Active Learning and Optimized Information Gathering

Lecture 8 – Active Learning

CS 101.2
Andreas Krause

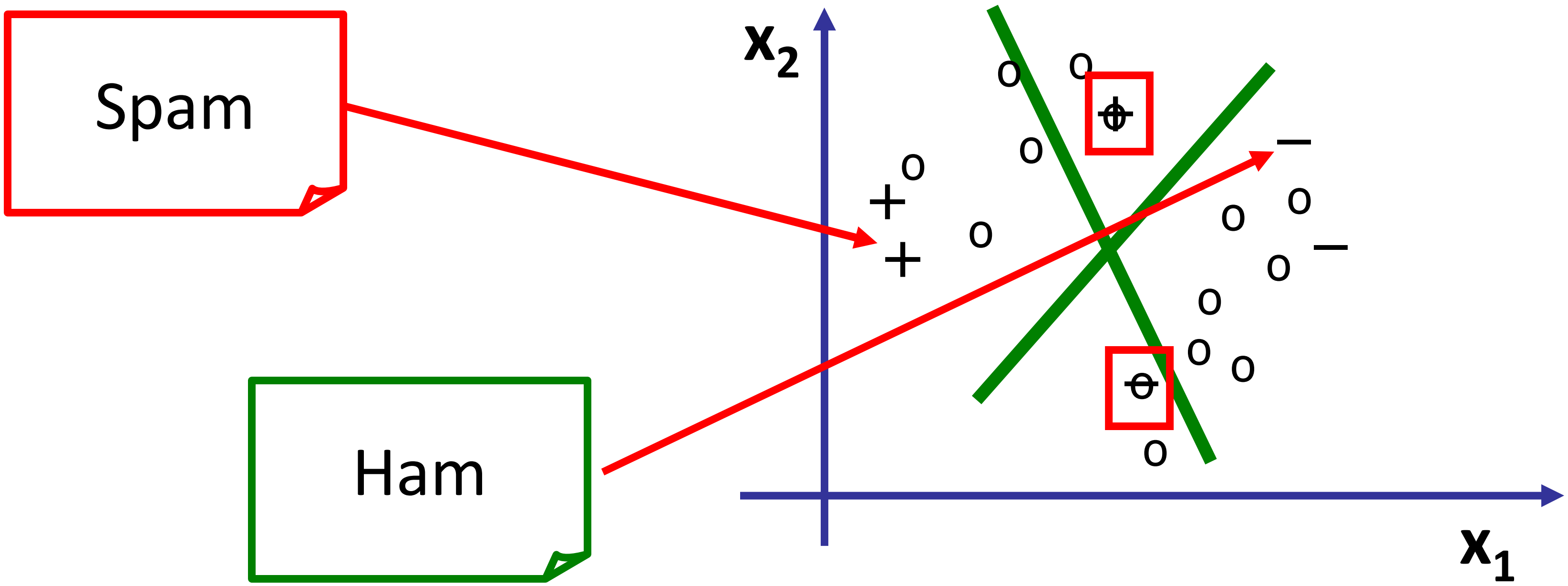
Announcements

- Homework 1: Due today
- Office hours
  - Come to office hours before your presentation!
  - Andreas: **Monday 3pm-4:30pm**, 260 Jorgensen
  - Ryan: Wednesday 4:00-6:00pm, 109 Moore

Outline

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

Spam or Ham?



$$\textbf{label} = \text{sign}(w_0 + w_1\, x_1 + w_2\, x_2)$$

(linear separator)

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

Recap: Concept learning

- Set X of instances, with distribution P_X
- True concept $c: X \rightarrow \{0,1\}$
- Data set $D = \{(x_1,y_1),\ldots,(x_n,y_n)\}$, $x_i \sim P_X$, $y_i = c(x_i)$
- Hypothesis $h: X \rightarrow \{0,1\}$ from $H = \{h_1, \ldots, h_n, \ldots\}$
- Assume $c \in H$ (c also called "target hypothesis")

- $error_{true}(h) = E_X\, |c(x)-h(x)|$
- $error_{train}(h) = (1/n) \sum_i |\underset{y_i}{\cancel{c(x_i)}}-h(x_i)|$

If n large enough, $error_{true}(h) \approx error_{train}(h)$ for all h

Recap: PAC Bounds

How many samples **n** to we need to get error $\leq \varepsilon$ with probability 1-δ ?

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

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

Requires that data is i.i.d.!

Today: Mainly no-noise case (more next week)

Statistical passive/active learning protocol

Data source P_X (produces inputs x_i)

Active learner assembles

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

by selectively obtaining labels

Learner outputs hypothesis h

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

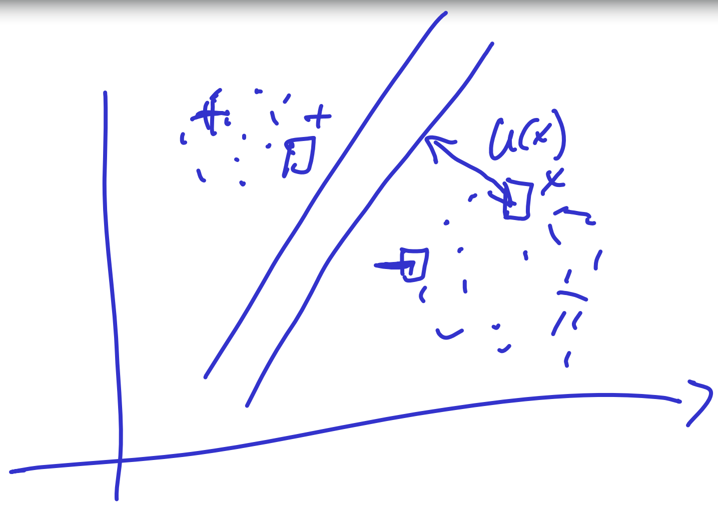
Data set NOT sampled i.i.d.!!

Example: Uncertainty sampling

- Budget of m labels
- Draw n unlabeled examples $m = \frac{n}{2}$
- Repeat until we've picked m labels
  - Assign each unlabeled data an "uncertainty score"
  - Greedily pick the most uncertain example

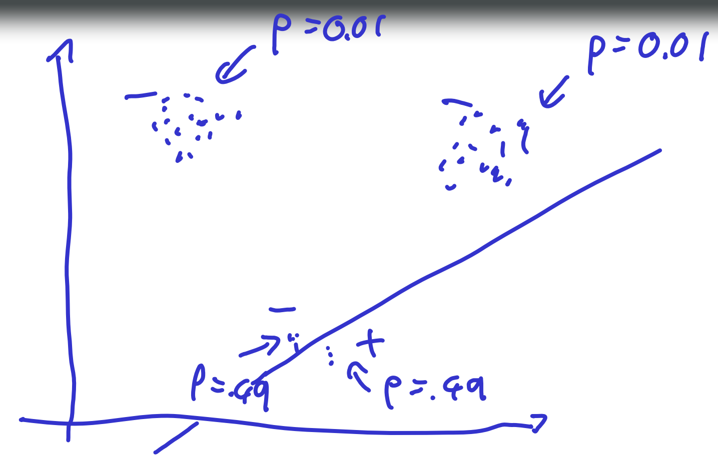
- One of the most commonly used class of heuristics!

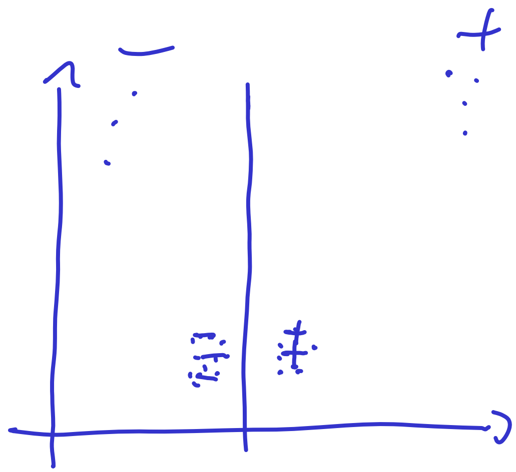
Uncertainty sampling for linear separators



$$U(x) = \frac{1}{\text{distance to separator}}$$

Active learning bias





Uncertainty sampling with $m = n/2$

Even if
$n \to \infty$
$m = \frac{n}{2}$

$\exists$ hypothesis h consistent with labels we've seen

$\text{error}(h) \geq 0.01$

Active learning bias

- If we can pick at most $m = n/2$ labels, with overwhelmingly high probability, US pick points such that there remains a hypothesis with error $> .01$!!!

- With standard passive learning, error $\rightarrow 0$ as $n \rightarrow \infty$

Wish list for active learning

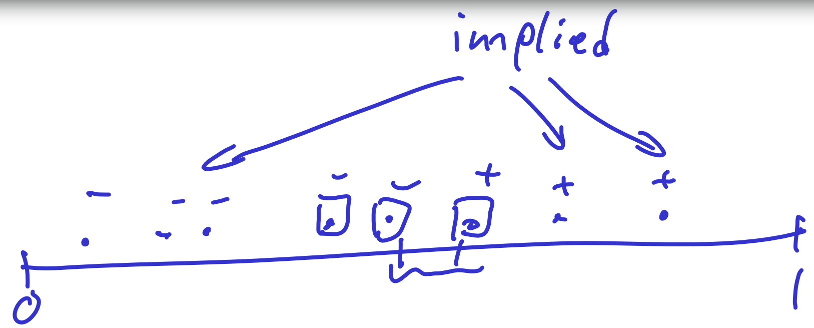
- Minimum requirement
  - **Consistency:** Generalization error should go to 0 asymptotically
- We'd like more than that:
  - **Fallback guarantee:** Convergence rate of error of active learning "at least as good" as passive learning
- What we're **really** after
  - **Rate improvement**: Error of active learning decreases much faster than for passive learning

From passive to active

- Passive PAC learning
  1. Collect data set D of $n \geq 1/\varepsilon$ (log |H| + log $1/\delta$) data points and their labels i.i.d. from P_X
  2. Output consistent hypothesis h
  3. With probability at least $1-\delta$, $error_{true}(h) \leq \varepsilon$

- Key idea
  - Sample **n unlabeled** data points $D_X=\{x_1,\ldots,x_n\}$ i.i.d.
  - Actively query labels until all hypotheses consistent with these labels **agree** on the labels of **all unlabeled** data

Why might this work?



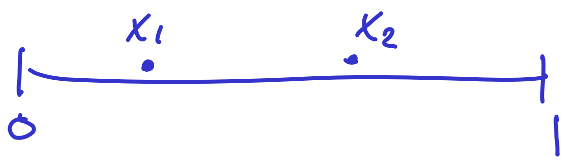
Formalization: “Relevant” hypothesis

- Data set $D = \{(x_1,y_1),\ldots,(x_n,y_n)\}$, Hypothesis space H
- Input data: $D_X = \{x_1,\ldots,x_n\}$
- Relevant hypothesis

  $H'(D_X) = H' =$ Restriction of H on D_X
- Formally:

  $H' = \{h': D_X \rightarrow \{0,1\} \; \exists \, h \in H \text{ s.t. } \forall \, x \in D_X: h'(x)=h(x)\}$

Example: Threshold functions



$$H = \{h(x) = [x \geq t] \text{ for some } t \in [0,1]\}$$

$$h(x) = \begin{cases} 1 & \text{if } x \geq t \\ 0 & \text{otw.} \end{cases}$$

$$H' = \{h_1, h_2, h_3\}$$

	x_1	x_2
h_1	−	−
h_2	−	+
h_3	+	+

Version space

- Input data $D_X = \{x_1,\ldots,x_n\}$
- Partially labeled: Have $L = \{(x_{i_1},y_{i_1}),\ldots,(x_{i_m},y_{i_m})\}$
- The (relevant) **version space** is the set of all **relevant** hypotheses **consistent** with the labels L
- Formally:

$$V(H', L) = V(L) = V = \{h \in H' : \forall (x,y) \in L : h(x) = y\}$$

- Why useful?
  - Partial labels L imply all remaining labels for $D_X \Leftrightarrow |V|=1$

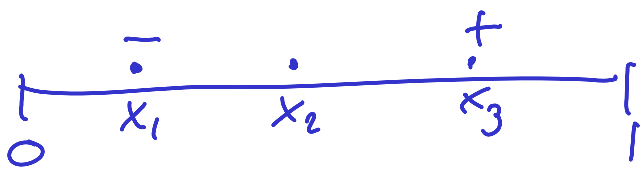
Version space

- Input data $D_X = \{x_1,\ldots,x_n\}$
- Partially labeled: Have $L = \{(x_{i_1},y_{i_1}),\ldots,(x_{i_m},y_{i_m})\}$
- The (relevant) **version space** is the set of all **relevant** hypotheses **consistent** with the labels L
- Formally:

$$V(D_X,L) = V = \{h' \in H'(D_X): h'(x_{i_j})=y_{i_j} \text{ for } 1 \leq j \leq m\}$$

- Why useful?
  - Partial labels L imply all remaining labels for $D_X \Leftrightarrow |V|=1$

Example: Binary thresholds



$V = \{h_1, h_2\}$

	x_1	x_2	x_3
h_1	−	−	+
h_2	−	+	+

Pool-based active learning with fallback

1. Collect $n \geq 1/\varepsilon \ (\log |H| + \log 1/\delta)$ **unlabeled** data points D_X from P_X
2. Actively request labels L until there remains a single hypothesis $h' \in H'$ that's consistent with these labels (i.e., $|V(H',L)| = 1$)
3. Output any hypothesis $h \in H$ consistent with the obtained labels. With probability $\geq 1-\delta$ $\text{error}_{\text{true}}(h) \leq \varepsilon$

Get PAC guarantees for active learning
Bounds on #labels for fixed error ε
carry over from passive to active
→ **Fallback guarantee**

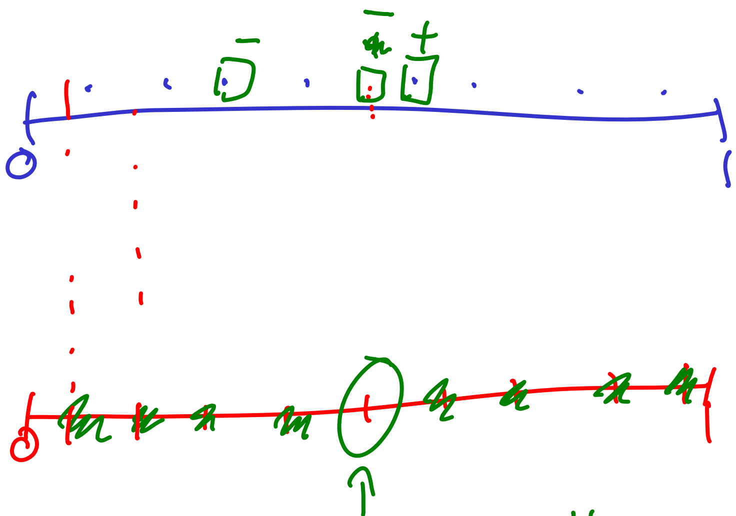
Wish list for active learning

- Minimum requirement
  - **Consistency:** Generalization error should go to 0 asymptotically
- We'd like more than that:
  - **Fallback guarantee:** Convergence rate of error of active learning "at least as good" as passive learning
- What we're **really** after
  - **Rate improvement**: Error of active learning decreases much faster than for passive learning

Pool-based active learning with fallback

1. Collect $n \geq 1/\varepsilon$ ($\log |H| + \log 1/\delta$) **unlabeled** data points D_X from P_X
2. Actively request labels L until there remains a single hypothesis $h' \in H'$ that's consistent with these labels (i.e., $|V(H',L)| = 1$)
3. Output any hypothesis $h \in H$ consistent with the obtained labels. With probability $\geq 1-\delta$ $\text{error}_{true}(h) \leq \varepsilon$

Example: Threshold functions



must be the only consistent relevant hyp

$\Rightarrow$ only need $O(\log n)$ labels

(instead of n for passive)

Generalizing binary search [Dasgupta '04]

- Want to shrink the version space (number of consistent hypotheses) as quickly as possible.
- Have labels L
- General (greedy) approach:
  - For each unlabeled instance x_i compute
    $v_{i,1}$ = $V(H', L \cup \{(x_i, 1)\})$ ← "hallucinate" label 1
    $v_{i,0}$ = $V(H', L \cup \{(x_i, 0)\})$ ← — " — 0
  - $v_i = \min \{v_{i,1}, v_{i,0}\}$
  - Obtain label y_i for x_i where $i = \text{argmax}_j \{v_j\}$

Ideal case

$L_0 = \{\} \subset L_1 \subset L_2 \subset \dots \subset L_m \qquad |L_m| = m$

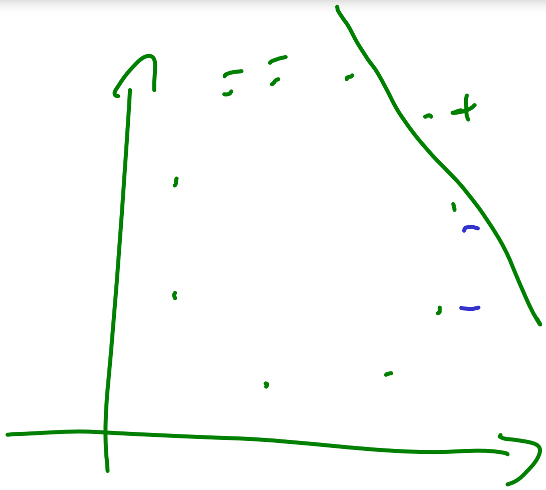
Ideal case: $\forall i \ \exists x : |V(L_i \cup \{(x_i, 1)\})| = |V(L_i \cup \{(x_i, 0)\})|$

$\Rightarrow |V(L_{i+1})| \leq \frac{1}{2} |V(L_i)|$

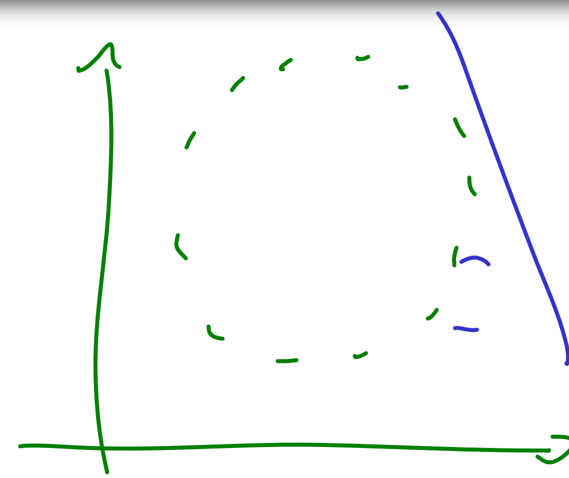
$\Rightarrow |V(L_m)| \leq \frac{1}{2}^m |H'| \overset{!}{\leq} 1$

$\Rightarrow m = O(\log n)$

Is it always possible to half the version space?

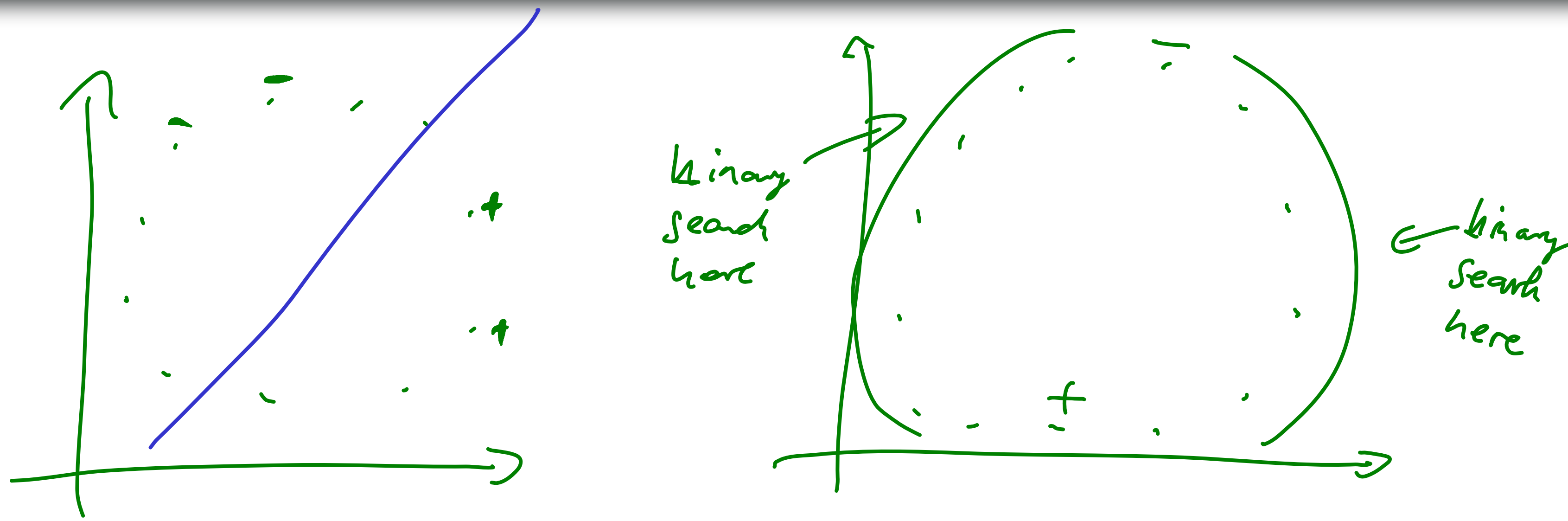


need to distinguish from



need to see all n data points

Typical case much more benign



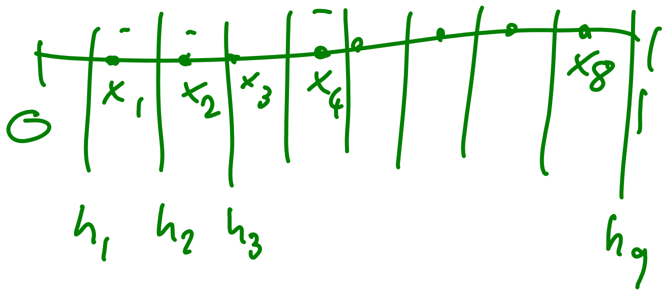
Typically: $O(\log n)$ samples suffice

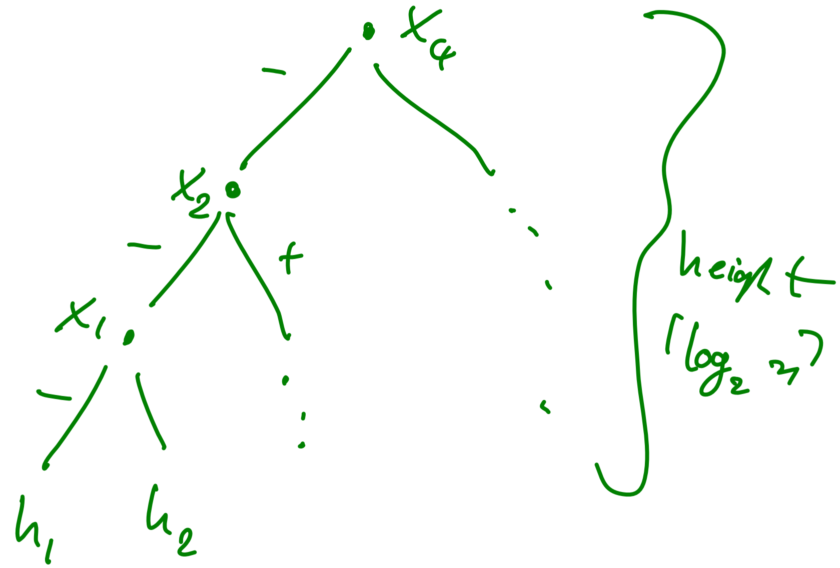
Query trees

- A **query tree** is a rooted, labeled tree on the relevant hypothesis H'
- Each node is labeled with an input $x \in D_X$
- Each edge is labeled with {0,1}
- Each path from root to hypothesis $h' \in H'$ is a labeling L such that $V(D_X,L) = \{h'\}$

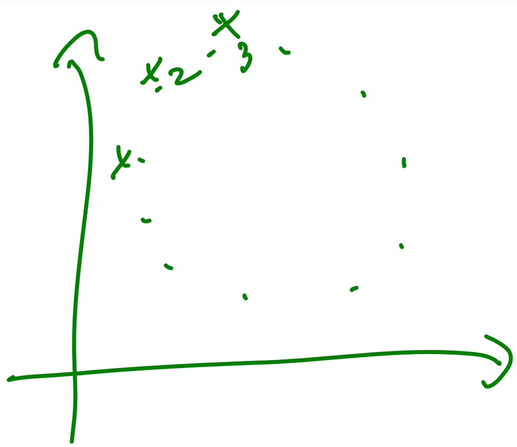
- **Want query trees of minimum height**

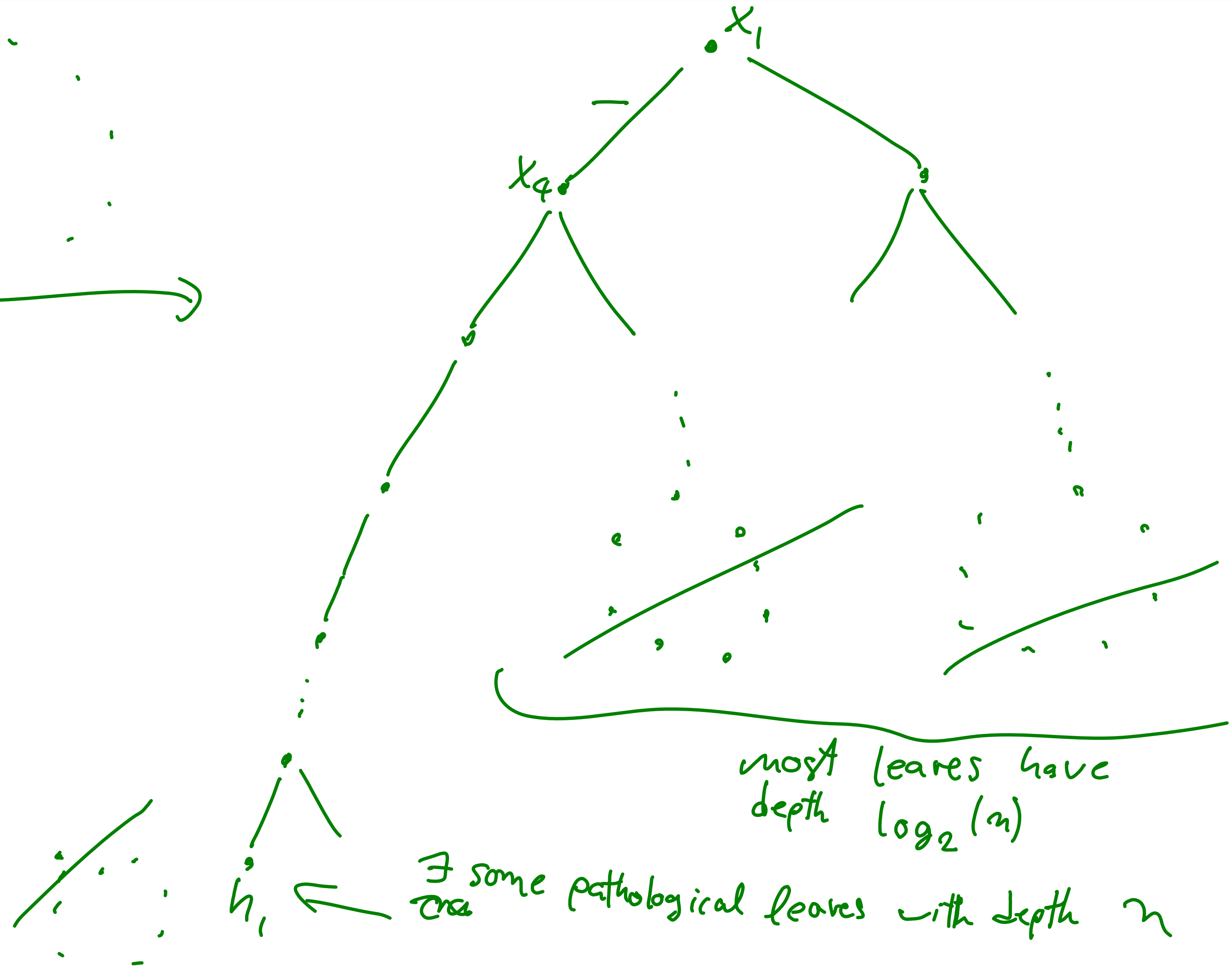
Example: Threshold functions





Example: linear separators (2D)





Number of labels needed to identify hypothesis

- Depends on target hypothesis!
- Binary thresholds (on n inputs D_X)
  - Optimal query tree needs O(log n) labels! ☺
- For linear separators in 2D (on n inputs D_X)
  - For some hypotheses, even optimal tree needs n labels ☹
  - On average, optimal query tree needs O(log n) labels! ☺

➔ Average-case analysis of active learning

Average case query tree learning

- Query tree T
- $Cost(T) = 1/|H'| \sum_{h' \in H'} depth(h',T)$

$$\boxed{\text{Want } T^* = \arg\min_T Cost(T)}$$

- Superexponential number of query trees ☹
- Finding the optimal one is hard ☹

Greedy construction of query trees [Dasgupta '04]

<u>Algorithm GreedyTree(D_X, L)</u>

$V' = H'(D_X$~~)~~ ,L)

If $V'=\{h\}$

    return Leaf(h)

Else

    For each unlabeled instance x_i compute

    $v_{i,1} = |V'(H',L \cup \{(x_i,1)\}|$ and $v_{i,0} = |V'(H',L \cup \{(x_i,0)\}|$

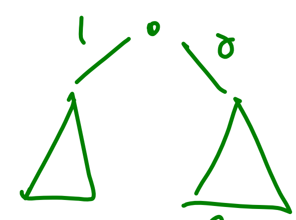
    $v_i = \min \{v_{i,1}, v_{i,0}\}$ max. "disagreement"

    Let $i = \mathrm{argmax}_j \{v_j\}$

    LeftSubTree = GreedyTree(D_X, $L \cup \{(x_i,1)\}$)

    RightSubTree = GreedyTree(D_X, $L \cup \{(x_i,0)\}$)

    return Node x_i with children LeftSubTree (1) and RightSubTree(0)



Near-optimality of greedy tree [Dasgupta '04]

Theorem: Let $T^* = \mathrm{argmin}_T \ \mathrm{Cost}(T)$

Then GreedyTree constructs a query tree T such that

$$\mathrm{Cost}(T) = O(\log |H'|)\ \mathrm{Cost}(T^*)$$

Limitations of this algorithm

- Often computationally intractable
  - Finding “most-disagreeing” hypothesis is difficult
- No-noise assumption
- Will see how we can relax these assumptions in the talks next week.

Bayesian or not Bayesian?

- Greedy querying needs at most O(log |H'|) queries more than optimal query tree **on average**
- Assumes prior distribution (uniform) on hypotheses
- If our assumption is wrong, generalization bound **still holds**! (but might need more labels)

Can also do a pure Bayesian analysis:

- Query by Committee algorithm [Freund et al '97]
- Assumes that Nature draws hypotheses from **known** prior distribution

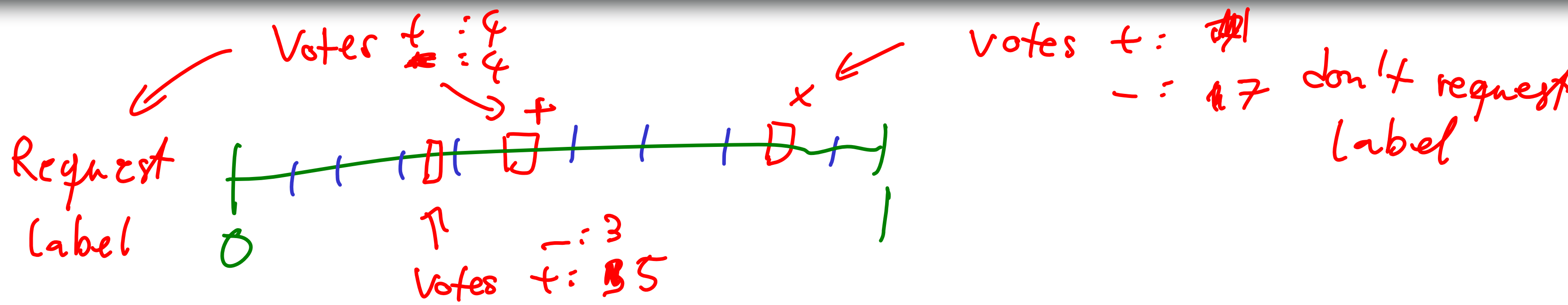
Query by Committee

- Assume prior distribution on hypotheses
- Sample a "committee" of 2k hypotheses drawn from the prior distribution
- Search for an input such that k "members" assign label 1, and k "members" assign 0, and query that label ("maximal disagreement")

Theorem [Freund et al '97]

For linear separators in R^d where both the coefficients w and the data X are drawn uniformly from the unit sphere, QBC requires exponentially fewer labels than passive learning to achieve same error

Example: Threshold functions



Wish list for active learning

- Minimum requirement
  - **Consistency:** Generalization error should go to 0 asymptotically
- We'd like more than that:
  - **Fallback guarantee:** Convergence rate of error of active learning "at least as good" as passive learning

✓ using pool-based active learning

What we're **really** after

- **Rate improvement**: Error of active learning decreases much faster than for passive learning

generalized binary search
is "competitive" with optimal querying
on average if hypotheses equally likely

Beyond pool-based analysis

- Pool-based active learning just one convenient analysis technique
  - Gets around active learning bias by generalizing from a pool drawn i.i.d. at random
  - In pool-based analysis, there are examples where active learning does not outperform passive learning
- Exciting recent theoretical results show that using a more involved analysis, active learning **always** helps (asymptotically) [Balcan, Hanneke, Wortman COLT ’08]
- Also other active learning paradigms
  - E.g.: Active querying (constructing rather than selecting inputs)

What you need to know

- Uncertainty sampling
- Active learning bias
- Pool-based active learning scheme
- Relevant hypotheses and version spaces
- Generalized binary search algorithm