CSE 417T
Introduction to Machine Learning

Instructor: Chien-Ju Ho

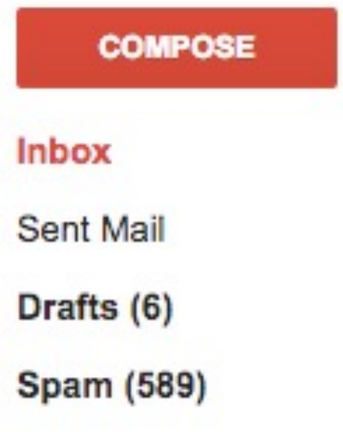
Plan for Today

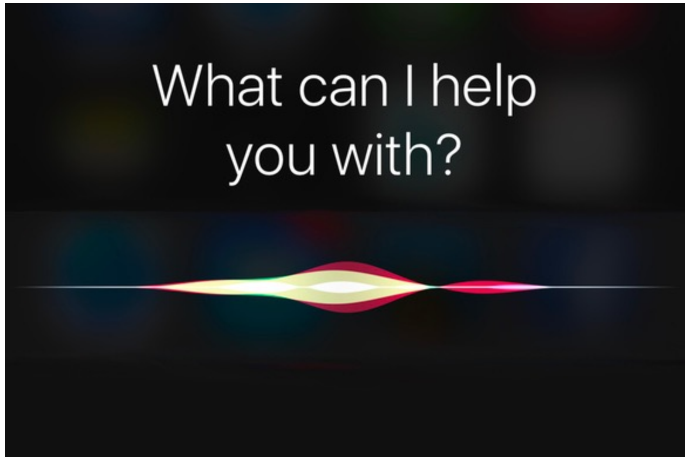
- Welcome and introduction
- What’s the class about?
- Logistics

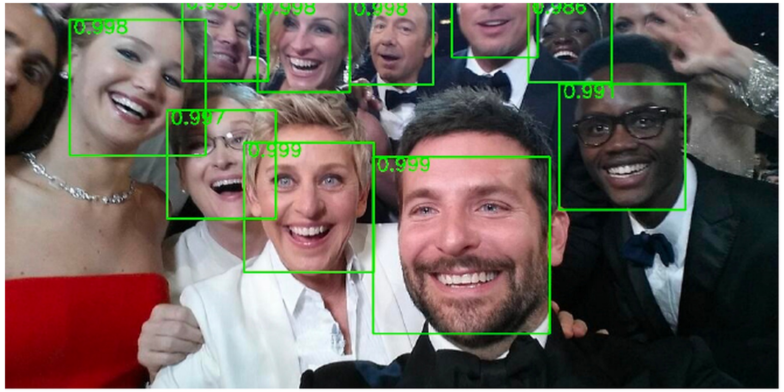
- Lecture
  - Setting up the learning problem
  - Perceptron learning algorithm

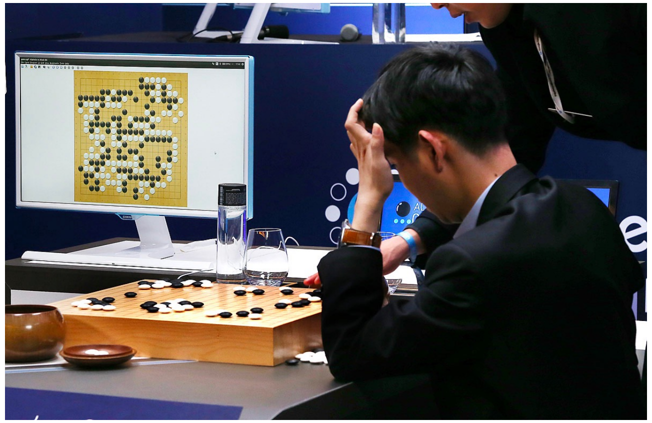
What is Machine Learning?

What Machine Learning Can Do?









Example: Credit Card Approval

Input: customer information

age	32 years
gender	male
salary	40,000
debt	26,000
years in job	1 year
years at home	3 years
. . .	. . .

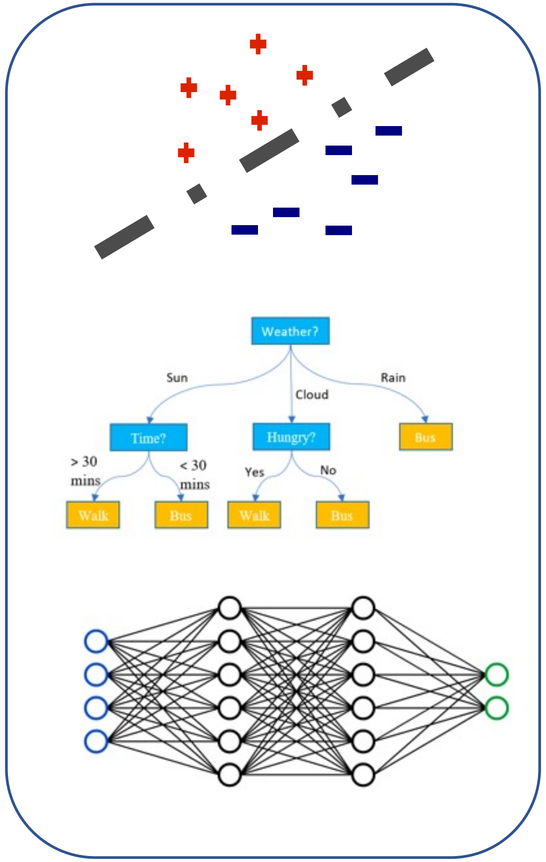
Output: a prediction

Will the customer be a good customer for the bank?

- A brute-force solution
  - Build a large table that maps all possible attribute combinations to a prediction
- Not really feasible at scale
  - Possible attribute combinations could be infinite
  - Storage, computation
- How to come up with the table?

A "Machine Learning" Approach

Hypothesis / Model (Some math function)



Data

(Historical Customer Data)

Find a hypothesis that "fits" the data
(The process requires a lot of computation)

What is Machine Learning?

“learning from data”

“using a set of observations to uncover an underlying pattern”

Use scenarios of machine learning

- A pattern exists
- No analytical solution: We cannot pin it down mathematically
- We have data on it

More Formally (For Supervised Learning)

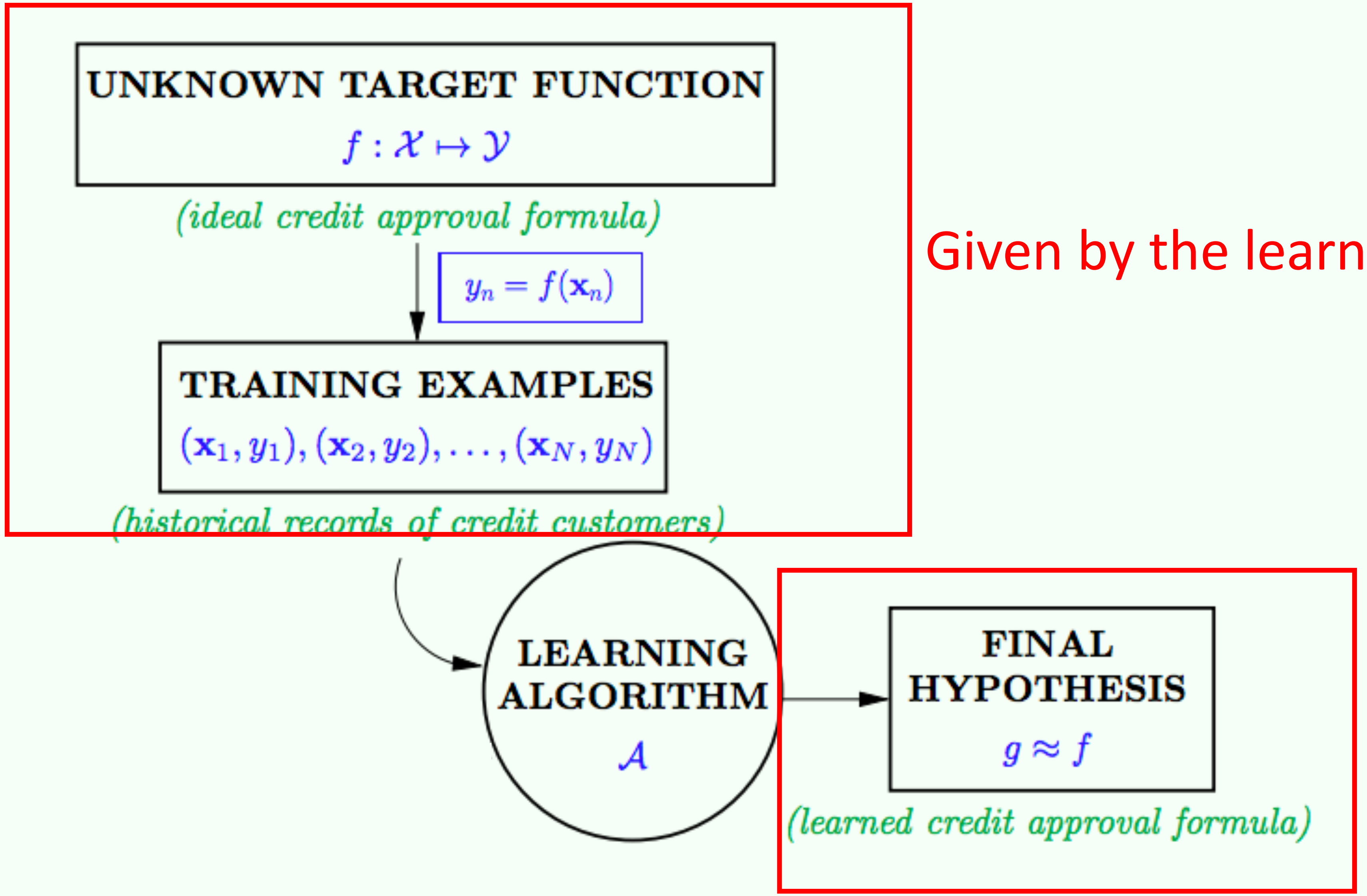
- Formulation: (credit card approval example)
  - input (features): $\vec{x} = (x_1, x_2, \ldots, x_d) \in X$ (customer's information)
  - output (label): $y \in Y$ (good/bad customer)
  - unknown target function: $f: X \to Y$ (ideal credit approval formula)
  - data $(\vec{x}_1, y_1), \ldots, (\vec{x}_N, y_N)$ (historical records)

  - goal: learn a g close to f (formula to be used)

- Two central questions
  - What can we say about how close g is to f?
  - How do we learn g?

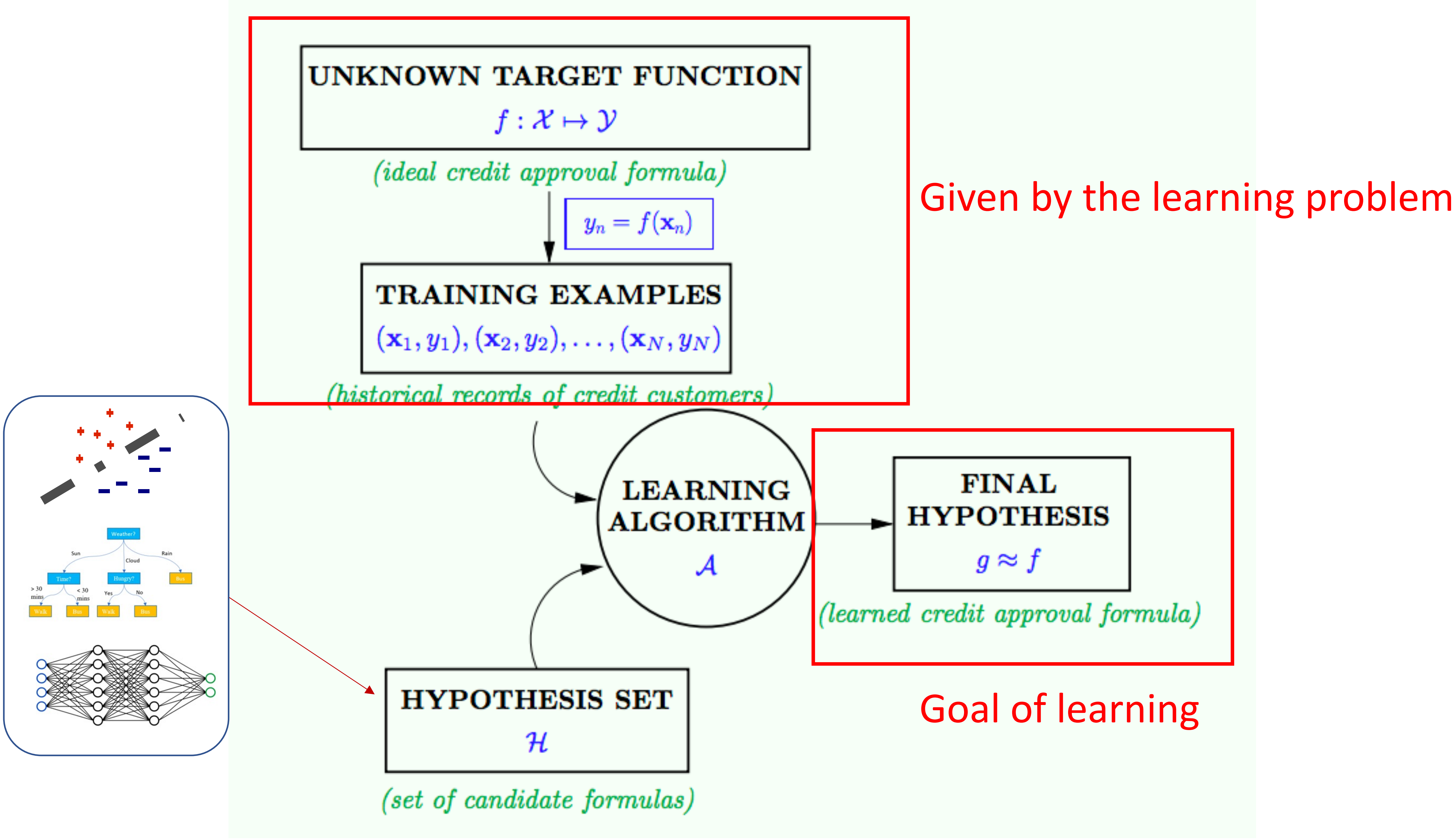
Note on notations:
We interchangeably use (bold font) $\boldsymbol{x}$ and $\vec{x}$ to denote a column vector in this course.

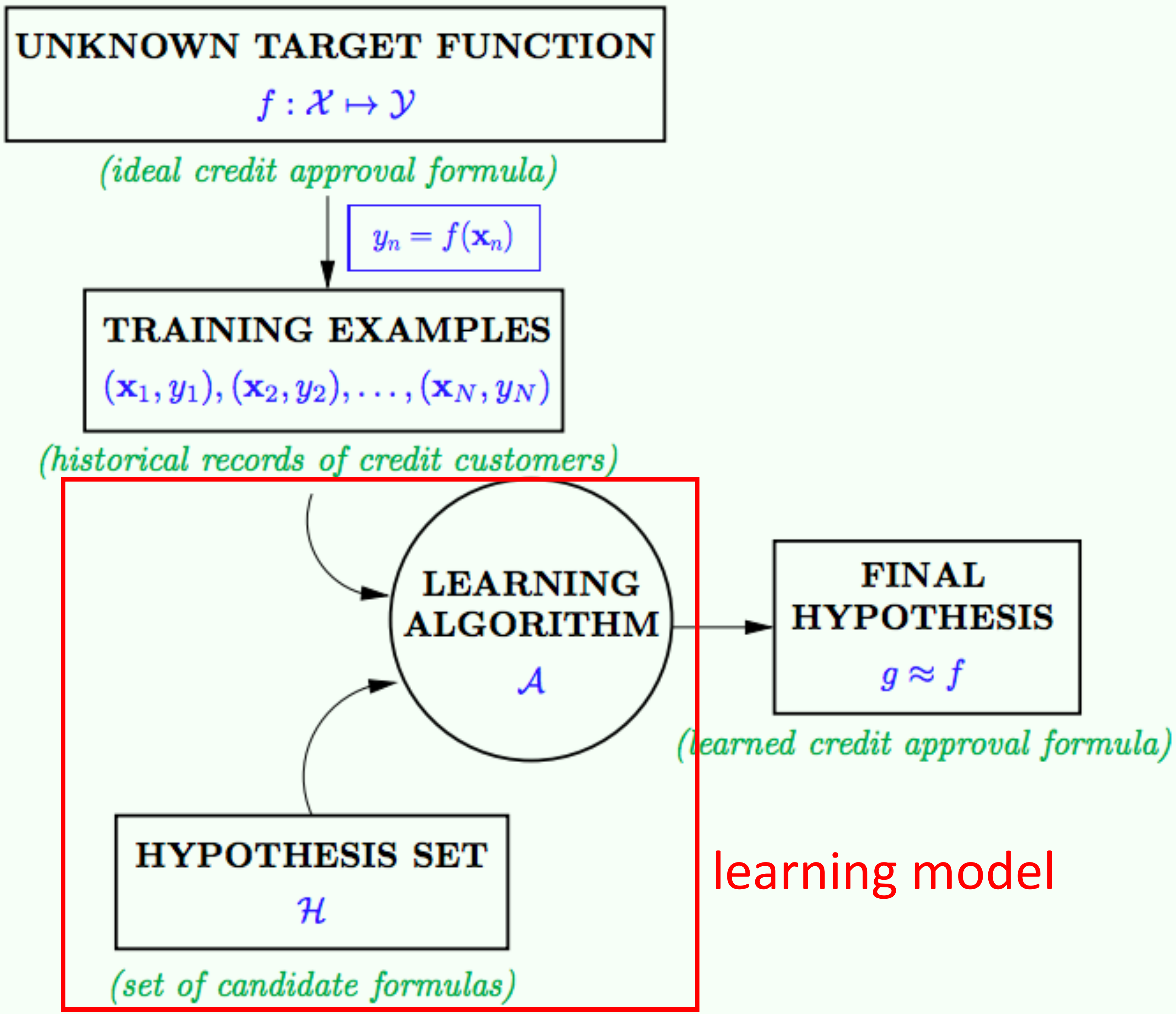
The former is used in the textbook.
the later is for the convenience of writing.



Given by the learning problem

Goal of learning


UNKNOWN TARGET FUNCTION
$f : \mathcal{X} \mapsto \mathcal{Y}$
(ideal credit approval formula)
$y_n = f(\mathbf{x}_n)$
TRAINING EXAMPLES
$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)$
(historical records of credit customers)
Given by the learning problem
LEARNING ALGORITHM
$\mathcal{A}$
FINAL HYPOTHESIS
$g \approx f$
(learned credit approval formula)
Goal of learning
HYPOTHESIS SET
$\mathcal{H}$
(set of candidate formulas)
Weather?
Sun
Cloud
Rain
Time?
Hungry?
Bus
> 30 mins
< 30 mins
Yes
No
Walk
Bus
Walk
Bus


UNKNOWN TARGET FUNCTION
$f : \mathcal{X} \mapsto \mathcal{Y}$
(ideal credit approval formula)
$y_n = f(\mathbf{x}_n)$
TRAINING EXAMPLES
$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)$
(historical records of credit customers)
LEARNING ALGORITHM
$\mathcal{A}$
FINAL HYPOTHESIS
$g \approx f$
(learned credit approval formula)
HYPOTHESIS SET
$\mathcal{H}$
(set of candidate formulas)
learning model

Course Plan

- First half of the semester: **Foundations**
  - Focus on **linear models**
    - Fundamental components of many other models
  - Discuss the theoretical foundations of machine learning
    - Heavy use of probability, linear algebra, and optimization

- Second half of the semester: **Techniques**
  - Discuss different learning models

Course Plan

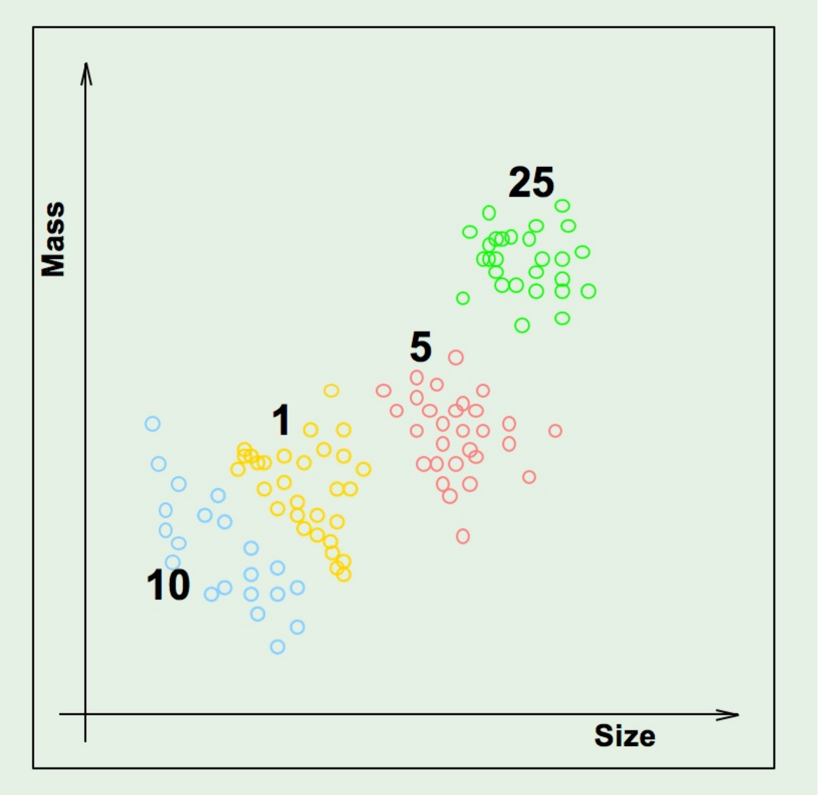
- Foundations
  - What’s machine learning
  - Feasibility of learning
  - Generalization
  - Linear models
  - Non-linear transformations
  - Overfitting and how to avoid it
    - Regularization
    - Validation
- Techniques
  - Nearest neighbors
  - Decision tree
  - Support vector machine
  - Boosting
  - Random forest
  - Neural networks
  - …

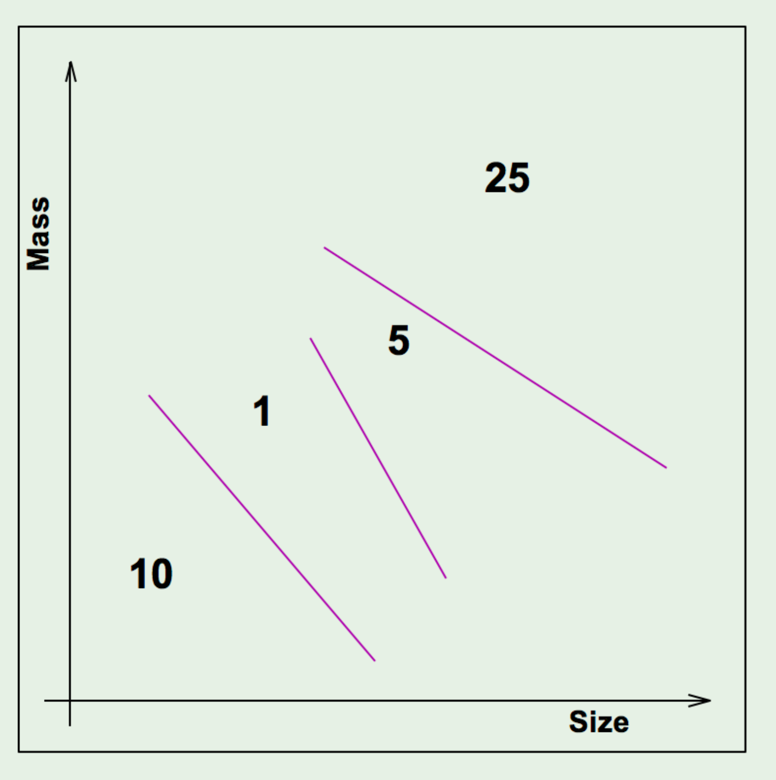
Types of Learning

- Supervised learning (the focus of this course)
  - Given training data (input, correct output)
  - Try to predict the output for data not seen before
- Unsupervised learning
  - Given data in the form of (input)
  - Find patterns in data
- Reinforcement learning
  - Learn how to act, based on rewards for actions

Supervised Learning

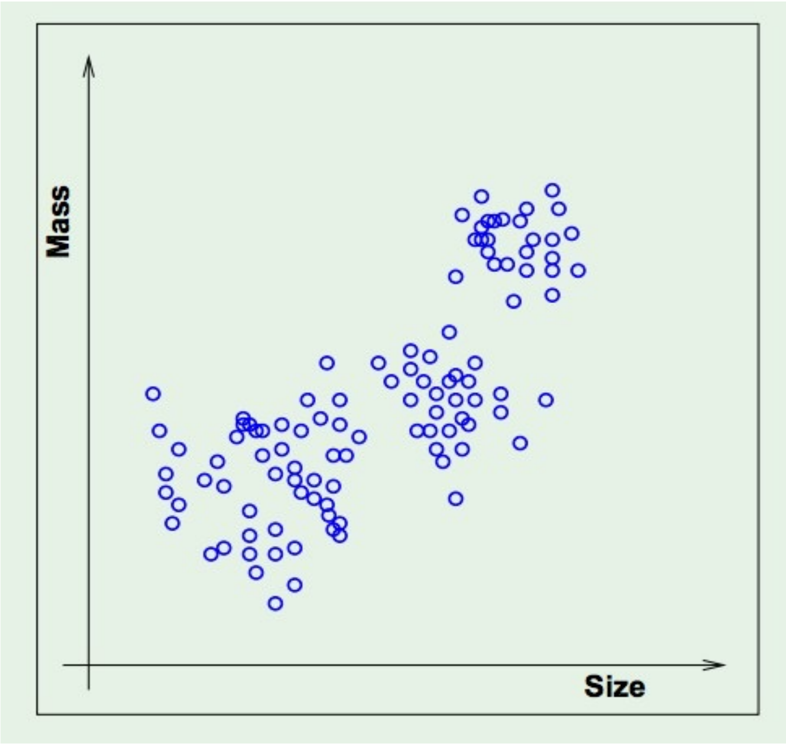
- Given data with (input, correct output), learn a pattern that can predict previously unseen data

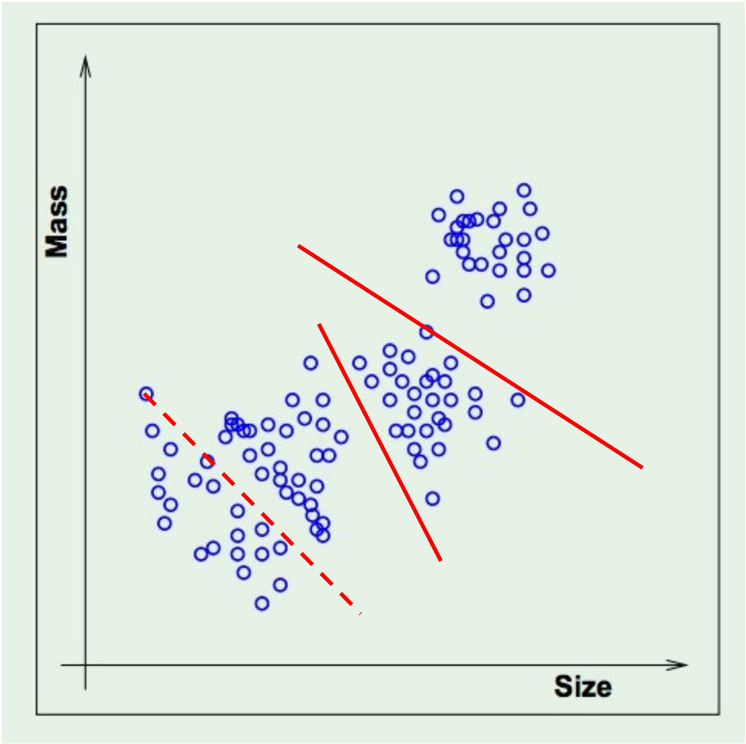




Unsupervised Learning (more in 517A)

- Suppose you only have the feature vectors but no labels. Still want to describe the data in some useful way.

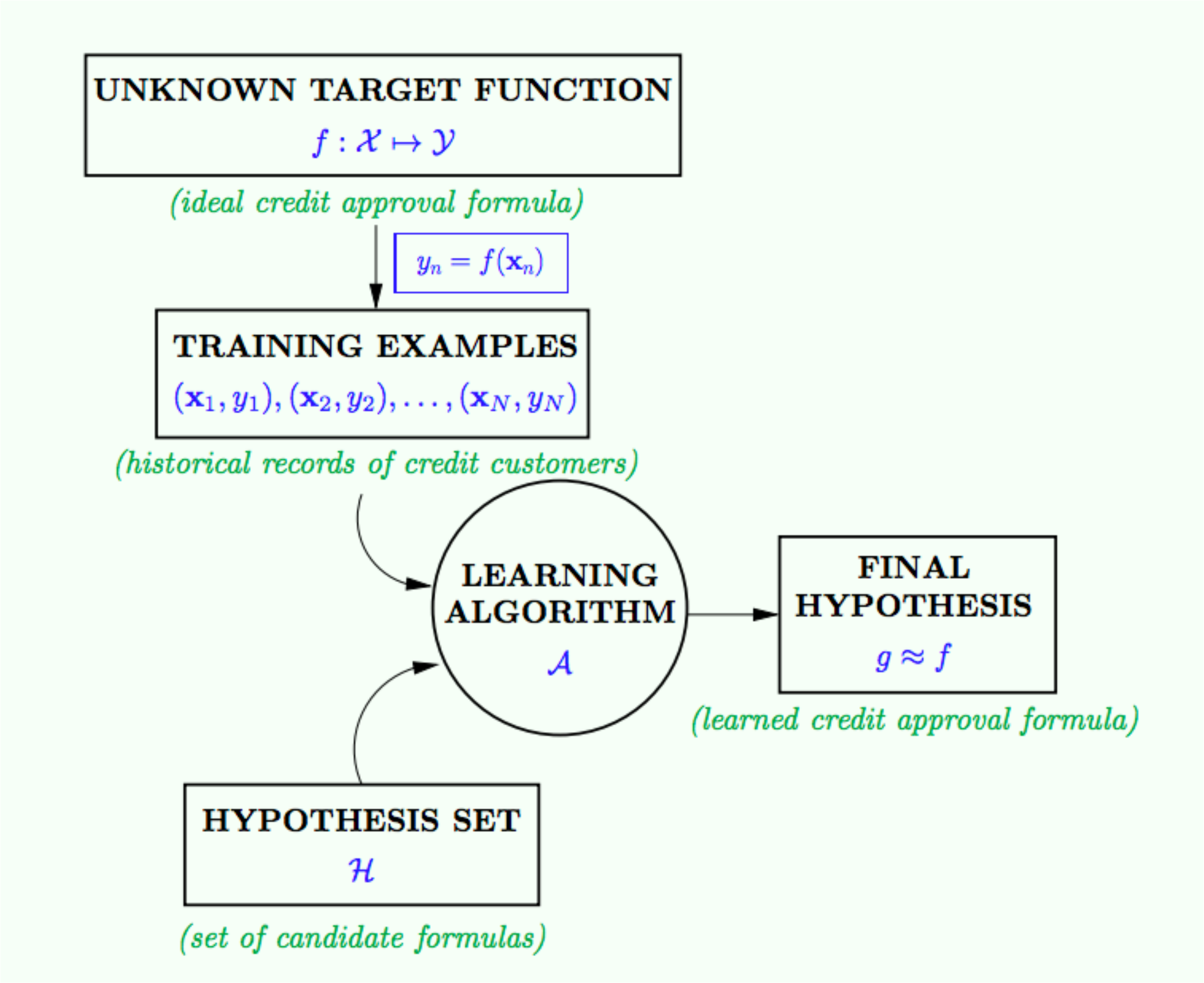




Reinforcement Learning (more in 412A)

- Agent interacts with the world by taking actions
- Feedback is in the form of rewards (or costs)
- Agent must learn a policy, which maps from the state of the world to an action
- Major challenges:
  - Exploration / exploitation
  - Delayed reward / credit assignment

Course plans (focus on supervised learning)



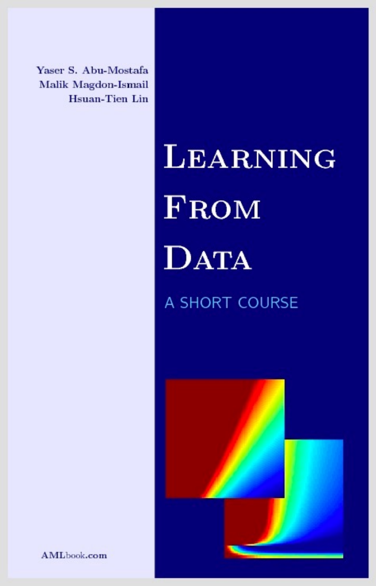
Logistics

- Websites
  - Course website: http://chienjuho.com/courses/cse417t
  - Piazza for discussion
  - Gradescope for homework submissions
  - You are responsible for following the announcements on website and Piazza

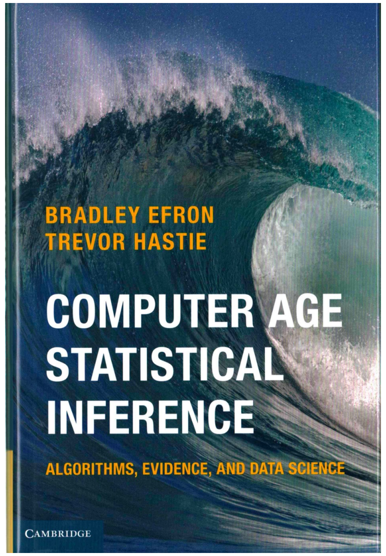
- TA and Office Hours
  - There will be several graduate/undergraduate TAs
  - Office hours will be announced in the 2nd week and start in the 3rd week

Course Information: Textbooks

- *Learning From Data*
  - Y. Abu-Mostafa, M. Magdon-Ismail, and H-T Lin.
  - http://amlbook.com/
  - We will go through this book in the first half of the semester.



- *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science*.
  - B. Efron and T. Hastie
  - https://web.stanford.edu/~hastie/CASI/
  - We will reference a few sections as the course materials. The PDF file is freely available on their website



Course Information: Grading

- Homework assignments (5 to 6): 50%
  - Mix of programming and pencil-and-paper problems
  - Worst score discounted by 50%
  - Programming language: Python
    - We don’t teach how to program Python
  - 5 total late days, no more than 2 on any one assignment

- Two exams: 50% (25% each)
  - One in the middle of the semester (sometime in mid-to-late October)
  - One on the last lecture of the semester
  - Each exam covers around half of the materials. No separate final exam.
  - More details will be announced later

Course Information: Academic Integrity

- Take a look at the syllabus
  - Collaborations
    - You are encouraged to discuss homework with other students
    - **Must write your own solutions**
    - **Must cite all external sources (including other students)**

  - Other accommodation resources

  - Academic integrity
    - **Zero tolerance** on the violation
    - Will be reported to the university
    - There will be permanent record if found guilty

Course Information

- Questions not covered in the syllabus?
  - Ask me!
  - Generally, I don’t grant individual exceptions
    - Can I do extra work to get more points?
    - I have another exam this week, can I get an additional day for the assignment?
    - I work really hard but can’t finish the assignment on time. Can I get an additional day for the assignment?
    - I work really hard. Can I get higher grades?
  - **No** to all the above.
    - Exceptions: family/medical emergencies

  - Rule of thumb:
    - I only say yes if I feel comfortable giving the same treatment to everyone in the class.

Getting in Touch

- Please use **Piazza** as the main communication channel
  - Emails will likely not be responded
- Use public posts in Piazza by default
  - Other students might have the same questions
  - Other students might help answer the questions
- When to use private posts
  - The questions involve disclosing your answers to the homework
    - Most of the time, you can reframe your questions to avoid this
  - The questions are about your personal matter

Is This Course for You?

- This is going to be a very theory-heavy course
  - The “T“ in CSE 417T stands for “Theory”
  - There will be **A LOT OF** math!
- We focus on understanding the foundations of machine learning
- If your main goal is to learn how to apply ML to solve problems, this might not be the best course for that.
  - Check out CSE 217A, CSE 412A, ESE 417, BME 440, INFO 558

Is This Course for You?

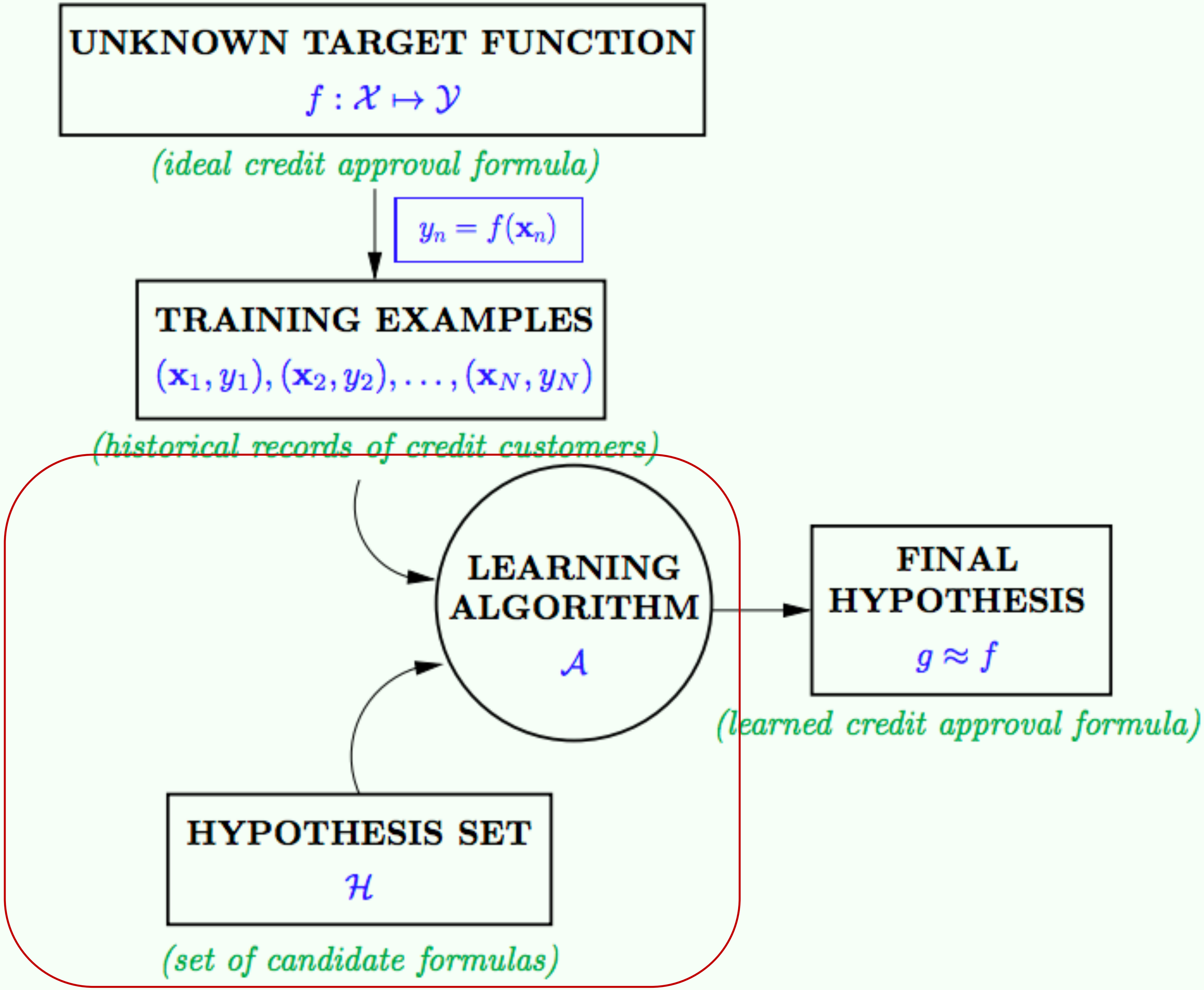
- Try homework 0 (posted on the website)
  - A subset of questions in homework 1.
  - You should feel comfortable working on these kinds of questions.
  - You should be ready to answer all questions there after the lecture today.

- If you are on the waitlist and want to get enrolled:
  - Complete homework 0
    - A mixture of math and programing questions
    - Explain why you want/need to take this course in this semester
  - Submit via Gradescope by noon next Tuesday (Sep 6)
  - Priorities will be given to students who benefit the most by taking this course this semester (condition on having satisfactory answers to the questions)

- If you are enrolled:
  - Don’t submit homework 0 yet, but you should check it out as well
  - The same questions will appear in homework 1
  - It helps you get a better sense of what this course is about

Questions?

Lecture Today



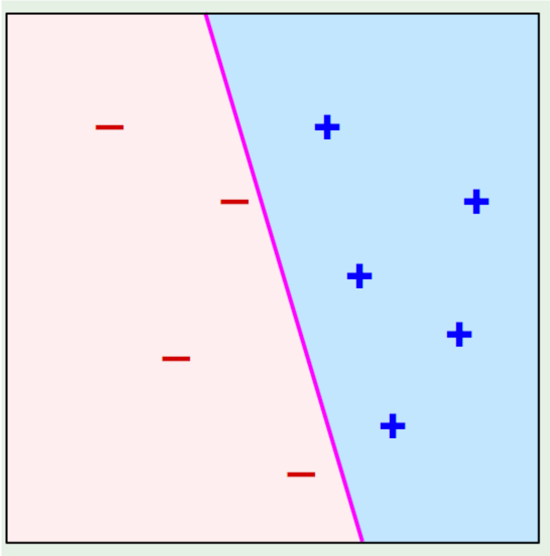
UNKNOWN TARGET FUNCTION
$f: \mathcal{X} \mapsto \mathcal{Y}$
(ideal credit approval formula)
$y_n = f(\mathbf{x}_n)$
TRAINING EXAMPLES
$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)$
(historical records of credit customers)
LEARNING ALGORITHM
$\mathcal{A}$
FINAL HYPOTHESIS
$g \approx f$
(learned credit approval formula)
HYPOTHESIS SET
$\mathcal{H}$
(set of candidate formulas)

Linear Hypothesis Space (Perceptron)

- Input $\vec{x} = (x_1, x_2, \dots, x_d)$
- Output $y \in \{-1, +1\}$

Note that $\vec{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix}$ is a column vector;

For convenience, we write $\vec{x} = (x_1, \dots, x_d)$



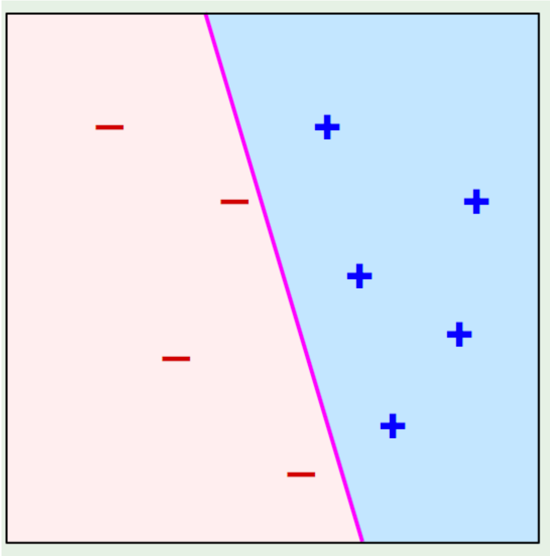
Linear Hypothesis Space (Perceptron)

- Input $\vec{x} = (x_1, x_2, \ldots, x_d)$
- Output $y \in \{-1, +1\}$

Note that $\vec{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_d \end{bmatrix}$ is a column vector;

For convenience, we write $\vec{x} = (x_1, \ldots, x_d)$

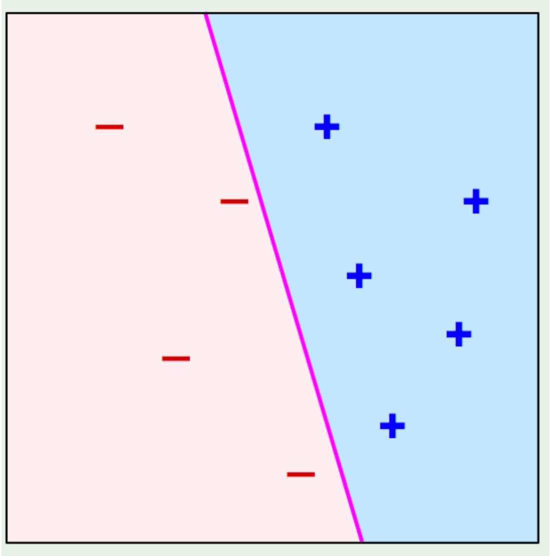
- A hypothesis h is a linear separator $\vec{w}^T\vec{x} = b$, characterized by
  - weight vector $\vec{w} = (w_1, \ldots w_d)$
  - threshold b



- $h(\vec{x}) = sign\left(\sum_{i=1}^{d} w_i x_i - b\right) = sign(\vec{w}^T\vec{x} - b)$
  - Predict $+1$ if $\vec{w}^T\vec{x} > b$
  - Predict -1 if $\vec{w}^T\vec{x} < b$

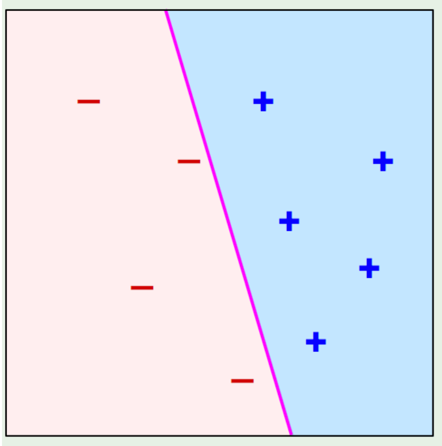
Linear Hypothesis Space (Perceptron)

- To simplify $h(\vec{x}) = sign(\vec{w}^T \vec{x} - b)$, define
  - $x_0 = 1$
  - $w_0 = -b$
- And we implicitly let
  - $\vec{x} = (x_0, x_1, \dots, x_d)$
  - $\vec{w} = (w_0, w_1, \dots, w_d)$
- A hypothesis can then be written as
  - $h(\vec{x}) = sign(\vec{w}^T \vec{x})$
    - We will interchangeably use h and $\vec{w}$ to express a hypothesis in Perceptron



Perceptron Learning Algorithm (PLA)

- Given a dataset $D = \{(\vec{x}_1, y_1), \ldots, (\vec{x}_N, y_N)\}$
- Assume the dataset is **linearly separable**
- How do we learn a hypothesis that separates the data?



- Perceptron Learning Algorithm
  - Initialize $\vec{w}(0) = \vec{0}$
  - For $t = 0, \ldots$
    - Find a misclassified data point $(\vec{x}(t), y(t))$ in D
      - That is, $\mathrm{sign}(\vec{w}(t)^T \vec{x}(t)) \neq y(t)$
    - If no such data point exists
      - Return $\vec{w}(t)$
    - Else
      - $\vec{w}(t+1) \leftarrow \vec{w}(t) + y(t)\vec{x}(t)$

Notation:
We use $\vec{w}(t)$ to denote the value of $\vec{w}$ at step t of the algorithm.

Similarly, we use $(\vec{x}(t), y(t))$ to denote the data point found at step t.

Some Intuitions

Perceptron Learning Algorithm (PLA)

- Theorem (informal):
  - If a dataset D is linearly separable, PLA find a linear separator that separates the data in D within a finite number of steps.

- HW0:
  - Prove Chebyshev's inequality (a form of the law of large numbers)
  - Prove the above theorem
  - Implement PLA using Python
  - Explain why you want/need to take this course this semester