{H,D}$ satisfies $error_{\mathcal{D}}(h) \leq \epsilon$

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 boolean literals). Then $|H| = 3^n$, and

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

or

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

How About *EnjoySport*?

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

If H is as given in *EnjoySport* then $|H| = 973$, and

$$m \geq \frac{1}{\epsilon}(\ln 973 + \ln(1/\delta))$$

... if want to assure that with probability 95%, VS contains only hypotheses with $error_{\mathcal{D}}(h) \leq .1$, then it is sufficient to have m examples, where

$$m \geq \frac{1}{.1}(\ln 973 + \ln(1/.05))$$

$$m \geq 10(\ln 973 + \ln 20)$$

$$m \geq 10(6.88 + 3.00)$$

$$m \geq 98.8$$

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 if for all $c \in C$, distributions $\mathcal{D}$ over X, ϵ such that $0 < \epsilon < 1/2$, and δ such that $0 < \delta < 1/2$,
learner L will with probability at least $(1 - \delta)$ output a hypothesis $h \in H$ such that $error_{\mathcal{D}}(h) \leq \epsilon$, in time that is polynomial in $1/\epsilon$, $1/\delta$, n and $size(c)$.

Agnostic Learning

So far, assumed $c \in H$

Agnostic learning setting: don't assume $c \in H$

- What do we want then?
  - The hypothesis h that makes fewest errors on training data
- What is sample complexity in this case?

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

derived from Hoeffding bounds:

$$Pr[error_{\mathcal{D}}(h) > error_D(h) + \epsilon] \leq e^{-2m\epsilon^2}$$

true error training error degree of overfitting

What if H is not finite?

- Can’t use our result for finite H

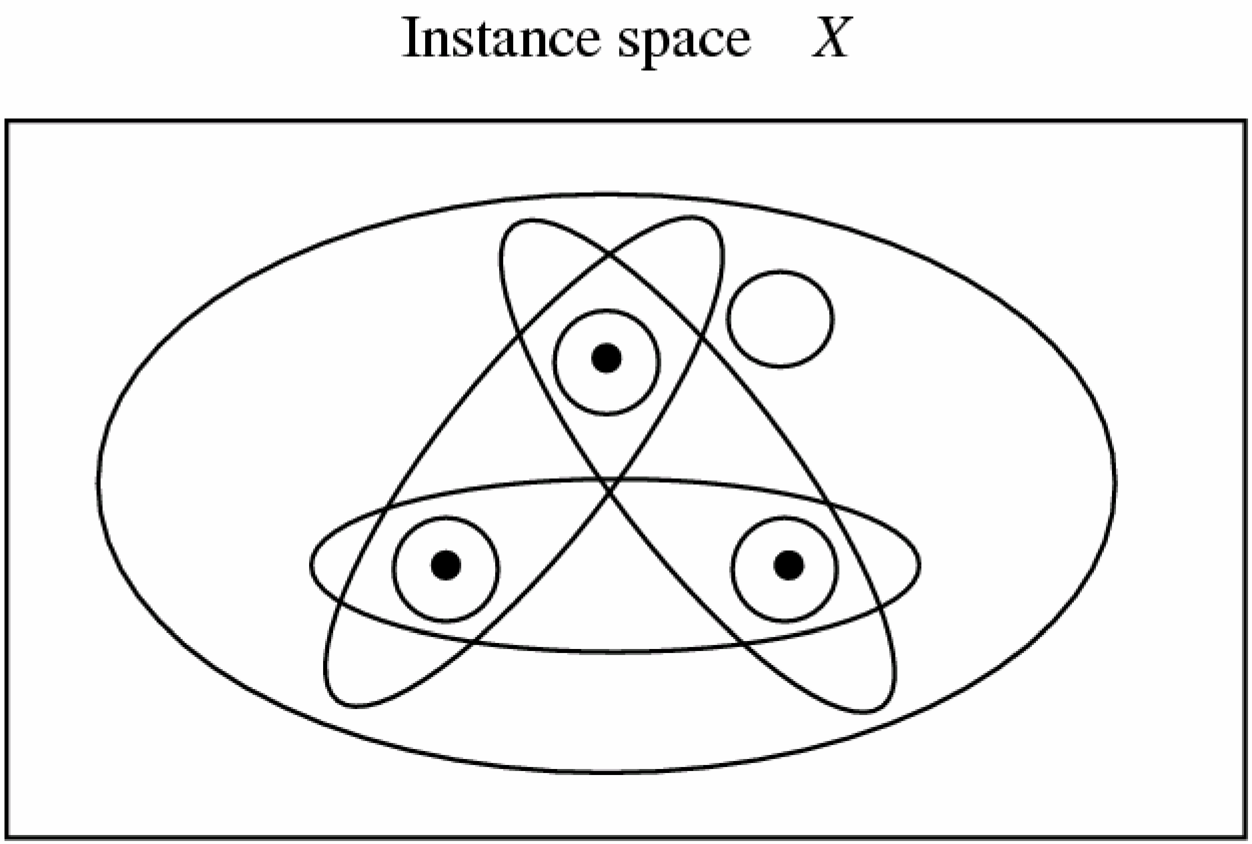
- Need some other measure of complexity for H
  – Vapnik-Chervonenkis dimension!

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



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 $VC(H) \equiv \infty$.

VC Dim. of Linear Decision Surfaces

(a) (b)

Sample Complexity from VC Dimension

How many randomly drawn examples suffice to ϵ-exhaust $VS_{H,D}$ with probability at least $(1-\delta)$?

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

Mistake Bounds

So far: how many examples needed to learn?

What about: how many mistakes before convergence?

Let's consider similar setting to PAC learning:

- Instances drawn at random from X according to distribution $\mathcal{D}$
- Learner must classify each instance before receiving correct classification from teacher
- Can we bound the number of mistakes learner makes before converging?

Mistake Bounds: Find-S

Consider Find-S when H = conjunction of boolean literals

Find-S:

- Initialize h to the most specific hypothesis $l_1 \wedge \neg l_1 \wedge l_2 \wedge \neg l_2 \ldots l_n \wedge \neg l_n$
- For each positive training instance x
  - Remove from h any literal that is not satisfied by x
- Output hypothesis h.

How many mistakes before converging to correct h?

Mistake Bounds: Halving Algorithm

1. Initialize VS to all hypotheses in H
2. For each training example,
   - remove from VS all hyps. that misclassify this example

Consider the Halving Algorithm:

- Learn concept using version space CANDIDATE-ELIMINATION algorithm
- Classify new instances by majority vote of version space members

How many mistakes before converging to correct h?

- ... in worst case?
- ... in best case?

Optimal Mistake Bounds

Let $M_A(C)$ be the max number of mistakes made by algorithm A to learn concepts in C. (maximum over all possible $c \in C$, and all possible training sequences)

$$M_A(C) \equiv \max_{c \in C} M_A(c)$$

Definition: Let C be an arbitrary non-empty concept class. The **optimal mistake bound** for C, denoted $Opt(C)$, is the minimum over all possible learning algorithms A of $M_A(C)$.

$$Opt(C) \equiv \min_{A \in learning\ algorithms} M_A(C)$$

$$\boxed{VC(C) \leq Opt(C) \leq M_{Halving}(C) \leq log_2(|C|).}$$

Weighted Majority Algorithm

---

a_i denotes the i^{th} prediction algorithm in the pool A of algorithms. w_i denotes the weight associated with a_i.

- For all i initialize $w_i \leftarrow 1$
- For each training example $\langle x, c(x) \rangle$
  * Initialize q_0 and q_1 to 0
  * For each prediction algorithm a_i
    · If $a_i(x) = 0$ then $q_0 \leftarrow q_0 + w_i$
    If $a_i(x) = 1$ then $q_1 \leftarrow q_1 + w_i$
  * If $q_1 > q_0$ then predict $c(x) = 1$
    If $q_0 > q_1$ then predict $c(x) = 0$
    If $q_1 = q_0$ then predict 0 or 1 at random for $c(x)$
  * For each prediction algorithm a_i in A do
    If $a_i(x) \neq c(x)$ then $w_i \leftarrow \beta w_i$

when β=0, equivalent to the Halving algorithm…

Weighted Majority

Even algorithms that learn or change over time…

[Relative mistake bound for WEIGHTED-MAJORITY] Let D be any sequence of training examples, let A be any set of n prediction algorithms, and let k be the minimum number of mistakes made by any algorithm in A for the training sequence D. Then the number of mistakes over D made by the WEIGHTED-MAJORITY algorithm using $\beta = \frac{1}{2}$ is at most

$$2.4(k + \log_2 n)$$