Computational Learning Theory

Overview

- Computational Learning Theory
- 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 concept is approximated
  - Manner in which training examples presented

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 example
   - Teacher provides sequence of examples of forms $<x,c(x)>$
3. If some random proces (e.g., nature) proposes instances
   - Instance x generated randomly, teacher provides $c(x)$

Target concept is the boolean-valued fn to be learned

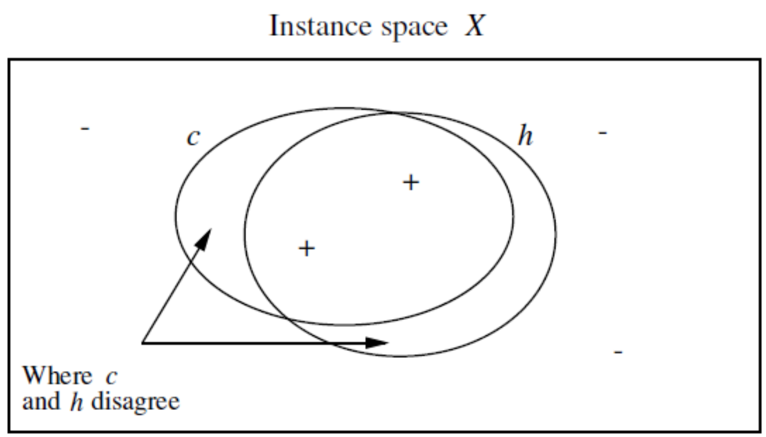
$c:X \rightarrow \{0,1\}$

Prototypical Concept Learning Task

- Given
  - Set of instances X
  - Set of Hyphoteses HTarget Function c:
  - Set of possible target concept c: $X \rightarrow \{0,1\}$
  - 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 $c \; \varepsilon \; C$
  - Instances x are drawn from distribution $\mathcal{D}$
  - Teacher provides target value c(x) for each instance
- Learner must output a hypothesis *h* estimatinc *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



- Definition: The **true error**, denoted by $error_D(h)$, of a 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.

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

Two Notions Error

Training error of hypothesis h with respect to target concept c

- How often $h(x) \neq c(x)$ over training instances D

$$error_D(h) \equiv \Pr_{x \in D}\left[c(x) \neq h(x)\right] \equiv \frac{\sum_{x \in D} \delta(c(x) \neq h(x))}{|D|}$$

Training examples

True error of hypothesis h with respect to c

- How often $h(x) \neq c(x)$ over future instances drawn at random from D

$$error_D(h) \equiv \Pr_{x \in D}\left[c(x) \neq h(x)\right]$$

Probability distribution P(x)

Our Concern:

- Can we bound the *true error* of h given the training error of h?

$$error_D(h) \equiv \Pr_{x \in D}\left[c(x) \neq h(x)\right] \equiv \frac{\sum_{x \in D} \delta(c(x) \neq h(x))}{|D|}$$

Training examples

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

Probability distribution P(x)

Can we bound
$\boldsymbol{error_{\mathcal{D}}(h)}$

in terms of
$\boldsymbol{error_D(h)}$

??

if D was a set of example drawn from $\mathcal{D}$ and independent of h, then we could use standard statistical confidence intervals to determine that with 95% probability, $error_{\mathcal{D}}(h)$ lies in interval :

$$error_D(h) \pm 1.96\sqrt{\frac{error_D(h)(1 - error_D(h))}{n}}$$

but D is the ***training data*** for h ...

Version Spaces

c : X → {0,1}

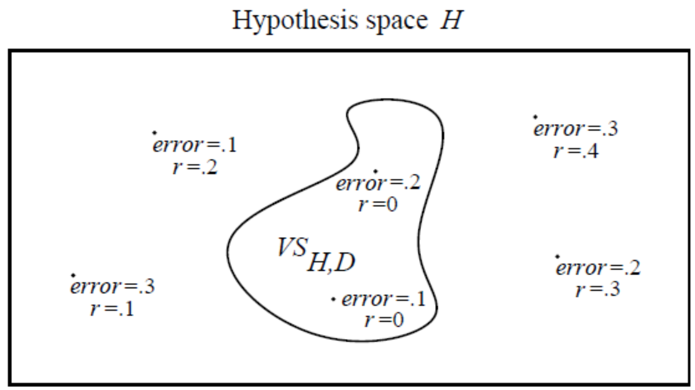
A hypothesis h is **consistent** with a set of training examples D of target concept c if and only if $h(x) = c(x)$ for each training example $<x,c(x)>$ in D

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

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.

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

Exhausting the Version Space



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

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

$$(\forall h \in VS_{H,D})\, error_D(h) < \in$$

How many examples will ∈-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

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

$$\Pr\left[(\exists h \in H) s.t. (error_{train}(h) = 0) \wedge (error_{true}(h) > \varepsilon)\right] \leq |H|e^{-\varepsilon m}$$

What it means

[Haussler, 1988] : probability that the version space is not ε-exhausted after m training examples is at most $|H|e^{-\varepsilon m}$ s

$$\Pr\left[(\exists h \in H) s.t. (error_{train}(h) = 0) \wedge (error_{true}(h) > \varepsilon)\right] \leq |H| e^{-\varepsilon m}$$

↑

Suppose we want this probability to be at most δ

1. How many training examples suffice ?

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

2. if $error_{train}(h) = 0$ then with probability at least (1- δ)

$$error_{true}(h) \leq \frac{1}{m}(\ln|H| + \ln(1/\delta))$$

Example : H is Conjunction of Boolean Literals

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

Consider classification problem f:X→Y:

- instances : $X = \langle X_1, X_2, X_3, X_4 \rangle$ where each X_i is boolean
- Learned hypotheses are rules of the form
  - IF $\langle X_1, X_2, X_3, X_4 \rangle = \langle 0, ?, 1, ? \rangle$, THEN Y=1, ELSE Y = 0
  - i.e., rules constrain any subset of the X_i

How many training examples m suffice to assure that with probability at least 0.99, *any* consistent learner will output a hypothesis with true error at most 0.05 ?

Example : H is Decision Tree with depth=2

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

Consider classification problem f:X→Y:

- instances : $X = <X_1, \ldots, X_N>$ where each X_i is boolean
- Learned hypotheses are decision trees of depth 2, using only two variables

How many training examples m suffice to assure that with probability at least 0.99, *any* consistent learner will output a hypothesis with true error at most 0.05 ?

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 error on training data
- What is sample complexity in this case ?

note ε here is the difference between the training error and true error

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

Derived from Hoeffdinng bounds :

$$\Pr\left[error_D(h) > error_D(h) + \varepsilon)\right] \leq e^{-2em^2}$$

true error

training error

degree of overfitting

PAC Learning

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

Definition : C is PAC-learnable by L using H if for all $c \in C$, distribution D over X, ε such that $0 < \varepsilon < ½$, and δ such that $0 < \delta < ½$,

learner L will with probability at least $(1- \delta)$ output a hypothesis $h \in H$ such that $error_{D}(H) \leq \varepsilon$,
<u>in time that is polynomial</u> in $1/ \varepsilon$, $1/ \delta$, and *size* (c)

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

Question : if H = {h|h:X→Y} is finite,

What measure of complexity should we use in place of |H| ?

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

Question : if H = {h|h:X→Y} is finite, What measure of complexity should we use in place of |H| ?

Answer : The largest subset of X for which H can guarantee zero training error (regardless of the target function c)

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

Question : if H = {h|h:X→Y} is finite,

What measure of complexity should we use in place of |H| ?

Answer : The largest subset of X for which H can guarantee zero training error (regardless of the target function c)

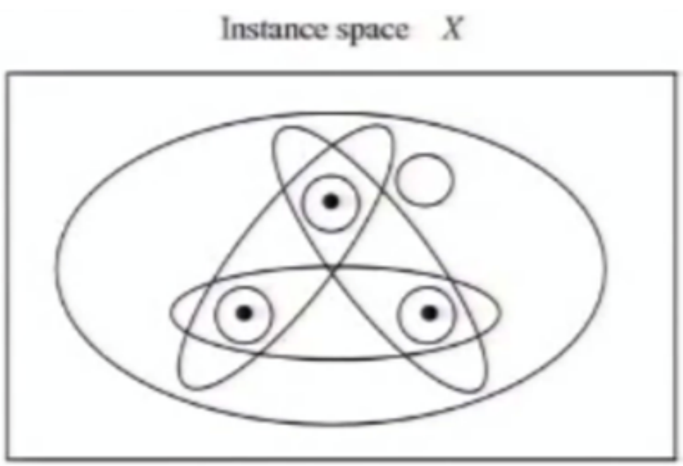
VC dimension of H is the size of this subset

Shattering Set of Instances

A labeling of each member of S as positive or negative

Definition : a **dichotomy** of a set *S* is a partition of *S* into two disjoint subsets.

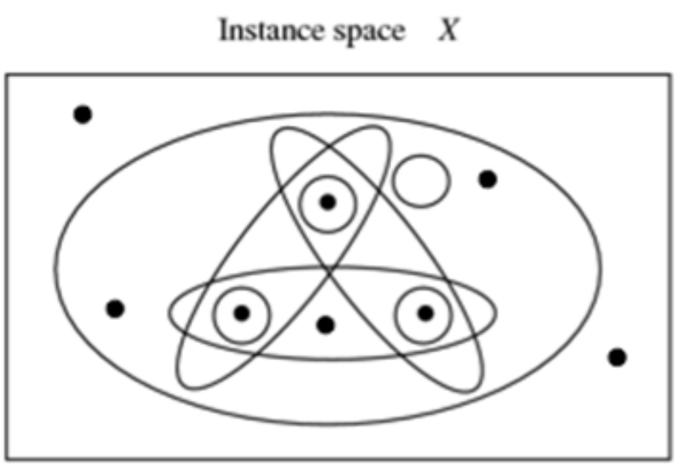
Definition : a set of instances *S* is **shattered** by hypothesis space *H* if and only if for every dichotomy of *S* there exists some hypothesis in *H* consistent with this dichotomy



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 $VC(H) \equiv \infty$



$VC(H) \equiv 3$

Sample Complexity based on VC dimension

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

ie., to guarantee that any hypothesis that perfectly fits the training data is probably (1- δ) approximately (ε) correct

$$m \geq 1/\varepsilon\ (4 \log_2(2/\ \delta)) + 8VC(H) \log_2(13/\varepsilon))$$

compare to our earlier result base on $|H|$:

$$m \geq 1/\varepsilon\ (\ln(1/\ \delta) + \ln|H|)$$

VC dimension : examples

Consider X =<, want to learn c:X→{0,1}

What is VC dimension of

- Open intervals :

  H1 : if $x > a$ then $y = 1$ else $y = 0$

  H2 : if $x > a$ then $y = 1$ else $y = 0$

  or, if $x > a$ then $y = 0$ else $y = 1$

- Closed intervals :

  H3 : if $a < x < b$ then $y = 1$ else $y = 0$

  H4 : if $a < x < b$ then $y = 1$ else $y = 0$

  or, if $a < x < b$ then $y = 0$ else $y = 1$

VC dimension : examples

$X = IR^2$

What is VC dimension of lines in a plane ?

- H2 = $\{((w_0 + w_1x_{1\,+}\, w_2x_2)>0 \rightarrow y=1)\}$

VC dimension : examples

What is VC dimension of

- H2 = { (($w_0 + w_1x_1 + w_2x_2$)>0 $\rightarrow$ y=1) }
  - VC(H_2)=3
- For H_n = linear separating hyperplanes in n dimensions, VC(Hn)=n+1

For any finite hypothesis space H, can you give an upper bound on VC(H) in terms of |H| ?

(hint:yes)

Tightness of Bounds on Sample Complexity

How many examples m suffice to assure that any hypothesis that fits the training data perfectly is probably (1-δ) approximately (ε) correct ?

$$m \geq 1/\varepsilon\,(4 \log_2(2/\delta)) + 8\,VC(H) \log_2(13/\varepsilon))$$

How tight is this bound ?

Lower bound on sample complexity (Ehrenfeucht et al., 1989) :

Consider any class C of concept such that VC(C) > 1, any learner L, any $0 < \varepsilon < 1/8$ and any $0 < \delta < 0.01$. Then there exists a distribution D and a target concept in C, such that if L observes fewer examples than

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

Then with probability at least δ, L outputs a hypothesis with $error_D(h) > \varepsilon$

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

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 and best case?

Optimal Mistakes Bounds

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

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

Optimal Mistake Bounds

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 $M_A(C)$

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

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

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

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

PAC Learning : What You Should Know

- PAC learning : Probably (1-δ) approximately (error ε) Correct.
- Problem setting

- Finite H, perfectly consistent learner result
- If target function is not in H, agnostic learning
- If $|H|=\infty$, use VC dimension to characterize H

- Most important :
  - Sample complexity grows with complexity of H
  - Quantitative characterization of overfitting

- Mistake Bounds