Learning Theory

Machine Learning
CS4824/ECE4424
Bert Huang
Virginia Tech

Outline

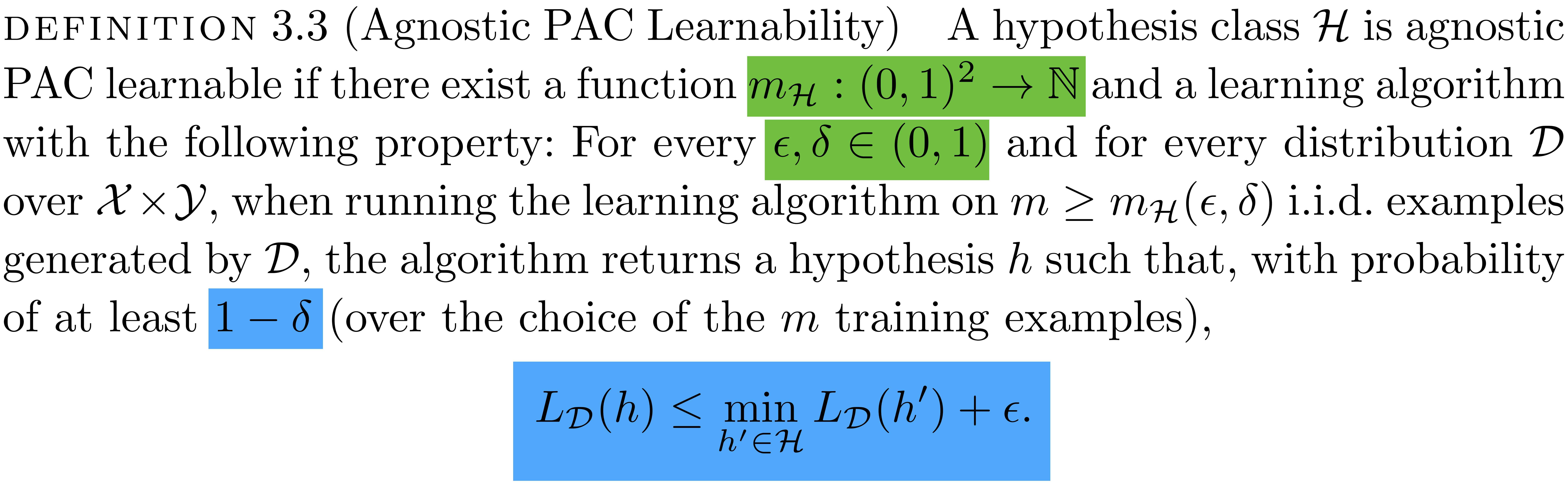
- Probably approximately correct learning
- Generalization bound of SVM
  - Vapnik-Chervonenkis dimension (VC dimension)

PAC Learning

DEFINITION 3.1 (PAC Learnability) A hypothesis class $\mathcal{H}$ is PAC learnable if there exist a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm with the following property: For every $\epsilon, \delta \in (0,1)$, for every distribution $\mathcal{D}$ over $\mathcal{X}$, and for every labeling function $f : \mathcal{X} \to \{0,1\}$, if the realizable assumption holds with respect to $\mathcal{H}, \mathcal{D}, f$, then when running the learning algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples generated by $\mathcal{D}$ and labeled by f, the algorithm returns a hypothesis h such that, with probability of at least $1 - \delta$ (over the choice of the examples), $L_{(\mathcal{D},f)}(h) \leq \epsilon$.

From Understanding Machine Learning by Shalev-Schwartz, Ben-David

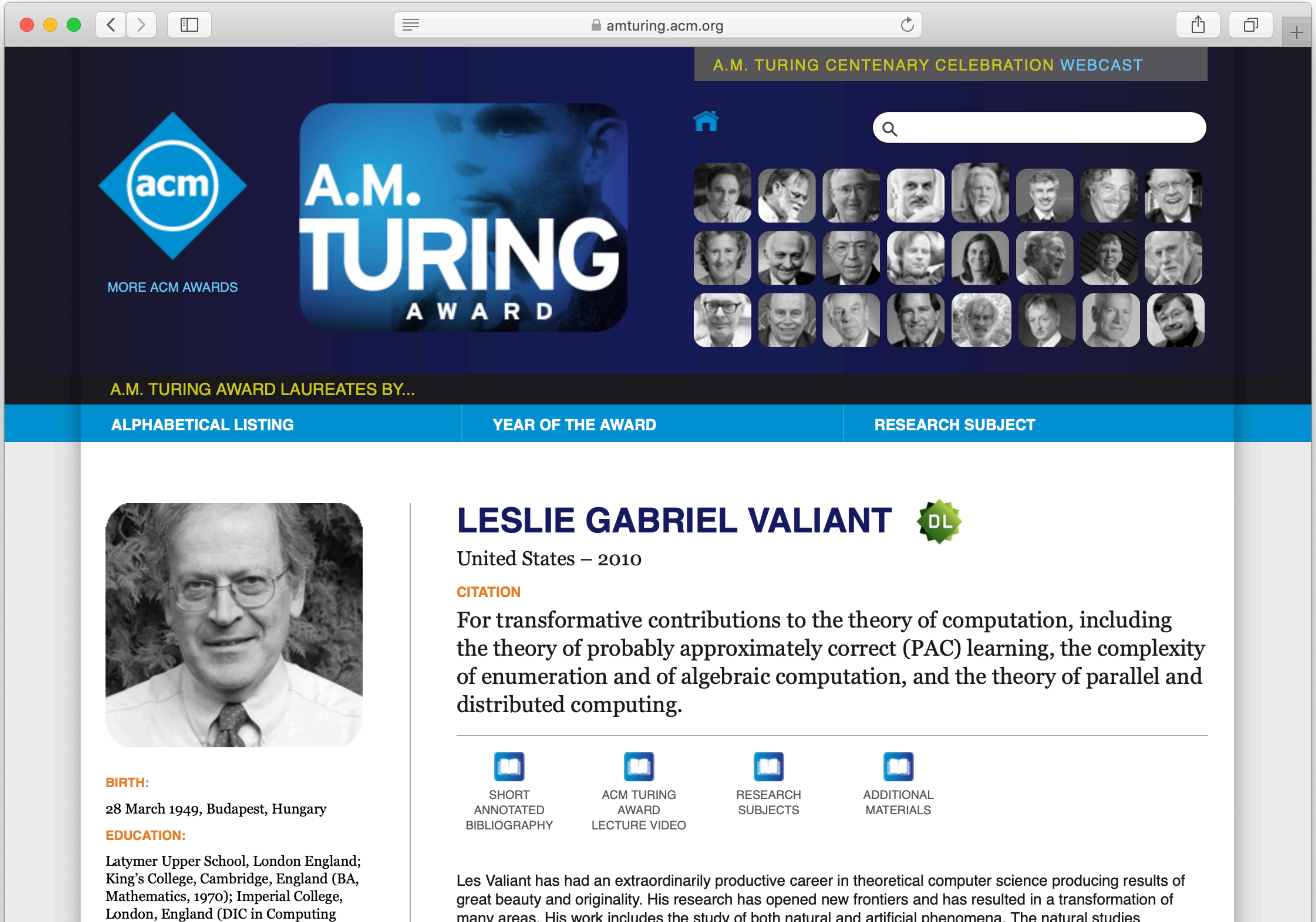
PAC Learning



DEFINITION 3.3 (Agnostic PAC Learnability) A hypothesis class $\mathcal{H}$ is agnostic PAC learnable if there exist a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm with the following property: For every $\epsilon, \delta \in (0,1)$ and for every distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, when running the learning algorithm on $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ i.i.d. examples generated by $\mathcal{D}$, the algorithm returns a hypothesis h such that, with probability of at least $1 - \delta$ (over the choice of the m training examples),

$$L_{\mathcal{D}}(h) \leq \min_{h' \in \mathcal{H}} L_{\mathcal{D}}(h') + \epsilon.$$

From Understanding Machine Learning by Shalev-Schwartz, Ben-David



LESLIE GABRIEL VALIANT

United States – 2010

CITATION

For transformative contributions to the theory of computation, including the theory of probably approximately correct (PAC) learning, the complexity of enumeration and of algebraic computation, and the theory of parallel and distributed computing.

BIRTH:

28 March 1949, Budapest, Hungary

EDUCATION:

Latymer Upper School, London England; King's College, Cambridge, England (BA, Mathematics, 1970); Imperial College, London, England (DIC in Computing

Les Valiant has had an extraordinarily productive career in theoretical computer science producing results of great beauty and originality. His research has opened new frontiers and has resulted in a transformation of many areas. His work includes the study of both natural and artificial phenomena. The natural studies

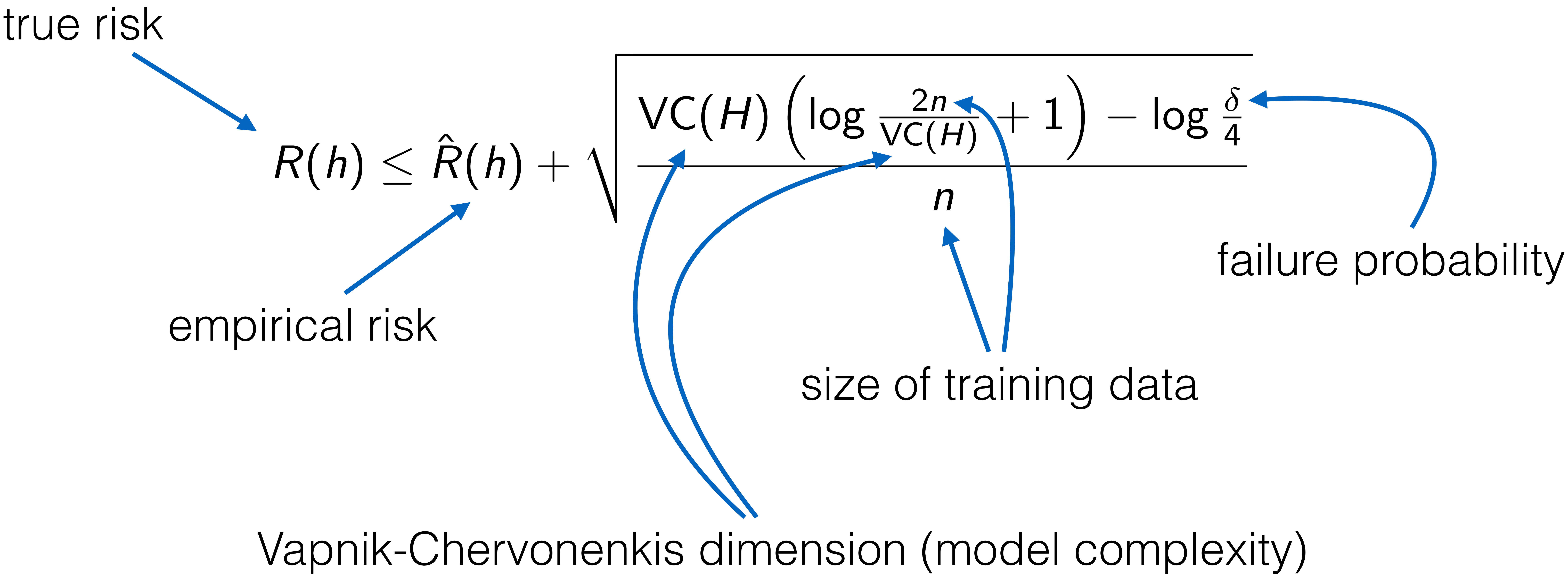
Batch Supervised Learning

- Draw data set $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ from distribution $\mathbb{D}$
- Algorithm A learns hypothesis $h \in H$ from set H of possible hypotheses $A(D) = h$
- We measure the quality of h as the expected **loss**: $\underset{(x,y) \in \mathbb{D}}{E} [\ell(y, h(x))]$
  - This quantity is known as the **risk**
  - E.g., loss could be the Hamming loss $\ell_{\text{Hamming}}(a, b) = \begin{cases} 0 & \text{if } a = b \\ 1 & \text{otherwise} \end{cases}$

Empirical Risk Minimization

- Choose a classifier (and parameter settings) to reduce *empirical risk*
    - Average loss of observable training set.
- E.g., SVM, logistic regression, perceptron

Generalization Error Bound



$$R(h) \leq \hat{R}(h) + \sqrt{\frac{\mathrm{VC}(H)\left(\log\frac{2n}{\mathrm{VC}(H)} + 1\right) - \log\frac{\delta}{4}}{n}}$$

true risk

empirical risk

size of training data

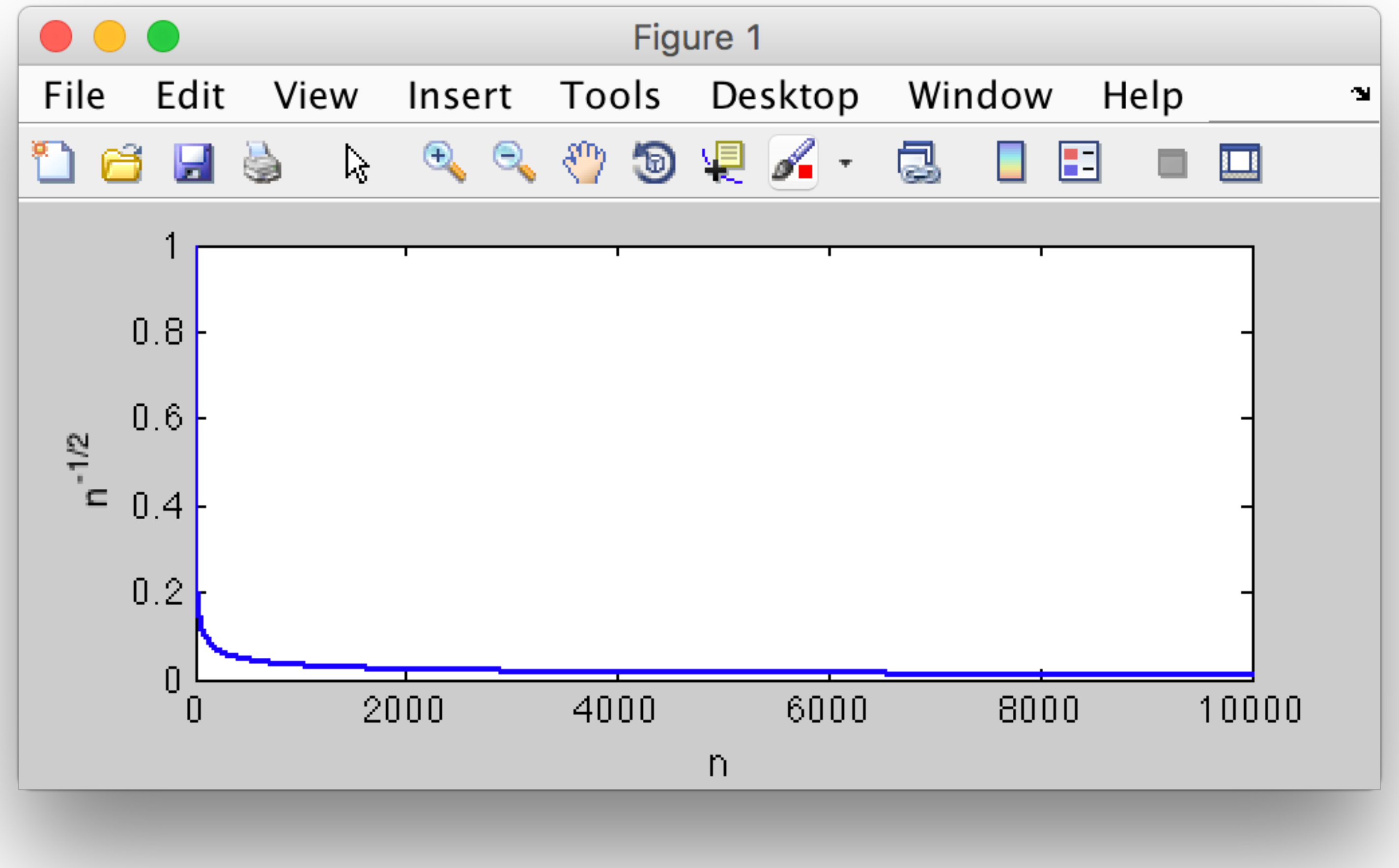
failure probability

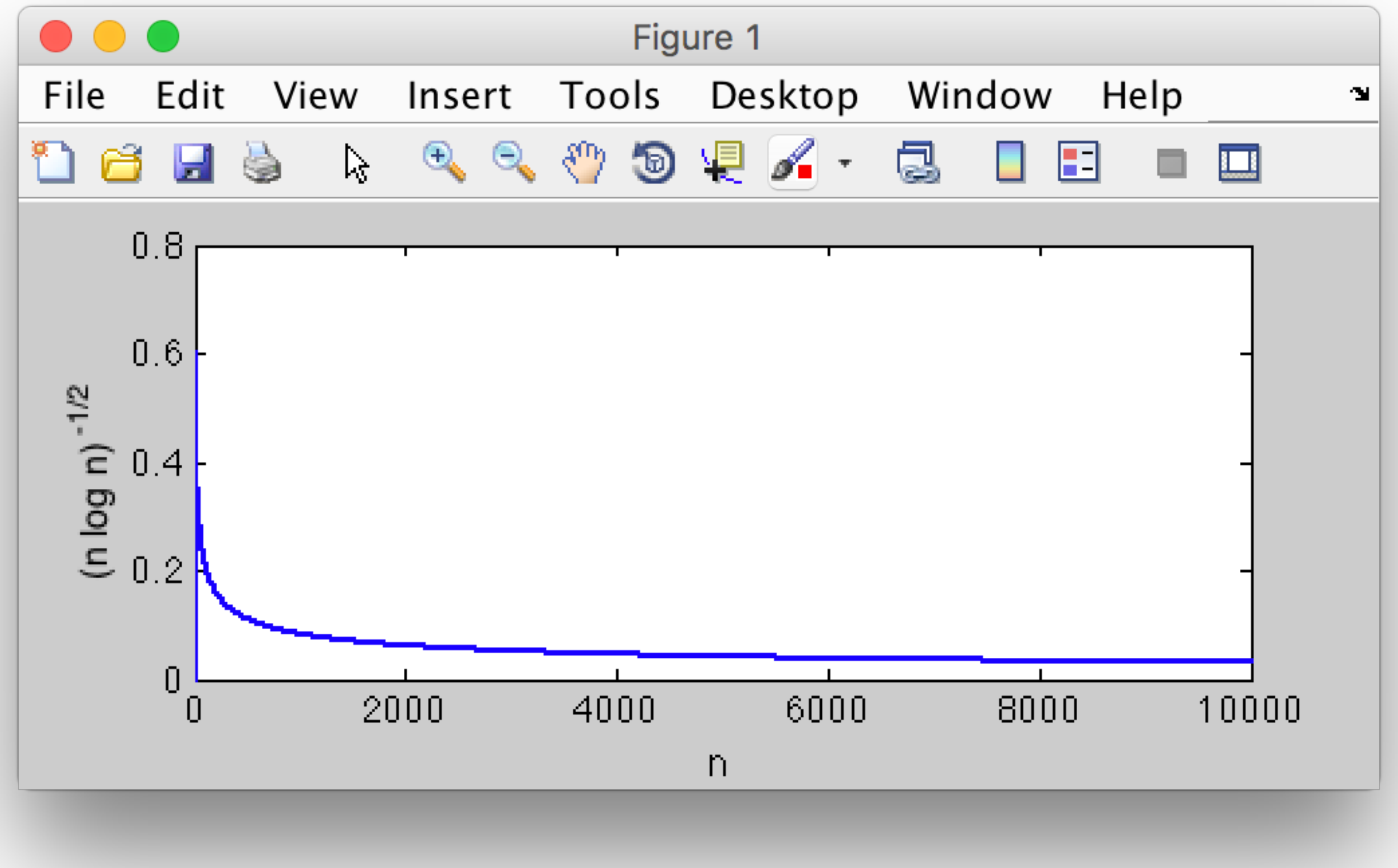
Vapnik-Chervonenkis dimension (model complexity)

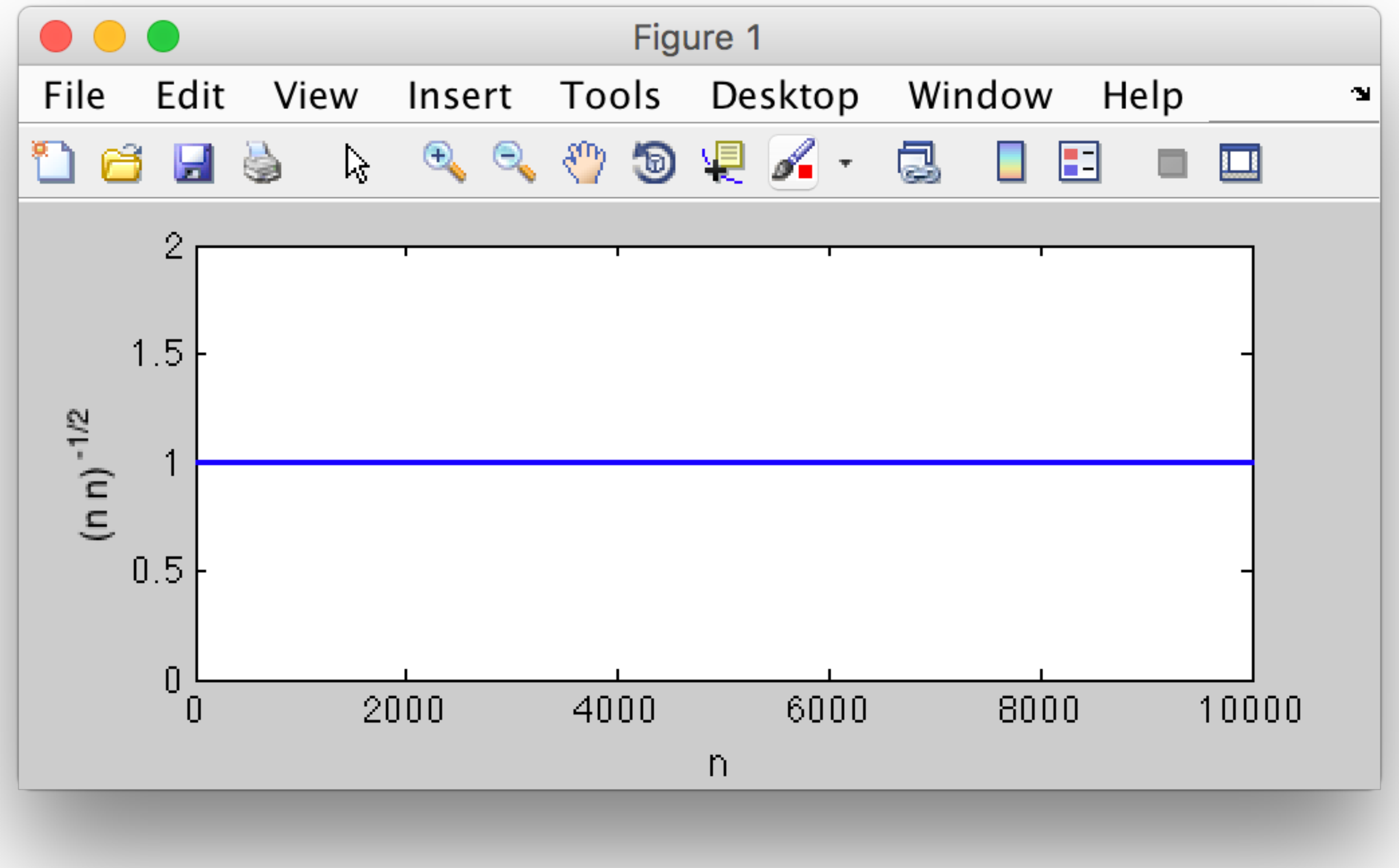
$$R(h) \leq \hat{R}(h) + \sqrt{\frac{\mathrm{VC}(H)\left(\log \frac{2n}{\mathrm{VC}(H)} + 1\right) - \log \frac{\delta}{4}}{n}} \approx \sqrt{\frac{\text{complexity(H)}}{n}}$$

if complexity is fixed

if complexity is O(n)







Vapnik-Chervonenkis Dimension

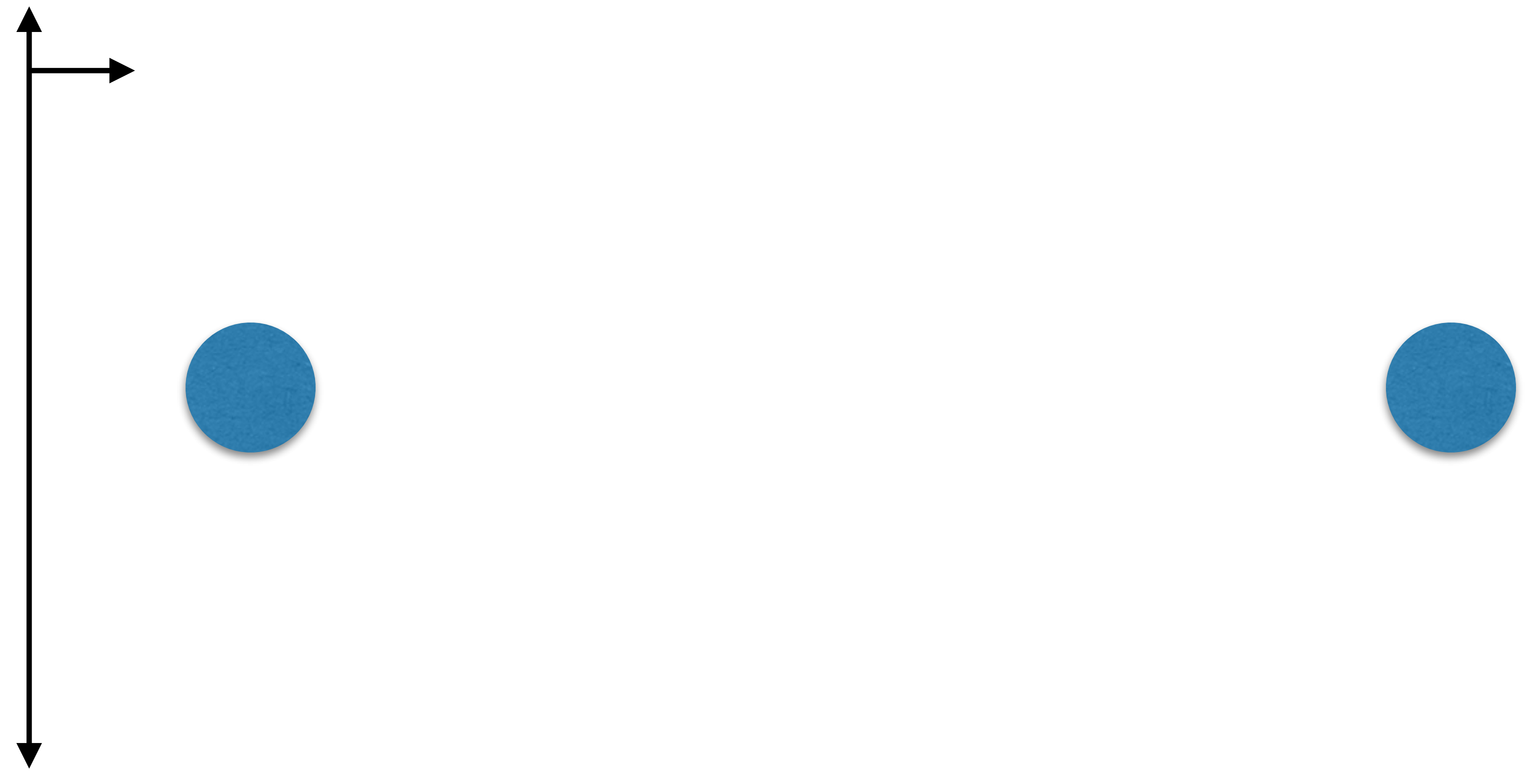
- Expressive power, or **capacity**, of a **hypothesis class**
  - Linear classifiers in d-dimensional space
  - Degree k polynomial classifiers
  - Hierarchical axis-parallel classifiers (decision trees)
- Measured by ability of hypothesis class to **shatter** n points

Shattering

Classify points into all possible labels

✔

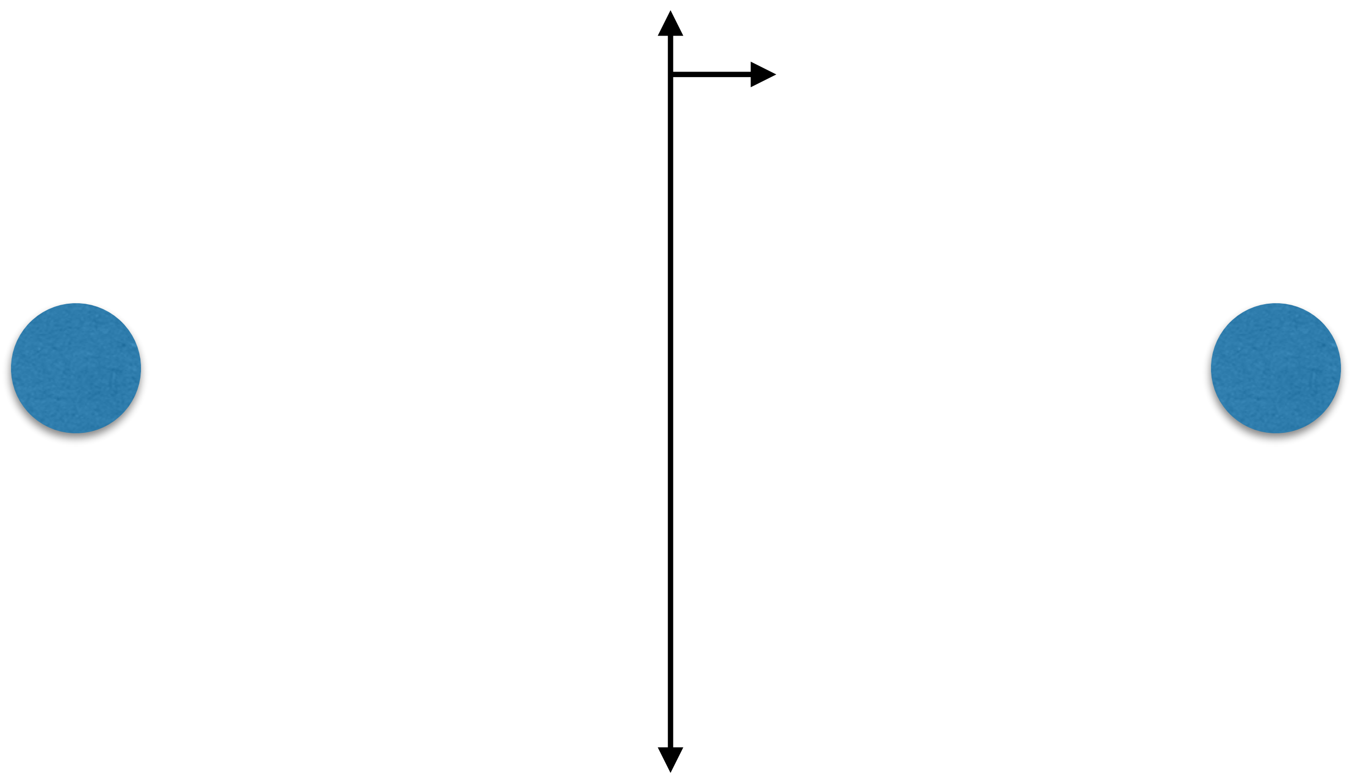
++, +-, -+, - -



Shattering

Classify points into all possible labels

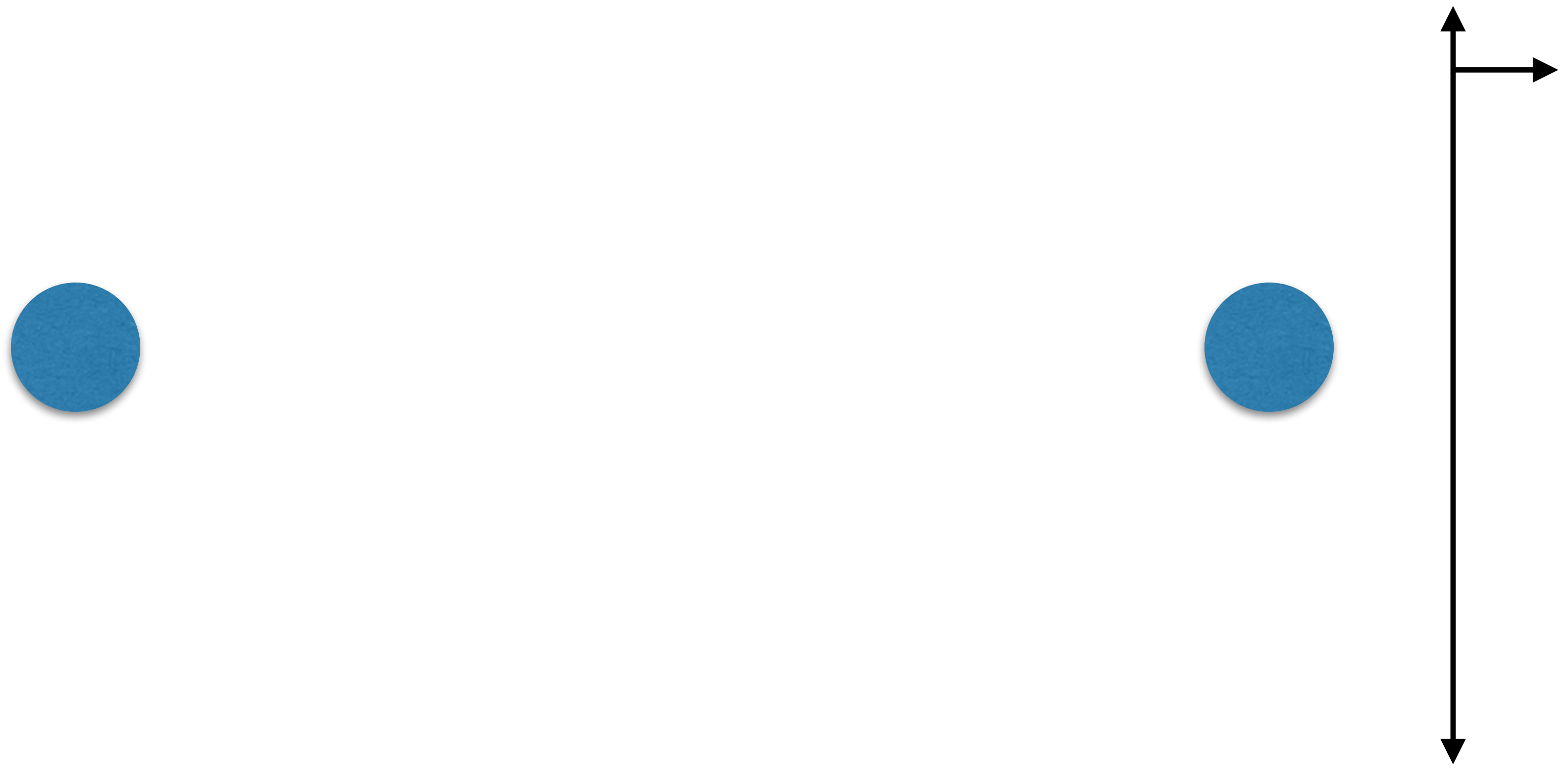
$\overset{\checkmark}{++}$, +-, $\overset{\checkmark}{-+}$, - -



Shattering

Classify points into all possible labels

✓ ++, +-, ✓ -+, ✓ - -

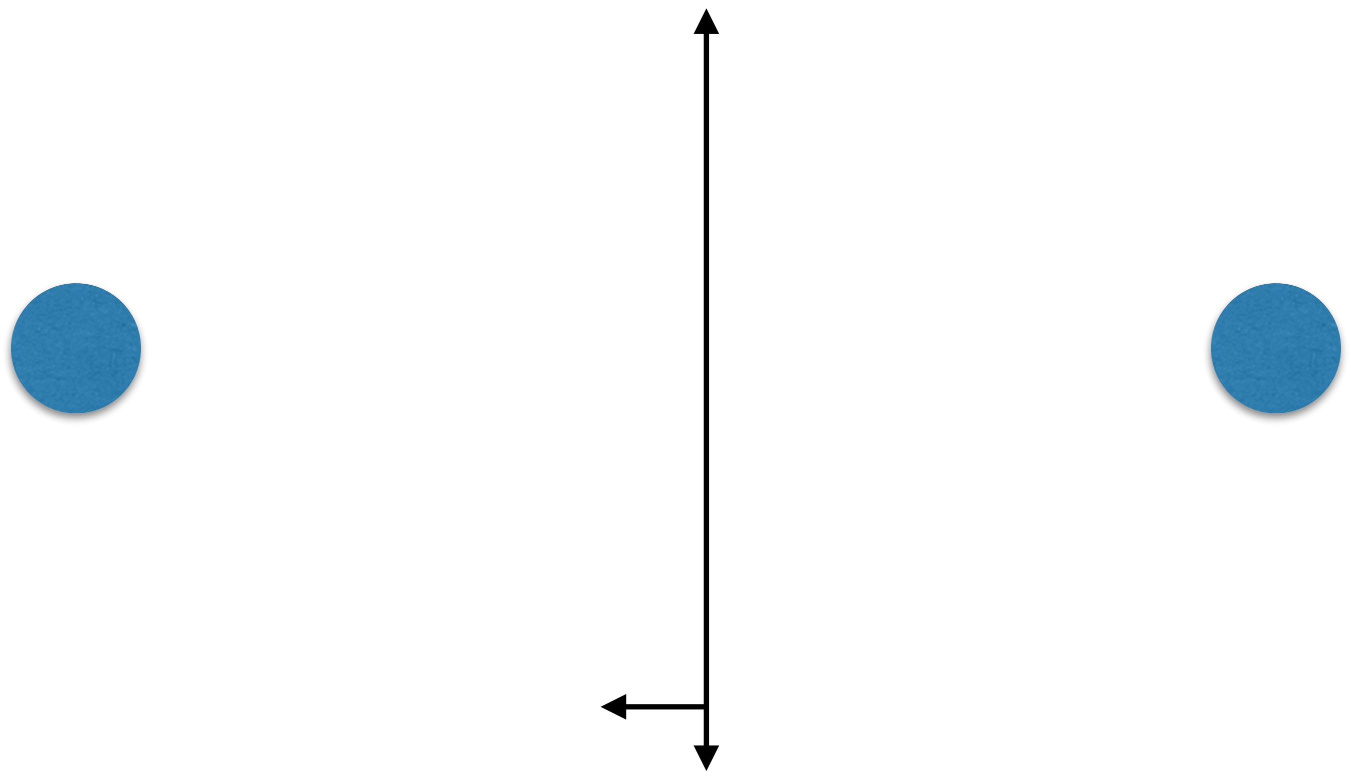


Shattering

Classify points into all possible labels

✔ ✔ ✔ ✔

++, +-, -+, - -

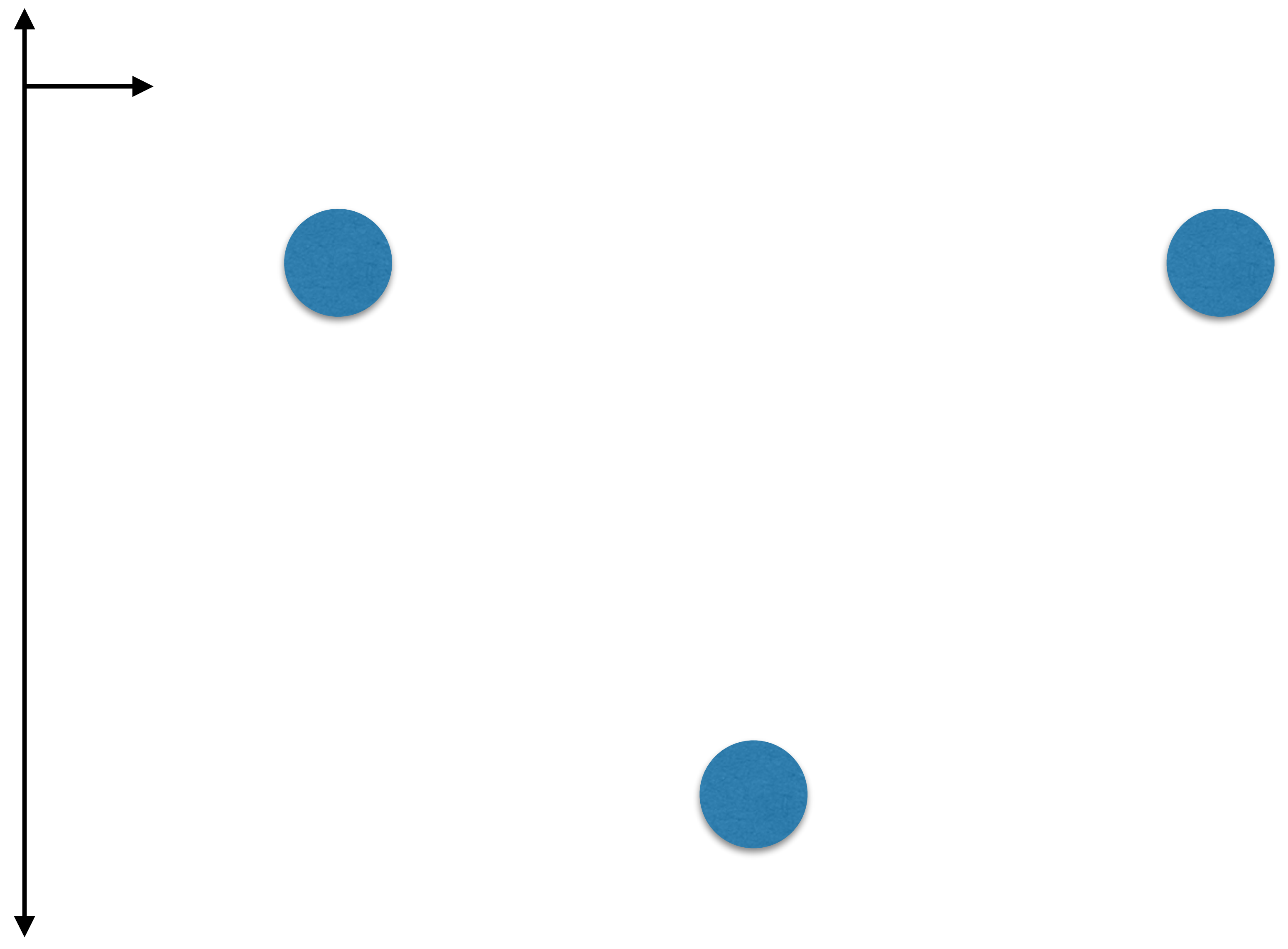


Shattering

Classify points into all possible labels

✔

+++, ++-, +-+, +- -, -++, - + -, - - -

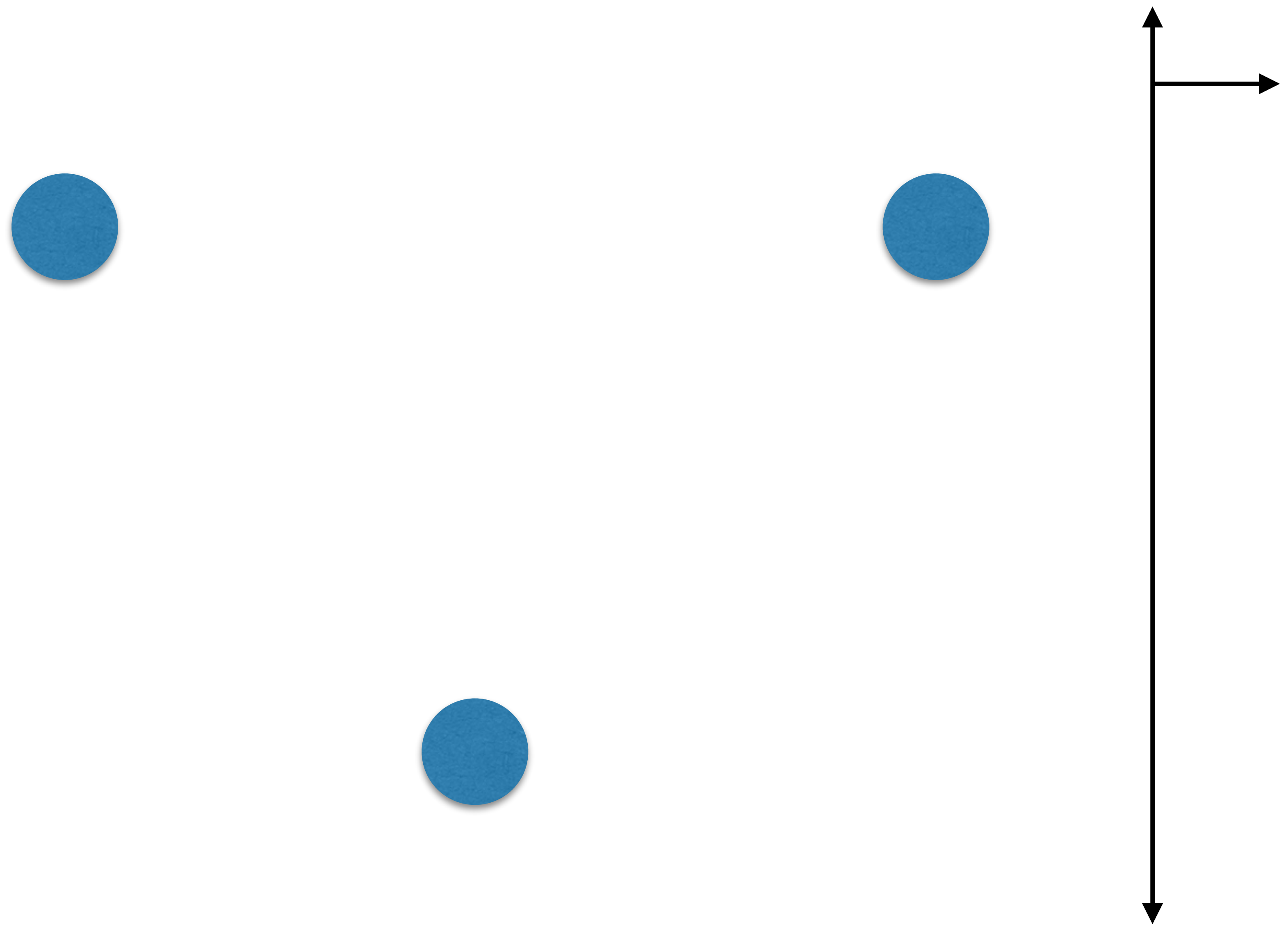


Shattering

Classify points into all possible labels

✔ ✔

+++, ++-, +-+, +- -, -++, - + -, - - -

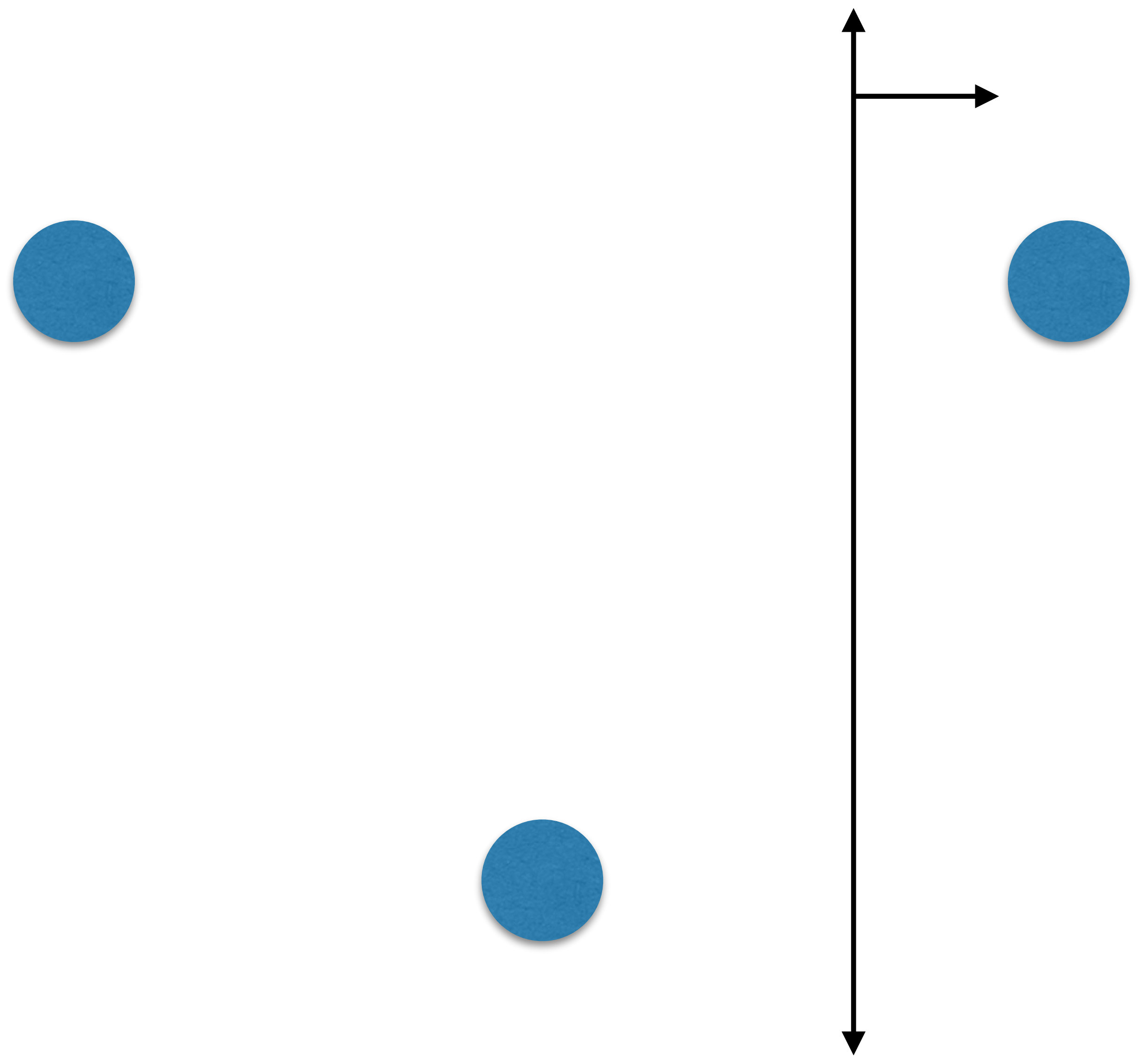


Shattering

Classify points into all possible labels

✔ ✔ ✔

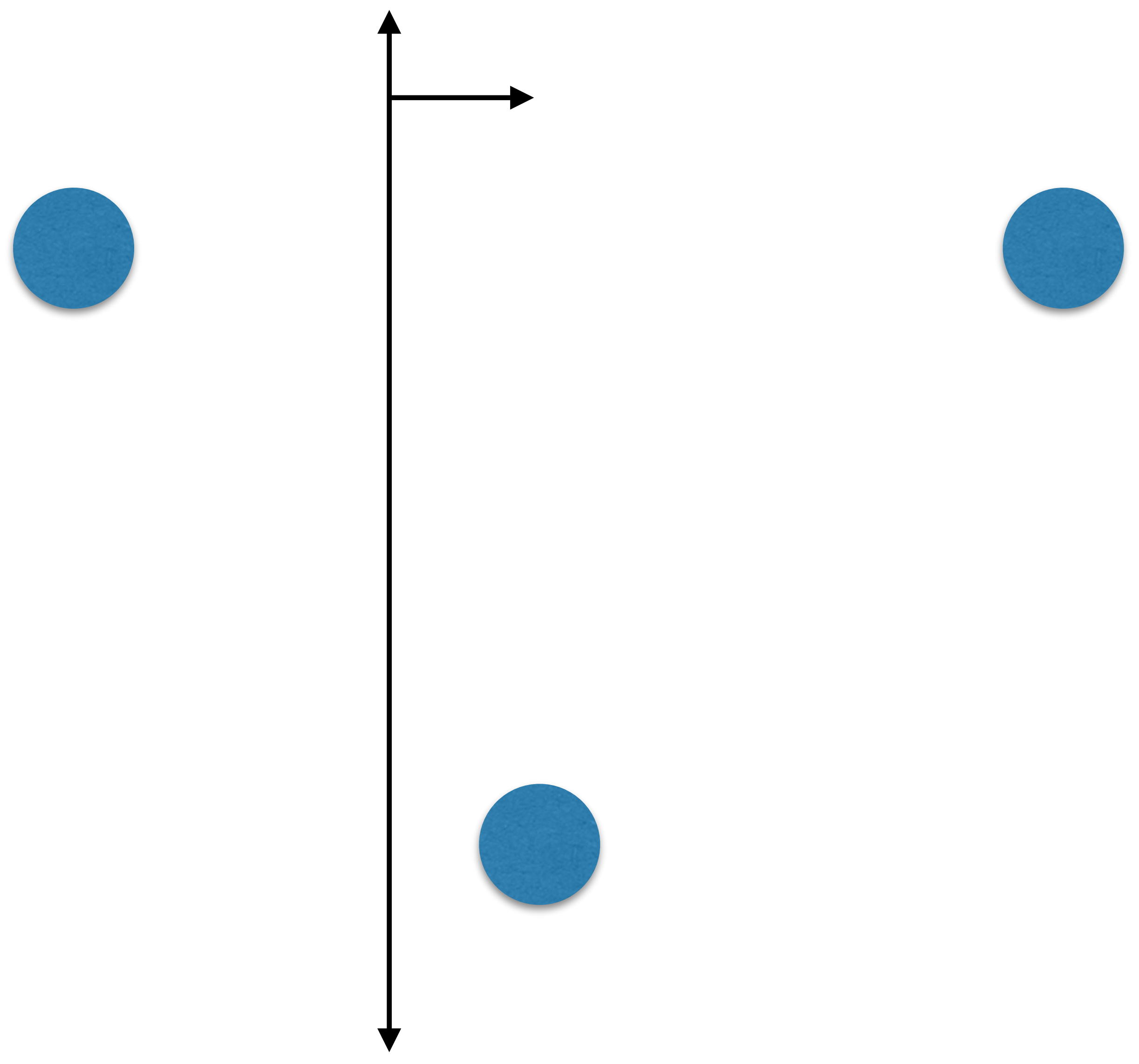
+++, ++-, +-+, +- -, -++, - + -, - - -



Shattering

Classify points into all possible labels

✓ ✓ ✓ ✓
+++, ++-, +-+, +- -, -++, - + -, - - -

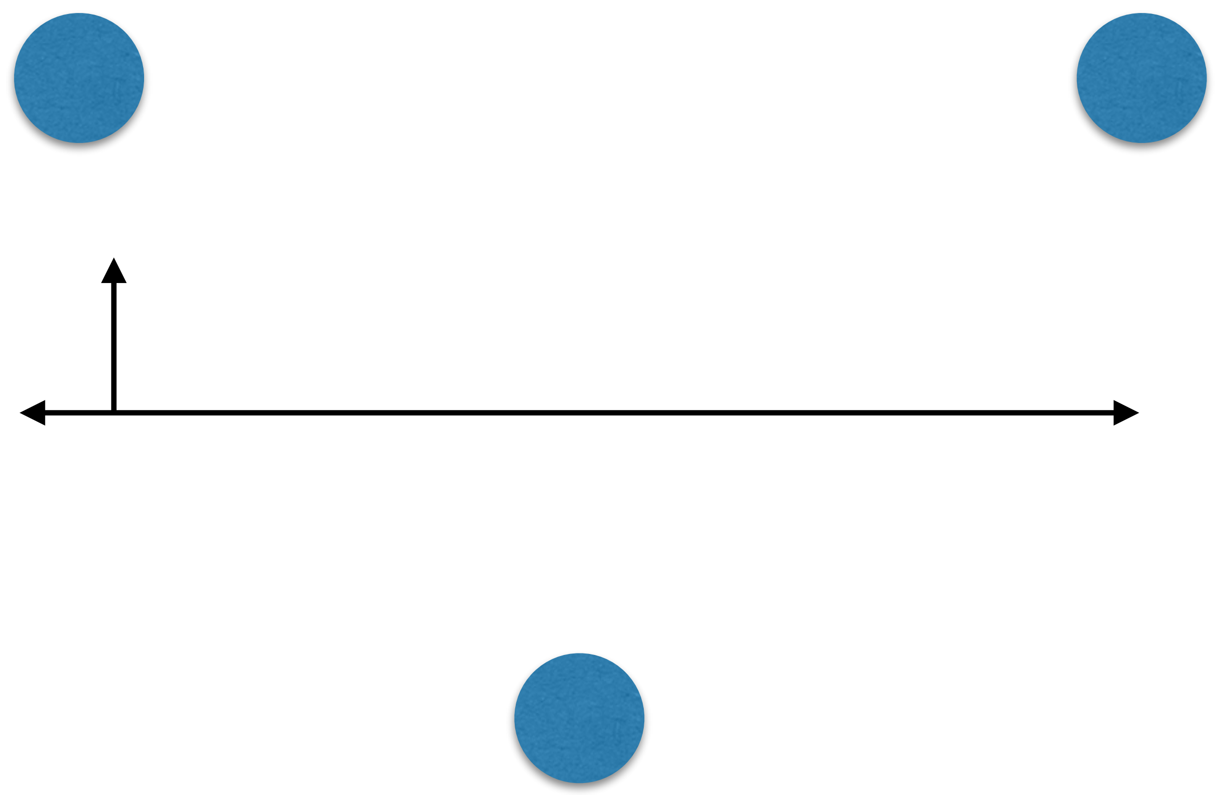


Shattering

Classify points into all possible labels

✔ ✔ ✔ ✔ ✔

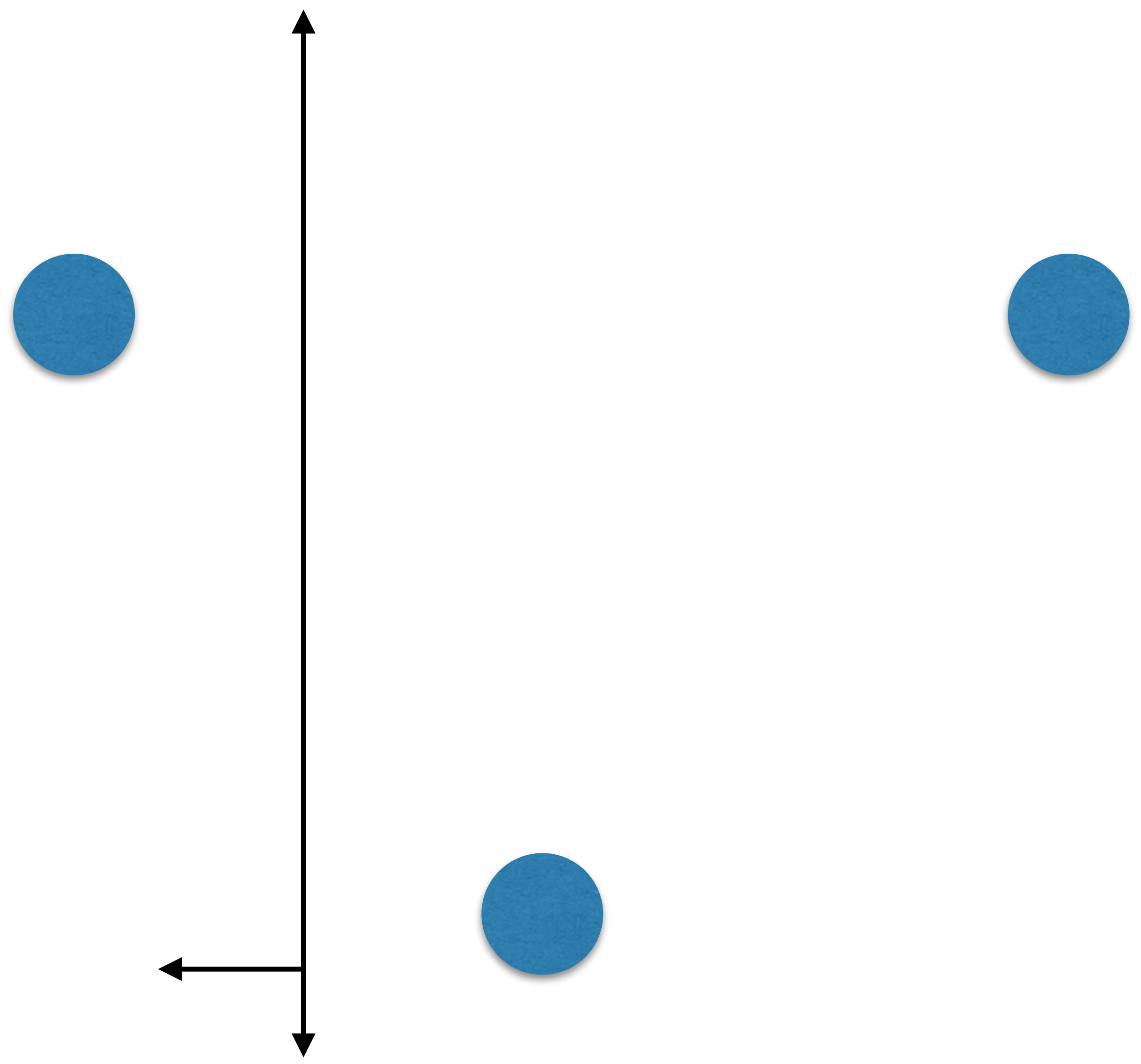
+++, ++-, +-+, +- -, -++, - + -, - - -



Shattering

Classify points into all possible labels

✔ ✔ ✔ ✔ ✔ ✔
+++, ++-, +-+, +- -, -++, - + -, - - -

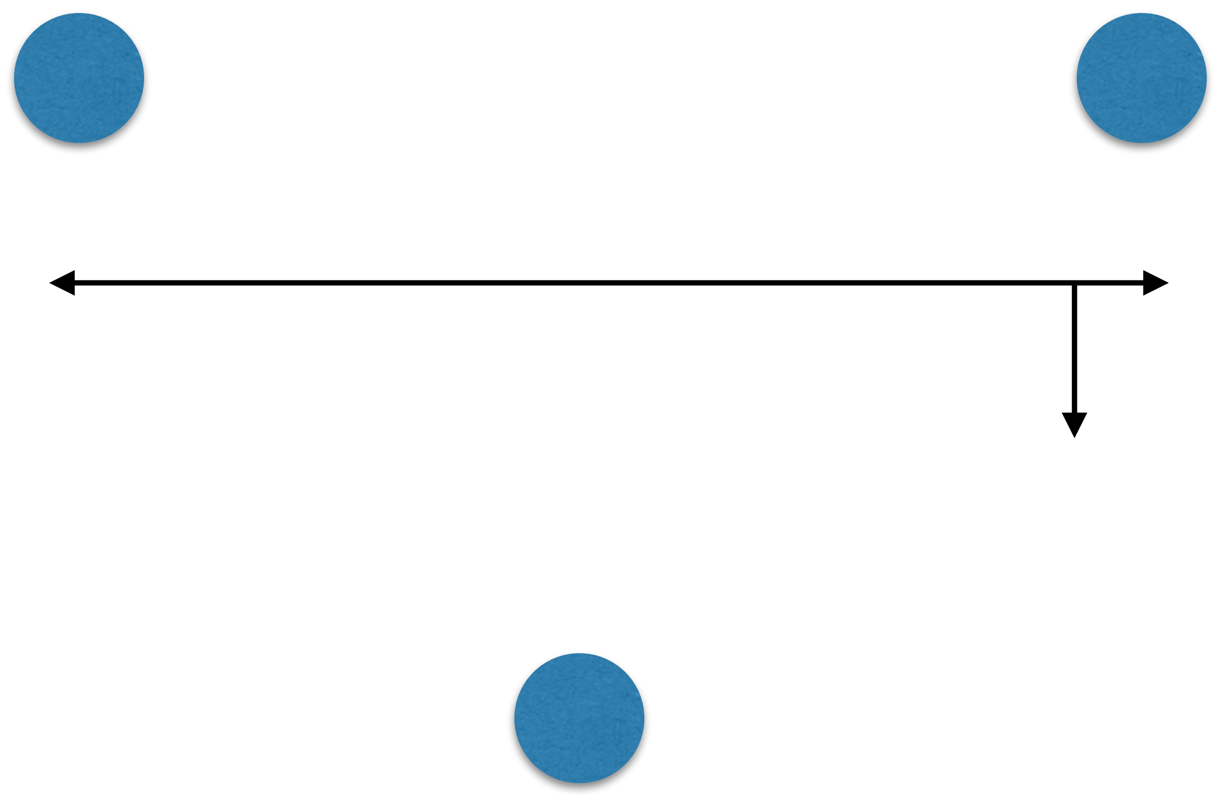


Shattering

Classify points into all possible labels

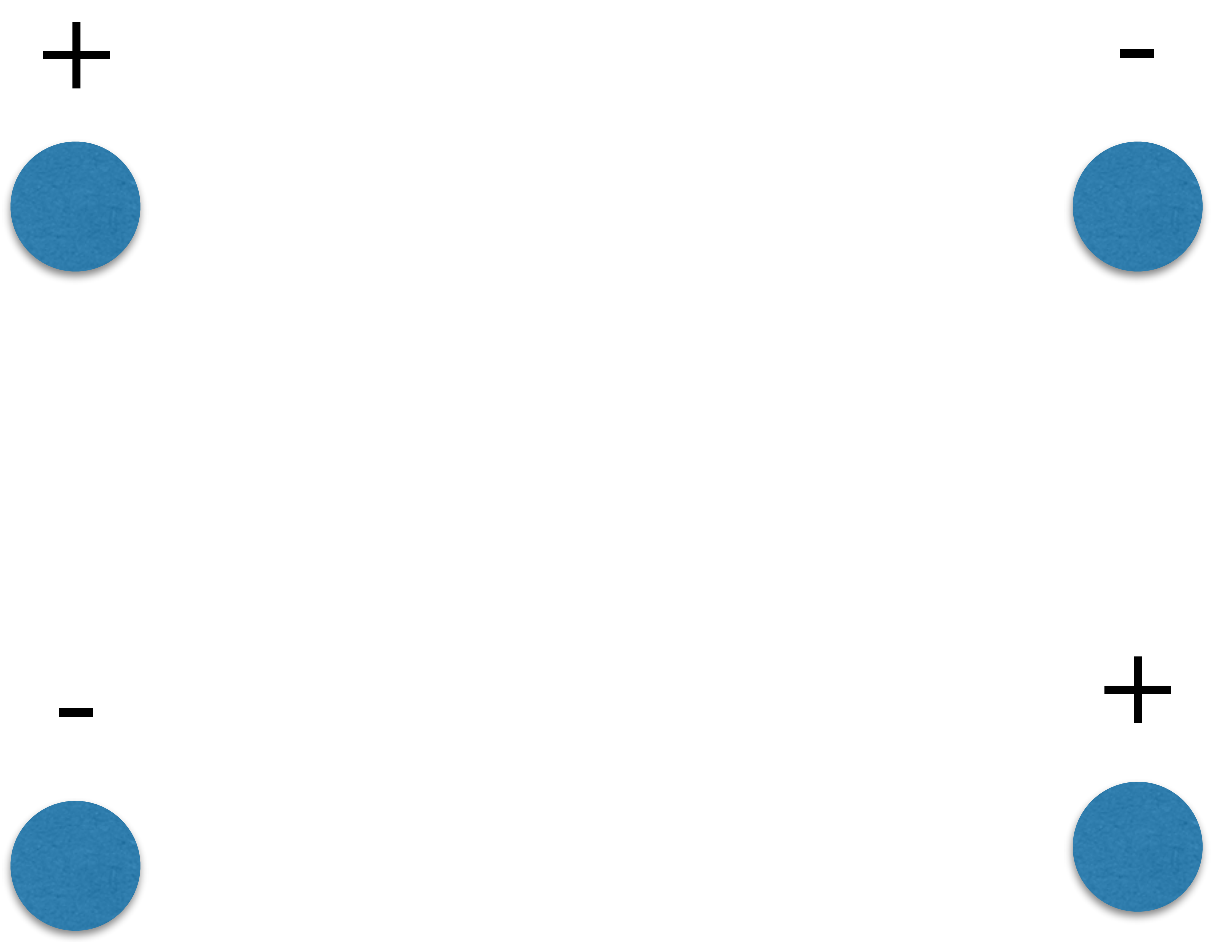
✔ ✔ ✔ ✔ ✔ ✔ ✔

+++, ++-, +-+, +- -, -++, - + -, - - -



Shattering

Classify points into all possible labels



4 points cannot be shattered by 2d linear classifier

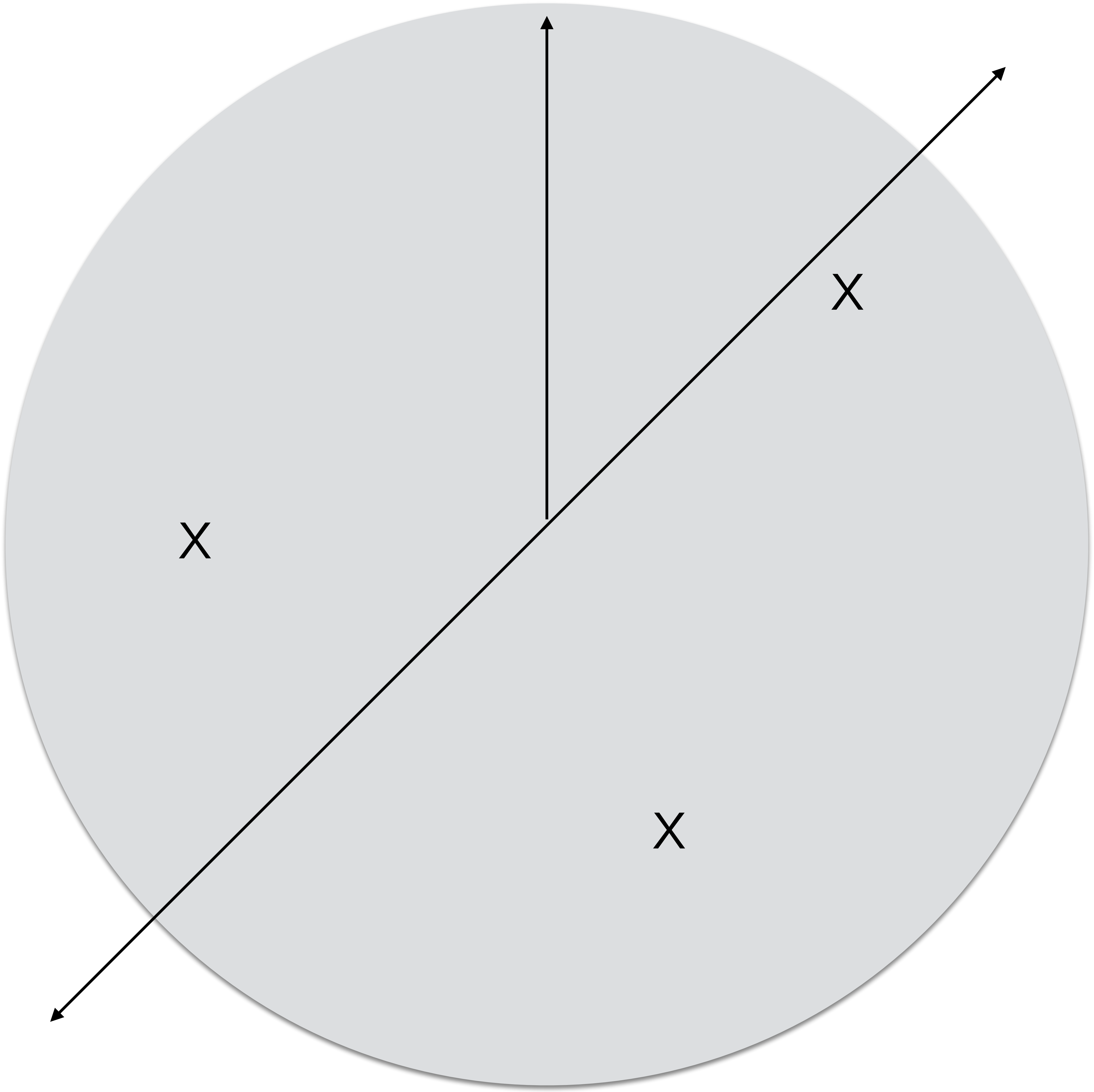
VC Dimension

- VC dimension of hypothesis class H:
- Maximum number of examples that can be shattered by H
- Examples can be arranged (feature values) in any way
- Must be shattered in same arrangement
- In general: linear classifier has VC dimension (d + 1)

VC Model Capacity Intuition

- How many points can this model class memorize?
- Game view:
  - We choose placement of points
  - Adversary chooses labeling
  - Can we classify labeling?
- Think of learning algorithm as function $A : \mathcal{X} \to \mathcal{H}$
  and hypothesis as a function $h : \mathcal{X} \to \mathcal{Y}$
- VC dimension |***y***| means ***A*** can output an ***h*** that can output any ***y***

Margin

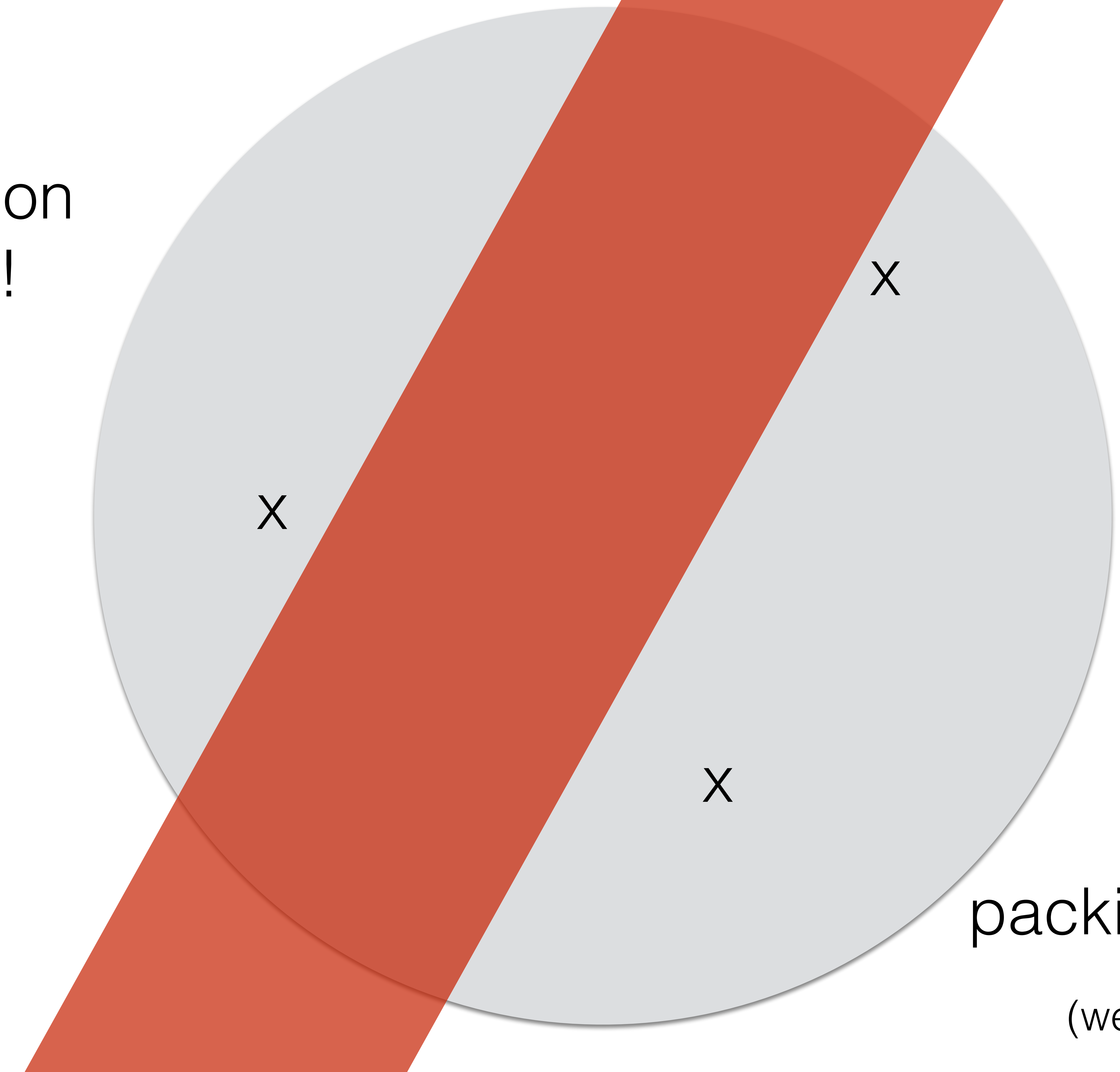


radius = R

Margin

$\mathrm{VC}(H) = R^2 w^T w$

doesn't depend on dimensionality!



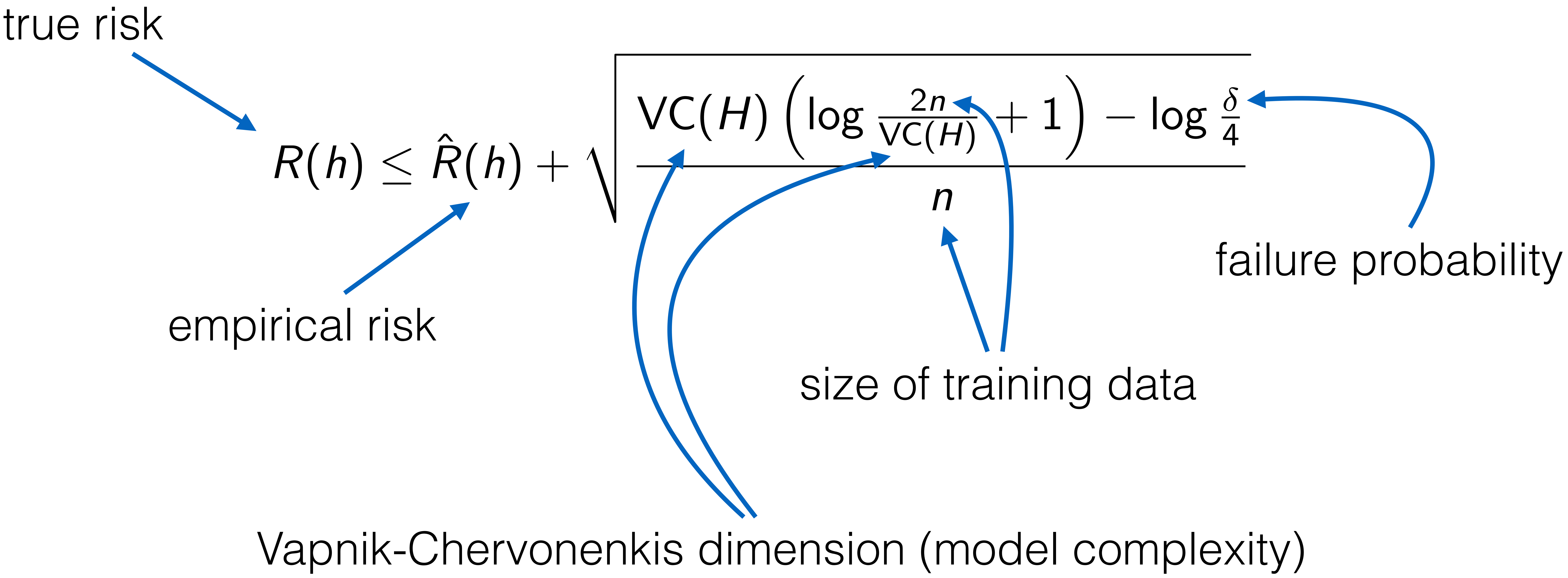
radius = R

packing points into a sphere

(we're skipping lots of details)

Generalization Error Bound

$\mathrm{VC}(H) = R^2\, w^T w$



$$R(h) \leq \hat{R}(h) + \sqrt{\frac{\mathrm{VC}(H)\left(\log \frac{2n}{\mathrm{VC}(H)} + 1\right) - \log \frac{\delta}{4}}{n}}$$

true risk

empirical risk

size of training data

failure probability

Vapnik-Chervonenkis dimension (model complexity)

Summary and Thoughts

- From analysis, SVM appears to minimize VC dimension
  - but bound assumes VC dimension is fixed
- Generalization bounds tend to be loose for real data sizes
- Formally describe trend, but are they useful?
  - Better (tighter) bounds are certainly useful
  - But loose bounds help us formally understand properties of learning algorithms