Lecture 1: Introduction to Reinforcement Learning

References:

1. Kevin Murphy, “Machine Learning A Probabilistic Perspective”, MIT Press, 2012
2. Prof. David Silver’s Couse, University College, London, Link: http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html
3. Prof. Sergey Levine’s Couse, UC Berkley, Link: http://rail.eecs.berkeley.edu/deeprlcourse-fa17/
4. Prof. Ben Recht’s notes on RL

What is Machine Learning?

What is Machine Learning?

Wikipedia: *“Machine learning is a field of computer science that uses statistical techniques to give computer systems the ability to "learn" (e.g., progressively improve performance on a specific task) with data, without being explicitly programmed”*

Murphy [2012]: *“We define machine learning as a set of methods that can automatically detect patterns in data, and then use the uncovered patterns to predict future data, or to perform other kinds of decision making under uncertainty”*

Three Main Classes of ML

1. Unsupervised Learning
2. Supervised Learning
3. Reinforcement Learning

	from input ***x***, output:
unsupervised	summary ***z***
supervised	prediction ***y***
reinforcement	action ***a*** to maximize reward ***r***

Descriptive. → Predictive → Prescriptive

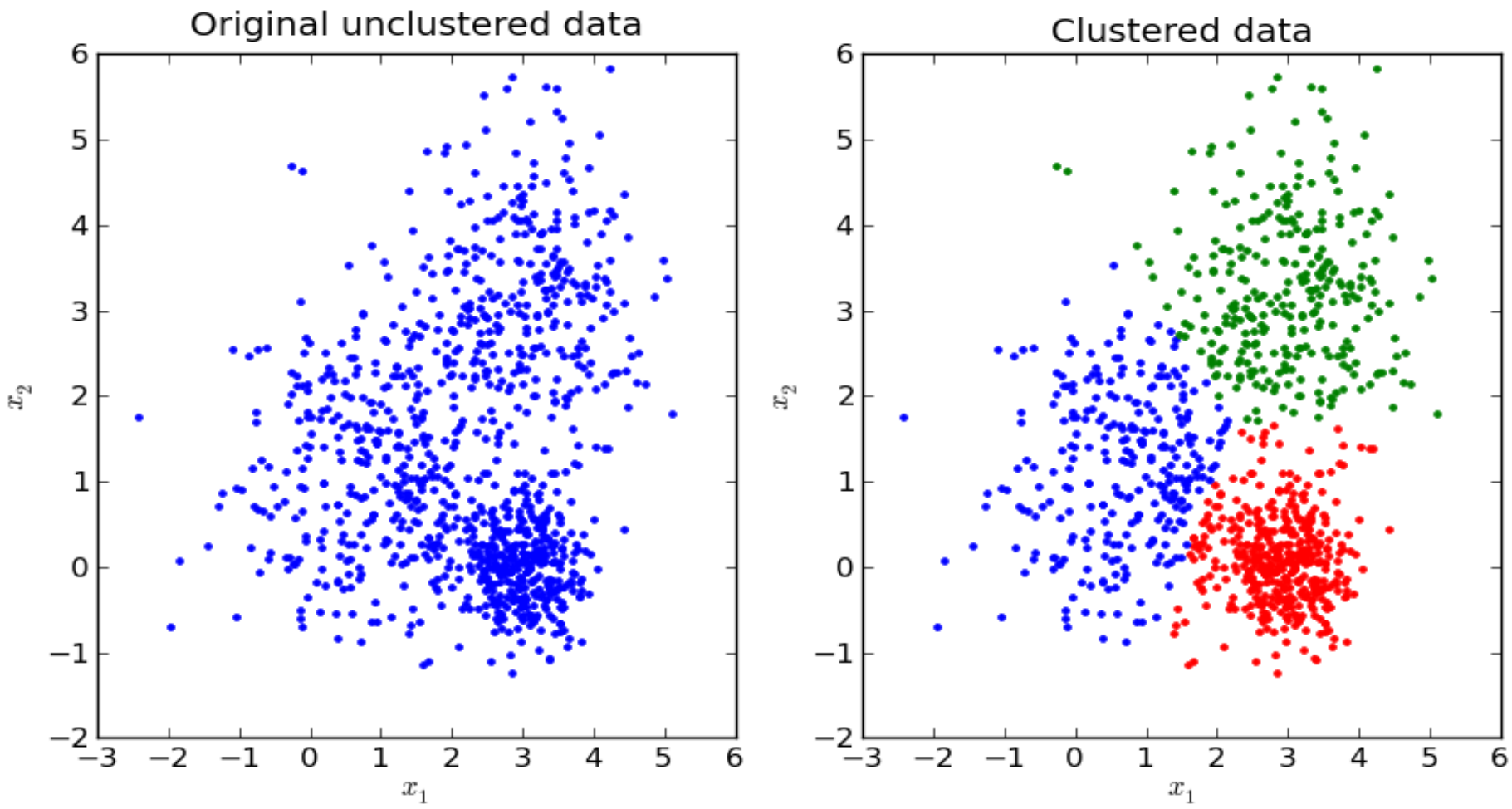
Unsupervised Learning

Unsupervised Learning

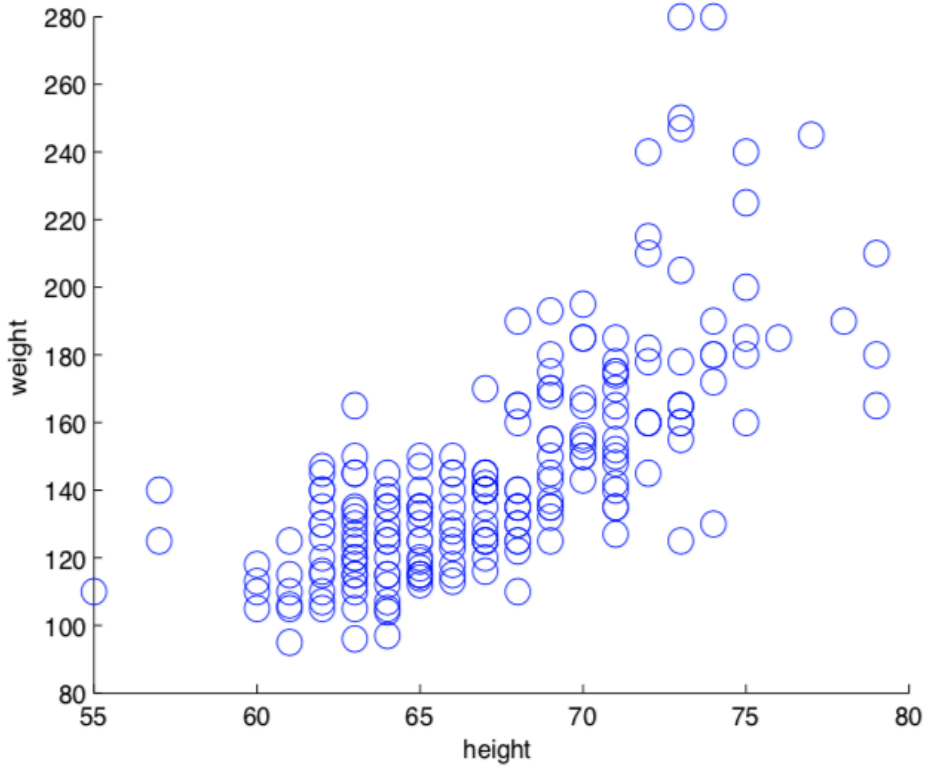
Wikipedia: *“Unsupervised machine learning is the machine learning task of inferring a function that describes the structure of "unlabeled" data (i.e. data that has not been classified or categorized)”*

Examples: Clustering, Dimensionality Reduction, Matrix Completion, Image Inpainting, Collaborative Filtering

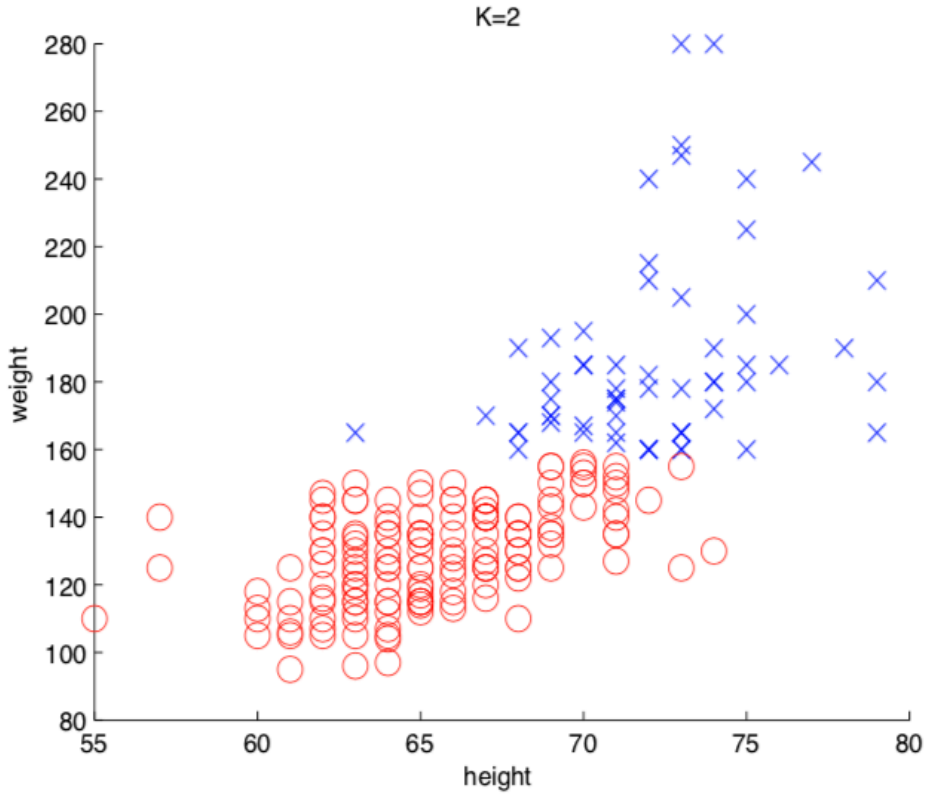
Unsupervised Learning: Clustering



Unsupervised Learning: Clustering

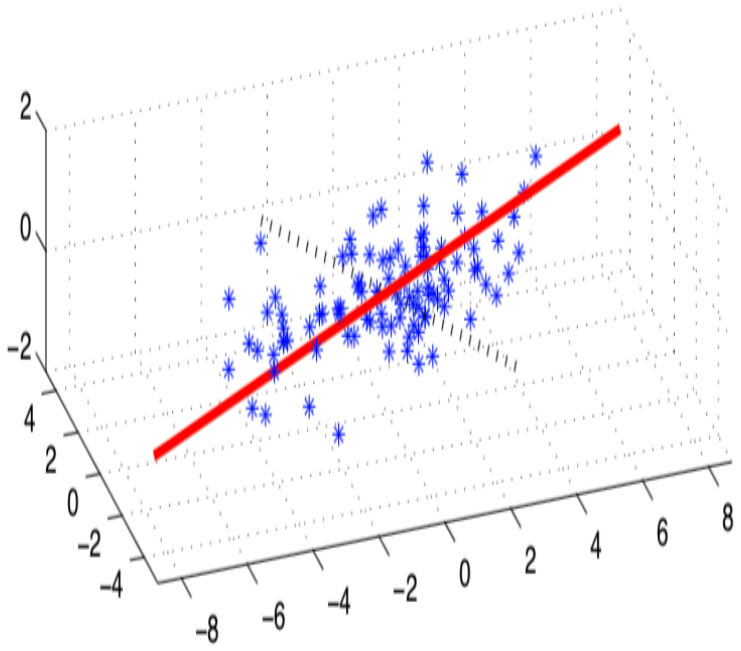


(a)

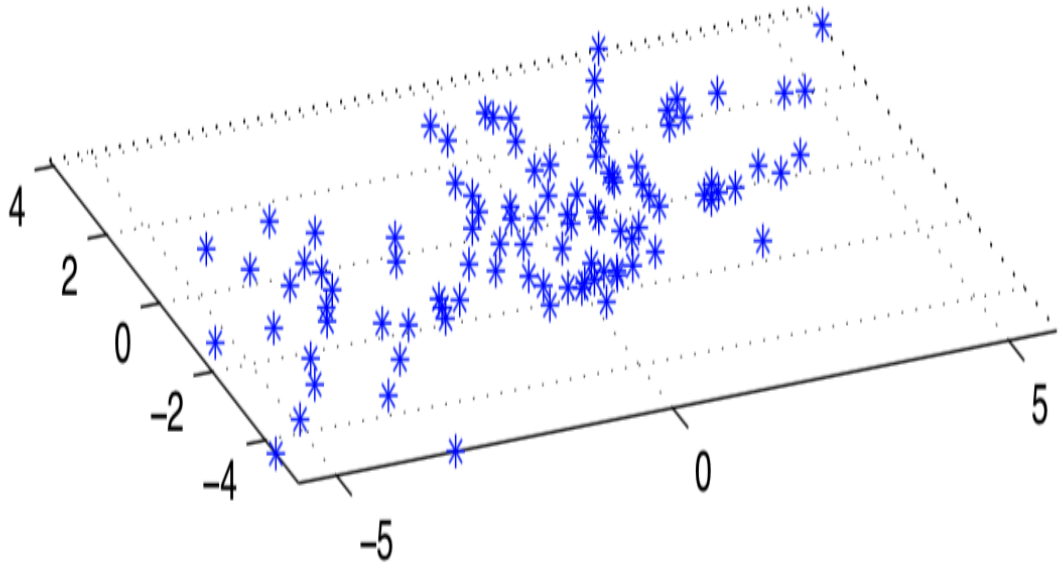


(b)

Unsupervised Learning: PCA



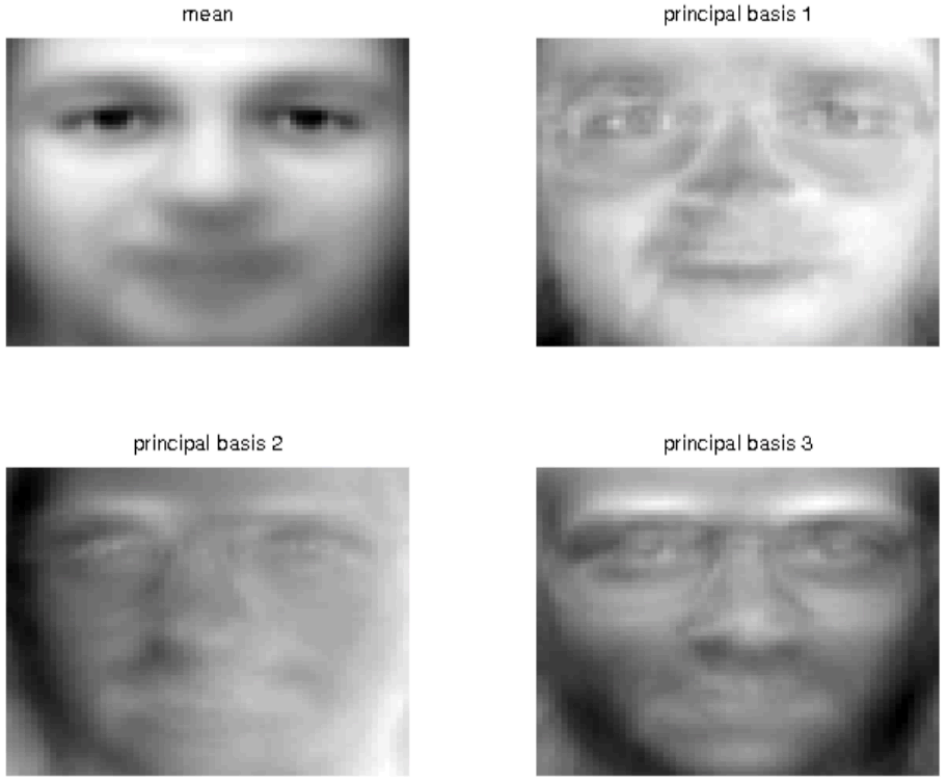
(a)



(b)

Unsupervised Learning: PCA

(a)



(b)

Unsupervised Learning

Descriptive analytics refers to summarizing data in a way to make it more interpretable

Unsupervised learning is a form of descriptive analytics

Goal is to create a shorter list of attributes ***z*** for each example that somehow summarizes the salient information in ***x***.

	from input *x*, output:
unsupervised	summary ***z***
supervised	prediction ***y***
reinforcement	action ***a*** to maximize reward ***r***

Supervised Learning

Supervised Learning

Wikipedia: *“Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs”*

Examples: Image classification, Handwriting recognition, Email spam filtering, Face recognition, Speech recognition

Supervised Learning: Image Classification

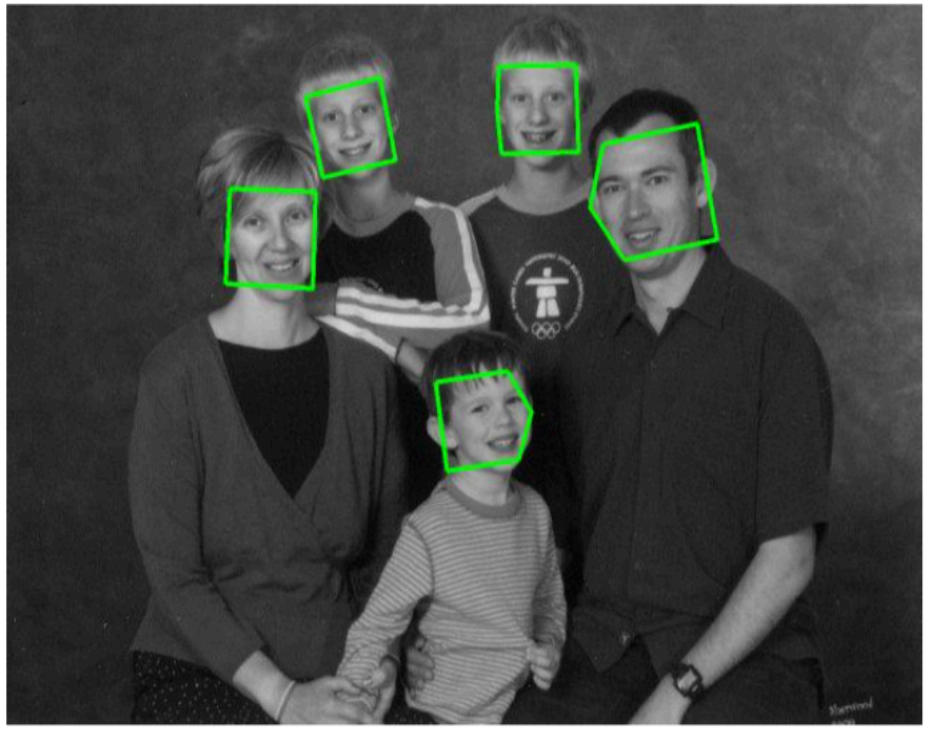
Training Data

Testing Data

Cat

Supervised Learning: Face Recognition

(a)

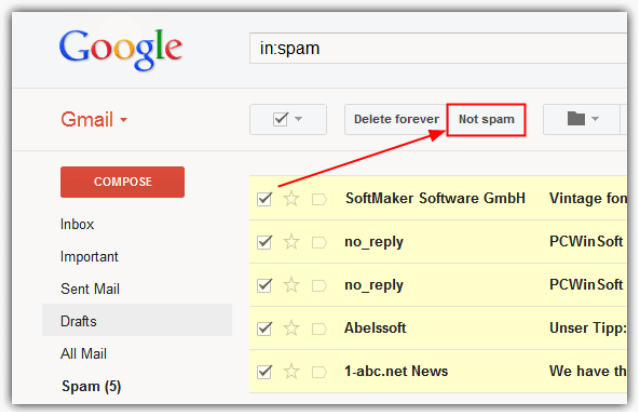


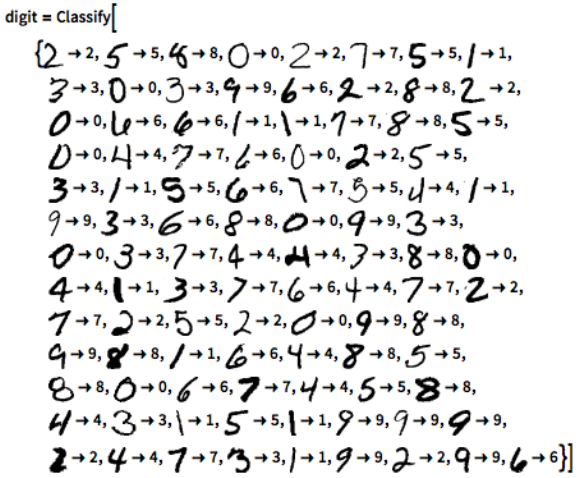
(b)

Supervised Learning

Wikipedia: *“Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs”*

Examples: Image classification, Handwriting recognition, Email spam filtering, Face recognition, Speech recognition, …





Supervised Learning

Predictive analytics aims to estimate outcomes from current data.

Supervised learning is a kind of predictive analytics.

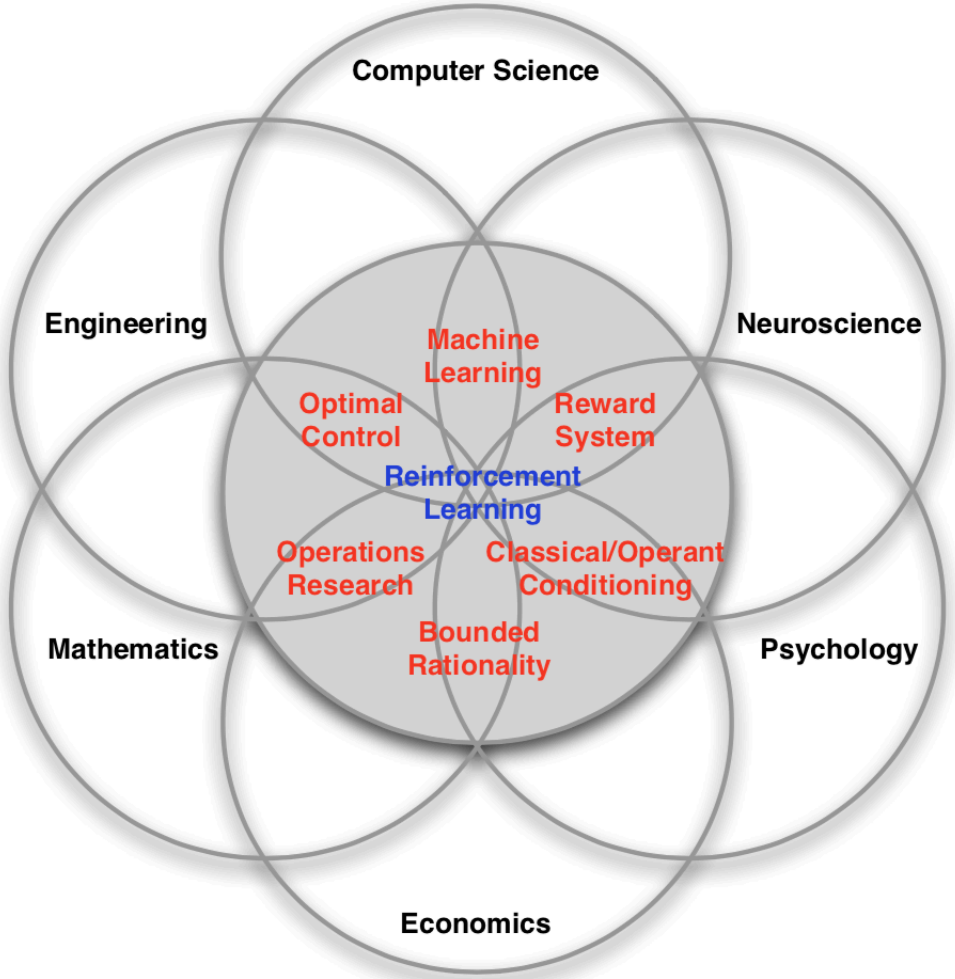
Goal is to predict ***y*** from ***x*** such that on new data you are accurately predicting ***y***

	from input ***x***, output:
unsupervised	summary ***z***
supervised	prediction ***y***
reinforcement	action ***a*** to maximize reward ***r***

Reinforcement Learning

Reinforcement Learning

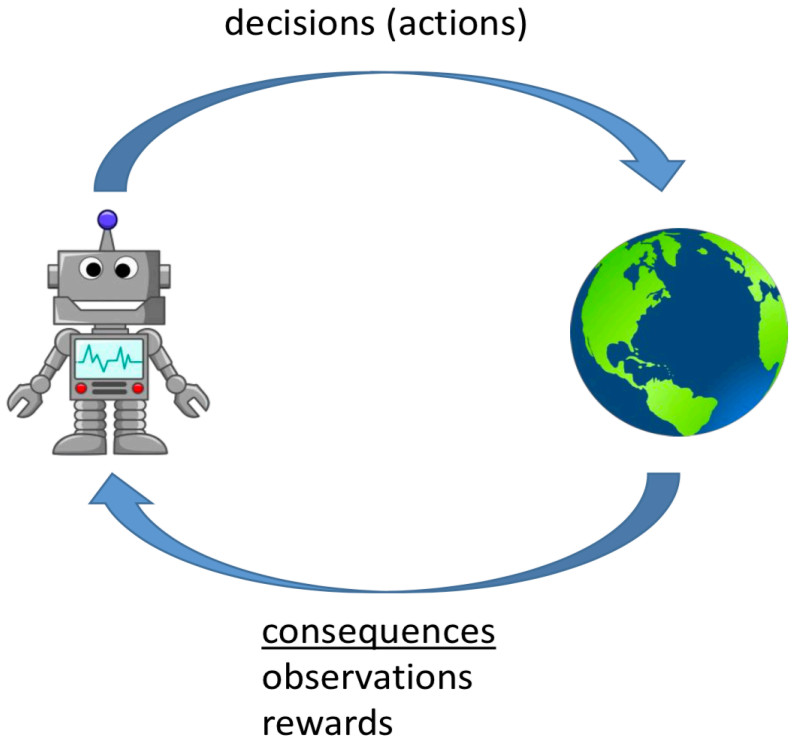
Wikipedia: *"Reinforcement learning (RL) is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward. The problem, due to its generality, is studied in many other disciplines, such as game theory, control theory, operations research, information theory, optimization, multi-agent systems, swarm intelligence, statistics, …"*



Why RL is Different

Learning to make a good sequence of decisions under uncertainty

- No supervisor, only a reward signal
- Feedback is delayed, not instantaneous
- Sequential decision making
- Actions have long-term consequences
- Non i.i.d. data
- Agent's actions affect the subsequent data it receives



RL Successes/Applications

- Fly stunt maneuvers in a helicopter
- Defeat the world champion at Backgammon
- Manage an investment portfolio
- Control a microgrid
- Make a humanoid robot walk
- Play many different Atari games better than humans
- Play Go games better than humans

Helicopter Maneuvers

Pieter Abbeel, Adam Coates, Morgan Quigley, and Andrew Y. Ng. "An application of reinforcement learning to aerobatic helicopter flight." In *Advances in neural information processing systems*, 2007.

Playing Atari

V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, et al. “Playing Atari with Deep Reinforcement Learning”. In *Advances in neural information processing systems,* 2013.

Robotic Arms

Sergey Levine, Peter Pastor, Alex Krizhevsky, Julian Ibarz, Deirdre Quillen, “Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection”, *International Journal of Robotics Research*, 2017

Playing Go

- D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, et al. “Mastering the game of Go with deep neural networks and tree search”. *Nature* (2016).

Reinforcement Learning

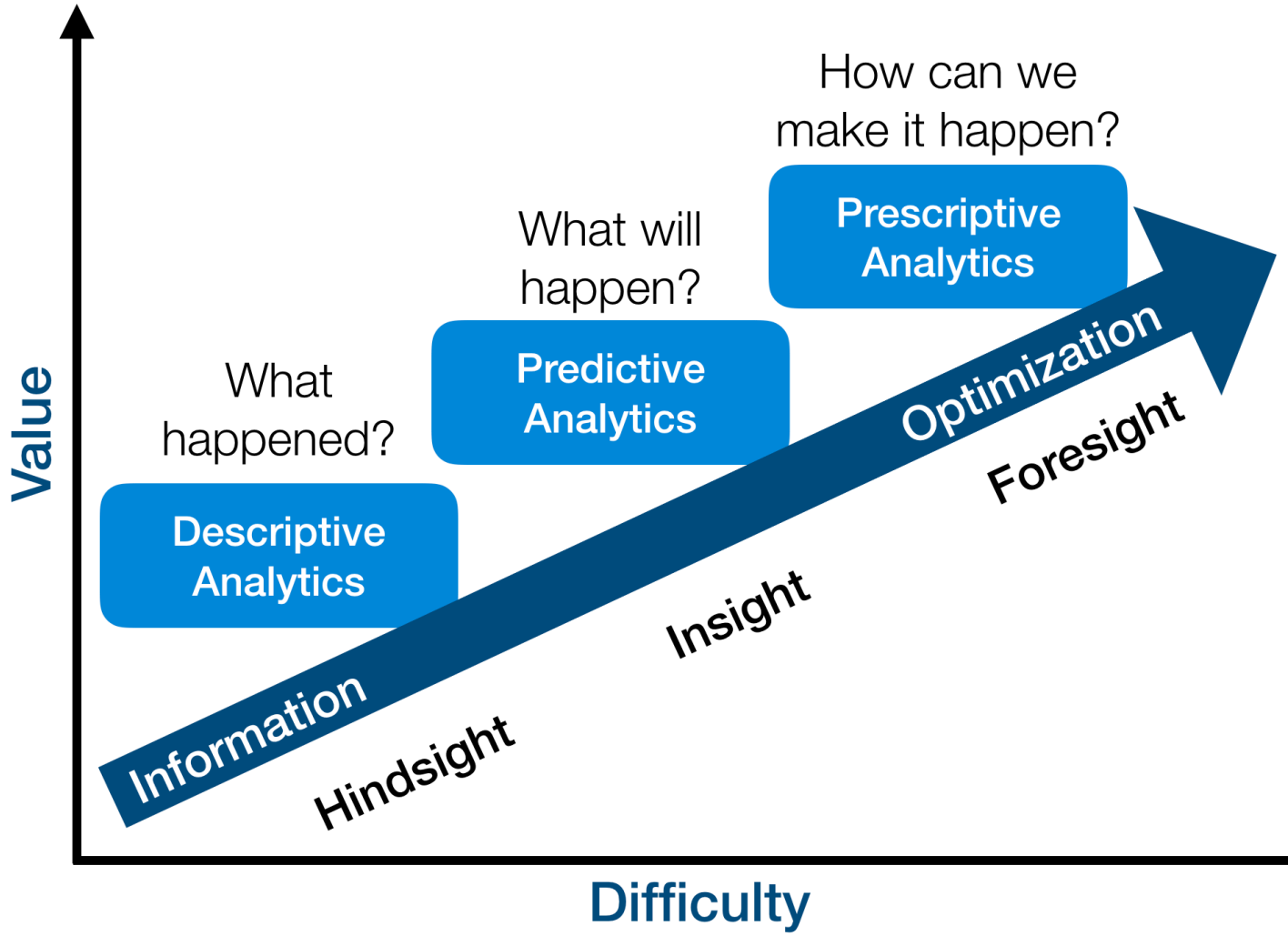
Prescriptive analytics guides actions to take in order to guarantee outcomes

Reinforcement learning is a kind of prescriptive analytics.

Goal is to analyze ***x*** and then subsequently choose ***a*** so that ***r*** is large.

	from input ***x***, output:
unsupervised	summary ***z***
supervised	prediction ***y***
reinforcement	action ***a*** to maximize reward ***r***

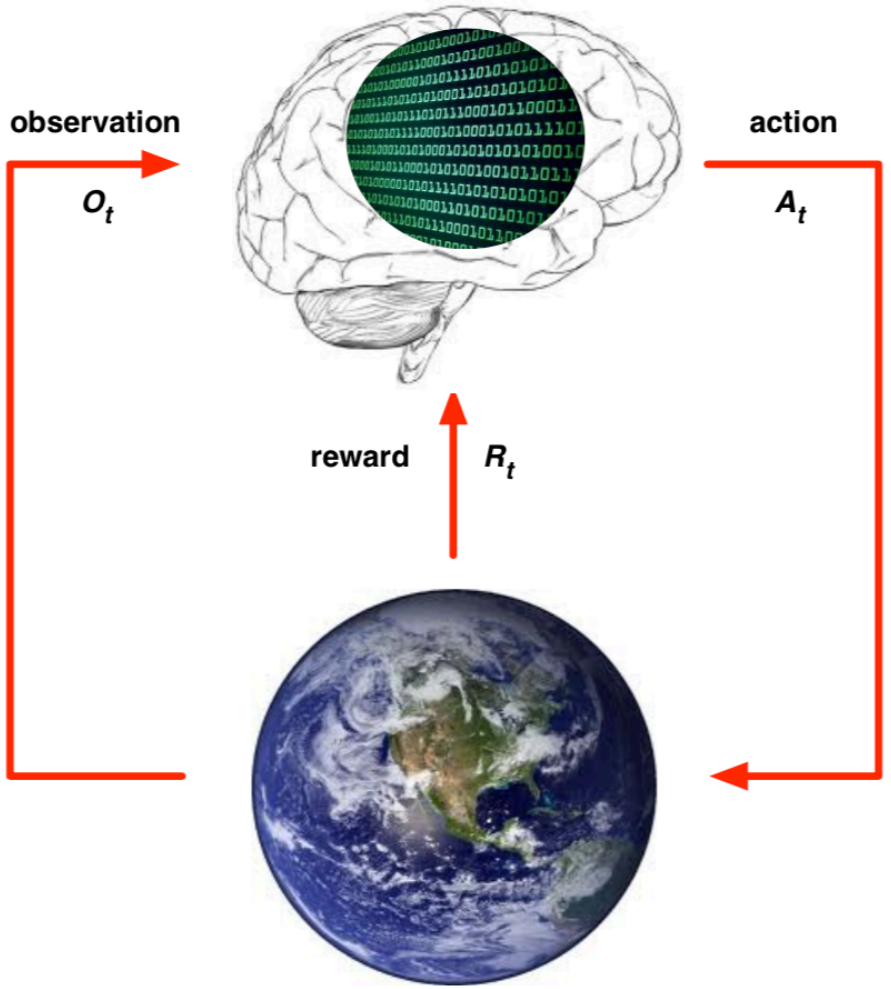
Machine Learning: Value and Difficulty



More on RL

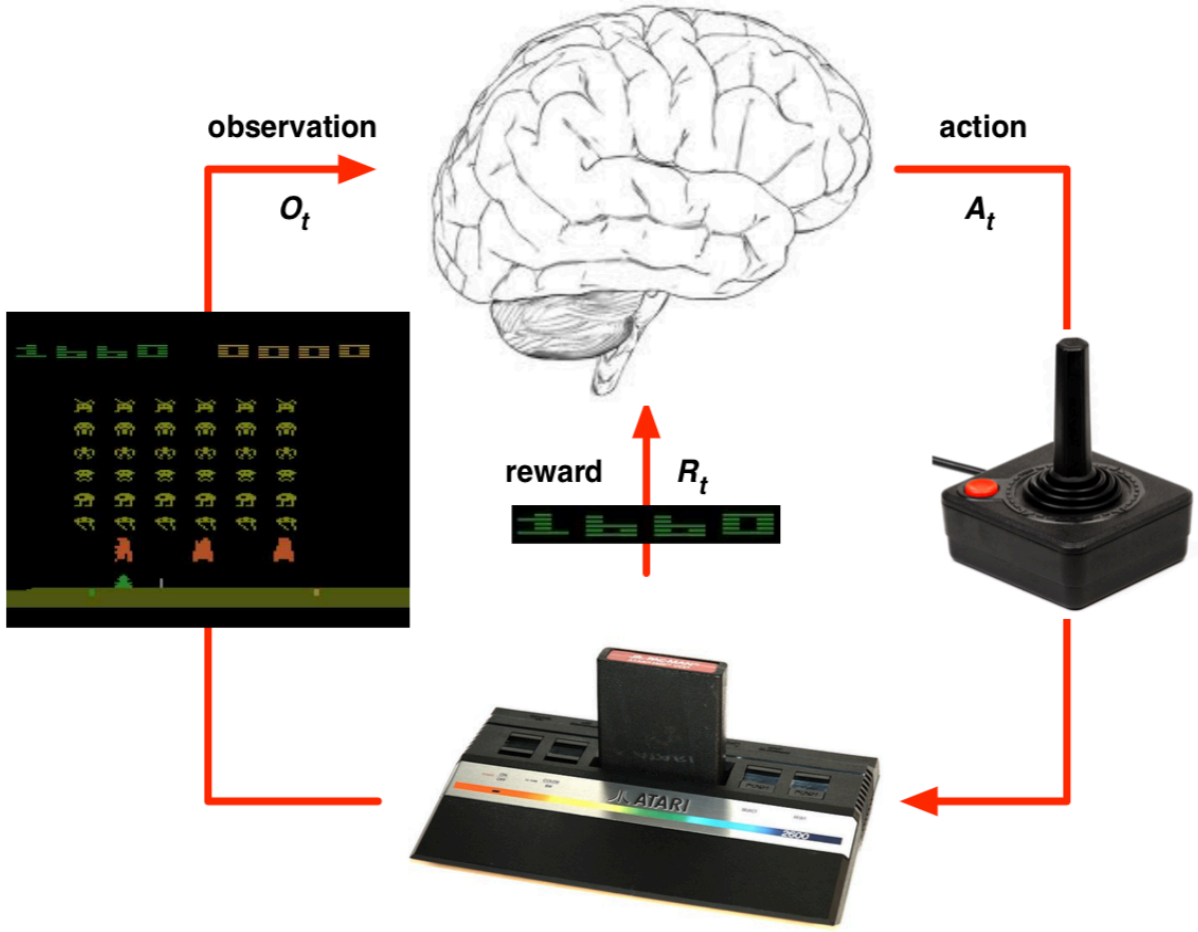
RL: Agent and Environment

- At each time step *t* the agent:
  - Executes an action A_t
  - Receives reward R_t
  - Receives observation O_{t+1}
- The environment:
  - Receives an action A_t
  - Emits reward R_t
  - Emits reward O_{t+1}
- Time *t* <-- *t+1*



RL: Agent and Environment

- At each time step *t* the agent:
  - Executes an action A_t
  - Receives reward R_t
  - Receives observation O_{t+1}
- The environment:
  - Receives an action A_t
  - Emits reward R_t
  - Emits reward O_{t+1}
- Time *t <-- t+1*



Rules of the game are unknown!

RL: Reward

- A reward is a scalar feedback signal
- Indicates how well agent is doing at step *t*
- The agent's job is to maximize the expected cumulative reward

Reward Hypothesis: All goals can be described by the maximization of expected cumulative reward

Reward Example

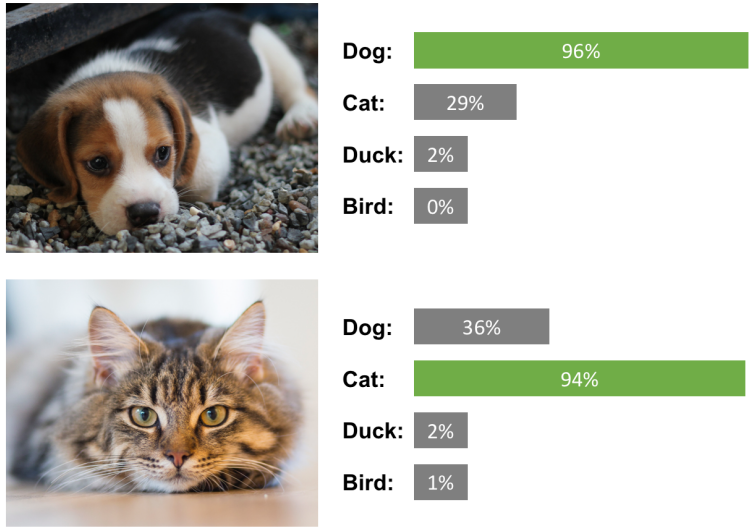
- Helicopter maneuvers
  - +ve reward for following desired trajectory
  - -ve reward for crashing
- Robotic Arm
  - +ve reward for correct grasping
  - -ve reward for dropping
- Atari games
  - +/- ve reward for increasing/deceasing score
- Controlling a (power) plant
  - -ve reward for exceeding the safety threshold

RL: Sequential Decision Making

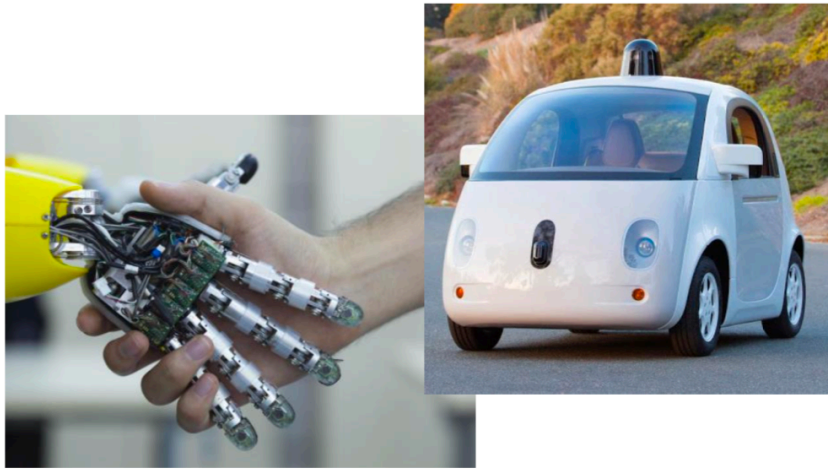
- Goal: select actions to maximize total future reward
- Actions may have long term consequences
- Reward may be delayed
- It may be better to sacrifice immediate reward to gain more long-term reward
  - A financial investment (may take months to mature)
  - Refueling a helicopter (might prevent a crash later)
  - Blocking opponent moves (might help winning chances many moves from now)

Sequential vs Onetime Decisions

- If the current action doesn't affect future decisions
  - Classification, regression, clustering, …



- If the current action affects future decisions
  - Robotics, self-driving, finance, games



RL: Exploration vs Exploitation

- Unlike supervised and unsupervised learning, data is not given before
- Agent learns about the environment by trying things out
- RL in some way a trial-and-error learning
- Agent should learn a good control policy:
  - From its experiences of the environment
  - Without loosing too much reward along the way
- Exploration finds more information about the environment
- Exploitation use the known information to maximize reward

How to balance exploration vs exploitation?

Exploration vs Exploitation: Examples

- Restaurant Selection
  - Exploitation: Go to your favorite restaurant
  - Exploration Try a new restaurant
- Online Banner Advertisements
  - Exploitation Show the most successful ad
  - Exploration Show a different ad
- Oil Drilling
  - Exploitation Drill at the best known location
  - Exploration Drill at a new location
- Game Playing
  - Exploitation Play the move you believe is best
  - Exploration Play an experimental move

Reinforcement Learning Problem

- Agent doesn’t know how the environment works
- Agent has to interact with the environment to learn
- Agent gets two feedback:
  - It can observe the state of the environment at each step
  - It gets a reward at each step
- Agent has to learn a control policy
  - Algorithm to select action sequentially
- Agent’s objective is to maximize the cumulative expected reward

Class Logistics

Basic Logistics

- Instructor: Dileep Kalathil
- Grader: TBA
- Meeting: MWF 11:30 AM - 12:20PM
- Office Hours: Monday, Tuesday 4:00 PM - 5:00 PM

Prerequisites

- Basic probability (ECEN 646 or equivalent)
- Basic multivariate calculus and linear algebra
- Basic idea about optimization (will cover quickly in class)
- Programming in Python (assignments will use Python/Tensorflow)

Syllabus

Week	Topics	Reading
1	Introduction to RL Linear Classification	RLI Ch.1 UML Ch.9, 15
2	Gradient Descent Algorithms Stochastic Gradient Descent Algorithms	COA Ch.2 UML Ch.14
3	Neural Networks Backpropagation	ESL Ch.11 UML Ch.20
4	Training Neural Networks	Class Notes
5	Markov Chains Markov Decision Processes	NDP Ch.2 RLI Ch.3
6	Dynamic Programming	NDP Ch.2, RLI Ch.4
7	Monte Carlo Methods Temporal Difference Learning	RLI Ch.5-6 NDP Ch. 5
8	Q-Learning, SARSA	RLI Ch.5, NDP Ch.5
9	Function Approximation Methods	RLI Ch.9, NDP Ch.6
10	Deep RL	Class Notes
11	Policy Gradient Methods	RLI Ch.13
12	Exploration/Exploitation Bandits Learning	RLI Ch.2 ARL Ch.3
13	Multi-Agent RL	Class Notes
14	Project Presentation	

Grading

- Assignments: 40%
- Midterm: 20% (November 5)
- Final project: 40%
  - Team of 2-3 students
  - You can select topics related your research areas
  - Some project topics suggestion will be posted later
  - Discuss with the instructor before finalizing the project
  - Grading: Project proposal (one page), project presentation and final report

Set higher goals! Projects can (*should*) lead to publications!