Learning theory: generalization and VC dimension

Yifeng Tao

School of Computer Science

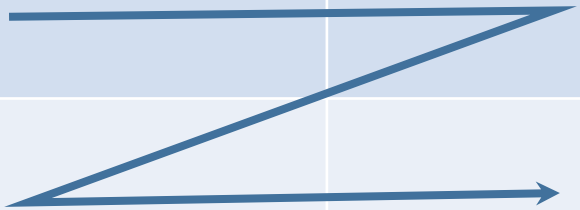
Carnegie Mellon University

Slides adapted from Eric Xing

Outline

- Computational learning theories
  - PAC framework
  - Agnostic framework
- VC dimension

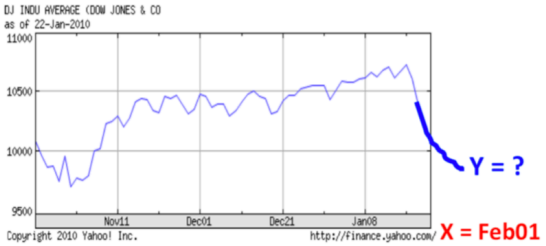
	PAC	Agnostic
|H| finite		
|H| infinite, but VC(H) finite		



Generalizability of Learning

- In machine learning it's really **generalization error** that we care, but most learning algorithms fit their models to the training set.
- Why should doing well on the training set tell us anything about generalization error? Specifically, can we relate **error on training set** to generalization error?
- Are there conditions under which we can actually prove that learning algorithms will work well?
- Lecture 1:

Sports
Science
News



Classification:

$$R(f) = P(f(X) \neq Y)$$

Probability of Error

minimizing empirical errors:

$$\sum_{i=1}^{n} I(f(X_i) \neq Y_i)/n$$

Regression:

$$R(f) = \mathbb{E}[(f(X) - Y)^2]$$

Mean Squared Error

$$\sum_{i=1}^{n} (f(X_i) - Y_i)^2/n$$

[Slide from Eric Xing]

What General Laws Constrain Inductive Learning?

- Want theory to relate:
  - Training examples: m
  - Complexity of hypothesis/concept space: H
  - Accuracy of approximation to target concept: ε
  - Probability of successful learning: δ
- All the results in O(…)

[Slide from Eric Xing]

Prototypical concept learning task

- Binary classification
  - Everything we'll say here generalizes to other, including regression and multi-class classification problems.
- Given:
  - Instances *X*: Possible days, each described by the attributes *Sky, AirTemp, Humidity, Wind, Water, Forecast*
  - Target function *c: EnjoySport* : X → {0, 1}
  - Hypotheses space *H*: Conjunctions of literals. E.g.
    - *(?, Cold, High, ?, ?, ¬EnjoySport)*.
  - Training examples *S*: iid positive and negative examples of the target function
    - *(x1, c(x1)), ... (xm, c(xm))*
- Determine:
  - A hypothesis *h* in *H* such that *h(x)* is "good" w.r.t *c(x)* for all *x* in *S*?
  - A hypothesis *h* in *H* such that *h(x)* is "good" w.r.t *c(x)* for all *x* in the true distribution D?

[Slide from Eric Xing]

Sample Complexity

- How many training examples *m* are sufficient to learn the target concept?
- Training scenarios:
  - If learner proposes instances, as queries to teacher
    - Learner proposes instance x, teacher provides *c(x)*
  - If teacher (who knows *c*) provides training examples
    - Teacher provides sequence of examples offer m(*x*,*c(x)*)
  - If some random process (e.g., nature) proposes instances
    - Instance *x* generated randomly, teacher provides *c(x)*

[Slide from Eric Xing]

Two Basic Competing Models

PAC framework	Agnostic framework
Sample labels are consistent with some h in H	*No prior restriction on the sample labels*
Learner's hypothesis required to meet *absolute* upper bound on its error	*The required upper bound on the hypothesis error is only* relative *(to the best hypothesis in the class)*

[Slide from Eric Xing]

Protocol

Given:

- set of examples X
- fixed (unknown) distribution D over X
- set of hypotheses H
- set of possible target concepts C

Learner observes sample $S = \{ \langle x_i, c(x_i) \rangle \}$

- instances x_i drawn from distr. D
- labeled by target concept $c \in C$

(Learner does NOT know $c(.)$, D)

Learner outputs $h \in H$ estimating c

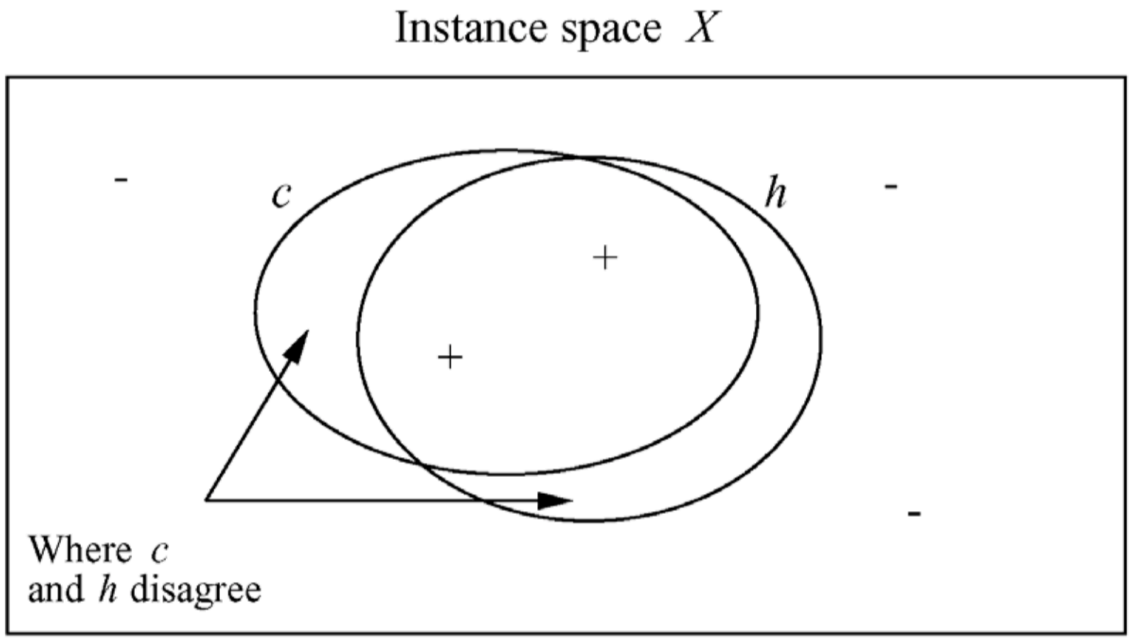
- h is evaluated by performance on subsequent instances drawn from D

For now:

- C = H (so $c \in H$)
- Noise-free data

[Slide from Eric Xing]

True error of a hypothesis



- Definition: The *true error* (denoted $\varepsilon_D(h)$) of 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.

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

[Slide from Eric Xing]

Two notions of error

- *Training error* (a.k.a., empirical risk or empirical error) of hypothesis *h* with respect to target concept c
  - How often *h(x) ≠ c(x)* over training instance from S

$$\hat{\epsilon}_{\mathcal{S}}(h) \equiv Pr_{x \in \mathcal{S}}[c(x) \neq h(x)] \equiv \frac{\sum_{x \in \mathcal{S}} \delta(c(x) \neq h(x))}{|\mathcal{S}|}$$

- *True error* of (a.k.a., generalization error, test error) hypothesis *h* with respect to *c*
  - How often *h(x) ≠ c(x)* over future random instances drew iid from D

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

Can we bound

$\hat{\epsilon}_{\mathcal{S}}(h)$

in terms of

$\epsilon_{\mathcal{D}}(h)$

??

[Slide from Eric Xing]

The Union Bound

Lemma. (The union bound). Let A_1;A_2, … , A_k be k different events (that may not be independent). Then

$$P(A_1 \cup A_2 \cup \ldots \cup A_k) \leq P(A_1) + P(A_2) + \ldots + P(A_k)$$

- In probability theory, the union bound is usually stated as an axiom (and thus we won't try to prove it), but it also makes intuitive sense: The probability of any one of k events happening is at most the sums of the probabilities of the k different events.

[Slide from Eric Xing]

Hoeffding inequality

Lemma. (Hoeding inequality) Let $Z_1, \ldots, Z_m$ be m independent and identically distributed (iid) random variables drawn from a Bernoulli(ϕ) distribution, i.e., $P(Z_i = 1) = \phi$, and $P(Z_i = 0) = 1 - \phi$.

Let $\hat{\phi} = (1/m) \sum_{i=1}^{m} Z_i$ be the mean of these random variables, and let any $\gamma > 0$ be fixed. Then

$$P(|\phi - \hat{\phi}| > \gamma) \leq 2 \exp(-2\gamma^2 m)$$

- This lemma (which in learning theory is also called the Chernoff bound) says that if we take $\hat{\phi}$ — the average of m Bernoulli(ϕ) random variables — to be our estimate of ϕ , then the probability of our being far from the true value is small, so long as m is large.

[Slide from Eric Xing]

Version Space

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

$$Consistent(h, \mathcal{S}) \models \quad h(x) = c(x), \forall \langle x, c(x) \rangle \in \mathcal{S}$$

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

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

[Slide from Eric Xing]

Consistent Learner

- A learner is ***consistent*** if it outputs hypothesis that perfectly fits the training data
  - This is a quite reasonable learning strategy
- Every consistent learning outputs a hypothesis belonging to the version space
- We want to know how such hypothesis generalizes

[Slide from Eric Xing]

Probably Approximately Correct

Goal:

PAC-Learner produces hypothesis $\hat{h}$ that

is approximately correct,

$\text{err}_D(\hat{h}) \approx 0$

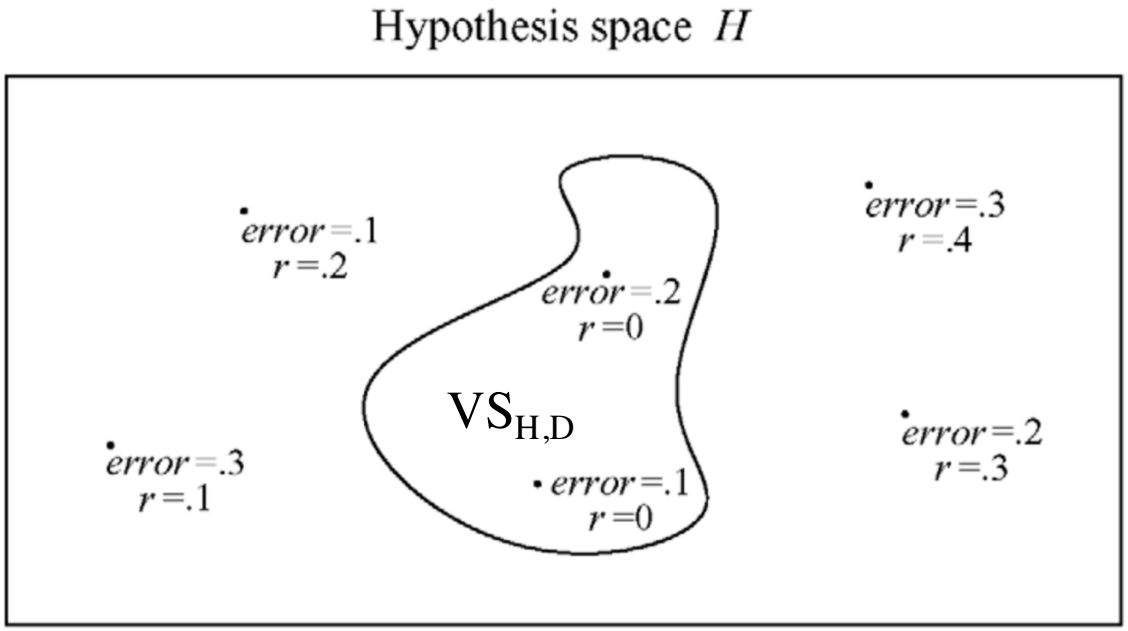
with high probability

$P(\ \text{err}_D(\hat{h}) \approx 0\ ) \approx 1$

- Double “hedging"
  - Approximately
  - Probably
- Need both!

[Slide from Eric Xing]

Exhausting the version space



$(r = \text{training error}, error = \text{true error})$

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

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

[Slide from Eric Xing]

How many examples will ε-exhaust the VS

Theorem: [Haussler, 1988].

- If the hypothesis space *H* is finite, and *S* is a sequence of m ≥ 1 independent random examples of some target concept *c*, then for any 0 ≤ ε ≤ 1/2, the probability that the version space with respect to *H* and *S* is **not** ε-exhausted (with respect to *c*) is less than

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

- This bounds the probability that any consistent learner will output a hypothesis *h* with ε(*h*) ≥ ε

[Slide from Eric Xing]

Proof

$$P(error_{true}(h_1) \geq \varepsilon \text{ OR } error_{true}(h_2) \geq \varepsilon \text{ OR } \ldots \text{ OR } error_{true}(h_{|H_c|}) \geq \varepsilon)$$

$$\leq \sum_k P(error_{true}(h_k) \geq \varepsilon) \quad \leftarrow \text{Union bound}$$

$$\leq \sum_k (1-\varepsilon)^m \quad \leftarrow \text{bound on individual } h_j s$$

$$\leq |H|(1-\varepsilon)^m \quad \leftarrow |H_c| \leq |H|$$

$$\leq |H| \, e^{-m\varepsilon} \quad \leftarrow (1-\varepsilon) \leq e^{-\varepsilon} \text{ for } 0 \leq \varepsilon \leq 1$$

[Slide from Eric Xing and David Sontag]

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

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

- How many training examples suffice?

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

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

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

[Slide from Eric Xing]

Learning Conjunctions of Boolean Literals

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

every h in $VS_{H,S}$ satisfies $\varepsilon_{\mathcal{D}}(h) \leq \varepsilon$

Use our theorem:

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

Suppose H contains conjunctions of constraints on up to n boolean attributes (i.e., n boolean literals).

Then $|H| = 3^n$, and

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

or

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

[Slide from Eric Xing]

PAC Learnability

- A learning algorithm is PAC learnable if it
  - Requires no more than polynomial computation per training example, and
  - no more than polynomial number of samples
- **Theorem**: conjunctions of Boolean literals is PAC learnable

[Slide from Eric Xing]

How about *EnjoySport*?

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

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

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

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

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

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

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

$$m \geq 98.8$$

[Slide from Eric Xing]

PAC-Learning

Learner L can draw labeled instance $\langle x, c(x) \rangle$ in unit time, $x \in X$ of length n drawn from distribution $\mathcal{D}$, labeled by target concept $c \in C$

Def'n: Learner L PAC-learns class C using hypothesis space H

if

1. for any target concept $c \in C$,
   any distribution $\mathcal{D}$, any ε such that $0 < \varepsilon < 1/2$, δ such that $0 < \delta < 1/2$,
   L returns $h \in H$ s.t.
   w/ prob. $\geq 1 - \delta$, $err_D(h) < \varepsilon$
2. L's run-time (and hence, sample complexity)
   is poly($|x|$, size(c), $1/\varepsilon$, $1/\delta$)

Sufficient:

1. Only poly(…) training instances – $|H| = 2^{poly()}$
2. Only poly time / instance …

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

[Slide from Eric Xing]

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?

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

derived from Hoeffding bounds:

$$\Pr[error_D(h) > error_S(h) + \varepsilon] \leq e^{-2m\varepsilon^2}$$

[Slide from Eric Xing]

Empirical Risk Minimization Paradigm

Choose a *Hypothesis Class* H of subsets of X.

For an input sample S, find some h in H that fits S "well".

For a new point x, predict a label according to its membership in h.

$$\hat{h} = \arg\min_{h \in H} \hat{\epsilon}_S(h)$$

Example:

- Consider linear classification, and let $h_\theta(x) = 1\{\theta^T x \geq 0\}$
  Then $H = \{h_\theta \ : \ h_\theta(x) = 1\{\theta^T x \geq 0\}, \ \theta \in R^{n+1}\}$

$$\hat{\theta} = \arg\min_{\theta} \hat{\epsilon}_S(h_\theta)$$

- We think of ERM as the most "basic" learning algorithm, and it will be this algorithm that we focus on in the remaining.
- In our study of learning theory, it will be useful to abstract away from the specific parameterization of hypotheses and from issues such as whether we're using a linear classier or an ANN

[Slide from Eric Xing]

The Case of Finite H

- H = $\{h_1, ..., h_k\}$ consisting of k hypotheses.
- We would like to give guarantees on the generalization error of $\hat{h}$.
- First, we will show that $\hat{\epsilon}(h)$ is a reliable estimate of $\varepsilon(h)$ for all h.
- Second, we will show that this implies an upper-bound on the generalization error of $\hat{h}$.

[Slide from Eric Xing]

Misclassification Probability

- The outcome of a binary classifier can be viewed as a Bernoulli random variable Z: $Z = 1\{h_i(x) \neq c(x)\}$

- For each sample: $Z_j = 1\{h_i(x_j) \neq c(x_j)\}$

$$\hat{\epsilon}(h_i) = \frac{1}{m}\sum_{j=1}^{m} Z_j$$

- Hoeffding inequality

$$P(|\epsilon(h_i) - \hat{\epsilon}(h_i)| > \gamma) \leq 2\exp(-2\gamma^2 m)$$

  - This shows that, for our particular h_i, training error will be close to generalization error with high probability, assuming m is large.

[Slide from Eric Xing]

Uniform Convergence

- But we don't just want to guarantee that $\hat{\epsilon}(h_i)$ will be close $\epsilon(h_i)$ (with high probability) for just only one particular h_i. We want to prove that this will be true for simultaneously for all $h_i \in H$
- For k hypothesis:

$$\begin{aligned} P(\exists h \in H,\ |\epsilon(h_i) - \hat{\epsilon}(h_i)| > \gamma) &= P(A_1 \cup \ldots \cup A_k) \\ &< \sum_{i=1}^{k} P(A_i) \\ &= \sum_{i=1}^{k} 2\exp(-2\gamma^2 m) \\ &= 2k\exp(-2\gamma^2 m) \end{aligned}$$

- This means:

$$\begin{aligned} P(\neg\exists h \in H,\ |\epsilon(h_i) - \hat{\epsilon}(h_i)| > \gamma) &= P(\forall h \in H,\ |\epsilon(h_i) - \hat{\epsilon}(h_i)| \leq \gamma) \\ &= 1 - 2k\exp(-2\gamma^2 m) \end{aligned}$$

[Slide from Eric Xing]

- In the discussion above, what we did was, for particular values of m and γ, given a bound on the probability that:

  for some $h_i \in H$

$$\left|\epsilon(h_i) - \hat{\epsilon}(h_i)\right| > \gamma$$

- There are three quantities of interest here: m and γ, and probability of error; we can bound either one in terms of the other two.

[Slide from Eric Xing]

Sample Complexity

- How many training examples we need in order make a guarantee?

$$P(\exists h \in H,\ |\epsilon(h) - \hat{\epsilon}(h)| > \gamma) \quad = \quad 2k \exp(-2\gamma^2 m)$$

- We find that if $m \geq \frac{1}{2\gamma^2} \log \frac{2k}{\delta}$

  then with probability at least 1-δ, we have that $|\epsilon(h_i) - \hat{\epsilon}(h_i)| \leq \gamma$ for all $h_i \in H$

- The key property of the bound above is that the number of training examples needed to make this guarantee is only logarithmic in k, the number of hypotheses in H. This will be important later.

[Slide from Eric Xing]

Generalization Error Bound

- Similarly, we can also hold m and δ fixed and solve for γ in the previous equation, and show [again, convince yourself that this is right!] that with probability 1- δ , we have that for all $h_i \in H$

$$|\hat{\epsilon}(h) - \epsilon(h)| \leq \sqrt{\frac{1}{m} \log \frac{2k}{\delta}}$$

- Define $h^* = \arg\min_{h \in H} \epsilon(h)$ to be the best possible hypothesis in H.

$$\begin{aligned} \epsilon(\hat{h}) &\leq \hat{\epsilon}(\hat{h}) + \gamma \\ &\leq \hat{\epsilon}(h^*) + \gamma \\ &\leq \epsilon(h^*) + 2\gamma \end{aligned}$$

  - If uniform convergence occurs, then the generalization error of $\hat{\epsilon}(h)$ is at most 2γ worse than the best possible hypothesis in H!

[Slide from Eric Xing]

Agnostic framework

Theorem. Let $|\mathcal{H}| = k$, and let any m, δ be fixed. Then with probability at least $1 - \delta$, we have that

$$\varepsilon(\hat{h}) \leq \left(\min_{h \in \mathcal{H}} \varepsilon(h) \right) + 2\sqrt{\frac{1}{2m} \log \frac{2k}{\delta}}.$$

Corollary. Let $|\mathcal{H}| = k$, and let any δ, γ be fixed. Then for $\varepsilon(\hat{h}) \leq \min_{h \in \mathcal{H}} \varepsilon(h) + 2\gamma$ to hold with probability at least $1 - \delta$, it suffices that

$$\begin{aligned} m &\geq \frac{1}{2\gamma^2} \log \frac{2k}{\delta} \\ &= O\left(\frac{1}{\gamma^2} \log \frac{k}{\delta} \right), \end{aligned}$$

[Slide from Eric Xing]

What if H is not finite?

- Can't use our result for infinite H

- Need some other measure of complexity for H
  - Vapnik-Chervonenkis (VC) dimension!

[Slide from Eric Xing]

How do we characterize “power”?

- Different machines have different amounts of “power”.
- Tradeoff between:
  - More power: Can model more complex classifiers but might overfit
  - Less power: Not going to overfit, but restricted in what it can model

- How do we characterize the amount of power?

[Slide from Eric Xing]

Shattering a Set of Instances

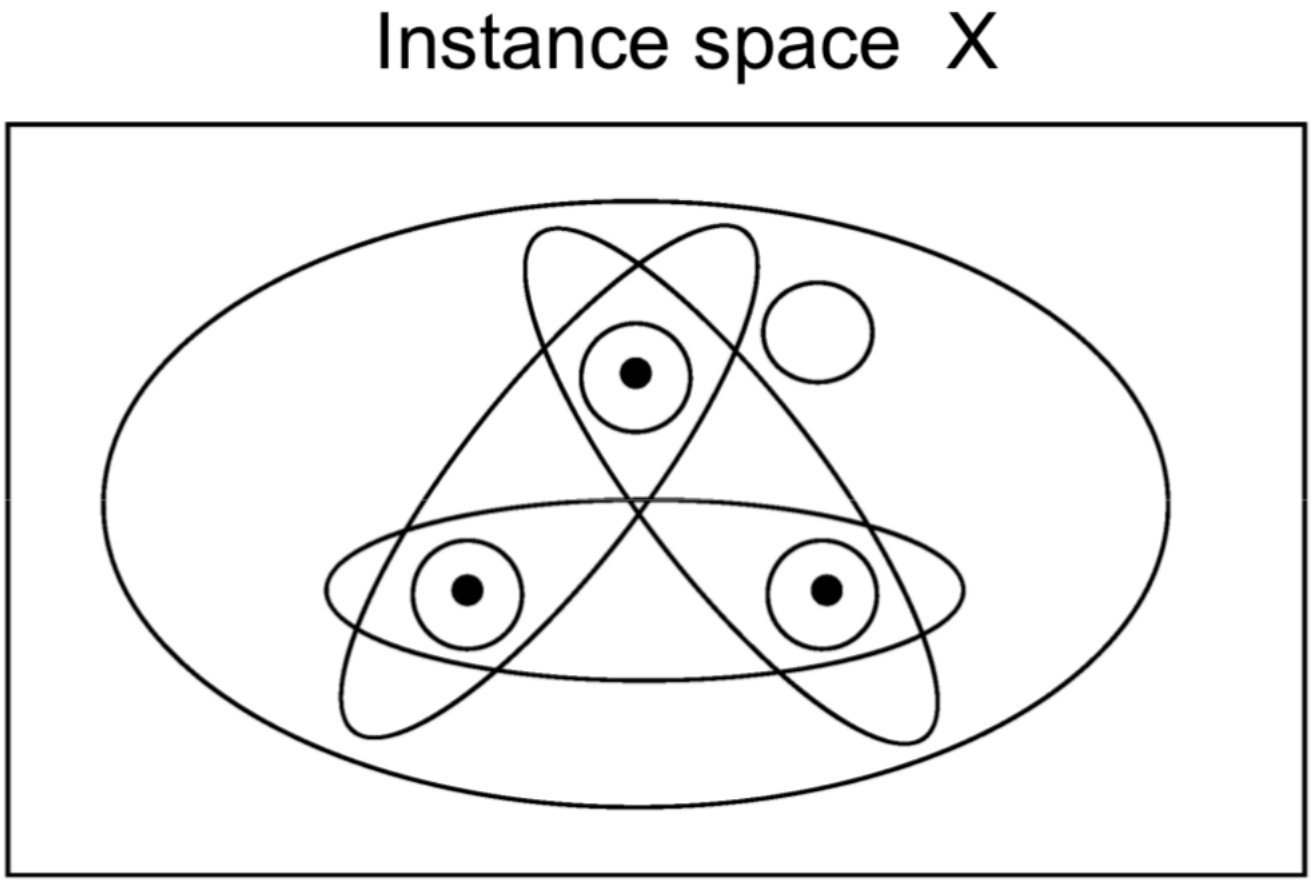
- *Definition*: Given a set $\mathcal{S} = \{x^{(1)}, \ldots, x^{(m)}\}$ (no relation to the training set) of points $x^{(i)} \in$ X, we say that $\mathcal{H}$ shatters $\mathcal{S}$ if $\mathcal{H}$ can realize any labeling on $\mathcal{S}$.

  I.e., if for any set of labels $\{y^{(1)}, \ldots, y^{(m)}\}$, there exists some $h \in \mathcal{H}$ so that $h(x^{(i)}) = y^{(i)}$ for all $i = 1, \ldots, m$.

- There are 2^m different ways to separate the sample into two sub-samples (a dichotomy)

[Slide from Eric Xing]

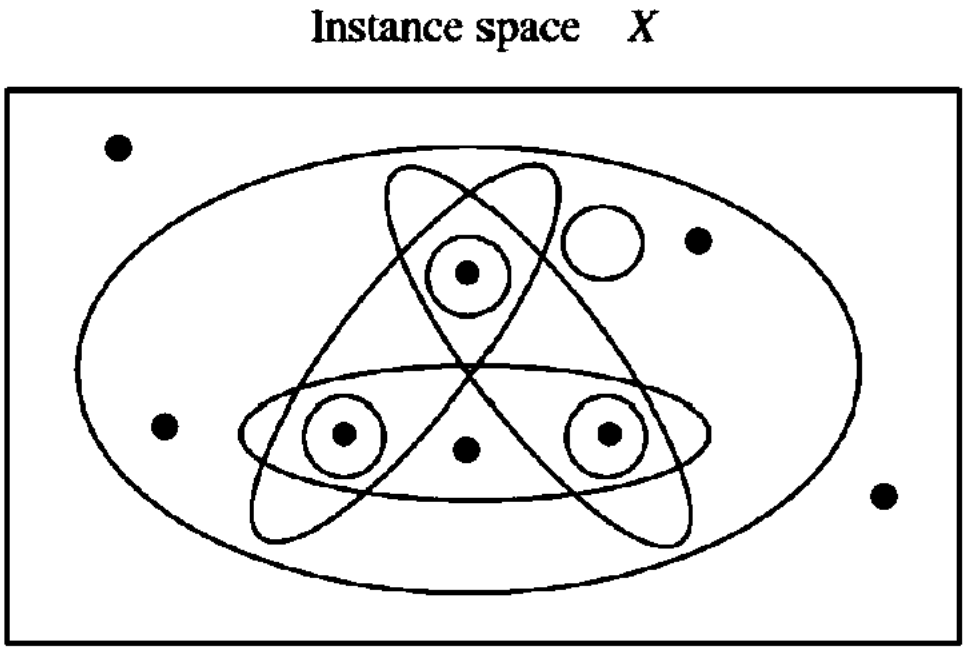
Three Instances Shattered



[Slide from Eric Xing]

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 $VC(H) \equiv \infty$.



[Slide from Eric Xing]

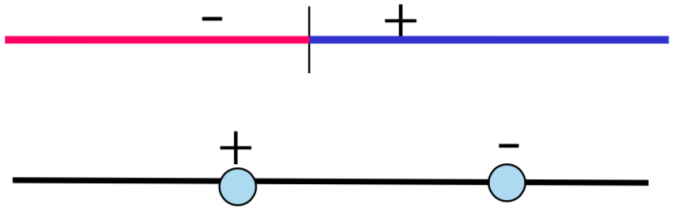
VC dimension: examples

Consider X = $\mathbb{R}$, want to learn c: X→{0,1}

What is VC dimension of

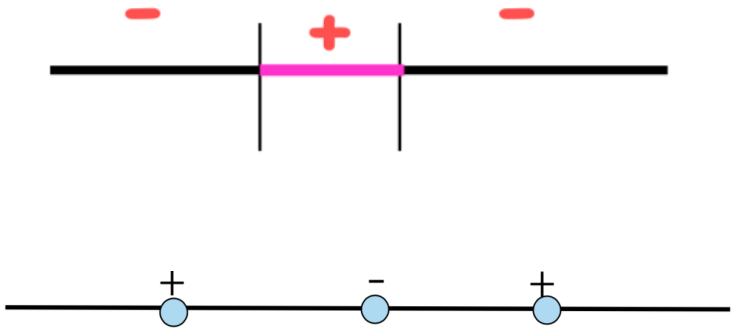
- Open intervals:

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



- Closed intervals:

  H2: if a<x<b, then y=1 else y=0



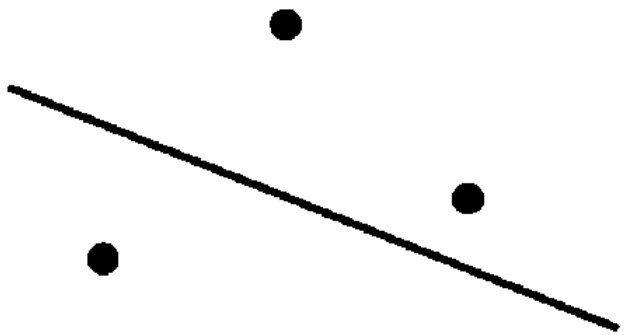
[Slide from Eric Xing and David Sontag]

VC dimension: examples

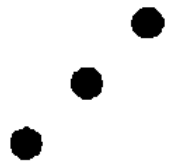
Consider X = $\mathbb{R}^2$, want to learn c: X→{0,1}

- What is VC dimension of lines in a plane?

  H= { ((wx+b)>0 → y=1) }

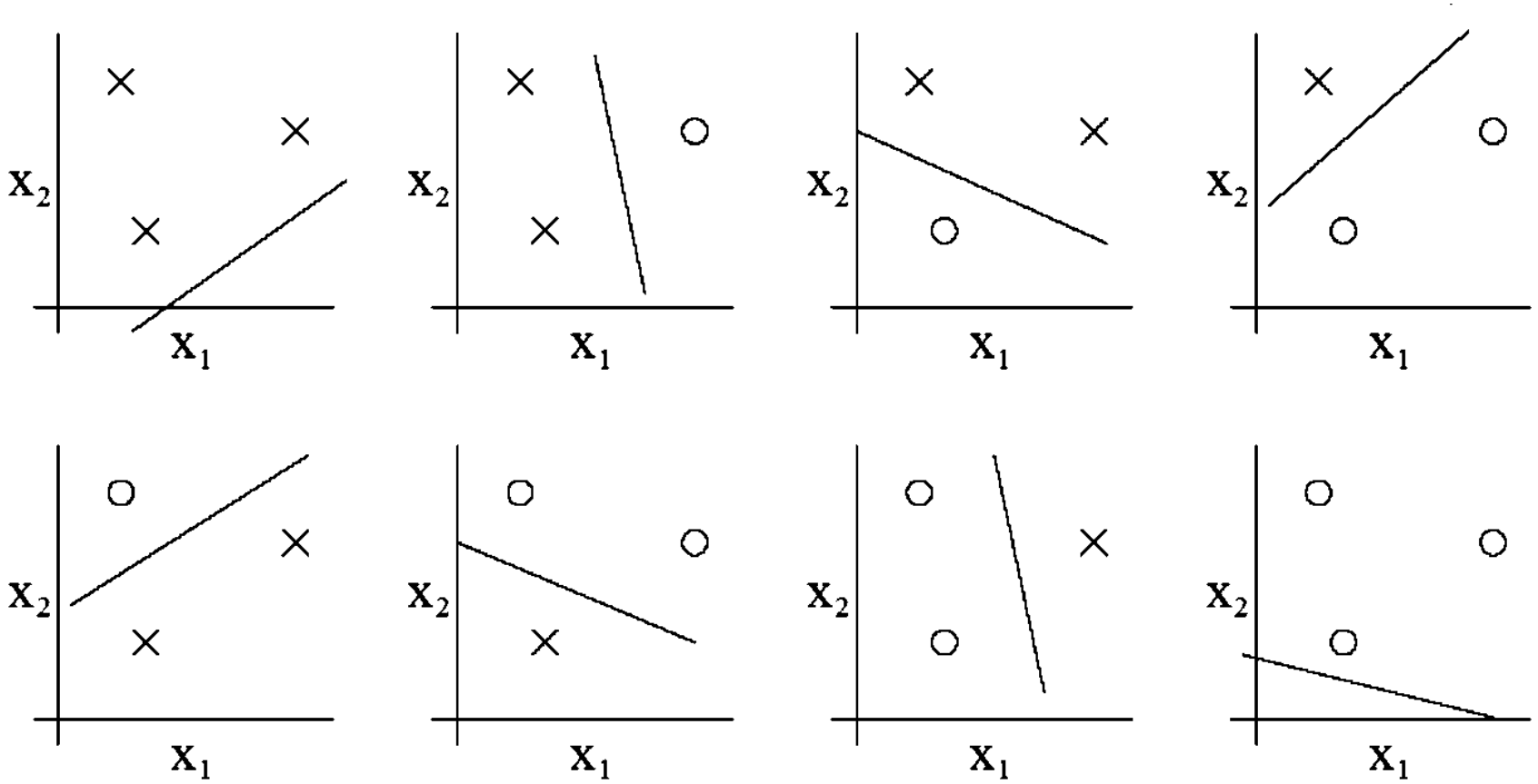


(a)



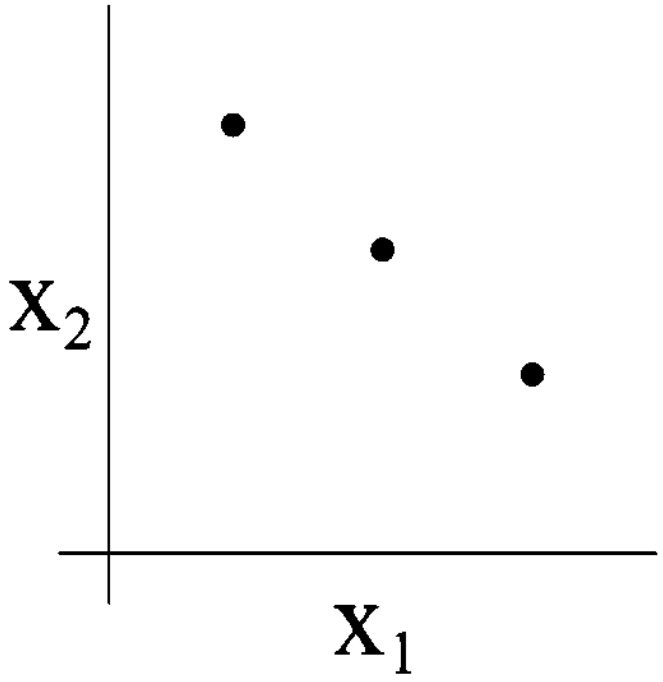
(b)

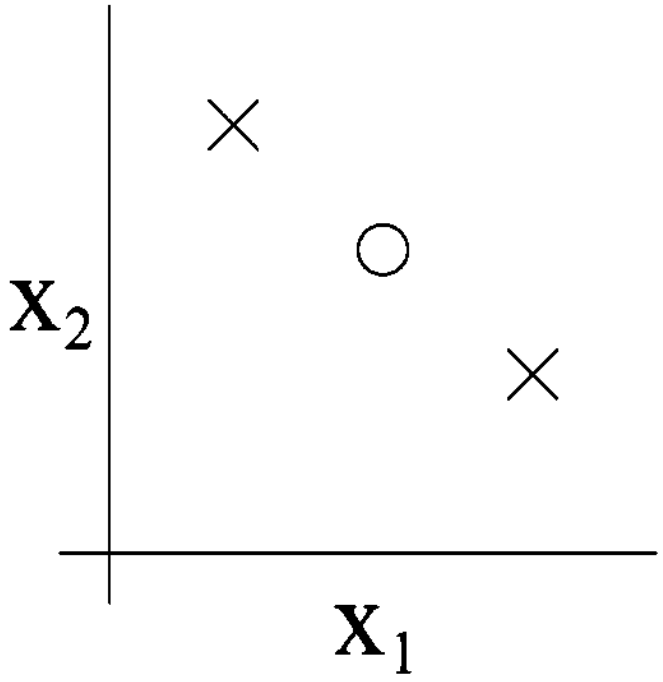
[Slide from Eric Xing]



- For any of the eight possible labelings of these points, we can find a linear classier that obtains "zero training error" on them.
- Moreover, it is possible to show that there is no set of 4 points that this hypothesis class can shatter.

[Slide from Eric Xing]





- The VC dimension of H here is 3 even though there may be sets of size 3 that it cannot shatter.
- under the definition of the VC dimension, in order to prove that VC(H) is at least *d*, we need to show only that there's at least one set of size d that H can shatter.

[Slide from Eric Xing]

- **Theorem** Consider some set of m points in $\mathbb{R}^n$. Choose any one of the points as origin. Then the m points can be shattered by oriented hyperplanes if and only if the position vectors of the remaining points are linearly independent.

- **Corollary**: The VC dimension of the set of oriented hyperplanes in $\mathbb{R}^n$ is n+1.

  Proof: we can always choose n + 1 points, and then choose one of the points as origin, such that the position vectors of the remaining n points are linearly independent, but can never choose n + 2 such points (since no n + 1 vectors in $\mathbb{R}^n$ can be linearly independent).

[Slide from Eric Xing]

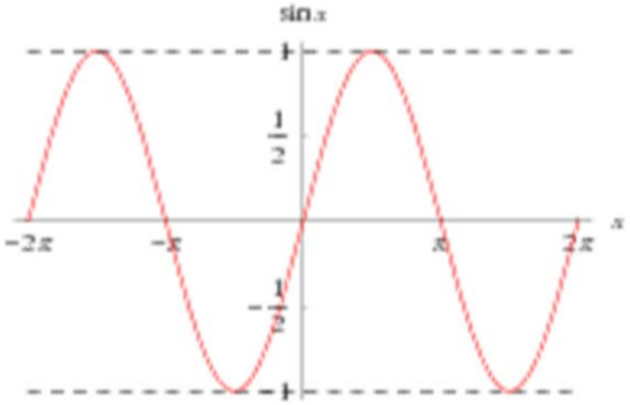
The VC Dimension and the Number of Parameters

- The VC dimension thus gives concreteness to the notion of the capacity of a given set of *h*.
- Is it true that learning machines with many parameters would have high VC dimension, while learning machines with few parameters would have low VC dimension?

- An infinite-VC function with just one parameter!

$$f(x, \alpha) \equiv \theta(\sin(\alpha x)), \quad x, \alpha \in R$$

where θ is an indicator function

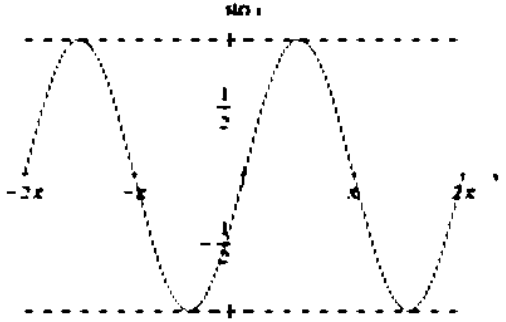


[Slide from Eric Xing]

An infinite-VC function with just one parameter

- You choose some number *l*, and present me with the task of finding *l* points that can be shattered. I choose them to be

$$x_i = 10^{-i} \quad i = 1, \ldots, l.$$



- You specify any labels you like:

$$y_1, y_2, \ldots, y_l, \quad y_i \in \{-1, 1\}$$

- Then $\theta(\alpha)$ gives this labeling if I choose α to be

$$\alpha = \pi(1 + \sum_{i=1}^{l} \frac{(1 - y_i)10^i}{2})$$

- Thus the VC dimension of this machine is infinite.

[Slide from Eric Xing]

Sample Complexity from VC Dimension

- How many randomly drawn examples suffice to ε-exhaust $VS_{H,S}$ 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 on testing data from the same distribution

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

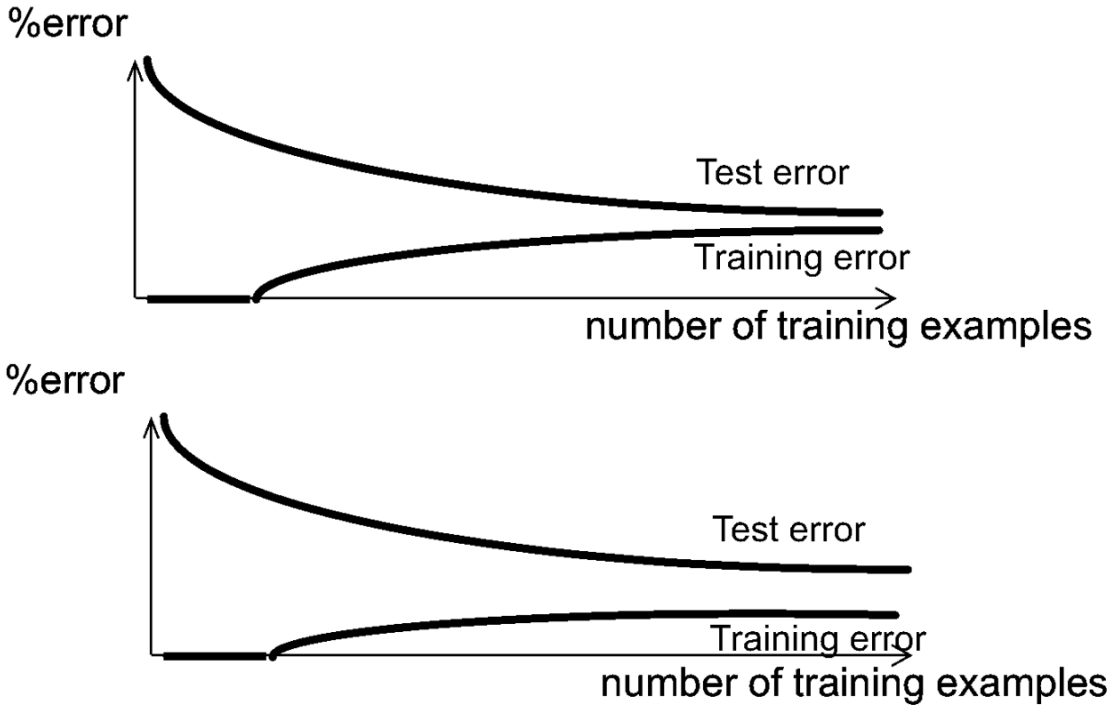
Compare to our earlier results based on $|H|$:

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

[Slide from Eric Xing]

Consistency

- A learning process (model) is said to be consistent if model error, measured on new data sampled from the same underlying probability laws of our original sample, converges, when original sample size increases, towards model error, measured on original sample.



[Slide from Eric Xing]

Vapnik main theorem

- Q : Under which conditions will a learning model be consistent?
- A : A model will be consistent if and only if the function *h* that defines the model comes from a family of functions *H* with finite VC dimension *d*

- A finite VC dimension d not only guarantees a generalization capacity (consistency), but to pick *h* in a family *H* with finite VC dimension *d* is the only way to build a model that generalizes.

[Slide from Eric Xing]

Agnostic Learning: VC Bounds

- ***Theorem***: Let H be given, and let d = VC(H). Then with probability at least 1-δ, we have that for all h $\in$ H,

$$|\hat{\epsilon}(h) - \epsilon(h)| \leq O\left(\sqrt{\frac{d}{m}\log\frac{m}{d} - \frac{1}{m}\log\delta}\right)$$

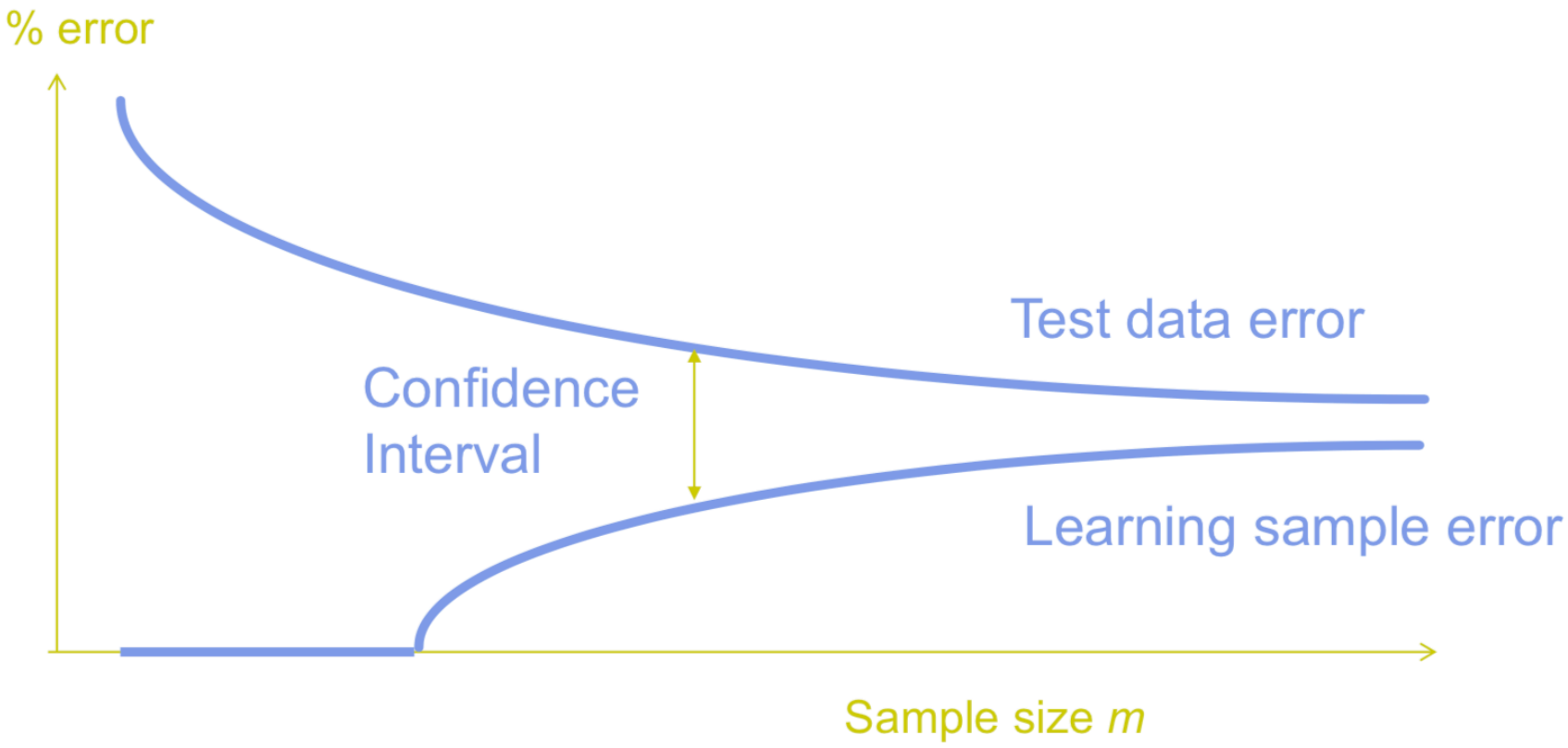
or $$\epsilon(h) \leq \hat{\epsilon}(h) + O\left(\sqrt{\frac{d}{m}\log\frac{m}{d} - \frac{1}{m}\log\delta}\right)$$

recall that in finite H case, we have:

$$|\hat{\epsilon}(h) - \epsilon(h)| \leq \sqrt{\frac{1}{m}\log 2k - \frac{1}{m}\log\delta}$$

[Slide from Eric Xing]

Model convergence speed



[Slide from Eric Xing]

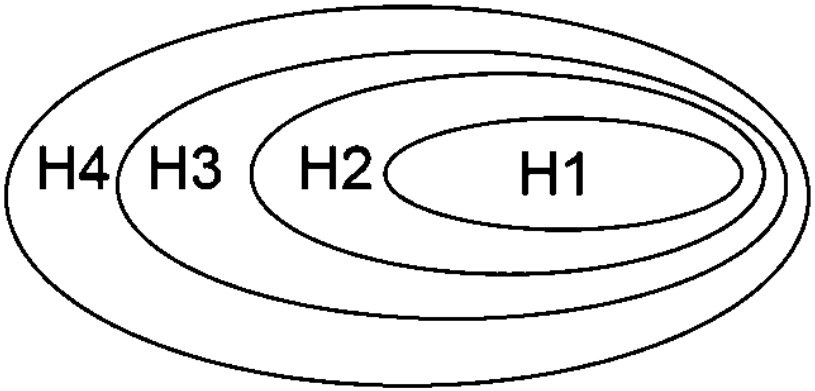
How to control model generalization capacity

- Risk Expectation = Empirical Risk + Confidence Interval
- To minimize Empirical Risk alone will not always give a good generalization capacity: one will want to minimize the sum of Empirical Risk and Confidence Interval
- What is important is not the numerical value of the Vapnik limit, most often too large to be of any practical use, it is the fact that this limit is a non decreasing function of model family function "richness"

[Slide from Eric Xing]

Structural Risk Minimization

- Which hypothesis space should we choose?
- Bias / variance tradeoff



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

$$\epsilon(h) \leq \hat{\epsilon}(h) + O\left(\sqrt{\frac{d}{m}\log\frac{m}{d} - \frac{1}{m}\log\delta}\right)$$

unfortunately a somewhat loose bound...

[Slide from Eric Xing]

SRM strategy

- With probability $1-\delta$,

$$\epsilon(h) \leq \hat{\epsilon}(h) + O\left(\sqrt{\frac{d}{m}\log\frac{m}{d} - \frac{1}{m}\log\delta}\right)$$

- When m/d is small (d too large), second term of equation becomes large

- SRM basic idea for strategy is to minimize simultaneously both terms standing on the right of above majoring equation for $\varepsilon(h)$

- To do this, one has to make d a controlled parameter

[Slide from Eric Xing]

SRM strategy

- Let us consider a sequence $H_1 < H_2 < .. < H_n$ of model family functions, with respective growing VC dimensions

$$d_1 < d_2 < .. < d_n$$

- For each family H_i of our sequence, the inequality

$$\epsilon(h) \leq \hat{\epsilon}(h) + O\left(\sqrt{\frac{d}{m}\log\frac{m}{d} - \frac{1}{m}\log\delta}\right)$$

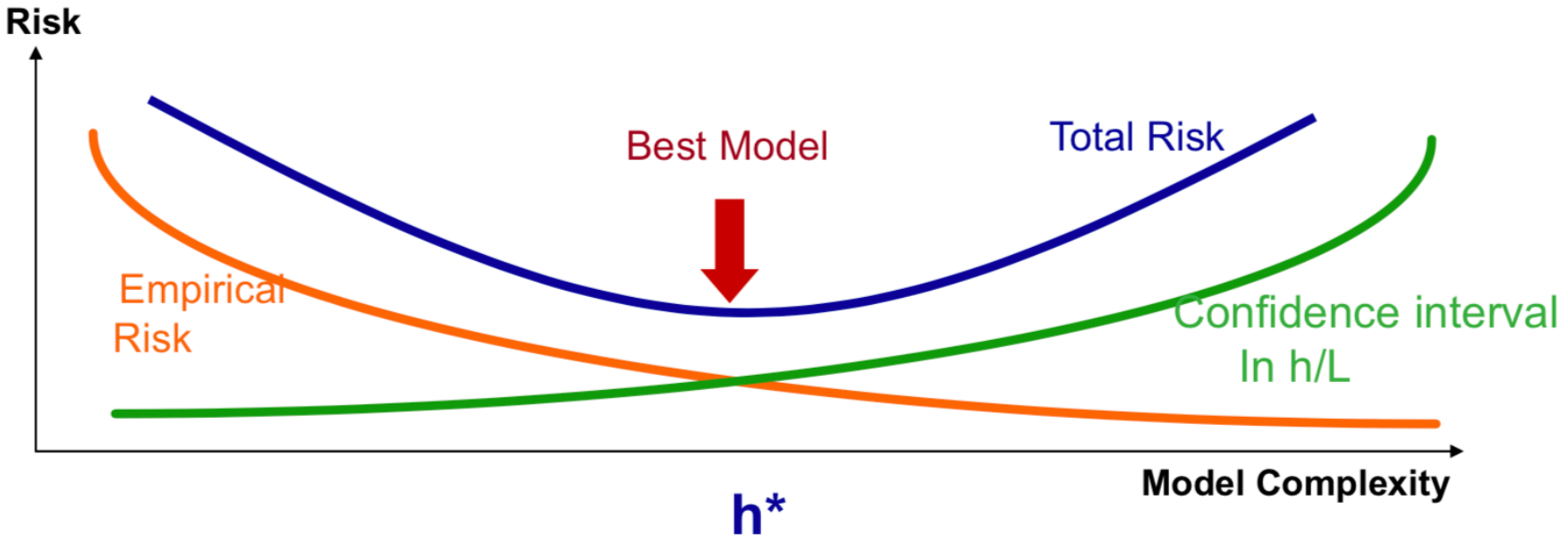
is valid

  - That is, for each subset, we must be able either to compute *d*, or to get a bound on *d* itself.

- SRM then consists of finding that subset of functions which minimizes the bound on the actual risk.

[Slide from Eric Xing]

SRM strategy



[Slide from Eric Xing]

Putting SRM into action: linear models case

- There are many SRM-based strategies to build models:
- In the case of linear models

$$y = w^T x + b$$

- one wants to make ||w|| a controlled parameter: let us call HC the linear model function family satisfying the constraint:

$$||w|| < C$$

- Vapnik Major theorem: When C decreases, $d(H_C)$ decreases

[Slide from Eric Xing]

Putting SRM into action: linear models case

To control ||w||, one can envision two routes to model:

- *Regularization/Ridge Regression*, ***ie min. over w and b***

$$RG(w,b) = S\{(\ y_i - w^T x_i - b\ )^2 \mid i=1,..,L\} + \lambda\ ||w||^2$$

- *Support Vector Machines (SVM)*, ***ie solve directly an optimization problem (classif. SVM, separable data)***

Minimize $||w||^2$,

with $(y_i = +/-1)$

and $y_i(\ \ w^T x_i + b\ \ ) >= 1$ *for all* $i=1,..,L$

[Slide from Eric Xing]

Take away message

- Sample complexity varies with the learning setting
  - Learner actively queries trainer
  - Examples provided 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
- Quantitative bounds characterizing bias/variance in choice of H

[Slide from Eric Xing]

Take home message

	PAC	Agnostic
Finite $\lvert\mathcal{H}\rvert$	**Thm. 1** $N \geq \frac{1}{\epsilon}\left[\log(\lvert\mathcal{H}\rvert) + \log(\frac{1}{\delta})\right]$ labeled examples are sufficient so that with probability $(1-\delta)$ all $h \in \mathcal{H}$ with $\hat{R}(h) = 0$ have $R(h) \leq \epsilon$.	**Thm. 2** $N \geq \frac{1}{2\epsilon^2}\left[\log(\lvert\mathcal{H}\rvert) + \log(\frac{2}{\delta})\right]$ labeled examples are sufficient so that with probability $(1-\delta)$ for all $h \in \mathcal{H}$ we have that $\lvert R(h) - \hat{R}(h)\rvert \leq \epsilon$.
Infinite $\lvert\mathcal{H}\rvert$	**Thm. 3** $N=O(\frac{1}{\epsilon}\left[\mathrm{VC}(\mathcal{H})\log(\frac{1}{\epsilon}) + \log(\frac{1}{\delta})\right])$ labeled examples are sufficient so that with probability $(1-\delta)$ all $h \in \mathcal{H}$ with $\hat{R}(h) = 0$ have $R(h) \leq \epsilon$.	**Thm. 4** $N = O(\frac{1}{\epsilon^2}\left[\mathrm{VC}(\mathcal{H}) + \log(\frac{1}{\delta})\right])$ labeled examples are sufficient so that with probability $(1-\delta)$ for all $h \in \mathcal{H}$ we have that $\lvert R(h) - \hat{R}(h)\rvert \leq \epsilon$.

[Slide from Matt Gormley]

References

- Eric Xing, Ziv Bar-Joseph. 10701 Introduction to Machine Learning: http://www.cs.cmu.edu/~epxing/Class/10701/
- Matt Gormley. 10601 Introduction to Machine Learning: http://www.cs.cmu.edu/~mgormley/courses/10601/index.html
- David Sontag. Introduction To Machine Learning. https://people.csail.mit.edu/dsontag/courses/ml12/slides/lecture14.pdf