CS 5806 Machine Learning II

Lecture 2 - Learning Formulation & Paradigms

August 23rd, 2023

Hoda Eldardiry

Lecture objective

- Learning Formulation
  - Define learning problems (learning components)
- Learning Paradigms
  - Connections & differences
  - When to use each approach
- Apply what we learn to brainstorm about course project

DEFINING LEARNING PROBLEMS

Learning problems

- Learning components
  —Task, T
  —Performance measure, P
  —Experience, E

- Learning definition*
  A computer program **learns** if its
  performance, measured by P,
  at **tasks in T**, improves with **experience E**

*Tom Mitchell
1997

Learning example

In-Class Exercise: Name each learning component below

Problem: Design a spam filter

- Classify emails as spam or not spam […]
- Watch you label emails as spam or not spam […]
- Number of emails correctly classified as spam/not spam […]

Learning example

Problem: Design a spam filter

- Classify emails as spam or not spam
  Task
- Watch you label emails as spam or not spam
  Experience
- Number of emails correctly classified as spam/not spam
  Performance measure

CAPTURING EXPERT KNOWLEDGE

Capturing expert knowledge

Problem — Learning to respond to voice commands (Siri)

Solution #1 — Expert systems

- **20 years ago:** rule-based system
- Ask the expert to:
  — Obtain a PhD in Linguistics
  — Introspect about language structure
  — Write down the rules they devise

Voice command	Rule to respond to voice command
Give me directions to Starbucks	If: “give me directions to X” Then: directions(here, nearest(X))
How do I get to Starbucks?	If: “how do i get to X” Then: directions(here, nearest(X))
Where is the nearest Starbucks?	If: “where is the nearest X” Then: directions(here, nearest(X))
I need directions to Starbucks	If: “I need directions to X” Then: directions(here, nearest(X))
Is there a Starbucks nearby?	If: “Is there an X nearby” Then: directions(here, nearest(X))
Starbucks directions	If: “X directions” Then: directions(here, nearest(X))

Capturing expert knowledge

Problem — Learning to respond to voice commands (Siri)

- **Experts**
  — Very good at answering questions about specific cases
  — Not very good at telling **how** they do it

- **1990s**
  — Why not just have experts tell you what they do on **specific cases**
  — Then let **ML** tell you how to come to the same decisions that they did

- **Solution #2 — Annotate Data and Learn**
  — Collect raw sentences $\{x_1, \ldots, x_n\}$
  — Experts annotate their meaning $\{y_1, \ldots, y_n\}$

Raw sentence x_i	Annotation y_i
x_1: How do I get to Starbucks?	y_1: directions(here, nearest(Starbucks))
x_2: Show me the closest Starbucks	y_2: map(nearest(Starbucks))
x_3: Send a text to John that I'll be late	y_3: txtmsg(John, I'll be late)
x_4: Set an alarm for seven in the morning	y_5: setalarm(7:00AM)

Learning problems

- **In-Class Exercise:**
  Identify T, P and E for the two learning problems below

- **Learning to play chess**
  Task, T:
  Performance measure, P:
  Experience, E:

- **Learning to respond to voice commands (Siri)**
  Task, T:
  Performance measure, P:
  Experience, E:

Learning problems

- Learning to play chess
  Task, T: playing chess
  Performance measure, P: probability program wins next game
  Experience, E: experience of playing many games

- Learning to respond to voice commands (Siri)
  Task, T: predict action from speech
  Performance measure, P: % correct actions taken
  Experience, E: examples of (speech, action) pairs

Learning tasks

- **In-Class Exercise: Example Learning Tasks**
  1. Select a task, T, from the list below
  2. Identify performance measure, P
  3. Identify experience, E

Example Tasks

—Identify objects in an image
—Translate from one human language to another
—Assess risk (e.g. in loan application)
—Make decisions (e.g. in loan application)
—Assess potential (e.g. in admission decisions)
—Categorize a complex situation (e.g. medical diagnosis)
—Predict outcome (e.g. medical prognosis, stock prices, inflation, temperature)
—Predict events (default on loans, quitting school, war)
—Plan ahead under partial knowledge (Poker, Bridge)
—Email spam filtering
—Plan ahead under perfect knowledge (chess)
—Recognize speech

ML Problem formulation

- The same **problem** can be formulated in various ways (ML Tasks)
- Example **learning problem**: Process loan applications
  – Task: Compute credit score (regression)
  – Task: Compute probability of default (density estimation)
  – Task: Generate loan decision (classification)

Structure of output prediction	Problem formulation
Boolean	Binary Classification
Categorical	Multi-class Classification
Ordinal	Ordinal Classification
Real	Regression
Ordering	Ranking
Multiple discrete	Structured Prediction
Multiple continuous	E.g. dynamical systems
Both discrete & continuous	E.g. mixed graphical models

Project Learning Components

In-Class Exercise: Identify a potential project topic

- What are the learning problem(s) involved?

- For each learning problem, list the learning components
  — Task?
  — Performance measure?
  — Experience?

LEARNING PARADIGMS

Objectives

- Learning paradigms
- Connections & differences
- When to use each approach
- Apply what we learn to brainstorm about course project

DIFFERENCES

Supervised & Unsupervised Learning

In-Class Exercise:

- What are some differences between supervised & unsupervised learning?
- Hints on next slide!

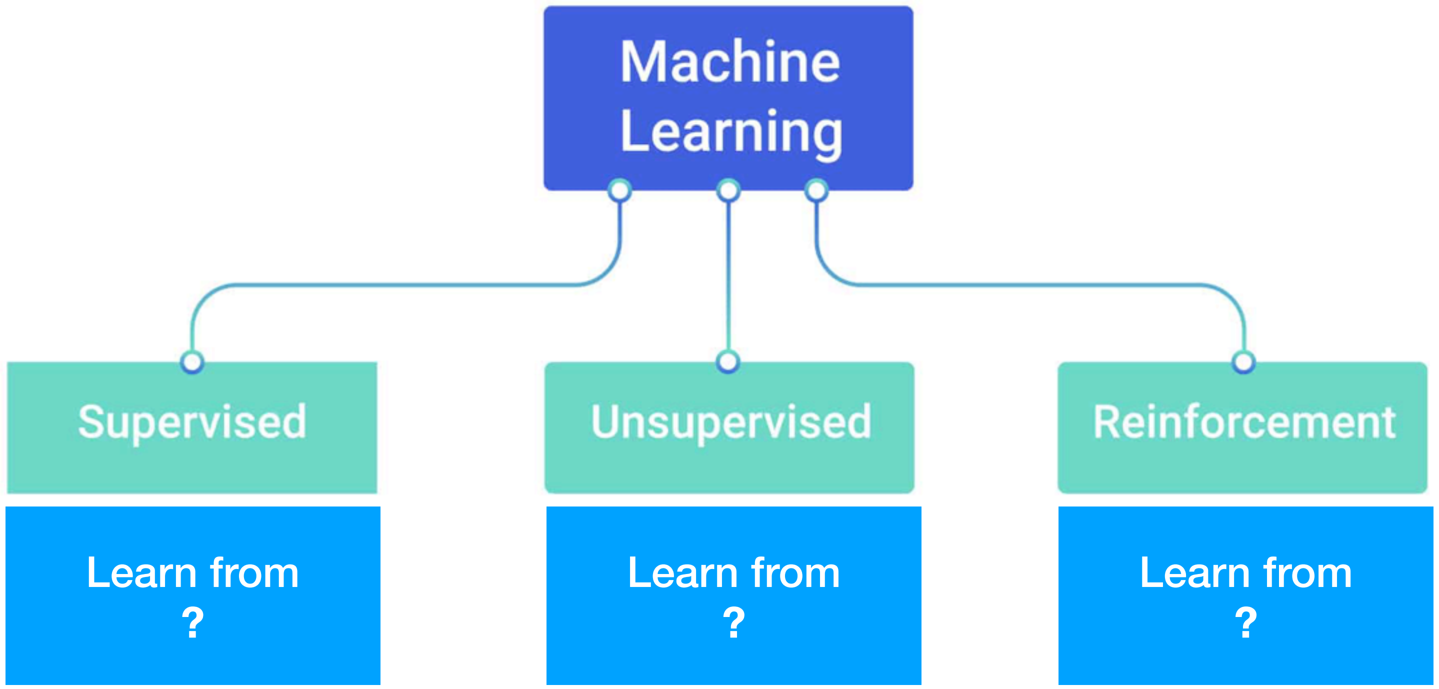
Supervised & Unsupervised Learning

In-Class Exercise:

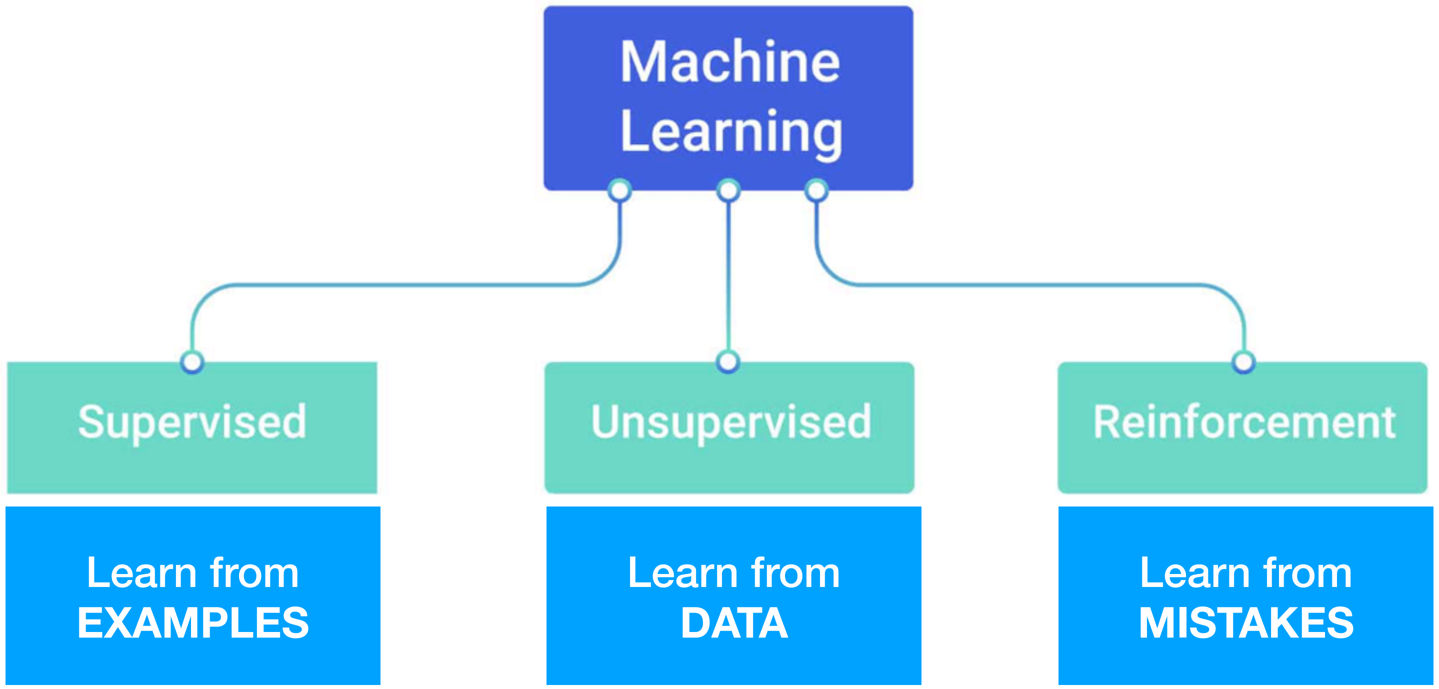
- What are some differences between supervised & unsupervised learning?

- Hints:
  — Input
  — Human involvement
  — Complexity
  — Dataset size
  — Accuracy
  — Constraints

Learning Paradigms



Learning Paradigms

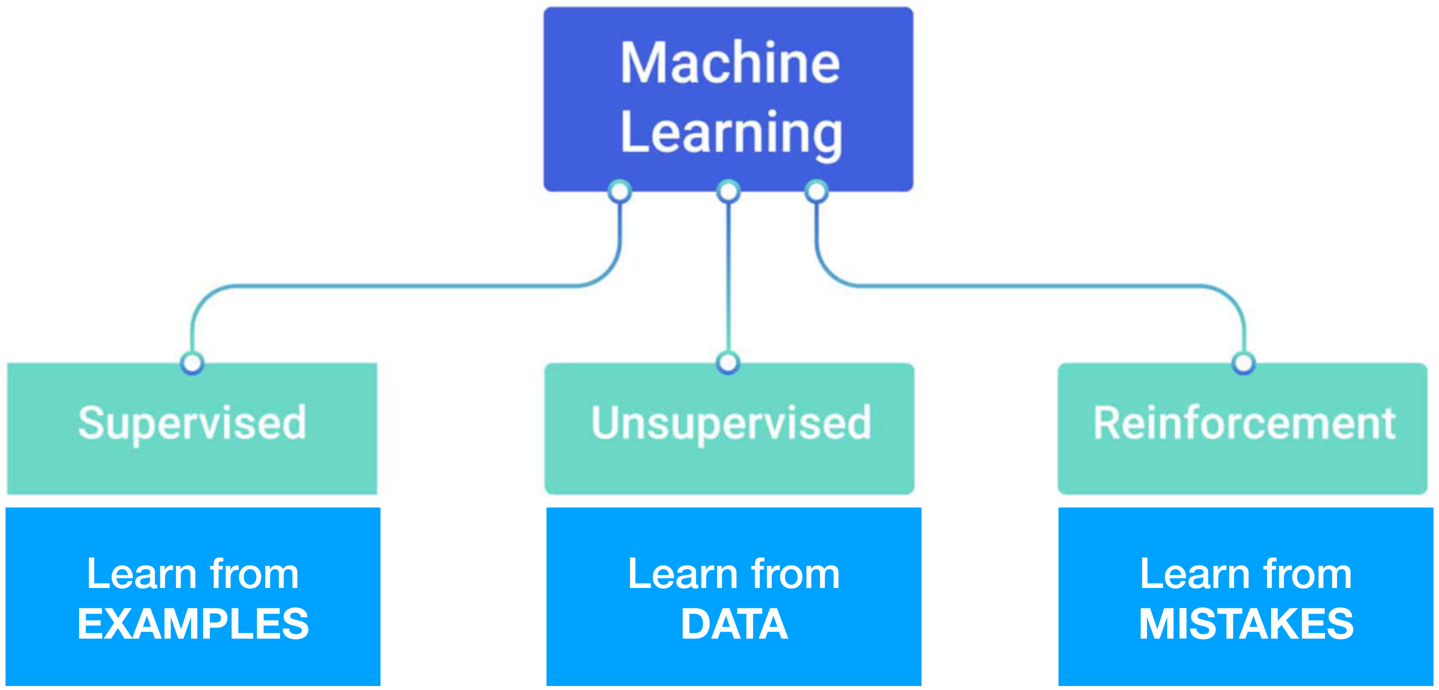


- Supervised learning
  — Input?
  — Output?

- Unsupervised learning
  — Input?
  — Output?

- Reinforcement learning
  — Input?
  — Output?

Learning Paradigms



- Supervised learning
  — Given sample of entity-label (example) pairs (x_i, y_i)
  — Find predictive relationship between entities and labels

- Unsupervised learning
  — Given a sample consisting of only entities
  — Find interesting structures in data (example: group similar entities)

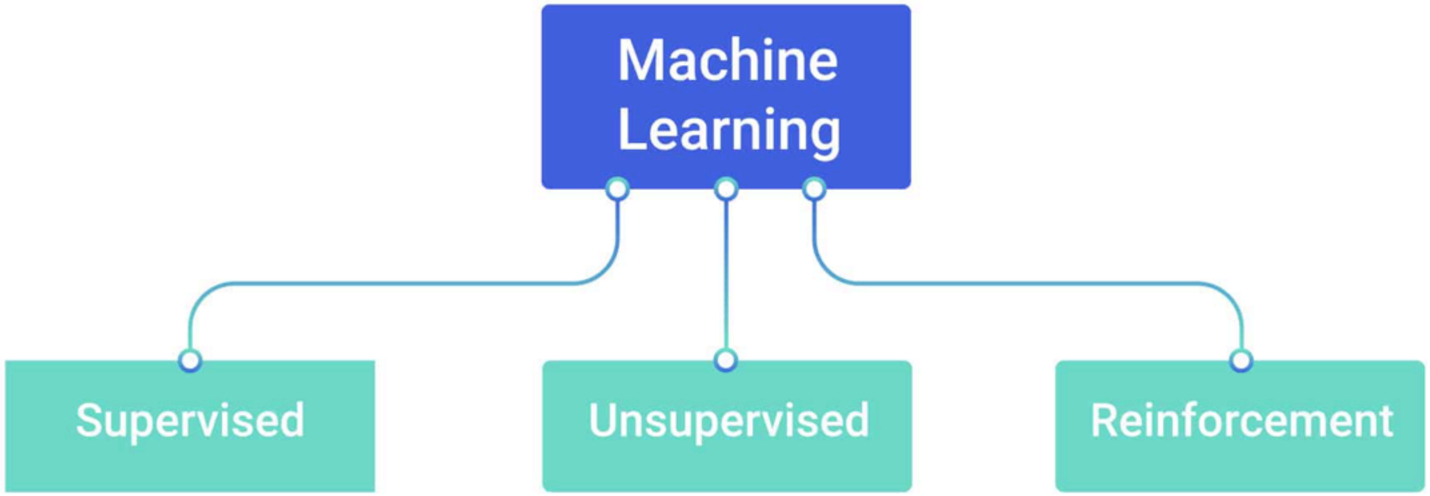
- Reinforcement learning
  — Agent takes action to maximize reward
  — Agent learns from trial and error

CONNECTIONS

Connections

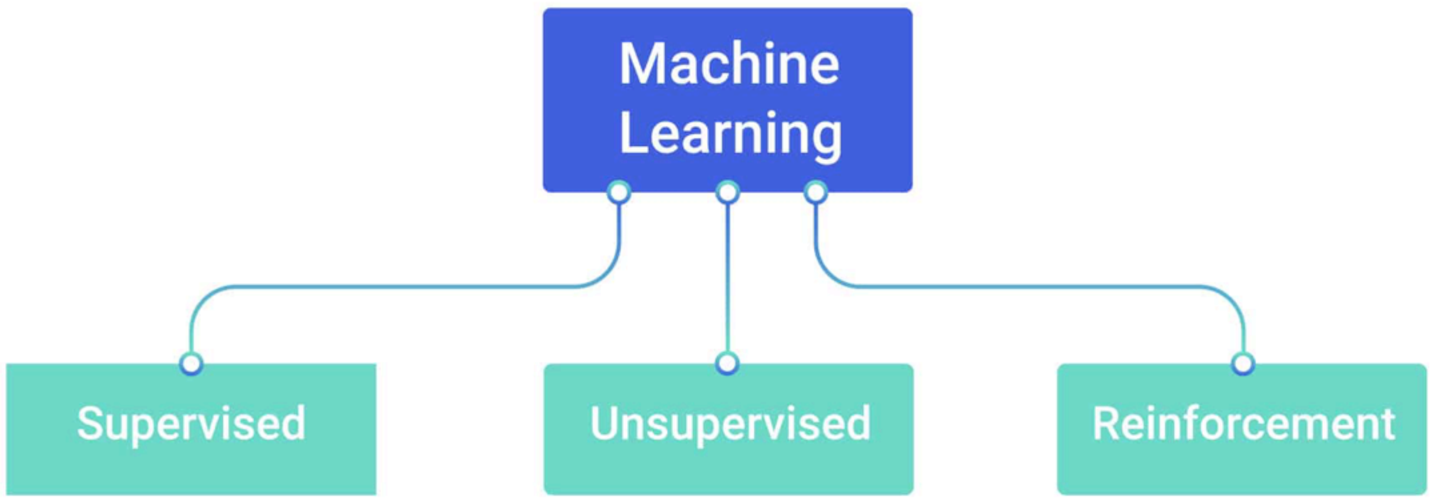
- Supervised, Unsupervised
- Semisupervised, Deep Learning, Recommender Systems

Reinforcement Learning



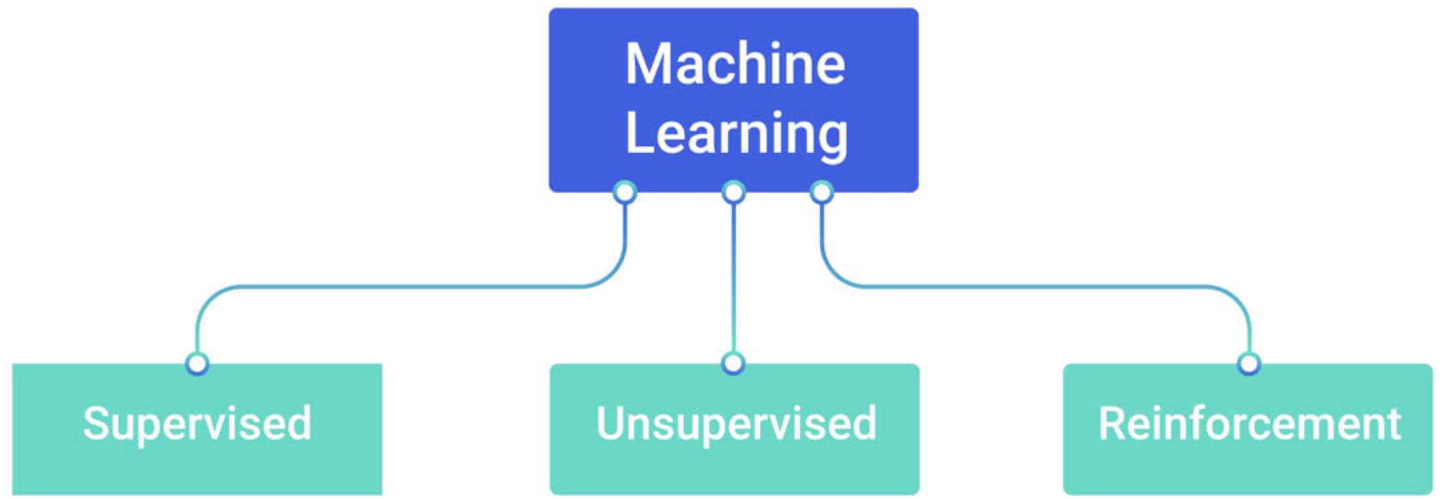
Technique	Learn from	Description
Reinforcement learning	trial & error	rewards from sequence of actions agent takes action to maximize reward

Supervised Learning



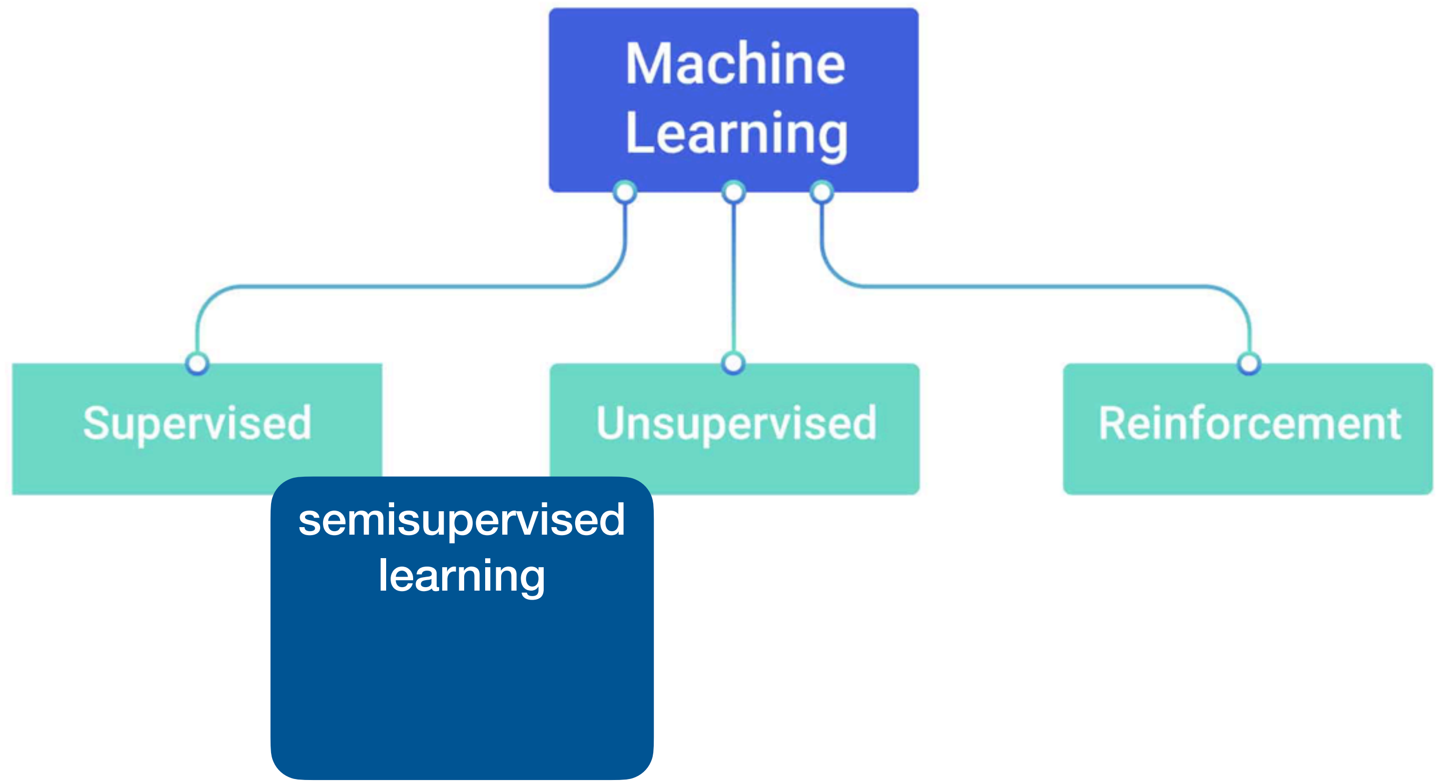
Technique	Learn from	Description
Supervised learning	examples	training data includes desired outputs e.g. classification, regression

Unsupervised Learning



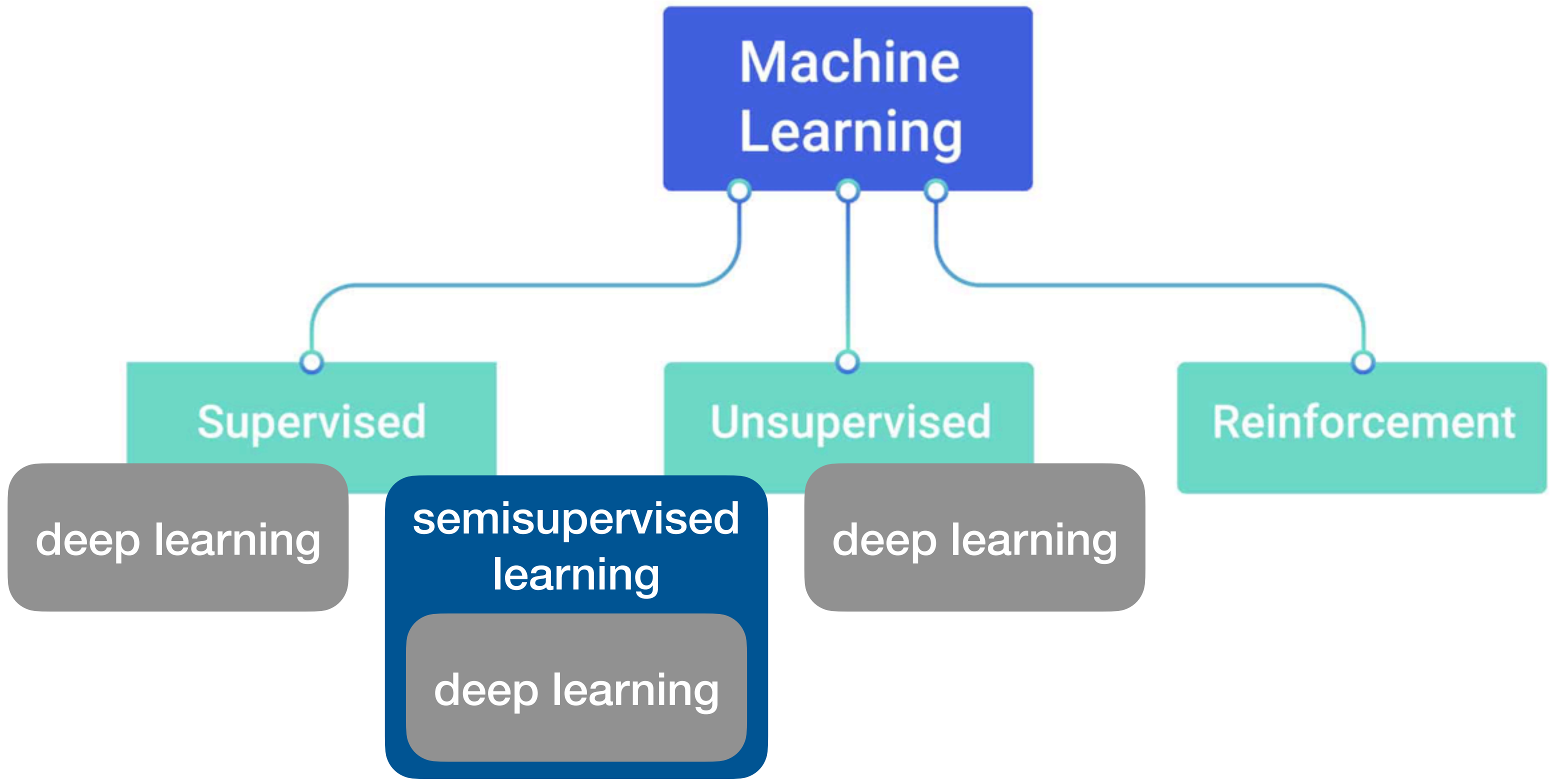
Technique	Learn from	Description
Unsupervised learning	observations	training data doesn't include desired outputs e.g. clustering, anomaly detection

Semisupervised Learning



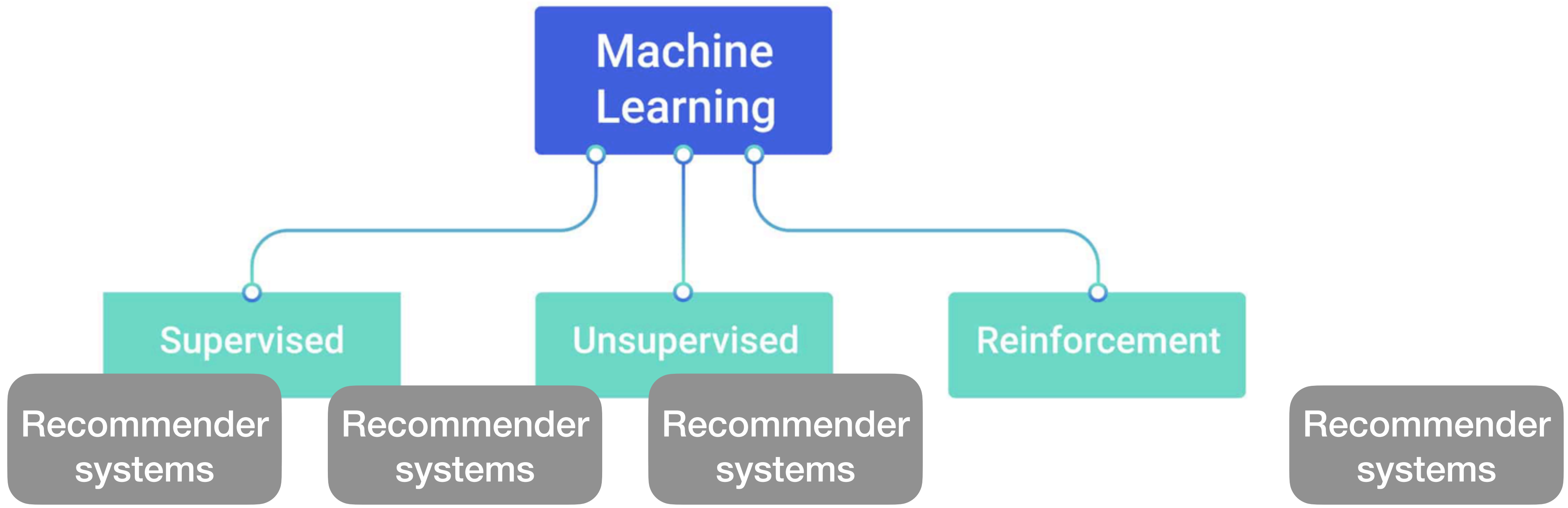
Technique	Learn from	Description
Semisupervised learning	observations & a few examples	hybrid supervised & unsupervised training data includes a few desired outputs

Deep Learning



Technique	Learn from	Description
Deep learning	observations &/or examples	can be supervised, unsupervised or semisupervised

Recommender Systems



Technique	Learn from	Description
Recommender systems	user info, item info, similar users	can be supervised, unsupervised, both or none

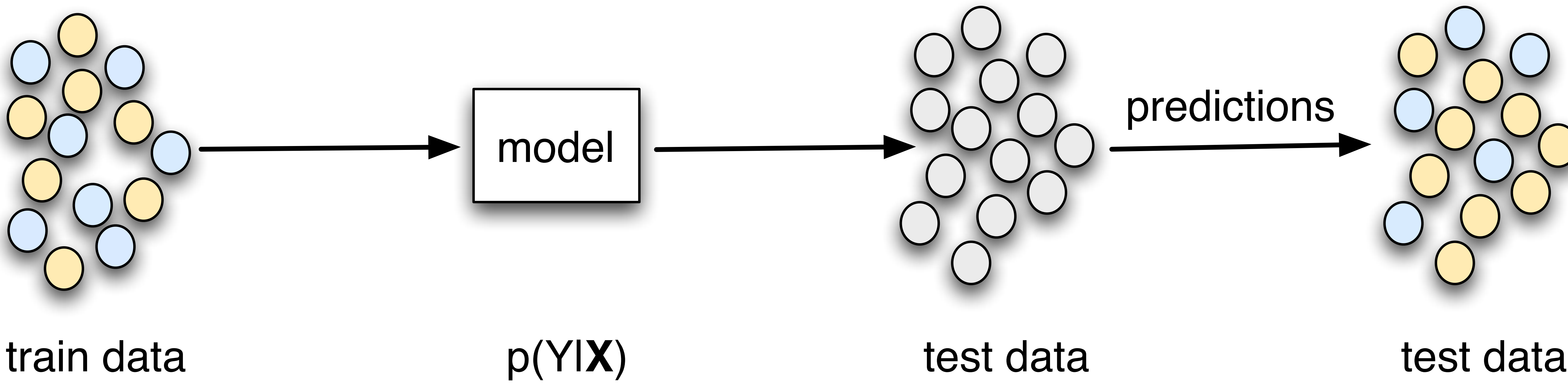
SUPERVISED LEARNING

Supervised Learning

In-Class Exercise:

- Supervised Learning techniques?
- Differences between techniques?

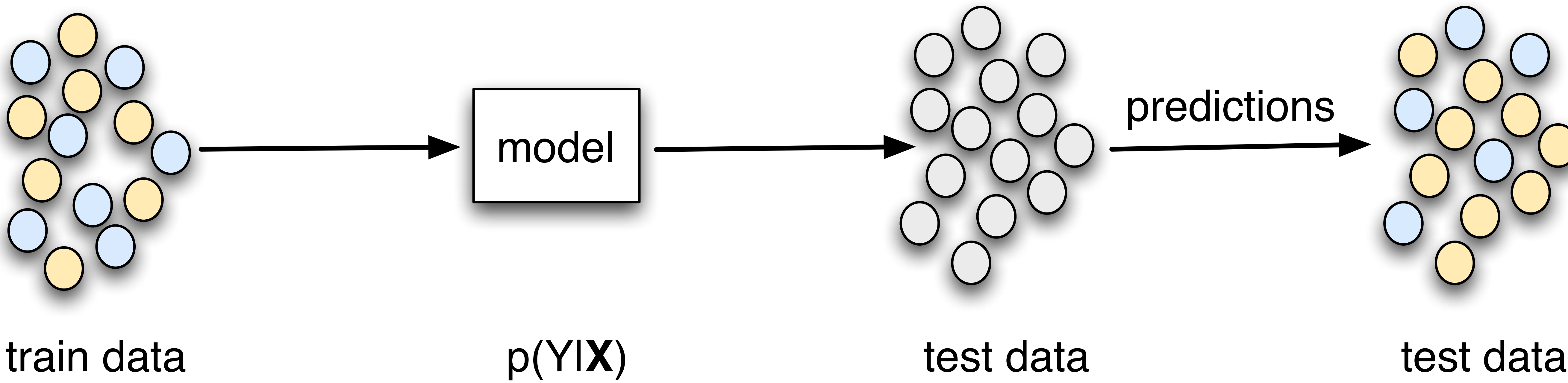
Supervised Learning



- **X**: input, Y: output
  We want to build a model that can
  accept observed instance **X** (as input) & generate label Y (as output)

- For a given dataset, we **know**
  — The correct output
  — There is a relationship between input & output
  — Goal: learn that relationship

Supervised Learning

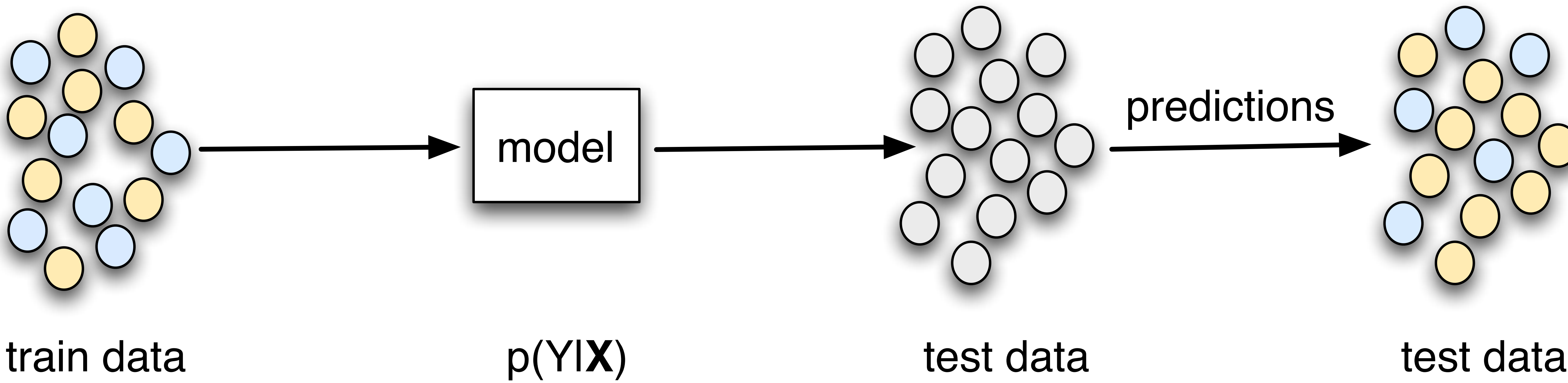


- **Classification**
  - Predict results in a **discrete** <u>output</u>
  - Map input variables into discrete categories

- **Regression**
  - Predict results within a **continuous** <u>output</u>
  - Map input variables to a continuous function

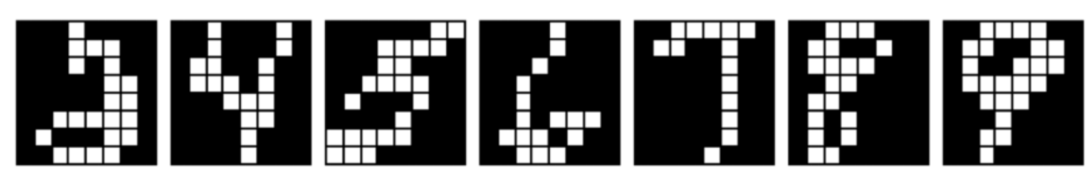
- What about the input (categorical/continuous)?

Supervised Learning



- The nature of <u>input</u> (categorical/continuous) does not matter!

- Example tasks
  - Digit recognition
  - Image classification
  - What else?



Supervised Learning

- **Inductive learning (for supervised learning)**
  – Given a **training set** of **examples** of the form (x, f(x))
    - x is the input, f(x) is the output

  – Return a function h that approximates f
    - h is called the **hypothesis**

- **Sample —> generalize to population**

CLASSIFICATION

Classification Example

- **Problem:** Will you enjoy an outdoor sport based on the weather?
- **Training set:**

Sky	Humidity	Wind	Water	Forecast	Enjoy Sport
Sunny	Normal	Strong	Warm	Same	Yes
Sunny	High	Strong	Warm	Same	Yes
Sunny	High	Strong	Warm	Change	No
Sunny	High	Strong	Cool	Change	Yes

$\underbrace{\text{Sky, Humidity, Wind, Water, Forecast}}_{x}$ $f(x)$

- **Possible Hypotheses:**
  - h_1: S=sunny —> enjoySport=yes
  - h_2: Wa=cool or F=same —> enjoySport=yes

Classification

Input / Output	Multiple Features—Multiple Types
Two classes	Binary Classification
Multiple classes	Multi-class Classification

Classification Examples

In-Class Exercise:

- Examples of classification?

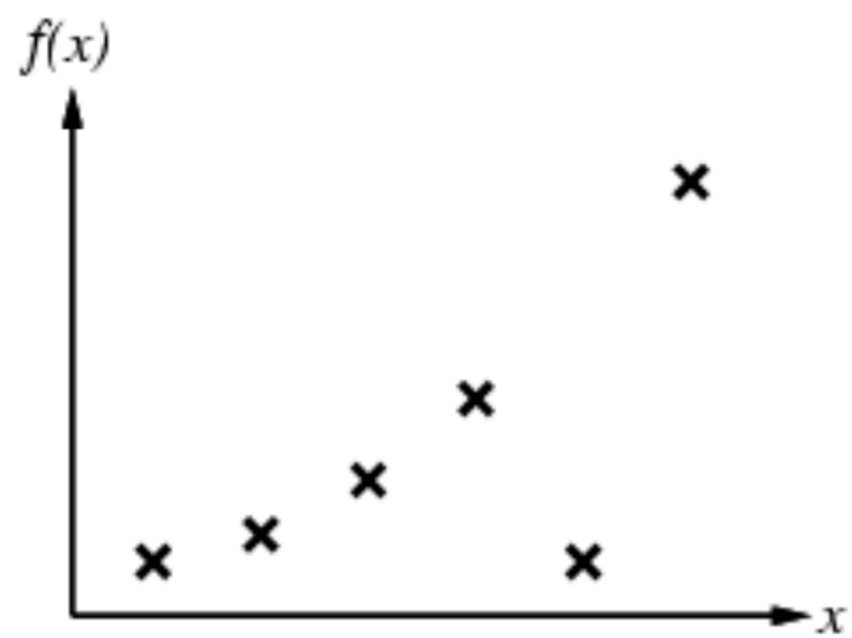
Classification examples

- Cancer detection
- Image classification
- Spotting eye disease
- Face recognition
- Machine translation
- Speech recognition
- Predicting aftershock patterns

REGRESSION

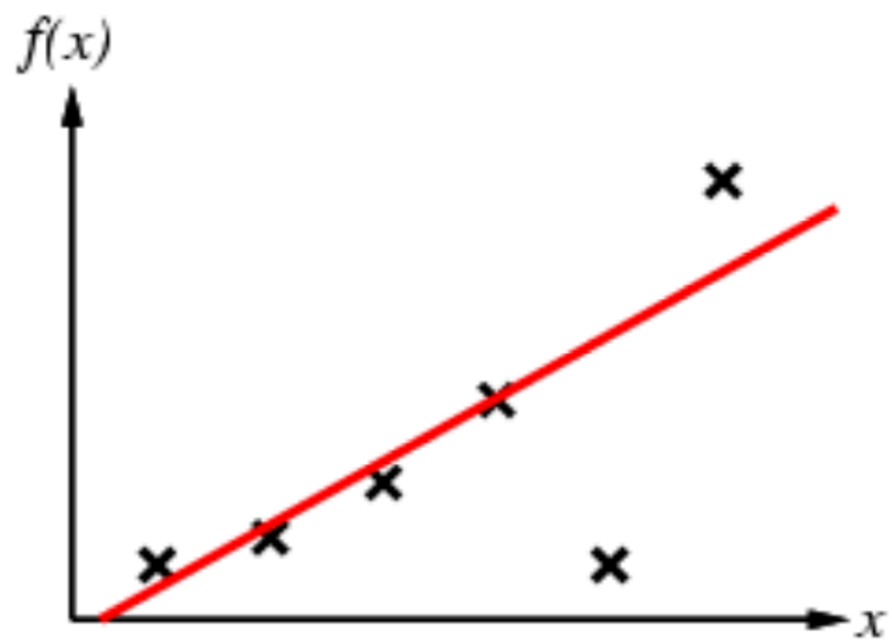
Regression

In-Class Exercise: Find function **h** that fits **f** at instances **x**



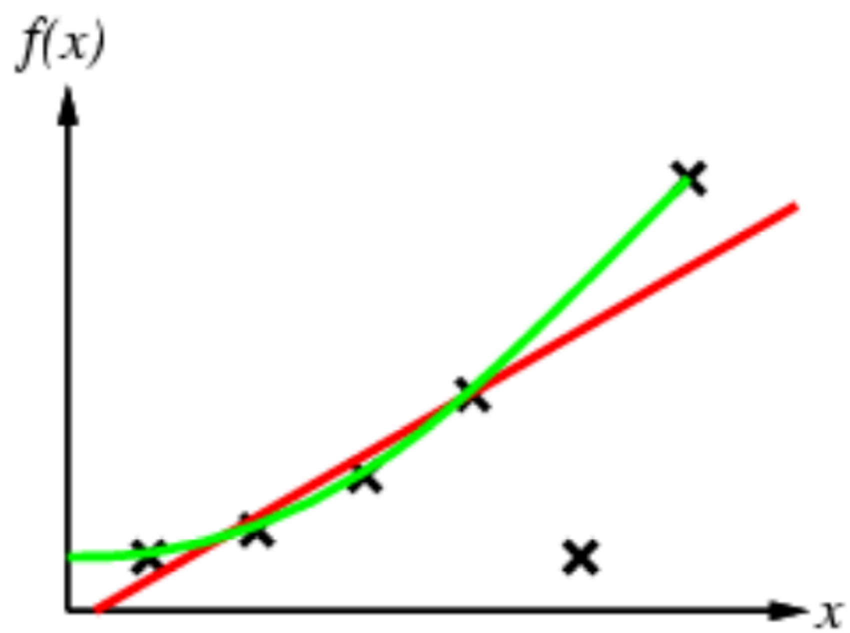
Regression

- Find function **h** that fits **f** at instances **x**



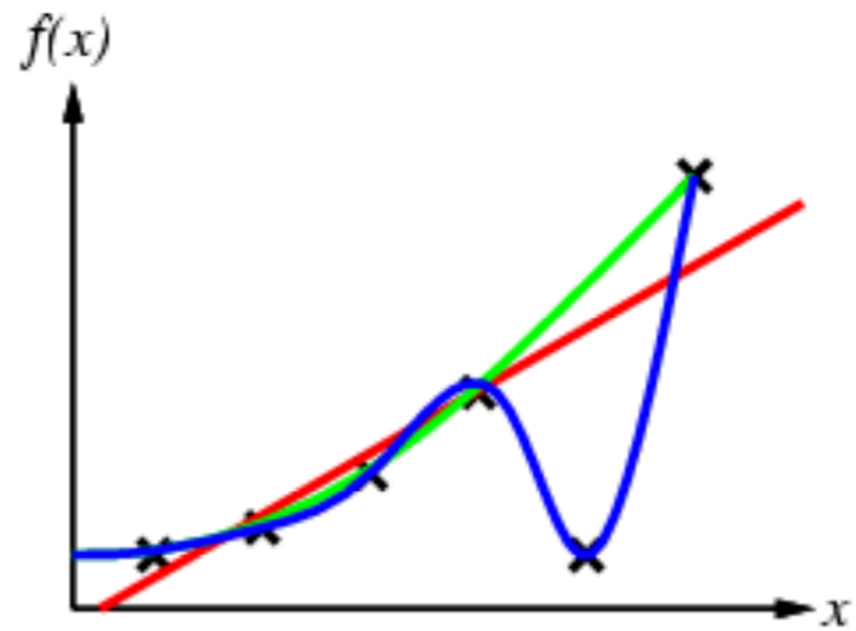
Regression

- Find function **h** that fits **f** at instances **x**



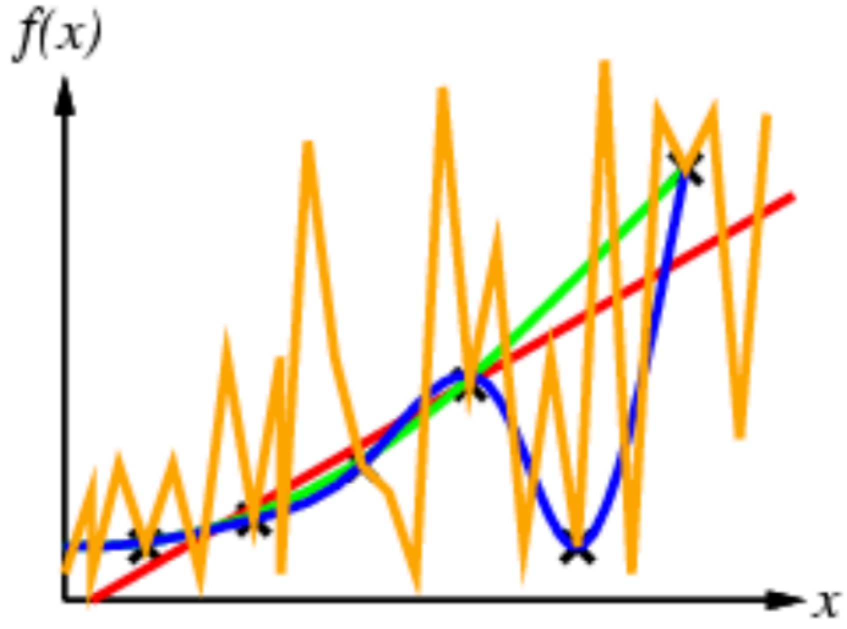
Regression

- Find function **h** that fits **f** at instances **x**



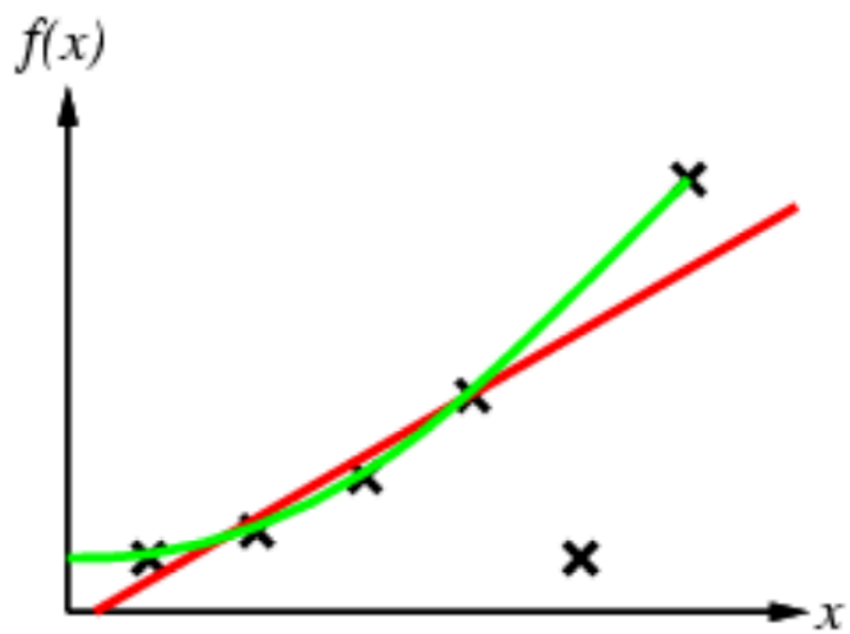
Regression

- Find function **h** that fits **f** at instances **x**



Regression

- Find function **h** that fits **f** at instances **x**



Generalization

- Key: a good hypothesis will **generalize well** (i.e. predict unseen examples correctly)
- Ockham's razor: prefer the simplest hypothesis consistent with the data

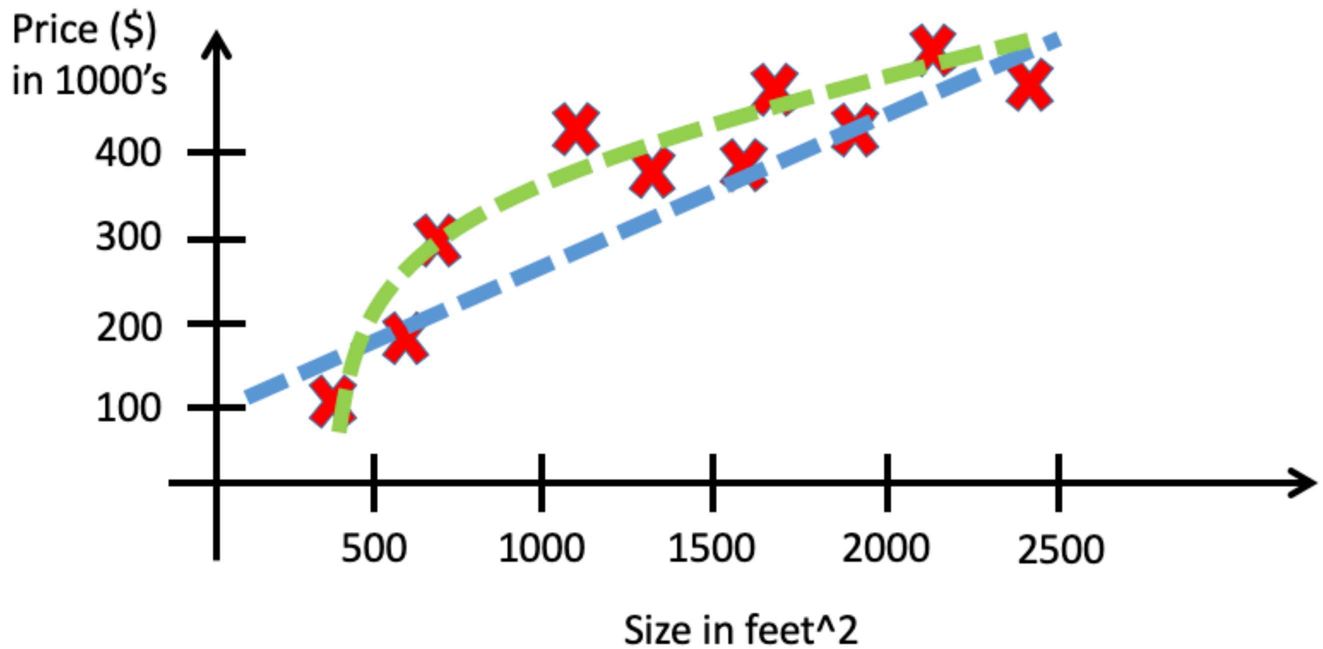
Regression Examples

In-Class Exercise:

- Examples of regression?

Regression examples

- **Housing price prediction**
- Stock market prediction
- Weather prediction
- Human pose estimation
- Facial landmark alignment (snapchat)



Classification/Regression examples in-class exercise

Problem	Task formulation	Domain	Range	Classification/ Regression
Housing valuation 1	predict house price given house size			
Housing valuation 2	predict whether the house "sells for more or less than the asking price" given house size			
Age prediction	predict person age based on a given picture			
Age prediction	predict person age category (high school, college, graduate) based on a given picture			
Loan approval prediction	approve or reject loan based on credit score			
Spam detection	classify email into spam / non-spam			
Stock price prediction	predict whether stock will be buy sell or hold			
Speech recognition	convert audio file into text			
Digit recognition	digit recognition			

Classification/Regression examples in-class exercise

Problem	Task formulation	Domain	Range	Classification/ Regression
Housing valuation 1	predict house price given house size	house size (cont)	price (cont)	Regression
Housing valuation 2	predict whether the house "sells for more or less than the asking price" given house size	house size (cont)	more/ less	Binary classification
Age prediction	predict person age based on a given picture	facial features	age range	Multi-class classification
Age prediction	predict person age category (high school, college, graduate) based on a given picture	image features	age category	Multi-class classification
Loan approval prediction	approve or reject loan based on credit score	credit score	approve / reject	Binary classification
Spam detection	classify email into spam / non-spam	keywords	spam/ non-spam	Binary classification
Stock price prediction	predict whether stock will be buy sell or hold	Stock values	buy, sell, hold	(time-series) Classification
Speech recognition	convert audio file into text	audio signals	words	Multi-class classification
Digit recognition	digit recognition	images of digits	0 to 9	Multi-class classification

UNSUPERVISED LEARNING

Unsupervised Learning

- Little/no idea what results should look like
  — Output is not given as part of training set
- Find model that explains the data
  — Derive structure from data
- There is no feedback based on the prediction results
  — There is no teacher to correct you

Unsupervised Learning

In-Class Exercise:

- Example techniques?

Unsupervised Learning Examples

- Derive structure by **clustering** data based on relationships among variables in data
- Generate a **compressed representation** of data
- Find **structure** in messy data

Unsupervised Learning Examples

- **Clustering**
  - Take a collection of 1000 essays on US Economy,
  - Find a way to automatically group them into a small number that are similar or related by different *variables*
  - Such as word frequency, sentence length, page count, ..

- **Dimensionality reduction**
  - “Cocktail Party Algorithm"
  - Find structure in messy data
  - Identify individual voices and music from a mesh of sounds at a cocktail party

Clustering Examples

In-Class Exercise:

- Examples of clustering?

Clustering examples

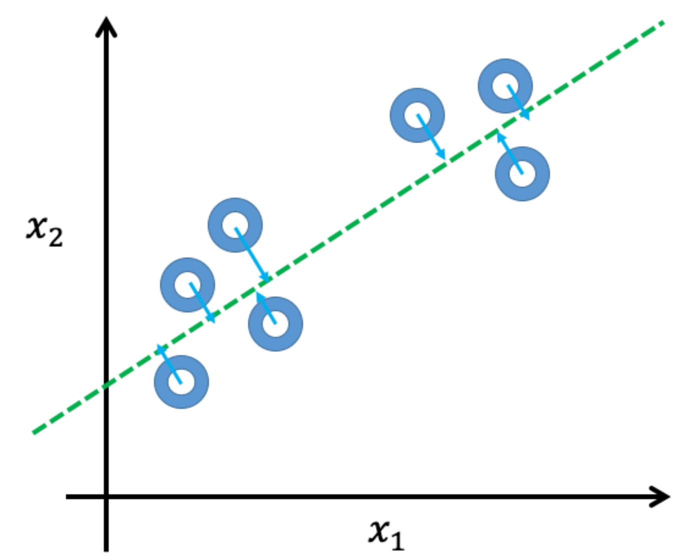
- Google news feed
- Clustering DNA microarray data
- Social network analysis
- Market segmentation
- Building energy consumption segmentation
- Astronomical data analysis

Dimensionality reduction

- Some problems involve high-dimensional data
- 3D face modeling
- Shape modeling
- Cocktail party problem
  —Use visual and auditory signals in a video to
  —Separate speech of a single speaker

Dimensionality reduction

- Working in high-dimensional spaces can be undesirable
High-dimensional data can be:
—Sparse
—Computationally intractable to analyze



- **Dimensionality reduction**
—Transform data from high-dim space into low-dim space
—So that low-dim representation retains some meaningful properties of original data

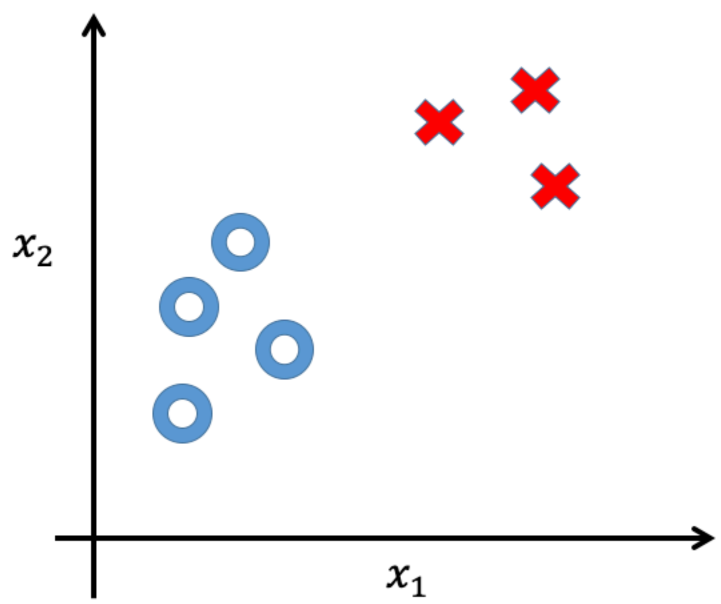
SUPERVISED VS. UNSUPERVISED LEARNING RECAP

Learning algorithms

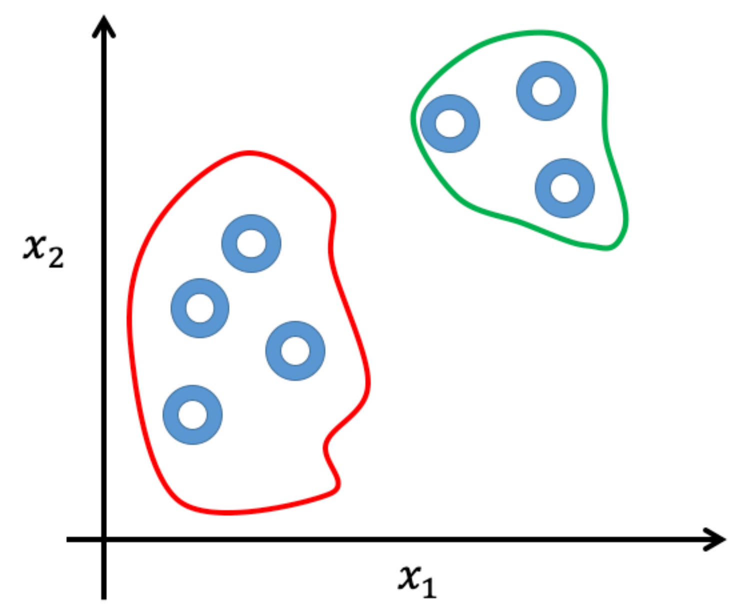
Algorithm / Output	Supervised Learning	Unsupervised Learning
Discrete	Classification	Clustering
Continuous	Regression	Dimensionality reduction

Supervised vs. unsupervised

Supervised Learning



Unsupervised Learning



Reinforcement Learning

Reinforcement Learning

- What is Reinforcement Learning?

Reinforcement Learning

- ML draws on analogies from **human learning** to create algorithms and paradigms

- RL draws on the process of **trial and error**
  the algorithm <u>tries an action</u>,
  <u>gets a reward</u>, and
  <u>updates its decision process</u> accordingly

RECALL
Intelligence vs. Learning Connection

Intelligence & Learning

Human Intelligence

Human perception, reasoning, .., learning

Learn from examples
—> supervised learning

Learn to find structure and patterns
—> unsupervised learning

Animal Psychology

Reinforcements are used to train animals

Negative reinforcements - pain/hunger
Positive reinforcements - pleasure/food

Let's do the same with computers!
Rewards: numerically indicate how good actions are
E.g., game win/loss, money, time, etc.
—> Reinforcement Learning

Reinforcement Learning

- **Agent** uses **decision process** to choose an **action** based on **observed state** of its **environment**

- **Action** updates **environment**, and is assigned some **reward**

- A positive reward encourages that action, a negative reward discourages it

- **Goal —** Learn to choose **actions** that maximize **rewards**

Tradeoff in RL

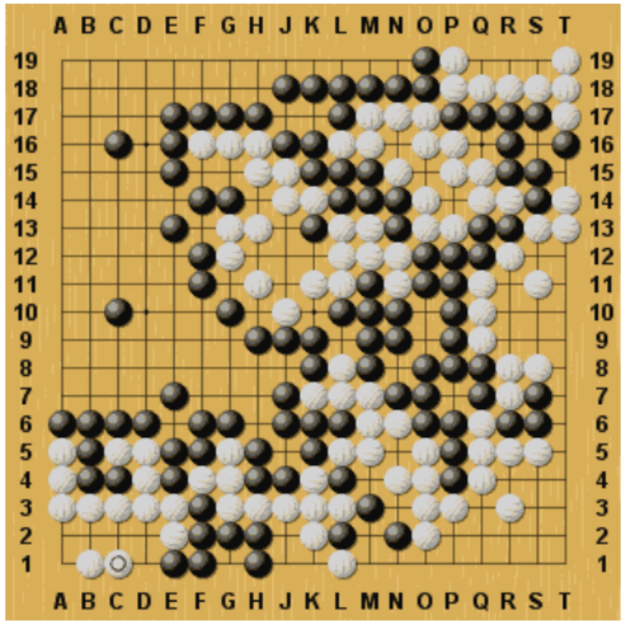
- An important tradeoff in Reinforcement Learning is between
  Exploration (trying new things) and
  Exploitation (using known good strategies)

Applications of RL

- **Robotic motion planning**

- **Game play**
  AlphaGo beat a professional Go player,
  OpenAI beat a professional DOTA player
  both with RL

- Go (one of the oldest and hardest board games)
  Agent: player
  Environment: opponent
  State: board configuration
  Action: next stone location
  Reward: +1 win / -1 loose



- 2016: AlphaGo defeats top player Lee Sedol (4-1)
  Game 2 move 37: AlphaGo plays unexpected move (odds 1/10,000)

Project Learning Paradigm

In-Class Exercise: Identify a potential project topic

- What is the learning **task**?
- What **data** can be used?
- What is a possible **learning paradigm**?

LECTURE SUMMARY

Lecture Objectives Recap

- Learning Formulation
- Learning paradigms
- Connections & differences
- When to use each approach
- Apply what we learn to brainstorm about course project

Next Lecture

- **Learning strategies**
  - **Theoretical** learning strategies (learning theory)
    - When can we learn?
    - How well can we learn?
  - **Practical** learning strategies (practical considerations)
    - What is needed? What is possible?
    - Ethical considerations