Learning Theory

CE-717: Machine Learning

Sharif University of Technology

M. Soleymani

Fall 2016

Topics

- Feasibility of learning
- PAC learning
- VC dimension
- Structural Risk Minimization (SRM)

Feasibility of learning

- Does the training set $\mathcal{D}$ tell us anything out of $\mathcal{D}$?
  - $\mathcal{D}$ does <u>not tells</u> us something <u>certain</u> about f outside of $\mathcal{D}$
  - However, it can <u>tell</u> us something <u>likely</u> about f outside of $\mathcal{D}$

- Probability helps us to find learning theory

Feasibility of learning

- These two questions:
  - $\boxed{\text{Can we make sure } E_{true}(f) \text{ is close to } E_{train}(f)\text{?}}$
  - Can we make $E_{train}(f)$ small enough?

Generalizability of Learning

- Generalization error is important to us

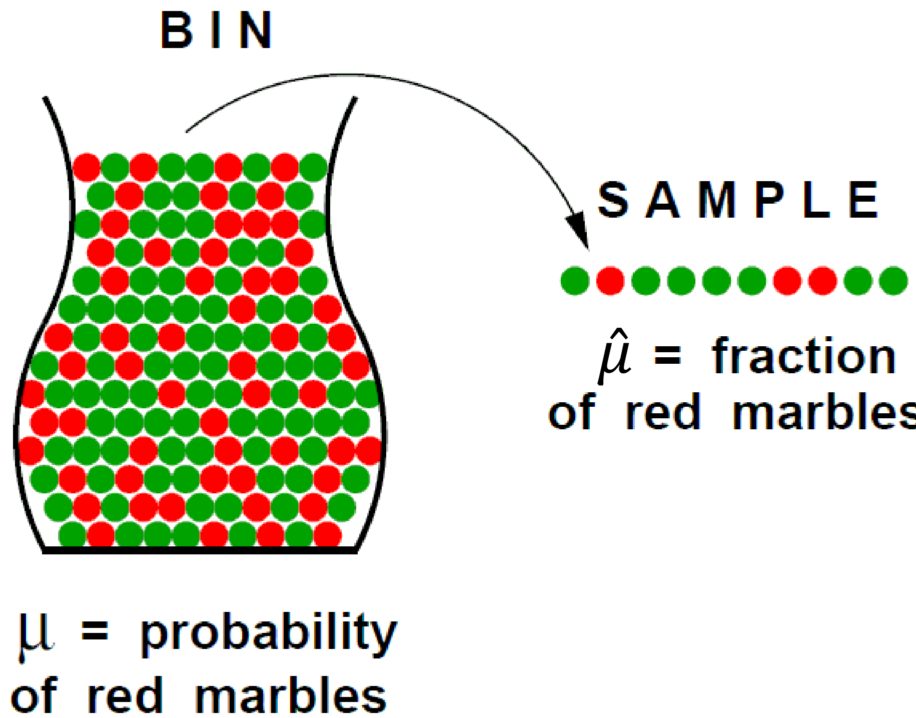
- Why should doing well on the training set tell us anything about generalization error?
  - Can we relate error on training set to generalization error?

- Which are conditions under which we can actually prove that learning algorithms will work well?

A related example

$\Pr[\text{picking a red marble}] = \mu$

$\Pr[\text{picking a green marble}] = 1 - \mu$



- Value of μ is **unknown** to us
- We pick N marbles independently
- The fraction of red marbles in sample $= \hat{\mu}$

Does $\hat{\mu}$ say anything about μ?

- No:
  - Samples can be mostly green while bin is mostly red
- Yes:
  - Sample frequency $\hat{\mu}$ is likely close to bin frequency μ

What does $\hat{\mu}$ say about μ?

- In a big sample (large N), ν is probably close to μ (within ϵ):

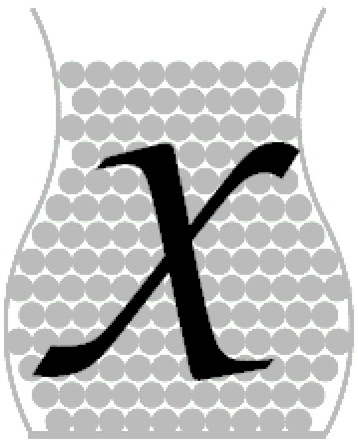
$$\Pr[|\hat{\mu} - \mu| > \epsilon] \leq 2e^{-2\epsilon^2 N}$$

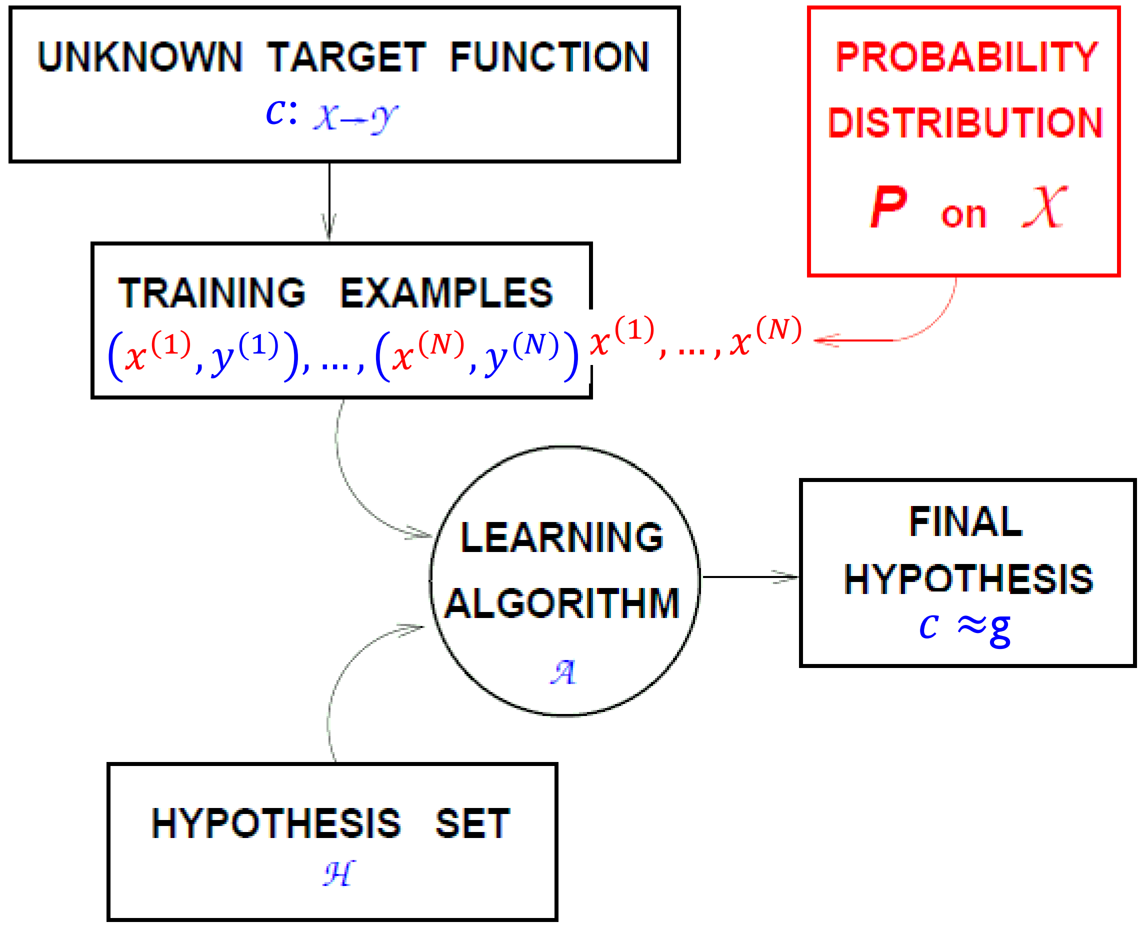
Hoeffding's Inequality

  - Valid for all N and ϵ
  - Bound does not depend on μ
  - Tradeoff: N, ϵ, and the bound

- In the other words, "$\hat{\mu} = \mu$" is Probably Approximately Correct (PAC)

Recall: Learning diagram





We assume that some random process proposes instances, and teacher labels them (i.e., instances drawn i.i.d. according to a distribution $P(\boldsymbol{x})$)

[Y.S. Abou Mostafa, et. al, "Learning From Data", 2012]

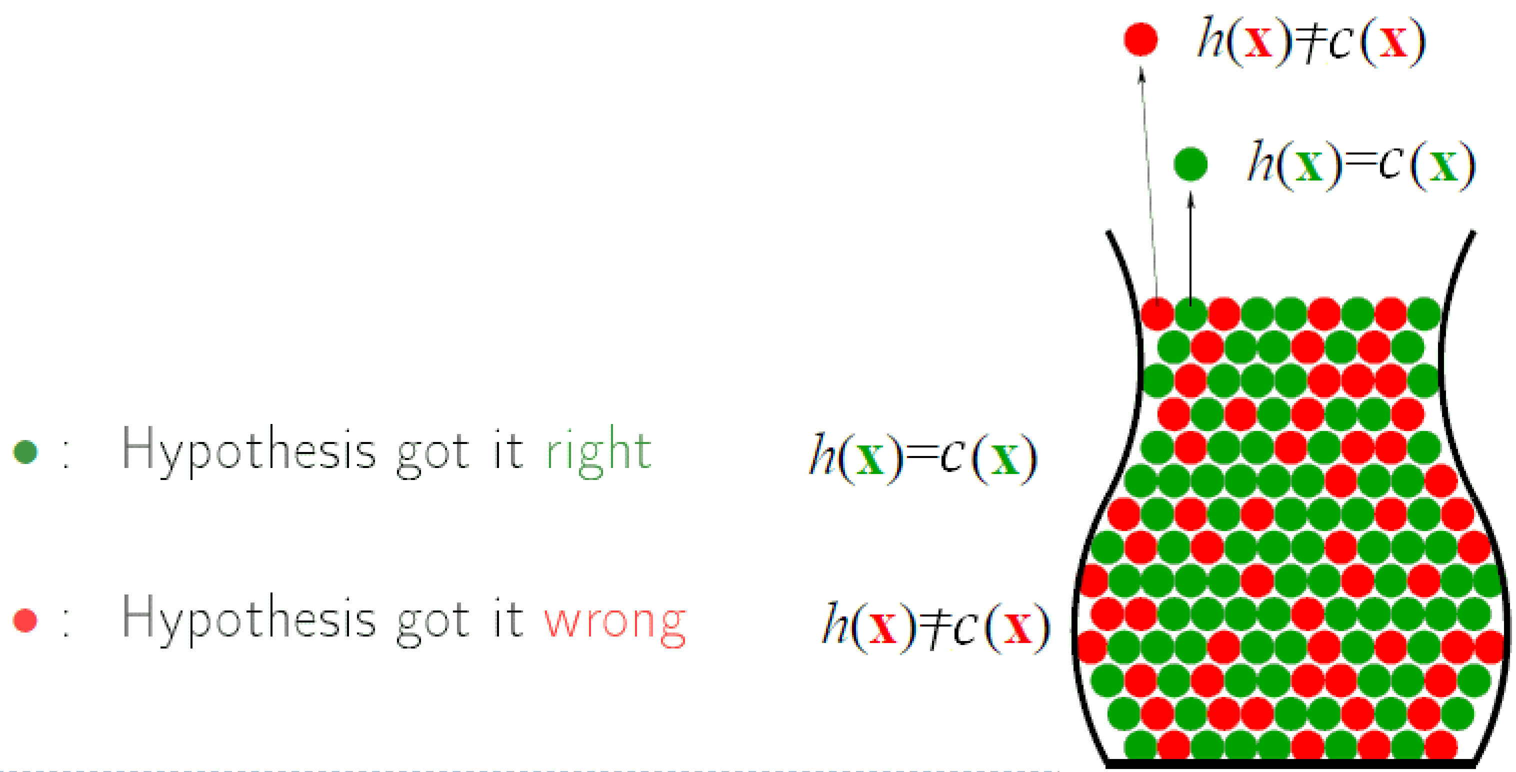
Learning: Problem settings

- Set of all instances $\mathcal{X}$
- Set of hypotheses $\mathcal{H}$
- Set of possible target functions $C = \{c: \mathcal{X} \to \mathcal{Y}\}$
- Sequence of N training instances $\mathcal{D} = \left\{\left(\boldsymbol{x}^{(n)}, c\left(\boldsymbol{x}^{(n)}\right)\right)\right\}_{n=1}^{N}$
  - $\boldsymbol{x}$ drawn at random from unknown distribution $P(\boldsymbol{x})$
  - Teacher provides **noise-free** label $c(\boldsymbol{x})$ for it

- **Learner** observes a set of training examples $\mathcal{D}$ for target function c and outputs a hypothesis $h \in \mathcal{H}$ estimating c

Connection of Hoeffding inequality to learning

- In the bin example, the unknown is μ
- In the learning problem the unknown is a function $c: \mathcal{X} \to \mathcal{Y}$



Two notions of error

- **Training error of *h***: how often $h(\boldsymbol{x}) \neq c(\boldsymbol{x})$ on training instances D

$$E_{train}(h) \equiv E_{\boldsymbol{x}\sim\mathcal{D}}[I(h(\boldsymbol{x}) \neq c(\boldsymbol{x}))]$$
$$= \frac{1}{|\mathcal{D}|}\sum_{\boldsymbol{x}\in\mathcal{D}} I\big(h(\boldsymbol{x}) \neq c(\boldsymbol{x})\big)$$

Training data

- **Test error of *h***: how often $h(\boldsymbol{x}) \neq c(\boldsymbol{x})$ over future instances drawn at random from $P(X)$

$$E_{true}(h) \equiv E_{\boldsymbol{x}\sim P(X)}[I(h(\boldsymbol{x}) \neq c(\boldsymbol{x}))]$$

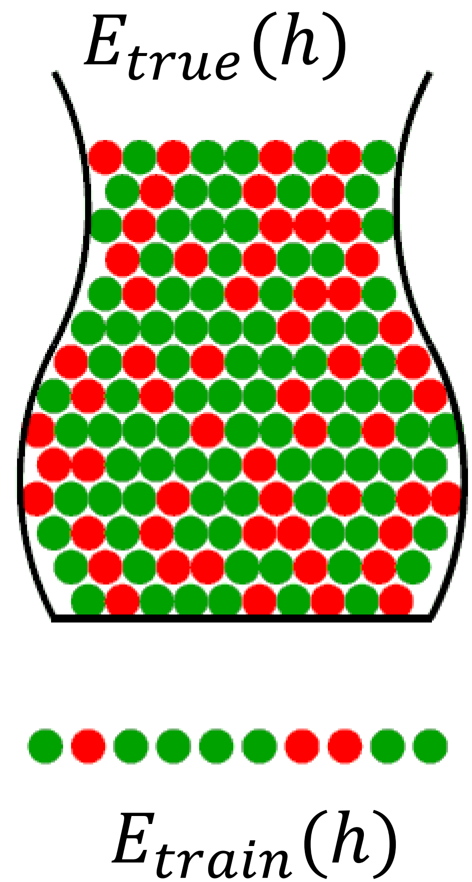
Probability distribution

Notation for learning

- Both μ and $\hat{\mu}$ depend on which hypothesis h
- $\hat{\mu}$ is "in sample" denoted by $E_{train}(h)$
- μ is "out of sample" denoted by $E_{true}(h)$

- The Hoeffding inequality becomes:

$$\Pr[|E_{train}(h) - E_{true}(h)| > \epsilon] \leq 2e^{-2\epsilon^2 N}$$



Are we done?

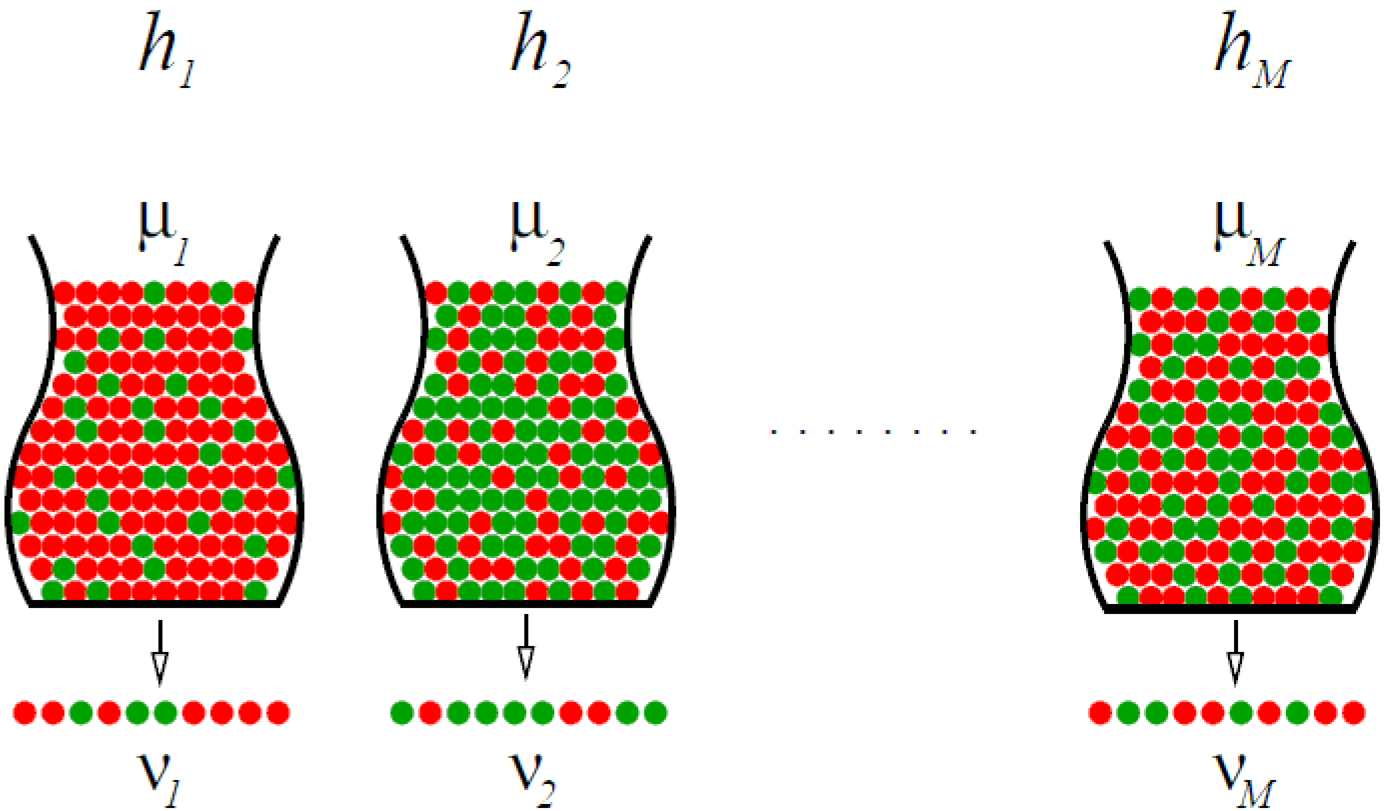
- We cannot use this bound for the learned f from data.

- Indeed, h is assumed fixed in this inequality and for this h, $E_{train}(h)$ generalizes to $E_{true}(h)$.
  - "verification" of h, not learning

- We need to choose from multiple h's and f is not fixed and instead is found according to the samples.

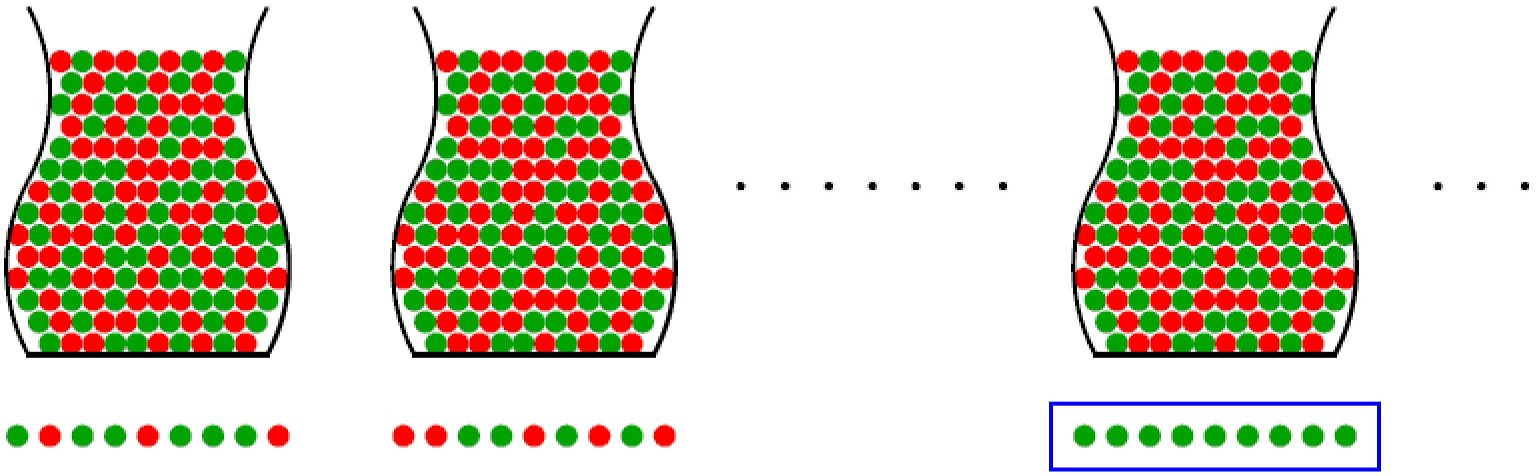
Hypothesis space as multiple bins

- Generalizing the bin model to more than one hypothesis:



Hypothesis space: Coin example

- Question: if you toss a fair coin 10 times, what is the probability that it will get 10 heads?
  - Answer: ≈ 0.1%
- Question: if you toss 1000 fair coins 10 times, what is the probability that some of them will get 10 heads?
  - Answer: ≈ 63%



A bound for the learning problem: Using Hoeffding inequality

$$\Pr[|E_{true}(f) - E_{train}(f)| > \epsilon]$$

$$\leq \Pr\begin{bmatrix} |E_{true}(h_1) - E_{train}(h_1)| > \epsilon \\ \text{or } |E_{true}(h_2) - E_{train}(h_2)| > \epsilon \\ \dots \\ \text{or} |E_{true}(h_M) - E_{train}(h_M)| > \epsilon \end{bmatrix}$$

$$\leq \sum_{i=1}^{M} \Pr[|E_{true}(h_i) - E_{train}(h_i)| > \epsilon]$$

$$\leq \sum_{i=1}^{M} 2e^{-2\epsilon^2 N}$$

$$\leq 2|\mathcal{H}|e^{-2\epsilon^2 N}$$

$|\mathcal{H}| = M$

PAC bound: Using Hoeffding inequality

$$\Pr[|E_{true}(h) - E_{train}(h)| > \epsilon] \leq 2|\mathcal{H}|e^{-2\epsilon^2 N}$$

$$\Rightarrow \Pr[|E_{true}(h) - E_{train}(h)| \leq \epsilon] \geq 1 - \delta$$

- With probability at least $(1 - \delta)$ every h satisfies

$$E_{true}(h) < E_{train}(h) + \sqrt{\frac{\ln 2|\mathcal{H}| + \ln\frac{1}{\delta}}{2N}}$$

Thus, we can we bound $E_{true}(h) - E_{train}(h)$ that shows the amount of overfiting

Sample complexity

- How many training examples suffice?
  - Given ϵ and δ, yields sample complexity:

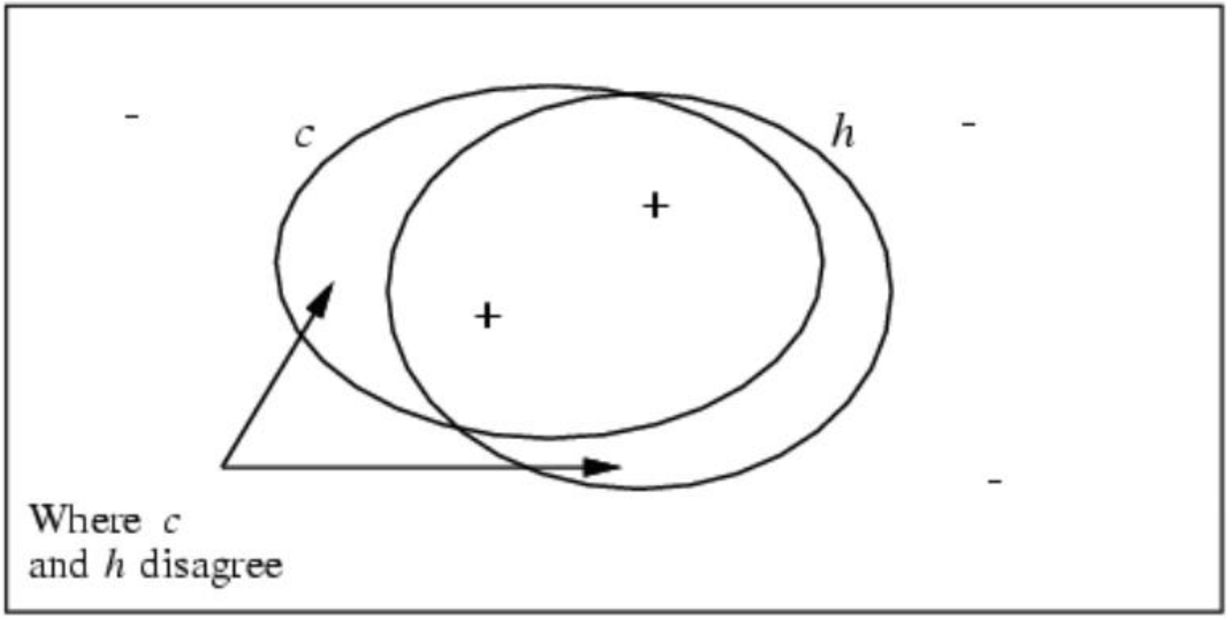
$$N \geq \frac{1}{2\epsilon^2}\left(\ln 2|\mathcal{H}| + \ln\left(\frac{1}{\delta}\right)\right)$$

- Thus, we found a theory that relates
  - Number of training examples
  - Complexity of hypothesis space
  - Accuracy to which target function is approximated
  - Probability that learner outputs a successful hypothesis

An other problem setting

- Finite number of possible hypothesis (e.g., decision trees of depth d_0)
- A learner finds a hypothesis h that is consistent with training data
  - $E_{train}(h) = 0$
- What is the probability that the true error of h will be more than ϵ?
  - $E_{true}(h) \geq \epsilon$

True error of a hypothesis



- *True error of h*: probability that it will misclassify an example drawn at random from $P(\boldsymbol{x})$

$$E_{true}(h) \equiv E_{\boldsymbol{x}\sim P(X)}[I(h(\boldsymbol{x}) \neq c(\boldsymbol{x}))]$$

How likely is a consistent learner to pick a bad hypothesis?

Bound on the probability that <u>any consistent learner</u> will output h with $E_{true}(h) > \epsilon$

<u>Theorem [Haussler, 1988]</u>: For target concept $c, \forall\ 0 \leq \epsilon \leq 1$

If H is finite and $\mathcal{D}$ contains $N \geq 1$ independent random samples

$$\Pr[\exists h \in \mathcal{H}, E_{train}(h) = 0 \wedge E_{true}(h) > \epsilon]$$

$$\leq |\mathcal{H}| e^{-\epsilon N}$$

Haussler bound: Proof

- What does the theorem mean?

$$\Pr[\exists h \in \mathcal{H}, E_{train}(h) = 0 \wedge E_{true}(h) > \epsilon]$$

$$\leq |\mathcal{H}|e^{-\epsilon N}$$

- For a fixed h, how likely is a bad hypothesis (i.e., $E_{true}(h) > \epsilon$) to label N training data points right?
  - $\Pr(h \text{ labels one data point correctly} | E_{true}(h) > \epsilon) \leq (1 - \epsilon)$
  - $\Pr(h \text{ labels } N \text{ i.i.d data points correctly} | E_{true}(h) > \epsilon) \leq (1 - \epsilon)^N$

Haussler bound: Proof (Cont'd)

- There may be many bad hypotheses $h_1, \dots, h_k$ (i.e., $E_{test}(h_1) > \epsilon$, …, $E_{test}(h_k) > \epsilon$) that are consistent with N training data
  - $E_{train}(h_1) = 0, E_{train}(h_2) = 0, \dots, E_{train}(h_k) = 0$
- How likely is the learner pick a bad hypothesis $(E_{test}(h) > \epsilon)$ among consistent ones $\{h_1, \dots, h_k\}$?

$\Pr(\exists h \in H, E_{true}(h) > \epsilon \wedge E_{train}(h) = 0)$

$= \Pr\big((E_{true}(h_1) > \epsilon \wedge E_{train}(h_1) = 0) \text{ or } \dots \text{ or } (E_{true}(h_k) > \epsilon \wedge E_{train}(h_k) = 0)\big)$

$\leq \sum_{i=1}^{k} \Pr(E_{train}(h_i) = 0 \ \wedge E_{true}(h_i) > \epsilon)$ $\quad [P(A \cup B) \leq P(A) + P(B)]$

$\leq \sum_{i=1}^{k} \Pr(E_{train}(h_i) = 0 | E_{true}(h_i) > \epsilon) \leq \sum_{i=1}^{k} (1-\epsilon)^N$

$\leq |\mathcal{H}|(1-\epsilon)^N$ $\quad [k \leq |\mathcal{H}|]$

$\leq |\mathcal{H}| e^{-\epsilon N}$ $\quad [1 - \epsilon \leq e^{-\epsilon} \quad 0 \leq \epsilon \leq 1]$

Haussler PAC Bound

- Theorem [Haussler'88]: Consider finite hypothesis space H, training set D with m i.i.d. samples, $0 < \epsilon < 1$:

$$\Pr[\exists h \in \mathcal{H}, E_{train}(h) = 0 | E_{true}(h) > \epsilon] \leq |\mathcal{H}| e^{-\epsilon N} \leq \delta$$

Suppose we want this probability to be at most δ.

- For any learned hypothesis $h \in \mathcal{H}$ that is consistent on the training set $\mathcal{D}$ (i.e., $E_{train}(h) = 0$), with probability at least $(1 - \delta)$:

$$E_{true}(h) \leq \epsilon$$

Haussler PAC bound: Sample complexity

- How many training examples suffice?
  - Given ϵ and δ, yields sample complexity:

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

There are enough training examples to guarantee that any consistent hypothesis has error at most ϵ with probability $1-\delta$.

- Given N and δ, yields error bound:

$$\epsilon \leq \frac{1}{N}\left(\ln|\mathcal{H}| + \ln\left(\frac{1}{\delta}\right)\right)$$

Error bound linear in $\frac{1}{N}$ and only logarithmic in $|\mathcal{H}|$.

Example: Conjunction of up to d Boolean literals

- Consider a Boolean classification problem $c: \mathcal{X} \rightarrow \mathcal{Y}$

- Hypothesis space: rules that are in the form of conjunction of up to d Boolean literals
  - Example: ($d = 5$ boolean features)

    if $\boldsymbol{x} = \underbrace{[0\ ?\ 1\ ?\ ?]}_{\neg x_1 \wedge x_3}$ then $y = 1$ else $y = 0$

- How many training examples N?

  "Any consistent learner using $\mathcal{H}$ with probability ≥ 0.99 will output a hypothesis with $E_{true} \leq 0.05$"?

  $d = 5 \Rightarrow N > 201$

  $d = 10 \Rightarrow N > 312$

  $d = 100 \Rightarrow N > 2290$

  $\delta = 0.01$
  $\epsilon = 0.05$
  $|\mathcal{H}| = 3^d$

Example: decision trees of limited depth

- Consider a Boolean classification problem
  - instances: vectors of d boolean features

- Hypothesis space: decision trees of depth 2

- How many training examples m?
  "Any consistent learner using $\mathcal{H}$ with probability $\geq$ 0.99 will output a hypothesis with $E_{true} \leq 0.05$"?

$$d = 4 \Rightarrow N > 219$$
$$d = 10 \Rightarrow N > 281$$
$$d = 100 \Rightarrow N > 423$$
$$d = 1000 \Rightarrow N > 562$$

$$\delta = 0.01$$
$$\epsilon = 0.05$$
$$|\mathcal{H}| = 16 \times d \times (d-1)^2$$

Limitations of Haussler'88 bound

- There are consistent classifiers in the hypothesis space: h such that $E_{train}(h) = 0$

- Dependence on the size of hypothesis space:
  - What if $|\mathcal{H}|$ is too big or $\mathcal{H}$ is continuous?

Limitation of the bounds

- Until now, we find bounds for two cases:
  - Haussler's bound with the assumption $\exists h \in \mathcal{H}, E_{train}(h) = 0$
  - Hoeffding's bound

- If $\mathcal{H} = \{h \mid h: \mathcal{X} \to \mathcal{Y}\}$ is infinite,
  - We seek a measure of complexity instead of $|\mathcal{H}|$?
    - The largest subset of $\mathcal{X}$ for which $\mathcal{H}$ can guarantee zero training error (regardless of the target function)
    - **VC dimension** of $\mathcal{H}$ is the size of this subset

Definitions

- **Dichotomy**:
  - An N-tuple of ± 1 assigned to samples $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)} \in \mathcal{X}$

- The dichotomies generated by $\mathcal{H}$ on the data points $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}$:

$$\mathcal{H}\left(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\right) = \left\{h\left(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\right) \middle| h \in \mathcal{H}\right\}$$

- The **growth function** of a hypothesis set $\mathcal{H}$ is defined as:

$$m_{\mathcal{H}}(N) = \max_{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)} \in \mathcal{X}} \left|\mathcal{H}\left(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\right)\right|$$

Shattering a set of instances

$$m_{\mathcal{H}}(N) \leq 2^N$$

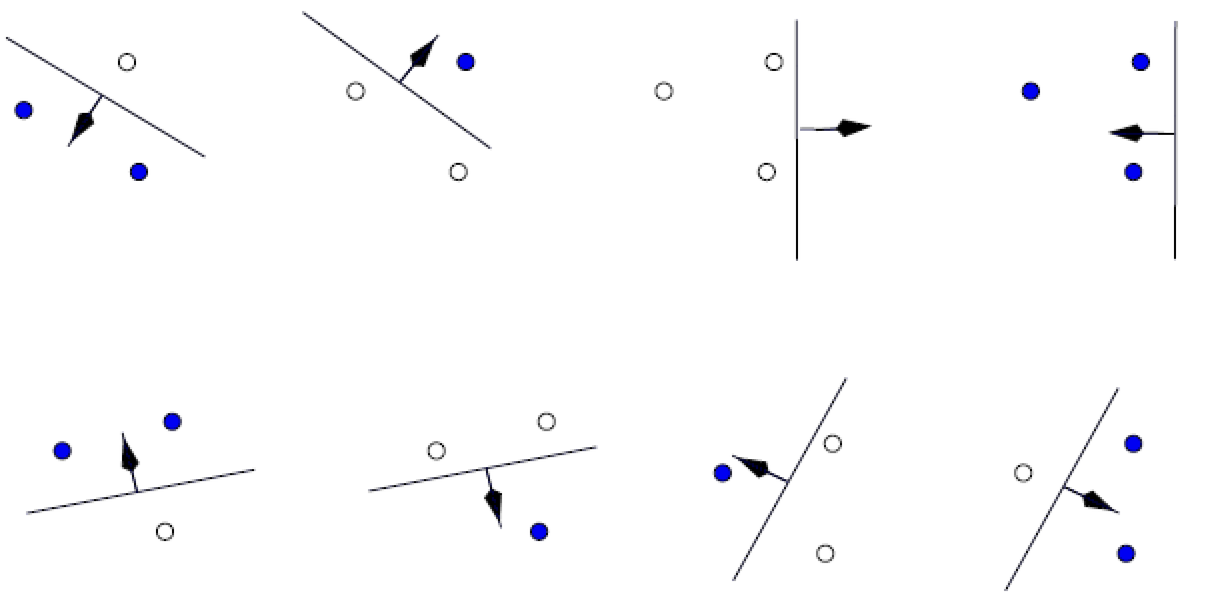
- A set $\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)}$ is **shattered** by $\mathcal{H}$ iff for every labeling of these samples there exists some hypotheses in $\mathcal{H}$ consistent with this labeling
  - (i.e., there exist hypotheses in $\mathcal{H}$ that can realize this labeling)

$$m_{\mathcal{H}}(N) = 2^N$$

- $\mathcal{H}$ is as diverse as can be on this particular sample.

Perceptron in a 2-dim feature space

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



Polynomial bound on $m_{\mathcal{H}}(k)$

- **Break point**: If no data set of size k can be shattered by $\mathcal{H}$, then k is said to be a break point for $\mathcal{H}$.

$$m_{\mathcal{H}}(k) < 2^k$$

- We can bound $m_{\mathcal{H}}(k)$ for all values of N by a simple polynomial based on this break point.

- **Theorem**: If $m_{\mathcal{H}}(k) < 2^k$ for some value k, then:

$$m_{\mathcal{H}}(N) \leq \sum_{i=0}^{k-1} \binom{N}{i}$$

Sauer's Lemma

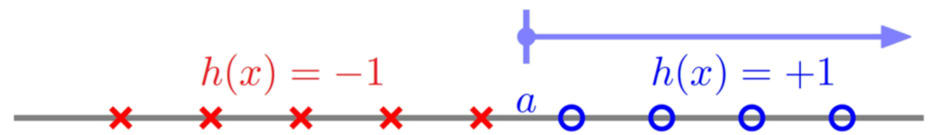
Maximum power is N^{k-1}

Break point: Example

- Example: None of 4 points can be shattered by the two-dimensional perceptron
  - This puts a significant constraint on # of dichotomies that can be realized by the perceptron on 5 or more points.

Growth function example: 1-D intervals

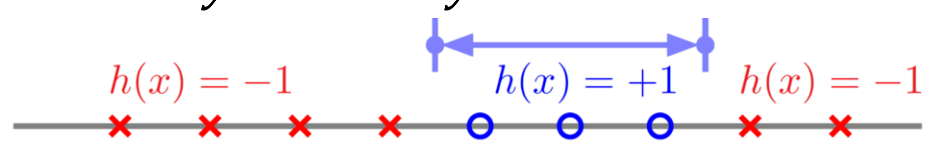
- $c: x \rightarrow \{0,1\}$
- What is VC dimension of:
  - Positive rays:



    - H1(open intervals to right):
      if $x > a$ then $y = 1$ else $y = 0$

$$m_{H_1}(N) = N + 1$$

  - Positive intervals:
    - H2 (inside intervals): if $a < x < b$ then $y = 1$ else $y = 0$



$$m_{H_2}(N) = \binom{N+1}{2} + 1$$

Generalization bound using growth function

$$\Pr[|E_{true}(h) - E_{train}(h)| > \epsilon] \leq 4 m_{\mathcal{H}}(2N) e^{-\frac{1}{8}\epsilon^2 N}$$

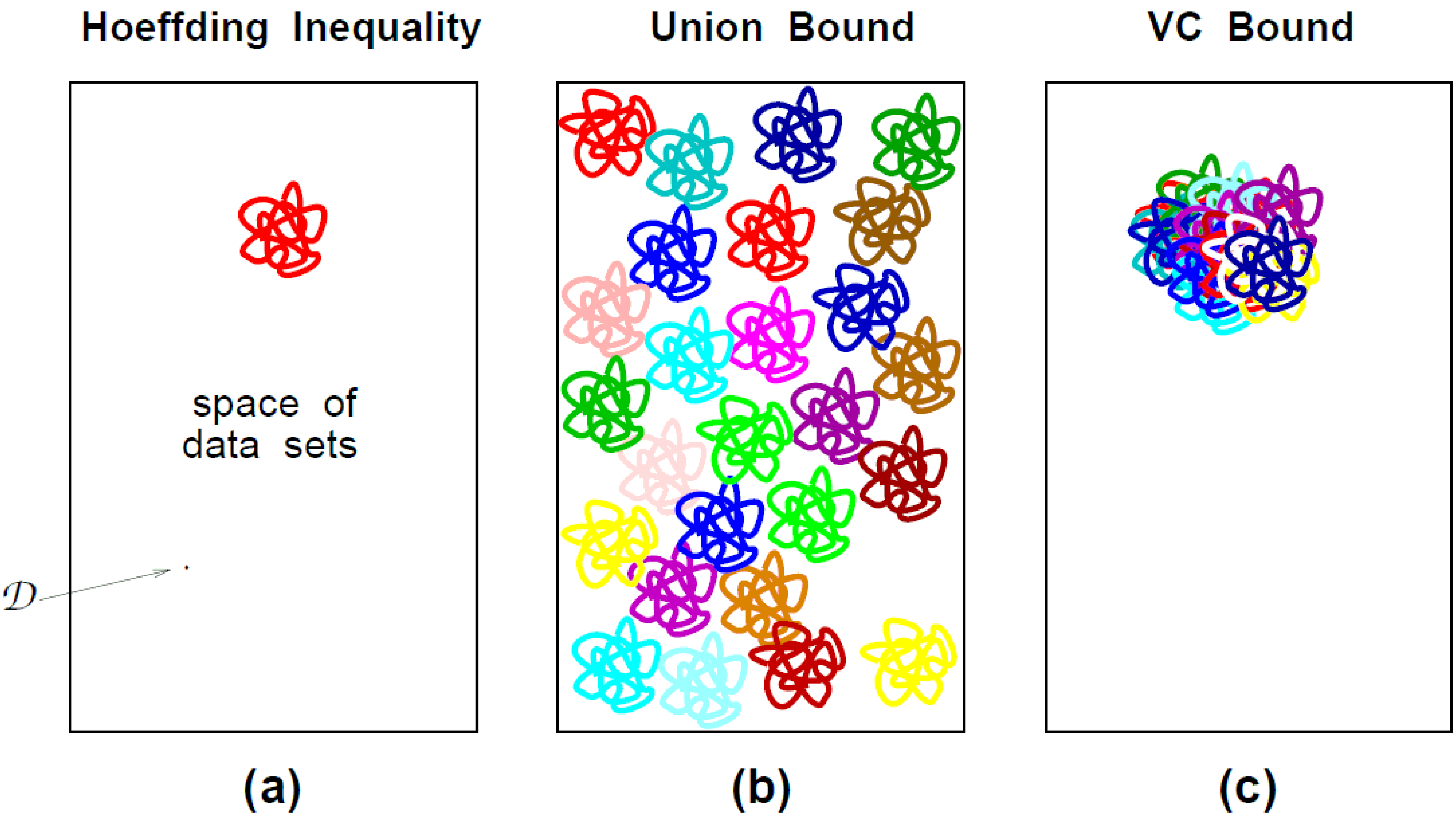
Vapnik-Chervonenkis inequality

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

$$E_{true} \leq E_{train} + \sqrt{\frac{8 \ln m_{\mathcal{H}}(2N) + 8 \ln \frac{4}{\delta}}{N}}$$

- In many cases, this bounds will be tighter than the previous bound for finite hypothesis spaces too.

$m_{\mathcal{H}}(N)$ relates to overlaps



Vapnick-Chervonenkis (VC) dimension

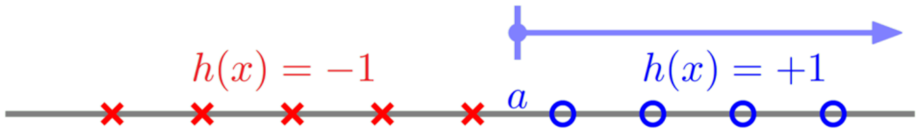
- The smaller break point, the tighter bound

- **Vapnik-Chervonenkis** $VC(\mathcal{H})$: the size of the largest set of samples that can be shattered by $\mathcal{H}$.
  - $VC(\mathcal{H})$ is the largest value of N for which $m_{\mathcal{H}}(N) = 2^N$

- In order to prove that $VC(\mathcal{H})$ is k:
  - There's at least one set of size k that $\mathcal{H}$ can shatter.
  - And there is no set of $k+1$ points that can be shattered.
    - for all $k+1$ points, there exists a labeling that cannot be shattered

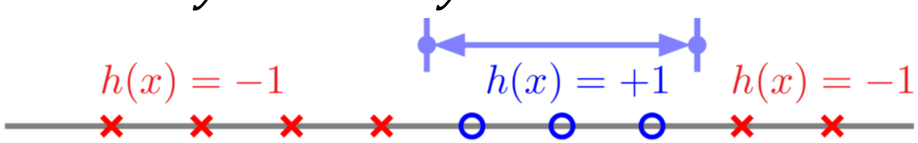
VC dimension: 1-D intervals

- $c: X \to \{0,1\}$
- What is VC dimension of:
  - Positive rays:
    - H1(open intervals to right):
      if $x > a$ then $y = 1$ else $y = 0$



$$VC(H_1) = 1$$
$$m_{H_1}(N) = N + 1$$

  - Positive intervals:
    - H2 (inside intervals): if $a < x < b$ then $y = 1$ else $y = 0$



$$VC(H_2) = 2$$
$$m_{H_2}(N) = \binom{N+1}{2} + 1$$

Bound on $m_{\mathcal{H}}(k)$ using VC

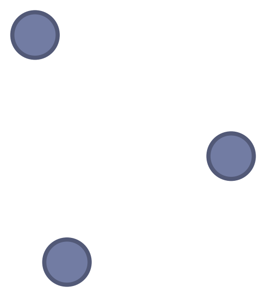
- Since $k = VC(\mathcal{H}) + 1$ is a break point for $m_{\mathcal{H}}(N)$:

$$m_{\mathcal{H}}(N) \leq \sum_{i=0}^{VC(\mathcal{H})} \binom{N}{i}$$

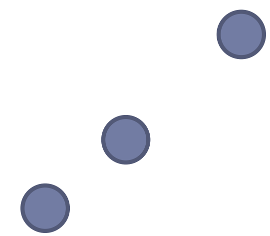
$$\sum_{i=0}^{k} \binom{N}{i} \leq N^{k} + 1$$

$$\Rightarrow m_{\mathcal{H}}(N) \leq N^{VC(\mathcal{H})} + 1$$

VC dimension: Perceptron in a 2-D space



Can be shattered by linear boundaries

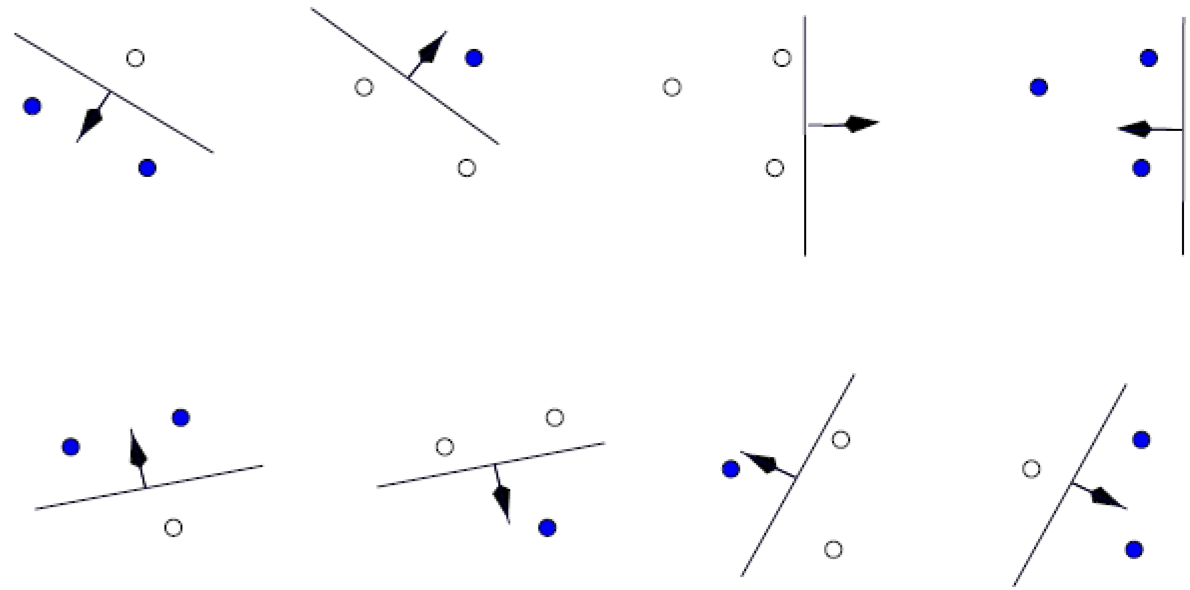


Cannot be shattered by linear boundaries

However, we seek the set of points with the most possible dichotomies

VC dimension: Perceptron in a 2-D space

- $VC(H) \geq 3$



- None of 4 points in a 2-D space can be shattered by perceptron
  - $VC(H) \leq 3$

$$\Rightarrow VC(H) = 3$$

VC of Perceptron

- $d = 2 \Longrightarrow VC = 3$

- In general $VC = d + 1$

Perceptron for d dimensional inputs

For $\mathcal{H}_d$ = linear separating hyper-planes in d dimensions, $VC(\mathcal{H}_d) = d + 1$

The following is a set of $N = d + 1$ samples in $\mathbb{R}^d$ that can be shattered by perceptron

$$\boldsymbol{X} = \begin{bmatrix} 1 & 0 & 0 \cdots & 0 \\ 1 & 1 & 0 \cdots & 0 \\ 1 & 0 & 1 \dots & 0 \\ \vdots & & \ddots & 0 \\ 1 & 0 & 0 \dots & 1 \end{bmatrix}$$

$\boldsymbol{X}$ is invertible

Perceptron for d dimensional inputs: Can we shatter this dataset?

- For any $\boldsymbol{y} = \begin{bmatrix} y^{(1)} \\ \vdots \\ y^{(d+1)} \end{bmatrix} = \begin{bmatrix} \pm 1 \\ \vdots \\ \pm 1 \end{bmatrix}$ can a vector $\boldsymbol{w}$ be found that correctly classifies all the data points:

For $\boldsymbol{w} = \boldsymbol{X}^{-\mathbf{1}}\boldsymbol{y}$ we have $\boldsymbol{X}\boldsymbol{w} = \boldsymbol{y}$ and thus $\text{sign}(\boldsymbol{X}\boldsymbol{w}) = \boldsymbol{y}$

Perceptron for d dimensional inputs

- So far we show that, we can shatter these $d + 1$ data points, thus we have $VC(\mathcal{H}) \geq d + 1$

- We also need to show that, we cannot shatter any set of $d + 2$ to prove that $VC(\mathcal{H}) = d + 1$

For any $d+2$ points

- $\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(d+1)}, \boldsymbol{x}^{(d+2)}$

- Since we have more points than dimensions, thus:

$$\exists m, \boldsymbol{x}^{(m)} = \sum_{n \neq m} a_n \boldsymbol{x}^{(n)}$$

where not all the a_n's are zero

For any $d + 2$ points, we cannot reach all dichotomies

$$\boldsymbol{x}^{(m)} = \sum_{n \neq m} a_n \boldsymbol{x}^{(n)}$$

$$\Rightarrow \boldsymbol{w}^T \boldsymbol{x}^{(m)} = \sum_{n \neq m} a_n \boldsymbol{w}^T \boldsymbol{x}^{(n)}$$

- If $y^{(n)} = \text{sign}\left(\boldsymbol{w}^T \boldsymbol{x}^{(n)}\right) = \text{sign}(a_i)$ then:

$$a_n \boldsymbol{w}^T \boldsymbol{x}^{(n)} > 0$$

- This forces $\boldsymbol{w}^T \boldsymbol{x}^{(m)} = \sum_{n \neq m} a_n \boldsymbol{w}^T \boldsymbol{x}^{(n)} > 0$
- Therefore, $\mathrm{y}^{(\mathrm{m})} = \text{sign}\left(\boldsymbol{w}^T \boldsymbol{x}^{(m)}\right) = +1$

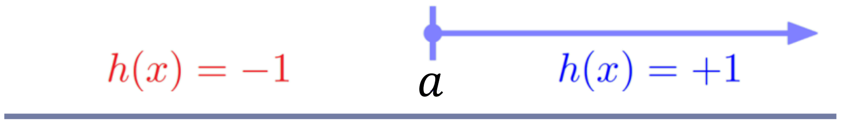
VC of perceptron in d-dimensional space

- We showed that $VC \geq d+1$ and $VC \leq d+1$ thus $VC = d+1$

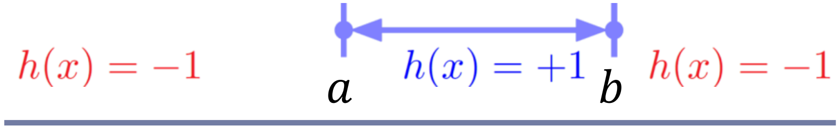
- In Perceptron the VC is the number of parameters $(w_0, w_1, \dots, w_d)$

Other examples

- Positive rays



- Positive intervals



VC dimension as degrees of freedom

- Parameters creates degrees of freedom

- VC as effective degrees of freedom
  - How expressive is this model
  - Not just the # of parameters
  - The effective number of parameters

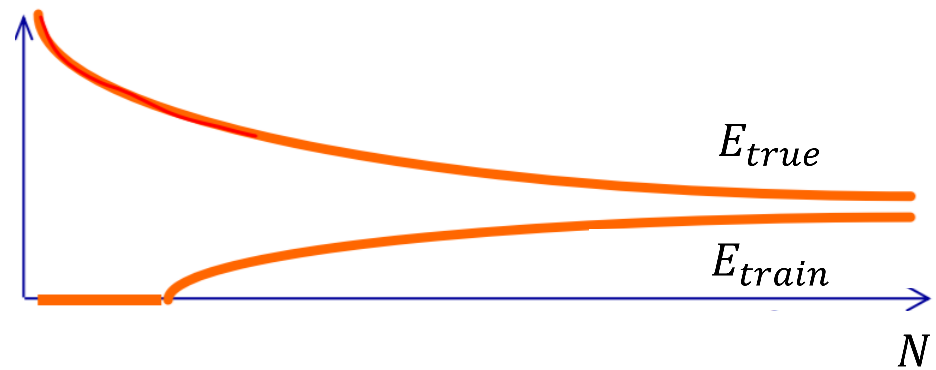
$VC(H) = \infty$

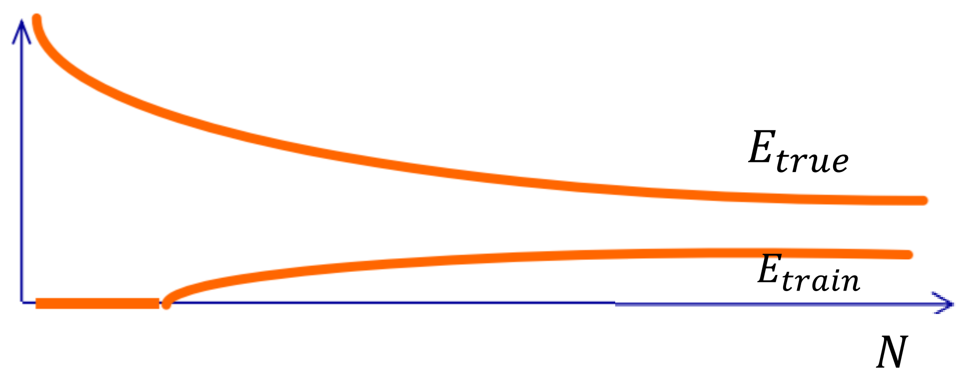
- If $\underline{m_{\mathcal{H}}(N) = 2^N \text{ for all } N}$ then $\underline{VC(H) = \infty}$

- If $VC(H) = \infty$ then no matter how large the data set is, we cannot make generalization conclusions based on the VC analysis.

Consistent learning

- E_{true} converges E_{train} when N increases





Vapnik main theorem

- A model is consistent <u>if and only if</u> the H has finite VC dimension

- A finite VC dimension not only guarantees consistency, but this is the only way to build a model that generalizes.

Main result

- No break point $\Longrightarrow m_{\mathcal{H}}(N) = 2^N$
- Any break point$\Longrightarrow m_{\mathcal{H}}(N)$ is polynomial in N

- Finite $VC(\mathcal{H}) \Rightarrow f \in \mathcal{H}$ will generalize

VC dimension and learning

- Independent of learning algorithm
- Independent of target function
- Independent of input distribution

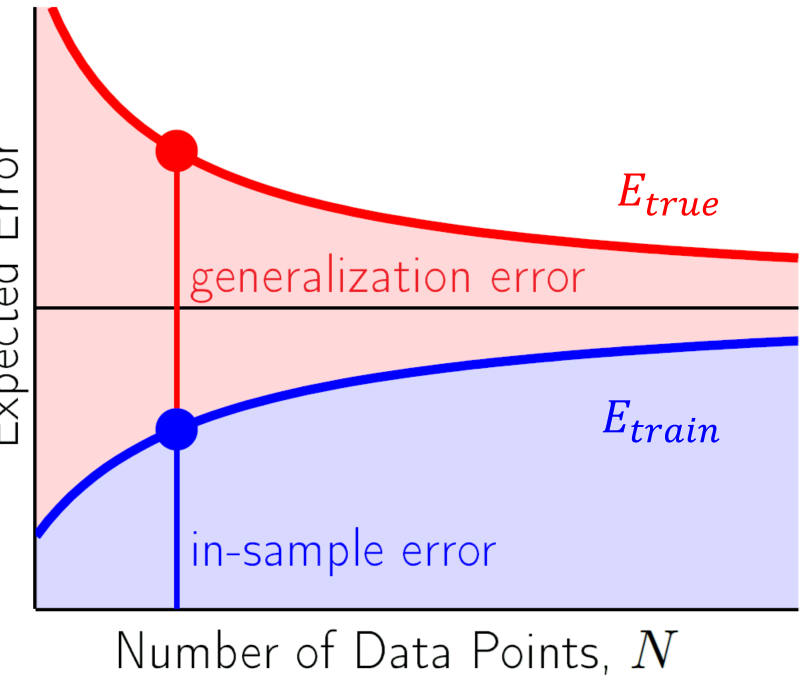
Practical issues

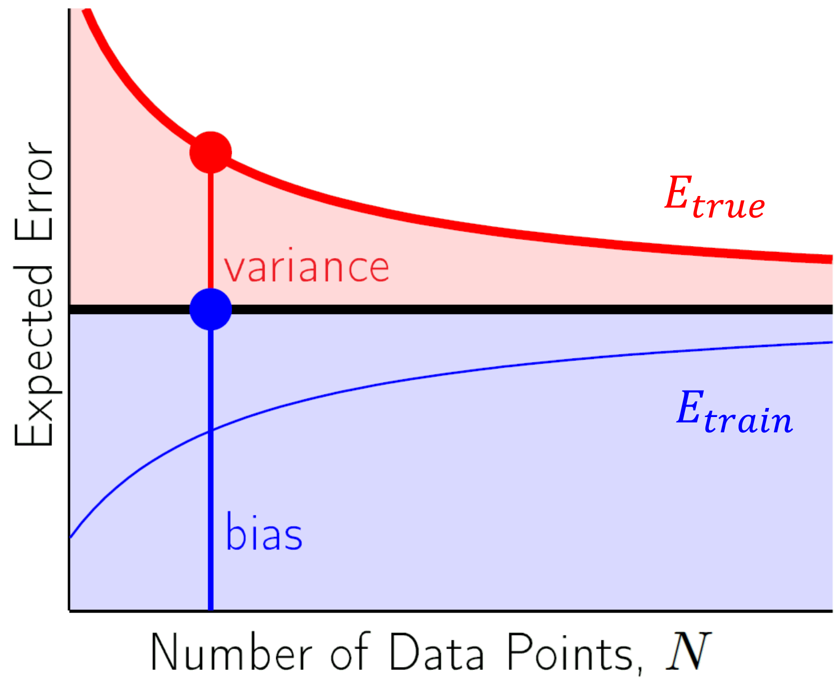
- The obtained bounds are loose.
- Although bound is loose, it can be useful for comparing the generalization of different methods
- In real application, models with lower VC tends to generalize better

Practical: how many samples do I need?

- Rule of thumb: requiring N to be at least $10 \times VC(H)$ to get decent generalization

VC vs. bias-variance





VC analysis

bias-variance

$$E_{true} = E_{\mathcal{D}}[E_{true}(f^{\mathcal{D}})]$$
$$E_{train} = E_{\mathcal{D}}[E_{train}(f^{\mathcal{D}})]$$

Summary of PAC bounds

With probability $\geq 1 - \delta$

- For all $h \in H$ s.t. $E_{train}(h) = 0$

$$E_{true}(h) \leq \epsilon = \frac{\ln|H| + \ln\frac{1}{\delta}}{2N}$$

- For all $h \in H$

$$|E_{true}(h) - E_{train}(h)| \leq \epsilon = \sqrt{\frac{\ln|2H| + \ln\frac{1}{\delta}}{2N}}$$

Finite hypothesis space

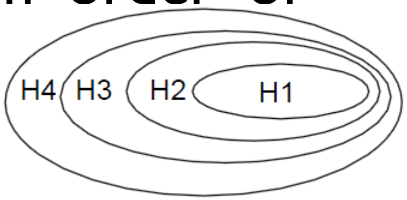
- For all $h \in H$

$$|E_{true}(h) - E_{train}(h)| \leq \epsilon = \sqrt{\frac{8\ln m_{\mathcal{H}}(2N) + 8\ln\frac{4}{\delta}}{N}}$$

Infinite hypothesis space

Using PAC bounds for model selection

- Consider nested model spaces $H_1, H_2, \dots, H_k, \dots$ in order of increasing complexity:
  - Finite hypothesis spaces: $|H_1| \le |H_2| \le \cdots \le |H_k| \le \cdots$
  - Infinite hypothesis spaces: $VC(H_1) \le VC(H_2) \le \cdots \le VC(H_k) \le \cdots$



- For each hypothesis space H_k, we know with high probability ($\ge 1 - \delta_k$), for all $h \in H_k$:

$$E_{true}(h) \le E_{train}(h) + \epsilon(H_k)$$

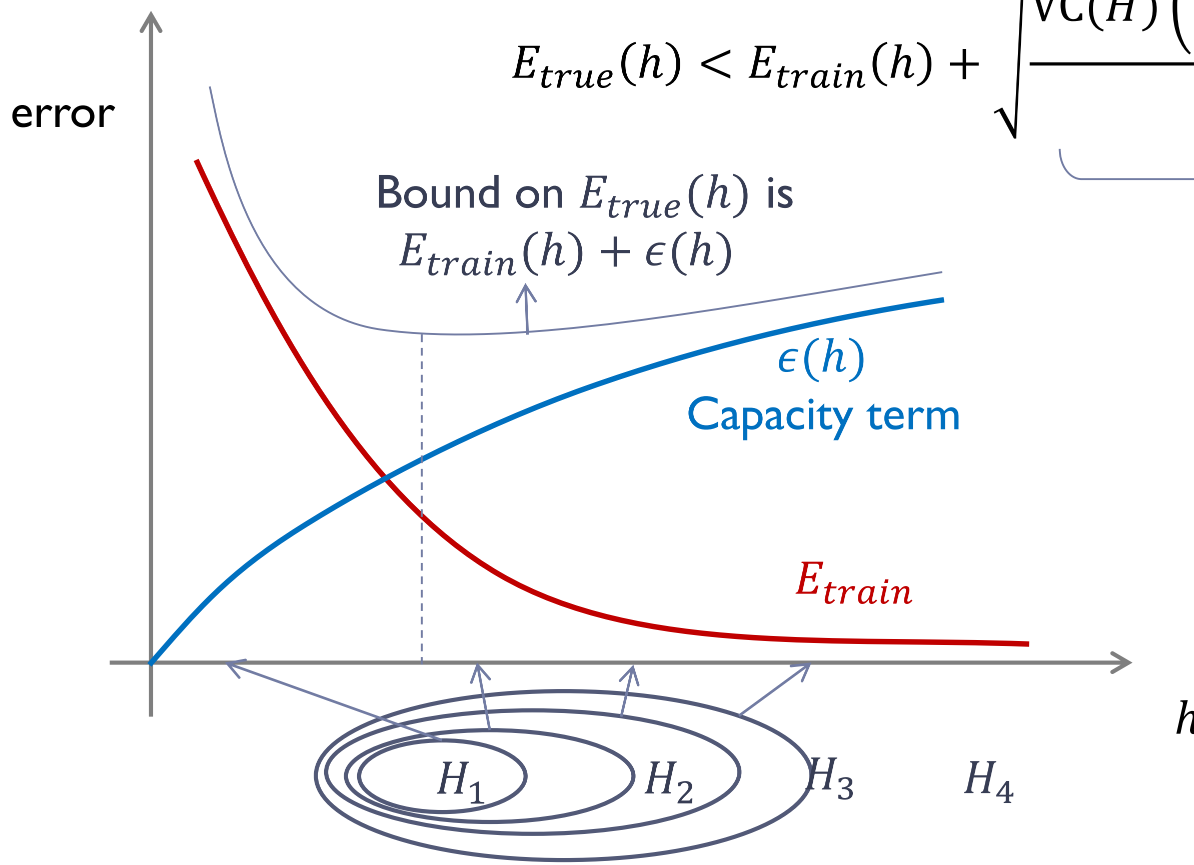
$\epsilon(H_k)$: capacity term that depends on $|H_k|$ or $VC(H_k)$

- As complexity k increases, E_{train} decreases but $\epsilon(H_k)$ increases (Bias variance tradeoff)

Model selection by SRM

- SRM finds the subset of functions which minimizes the bound on the true error (risk)

$$E_{true}(h) < E_{train}(h) + \underbrace{\sqrt{\frac{\mathrm{VC}(H)\left(\ln\frac{2N}{VC(H)}+1\right)+\ln\frac{4}{\delta}}{N}}}_{\epsilon(h)}$$



trade-off between hypothesis space complexity
and the quality of fitting the training data

Model selection by SRM

- Structural Risk Minimization (SRM):
  - Within each model space, find the best hypothesis using Empirical Risk Minimization (ERM):

$$\hat{h} = \underset{h \in H}{\operatorname{argmin}}\, E_{train}(h)$$

  - Choose model space that minimizes the upper bound on E_{true}:

$$\hat{k} = \underset{k \geq 1}{\operatorname{argmin}} \left\{ E_{train}\left(\hat{h}_k\right) + \epsilon(H_k) \right\}$$

  - Final hypothesis is $\hat{h} = \hat{h}_{\hat{k}}$

Summary

- PAC bounds on true error in terms of training error and complexity of hypothesis space
  - Bound for perfectly consistent learner ($E_{train}(h^*) = 0$)
  - Bound for agnostic learning ($E_{train}(h^*) > 0$)
  - $|H| = \infty \Rightarrow$ VC dimension
  - VC provides much tighter bounds in many cases

- Complexity of the classifier depends on number of points that can be classified exactly
  - Finite case: Number of hypothesis
  - Infinite case: VC dimension

- SRM
  - Bias-Variance tradeoff in learning theory
  - Model selection using SRM
    - Bounds are often too loose in practice

References

- T. Mitchell, "Machine Learning", 1998, Chapter 7.
- Yaser S. Abu-Mostafa et. al, "**Learning from Data**", Chapter 2.