PAC Learning

Adopt slides by Alexander Ihler and Andrew Moore

Learners and Complexity

- We've seen many versions of underfit/overfit trade-off
  - Complexity of the learner
  - "Representational Power"
- Different learners have different power

- Usual trade-off:
  - More power = represent more complex systems, might overfit
  - Less power = won't overfit, but may not find "best" learner

- How can we balance the trade-off in theory?
  - Quantify the performance of the model
  - Quantify representational power

Some Notions

- Define "risk" and "empirical risk"
  - These are just "long term" test and observed training error
  - Risk, i.e., test error, true error

$$R(\theta) = \text{TestError} = \mathbb{E}[\mathbb{1}[c \neq \hat{c}(x\,;\,\theta)]]$$

Always unknown

  - Empirical risk, i.e., training error

$$R^{\text{emp}}(\theta) = \text{TrainError} = \frac{1}{m}\sum_i \mathbb{1}[c^{(i)} \neq \hat{c}(x^{(i)}\,;\,\theta)]$$

Can measure on training data

PAC Learning

- PAC: Probably Approximately Correct
- The **PAC criterion** is that a learner produces a highly accurate hypothesis with high probability:

$$P(|R(\theta) - R^{emp}(\theta)| \leq \epsilon) \geq 1 - \delta$$

- Given ϵ, δ, under what conditions a learner is PAC?
  - Learner complexity
  - …

Bounding excess risk

- Given ϵ, δ, bound the difference between risk $R(\theta)$ and empirical risk $R^{emp}(\theta)$.

Bounding test error for finite hypothesis space

- Hoeffding's inequality
  - Let $x^{(1)}, \cdots, x^{(m)}$ be independent random variables in $[0,1]$
  - $\bar{X} = \frac{1}{m}\left(x^{(1)} + \cdots + x^{(m)}\right)$
  - Then
  - $P(\mathrm{E}[\bar{X}] - \bar{X} \geq \epsilon) \leq e^{-2m\epsilon^2}$
- Union bound
  - If $A_1, \cdots, A_d$ are a set of events, then
  - $P\left(\cup_{i=1}^{d} A_i\right) \leq \sum_{i=1}^{d} P(A_i)$

Bounding test error for finite hypothesis space

- Consider loss of training examples of an arbitrary model h_θ as independent random variables
- $R^{emp}(\theta) \rightarrow \bar{X}$
- $R(\theta) \rightarrow \mathrm{E}[\bar{X}]$

So that the bound works for the trained model

- Bound the difference $R(\theta) - R^{emp}(\theta)$ for any possible $h_\theta \in \mathcal{H}$, or

$$P\left(\max_{h_\theta \in \mathcal{H}} \{R(\theta) - R^{emp}(\theta)\} \geq \epsilon\right) \leq ?$$

Bounding test error for finite hypothesis space

$$P\left(\max_{h_\theta \in \mathcal{H}} \{R(\theta) - R^{emp}(\theta)\} \geq \epsilon\right)$$

Definition

$$= P\left(\bigcup_{h_\theta \in \mathcal{H}} (R(\theta) - R^{emp}(\theta) \geq \epsilon)\right)$$

Union bound

$$\leq \sum_{h_\theta \in \mathcal{H}} P(R(\theta) - R^{emp}(\theta) \geq \epsilon)$$

Hoeffding's inequality

$$\leq \sum_{h_\theta \in \mathcal{H}} e^{-2m\epsilon^2} = He^{-2m\epsilon^2}$$

Bounding test error for finite hypothesis space

$$P(R(\theta^*) - R^{emp}(\theta^*) \leq \epsilon) \geq 1 - He^{-2m\epsilon^2}$$

With probability of at least $(1 - \delta)$, we have

$$R(\theta^*) - R^{emp}(\theta^*) \leq \sqrt{\frac{\log H - \log \delta}{2m}}$$

$$R(\theta^*) \leq R^{emp}(\theta^*) + \sqrt{\frac{\log H - \log \delta}{2m}}$$

Bounding test error for infinite hypothesis space

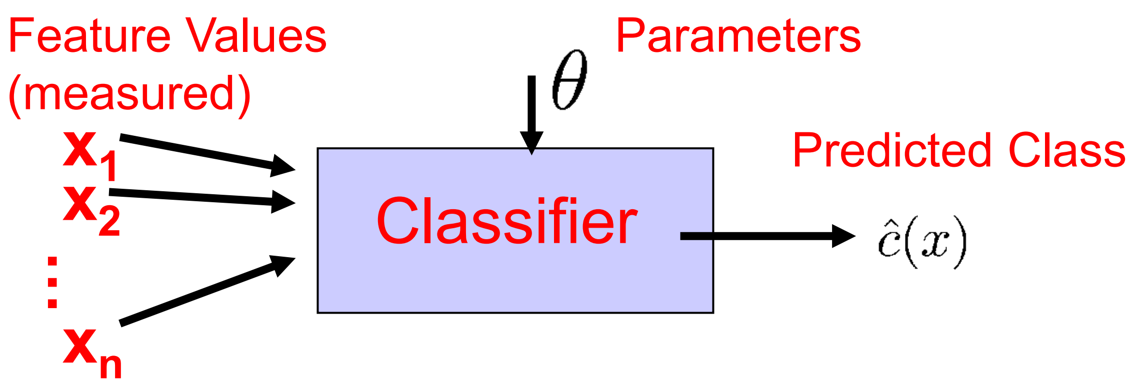
- If the hypothesis space $\mathcal{H}$ is infinite (e.g., we have real-valued parameters), we cannot use the size of $\mathcal{H}$.
- Instead, we can use a quantity called the **Vapnik-Chervonenkis** or **VC** dimension (denoted by H) of the hypothesis class.

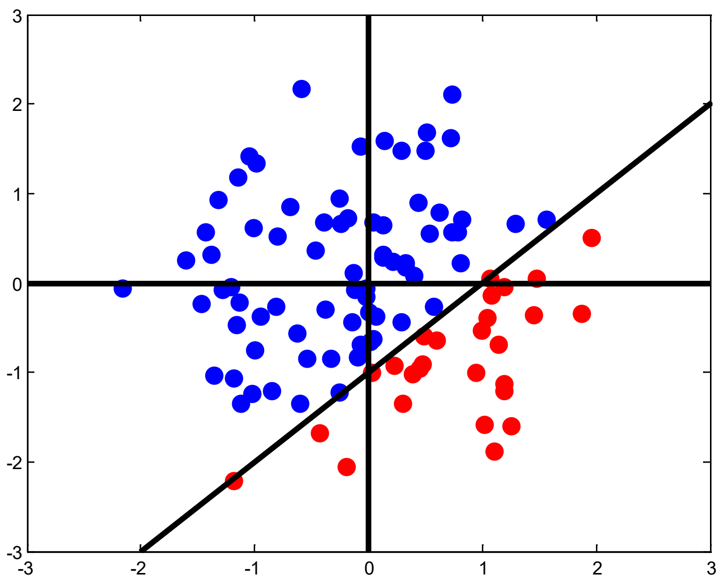
$$R(\theta^*) \leq R^{emp}(\theta^*) + \sqrt{\frac{H\log\frac{2m}{H} + H - \log\frac{\delta}{4}}{m}}$$

VC DIMENSION

Learners and Complexity

- We've seen many versions of underfit/overfit trade-off
  - Complexity of the learner
  - "Representational Power"
- Different learners have different power



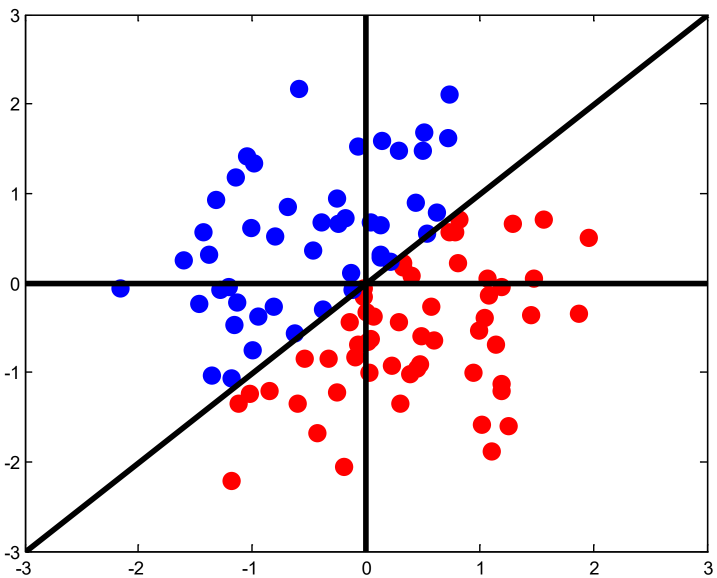


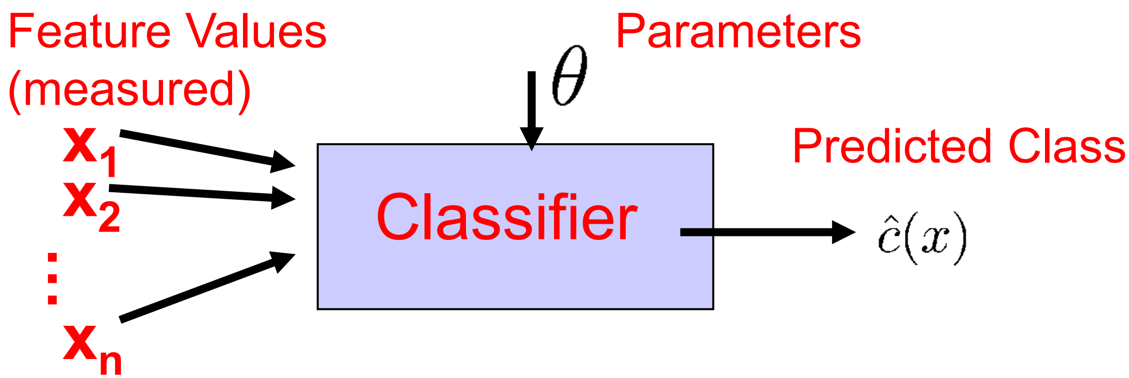
Example:

$$\hat{c}(x) = \text{sign}(\theta_1 x_1 + \theta_2 x_2 + \theta_0)$$

Learners and Complexity

- We've seen many versions of underfit/overfit trade-off
  - Complexity of the learner
  - "Representational Power"
- Different learners have different power



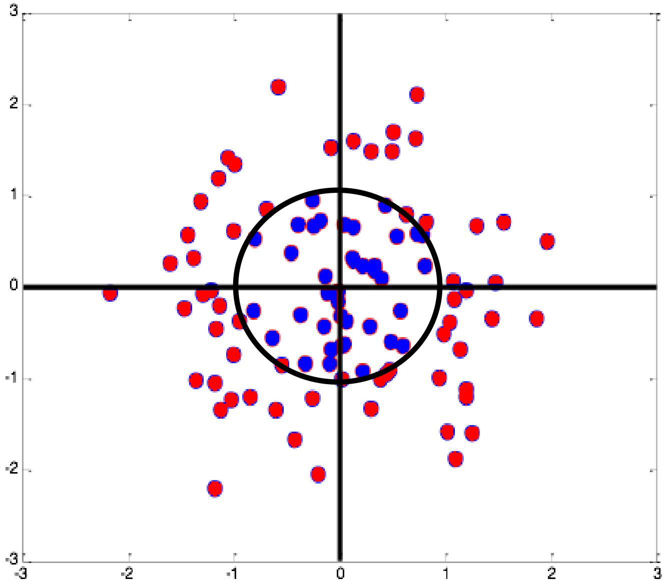


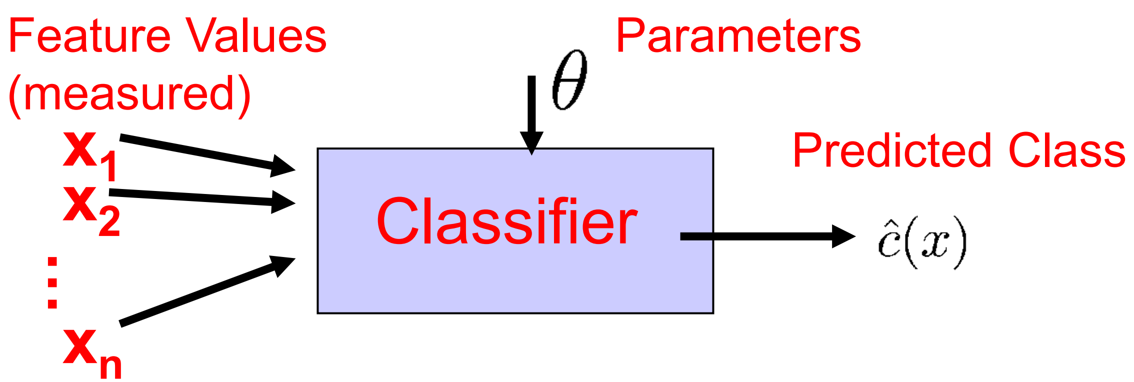
Example:

$$\hat{c}(x) = \text{sign}(\theta_1 x_1 + \theta_2 x_2)$$

Learners and Complexity

- We've seen many versions of underfit/overfit trade-off
  - Complexity of the learner
  - "Representational Power"
- Different learners have different power





Example:

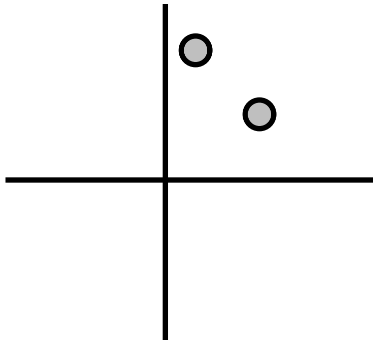
$$\hat{c}(x) = \text{sign}(\ (x_1^2 + x_2^2) - \theta_0)$$

Shattering

- We say a learner f(x) can shatter points $x^{(1)} \ldots x^{(h)}$ iff for *all* $y^{(1)} \ldots y^{(h)}$, f(x) can achieve zero error on training data $(x^{(1)},y^{(1)})$, $(x^{(2)},y^{(2)})$, … $(x^{(h)},y^{(h)})$

  (i.e., there exists some θ that gets zero error)

- Can $f(x;\theta) = sign(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$ shatter these points?

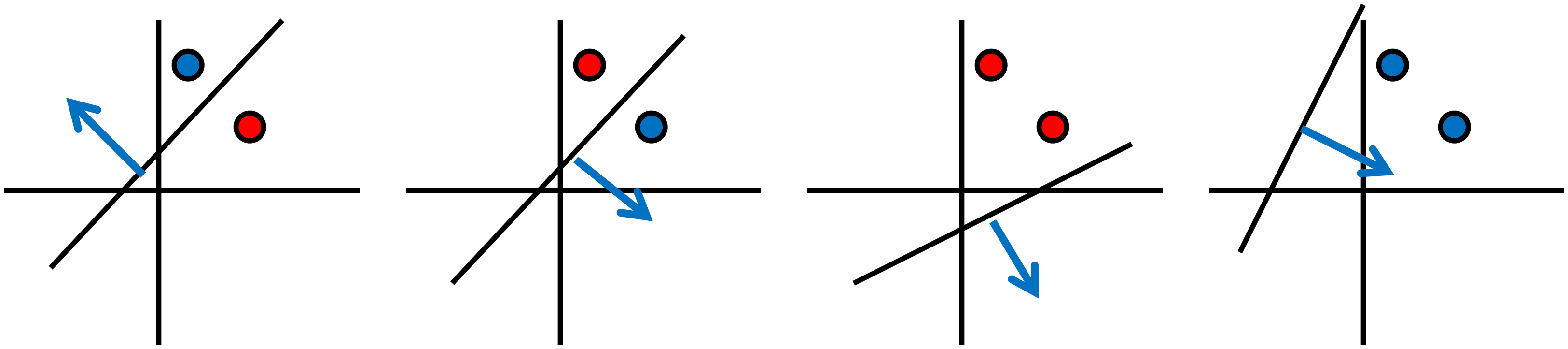


Shattering

- We say a learner f(x) can shatter points $x^{(1)} \ldots x^{(h)}$ iff for *all* $y^{(1)} \ldots y^{(h)}$, f(x) can achieve zero error on training data $(x^{(1)},y^{(1)})$, $(x^{(2)},y^{(2)})$, … $(x^{(h)},y^{(h)})$

  (i.e., there exists some θ that gets zero error)

- Can $f(x;\theta) = sign(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$ shatter these points?
- Yes: there are 4 possible training sets…

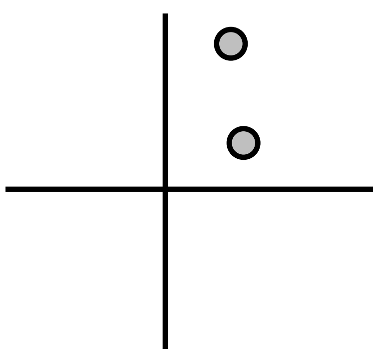


Shattering

- We say a learner f(x) can shatter points $x^{(1)} \ldots x^{(h)}$ iff for *all* $y^{(1)} \ldots y^{(h)}$, f(x) can achieve zero error on training data $(x^{(1)},y^{(1)})$, $(x^{(2)},y^{(2)})$, … $(x^{(h)},y^{(h)})$

  (i.e., there exists some θ that gets zero error)

- Can $f(x;\theta) = sign(x_1{}^2 + x_2{}^2 - \theta)$ shatter these points?

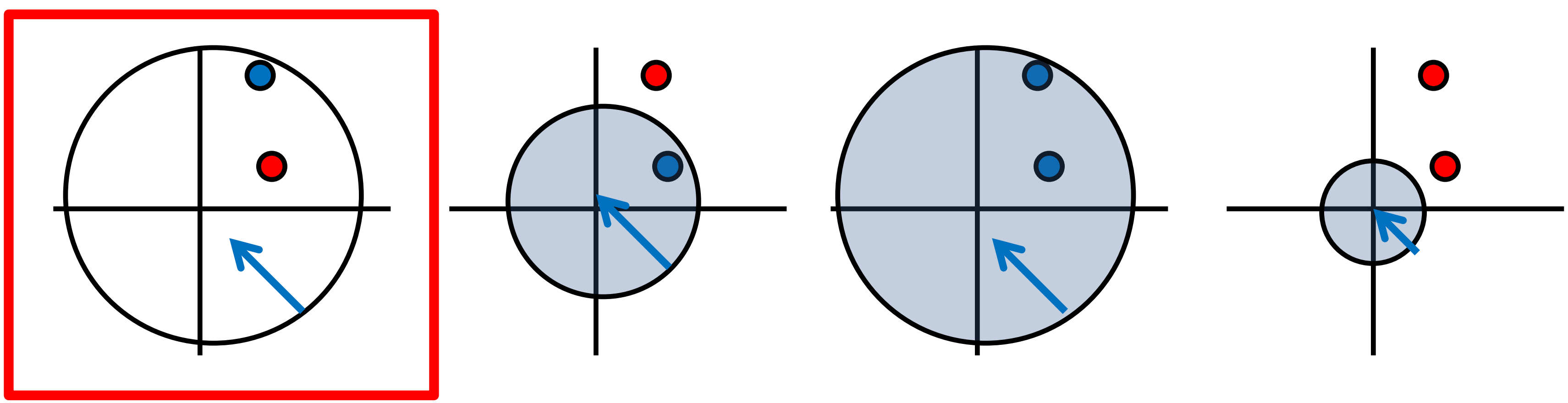


Shattering

- We say a learner f(x) can shatter points $x^{(1)} \ldots x^{(h)}$ iff for *all* $y^{(1)} \ldots y^{(h)}$, f(x) can achieve zero error on training data $(x^{(1)},y^{(1)})$, $(x^{(2)},y^{(2)})$, … $(x^{(h)},y^{(h)})$

  (i.e., there exists some θ that gets zero error)

- Can $f(x;\theta) = sign\big(\theta - ({x_1}^2 + {x_2}^2)\big)$ shatter these points?
- Nope!



VC Dimension

- The VC dimension H is defined as

  The maximum number of points h that *can be arranged* so that f(x) can shatter them

- A game:
  - Fix the definition of $f(x;\theta)$
  - Player 1: choose locations $x^{(1)} \ldots x^{(h)}$
  - Player 2: choose target labels $y^{(1)} \ldots y^{(h)}$
  - Player 1: choose value of θ
  - If $f(x;\theta)$ can reproduce the target labels, P1 wins

$$\exists \{x^{(1)} \ldots x^{(h)}\} \; s.t. \;\; \forall \{y^{(1)} \ldots y^{(h)}\} \;\; \exists \theta \; s.t. \;\; \forall i \;\; f(x^{(i)};\theta) = y^{(i)}$$

VC Dimension

- The VC dimension H is defined as

  The maximum number of points h that *can be arranged* so that f(x) can shatter them

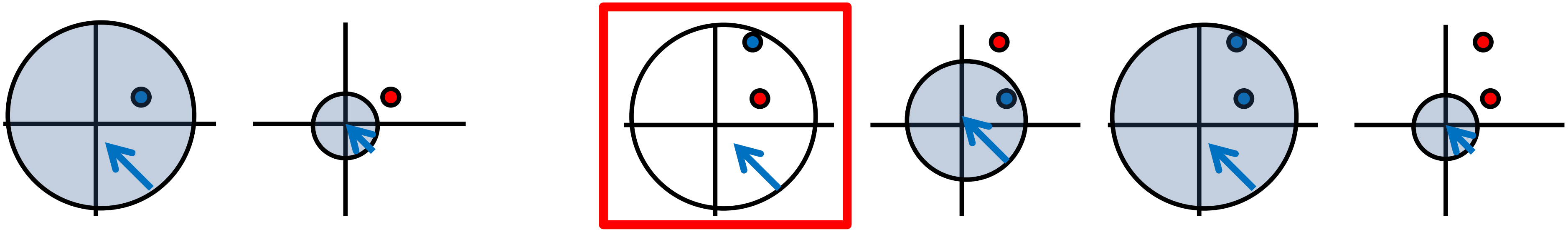
- Example: what's the VC dimension of the (zero-centered) circle, $f(x;\theta) = sign(x_1^2 + x_2^2 - \theta)$?

VC Dimension

- The VC dimension H is defined as

  The maximum number of points h that *can be arranged* so that f(x) can shatter them

- Example: what's the VC dimension of the (zero-centered) circle, $f(x;\theta) = sign(\theta - (x_1{}^2 + x_2{}^2))$?
- VCdim = 1 : can arrange one point, cannot arrange two (previous example was general)



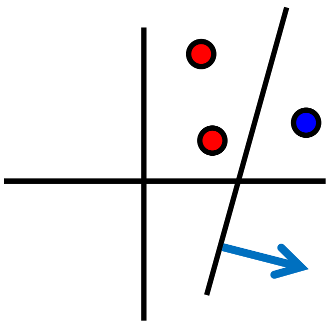
VC Dimension

- Example: what's the VC dimension of the two-dimensional line, $f(x;\theta) = sign(\theta_1 x_1 + \theta_2 x_2 + \theta_0)$?

VC Dimension

- Example: what's the VC dimension of the two-dimensional line, $f(x;\theta) = sign(\theta_1 x_1 + \theta_2 x_2 + \theta_0)$?

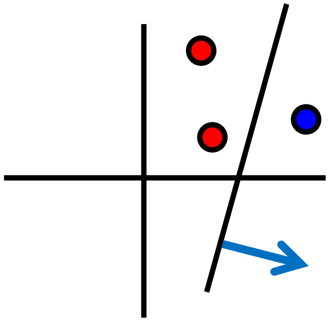
- VC dim >= 3?



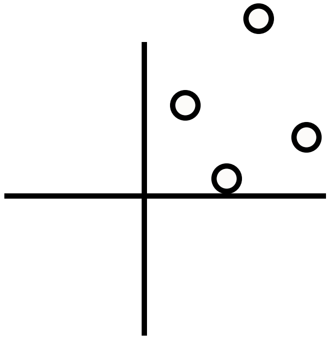
VC Dimension

- Example: what's the VC dimension of the two-dimensional line, $f(x;\theta) = sign(\theta_1 x_1 + \theta_2 x_2 + \theta_0)$?

- VC dim >= 3? Yes



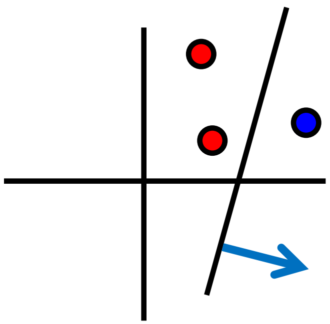
- VC dim >= 4?



VC Dimension

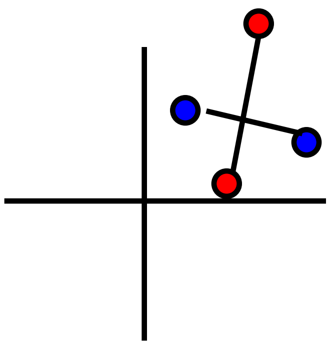
- Example: what's the VC dimension of the two-dimensional line, $f(x;\theta) = sign(\theta_1 x_1 + \theta_2 x_2 + \theta_0)$?

- VC dim >= 3? Yes



- VC dim >= 4? No…

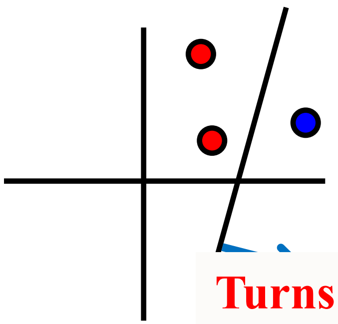
  Any line through these points must split one pair (by crossing one of the lines)



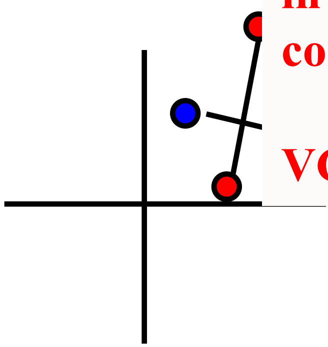
VC Dimension

- Example: what’s the VC dimension of the two-dimensional line, $f(x;\theta) = sign(\theta_1 x_1 + \theta_2 x_2 + \theta_0)$?

- VC dim >= 3? Yes



- VC dim >= 4? No…
  Any line through these points must split one pair (by crossing one of the lines)



Turns out:
For a general , linear classifier (perceptron) in d dimensions with a constant term:

VC dim = d+1

VC Dimension

- VC dimension measures the "power" of the learner
- Does *not* necessarily equal the # of parameters!

- Number of parameters does not necessarily equal complexity
  - Can define a classifier with a lot of parameters but not much power (how?)
  - Can define a classifier with one parameter but lots of power (how?)

- Lots of work to determine what the VC dimension of various learners is…
  - The VC dimension of neural networks with sigmoid activation functions is at most $O(|E|^2 \cdot |V|^2)$, and $O(|E|)$ if weights are limited to numbers that can represented by computer.

Using VC dimension

- Used validation / cross-validation to select complexity
- Use VC dimension based bound on test error similarly

- "Structural Risk Minimization" (SRM)

$$\text{TestError} \leq \text{TrainError} + \sqrt{\frac{H \log(2m/H) + H - \log(\eta/4)}{m}}$$