Reinforcement Learning Lecture 1a

Course Introduction
[SutBar] Chapter 1, [Sze] Chapter 1

Outline

- Introduction to Reinforcement Learning
- Course website and logistics

Books

[SutBar] Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction (2nd edition, 2018) freely available online
[Sze] Csaba Szepesvari, Algorithms for Reinforcement Learning freely available online
[ZB] Alex Zai and Brandon Brown, Deep Reinforcement Learning in Action (2nd edition, 2020) freely available online
[GBC] Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning (2016) freely available online
[L] Maxim Lapan, Deep Reinforcement Learning Hands On (2020)
[GK] Laura Graesser and Wah Loon Keng, Foundations of Deep Reinforcement Learning: Theory and Practice in Python (2020)
[SigBuf] Olivier Sigaud and Olivier Buffet (editors), Markov Decision Processes in Artificial Intelligence (2013)
[Put] Martin L. Puterman, Markov Decision Processes: Discrete Stochastic Dynamic Programming (2008)
[Ber] Dimitri P. Bertsekas, Dynamic Programming and Optimal Control (2017)
[Pow] Warren B. Powell, Approximate Dynamic Programming: Solving the Curses of Dimensionality (2015)
[RusNor] Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach (4th Edition) (2020)

Leading Light

Richard S. Sutton
(founding father of RL)

David Silver
(AlphaGo, AlphaStar)

Pieter Abbeel
(SAC, HER…)

Sergey Levine
(GPS, TRPO, GAE…)

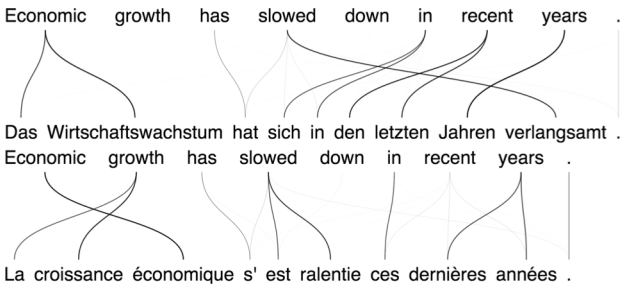
John Schulman
(OpenAI Gym, PPO…)

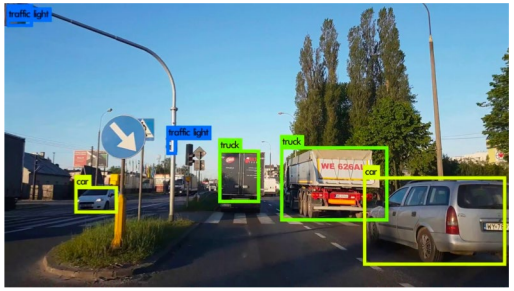
Pascal Poupart
(POMDP, Bayesian RL)

Machine Learning

- Traditional computer science
  - Program computer for every task

- New paradigm
  - Provide examples to machine
  - Machine learns to accomplish a task based on the examples





Definitions

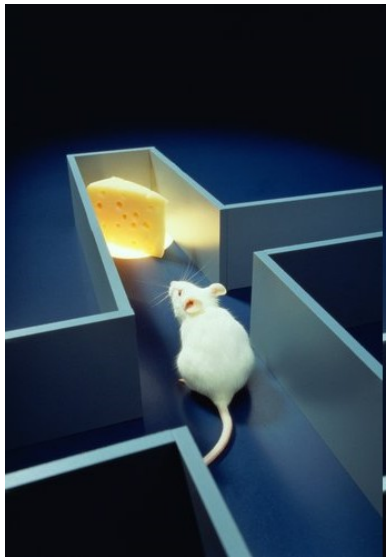
- Arthur Samuel (1959): **Machine learning** is the field of study that gives computers the ability to learn without being explicitly programmed.

- Tom Mitchell (1998): A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

Three Categories

Supervised learning

Reinforcement learning



Unsupervised learning

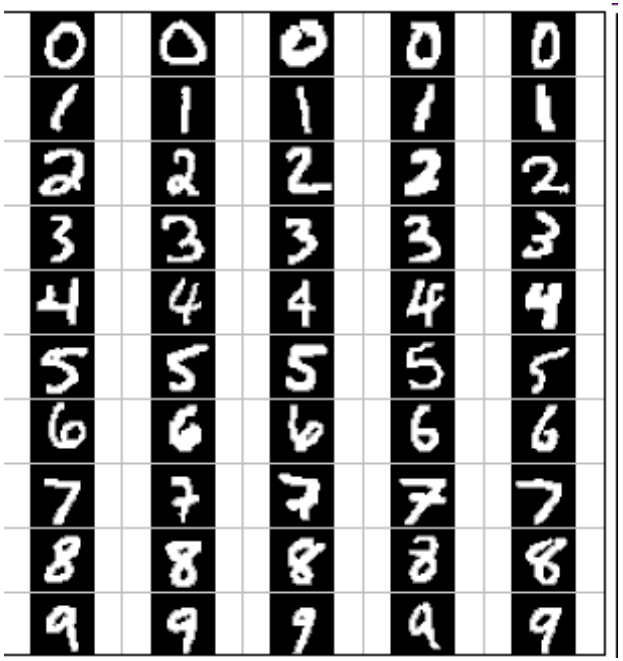
Machine Learning

- Success mostly due to supervised learning
  - Bottleneck: need lots of labeled data
- Alternatives
  - Unsupervised learning, semi-supervised learning
  - Reinforcement Learning

Supervised Learning

- Example: digit recognition (postal code)

- Simplest approach: memorization



Supervised Learning

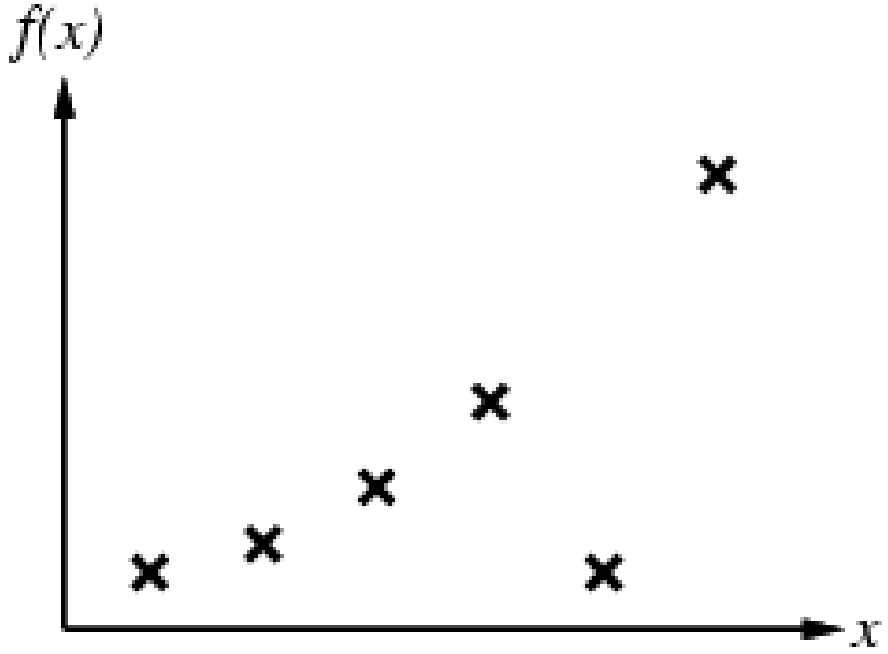
- Nearest neighbour:

More Formally

- 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

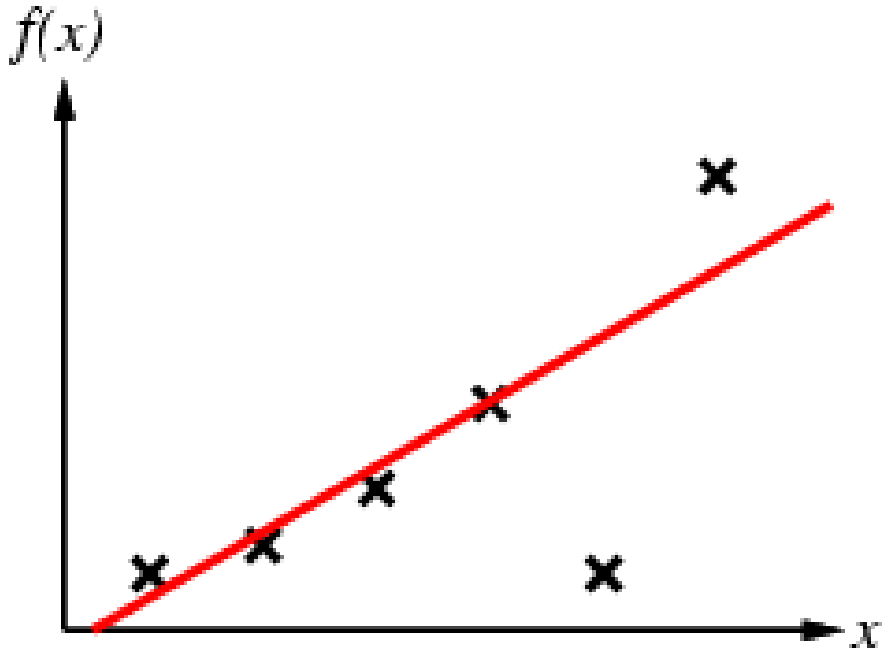
Prediction

- Find function h that fits f at instances x



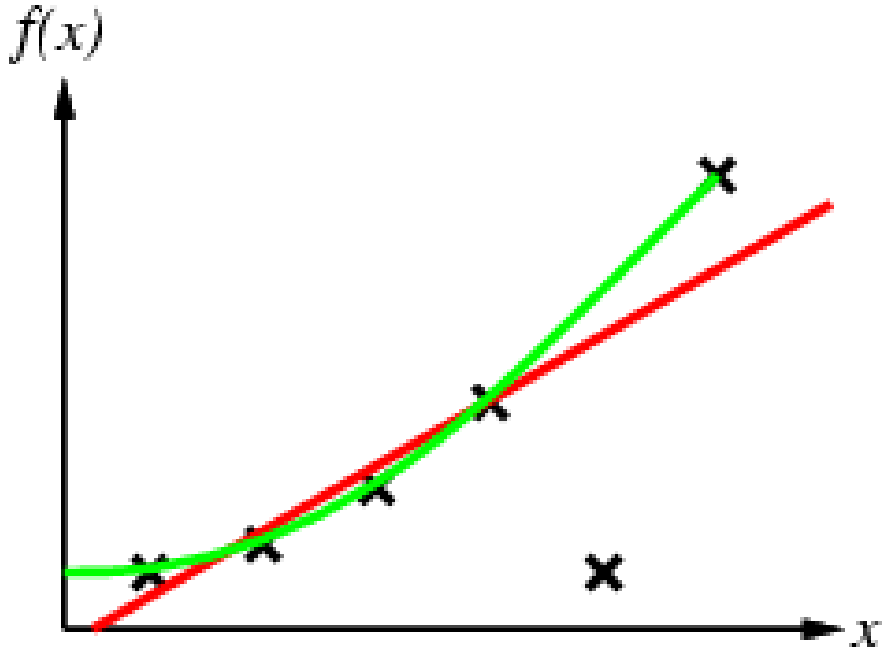
Prediction

- Find function h that fits f at instances x



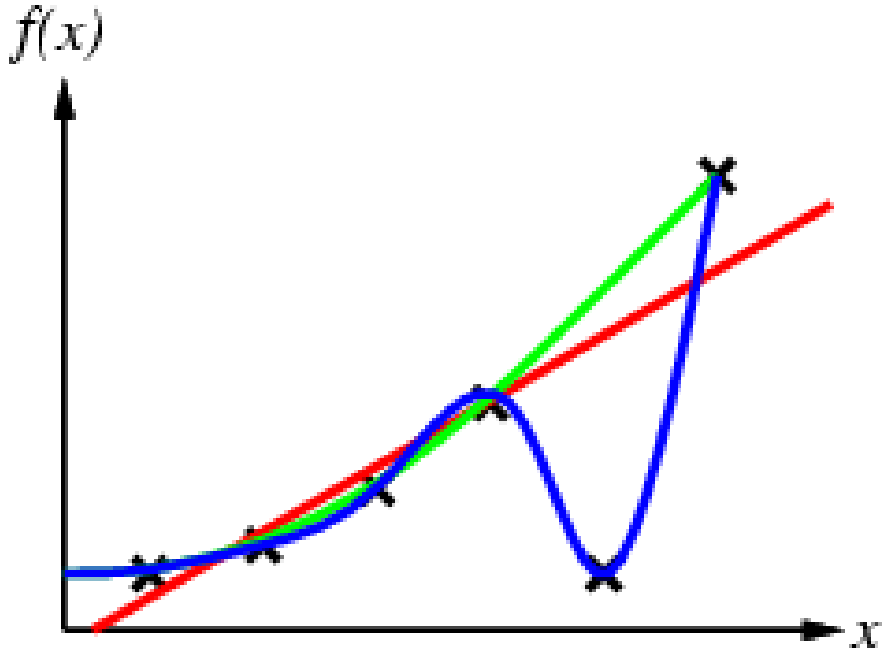
Prediction

- Find function h that fits f at instances x



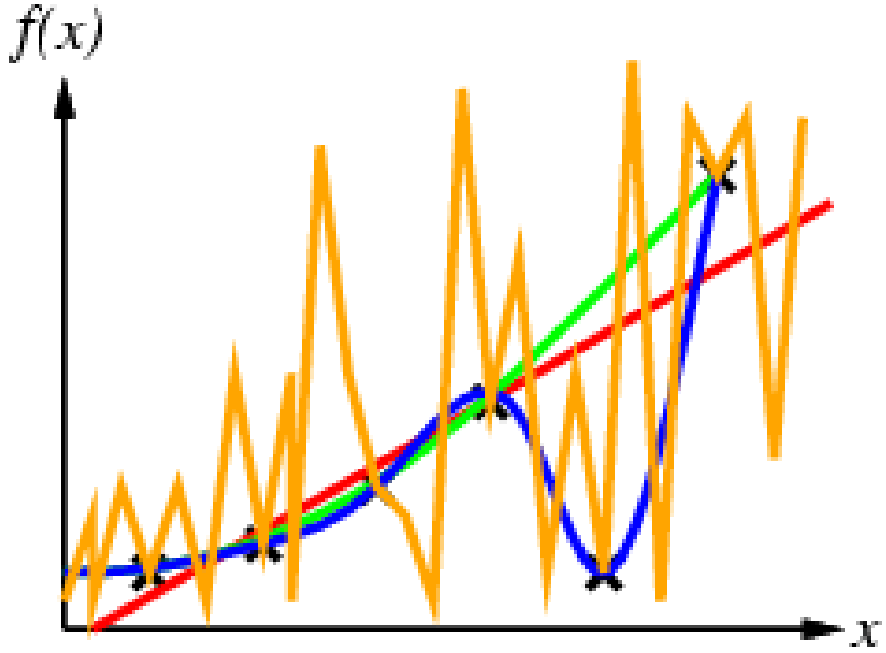
Prediction

- Find function h that fits f at instances x



Prediction

- Find function h that fits f at instances x



Generalization

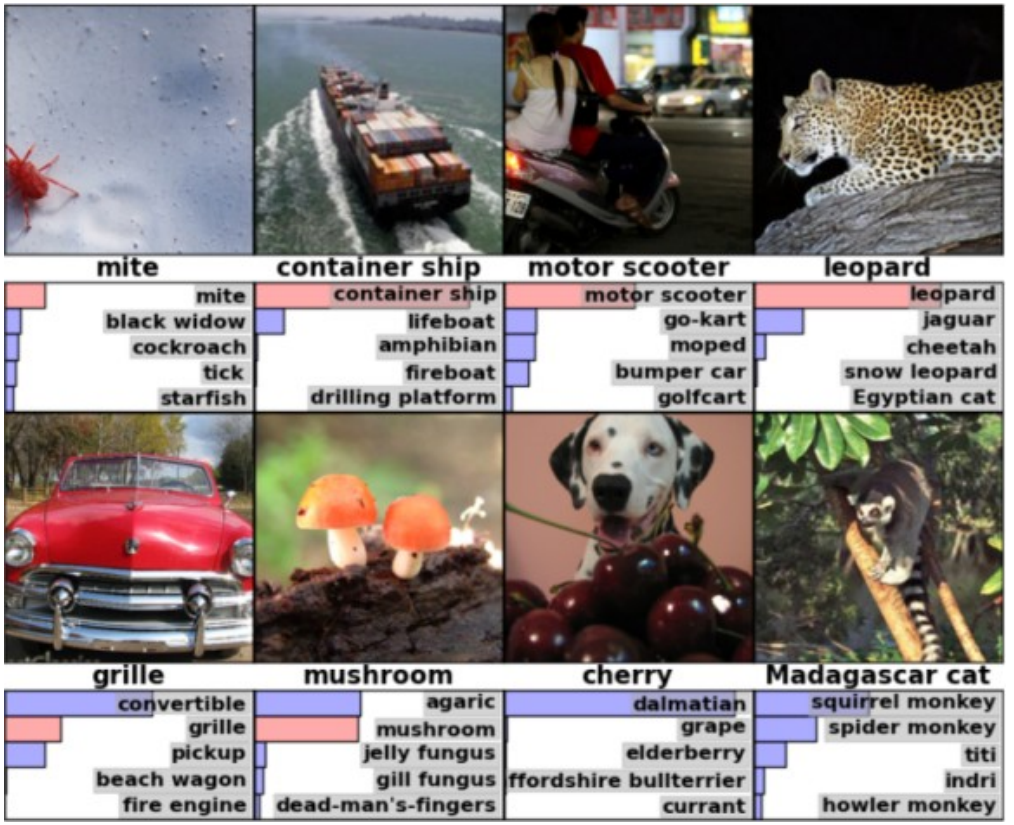
- Key: a good hypothesis will generalize well (i.e. predict unseen examples correctly)

- Ockham's razor: prefer the simplest hypothesis consistent with data

ImageNet Classification

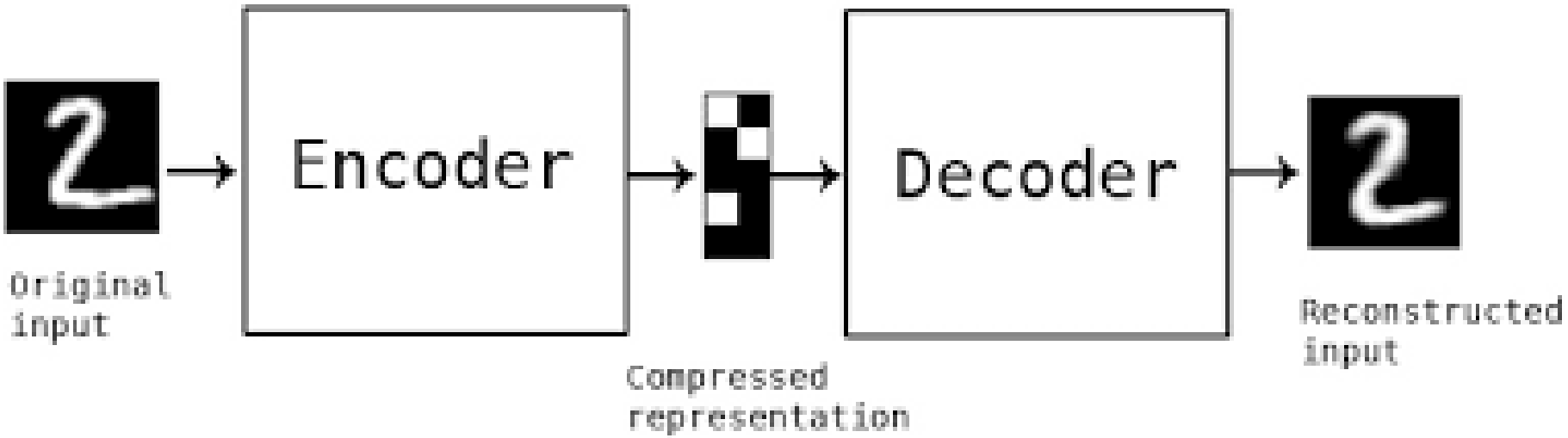
- 1000 classes
- 1 million images

- Deep neural networks (supervised learning)



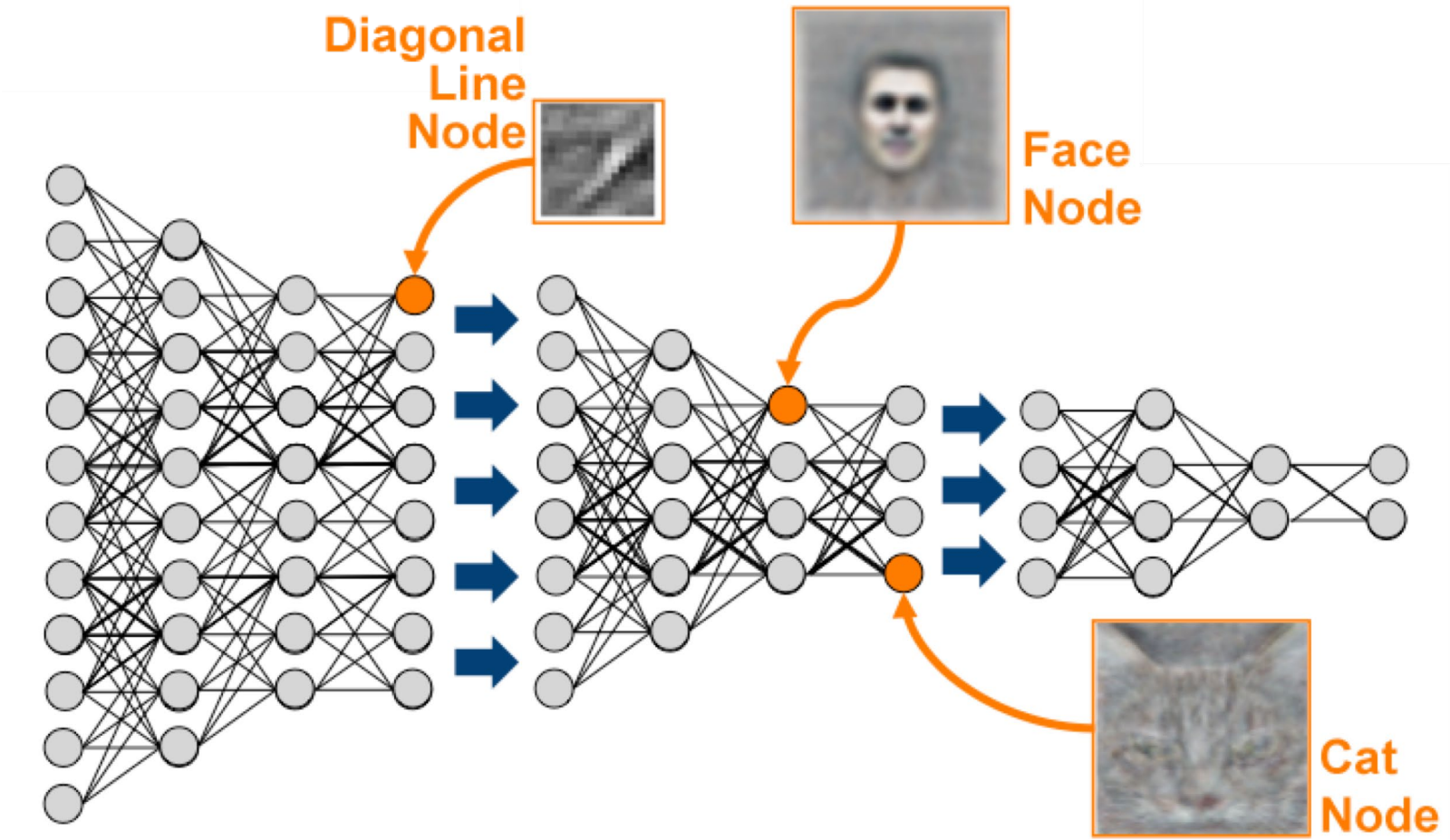
Unsupervised Learning

- Output is not given as part of training set
- Find model that explains the data
  - E.g. clustering, compressed representation, features, generative model



Unsupervised Feature Generation

- Encoder trained on large number of images



What is Reinforcement Learning?

- Reinforcement learning is also known as
  - Optimal control
  - Approximate dynamic programming
  - Neuro-dynamic programming

- Wikipedia: reinforcement learning is an area of machine learning inspired by behavioural psychology, concerned with how software **agents** ought to take **actions** in an **environment** so as to maximize some notion of cumulative **reward**.

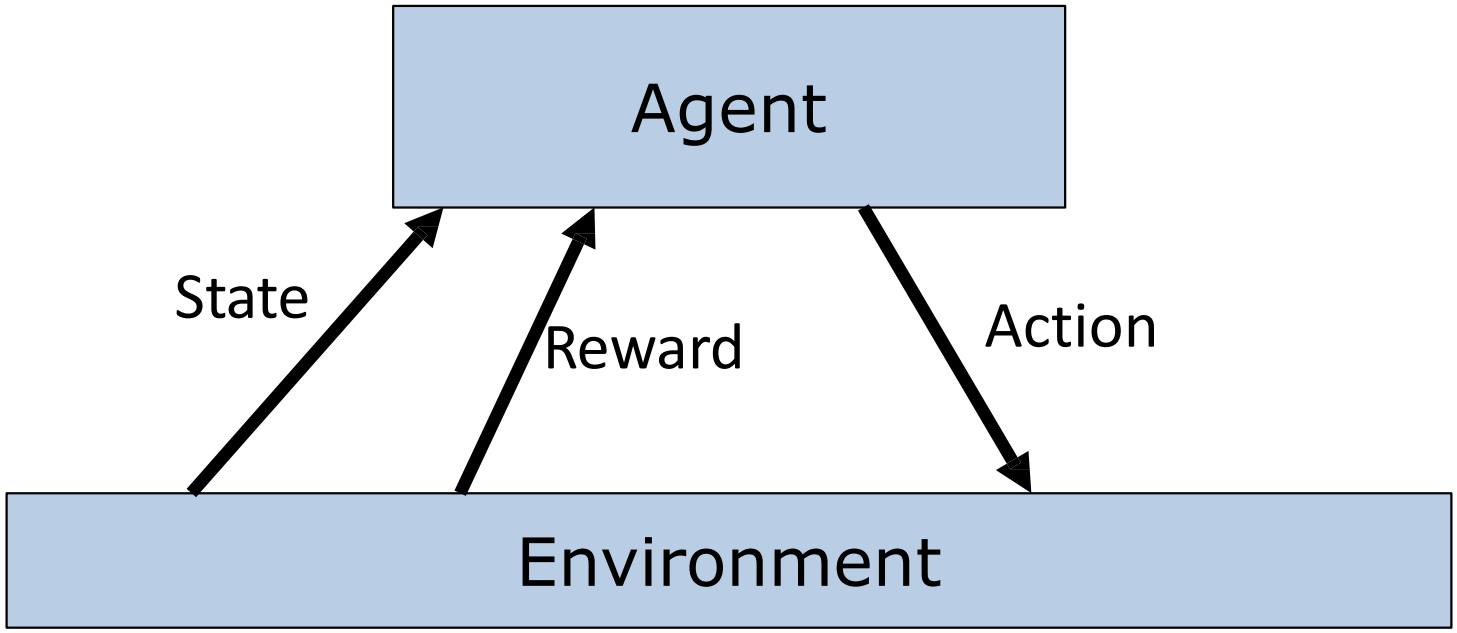
Animal Psychology

- Negative reinforcements:
  - Pain and hunger
- Positive reinforcements:
  - Pleasure and food

- Reinforcements used to train animals

- Let's do the same with computers!

Reinforcement Learning Problem



Goal: Learn to choose actions that maximize rewards

RL Examples

- Game playing (go, atari, backgammon)
- Operations research (pricing, vehicle routing)
- Elevator scheduling
- Helicopter control
- Spoken dialog systems
- Data center energy optimization
- Self-managing network systems
- Autonomous vehicles
- Computational finance

Operations research

- Example: vehicle routing

- **Agent:** vehicle routing software
- **Environment:** stochastic demand
- **State:** vehicle location, capacity and depot requests
- **Action:** vehicle route
- **Reward:** - travel costs

Robotic Control

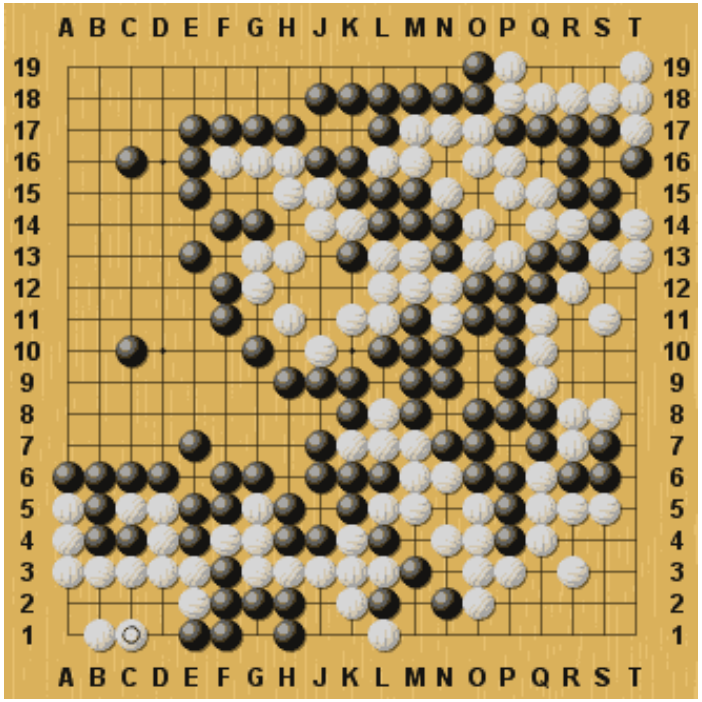
- Example: helicopter control

- **Agent:** controller
- **Environment:** helicopter
- **State:** position, orientation, velocity and angular velocity
- **Action:** collective pitch, cyclic pitch, tail rotor control
- **Reward:** - deviation from desired trajectory

- 2008 (Andrew Ng): automated helicopter wins acrobatic competition against humans

Game Playing

- Example: Go (one of the oldest and hardest board games)



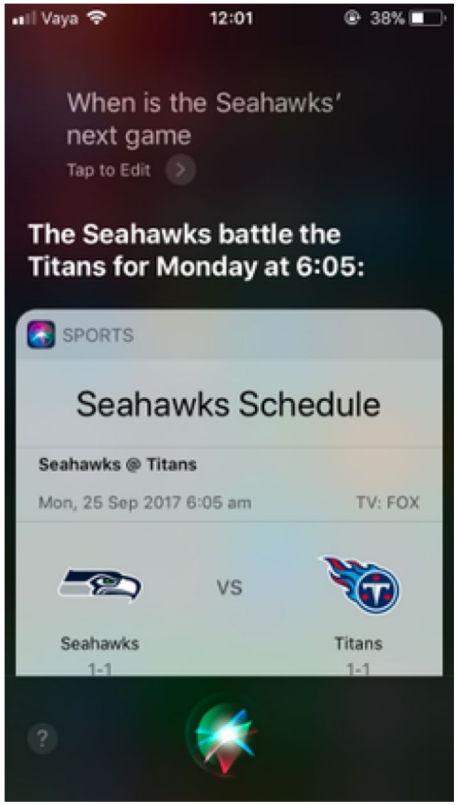
- **Agent:** player
- **Environment:** opponent
- **State:** board configuration
- **Action:** next stone location
- **Reward:** +1 win / -1 loose

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

Conversational agent

- **Agent:** virtual assistant
- **Environment:** user
- **State:** conversation history
- **Action:** next utterance
- **Reward:** points based on task completion, user satisfaction, etc.

- Today: active area of research



Computational Finance

- Automated trading

- **Agent:** trading software
- **Environment:** other traders
- **State:** price history
- **Action:** buy/sell/hold
- **Reward:** amount of profit

Example: how to purchase a large # of shares in a short period of time without affecting the price

Reinforcement Learning

- Comprehensive, but challenging form of machine learning
  - Stochastic environment
  - Incomplete model
  - Interdependent sequence of decisions
  - No supervision
  - Partial and delayed feedback

- **Long term goal**: lifelong machine learning

Course Website

乐学 简体中文 (zh_cn)

强化学习-2022

网络教室 / 我的课程 / 2022-2023第一学期本科生 / 计算机学院 / 强化学习-2022

新闻通告

授课教师：礼欣

个人主页：https://cs.bit.edu.cn/szdw/jsml/js/lixin/index.htm

强化学习小组主页：https://bit1029public.github.io/

助教：郁杰、臧宏宇

选课密码：@lixin

Course Description:

The course introduces students to the design of algorithms that enable machines to learn based on reinforcements. In contrast to supervised learning where machines learn from examples that include the correct decision and unsupervised learning where machines discover patterns in the data, reinforcement learning allows machines to learn from partial, implicit, and delayed feedback. This is particularly useful in sequential decision making tasks where a machine repeatedly interacts with the environment or users. Applications of reinforcement learning include robotic control, autonomous vehicles, game playing, conversational agents, assistive technologies, computational finance, operations research, etc.

Course Objectives:

At the end of the course, students should have the ability to:

- Model tasks as reinforcement learning problems
- Identify suitable algorithms and apply them to different reinforcement learning problems
- Design new reinforcement learning algorithms

Course Overview

Topics

- Markov decision processes
- Bandits
- Model free reinforcement learning
- Model based reinforcement learning
- Partially observable reinforcement learning
- Deep reinforcement learning
- Hierarchical reinforcement learning
- ~~Bayesian reinforcement learning~~
- ~~Distributional reinforcement learning~~
- Immitation learning
- Inverse reinforcement learning
- Multi-Task reinforcement learning
- ~~Meta reinforcement learning~~

Preparing the Machine

- pytorch

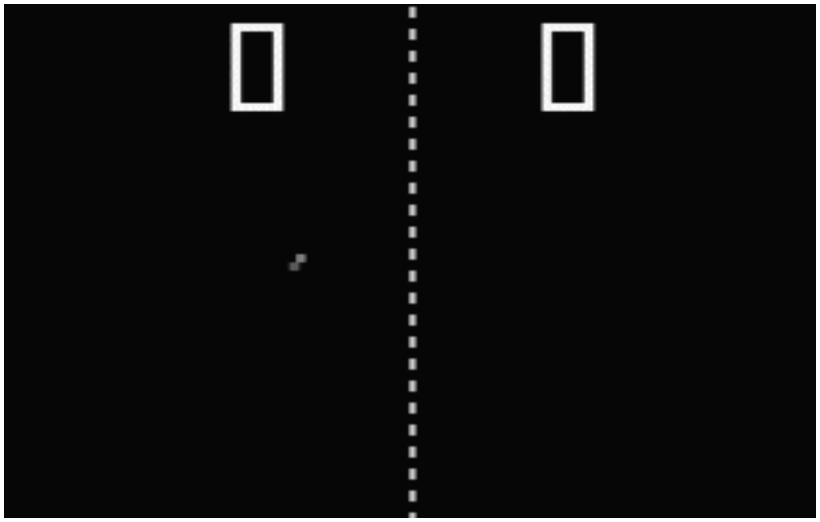
  pip install torch -i http://pypi.douban.com/simple/ --trusted-host pypi.douban.com
- matplotlib

  pip install matplotlib -i http://pypi.douban.com/simple/ --trusted-host pypi.douban.com
- numpy

  pip install numpy -i http://pypi.douban.com/simple/ --trusted-host pypi.douban.com
- scikit-learn

  pip install scikit-learn -i http://pypi.douban.com/simple/ --trusted-host pypi.douban.com

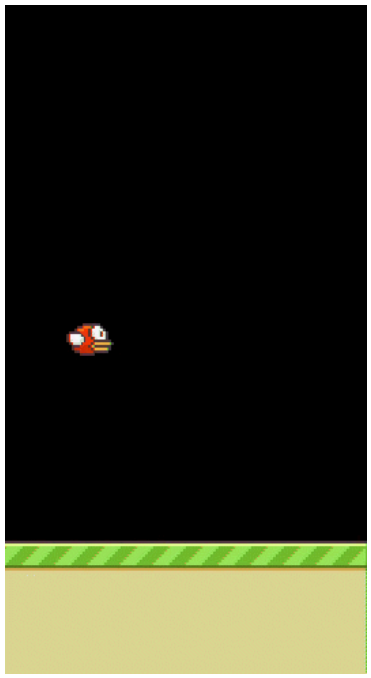
Show cases for Atari Games



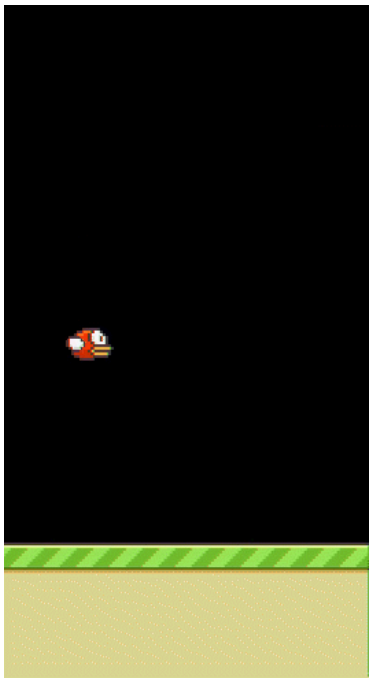
Atari游戏-Pong

Atari游戏-Breakout

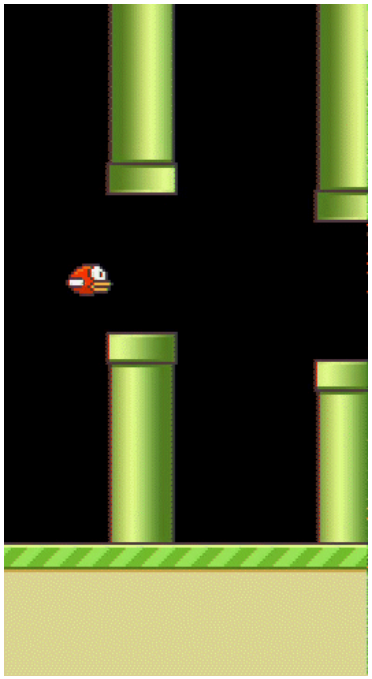
Show Case for Flappy Bird



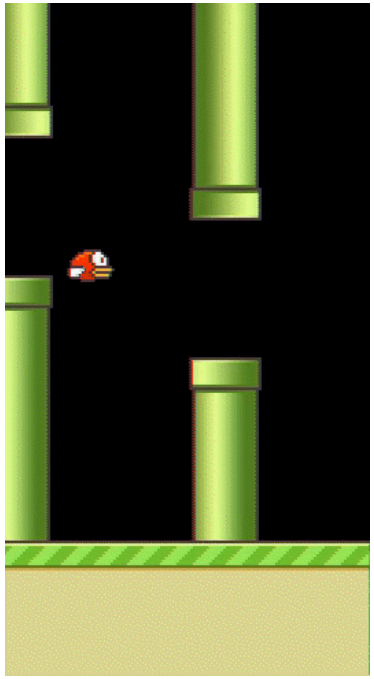
t=25万 2-3次



t=100万 24次



t=210万 死不掉



t=280万 死不掉

以上为不同迭代次数下，训练的神经网络能达到的水平。在迭代到200万次以上时，神经网络控制的小鸟已经很难死掉了。