# Learning Theory

Adrian Barbu

# Complexity of Learning

- The complexity of learning is measured mainly along two axes:
  - Information
  - Computation.
- The Information complexity is concerned with the generalization performance of learning;
  - *How many training examples are needed?*
  - *How fast do learners estimate converge to the true population parameters?*
- The Computational complexity concerns the computation resources applied to the training data to extract the predictions from it.

- Usually, when an algorithm improves with respect to one of these measures it deteriorates with respect to the other.

# Typical Learning Task

- Binary classification
  - Generalizes to other tasks: regression, multiclass classification
- Given:
  - Instances (observations) *X.*
    - E.g. possible day scenarios, each described by the attributes *Sky, AirTemp, Humidity, Wind, Water, Forecast*
  - Target function *c*: X $\longrightarrow$ {0, 1}
    - *E.g. c=EnjoySport* , given by a table
  - Hypothesis space $\mathcal{H}$
    - E.g. space of Boolean combinations of conditions
  - Training examples *S*: iid positive and negative examples of the target function
    - *E.g.* $(x_1, c(x_1)), \ldots (x_n, c(x_n))$
- Determine:
  - A hypothesis $h \in \mathcal{H}$ such that *h(x)* is "good" w.r.t *c(x)* for all *x* in *S*
  - A hypothesis $h \in \mathcal{H}$ such that *h(x)* is "good" w.r.t *c(x)* for all *x* in the true distribution D

# Two Learning Frameworks

- PAC (Probable Approximately Correct) framework
  - Probable: Most sample labels are consistent with some $h \in \mathcal{H}$
  - Approximately Correct: The hypothesis *h* is required to meet an *absolute* upper bound on its error

- Agnostic framework
  - No prior restriction on the sample labels
  - The required upper bound on the hypothesis error is only relative to the best hypothesis in the class

# Training Scenarios

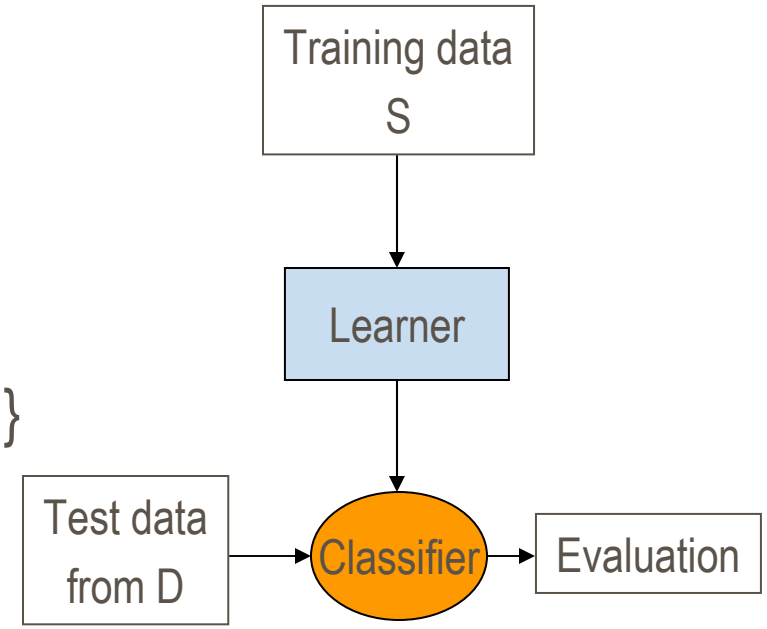
- Supervised learning
  - The teacher (e.g. human) knows target function $c$: $X \rightarrow \{0, 1\}$

Three Scenarios

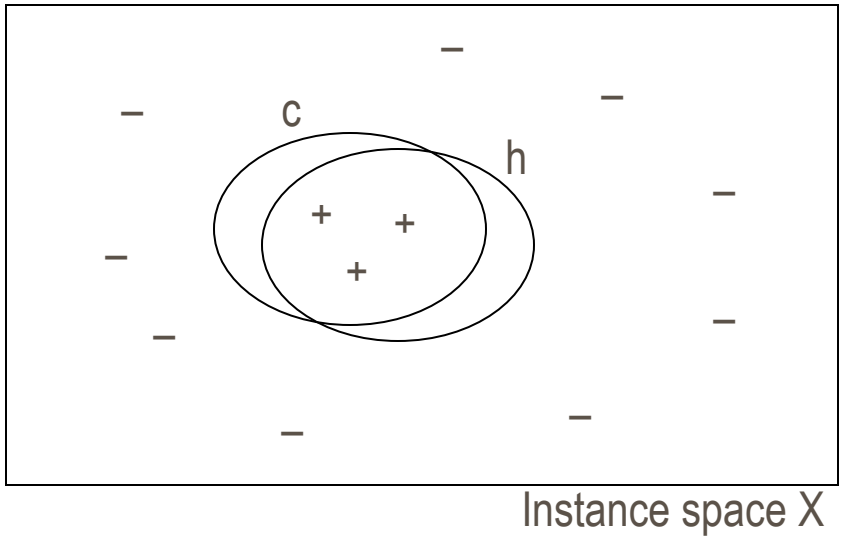
1. Learner (algorithm) proposes instances as queries to the teacher
   - Learner proposes instance x, teacher provides $c(x)$
   - A.k.a *Active Learning*
2. Teacher provides training examples
   - Teacher provides a sequence of examples of the form $(x, c(x))$
3. Some random process (e.g., nature) proposes instances
   - Instance $x$ generated randomly, teacher provides $c(x)$

# Learning and Evaluation Protocol

- Given:
  - A space of instances *X*
  - A fixed (unknown) distribution *D* over *X*
  - A set of hypotheses $\mathcal{H}$
  - A set of possible target functions *C*
- Learner observes sample S = { ($x_i$, $c(x_i)$ ),*i* }
  - Instances $x_i$ drawn from distribution *D*
  - Labeled by target function $c \in C$
    - Learner does NOT know *c(.)* or *D*
- Learner outputs $h \in \mathcal{H}$ estimating *c*
  - *h* is evaluated by performance on subsequent instances drawn from *D*
- For now assume:
  - $C = \mathcal{H}$ (so $c \in \mathcal{H}$)
  - Noise-free data



# True Error of a Hypothesis




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

$$\epsilon_D(h) = Pr_{x \in D}[c(x) \neq h(x)]$$

# Two Errors

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

$$\epsilon_S(h) = Pr_{x \in S}[c(x) \neq h(x)] = \frac{1}{|S|} \sum_{x \in S} \delta(c(x) \neq h(x))$$

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

$$\epsilon_D(h) = Pr_{x \in D}[c(x) \neq h(x)]$$

# Hoeffding Inequality

- Lemma. (Hoeffding inequality) Let $Z_1,\ldots,Z_m$ be iid random variables drawn from a Bernoulli(φ) distribution, i.e., $P(Z_i = 1) = \varphi$ , and $P(Z_i = 0) = 1- \varphi$. Let $\hat{\phi} = \frac{1}{m}\sum_{i=1}^{m} Z_i$ be their mean, and let any $\gamma > 0$ be fixed. Then

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

  - A.k.a. the Chernoff bound
  - It says that if we take $\hat{\phi}$ - the average of m Bernoulli($\phi$) random variables - as our estimate of $\phi$, then the probability of our being far from the true value is small, for large enough m

# Consistency and Version Space

- A hypothesis *h* is consistent with a set of training examples S of a target function c if and only if *h*(x)=c(x) for each training example $(x_i, c(x_i)) \in S$.

$$Consistent(h, S) \models \quad h(x) = c(x),\ \forall (x, c(x)) \in S$$

- The version space, $VS_{H,S}$, w.r.t hypothesis space $\mathcal{H}$ and training examples S is the subset of hypotheses from $\mathcal{H}$ consistent with all training examples in S.

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

# Consistent Learner

- A learner is *consistent* if it outputs a hypothesis that perfectly fits the training data (i.e. is consistent)
  - A reasonable learning strategy

- Every consistent learning outputs a hypothesis belonging to the version space

- We want to know how such hypothesis generalizes

# Probable Approximately Correct

Verifying generalization ability

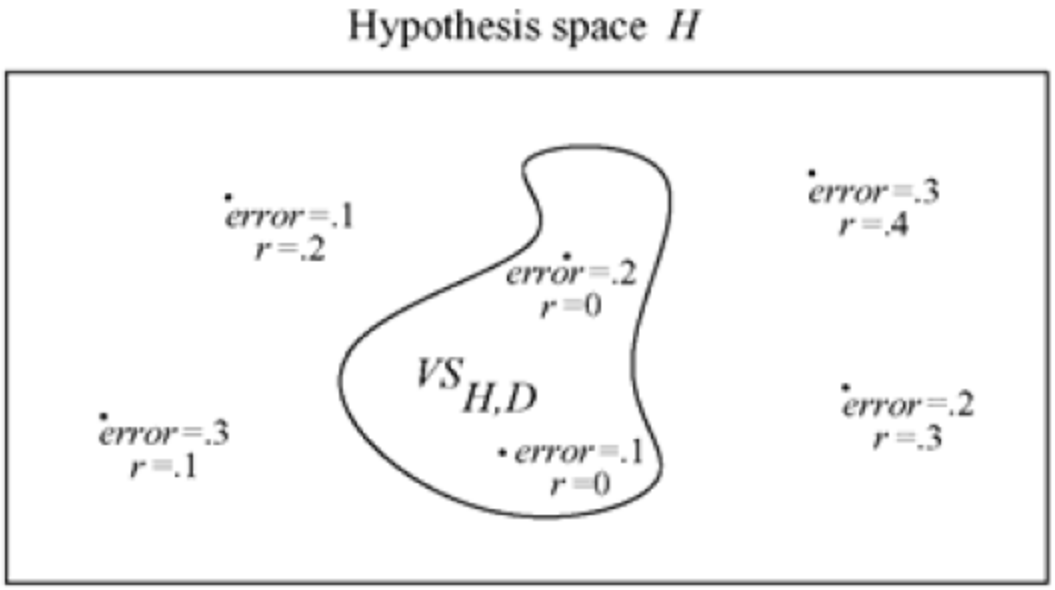
- Goal:
  - A PAC-Learner produces a hypothesis $\hat{h}$ that is approximately correct,

    $$err_D(\hat{h}) \approx 0$$

    with high probability

    $$P( err_D(\hat{h}) \approx 0 ) \approx 1$$
- Double “hedging"
  - Approximately correct
  - Probably i.e. almost all the time
- Need both!

# Exhausting the Version Space




($r$ = training error, $error$ = 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 *D*.

$$\epsilon_D(h) < \epsilon, \quad \forall h \in VS_{\mathcal{H},S}$$

# Error Bound from Training Examples

Theorem: [Haussler, 1988].

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

$$|\mathcal{H}| \exp(-\epsilon m)$$

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

# How Many Training Examples?

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

$$Pr(\exists h \in \mathcal{H}, err_{train}(h) = 0, err_{true}(h) > \epsilon) < |\mathcal{H}| e^{-\epsilon m}$$

- If we want this probability to be at most δ

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

1. How many training examples suffice?

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

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

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

# Conjunction on Booleans

- How many examples are sufficient to assure with probability at least (1 - δ) that every *h* in $VS_{H,S}$ satisfies $\varepsilon_D(h) \leq \varepsilon$
- Use the theorem:

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

Example

- Suppose *H* contains conjunctions of constraints on up to *n* Boolean attributes (i.e., *n* Boolean literals).
  - Then $|H| = 3^n$, and the number of training examples to have $\varepsilon_D(h) \leq \varepsilon$ with probability at least (1 - δ)

$$m \geq \frac{1}{\epsilon}(\ln |\mathcal{H}| + \ln \frac{1}{\delta}) = \frac{1}{\epsilon}(n \ln 3 + \ln \frac{1}{\delta})$$

  - Linear with the classifier complexity

# PAC Learnability

- A learning algorithm is PAC learnable if it
  - Requires no more than polynomial computation per training example, and
  - No more than a polynomial number of examples

- **Theorem**: conjunctions of Boolean literals is PAC learnable

# Enjoy Sport

| Sky | Temp | Humid | Wind | Water | Forecast | EnjoySport |
|---|---|---|---|---|---|---|
| Sunny | Warm | Normal | Strong | Warm | Same | Yes |
| Sunny | Warm | High | Strong | Warm | Same | Yes |
| Rain | Cold | High | Strong | Warm | Change | No |
| Sunny | Warm | High | Strong | Cool | Change | Yes |

- Totally, 973 possible distinct functions, so $|H| = 973$, and

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

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

$$m \geq \frac{1}{0.1}(\ln 973 + \ln \frac{1}{0.05}) \approx 98.8$$

# PAC Learning

- Learner L can draw labeled instances (x, c(x)) in unit time, $x \in X$ of length *n* drawn from distribution D, labeled by target function $c \in C$

- **Definition**: Learner L PAC-learns class C using hypothesis space H if
  1. For any target function $c \in C$, any distribution 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.≥1-δ, $err_D(h) < \varepsilon$
  2. L's run-time (and hence, sample complexity) is poly(|x|, size(c), 1/ε, 1/δ)
- Sufficient:
  - Only poly(…) training instances – $|H| = 2^{poly()}$
  - Only poly time / instance …
- Often C = H

# Agnostic Learning

- So far, assumed $c \in H$
- Agnostic learning: don't assume $c \in$ H

- What do we want then?
  - The hypothesis *h* that makes fewest errors on training data

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

- What is sample complexity in this case?
  - derived from Hoeffding bounds:

$$Pr[err_D(h) > err_S(h) + \epsilon] < e^{-2m\epsilon^2}$$

# Empirical Risk Minimization

- Choose a *Hypothesis Class H* of subsets of *X*.
  - For an training set *S* , find some $h \in H$ that fits *S* "well".
- For a new point *x* , predict a label according to its membership in *h*.

- Example:
  - Consider linear classification, and let $h_\theta(x) = \delta(\theta^T x > 0)$
  - Then $\hat{\theta} = \arg\min_\theta \epsilon_S(h_\theta)$
  - 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

# Uniform Convergence

- We don't just want to guarantee errors (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 $k$ hypotheses:

$$Pr[\exists h, |\epsilon(h) - \hat{\epsilon}(h)| > \gamma] < 2ke^{-2m\gamma^2}$$

- Hence

$$Pr[\nexists h, |\epsilon(h) - \hat{\epsilon}(h)| > \gamma] > 1 - 2ke^{-2m\gamma^2}$$

# Sample Complexity

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

$$Pr[\exists h, |\epsilon(h) - \hat{\epsilon}(h)| > \gamma] < 2ke^{-2m\gamma^2}$$

- We find that if

$$m \geq \frac{1}{2\gamma^2} \ln \frac{2|\mathcal{H}|}{\delta} = \frac{1}{2\gamma^2}(\ln |\mathcal{H}| + \ln \frac{2}{\delta})$$

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

- The number of training examples needed to make this guarantee is only logarithmic in $|H|$, the number of hypotheses in H. This will be important later.

# Generalization Error Bound

- Similarly, we can hold *m* and $\delta$ fixed and solve for $\gamma$ in the previous equation, and show that with probability 1- $\delta$ , we have that for all $h_i \in H$

$$|\hat{\epsilon}(h) - \epsilon(h)| < \sqrt{\frac{1}{m} \log \frac{2|\mathcal{H}|}{\delta}}$$

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

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

- If uniform convergence occurs, then the generalization error of is at most $2\gamma$ worse than the best possible hypothesis in H!

# Summary

- Theorem. If m,$\delta$ are fixed, then with probability at least 1-$\delta$, we have

$$\epsilon(\hat{h}) < (\min_{h \in \mathcal{H}} \epsilon(h)) + 2\sqrt{\frac{1}{2m} \log \frac{2|\mathcal{H}|}{\delta}}$$

- Corollary. If $\delta$,$\gamma$ are fixed, then

$\epsilon(\hat{h}) < (\min_{h \in \mathcal{H}} \epsilon(h)) + 2\gamma$ with probability 1-$\delta$ if

$$m \geq \frac{1}{2\gamma^2} \log \frac{2|\mathcal{H}|}{\delta} = O(\frac{1}{\gamma^2} \log \frac{|\mathcal{H}|}{\delta})$$

# Sample Complexity from VC Dimension

- How many randomly drawn examples suffice to ε-exhaust $VS_{H,S}$ with probability at least (1 - $\delta$)?
  - ie., to guarantee that any hypothesis that perfectly fits the training data is probably (1-$\delta$) approximately (ε) correct on testing data from the same distribution

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

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

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

# Mistake Bounds

- So far: how many examples needed to learn?
- Another question of interest:
  - how many mistakes before convergence?

- A similar setting to PAC learning:
  - Instances drawn at random from *X* according to distribution *D*
  - Learner must classify each instance before receiving correct classification from teacher
  - Can we bound the number of mistakes learner makes before converging?

# Statistical Learning

- A model is a function $h(X, w)$ with some parameters w.

- Problem : minimize in *w* the Expected Risk

$$R(w) = \int Q(z, w) dP(z)$$

  - *w* : a parameter that specifies the chosen model
  - *z* = *(X, y)* are possible values for attributes (variables)
  - *Q* measures (quantifies) model error cost
  - *P(z)* is the underlying probability distribution (unknown) for data z

# Empirical Risk

- We get *L* data points $(z_1, .. , z_L)$, and we suppose they are iid sampled from law *P(z)*.
- To minimize *R(w)*, we start by minimizing Empirical Risk over these samples:

$$E(w) = \frac{1}{L} \sum_{i=1}^{L} Q(z_i, w)$$

- We can use such an approach for :
  - classification (eg. *Q* can be a cost function based on misclassified points, e.g. misclassification error, logistic loss, etc)
  - regression (eg. *Q* can be a cost such as sum of squares)

# Statistical Learning

- Central problem for Statistical Learning Theory:

  What is the relation between
  - Expected Risk *R(W)*
  - Empirical Risk *E(W)*

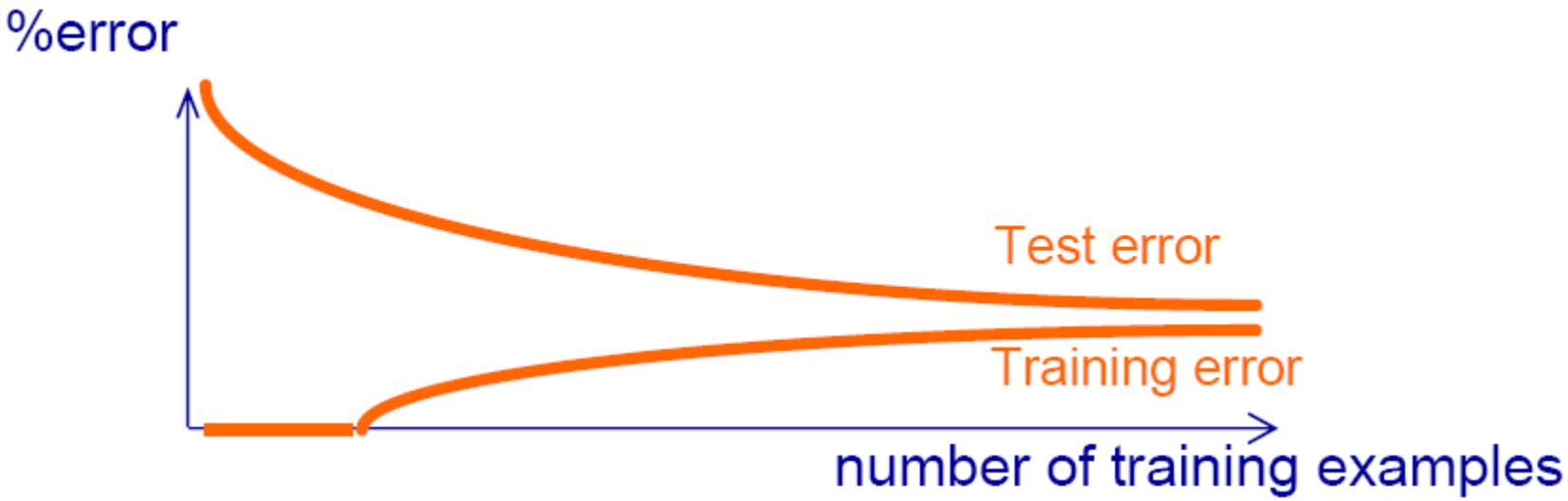
- How to define and measure the generalization ability ("robustness") for a model ?

# Four Basic Issues for SLT

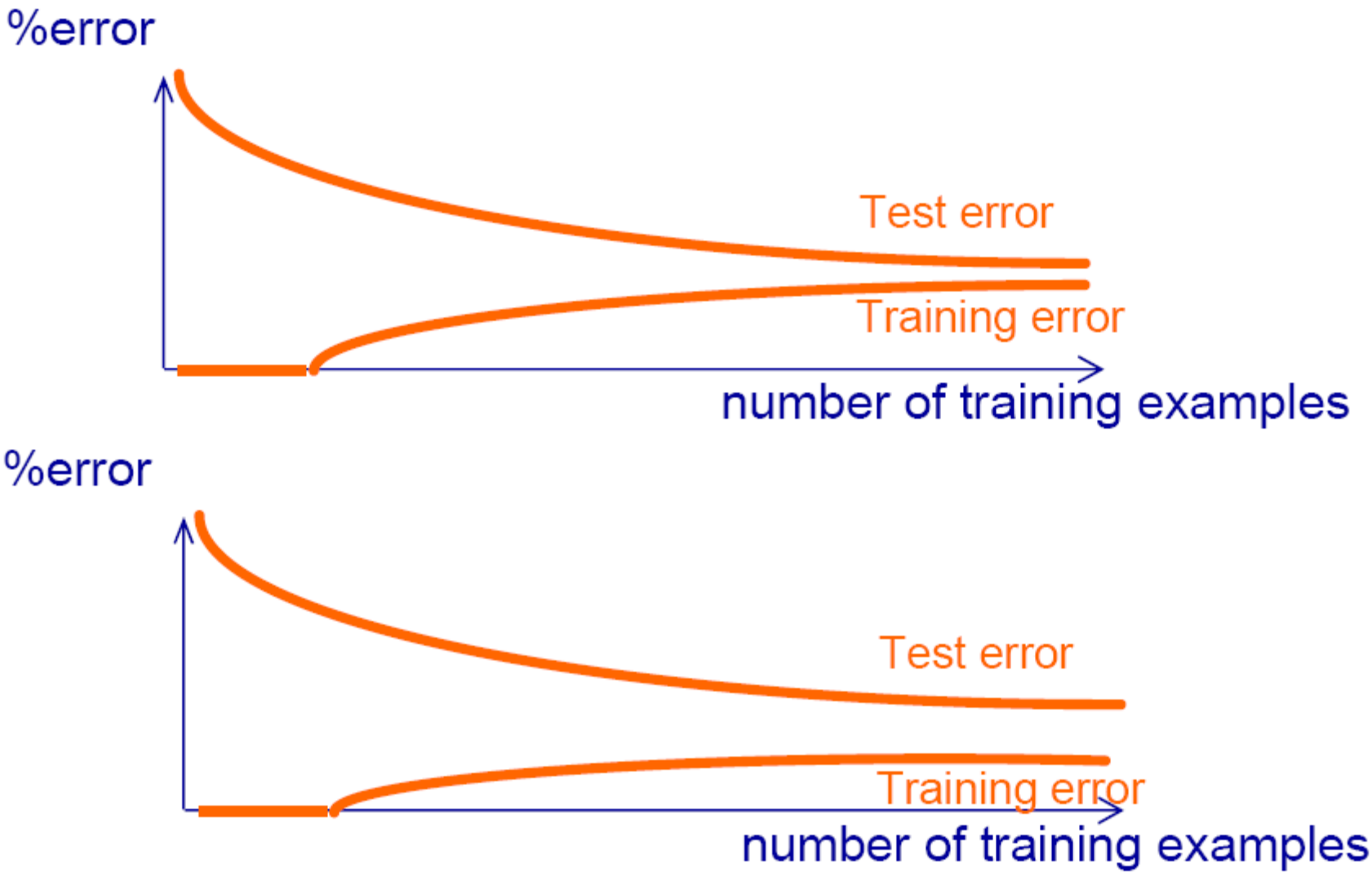
- Consistency (guarantees generalization)
  - Under what conditions will a model be consistent ?
- Model convergence speed (a measure of generalization)
  - How does generalization ability improve when sample size L grows?
- Generalization ability control
  - How to control in an efficient way model generalization starting with the only given the information we have: our sample data?
- A strategy for good learning algorithms
  - Is there a strategy that guarantees, measures and controls our learned model's generalization ability ?

# Consistency

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



# Consistent vs Not Consistent



# The Vapnik-Chervonenkis Dimension

- Learning=function interpolation
  - The space of possible instances X
  - The space of possible function values Y
- Remember the space of hypotheses (allowable functions)

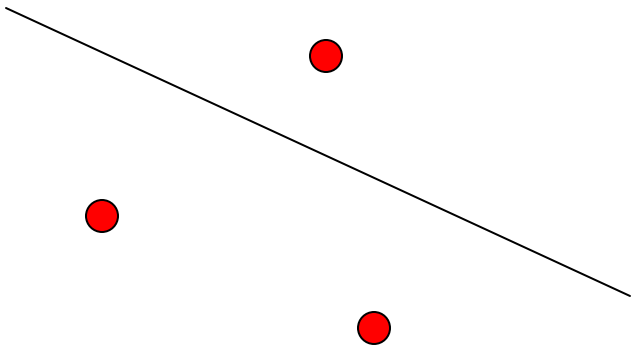
$$\mathcal{H} \subset \{h : X \to Y\}$$

- *Definition*: Given a set S = {$x_1$, … , $x_d$} of points $x_i \in$ X, we say that $\mathcal{H}$ shatters S if $\mathcal{H}$ can realize any labeling on S.

  i.e., if for any set of labels {$y_1$, … , $y_d$}, there exists some $h \in \mathcal{H}$ such that

$$h(x_i) = y_i, \forall i \in \{1, .., d\}$$

# Example: Three Points Shattered



- Can we shatter the three points using the family $\mathcal{H}$ of Logistic Regressors?

# The Vapnik-Chervonenkis Dimension

- *Definition*: The Vapnik-Chervonenkis (VC) dimension, $VC(\mathcal{H})$ is the size of the *largest finite subset* of *X* shattered by $\mathcal{H}$.
  - If arbitrarily large finite sets of *X* can be shattered by $\mathcal{H}$, then

$$VC(\mathcal{H}) = \infty$$

- Measure of how powerful is $\mathcal{H}$ to model data.

# Example: 1d

- VC dimension of
  - Infinite intervals

$$\mathcal{H} : h(x) = 0 \text{ if } x < a, \text{ else } 1$$

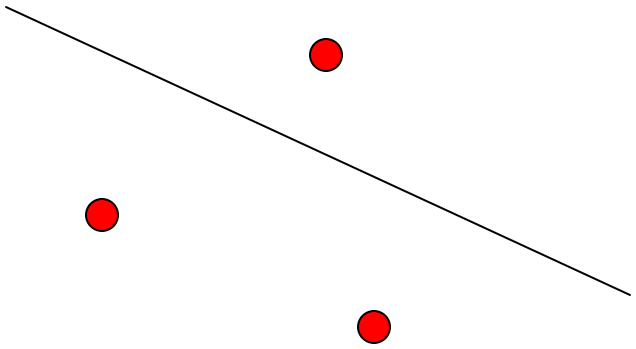
  - Finite intervals

$$\mathcal{H} : h(x) = 1 \text{ if } a < x < b, \text{ else } 0$$

# Example: 2D

- VC Dimension of linear separator (e.g. logistic regressor):

$$\mathcal{H} : h(x) = 0 \text{ if } wx + b < 0, \text{ else } 1$$



- It is possible to show that there is no set of 4 points that this hypothesis class can shatter.

# VC Dimension of Hyperplanes

Consider *m* points in $R^n$. Choose any one of the points as origin.

- **Theorem:** The *m* points can be shattered by oriented hyperplanes (e.g. logistic regressors) 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 $R^n$ is *n*+1.

# VC Dimension and Number of Parameters

- Is it true that hypotheses with many parameters would have high VC dimension, while hypotheses with few parameters would have low VC dimension?
  - No

- An infinite-VC function with just one parameter!

$$\mathcal{H} : h(x, \alpha) = \theta(\sin(x\alpha))$$

where θ is an indicator function

  - Can find sets of points of any size that can be shattered using appropriate values of $\alpha$

# Infinite VC Function with One Parameter

- For any dimension d, choose the d points to be

$$x_i = 10^{-i}, i = 1, ..., d$$

- For any labeling $y_1,\ldots,y_d \in \{-1,1\}^d$
- Choose

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

- Can check that *f*(α) gives this labeling

- Thus the VC dimension of this set of hypotheses is infinite.

# Vapnik's 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*

$$\epsilon(h) < \hat{\epsilon}(h) + \sqrt{\frac{d(\log \frac{2m}{d} + 1) + \log \frac{1}{4\delta}}{m}}$$

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

# Model Convergence Speed

- Q : What is the nature of model error difference between training data and test data, for a finite number of samples *m*?
- A : This difference is no greater than a limit that only depends on the ratio between the VC dimension *d* of the function family *H*, and the number of samples i.e., *d/m*
- This statement is a new theorem that belongs to Kolmogorov-Smirnov way of results, i.e., theorems that do not depend on data's underlying probability law.

# 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∈H,

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

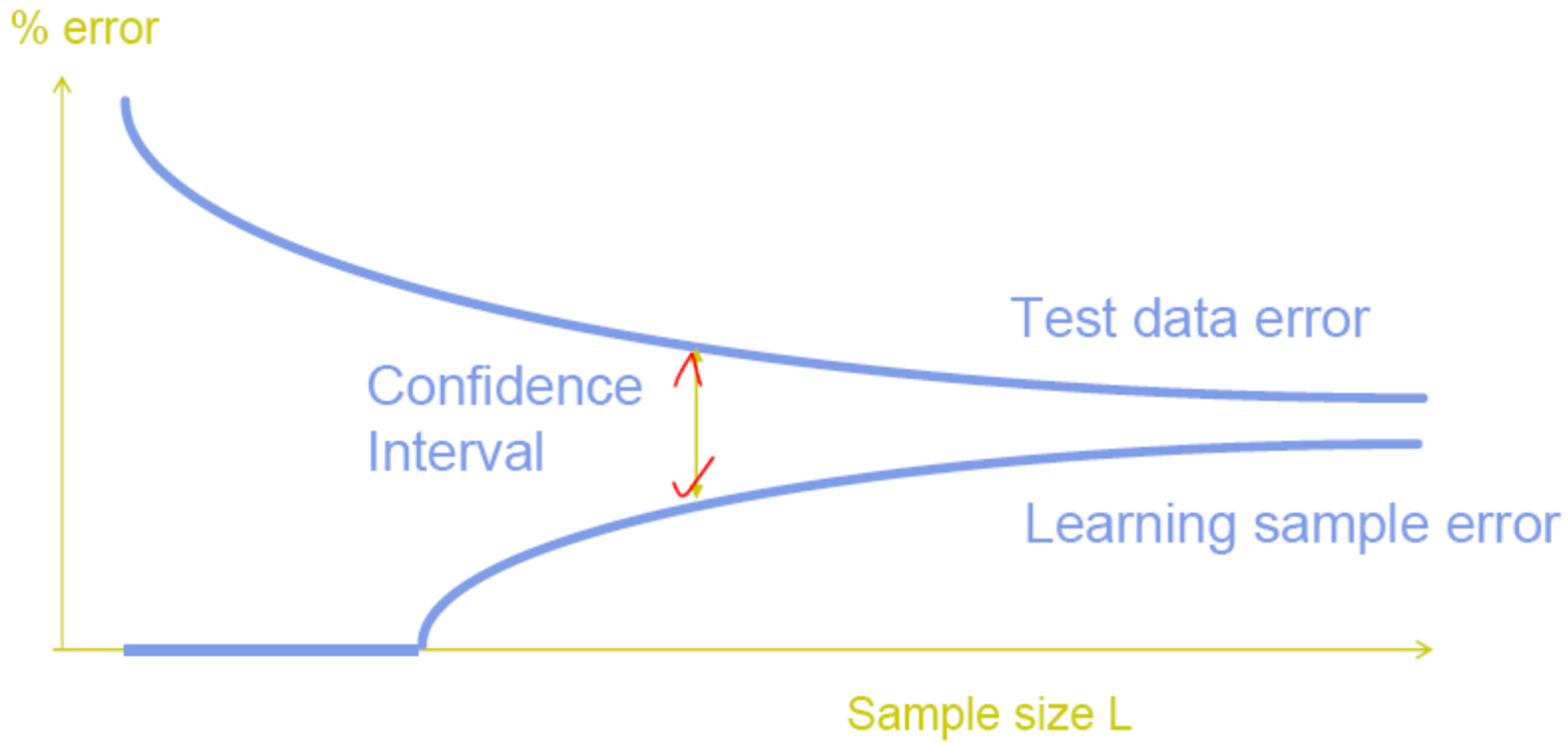
or

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

recall that in the finite H case, we have:

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

# Convergence Speed




- Confidence interval bound

$$|\widehat{\epsilon}(h) - \epsilon(h)| < O(\sqrt{\frac{d}{m}\log\frac{m}{d} - \frac{1}{m}\log\delta})$$

# How to Control Generalization Ability

- Risk Expectation = Empirical Risk + Confidence Interval
  - To minimize Empirical Risk alone will not always give a good generalization ability: 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” encoded in the VC dimension d

# Expected Risk Bound

- With probability *1-*$\delta$, the following inequality is true:

$$\int (y - f(x, w_0))^2 dP(x, y) <$$

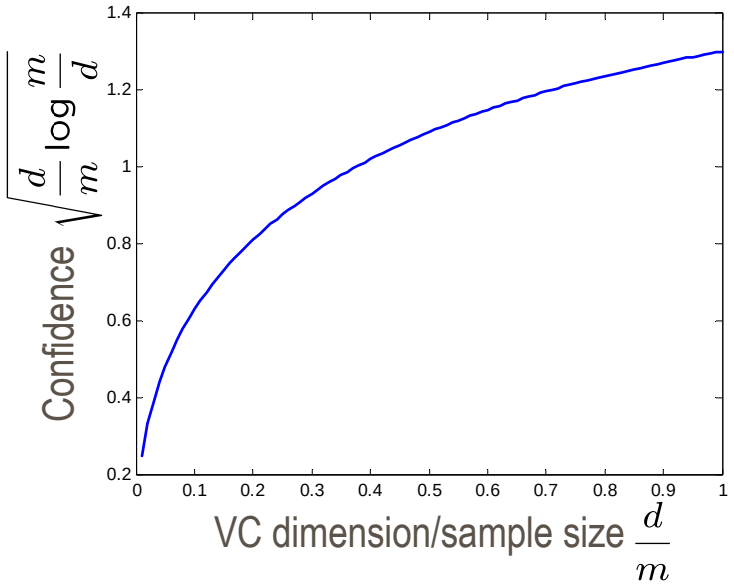
$$\frac{1}{m} \sum_{i=1}^{m} (y_i - f(x_i, w_o))^2 + \sqrt{\frac{d(\log \frac{2m}{d} + 1) - \ln \delta}{m}}$$

where $w_0$ is the parameter w value that minimizes Empirical Risk:

$$E(w) = \frac{1}{m} \sum_{i=1}^{m} (y_i - f(x_i, w_o))^2$$

# Minimizing the Bound by Minimizing d

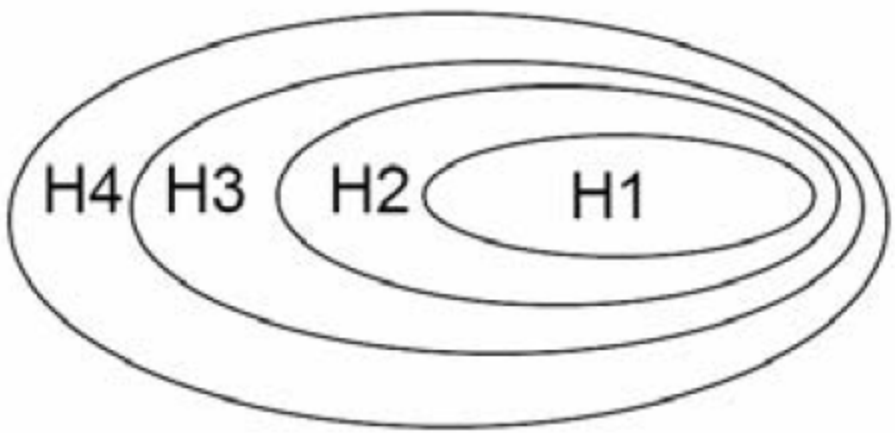
- Given some selection of learning machines whose empirical risk is zero, one wants to choose that learning machine whose associated set of functions has minimal VC dimension.
- Minimal VC dimension
  → small sample size for good generalization




- By doing this we can attain an upper bound on the actual risk. This does not prevent a particular machine with the same empirical risk, and whose function set has higher VC dimension, from having better performance.

# Structural Risk Minimization

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




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

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

unfortunately a somewhat loose bound...

# SRM Strategy

- Consider a sequence $H_1 \subset H_2 \subset \ldots \subset H_n$ 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 holds

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

  - 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.

# SRM for Linear Models

- In the case of linear models

$$y = wx + b$$

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

$$H_C : ||w|| < C$$

  - Margin is $1/w$

- Vapnik's Major theorem:

  When C decreases, d($H_C$) decreases (assuming ||x|| < R)

$$d(H_c) \leq 4R^2C^2 + 1$$

# SRM for Linear Models

To control ||w||, one can envision two ways:

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

$$RG(w,b) = \sum_i (y_i - wx_i - b)^2 + \lambda ||w||^2$$

- *Support Vector Machines (SVM),* i.e. solve directly an optimization problem (assume separable data)

*Minimize ||w||²,*

*with ($y_i$ = +/-1) and*

$$y_i(wx_i + b) \geq 1, \quad i = 1, ..., m$$

# VC Dimension of SVM

- *Mercer's Condition for Kernels*

  There exists a mapping $\phi(.)$ and an expansion

$$K(x,y) = \sum_i \phi_i(x)\phi_i(y)$$

  iff for any g(x) such that $\int g(x)^2 dx$ is finite, then

$$\int\int K(x,y)g(x)g(y)dxdy \geq 0$$

# VC Dimension of SVM

- We will call any kernel that satisfies Mercer's condition a positive kernel, and the corresponding space *H* the embedding space.
- We will also call any embedding space with minimal dimension for a given kernel a "minimal embedding space".

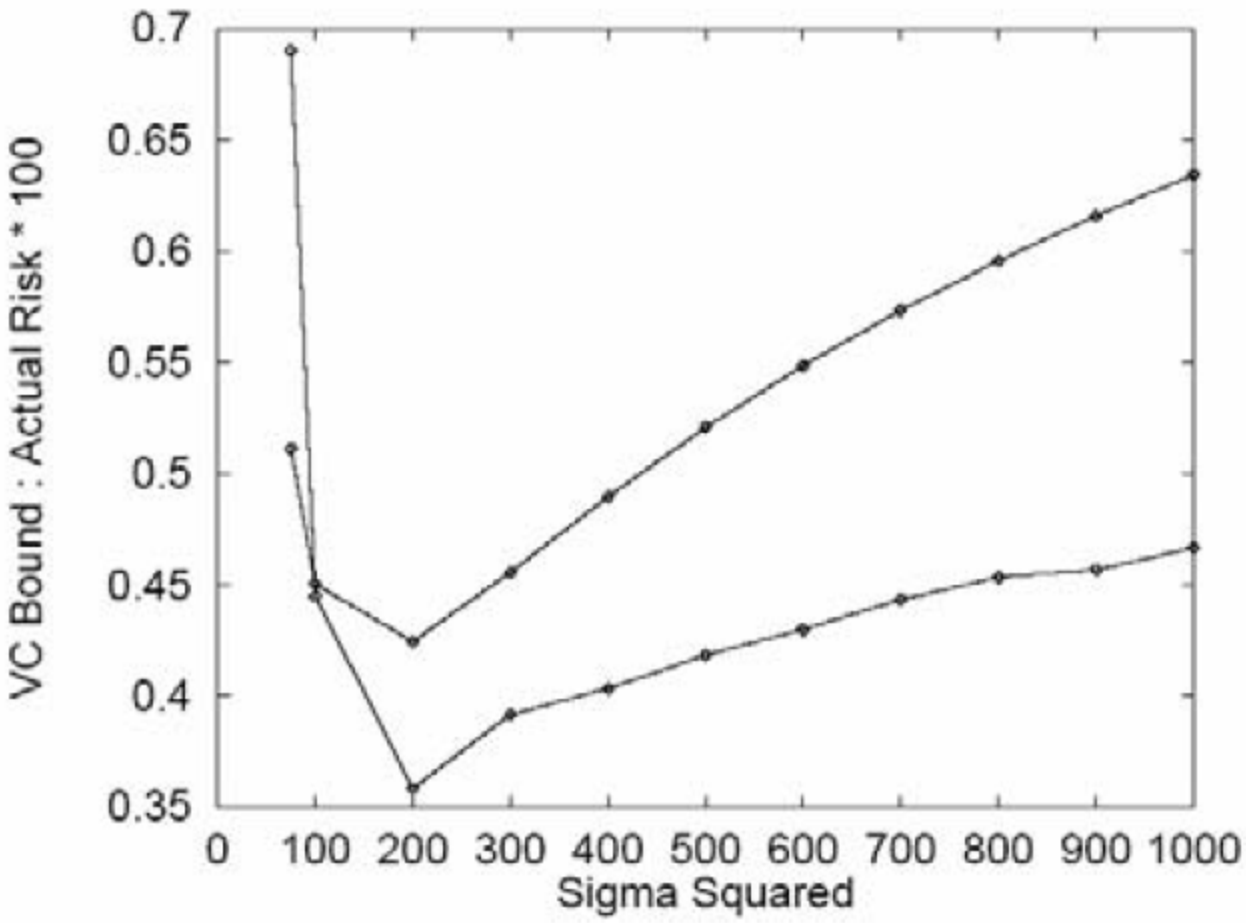
- Theorem: *Let K be a positive kernel that corresponds to a minimal embedding space H. Then the VC dimension of the corresponding SVM (where the error penalty C is allowed to take all values) is dim(H) + 1*
  - Can compute VC dimension of polynomial kernel SVM

# VC Dimensions

The VC dimension of a

- Linear classifier: d+1 in dimension d
- Circle in 2d = 3
- Neural networks = #parameters
- Decision Stumps = $d + 1$
- Decision Trees (fully grown) = $\infty$
- kNN = $\infty$
- SVM with Gaussian Kernel and arbitrary margin = $\infty$
- SVM with margin $\rho$ is $\leq \frac{4R^2}{\rho^2} + 1$ where $||x_i|| < R$

# Actual Risk and the VC Bound




- the VC bound, although loose, predicts the correct $\sigma$

# Feature Selection Bounds

- Better bounds can be obtained assuming sparsity
- Let dim $\Omega$ = M.

Theorem [Schapire 1998] Assume $\|\mathbf{x}\|_\infty \leq 1, \forall x \in \Omega$

Consider the family of hyperplanes w with nonnegative weights and $\|w\|_1 \leq 1$. Then there exists a constant C so that with probability 1-$\delta$ we have for all $\mu$

$$\epsilon(\mathbf{w}) \leq \frac{k_\mu}{m} + \sqrt{\frac{C}{m}\left(\frac{\ln M \ln m}{\mu^2} - \ln \delta\right)}$$

where $k_\mu = |\{i : \mathbf{w}^T \mathbf{x}^i y^i < \mu\}|$ is the number of samples with margin less than $\mu$.

# Feature Selection Bounds

- Let dim $\Omega$ = M.

Theorem [Kakade 2008] Let $\mu$>0 and

$$A = 4 \max_{\mathbf{x} \in \Omega} \|\mathbf{x}\|_\infty \max_{w} \|\mathbf{w}\|_1$$

then with probability 1-$\delta$ we have

$$\epsilon(\mathbf{w}) \leq \frac{k_\mu}{m} + \frac{1}{\sqrt{m}} \left( \frac{A}{\mu} \sqrt{2 \ln M} + \sqrt{\ln(\log_2 \frac{A}{\mu})} + \sqrt{\frac{-\ln \delta}{2}} \right)$$

where $k_\mu = |\{i : \mathbf{w}^T \mathbf{x}^i y^i < \mu\}|$ is the number of samples with margin less than $\mu$.

# Feature Selection vs VC Dimension

- VC dimension of axis aligned linear classifiers (decision stumps) $H$={ $\beta_0+\beta_1 x_i$, i=1...,M} is M+1
- Bounds from Vapnik's theorem $\approx O(\sqrt{\frac{M}{m}})$
- Bounds from sparsity $O(\sqrt{\frac{\ln M}{m}})$

Conclusion: Sparsity improves generalization error.

# Oracle Inequalities for the Lasso

- Lasso= $L_1$ penalized least squares
- Given $(x_j, y_j)$, j=1,…,N training examples

$$\mathbf{w} = \arg\min_{\mathbf{w}} \frac{1}{N} \sum_{j=1}^{N} (y_j - \mathbf{x}_j' \mathbf{w})^2 + \lambda \|\mathbf{w}\|_1$$

- Let M(w) be the number of nonzero entries in w

$$M(\mathbf{w}) = \|\mathbf{w}\|_0$$

# Oracle Inequalities for the Lasso

Let N= number of training examples, M= number of features

Let $k^*$=M(w*) be the true number of variables

Theorem (Bunea et al, 2007) There exist constant B,C,L,$L_0$,$k_M$, $\lambda_{N,M}$ such that (under many assumptions) we have with probability at least 1-$\pi_{N.M}$

$$|\widehat{\mathbf{w}} - \mathbf{w}^*|_1 \leq \frac{B\lambda_{N,M}}{k_M} k^*$$

where

$$\pi_{N,M} = 10M^2 \exp\left(-CN \min\left[\lambda_{N,M}^2, \frac{\lambda_{N,M}}{L}, \frac{1}{L^2}, \frac{k_M^2}{L_0 k^{*2}}, \frac{k_M}{L^2 k^*}\right]\right)$$

$$\lambda_{N,M} = O(\sqrt{\frac{\ln M}{N}})$$

# Oracle Inequalities

- Similar results for $L_1$ Penalized Logistic Regression (Bunea 2008)

- No oracle results for Boosting but
  - Oracle results for Regularized Boosting (Blanchard et al, 2003)
- Some oracle results for SVM (Steinwart et al, 2007)

# Summary

- Test Error <= Training error + error bound
  - Error Bounds in terms of VC dimension
  - VC dimension = measure of function space complexity
  - Sparsity improves error bound
- Structural Risk Minimization:
  - Minimize Training error + error bound
  - Can sometimes predict correct model parameters
- Alternative: Cross-validation
- Consistency: when the difference btw training and test error goes to zero when sample size goes to $\infty$
  - Consistent if and only if VC dimension is finite