Introduction to Machine Learning (IML)

LECTURE #5:
STATISTICAL ASPECTS OF LEARNING

236756 – 2023-2024 WINTER – TECHNION

LECTURER: YONATAN BELINKOV

Today

- **part II**: *the different aspects of learning*
    1. Statistics: generalization and PAC theory (today)
    2. Modeling: model selection and evaluation
    3. Optimization: convexity, gradient descent
    4. Practical aspects and potential pitfalls
- (will mostly use SVM as use case)

SVM – wrap up

Duality – general case

- Hard SVM: $\text{argmin}_{w \in \mathbb{R}^d} \|w\|_2^2$ s.t. $y_i w^\top x_i \geq 1 \;\; \forall i \in [m]$
- **Lagrangian**: $L(w, \alpha) = \|w\|_2^2 - \sum_{i=1}^m \alpha_i (y_i w^\top x_i - 1), \alpha \in \mathbb{R}_+^m$ (multiple constraints => sum)
- Primal objective: $\boxed{\min_{w \in \mathbb{R}^d} \max_{\alpha \in \mathbb{R}_+^m} L(w, \alpha) \;\; = \;\; \max_{\alpha \in \mathbb{R}_+^m} \min_{w \in \mathbb{R}^d} L(w, \alpha)}$ (*dual objective*)

- **Dubious move**: swap min ↔ max
- In general, minmax ≥ maxmin ("max min inequality"; see wiki)
- **But**: convex in w (for fixed α) + concave in α (for fixed w) ⇒ **equality**!
- (aka minimax theorem; won't prove)
- **Bonus**: lies at core of game theory (zero-sum games); adversarial learning, GANs.

Aspects of learning:
Statistics and Generalization

Reasoning about generalization

Recall:

- **Want**: low $L_D(h) = \mathbb{P}_D[y \neq h(x)] = \mathbb{E}_D[\mathbb{1}\{y \neq h(x)\}]$
- **Have**: low $L_S(h) = \mathbb{P}_S[y \neq h(x)] = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}\{y_i \neq h(x_i)\}$
- **ERM:** $h_S = \operatorname*{argmin}_{h \in H} L_S(h) = A(S)$ (=output of learning *algorithm*)
- **Generalization**: $L_D(h_S) = L_D\big(A(S)\big)$

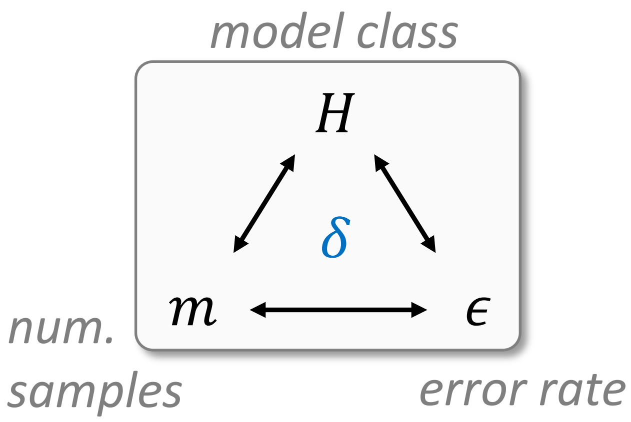
Reasoning about generalization

Recall:

- **Want**: low $L_D(h) = \mathbb{P}_D[y \neq h(x)] = \mathbb{E}_D[\mathbb{1}\{y \neq h(x)\}]$
- **Have**: low $L_S(h) = \mathbb{P}_S[y \neq h(x)] = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}\{y_i \neq h(x_i)\}$
- **ERM:** $h_S = \underset{h \in H}{\operatorname{argmin}}\, L_S(h) = A(S)$ (=output of learning *algorithm*)
- **Generalization**: $L_D(h_S) = L_D\big(A(S)\big)$
- **Today**: what can we say about $L_D(h_S)$? $\left\{ \begin{array}{l} \text{can't optimize} \\ \text{can't compute} \\ \textbf{can bound!} \\ \text{can estimate... but at some cost – next week!} \end{array} \right.$

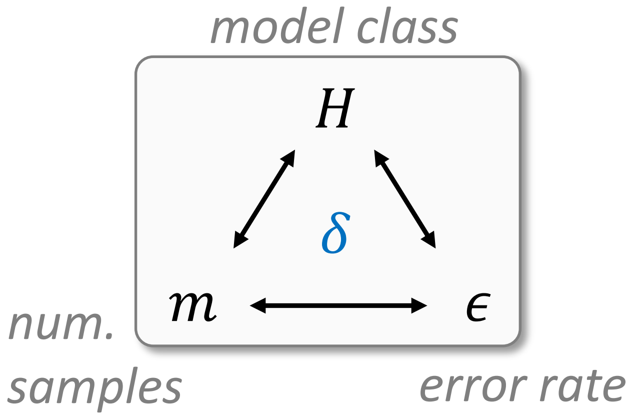
Statistical learning theory

- Theory (at large) can help *forecast* (think physics)
  (practically, theory can help plan and make decisions)
- We would like to "forecast" $L_D(h_S)$
- **Key players**: H, m, ϵ (<u>*not*</u> D!)
- Theory will help establish their relations



Statistical learning theory

- Theory (at large) can help *forecast* (think physics)
  (practically, theory can help plan and make decisions)
- We would like to "forecast" $L_D(h_S)$
- **Key players**: H, m, ϵ (not D!)
- Theory will help establish their relations
- What are useful forecasts for learning problems?
  1. Fixing H, for given m, what can we expect ϵ to be?
  2. Fixing H, to ensure error $\leq \epsilon$, how large must m be?
  3. For given m, to ensure error $\leq \epsilon$, what H can we use?
     (for halfspaces – how large can d (=num. features) be?)
- [from here – on board]



PAC learning: Realizable case

- Assume **Realizability**: $\exists h \in H$ s.t. $L_D(h) = 0$
  - $\Rightarrow L_D(h^*) = 0$
  - $\Rightarrow L_S(h_S) = 0$
- **Want**: $P_{S\sim D^m}(L_D(h_S) \geq \epsilon) \leq ?$ (upper bound on probability of finding a bad model)
- Assume **finite** H:

$$P_S(L_D(h_S) \geq \epsilon) =_{ERM} P_S(L_D(h_S) \geq \epsilon, L_S(h_S) = 0)$$

$$\leq P_S(\exists h \in H \;\; L_D(h) \geq \epsilon, L_S(h) = 0) = P_S(\cup_{h\in B} L_S(h) = 0) \qquad [B = \{h \in H : L_D(h) \geq \epsilon\}]$$

$$\leq_{union\ bound} \; \sum_{h\in B} P_S(L_S(h) = 0) = \sum_{h\in B} P_S(\forall i \in [m] \;\; h(x_i) = y_i)$$

$$=_{iid} \sum_{h\in B} \prod_i P_D(h(x) = y) \leq_{h\in B} \sum_{h\in B} (1-\epsilon)^m \leq |B|e^{-\epsilon m} \leq_{worst\ case} |H|e^{-\epsilon m}$$

PAC learning: Realizable case

- **Got**: $P_{S\sim D^m}(L_D(h_S) \geq \epsilon) \leq |H|e^{-\epsilon m} \leq \delta$ (We ask: when bounded by some δ)

$e^{-\varepsilon m} \leq \frac{\delta}{|H|} \Rightarrow -\varepsilon m \leq \log \frac{\delta}{|H|} \Rightarrow m \geq \frac{\log|H| + \log\frac{1}{\delta}}{\varepsilon}$

1. $m \geq \frac{\log|H| + \log\frac{1}{\delta}}{\epsilon}$
2. $\epsilon \geq \frac{\log|H| + \log\frac{1}{\delta}}{m}$

- **PAC:** H is PAC-learnable if $\exists A, \exists\, m_H(\epsilon,\delta) \in \mathrm{poly}(\frac{1}{\epsilon},\frac{1}{\delta})$ such that $\forall D$ (for which H is realizable) and $\forall \epsilon,\delta \in [0,1]$, if $m \geq m_H(\epsilon,\delta)$, then:

$$P_{S\sim D^m}(L_D(h_S) \geq \epsilon) \leq \delta$$

- ϵ = "approximately" correct
- δ = "probably" correct
- **PAC = Probably Approximately Correct**
- $m_H(\epsilon,\delta)$ = sample complexity

Agnostic PAC learning

- **Let's drop realizability**
- We'll still look at **finite** H

- **Definition**:
  H is Agnostic-PAC-learnable if $\exists A, \exists\, m_H(\epsilon,\delta) \in \text{poly}\left(\frac{1}{\epsilon},\frac{1}{\delta}\right)$
  such that $\forall D$ and $\forall \epsilon,\delta \in [0,1]$, if $m \geq m_H(\epsilon,\delta)$, then:
  $$P_{S\sim D^m}\left(L_D(h_S) - L_D(h^*) \geq \epsilon\right) \leq \delta$$

- Compare with PAC-learnable (**realizable** case):
  H is PAC-learnable if …
  $$P_{S\sim D^m}\left(L_D(h_S) \geq \epsilon\right) \leq \delta$$

Agnostic PAC learning

- **Want**: $P_{S\sim D^m}(L_D(h_S) - L_D(h^*) \geq \epsilon) \leq \delta$ (*Agnostic* PAC)

- **Def**: S is ϵ-representative if $|L_S(h) - L_D(h)| \leq \epsilon \quad \forall h \in H$

- **Lemma**: S is $\frac{\epsilon}{2}$-rep. $\Rightarrow L_D(h_S) - L_D(h^*) \leq \epsilon$ [1]
- **Proof**: $L_D(h_S) \leq_{rep} L_S(h_S) + \frac{\epsilon}{2} \leq_{ERM} \underbrace{L_S(h^*) + \frac{\epsilon}{2}}_{-L_D(h^*)+L_D(h^*)} \leq_{rep} L_D(h^*) + \epsilon$

- **Hoeffding concentration bound**: let $z \in [0,1]$ be a random variable. Denote the mean by $\bar{z} = \frac{1}{m}\sum_i z_i$ and expectation $\mu = \mathbb{E}[z]$. then:

$$P(|\bar{z} - \mu| \geq \epsilon) \leq 2e^{-2m\epsilon^2} \qquad [2]$$

Agnostic PAC learning

[1] lemma: $S\,\frac{\epsilon}{2}\text{rep.} \Rightarrow L_D(h_S) - L_D(h^*) \le \epsilon$
[2] Hoeffding: $P(|\bar{z} - \mu| \ge \epsilon) \le 2e^{-2m\epsilon^2}$

Finite H:

- **Want**: $P_{S\sim D^m}(L_D(h_S) - L_D(h^*) \ge \epsilon) \le \delta$ (*Agnostic* PAC)
- Equivalently: $P_{S\sim D^m}(L_D(h_S) - L_D(h^*) \le \epsilon) \ge 1 - \delta$

- By lemma [1], enough to show $P_S\left(S\ is\ \frac{\epsilon}{2} - rep.\right) \ge 1 - \delta$
- Equivalently, enough to show $P_S\left(S\ is\ \boldsymbol{not}\ \frac{\epsilon}{2} - rep.\right) \le \delta$
- $P_S\left(S\ is\ \boldsymbol{not}\ \frac{\epsilon}{2} - rep.\right) = P_S\left(\exists h \in H\ \ |L_S(h) - L_D(h)| > \frac{\epsilon}{2}\right)$
  $\le\ P_S(\cup_{h\in H}\ |L_S(h) - L_D(h)| > \epsilon/2) \le_{UB} \sum_{h\in H} P_S(|L_S(h) - L_D(h)| > \epsilon/2)$
  $\le_{Hoeffding}\ |H|2e^{-2m(\epsilon/2)^2} \le |H|2e^{-m\epsilon^2/2}$

- **Got**: $P_{S\sim D^m}(L_D(h_S) - L_D(h^*) \ge \epsilon) \le |H|2e^{-m\epsilon^2/2} \le \delta$ <- fix

Agnostic PAC learning

Finite $\boldsymbol{H}$:

Agnostic case:

- $P_{S\sim D^m}(L_D(h_S) - L_D(h^*) \geq \epsilon) \leq |H| 2e^{-m\epsilon^2/2} \leq \delta$

1. $m \geq \frac{2\log 2|H| + \log\frac{1}{\delta}}{\epsilon^2}$
2. $\epsilon \geq \sqrt{\frac{2\log 2|H| + \log\frac{1}{\delta}}{m}} \approx \frac{1}{\sqrt{m}}$

Compare with realizable case:

- $P_{S\sim D^m}(L_D(h_S) \geq \epsilon) \leq |H| e^{-\epsilon m} \leq \delta$

1. $m \geq \frac{\log|H| + \log\frac{1}{\delta}}{\epsilon}$
2. $\epsilon \geq \frac{\log|H| + \log\frac{1}{\delta}}{m} \approx \frac{1}{m}$

Beyond finite classes

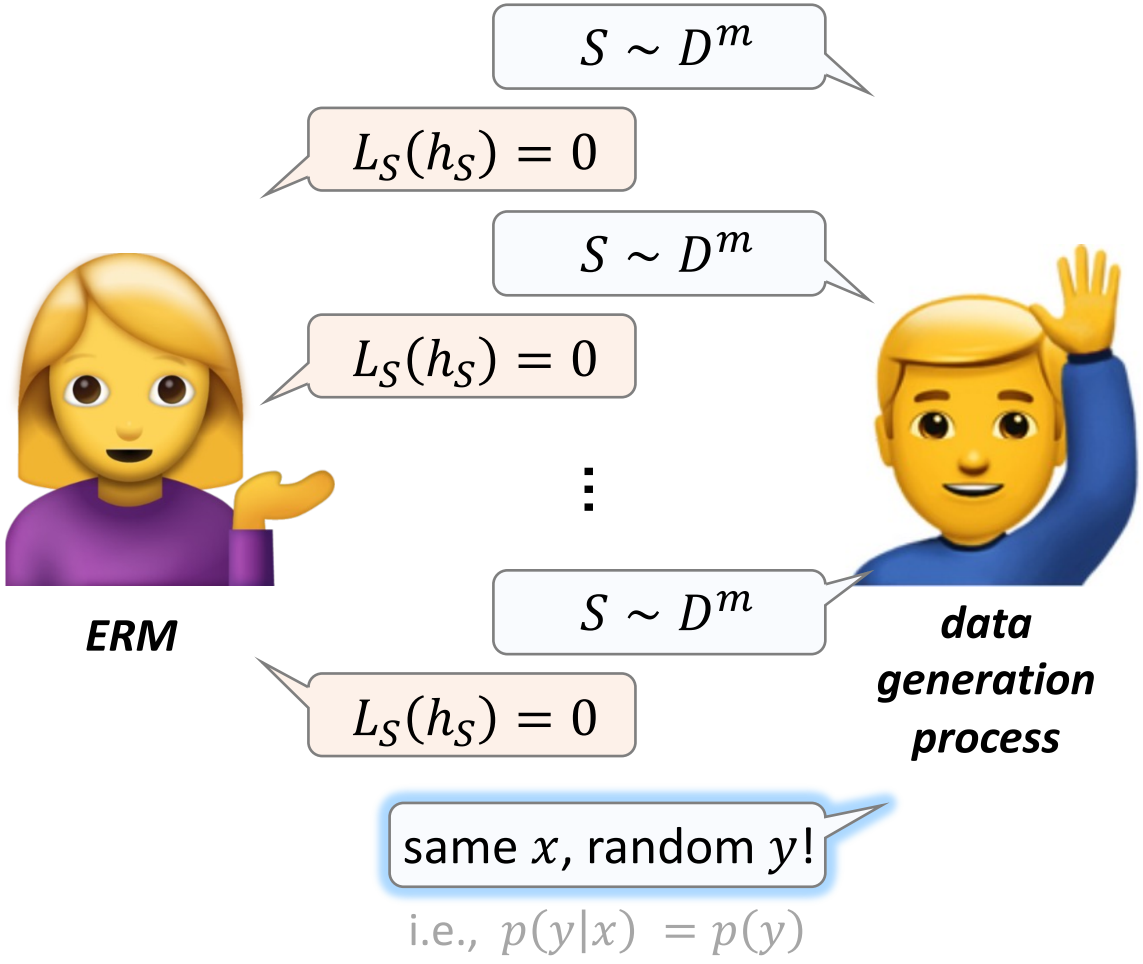
- The previous bound characterizes learnability of H using $\log|H|$
- Is this bound useful for…
  - decision trees? (think!)
  - linear halfspaces? (think!)
  - RBF kernels? (think!)
- **Q**: If $|H| = \infty$, should we give up?
- **A**: Not necessarily!
- **Recall**: for 1D thresholds (infinite class!), we showed $\epsilon \approx O\left(\frac{1}{m}\right)$ (under realizability)
- **Conclusion**: $|H|$ is probably not the "correct" measure
- **Note**: there is no single "correct" measure, only *useful* measures; we will see one next

VC dimension

- **Idea:**
  consider not what each h <u>is</u>, but what it <u>does</u>

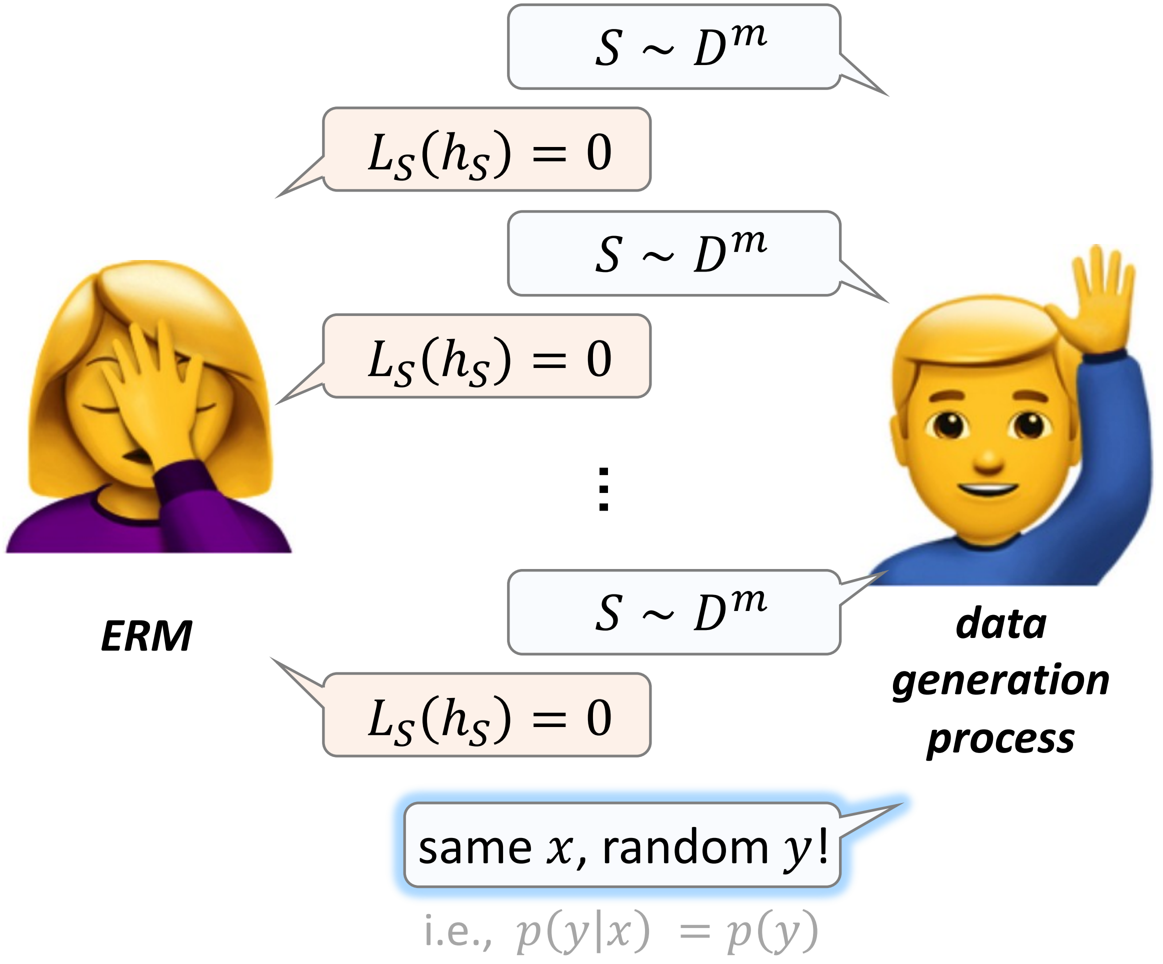
VC dimension

- **Idea:**
  consider not what each h is, but what it does
- **Intuition – when learning fails**



VC dimension

- **Idea:**
  consider not what each h is, but what it does
- **Intuition – when learning fails**
- **Take away:**
  "explaining everything ≡ explaining nothing"
- **VC theory** quantifies this idea (Vapnik–Chervonenkis)
- The **VC dimension** of H is the largest set on which $L_S = 0$ is possible for any labeling
- **Main result**: learning breaks once H can perfectly fit arbitrary label assignments (= noise! Remember overfitting?)



VC dimension

- The notion of "explaining everything" is defined using *shattering*.

- **Definition:** Let $C = \{x_i\}_{i=1}^m \in \mathcal{X}^m$, then *$H$ shatters C* if:

$$\forall \{y_i\} \in \{\pm 1\}^m \quad \exists\, h \in H \quad \text{s.t.} \quad h(x_i) = y_i \;\forall i \in [m]$$

i.e., for any labeling of C, applying ERM to $S(C) = \{(x_i, y_i)\}_{i=1}^m$ gives $L_{S(C)}\left(h_{S(C)}\right) = 0$.

- **Definition**: The *VC-dimension of H* is the size of the largest set that H shatters, denoted $VC(h)$

VC dimension

- The notion of "explaining everything" is defined using *shattering*.

- **Definition:** Let $C = \{x_i\}_{i=1}^m \in \mathcal{X}^m$, then *$H$ shatters C* if:

  $\forall \{y_i\} \in \{\pm 1\}^m \quad \exists\, h \in H \quad \text{s.t.} \quad h(x_i) = y_i \ \forall i \in [m]$

i.e., for any labeling of C, applying ERM to $S(C) = \{(x_i, y_i)\}_{i=1}^m$ gives $L_{S(C)}\left(h_{S(C)}\right) = 0$.

- **Definition**: The *VC-dimension of H* is the size of the largest set that H shatters, denoted $VC(h)$

- **Fundamental theorem of learning:** (partial; won't prove)

  If $VC(H) < \infty$, then H is:

  1. **PAC-learnable** with

     vs. $\log|H|$

     $$m_H(\epsilon, \delta) = \Theta\left(\frac{VC(H) \log 1/\epsilon + \log 1/\delta}{\epsilon}\right)$$

  2. **Agnostic PAC-learnable** with

     $$m_H(\epsilon, \delta) = \Theta\left(\frac{VC(H) + \log 1/\delta}{\epsilon^2}\right)$$

what about $|H| = \infty$*?*

exact characterization

(almost) same ϵ, δ rates as in finite H

Finding VC

- **Rules of the game**: find m such that
  1. Exists C of size m that H shatters
  2. H does not shatter all sets C of size $m+1$

$\Rightarrow$

m:	$\exists\, x\, \forall\, y$ shatter
$m+1$:	$\forall\, x\, \exists\, y$ not shatter

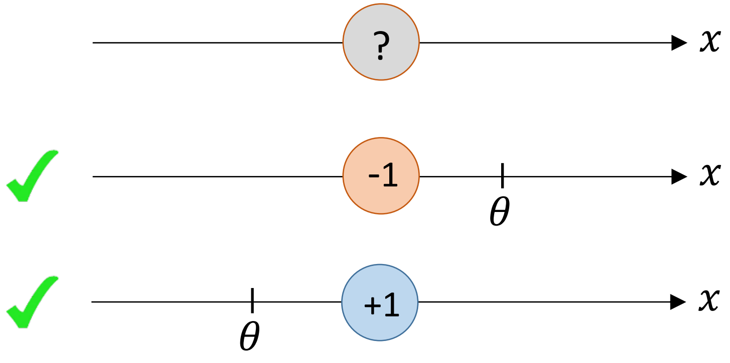
- **Examples:**
  1. 1D thresholds [on board]
  2. 1D intervals [on board]
  3. Linear halfspaces? (tirgul!)
  4. RBF kernel? (think!)

Example: Threshold functions

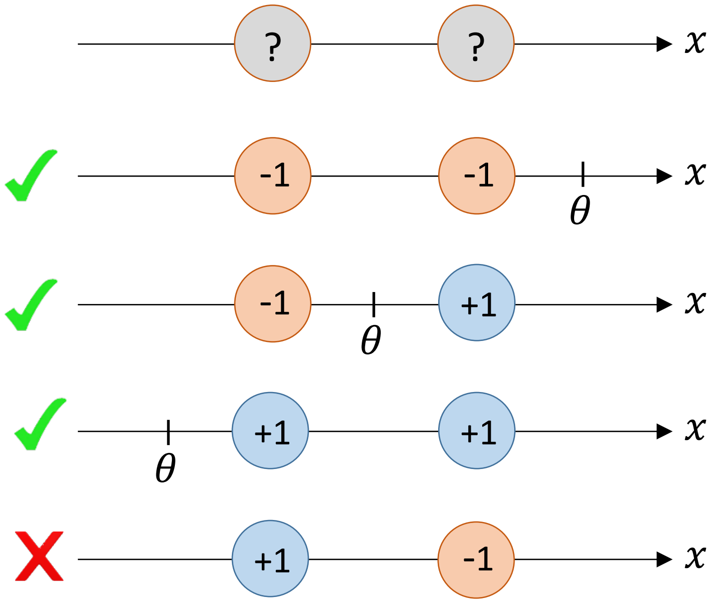
- In the lecture, we defined the following hypothesis class:

$$\mathcal{X} = \mathbb{R}, \qquad \mathcal{H} = \{x \mapsto sign(x - \theta) : \theta \in \mathbb{R}\}$$

- There exists a single point which is shattered:



- Any two points cannot be shattered

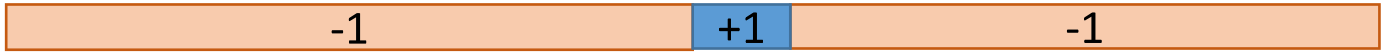


$$\Longrightarrow \text{VCdim}(\mathcal{H}) = 1$$

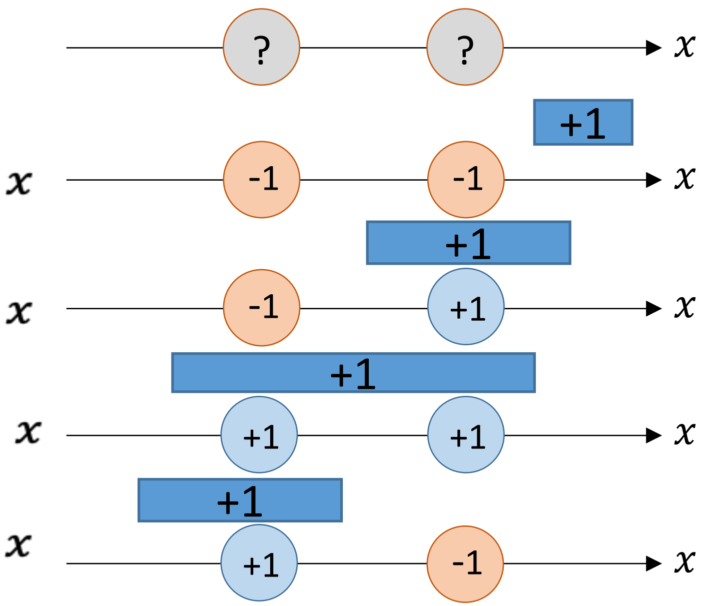
Example: 1-D Intervals

- Recall the following hypothesis class:

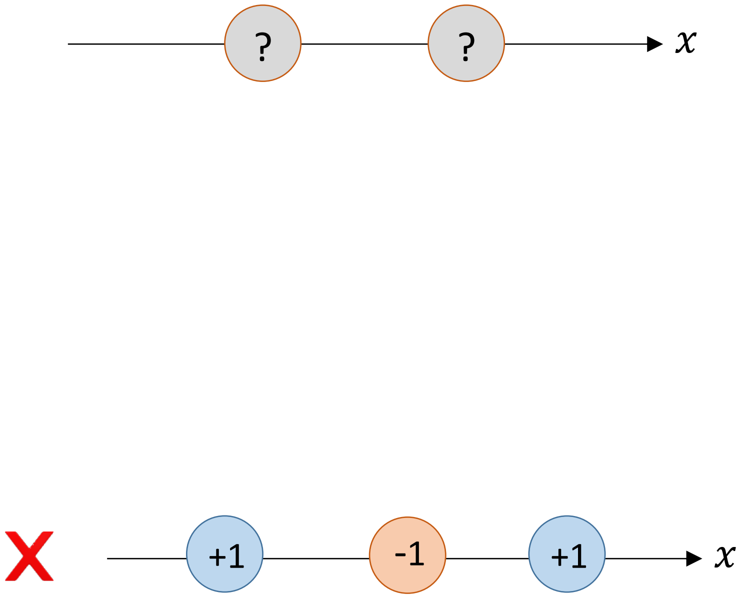
$$\mathcal{H} = \{h_{a,b}\} \qquad h_{a,b}(x) = \mathbb{1}\{x \in (a, b\,)\}$$



- There exists a set of two points which is shattered:



- Any three points cannot be shattered



$$\Longrightarrow \mathrm{VCdim}(\mathcal{H}) = 2$$

Discussion

- The learnability of H depends on how "expressive" it is
- We saw that $|H|$ is a good measure for finite classes
- For infinite classes (e.g., hyperplanes), **VC theory** can establish learnability

- **The VC dimension:**
  - measures the *capacity* of model classes to express binary patterns
  - shows that full capacity is right where *learning breaks*
  - is a *combinatorial measure* – no statistics involved!
  - works because binary classification is a discrete problem:
    - Reveals that what's important is possible ways to label
    - Hints that solving ERM "requires" searching over this combinatorial space
  - is *worst-case* measure – price of being distribution-independent!

Uses and limitations

- Say you want to learn with SVM (and assume you know the VC of halfspaces*)
- **Theory is your friend:**
  - **Theory asks**: tell me your desired ϵ and δ (this is unavoidable!)
  - **Theory says**: you need (order of) $m = m_H(\epsilon, \delta)$ examples!

- **Great, but need to remember:**

1. VC and PAC are worst-case (are you really doomed if you only have $< m$ samples?)
2. ERM is (computationally) hard! SVM minimizes *hinge loss*, not 0/1 loss (We assumed exact ERM)
3. Even agnostic PAC relies on distributional assumptions (the elephant in the room: i.i.d.)
4. Guarantees are probabilistic but (in most cases) **you only see one sample set** -> next week

*Wait till tirgul

Beyond VC

- Other statistical learning approach exist that:
  - Can relate proxy losses (e.g., hinge) to 0/1 loss
    (Bonus: Rademacher complexities are data-dependent and use smoothness)
  - Work for classes with $VC = \infty$
    (Bonus: margin-based bounds show RBF is learnable when margin is "large enough")
  - Apply to non-ERM algorithms
    (Bonus: a learning algorithm is useful if it is "stable" under small changes to the data)
  - …
- (Not to worry – these are all outside of our scope)

Next week(s)

- **Part II**: *the different aspects of learning*
  1. ~~Statistics: generalization and PAC theory~~
  2. Modeling: model selection and evaluation
  3. Optimization: convexity, gradient descent
  4. Practical aspects and potential pitfalls
- (will mostly use SVM as use case)