10-601 Introduction to Machine Learning

Machine Learning Department
School of Computer Science
Carnegie Mellon University

PAC Learning

Matt Gormley
Lecture 24
Apr. 26, 2021

Reminders

- Homework 7: Graphical Models
  - Out: Mon, Apr. 19
  - Due: Fri, Apr. 30 at 11:59pm
- Homework 8: Learning Paradigms
  - Out: Fri, Apr. 30
  - Due: Fri, May. 7 at 11:59pm

LEARNING THEORY

PAC-MAN Learning

For some hypothesis $h \in \mathcal{H}$:

1. True Error

$$R(h)$$

2. Training Error

$$\hat{R}(h)$$

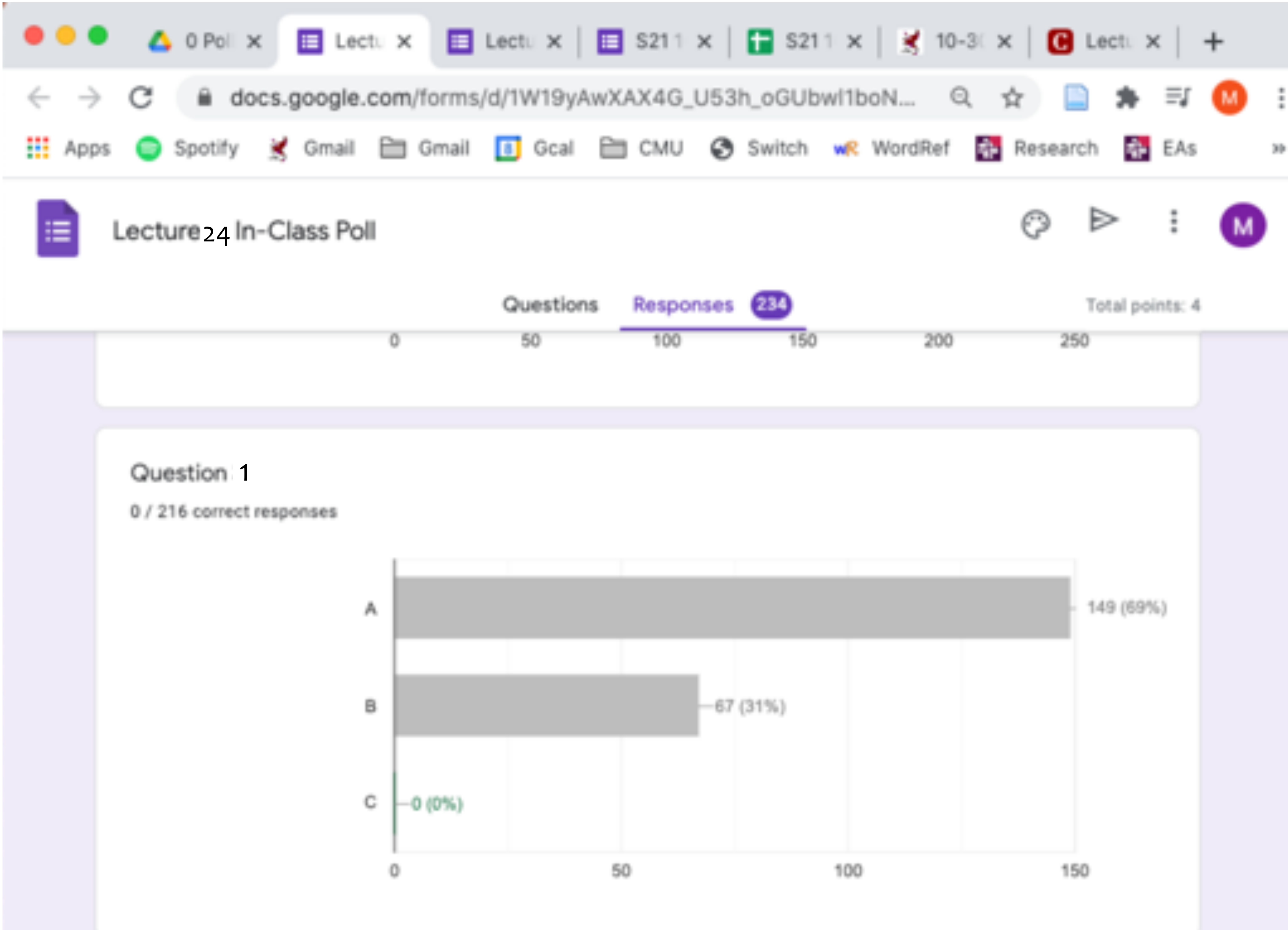
Question: (version B)

What is the expected number of PAC-MAN levels Matt will complete before a **Game-Over**?

A. 1-10

B. 11-20

C. 21-30


docs.google.com/forms/d/1W19yAwXAX4G_U53h_oGUbwl1boN...
Apps
Spotify
Gmail
Gmail
Gcal
CMU
Switch
WordRef
Research
EAs
Lecture 24 In-Class Poll
Questions
Responses 234
Total points: 4
0
50
100
150
200
250
Question 1
0 / 216 correct responses
A
149 (69%)
B
67 (31%)
C
0 (0%)
0
50
100
150

PAC-MAN Learning

For some hypothesis $h \in \mathcal{H}$:

1. True Error

$$R(h)$$

2. Training Error

$$\hat{R}(h)$$

Questions For Today

1. Given a classifier with **zero training error**, what can we say about **true error** (aka. generalization error)?
   (Sample Complexity, Realizable Case)

2. Given a classifier with **low training error**, what can we say about **true error** (aka. generalization error)?
   (Sample Complexity, Agnostic Case)

3. Is there a **theoretical justification for regularization** to avoid overfitting?
   (Structural Risk Minimization)

Medical Diagnosis Dataset

Doctor diagnoses the patient as sick or not $y \in \{+, -\}$ based on attributes of the patient $x_1, x_1, \ldots, x_M$

	y	x_1	x_2	x_3	x_4
i	**allergic?**	**hives?**	**sneezing?**	**red eye?**	**cat?**
1	-	Y	N	N	N

Medical Diagnosis Dataset

Doctor diagnoses the patient as sick or not $y \in \{+, -\}$ based on attributes of the patient $x_1, x_1, \ldots, x_M$

	y	x_1	x_2	x_3	x_4
i	**allergic?**	**hives?**	**sneezing?**	**red eye?**	**cat?**
1	-	Y	N	N	N
2	-	N	Y	N	N
3	+	Y	Y	N	N
4	-	Y	N	Y	Y
5	+	N	Y	Y	N

Medical Diagnosis Dataset

Doctor diagnoses the patient as sick or not $y \in \{+, -\}$ based on attributes of the patient $x_1, x_1, \ldots, x_M$

	y	x_1	x_2	x_3	x_4
i	**allergic?**	**hives?**	**sneezing?**	**red eye?**	**cat?**
1	$y^{(1)}$ -	$x_1^{(1)}$ Y	$x_2^{(1)}$ N	$x_3^{(1)}$ N	$x_4^{(1)}$ N
2	$y^{(2)}$ -	$x_1^{(2)}$ N	$x_2^{(2)}$ Y	$x_3^{(2)}$ N	$x_4^{(2)}$ N
3	$y^{(3)}$ +	$x_1^{(3)}$ Y	$x_2^{(3)}$ Y	$x_3^{(3)}$ N	$x_4^{(3)}$ N
4	$y^{(4)}$ -	$x_1^{(3)}$ Y	$x_2^{(3)}$ N	$x_3^{(3)}$ Y	$x_4^{(3)}$ Y
5	$y^{(5)}$ +	$x_1^{(4)}$ N	$x_2^{(4)}$ Y	$x_3^{(4)}$ Y	$x_4^{(4)}$ N

Medical Diagnosis Dataset

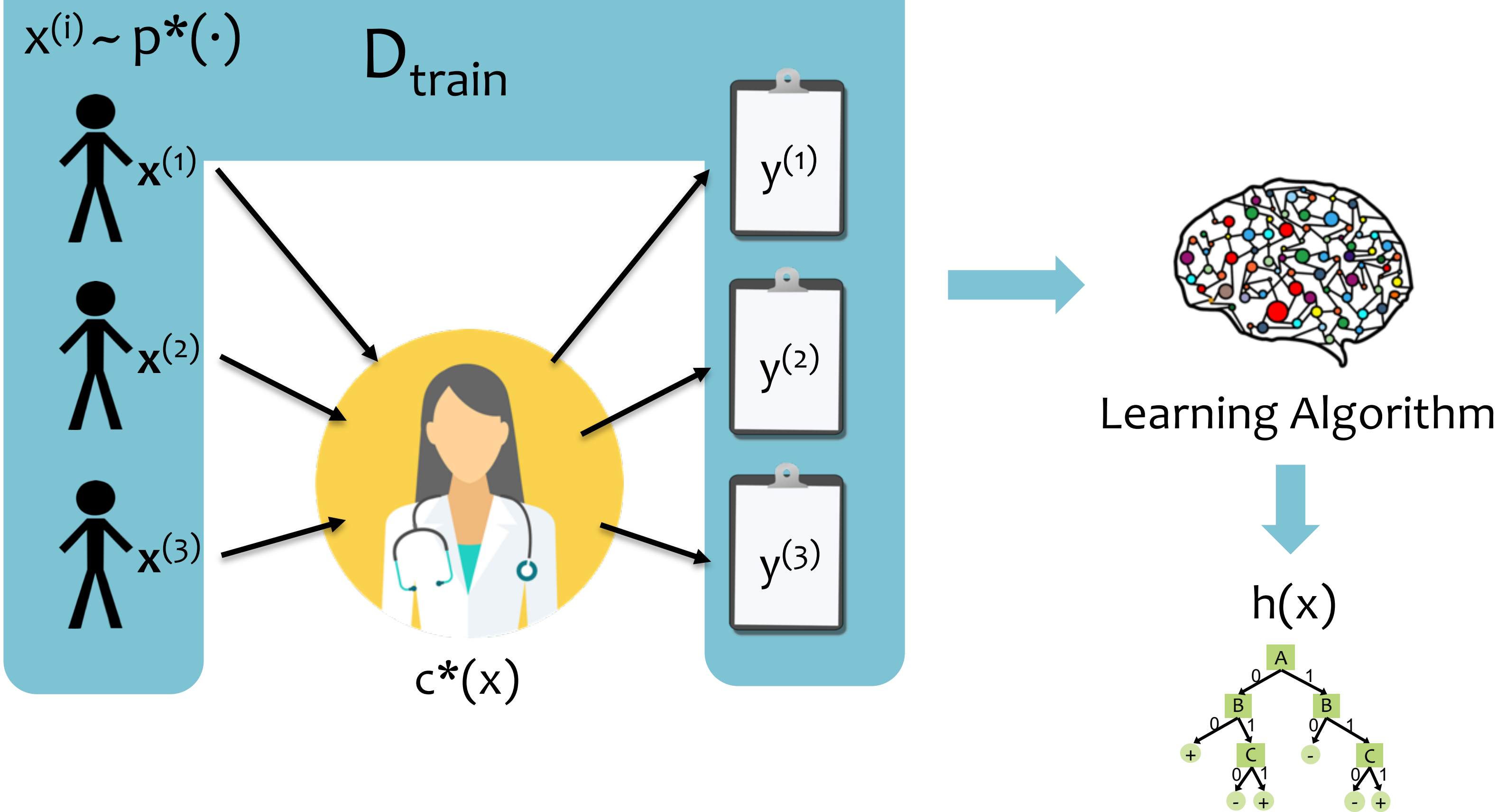
Doctor diagnoses the patient as sick or not $y \in \{+, -\}$ based on attributes of the patient $x_1, x_1, \ldots, x_M$

	y	x_1	x_2	x_3	x_4	
i	**allergic?**	**hives?**	**sneezing?**	**red eye?**	**cat?**	
1	$y^{(1)}$ -	$x_1^{(1)}$ Y	$x_2^{(1)}$ N	$x_3^{(1)}$ N	$x_4^{(1)}$ N	$\mathbf{x}^{(1)}$
2	$y^{(2)}$ -	$x_1^{(2)}$ N	$x_2^{(2)}$ Y	$x_3^{(2)}$ N	$x_4^{(2)}$ N	$\mathbf{x}^{(2)}$
3	$y^{(3)}$ +	$x_1^{(3)}$ Y	$x_2^{(3)}$ Y	$x_3^{(3)}$ N	$x_4^{(3)}$ N	$\mathbf{x}^{(3)}$
4	$y^{(4)}$ -	$x_1^{(3)}$ Y	$x_2^{(3)}$ N	$x_3^{(3)}$ Y	$x_4^{(3)}$ Y	$\mathbf{x}^{(4)}$
5	$y^{(5)}$ +	$x_1^{(4)}$ N	$x_2^{(4)}$ Y	$x_3^{(4)}$ Y	$x_4^{(4)}$ N	$\mathbf{x}^{(5)}$

N = 5 training examples

M = 4 attributes

PAC/SLT Model for Supervised ML



PAC/SLT Model for Supervised ML

- **Problem Setting**
  - Set of possible inputs, $\mathbf{x} \in \mathcal{X}$ (all possible patients)
  - Set of possible outputs, $y \in \mathcal{Y}$ (all possible diagnoses)
  - Distribution over instances, $p^*(\cdot)$
  - Exists an unknown target function, $c^* : \mathcal{X} \rightarrow \mathcal{Y}$ (the doctor's brain)
  - Set, $\mathcal{H}$, of candidate hypothesis functions, $h : \mathcal{X} \rightarrow \mathcal{Y}$ (all possible decision trees)
- **Learner is given** N training examples $D = \{(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \ldots, (\mathbf{x}^{(N)}, y^{(N)})\}$ where $x^{(i)} \sim p^*(\cdot)$ and $y^{(i)} = c^*(\mathbf{x}^{(i)})$ (history of patients and their diagnoses)
- **Learner produces** a hypothesis function, $\hat{y} = h(x)$, that best approximates unknown target function $y = c^*(x)$ on the training data

PAC/SLT Model for Supervised ML

- **Problem Setting**
  - Set of possible inputs, $\mathbf{x} \in \mathcal{X}$ (all possible patients)
  - Set of possible outputs, $y \in \mathcal{Y}$ (all possible diagnoses)
  - Distribution [illegible] er instances, $p^*(\cdot)$
  - Exists an unkno [illegible] n target function, $c^* : \mathcal{X} \to \mathcal{Y}$
    (the doctor's brain
  - Set, $\mathcal{H}$, of candida
    (all possible decisi
- **Learner is given** N t
  D = $\{(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)},$
  where $x^{(i)} \sim p^*(\cdot)$ an
  (history of patients
- **Learner produces** a
  best approximates
  the training data

Two important settings we'll consider:

1. **Classification:** the possible outputs are **discrete**
2. **Regression:** the possible outputs are **real-valued**

Medical Diagnosis Dataset

Doctor diagnoses the patient as sick or not $y \in \{+, -\}$ based on attributes of the patient $x_1, x_1, \ldots, x_M$

	y	x_1	x_2	x_3	x_4	
i	**allergic?**	**hives?**	**sneezing?**	**red eye?**	**cat?**	
1	$y^{(1)}$ -	$x_1^{(1)}$ Y	$x_2^{(1)}$ N	$x_3^{(1)}$ N	$x_4^{(1)}$ N	$\mathbf{x}^{(1)}$
2	$y^{(2)}$ -	$x_1^{(2)}$ N	$x_2^{(2)}$ Y	$x_3^{(2)}$ N	$x_4^{(2)}$ N	$\mathbf{x}^{(2)}$
3	$y^{(3)}$ +	$x_1^{(3)}$ Y	$x_2^{(3)}$ Y	$x_3^{(3)}$ N	$x_4^{(3)}$ N	$\mathbf{x}^{(3)}$
4	$y^{(4)}$ -	$x_1^{(3)}$ Y	$x_2^{(3)}$ N	$x_3^{(3)}$ Y	$x_4^{(3)}$ Y	$\mathbf{x}^{(4)}$
5	$y^{(5)}$ +	$x_1^{(4)}$ N	$x_2^{(4)}$ Y	$x_3^{(4)}$ Y	$x_4^{(4)}$ N	$\mathbf{x}^{(5)}$

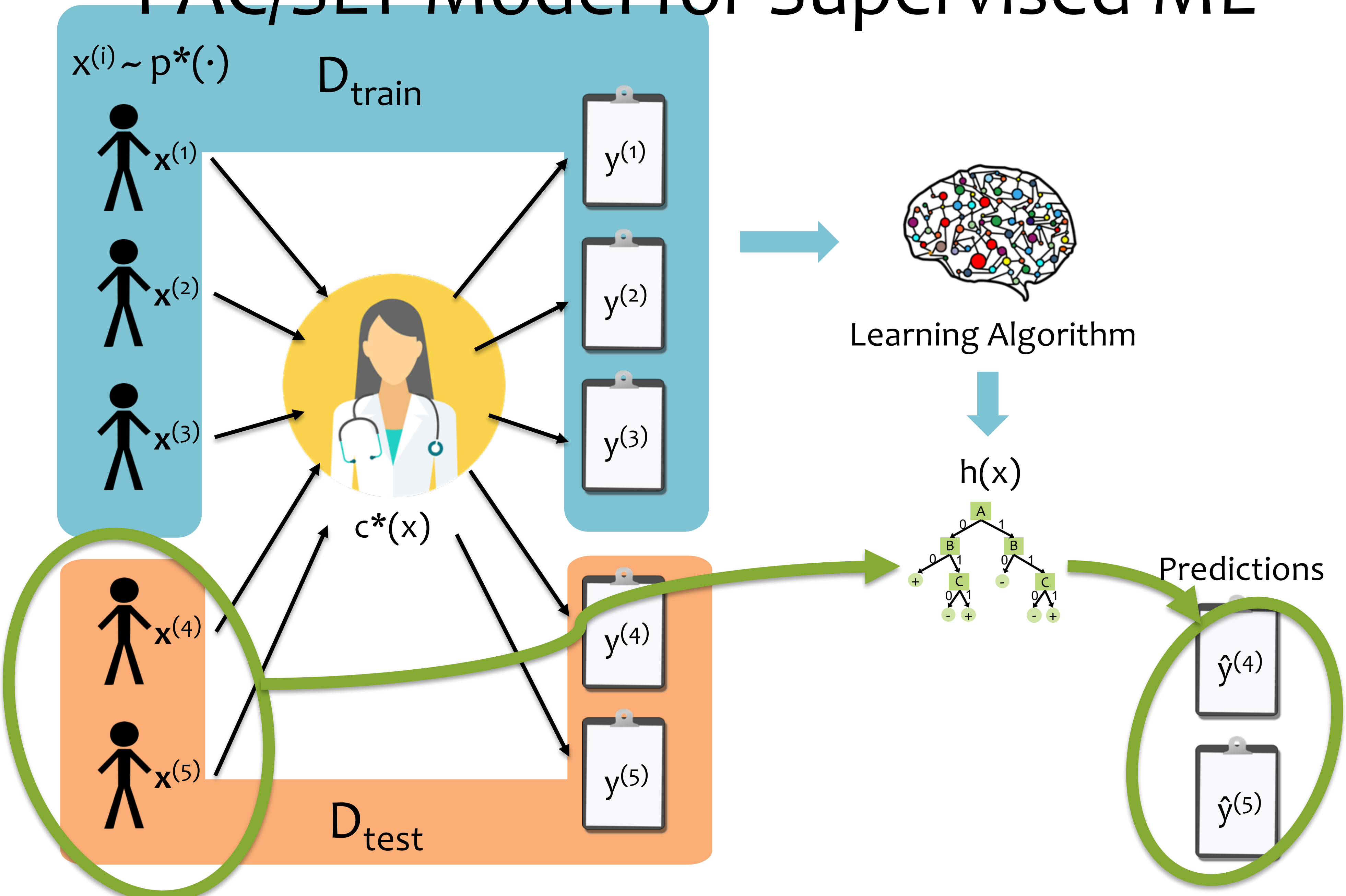
N = 5 training examples

M = 4 attributes

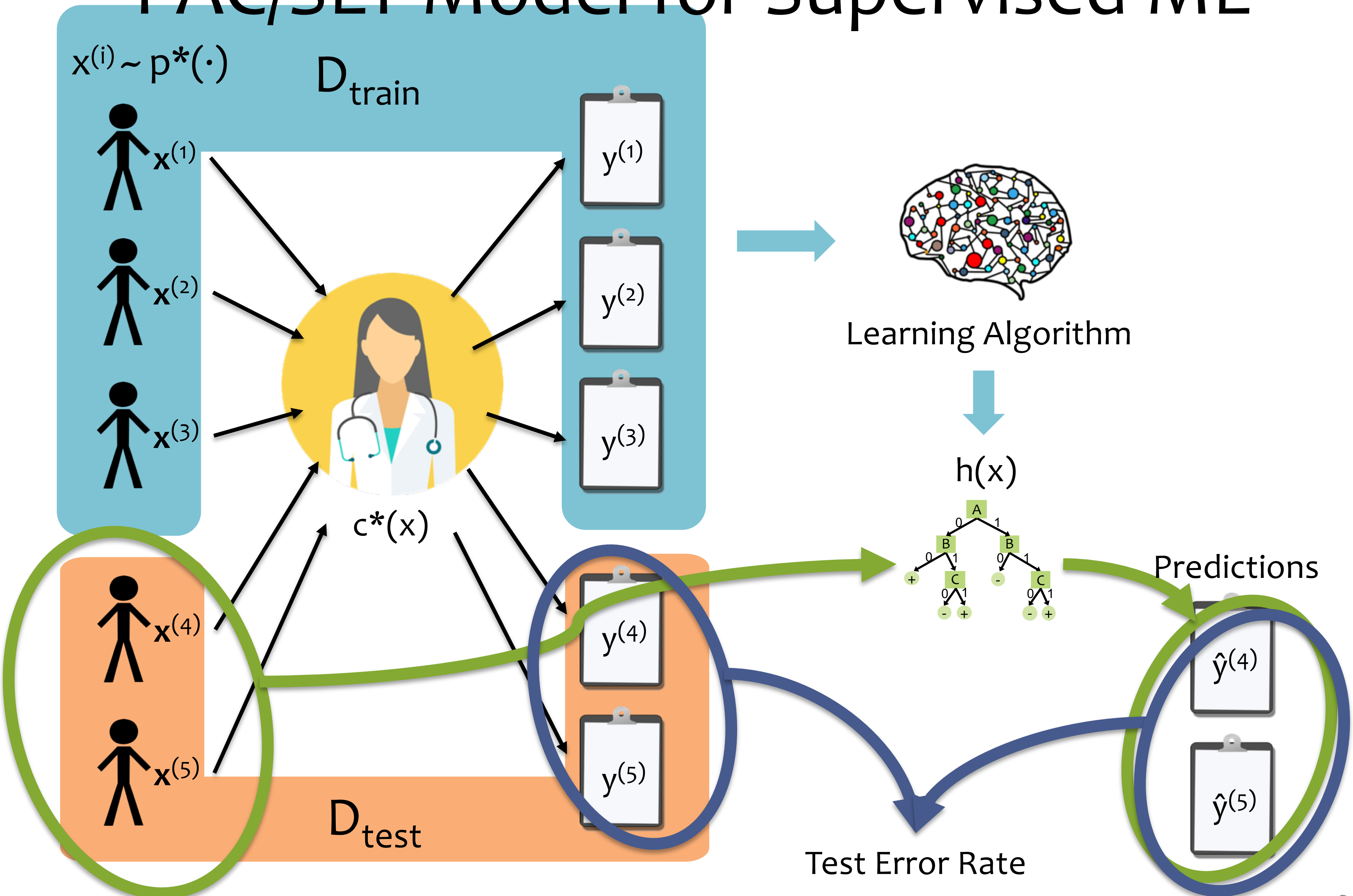
Example hypothesis function:

$$h(\mathbf{x}) = \begin{cases} + & \text{if sneezing = Y} \\ - & \text{otherwise} \end{cases}$$

PAC/SLT Model for Supervised ML



PAC/SLT Model for Supervised ML



Two Types of Error

1. True Error (aka. **expected risk**)

$$R(h) = P_{\mathbf{x} \sim p^*(\mathbf{x})}(c^*(\mathbf{x}) \neq h(\mathbf{x}))$$

This quantity is always **unknown**

2. Train Error (aka. **empirical risk**)

$$\begin{aligned} \hat{R}(h) &= P_{\mathbf{x} \sim \mathcal{S}}(c^*(\mathbf{x}) \neq h(\mathbf{x})) \\ &= \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}(c^*(\mathbf{x}^{(i)}) \neq h(\mathbf{x}^{(i)})) \\ &= \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}(y^{(i)} \neq h(\mathbf{x}^{(i)})) \end{aligned}$$

We can **measure** this on the training data

where $\mathcal{S} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)})\}_{i=1}^{N}$ is the training data set, and $\mathbf{x} \sim \mathcal{S}$ denotes that $\mathbf{x}$ is sampled from the empirical distribution.

PAC / SLT Model

We've also referred to this as the "Function Approximation View"

1. Generate instances from *unknown* distribution p^*

$$\mathbf{x}^{(i)} \sim p^*(\mathbf{x}), \forall i \quad (1)$$

2. Oracle labels each instance with *unknown* function c^*

$$y^{(i)} = c^*(\mathbf{x}^{(i)}), \forall i \quad (2)$$

3. Learning algorithm chooses hypothesis $h \in \mathcal{H}$ with low(est) training error, $\hat{R}(h)$

$$\hat{h} = \underset{h}{\operatorname{argmin}}\, \hat{R}(h) \quad (3)$$

4. Goal: Choose an h with low generalization error $R(h)$

Three Hypotheses of Interest

The **true function** c^* is the one we are trying to learn and that labeled the training data:

$$y^{(i)} = c^*(\mathbf{x}^{(i)}), \forall i \quad (1)$$

The **expected risk minimizer** has lowest true error:

$$h^* = \operatorname*{argmin}_{h \in \mathcal{H}} R(h)$$

Question:
True or False:
h* and c* are always equal.

The **empirical risk minimizer** has lowest training error:

$$\hat{h} = \operatorname*{argmin}_{h \in \mathcal{H}} \hat{R}(h) \quad (3)$$

Three Hypotheses of Interest

Whiteboard:

– Discussion of Poll Question

PAC LEARNING

Probably Approximately Correct (PAC) Learning

Whiteboard:

- PAC Criterion
- Meaning of "Probably Approximately Correct"
- Def: PAC Learner
- Sample Complexity
- Consistent Learner

PAC Learning

The **PAC criterion** is that our learner produces a high accuracy learner with high probability:

$$P(|R(h) - \hat{R}(h)| \leq \epsilon) \geq 1 - \delta \tag{1}$$

Suppose we have a learner that produces a hypothesis $h \in \mathcal{H}$ given a sample of N training examples. The algorithm is called **consistent** if for every ϵ and δ, there exists a positive number of training examples N such that for any distribution p^*, we have that:

$$P(|R(h) - \hat{R}(h)| > \epsilon) < \delta \tag{2}$$

The **sample complexity** is the minimum value of N for which this statement holds. If N is finite for some learning algorithm, then $\mathcal{H}$ is said to be **learnable.** If N is a polynomial function of $\frac{1}{\epsilon}$ and $\frac{1}{\delta}$ for some learning algorithm, then $\mathcal{H}$ is said to be **PAC learnable.**

SAMPLE COMPLEXITY RESULTS

Sample Complexity Results

Definition 0.1. The **sample complexity** of a learning algorithm is the number of examples required to achieve arbitrarily small error (with respect to the optimal hypothesis) with high probability (i.e. close to 1).

Four Cases we care about…

We'll start with the finite case…

	Realizable	Agnostic
Finite $\|\mathcal{H}\|$		
Infinite $\|\mathcal{H}\|$		

Generalization and Overfitting

Whiteboard:

- Realizable vs. Agnostic Cases
- Finite vs. Infinite Hypothesis Spaces
- Theorem 1: Realizable Case, Finite |H|
- Proof of Theorem 1

Sample Complexity Results

Definition 0.1. The **sample complexity** of a learning algorithm is the number of examples required to achieve arbitrarily small error (with respect to the optimal hypothesis) with high probability (i.e. close to 1).

Four Cases we care about…

	Realizable	Agnostic
Finite $\lvert\mathcal{H}\rvert$	**Thm. 1** $N \geq \frac{1}{\epsilon}\left[\log(\lvert\mathcal{H}\rvert) + \log(\frac{1}{\delta})\right]$ labeled examples are sufficient so that with probability $(1-\delta)$ all $h \in \mathcal{H}$ with $\hat{R}(h) = 0$ have $R(h) \leq \epsilon$.	
Infinite $\lvert\mathcal{H}\rvert$		

Example: Conjunctions

Question:

Suppose H = class of conjunctions over **x** in $\{0,1\}^M$

Example hypotheses:

$h(\mathbf{x}) = x_1 (1-x_3) x_5$

$h(\mathbf{x}) = x_1 (1-x_2) x_4 (1-x_5)$

If M = 10, ε = 0.1, δ = 0.01, how many examples suffice according to Theorem 1?

Answer:

A. 10*(2*ln(10)+ln(100)) ≈ 92
B. 10*(3*ln(10)+ln(100)) ≈ 116
C. 10*(10*ln(2)+ln(100)) ≈ 116
D. 10*(10*ln(3)+ln(100)) ≈ 156
E. 100*(2*ln(10)+ln(10)) ≈ 691
F. 100*(3*ln(10)+ln(10)) ≈ 922
G. 100*(10*ln(2)+ln(10)) ≈ 924
H. 100*(10*ln(3)+ln(10)) ≈ 1329

Thm. 1 $N \geq \frac{1}{\epsilon}\left[\log(|\mathcal{H}|) + \log(\frac{1}{\delta})\right]$ labeled examples are sufficient so that with probability $(1-\delta)$ all $h \in \mathcal{H}$ with $\hat{R}(h) = 0$ have $R(h) \leq \epsilon$.

Sample Complexity Results

Definition 0.1. The **sample complexity** of a learning algorithm is the number of examples required to achieve arbitrarily small error (with respect to the optimal hypothesis) with high probability (i.e. close to 1).

Four Cases we care about…

	Realizable	Agnostic
Finite $\|\mathcal{H}\|$	**Thm. 1** $N \geq \frac{1}{\epsilon}\left[\log(\|\mathcal{H}\|) + \log(\frac{1}{\delta})\right]$ labeled examples are sufficient so that with probability $(1-\delta)$ all $h \in \mathcal{H}$ with $\hat{R}(h) = 0$ have $R(h) \leq \epsilon$.	**Thm. 2** $N \geq \frac{1}{2\epsilon^2}\left[\log(\|\mathcal{H}\|) + \log(\frac{2}{\delta})\right]$ labeled examples are sufficient so that with probability $(1-\delta)$ for all $h \in \mathcal{H}$ we have that $\|R(h) - \hat{R}(h)\| \leq \epsilon$.
Infinite $\|\mathcal{H}\|$		