10-301/601 Introduction to Machine Learning

Machine Learning Department
School of Computer Science
Carnegie Mellon University

Decision Trees (Part I)

Matt Gormley & Henry Chai
Lecture 2
Sep. 1, 2021

Q&A

Q: Is my home here in Pittsburgh going to be washed away in a flood?

A: Probably not, unless you live down by the river. You can explore flood risk information at https://pafloodrisk.psu.edu/?address=15213

Q&A

Q: In Lecture 1, why did we use the term **experience** instead of just **data**?

A: Because our concern isn't just the data itself, but also where the data comes from (e.g. an agent interacting with the world vs. knowledge from a book).

As well, the word *experience* better aligns with the notion of what humans require in order to learn.

Q&A

Q: Who is the single person that will most ensure that this course runs smoothly this semester?

A: Okay… it's actually two people:

Joshmin Ray and Fatima Kizilkaya

Q&A

Q: Are we using Canvas?

A: No.

Q&A

Q: Can we have the handwritten notes from lectures?

A: Okay fine…

https://1drv.ms/u/s!Aqk9RupCw3gqixxHH34qLcj5uJTQ?e=E9OYu7

…but just be warned that lots of education research suggests that taking your own notes is the best way to learn!

Reminders

- **Homework 1: Background**
  - **Out: Wed, Sep 1 (2nd lecture)**
  - **Due: Wed, Sep 8 at 11:59pm**
  - Two parts:
    1. written part to Gradescope
    2. programming part to Gradescope
  - unique policy for this assignment:
    1. **two submissions** for written (see writeup for details)
    2. **unlimited submissions** for programming (i.e. keep submitting until you get 100%)
  - unique policy for this assignment: we will grant (essentially) any and all extension requests

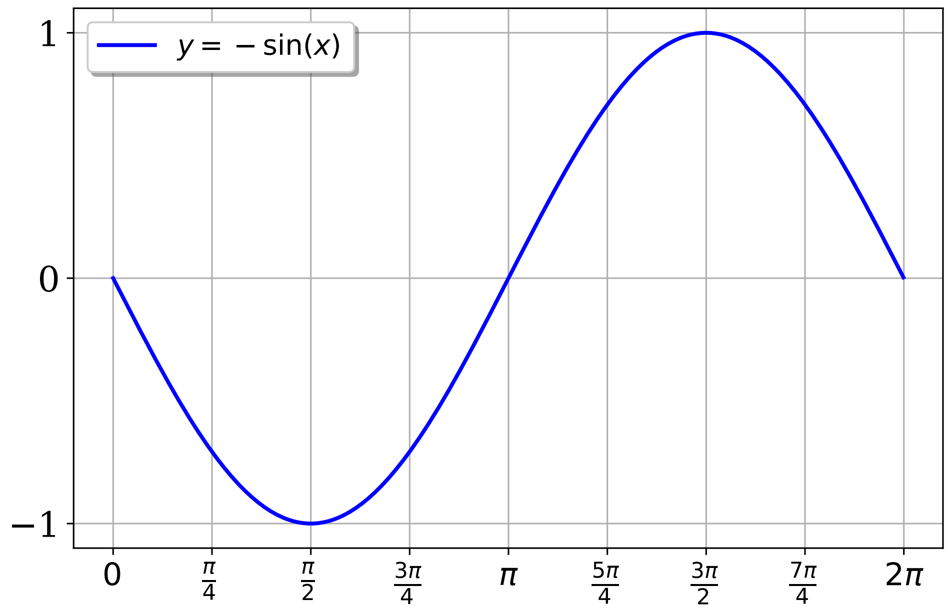
Big Ideas

1. How to formalize a learning problem
2. How to learn an expert system (i.e. Decision Tree)
3. Importance of inductive bias for generalization
4. Overfitting

FUNCTION APPROXIMATION

Function Approximation

Quiz: Implement a simple function which returns -sin(x).



A few constraints are imposed:

1. You can't call any other trigonometric functions
2. You *can* call an existing implementation of sin(x) a few times (e.g. 100) to test your solution
3. You only need to evaluate it for x in [0, 2*pi]

SUPERVISED MACHINE LEARNING

Medical Diagnosis

- Setting:
  - Doctor must decide whether or not patient is sick
  - Looks at attributes of a patient to make a medical diagnosis
  - (Prescribes treatment if diagnosis is positive)
- Key problem area for Machine Learning
- Potential to reshape health care

Medical Diagnosis

Interview Transcript
Date: Aug. 15, 2021
Parties: Matt Gormley and Doctor S.
Topic: Medical decision making

- Matt: Welcome. Thanks for interviewing with me today.
- Dr. S: Interviewing…?
- Matt: Yes. For the record, what type of doctor are you?
- Dr. S: Who said I'm a doctor?
- Matt: I thought when we set up this interview you said—
- Dr. S: I'm a preschooler.
- Matt: Good enough. Today, I'd like to learn how you would determine whether or not your little brother is allergic to cats given his symptoms.
- Dr. S: He's not allergic.
- Matt: We haven't started yet. Now, suppose he is sneezing. Does he have allergies to cats?
- Dr. S: Well, we don't even have a cat, so that doesn't make any sense.
- Matt: What if he is itchy; Does he have allergies?
- Dr. S: No, that's just a mosquito.
- [Editor's note: preschoolers unilaterally agree that itchiness is always caused by mosquitos, regardless of whether mosquitos were/are present.]
- Matt: What if he's both sneezing and itchy?
- Dr. S: Then he's allergic.
- Matt: Got it. What if your little brother is sneezing and itchy, plus he's a doctor.
- Dr. S: Then, thumbs down, he's not allergic.
- Matt: How do you know?
- Dr. S: Doctors don't get allergies.
- Matt: What if he is not sneezing, but is itchy, and he is a fox….
- Matt: …and the fox is in the bottle where the tweetle beetles battle with their paddles in a puddle on a noodle-eating poodle.
- Dr. S: Then he is must be a tweetle beetle noodle poodle bottled paddled muddled duddled fuddled wuddled fox in socks, sir. That means he's definitely allergic.
- Matt: Got it. Can I use this conversation in my lecture?
- Dr. S: Yes

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?**	**has 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?**	**has 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?**	**has 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?**	**has 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

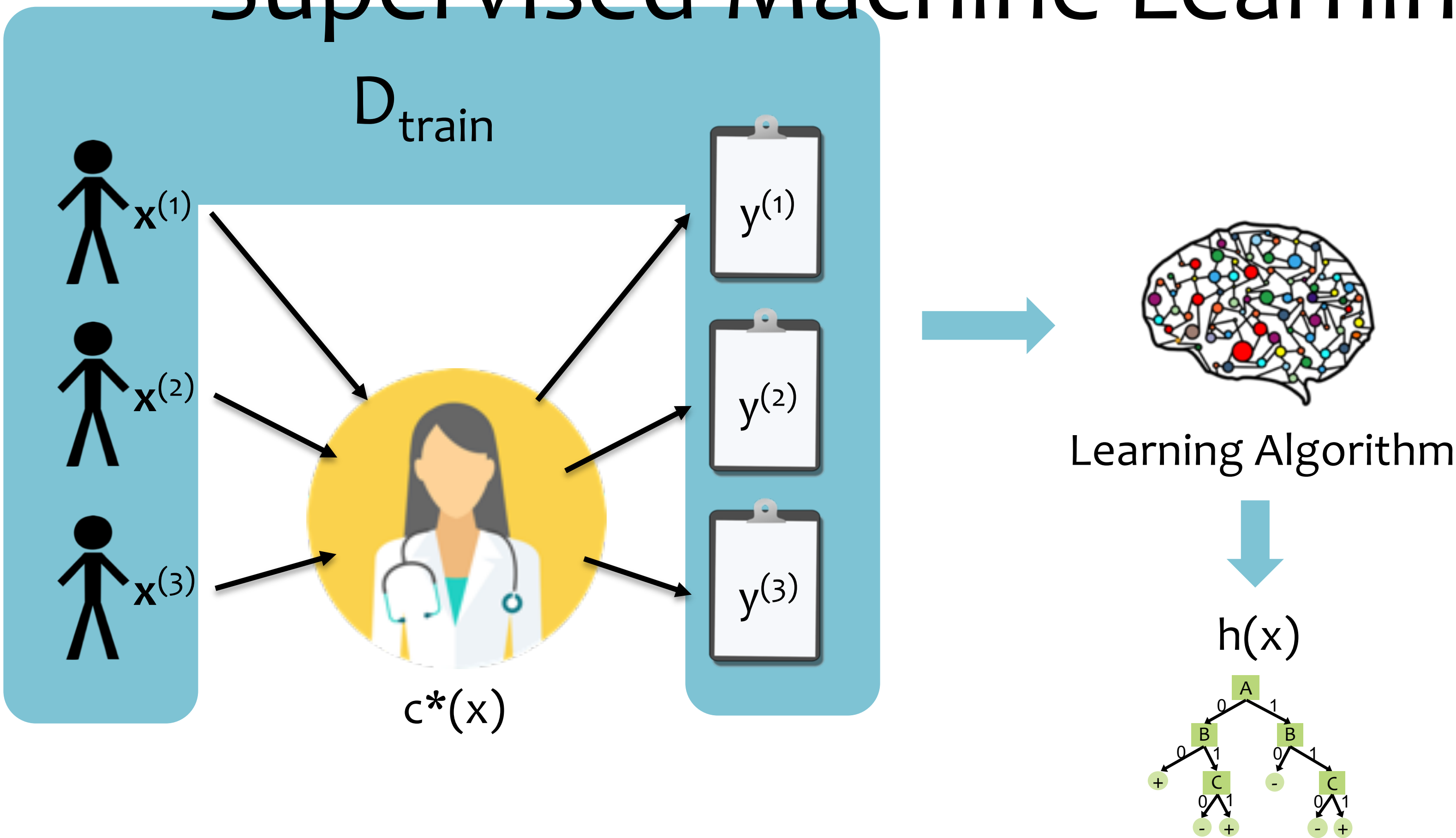
M = 4 attributes

ML as Function Approximation

Chalkboard

- ML as Function Approximation
  - Problem setting
  - Input space
  - Output space
  - Unknown target function
  - Hypothesis space
  - Training examples
  - Goal of Learning

Supervised Machine Learning



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?**	**has 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:

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

Supervised Machine Learning

- **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)
  - 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 $y^{(i)} = c^*(\mathbf{x}^{(i)})$ (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

Supervised Machine Learning

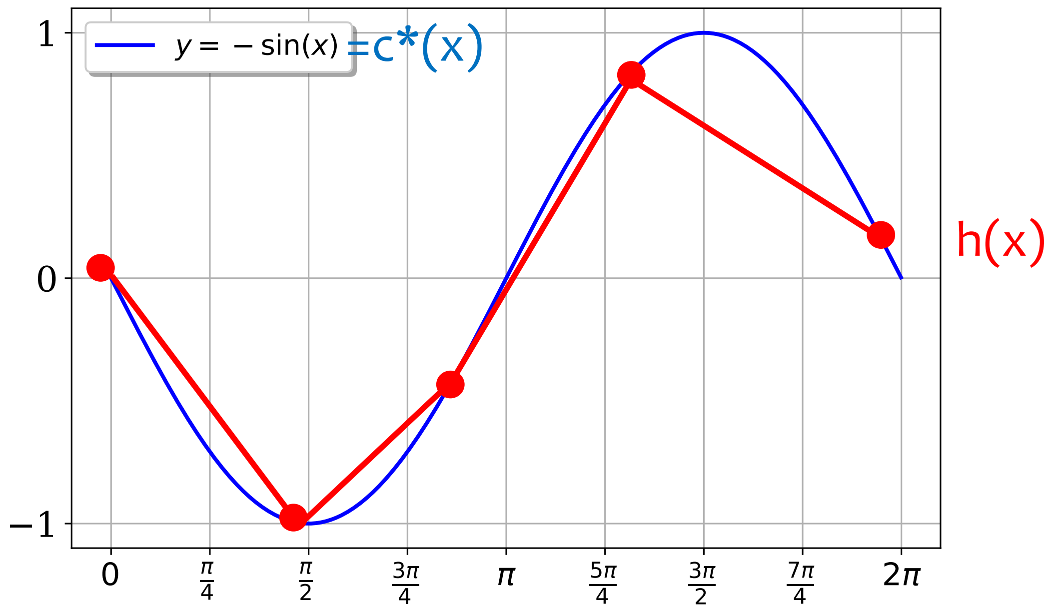
- **Problem Setting**
  - Set of possible inputs, $\mathbf{x} \in \mathcal{X}$ (all possible patients)
  - Set of possible outputs, $y \in \mathcal{Y}$ (all possible diagnoses)
  - Exists an un[...]own target function, $c^* : \mathcal{X} \rightarrow \mathcal{Y}$ (the doctor's b[...]
  - Set, $\mathcal{H}$, of candid[...] (all possible deci[...]
- **Learner is given** N[...] $D = \{(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2}$[...] where $y^{(i)} = c^*(\mathbf{x}^{(i)}$[...] (history of patient[...]
- **Learner produces**[...] that best approxi[...] $c^*(x)$ on the traini[...]

Two important settings we'll consider:

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

Function Approximation

Quiz: Implement a simple function which returns -sin(x).



A few constraints are imposed:

1. You can't call any other trigonometric functions
2. You *can* call an existing implementation of sin(x) a few times (e.g. 100) to test your solution
3. You only need to evaluate it for x in [0, 2*pi]

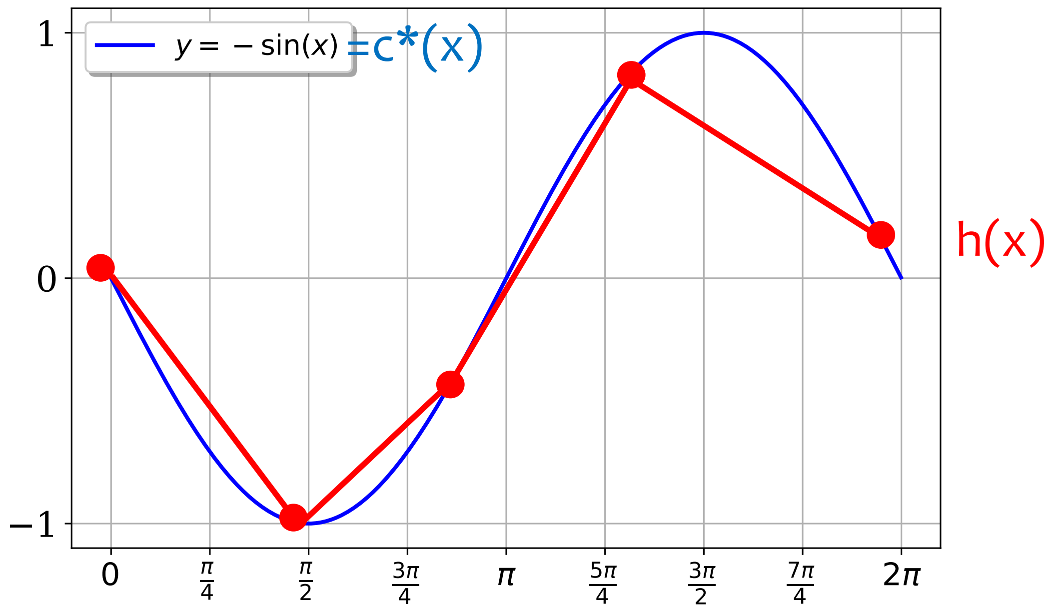
Supervised Machine Learning

- **Problem Setting**
  - Set of possible inputs, $\mathbf{x} \in \mathcal{X}$ (all values in [0, 2*pi])
  - Set of possible outputs, $y \in \mathcal{Y}$ (all values in [-1,1])
  - Exists an unknown target function, $c^* : \mathcal{X} \rightarrow \mathcal{Y}$
    ($c^*(x) = \sin(x)$)
  - Set, $\mathcal{H}$, of candidate hypothesis functions, $h : \mathcal{X} \rightarrow \mathcal{Y}$
    (all possible piecewise linear functions)
- **Learner is given** N training examples
  $D = \{(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \ldots, (\mathbf{x}^{(N)}, y^{(N)})\}$
  where $y^{(i)} = c^*(\mathbf{x}^{(i)})$
  (true values of sin(x) for a few random x's)
- **Learner produces** a hypothesis function, $\hat{y} = h(x)$, that best approximates unknown target function $y = c^*(x)$ on the training data

EVALUATION OF MACHINE LEARNING ALGORITHM

Function Approximation

Quiz: Implement a simple function which returns -sin(x).



How well does h(x) approximate c*(x)?

A few constraints are imposed:

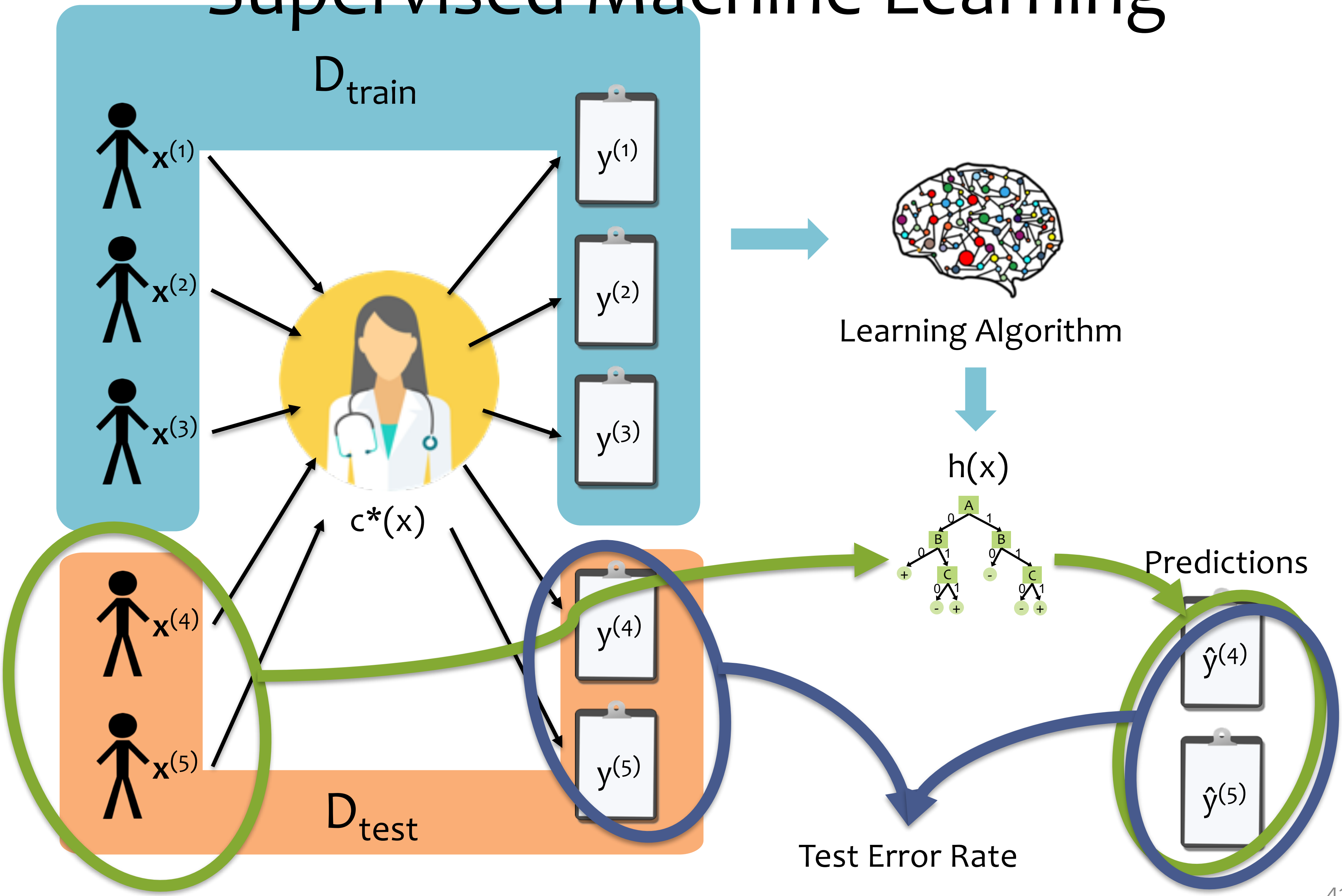
1. You can't call any other trigonometric functions
2. You *can* call an existing implementation of sin(x) a few times (e.g. 100) to test your solution
3. You only need to evaluate it for x in [0, 2*pi]

Evaluation of ML Algorithms

Chalkboard

- How to evaluate an ML algorithm?
- Definition: Loss function
  - Example for regression
  - Example for classification
- Definition: Error Rate
- Test dataset
- "Training" vs. "Testing"

Supervised Machine Learning



Error Rate

- Consider a hypothesis *h* its…
  - … error rate over all training data: error(h, D_{train})
  - … error rate over all test data: error(h, D_{test})
  - … true error over all data: $error_{true}$(h)

In practice, $error_{true}$(h) is **unknown**

LEARNING ALGORITHMS FOR SUPERVISED CLASSIFICATION

ML as Function Approximation

Chalkboard

- Algorithm 0: Memorizer
- Aside: Does memorization = learning?
- Algorithm 1: Majority Vote

Majority Vote Classifier Example

Dataset:

Output Y, Attributes A and B

Y	A	B
-	1	0
-	1	0
+	1	0
+	1	0
+	1	1
+	1	1
+	1	1
+	1	1

In-Class Exercise

What is the **training error** (i.e. *error rate on the training data*) of the **majority vote classifier** on this dataset?

Choose one of:
{0/8, 1/8, 2/8, …, 8/8}

ML as Function Approximation

Chalkboard

- Algorithm 2: Decision Stump
- Algorithm 3 (preview): Decision Tree