Machine Learning 10-601

Tom M. Mitchell
Machine Learning Department
Carnegie Mellon University

October 30, 2017

Today:

- Computational Learning Theory
- Probably Approximately Coorrect (PAC) learning theorem
- Vapnik-Chervonenkis (VC) dimension

Recommended reading:

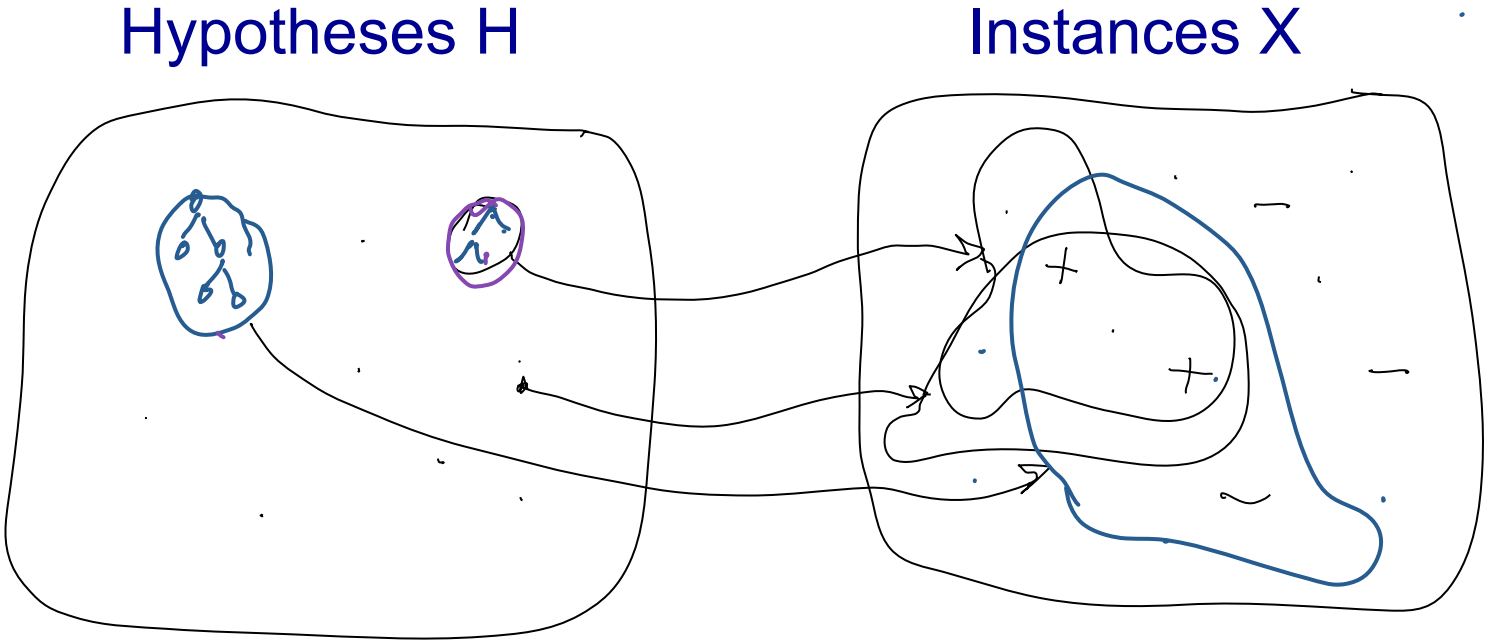
- Prof. Balcan notes: see Piazza syllabus
- Mitchell Ch. 7

Computational Learning Theory

- What general laws constrain inductive learning?
- Want theory to relate
  - Number of training examples
  - Complexity of hypothesis space
  - Accuracy to which target function is approximated
  - Manner in which training examples are presented
  - Probability of successful learning

* See annual Conference on Computational Learning Theory

Function Approximation: The Big Picture



Sample Complexity

How many training examples suffice to learn target concept

1. If learner proposes instances as queries to teacher?
   - learner proposes x, teacher provides f(x)

2. If teacher (who knows f(x)) generates training examples?
   - teacher proposes sequence {$<x^1, f(x^1)>$, … $<x^n, f(x^n)>$

3. If some random process (e.g., nature) generates instances, and teacher labels them?
   - instances drawn according to $P(X)$

1. If learner proposes instances as queries to teacher?
   - learner proposes x,teacher provides f(x)

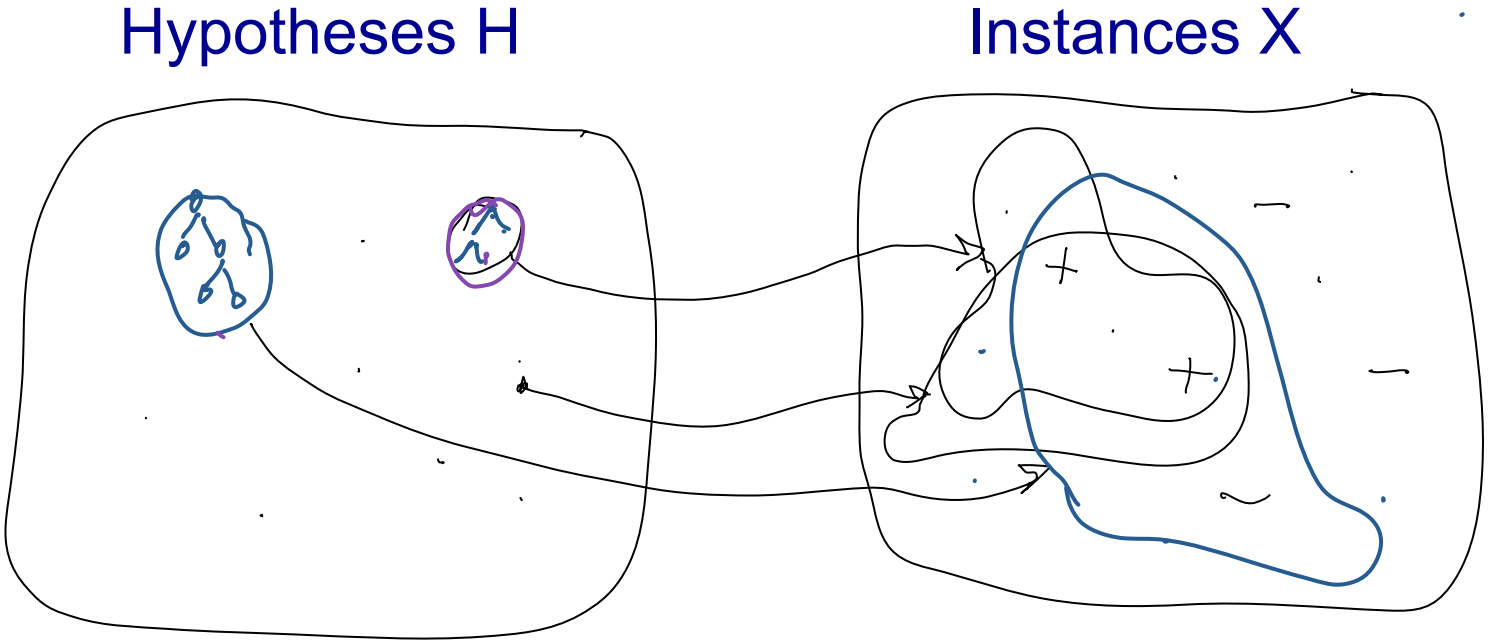
Example:

X=points <x1,x2> in the plane

H=rectangles, classify x positive if inside rectangle

h= if a<x1<b and c<x2<d then Y=1, else Y=0

Function Approximation: The Big Picture



1. If learner proposes instances as queries to teacher?
   - learner proposes x,teacher provides f(x)

Best case: learner plays 20 questions: chooses each x so that half the remaining hypotheses label it positive, half neg.

→ learn in $\log_2 |H|$ queries in best case

Sample Complexity 3

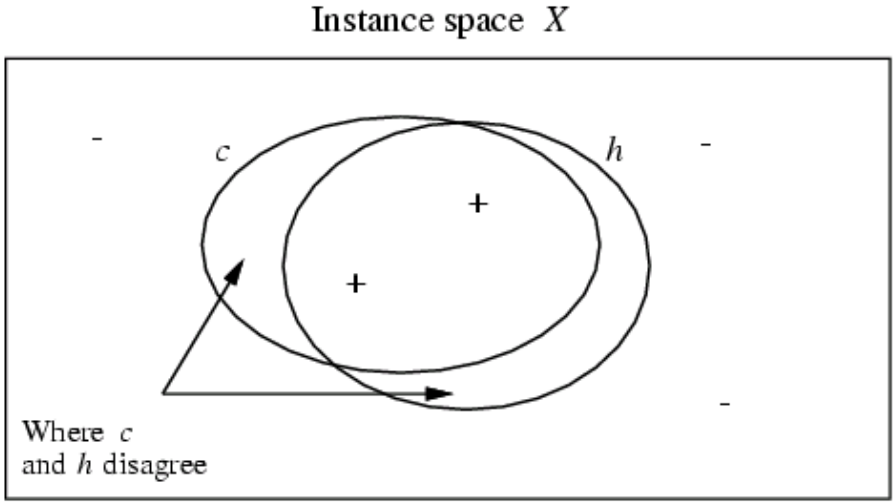
Problem setting:

- Set of instances X
- Set of hypotheses $H = \{h : X \rightarrow \{0,1\}\}$
- Set of possible target functions $C = \{c : X \rightarrow \{0,1\}\}$
- Sequence of training instances drawn at random from $P(X)$ teacher provides noise-free label $c(x)$

Learner outputs a hypothesis $h \in H$ such that

$$h = \arg\min_{h \in H} \; error_{train}(h)$$

True Error of a Hypothesis



The *true error* of h is the probability that it will misclassify an example drawn at random from $P(X)$

$$error_{true}(h) \equiv \Pr_{x \sim P(X)}[h(x) \neq c(x)]$$

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 D

$$error_{train}(h) \equiv \Pr_{x \in D}[h(x) \neq c(x)] = \frac{1}{|D|} \sum_{x \in D} \delta(h(x) \neq c(x))$$

training examples D

True error of hypothesis h with respect to c

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

$$error_{true}(h) \equiv \Pr_{x \sim P(X)}[h(x) \neq c(x)]$$

Probability distribution P(X)

Overfitting

Consider a hypothesis h and its

- Error rate over training data: $error_{train}(h)$
- True error rate over all data: $error_{true}(h)$

We say h overfits the training data if

$$error_{true}(h) > error_{train}(h)$$

Amount of overfitting =

$$error_{true}(h) - error_{train}(h)$$

Overfitting

Consider a hypothesis h and its

- Error rate over training data: $error_{train}(h)$
- True error rate over all data: $error_{true}(h)$

We say h overfits the training data if

$$error_{true}(h) > error_{train}(h)$$

Amount of overfitting =

$$error_{true}(h) - error_{train}(h)$$

Can we bound $error_{true}(h)$ in terms of $error_{train}(h)$??

$$error_{train} \equiv \Pr_{x \in D}[h(x) \neq c(x)] = \frac{1}{|D|} \sum_{x \in D} \frac{\delta(h(x) \neq c(x))}{|D|}$$

training examples

$$error_{true}(h) \equiv \Pr_{x \sim P(X)}[h(x) \neq c(x)]$$

Probability distribution P(x)

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

$$error_{\mathrm{D}}(h) \pm 1.96 \sqrt{\frac{error_{\mathrm{D}}(h)\,(1 - error_{\mathrm{D}}(h))}{n}}$$

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

Version Spaces

c: X → {0,1}

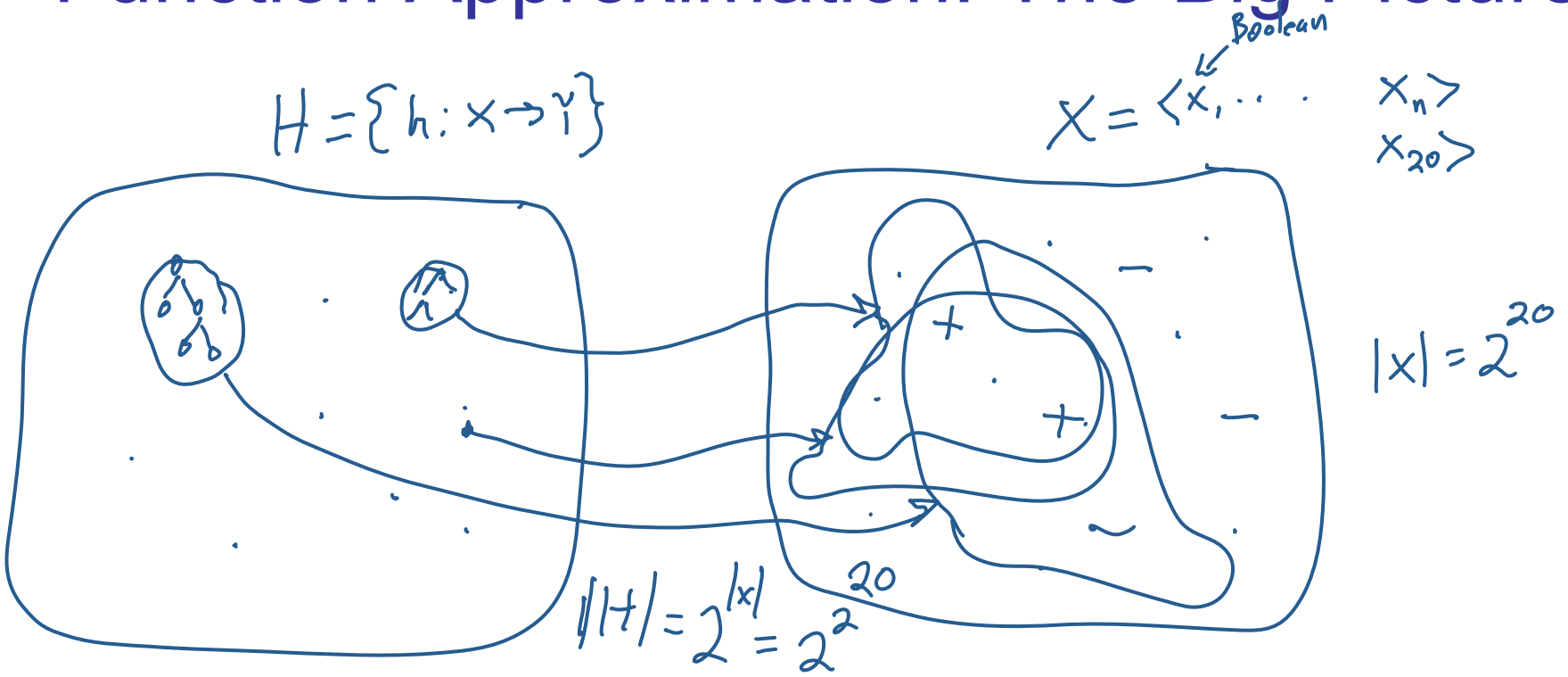
A hypothesis h is **consistent** with a set of training examples D of target concept c if and only if $h(x) = c(x)$ for each training example $\langle x, c(x)\rangle$ in D.

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

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.

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

Function Approximation: The Big Picture



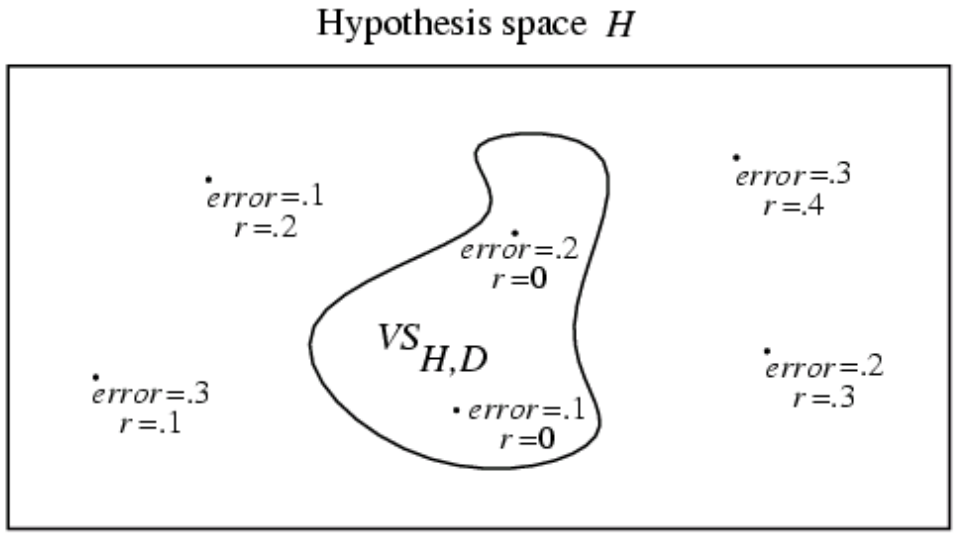
How many labeled examples are needed in order to determine which of the $2^{2^{20}}$ hypotheses is the correct one?

All 2^{20} instances in X must be labeled!

There is no free lunch!

Inductive inference - generalizing beyond the training data is impossible unless we add more assumptions (eg. priors over H)

Exhausting the Version Space



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

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

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

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

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

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

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

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

<u>Any(!)</u> learner that outputs a hypothesis consistent with all training examples (i.e., an h contained in $VS_{H,D}$)

What it means

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

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

Suppose we want this probability to be at most δ

1. How many training examples suffice?

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

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

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

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).

E.g.,

X=< X1, X2, … Xn >

Each h ∈ H constrains each Xi to be 1, 0, or "don't care"

In other words, each h is a rule such as:

If X2=0 and X5=1

Then Y=1, else Y=0

Example: H is Conjunction of up to N Boolean Literals

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

Consider classification problem f:X→Y:

- instances: $X = <X_1\ X_2\ X_3\ X_4>$ where each X_i is boolean
- Each hypothesis in H is a rule of the form:
  - IF $<X_1\ X_2\ X_3\ X_4>$ = $<0,?,1,?>$, THEN Y=1, ELSE Y=0
  - i.e., rules constrain the values of any subset of the X_i

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

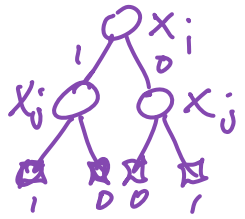
hint: ln(3)=1.1, ln(100)=4.6

Example: Depth 2 Decision Trees

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

Consider classification problem f:X→Y:

- instances: $X = \langle X_1 \ldots X_N \rangle$ 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* learner that outputs a consistent depth 2 decision tree will have true error at most 0.05?

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 $size(c)$.

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 $size(c)$.

Sufficient condition:

Holds if learner L requires only a polynomial number of training examples, and processing per example is polynomial

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?

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

Here ε is the difference between the training error and true error of the output hypothesis (this holds for all h in H)

Additive Hoeffding Bounds – Agnostic Learning

- Given *m* independent flips of a coin with true Pr(heads) = θ
  we can bound the error ϵ of the maximum likelihood estimate $\hat{\theta}$

$$\Pr[\theta > \hat{\theta} + \epsilon] \leq e^{-2m\epsilon^2}$$

- Relevance to agnostic learning: for any *single* hypothesis *h*

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

- But we must consider all hypotheses in H

$$\Pr[(\exists h \in H) error_{true}(h) > error_{train}(h) + \epsilon] \leq |H| e^{-2m\epsilon^2}$$

- So, with probability at least (1-δ) *every* h satisfies

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

General Hoeffding Bounds

- When estimating parameter θ inside [a,b] from *m* examples

$$P(|\hat{\theta} - E[\hat{\theta}]| > \epsilon) \leq 2e^{\frac{-2m\epsilon^2}{(b-a)^2}}$$

- When estimating a probability θ is inside [0,1], so

$$P(|\hat{\theta} - E[\hat{\theta}]| > \epsilon) \leq 2e^{-2m\epsilon^2}$$

- And if we're interested in only one-sided error, then

$$P((\hat{\theta} - E[\hat{\theta}]) > \epsilon) \leq e^{-2m\epsilon^2}$$

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

Here ε is the difference between the training error and true error of the output hypothesis (this holds for all h in H)

But, the output h with lowest training error might not give us the h^* with lowest true error. How far can true error of h be from h^* ?

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

Here ε is the difference between the training error and true error of the output hypothesis (this holds for all h in H)

But, the output h with lowest training error might not give us the h^* with lowest true error. How far can true error of h be from h^* ?

$$error_{true}(h) \leq error_{true}(h^*) + 2\epsilon$$

best training error hypothesis

best true error hypothesis

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

Question: If H = {h | h: X → Y} is infinite, what measure of complexity should we use in place of |H| ?

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

Question: If H = {h | h: X → Y} is infinite, 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 how it is labeled)

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

Question: If H = {h | h: X → Y} is infinite, what measure of complexity should we use in place of |H| ?

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

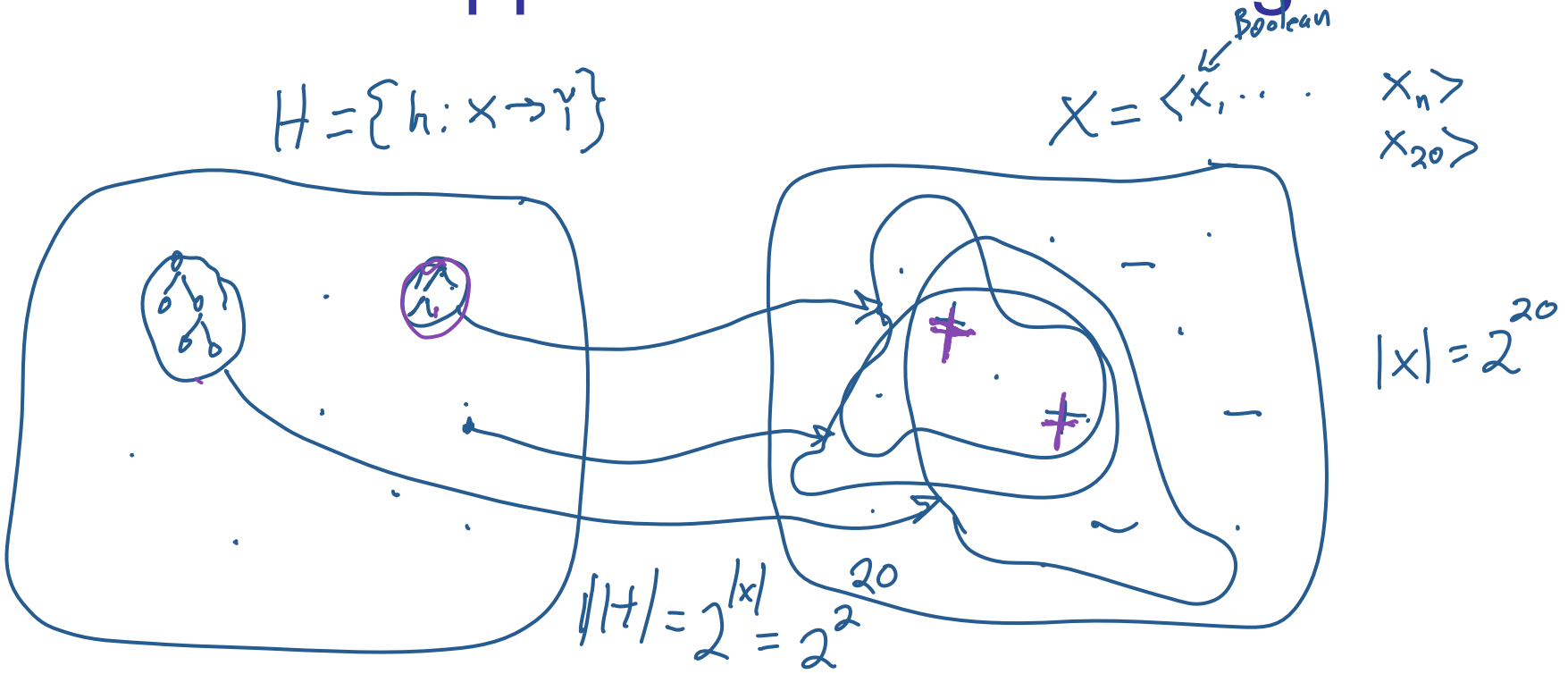
this is the VC dimension of H

Question: If H = {h | h: X → Y} is infinite, 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)

Informal intuition:

Function Approximation: The Big Picture



How many labeled examples are needed in order to determine which of the $2^{2^{20}}$ hypotheses is the correct one?

All 2^{20} instances in X must be labeled!

There is no free lunch!

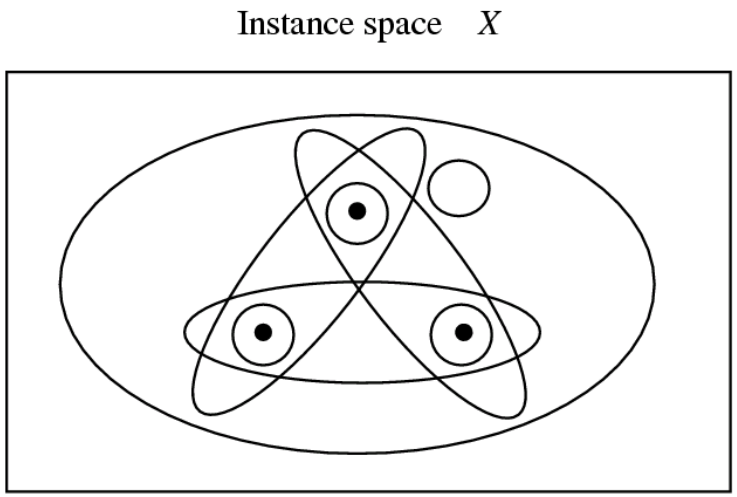
Inductive inference - generalizing beyond the training data is impossible unless we add more assumptions (eg. priors over H)

Shattering a Set of Instances

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

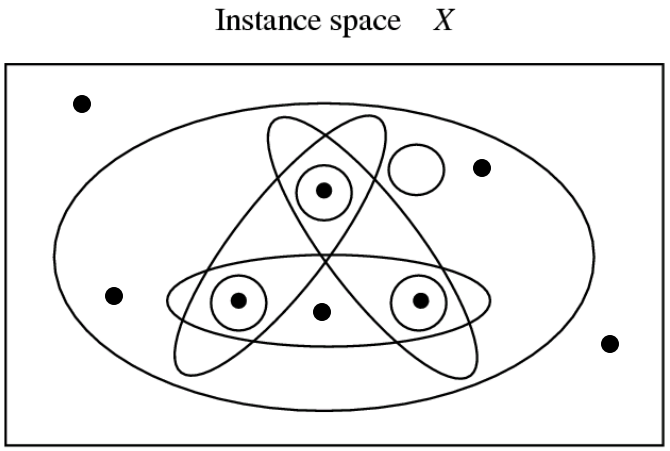
a labeling of each member of S as positive or negative

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)=3

Sample Complexity based on VC dimension

How many randomly drawn examples 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

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

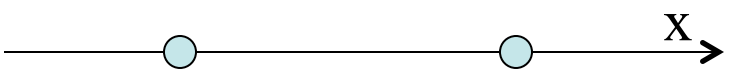
Compare to our earlier results based on |*H*|:

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

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:

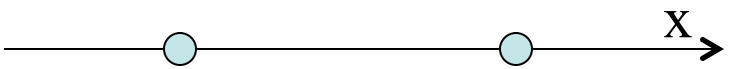
  H3: if $a < x < b$ then $y = 1$ else $y = 0$

  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

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

What is VC dimension of



- Open intervals:

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

  H2: if $x > a$ then $y = 1$ else $y = 0$ VC(H2)=2
  or, if $x > a$ then $y = 0$ else $y = 1$

- Closed intervals:

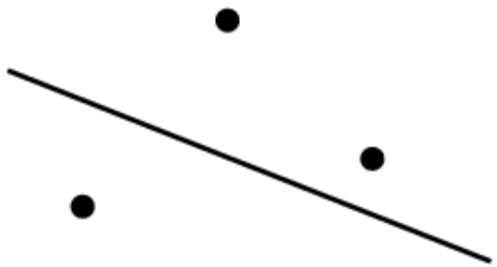
  H3: if $a < x < b$ then $y = 1$ else $y = 0$ VC(H3)=2

  H4: if $a < x < b$ then $y = 1$ else $y = 0$ VC(H4)=3
  or, if $a < x < b$ then $y = 0$ else $y = 1$

VC dimension: examples

What is VC dimension of lines in a plane?

- $H_2 = \{ ((w_0 + w_1 x_1 + w_2 x_2) > 0 \rightarrow y=1) \}$



VC dimension: examples

What is VC dimension of

- $H_2 = \{ ((w_0 + w_1 x_1 + w_2 x_2) > 0 \rightarrow y=1) \}$

  $VC(H_2)=3$

- For H_n = linear separating hyperplanes in n dimensions, $VC(H_n)=n+1$

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

More VC Dimension Examples to Think About

- Logistic regression over n continuous features
  - Over n boolean features?

- Decision trees defined over n boolean features

  F: $<X_1, \ldots X_n> \rightarrow Y$

- Decision trees of depth 2 defined over n features

- Naïve Bayes defined over n boolean features

- How about 1-nearest neighbor?

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?

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

How tight is this bound?

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?

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

How tight is this bound?

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

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

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

Then with probability at least δ, L outputs a hypothesis with $error_{\mathcal{D}}(h) > \epsilon$

Shatter coefficient H[m]

for $S \subseteq X$, where $S = \{x_1 \ldots x_m\}$, define $H(S)$ as the set of distinct labelings of S induced by H

$$H(S) \equiv \{\langle h(x_1) \ldots, h(x_m)\rangle \mid h \in H)\}$$

and define $H[m]$ as the maximum number of ways to label m instances of X

$$H[m] \equiv \max_{S \subseteq X, |S|=m} |H(S)|$$

If H can shatter a subset of size m, then $H[m] =$

Note $VCdim(H) \equiv$ largest m for which $H[m] = 2^m$

Shatter coefficient H[m]

Sauer's Lemma: Let $VCdim(H) = d$. Then

1. for all m, $H[m] \leq \Phi_d(m)$, where $\Phi_d(m) \equiv \sum_{i=0}^{d} \binom{m}{i}$
2. for $m > d$,

$$\Phi_d(m) \leq (1+m)^d$$

$$\Phi_d(m) \leq \left(\frac{em}{d}\right)^d$$

Sample Complexity - Summary

How many randomly drawn examples 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

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

|H|

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

VC(H)

$$m > \frac{2}{\epsilon}(\log_2(1/\delta) + \log_2(3\ H[2m]))$$

H[m]

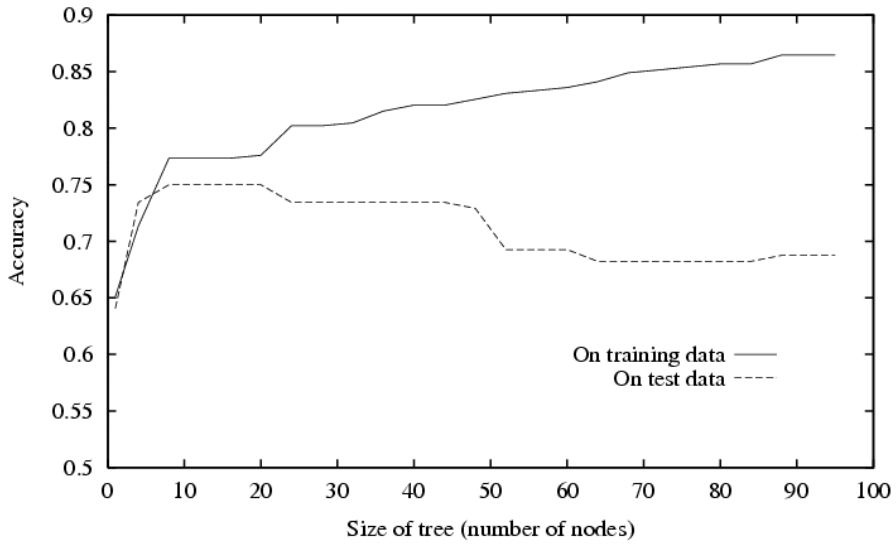
* also Rademacher complexity

Agnostic Learning: VC Bounds

[Schölkopf and Smola, 2002]

With probability at least (1-δ) every $h \in H$ satisfies

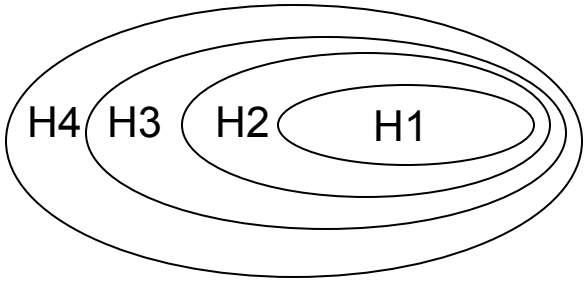
$$error_{true}(h) < error_{train}(h) + \sqrt{\frac{VC(H)(\ln \frac{2m}{VC(H)} + 1) + \ln \frac{4}{\delta}}{m}}$$



Structural Risk Minimization [Vapnik]

Which hypothesis space should we choose?

- Bias / variance tradeoff



SRM: choose H to minimize bound on expected true error!

$$error_{true}(h) < error_{train}(h) + \sqrt{\frac{VC(H)(\ln \frac{2m}{VC(H)} + 1) + \ln \frac{4}{\delta}}{m}}$$

* unfortunately a somewhat loose bound...

Rademacher Complexity

Key idea: complexity of H is its ability to fit noise labels.

Advantages:

- applies to real-valued functions (e.g., regression)
- is sensitive to P(X), and particular training set
- gives tighter bounds than VC dimension
- widely used in modern learning theory

Rademacher Complexity Setting

Learn $f : X \to Y$, where $Y \in \{-1, +1\}$

Note:

if $h(x) = y$, then $yh(x) = 1$

if $h(x) \neq y$, then $yh(x) = -1$

so error of h on sample $S = \{\langle x_1, y_1 \rangle, \ldots, \langle x_m, y_m \rangle\}$ is :

$$error_S(h) = \frac{1}{m} \sum_{i=1}^{m} \delta(h(x_i) \neq y_i) = \frac{1}{m} \sum_{i=1}^{m} \frac{1 - y_i h(x_i)}{2}$$

and the hypothesis h with the lowest $error_S(h)$ is

$$\arg\max_{h \in H} \frac{1}{m} \sum_{i=1}^{m} y_i h(x_i)$$

Rademacher complexity

Given data sample $S = \{\langle x_1, y_1\rangle, \ldots, \langle x_m, y_m\rangle\}$
define corresponding set of random labels $\{\sigma_1, \ldots, \sigma_m\}$
where $\sigma_i \in \{-1, 1\}$, $P(\sigma_i = -1) = 0.5 = P(\sigma_i = 1)$.
Note the hypothesis h that best fits these random labels is

$$\arg\max_{h \in H} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(x_i)$$

Define *empirical Rademacher complexity* $\hat{R}_S(H)$ with respect to S:

$$\hat{R}_S(H) \equiv E_\sigma \left[\max_{h \in H} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(x_i) \right]$$

Rademacher complexity

Given data sample $S = \{\langle x_1, y_1 \rangle, \ldots, \langle x_m, y_m \rangle\}$
define corresponding set of random labels $\{\sigma_1, \ldots, \sigma_m\}$
where $\sigma_i \in \{-1, 1\}$, $P(\sigma_i = -1) = 0.5 = P(\sigma_i = 1)$.
Note the hypothesis h that best fits these random labels is

$$\arg\max_{h \in H} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(x_i)$$

Define *empirical Rademacher complexity* $\hat{R}_S(H)$ with respect to S:

$$\hat{R}_S(H) \equiv E_\sigma \left[\max_{h \in H} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(x_i) \right]$$

then in the agnostic PAC learning setting, with probability $(1 - \delta)$:

$$error_{true}(h) \leq error_{train}(h) + \hat{R}_{train}(H) + 3\sqrt{\frac{\log(2/\delta)}{m}}$$

Rademacher complexity

$$\hat{R}_S(H) \equiv E_\sigma \left[\max_{h \in H} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(x_i) \right]$$

What is $\hat{R}_S(H)$ when:

$H = \{h_1\}$ has only one hypothesis?

H can shatter the training set S?

Empirical Rademacher Complexity

$$\hat{R}_S(H) \equiv E_\sigma \left[\max_{h \in H} \frac{1}{m} \sum_{i=1}^{m} \sigma_i h(x_i) \right]$$

Also define full *Rademacher complexity*

$$R_m(H) \equiv E_{S \text{ of size } m}[\hat{R}_S(H)]$$

Rademacher complexity:

- can be applied to real-valued functions (e.g., regression)
- is sensitive to P(X), and the particular training set
- can give tighter bounds than VC dimension

With probability $\geq (1-\delta)$, $(error_{true} - error_{train}) \leq \epsilon$

(1) for all $h \in H$ such that $error_{train} = 0$,

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

finite H

(2) for all $h \in H$

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

finite H

(3) for all $h \in H$

$$\epsilon = 8\sqrt{\frac{VC(H)(\ln \frac{m}{VC(H)} + 1) + \ln(8/\delta)}{2m}}$$

infinite H

(4) for all $h \in H$

$$\epsilon = \hat{R}_{train}(H) + 3\sqrt{\frac{\log(2/\delta)}{m}}$$

infinite H

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

$x =< x_1, \ldots, x_n >,\ x_i \in \{0, 1\}$

Consider Find-S when H = conjunction of boolean literals

FIND-S:

- Initialize h to the most specific hypothesis $x_1 \wedge \neg x_1 \wedge x_2 \wedge \neg x_2 \wedge \ldots \neg x_n \rightarrow y = 1$ else $y = 0$
- 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 ← H
2. For each training example,
   - remove from VS every hypothesis that misclassifies 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 $M_A(C)$.

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

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

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

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

Perceptron Algorithm

Perceptron Algorithm: learn $\hat{y} = h(x) = sign(\vec{w} \cdot \vec{x})$, where $\vec{x} =< 1, x_1, \ldots, x_n >$, $\vec{w} =< w_0, w_1 \ldots, w_n >$, $y \in \{-1, +1\}$

Input: $\{< \vec{x}_1, y_1 > \ldots < \vec{x}_m, y_m >\}$

Initialize $\vec{w} = 0$;

repeat

- for $i = 1$ to m
  - **if** $y_i \neq sign(\vec{w} \cdot \vec{x}_i)$ **then** $\vec{w} \leftarrow \vec{w} + y_i \vec{x}_i$;

until converged

Mistake Bounds for Perceptron

When data is linearly separable:

THEOREM 1 (BLOCK, NOVIKOFF) *Let* $\langle (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m) \rangle$ *be a sequence of labeled examples with* $||\mathbf{x}_i|| \leq R$. *Suppose that there exists a vector* $\mathbf{u}$ *such that* $||\mathbf{u}|| = 1$ *and* $y_i(\mathbf{u} \cdot \mathbf{x}_i) \geq \gamma$ *for all examples in the sequence. Then the number of mistakes made by the online perceptron algorithm on this sequence is at most* $(R/\gamma)^2$.

Mistake Bounds for Perceptron

When data is linearly separable:

THEOREM 1 (BLOCK, NOVIKOFF) *Let* $\langle(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\rangle$ *be a sequence of labeled examples with* $||\mathbf{x}_i|| \leq R$. *Suppose that there exists a vector* $\mathbf{u}$ *such that* $||\mathbf{u}|| = 1$ *and* $y_i(\mathbf{u} \cdot \mathbf{x}_i) \geq \gamma$ *for all examples in the sequence. Then the number of mistakes made by the online perceptron algorithm on this sequence is at most* $(R/\gamma)^2$.

When not linearly separable: [Freund & Schapire]

THEOREM 2 *Let* $\langle(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\rangle$ *be a sequence of labeled examples with* $||\mathbf{x}_i|| \leq R$. *Let* $\mathbf{u}$ *be any vector with* $||\mathbf{u}|| = 1$ *and let* $\gamma > 0$. *Define the deviation of each example as*

$$d_i = \max\{0, \gamma - y_i(\mathbf{u} \cdot \mathbf{x}_i)\} ,$$

and define $D = \sqrt{\sum_{i=1}^{m} d_i^2}$. *Then the number of mistakes of the online perceptron algorithm on this sequence is bounded by*

$$\left(\frac{R + D}{\gamma}\right)^2 .$$

What You Should Know

- Sample complexity varies with the learning setting
  - Learner actively queries trainer
  - Examples arrive at random
  - …

- Within the PAC learning setting, we can bound the probability that learner will output hypothesis with given error
  - For ANY consistent learner (case where $c \in H$)
  - For ANY "best fit" hypothesis (agnostic learning, where perhaps c not in H)

- VC dimension as measure of complexity of H
- H[m] shatter coefficient (a.k.a. growth function) measure
- Rademacher complexity measure

- Mistake bounds

- Conference on Learning Theory: http://www.learningtheory.org
- ML Department course on Machine Learning Theory