Active Learning and Optimized Information Gathering

Lecture 7 – Learning Theory

CS 101.2
Andreas Krause

Announcements

- Project proposal: Due ~~tomorrow~~ today 1/27
- Homework 1: Due Thursday 1/29
  - Any time is ok.
- Office hours
  - Come to office hours before your presentation!
  - Andreas: **Monday 3pm-4:30pm**, 260 Jorgensen
  - Ryan: Wednesday 4:00-6:00pm, 109 Moore

Recap Bandit Problems

- Bandit problems
  - Online optimization under limited feedback
- Exploration—Exploitation dilemma
- Algorithms with low regret:
  - ε-greedy, UCB1
- Payoffs can be
  - Probabilistic
  - Adversarial (oblivious / adaptive)

More complex bandits

- Bandits with many arms
  - Online linear optimization (online shortest paths …)
  - X-armed bandits (Lipschitz mean payoff function)
  - Gaussian process optimization (Bayesian assumptions about mean payoffs)
- Bandits with state
  - Contextual bandits
  - Reinforcement learning
- **Key tool**: Optimism in the face of uncertainty ☺

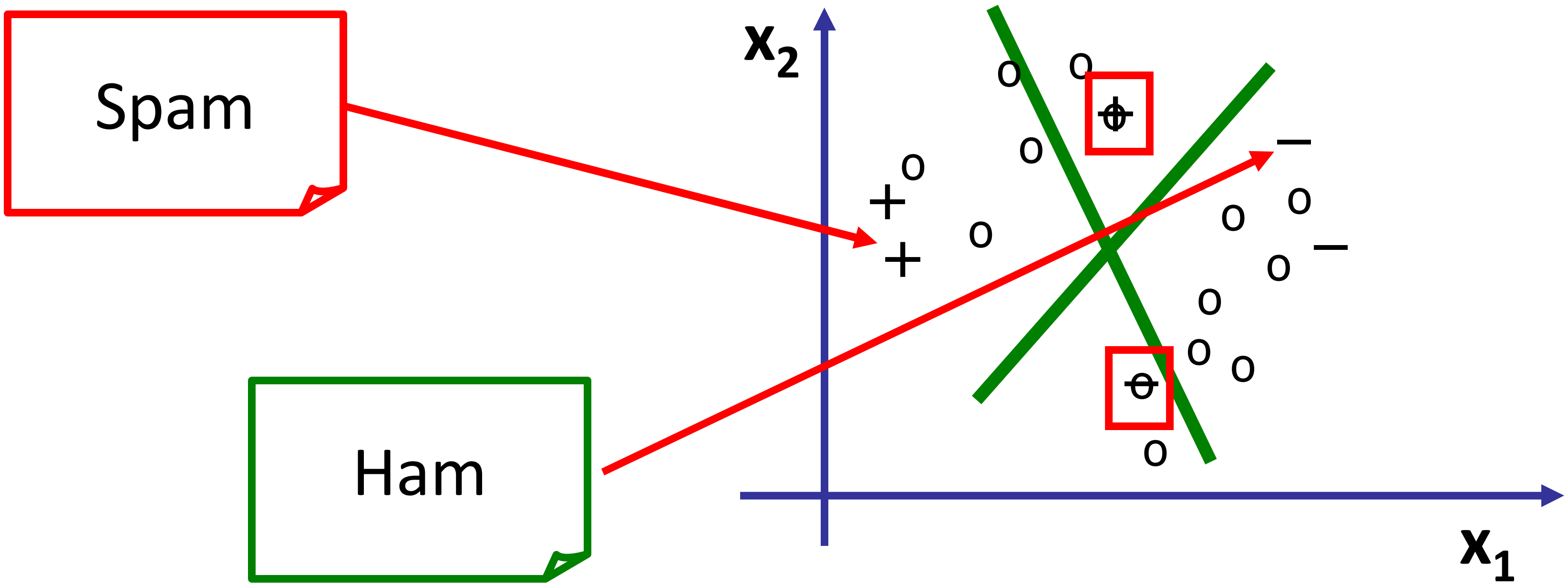
Course outline

1. Online decision making

2. Statistical active learning

3. Combinatorial approaches

Spam or Ham?



label = sign($w_0 + w_1 x_1 + w_2 x_2$)
(linear separator)

- Labels are expensive (need to ask expert)
- **Which labels should we obtain to maximize classification accuracy?**

Outline

- Background in learning theory
- Sample complexity
- Key challenges
- Heuristics for active learning
- Principled algorithms for active learning

Credit scoring

Credit score	Defaulted?
70	0
42	1
36	1
82	0
50	**???**

Want decision rule that performs well for unseen examples (*generalization*)

More general: Concept learning

- Set X of instances

  $X = \{1, \ldots, 100\}$

- True concept c: X → {0,1}

  $c(x) = 1$ if $x > t$
  $c(x) = 0$ if $x \leq t$

- Hypothesis h: X → {0,1}

  $h(x) = 1$ if $x \geq t'$
  $h(x) = 0$ if $x \leq t'$

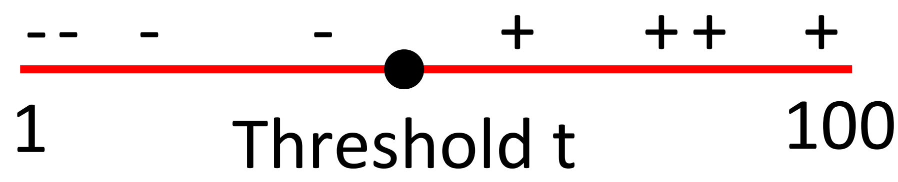
- Hypothesis space H = {h_1, …, h_n, …}

- Want to pick good hypothesis
  - (agrees with true concept on most instances)

Example: Binary thresholds

- Input domain: X={1,2,…,100}
- True concept c:

$$c(x) = +1 \text{ if } x \geq t$$
$$c(x) = -1 \text{ if } x < t$$



How good is a hypothesis?

- Set X of instances, concept c: X → {0,1}
- Hypothesis h: X → {0,1} , H = {h_1, …, h_n, …}
- Distribution P_X over X

don't know

- $\text{error}_{true}(h)$ = $P(\{x \in X : h(x) \neq c(x)\}) = E_{x \sim P_X}[\,|h(x) - c(x)|\,]$
- Want $h^* = \text{argmin}_{h \in H}\ \text{error}_{true}(h)$
- **Can't compute $\text{error}_{true}(h)$!**

Concept learning

- Data set $D = \{(x_1,y_1),\ldots,(x_N,y_N)\}$, $x_i \in X$, $y_i \in \{0,1\}$
- Assume x_i drawn **independently** from P_X; $y_i = c(x_i)$
- Also assume $c \in H$

- h consistent with D $\Leftrightarrow$ $\forall i\ h(x_i) = y_i$

- More data ➔ fewer consistent hypotheses

Learning strategy:

- Collect "enough" data
- Output consistent hypothesis h
- Hope that $\text{error}_{\text{true}}(h)$ is small

Sample complexity

- Let ε>0
- How many samples do we need s.t. **all** consistent hypotheses have error< ε??
- Def: h ∈ H bad ⇔ $error_{true}(h) > \varepsilon$
- Suppose h∈ H is bad. Let x ∼ P_X, y = c(x).

  Then: $P(h(x) \neq c(x)) \geq \varepsilon$

Sample complexity

- P(h bad and "survives" 1 data point) = $\leq 1-\varepsilon$
- P(h bad and "survives" n data points) = $\leq (1-\varepsilon)^n$

$H = \{h_1 \ldots h_N\}$

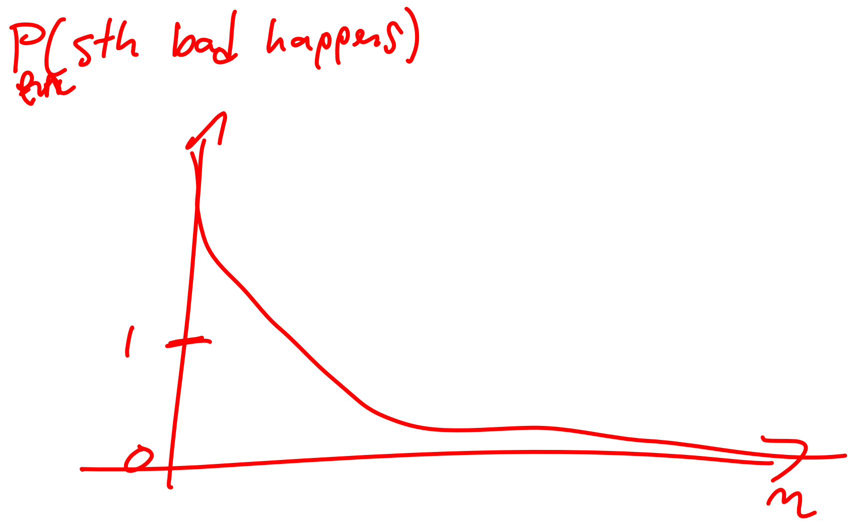
- P(remains $\geq$ 1 bad h after n data points) =

$P(h_1 \text{ bad} \vee h_2 \text{ bad} \vee h_3 \text{ bad} \ldots \vee h_N \text{ bad}) \leq$

$P(h_1 \text{ bad}) + P(h_2 \text{ bad}) + \ldots + P(h_N \text{ bad}) \leq |H| (1-\varepsilon)^n$

Probability of bad hypothesis

$$P\left(\text{remains} \geq 1 \text{ bad hypothesis after } n \text{ data points}\right) \leq |H|\,\underbrace{(1-\varepsilon)^n}_{} \leq \exp(-\varepsilon n)\cdot|H|$$



Sample complexity for finite hypothesis spaces [Haussler '88]

Theorem: Suppose

- $|H| < \infty$,
- Data set $|D|=n$ drawn i.i.d. from P_X (no noise)
- $0<\varepsilon<1$

Then for any $h \in H$ consistent with D:

$$P(\, error_{true}(h) > \varepsilon) \leq |H| \exp(-\varepsilon\, n) = |H| e^{-\varepsilon n}$$

"PAC-bound" (probably approximately correct)

How can we use this result?

$$P(\ error_{true}(h) \geq \varepsilon\ ) \leq |H|\ \exp(-\varepsilon\ n) = \delta$$

Possibilities:

- Given δ, n solve for ε
- Given ε and δ, solve for n
- (Given ε, n, solve for δ)

E.g.: $|H|\, e^{-\varepsilon n} \leq \delta$

$\Leftrightarrow \log|H| - \varepsilon n \leq \log \delta$

$\Leftrightarrow \varepsilon n \geq \log|H| + \log \frac{1}{\delta}$

$\Leftrightarrow n \geq \frac{1}{\varepsilon}\left(\log|H| + \log\frac{1}{\delta}\right)$

Example: Credit scoring

- X = {1,2,...1000}
- H = binary thresholds on X
- |H| = 1001
- Want error $\leq$ 0.01 with probability .999

Need n $\geq$ 1382 samples

Limitations

- How do we **find** consistent hypothesis?

- What if $|H| = \infty$?

- What if there's noise in the data? (or $c \notin H$)

Credit scoring

Credit score	Defaulted?
36	1
48	0
52	1
70	0
81	0
44	**???**

No binary threshold function explains this data with 0 error

Noisy data

- Sets of instances X and labels Y = {0,1}
- Suppose $(X,Y) \sim P_{XY}$
- Hypothesis space H

$$\text{error}_{true}(h) = E_{x,y}[\ |h(x) - y|\ ] = P(\{(x,y) : h(x) \neq y\})$$

Want to find

$$\text{argmin}_{h \in H}\ \text{error}_{true}(h)$$

Learning from noisy data

- Suppose $D = \{(x_1,y_1),\ldots,(x_n,y_n)\}$ where $(x_i,y_i) \sim P_{X,Y}$

$$\text{error}_{true}(h) = E_{(x,y)\sim P_{X,Y}} |h(x) - y|$$

$$\text{error}_{train}(h) = (1/n) \sum_i |h(x_i) - y_i|$$

sample avg.

Learning strategy with noisy data

- Collect "enough" data
- Output $h' = \text{argmin}_{h \in H} \text{error}_{train}(h)$
- Hope that $\text{error}_{true}(h') \approx \min_{h \in H} \text{error}_{true}(h)$

Estimating error

- How many samples do we need to accurately estimate the true error?

- Data set $D = \{(x_1,y_1),\ldots,(x_n,y_n)\}$ where $(x_i,y_i) \sim P_{X,Y}$

  $z_i = |h(x_i) - y_i| \in \{0,1\}$
- z_i are i.i.d. samples from Bernoulli RV $Z = |h(X) - Y|$

$\text{error}_{train}(h) =$ $\frac{1}{n}\sum_{i=1}^{n} z_i$

$\text{error}_{true}(h) =$ $\hat{E}[Z]$

How many samples s.t.

$|\text{error}_{train}(h) – \text{error}_{true}(h)|$ is small??

Estimating error

How many samples do we need to accurately estimate the true error?

Applying Chernoff-Hoeffding bound:

$$P(\ |error_{true}(h) - error_{train}(h)| \geq \varepsilon) \leq \exp(-2n\ \varepsilon^2)$$

for no noise
$\leq \exp(-\varepsilon n)$

Sample complexity with noise

Call $h \in H$ bad if

$$\text{error}_{\text{true}}(h) > \text{error}_{\text{train}}(h) + \varepsilon$$

P(h bad "survives" n training examples) $\leq \exp(-2\, n\, \varepsilon^2)$

P(remains $\geq$ 1 bad h after n examples) $\leq |H| \exp(-2 n \varepsilon^2)$

PAC Bound for noisy data

Theorem: Suppose

- $|H| < \infty$,
- Data set $|D|=n$ drawn i.i.d. from P_{XY}
- ~~$0<\varepsilon<1$~~

Then for ~~any~~ all $h \in H$ it holds that with prob $1-\delta$

$$error_{true}(h) \leq error_{train}(h) + \underbrace{\sqrt{\frac{\log|H| + \log 1/\delta}{2n}}}_{\varepsilon}$$

PAC Bounds: Noise vs. no noise

Want error $\leq \varepsilon$ with probability 1-δ

No noise: $n \geq 1/\varepsilon \, (\log |H| + \log 1/\delta)$

Noise: $n \geq 1/\varepsilon^2 \, (\log |H| + \log 1/\delta)$

Limitations

How do we **find** consistent hypothesis?

What if |H| = ∞?

What if there's noise in the data? (or $c \notin H$)

Credit scoring

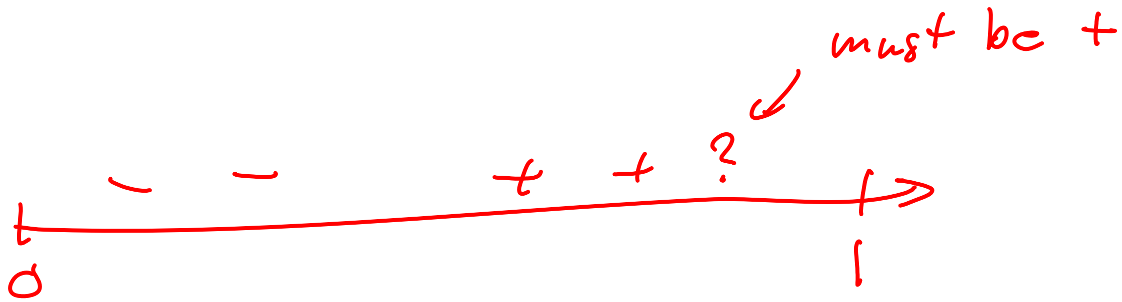
Credit score	Defaulted?
36.1200	1
48.7983	1
52.3847	1
70.1111	0
81.3321	0
44.3141	**???**

Want to classify continuous instance space

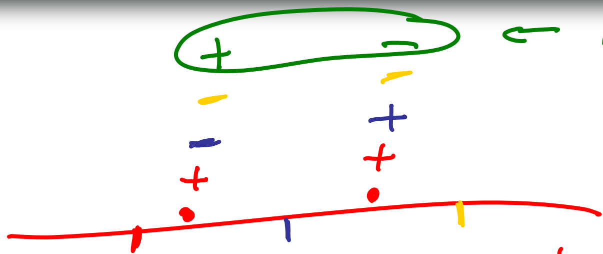
$$|H| = \infty$$

Large hypothesis spaces

- **Idea**: Labels of few data points imply labels of many unlabeled data points

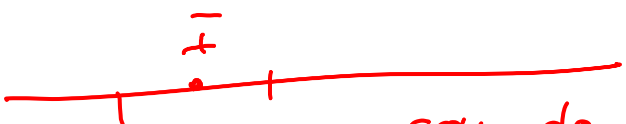


How many points can be *arbitrarily classified* using binary thresholds?



← no threshold classifier consistent with this labeling

cannot do 2 data points



can do 1 data point

$h(x) = 1$ if $x > t$

$\quad\;\; 0$ if $x < t$

What if $w_1 \in \{-1, 1\}$

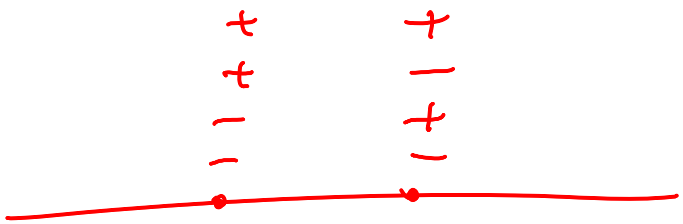
$h(x) = 1$ if $w_1 \cdot x > w_0$

$\quad\;\; 0$ if $w_1 x < w_0$

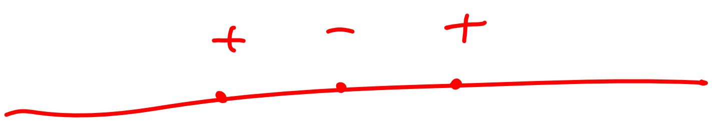
Can now realize

How many points can be *arbitrarily classified* using linear separators? (1D)

$h(x) = \text{sign}(w_0 + w_1 \cdot x)$



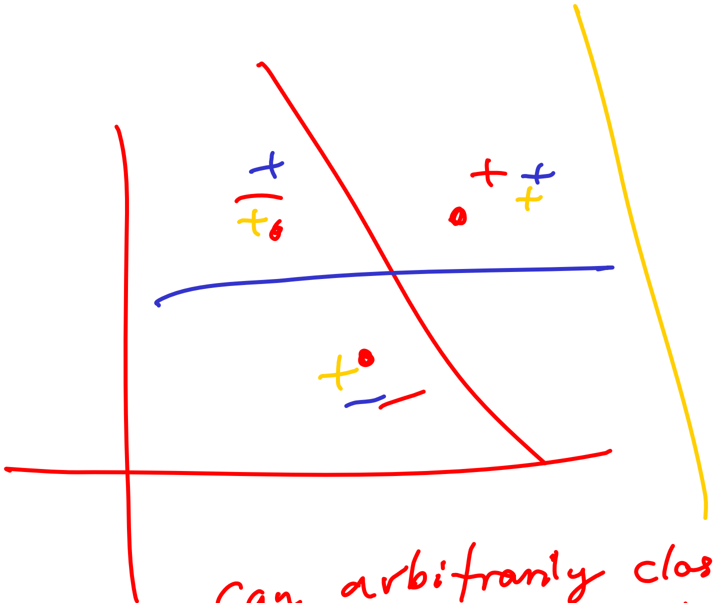
can arbitrarily classify any two data points

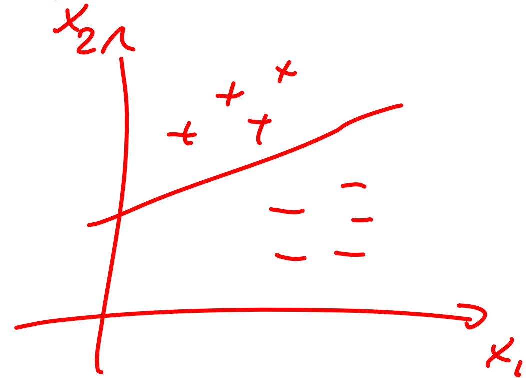


cannot arbitrarily classify any three data points

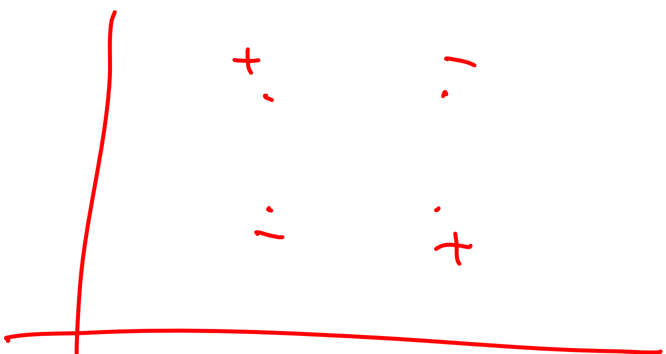
How many points can be *arbitrarily classified* using linear separators? (2D)

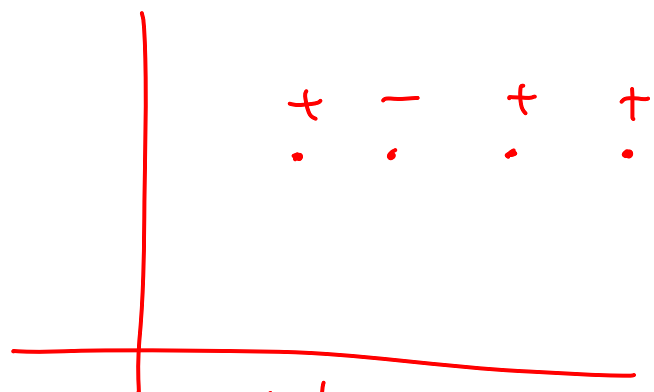
$h(x) = \text{sign}(w_0 + w_1 x_1 + w_2 x_2)$





Can arbitrarily classify 3 data points





cannot arbitrarily classif 4 data points

VC dimension

- Let $S \subseteq X$ be a set of instances
- A **Dichotomy** is a nontrivial partition of $S = S_1 \cup S_0$

  "labeled +" "labeled -"
- S is **shattered** by hypothesis space H if for any dichotomy, there exists a consistent hypothesis h (i.e., $h(x)=1$ if $x \in S_1$ and $h(x)=0$ if $x \in S_0$)

  $S_1 \cap S_0 = \emptyset$ $|S_i| > 0$

- The **VC (Vapnik-Chervonenkis) dimension** VC(H) of H is the size of the largest set S shattered by H (possibly ∞)

- $VC(H) \leq \log |H|$

VC Generalization bound

Bound for finite hypothesis spaces

$$error_{true}(h) \leq error_{train}(h) + \sqrt{\frac{\log |H| + \log 1/\delta}{2n}}$$

VC-dimension based bound

$\leq \log|H|$

$$error_{true}(h) \leq error_{train}(h) + \sqrt{\frac{VC(H)\left(1 + \log \frac{2n}{VC(H)}\right) + \log \frac{4}{\delta}}{n}}$$

Applications

- Allows to prove generalization bounds for large hypothesis spaces with structure.
- For many popular hypothesis classes, VC dimension known
  - Binary thresholds
  - Linear classifiers
  - Decision trees
  - Neural networks

Passive learning protocol

Data source $P_{X,Y}$ (produces inputs x_i and labels y_i)

⇩

Data set $D_n = \{(x_1,y_1),\ldots,(x_n,y_n)\}$

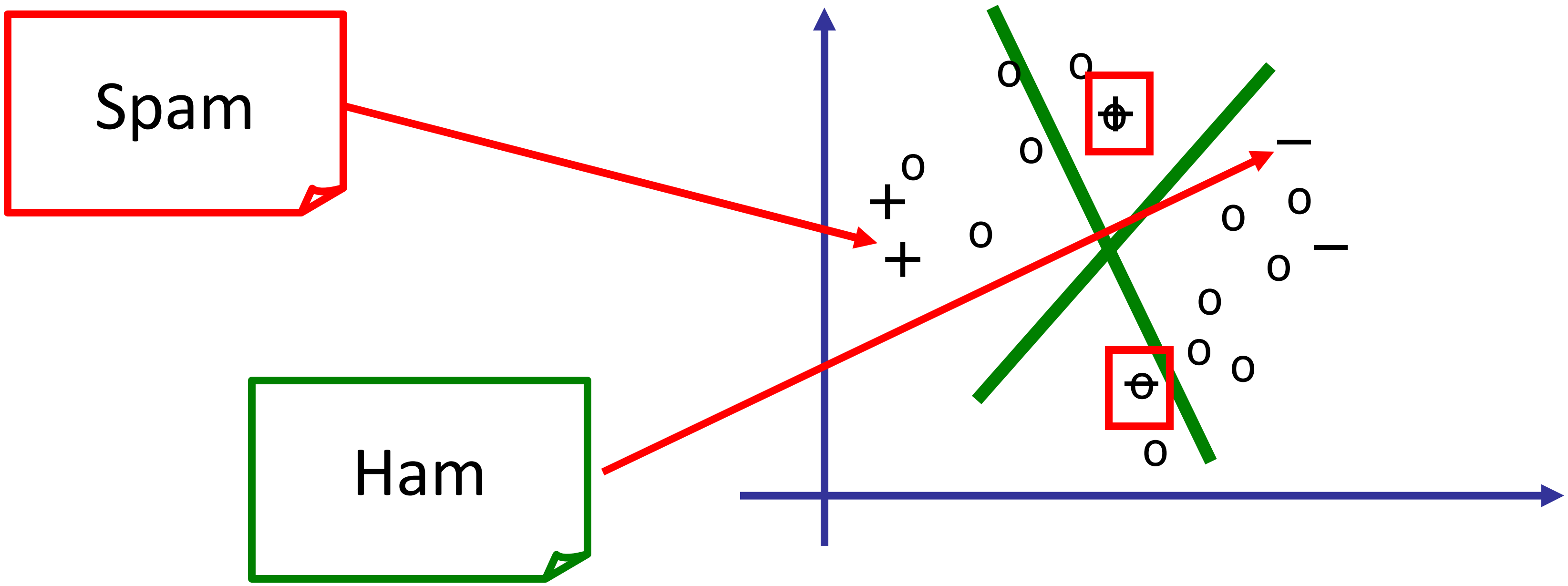
⇩

Learner outputs hypothesis h

⇩

$\mathrm{error}_{true}(h) = E_{x,y}\,|h(x) - y|$

From passive to active learning



Some labels "more informative" than others

Statistical passive/active learning protocol

Data source P_X (produces inputs x_i)

⇩

Active learner assembles

data set $D_n = \{(x_1,y_1),\dots,(x_n,y_n)\}$

by selectively obtaining labels

⇩

Learner outputs hypothesis h

⇩

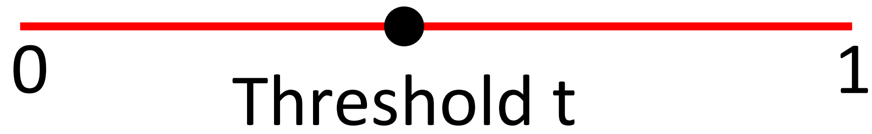
$error_{true}(h) = E_{x\sim P}[h(x) \neq c(x)]$

Passive learning

- Input domain: D=[0,1]
- True concept c:

  $c(x) = +1$ if $x \geq t$

  $c(x) = -1$ if $x < t$



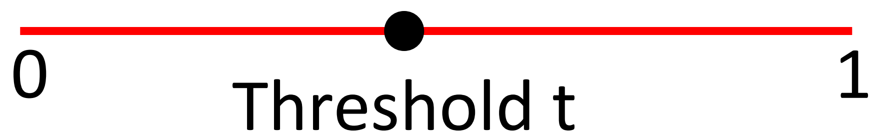
- Passive learning:
  Acquire all labels $y_i \in \{+,-\}$

Active learning

- Input domain: D=[0,1]
- True concept c:

  $c(x) = +1$ if $x \geq t$

  $c(x) = -1$ if $x < t$



- Passive learning:
  Acquire all labels $y_i \in \{+,-\}$
- Active learning:
  Decide which labels to obtain

Comparison

	Labels needed to learn with classification error ε
Passive learning	Ω(1/ε)
Active learning	O(log 1/ε)

Active learning can exponentially reduce the number of required labels!

Key challenges

- PAC Bounds we've seen so far **crucially** depend on i.i.d. data!!
- Actively assembling data set causes **bias**!
  - If we're not careful, active learning can do **worse**!

What you need to know

- Concepts, hypotheses
- PAC bounds (probably approximate correct)
  - For noiseless (“realizable”) case
  - For noisy (“unrealizable”) case
- VC dimension
- Active learning protocol