Introduction

Dit-Yan Yeung

Department of Computer Science and Engineering
Hong Kong University of Science and Technology

COMP 4211: Machine Learning (Fall 2022)

What is Machine Learning?

- **Machine Learning**
  is the science of
  making computer artifacts improve their performance
  with respect to a certain performance criterion
  using example data or past experience,
  without requiring humans to program their behavior explicitly.

- Major learning problem types:
  - Supervised learning: well-posed and more mature; $\sim$60–70% of this course
  - Unsupervised learning: ill-posed; $\sim$20-30% of this course
  - Reinforcement learning: well-posed but less mature; $\sim$10% of this course.

YouTube Time:
Some 'Introduction to Machine Learning' Videos

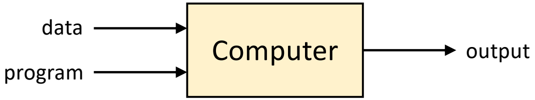
- https://www.youtube.com/watch?v=ukzFI9rgwfU
- https://www.youtube.com/watch?v=6M5VXKLf4D4
- And many more ...

Problem Formulation

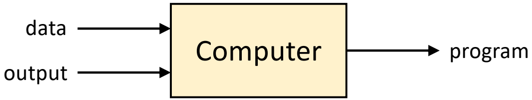
- Training:
  - Given a training set $\mathcal{S} = \left\{(\mathbf{x}^{(\ell)}, \mathbf{y}^{(\ell)})\right\}_{\ell=1}^{N}$ of N labeled examples each of which is in the form of an input-output pair.
  - The problem is to find a function $f(\cdot)$ using $\mathcal{S}$ such that the predicted output $f(\mathbf{x}^{(\ell)})$ for each input $\mathbf{x}^{(\ell)}$ is (usually) close to the actual output $\mathbf{y}^{(\ell)}$. Moreover, we want this to hold also for unseen examples sampled from the same data distribution.
- Testing:
  - Unlabeled examples with only input available are fed one at a time into the function $f(\cdot)$ learned during the training phase to obtain the corresponding predicted output.

Conventional Programming vs. Supervised Learning

Conventional programming



Supervised learning



Types of Supervised Learning Problems

- Two main types (focus of this course):
  - Classification: output is categorical (e.g., ‘dog’, ‘cat’, ‘chair’)
  - Regression: output is continuous-valued (e.g., 0.382, 0.819)
- Variations and generalizations:
  - Learning to rank
  - Structured prediction
  - Active learning
  - Semi-supervised learning
  - And many more ...

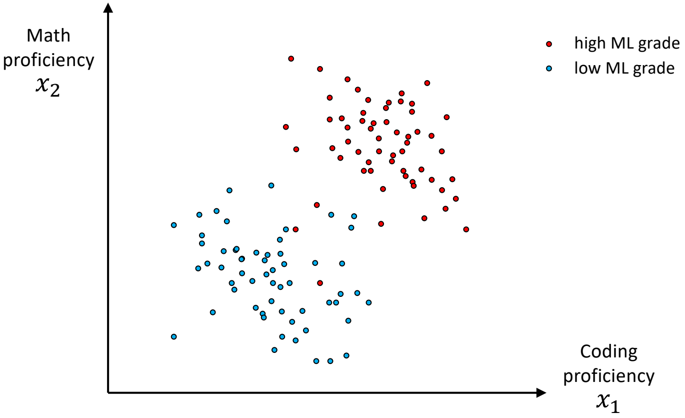
A Classification Example: Problem to Solve

- A professor has been teaching a machine learning course for many years without specifying prerequisites for enrolling in the course.
- To ensure that students enrolled get the most out of the course, she wants to introduce appropriate prerequisites.
- From her experience, she hypothesizes that students with good coding and mathematical skills tend to do well in the course.
- To validate her hypothesis, she wants to make use of the student data from previous offerings of the course to train a classifier to predict whether the performance of a student in the course is good or bad based on the coding proficiency and mathematical proficiency of the student as observed from the performance in relevant courses.

A Classification Example: Solution

- With the help of the administrative department responsible for maintaining student records, the professor has obtained the grades of the relevant courses for all the students who enrolled in previous offerings of her machine learning course.
- She builds a classifier with two input features corresponding to the coding proficiency and mathematical proficiency of a student. The output indicates whether the student is predicted to have high or low grade in the machine learning course.
- If the classifier trained using data from some previous offerings of the course has high prediction accuracy in other offerings, some relevant courses used for defining the input features may be specified as prerequisites.

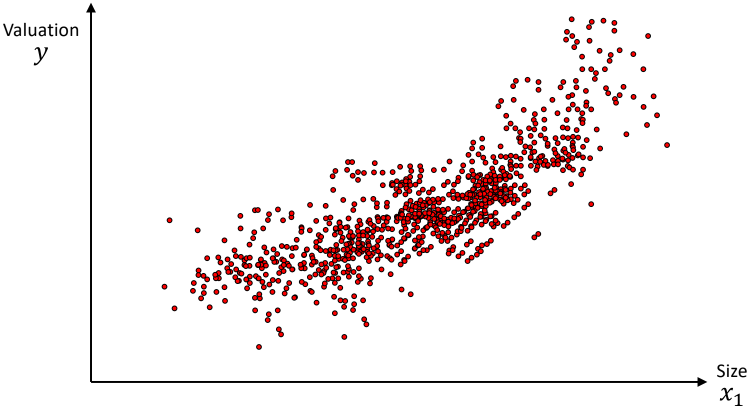
A Classification Example: Data Distribution



A Regression Example

- Problem to solve:
  - Suppose the land registry of a government makes the house price data for a city publicly available.
  - In the dataset, associated with each residential property is its location, year of completion, size, valuation (i.e., estimated price), among other attributes.
  - Instead of relying on the property valuation services provided by banks, we want to make use of the dataset to do our own property valuation.
- Solution:
  - We can train a regression model using the dataset to estimate the valuation of a property given its other (more objective) attributes.
  - We may use only a subset of the attributes (excluding the valuation) as input features to the regression model (for simplicity, the data distribution shown below uses only the size as input).

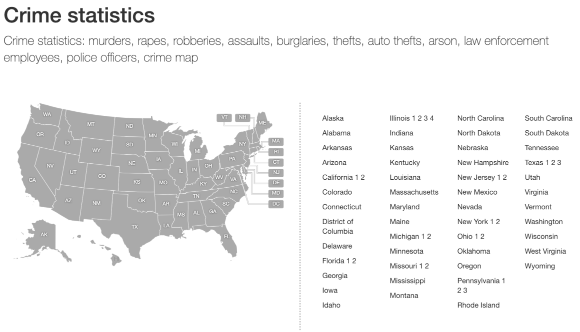
A Regression Example: Data Distribution



Unsupervised Learning

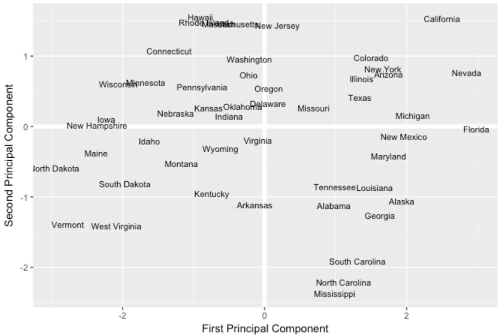
- Given a set $\mathcal{S} = \{\mathbf{x}^{(\ell)}\}_{\ell=1}^{N}$ of N unlabeled examples (like the test data in supervised learning problems).
- Unsupervised learning problems include:
  - Dimensionality reduction: representing the examples using fewer dimensions
  - Clustering: finding natural groups in the examples
  - Density estimation: estimating the underlying probability density function or probability mass function
  - Anomaly detection: detecting anomalies or outliers in the examples
- We will only study dimensionality reduction and clustering in this course.

A Dimensionality Reduction Example: Problem to Solve



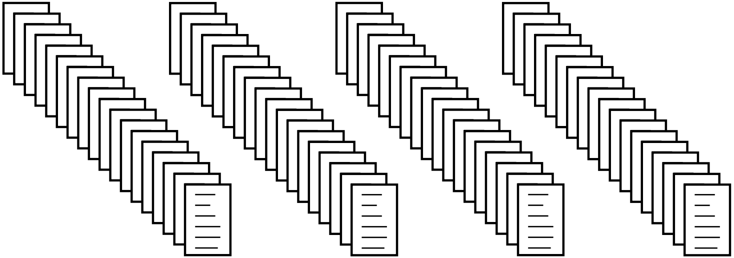
- Given the crime statistics of all the states in USA.
- We want to visualize the data in such a way that cities which are more similar in their crime statistics are closer to each other.

A Dimensionality Reduction Example: Solution



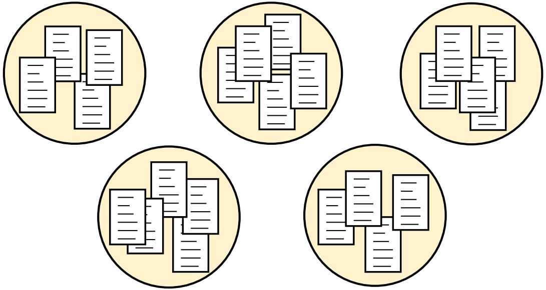
- We can apply a dimensionality reduction model to project the crime data to a 2D space.

A Clustering Example: Problem to Solve



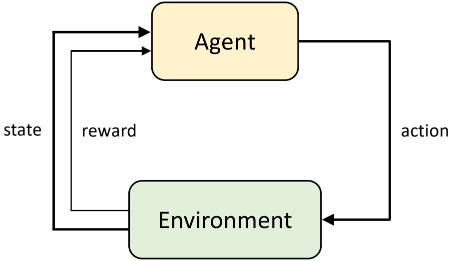
- Given a large collection of online textual documents.
- We want to group the documents according to topic similarity.

A Clustering Example: Solution



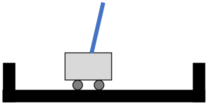
- Each (unstructured) document is converted into a (structured) vector of features characterizing its content.
- The feature vectors are clustered into groups according to content similarity.

Reinforcement Learning



- A reinforcement learning agent learns via interacting with the environment.
- The goal is to maximize the rewards by learning to take appropriate actions at different states.

A Reinforcement Learning Example: Problem to Solve

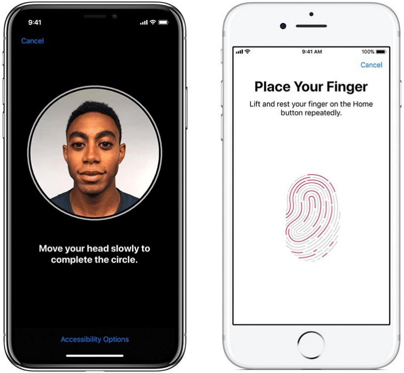


- A cart supporting a hinged, inverted pendulum is placed on a track.
- The goal is to learn to balance the pendulum in an upright position without hitting the end of the track.
- The state represents the current configuration of the cart-pole system.
- Two actions are available to the agent in each state: move the cart left, or move the cart right.
- The reward function is zero everywhere except for the states in which the pole falls or the cart hits the end of the track, in which case the agent receives a -1 reward.

A Reinforcement Learning Example: Solution

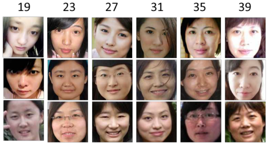
- The reinforcement learning agent learns the sequence of actions necessary to balance the pole and avoid the -1 reward so as to maximize the total rewards.

Biometric Authentication

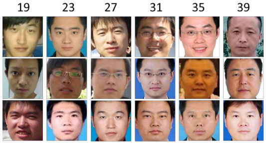


- Face ID and Touch ID as classification problems

Age Estimation



(a) female



(b) male

Dataset	Metric Regression			Ordinal Regression		
	BIFs + LSVR [15]	BIFs + CCA [14]	CNN + LSVR [31]	BIFS + OR-SVM [5]	BIFS + OHRank [6]	Ours (OR-CNN)
MORPH II	4.31	4.73	5.13 (4.77 in [31])	4.21	3.82	**3.27**
AFAD	4.13	4.40	5.56	4.36	3.84	**3.34**

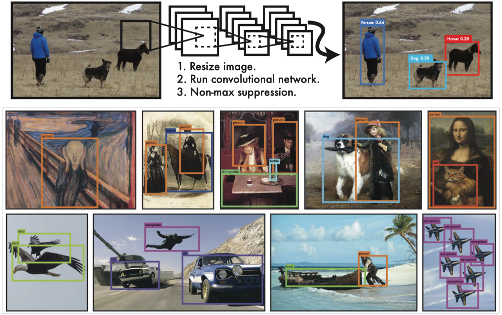
Niu et al., CVPR 2016

- Age estimation from facial images as a regression problem

Personalized Product Recommendation

- Predicting how much a user likes a product as a special type of regression problem

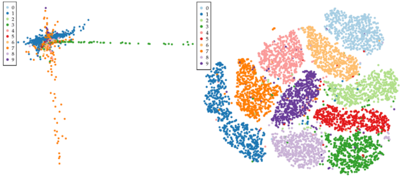
Object Detection



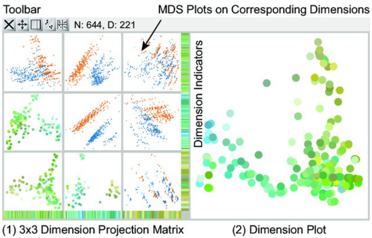
Redmon et al., CVPR 2016

- Object detection involves solving both classification (predicting object category) and regression (predicting object location) problems

Visualization of High-dimensional Data

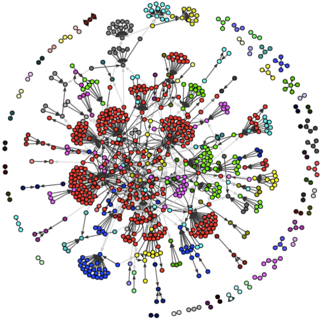


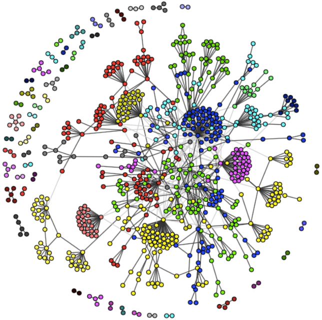
LLE t-SNE



- Dimensionality reduction via projection of high-dimensional data to a low-dimensional space

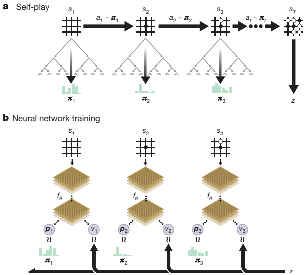
Criminal Organization Detection

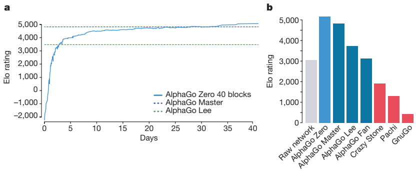




- Applying cluster analysis for community detection in phone call or online social networks

Playing the 'Go' Game





Silver et al., Nature 2017

- Applying reinforcement learning from scratch to become the world champion

Going Beyond the ‘Go’ Game

REPORT

A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play

David Silver[1,2,*,†], Thomas Hubert[1,*], Julian Schrittwieser[1,*], Ioannis Antonoglou[1], Matthew Lai[1], Arthur Guez[1], Marc Lanctot[1], Laurent Sifre[1], Dharshan Kumaran[1], Thore Graepel[1], Timothy Lillicrap[1], Karen Simonyan[1], Demis Hassabis[1,†]

[1]DeepMind, 6 Pancras Square, London N1C 4AG, UK.

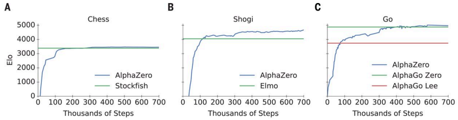
[2]University College London, Gower Street, London WC1E 6BT, UK.

↵†Corresponding author. Email: davidsilver@google.com (D.S.); dhcontact@google.com (D.H.)

↵* These authors contributed equally to this work.

– Hide authors and affiliations

Science 07 Dec 2018:
Vol. 362, Issue 6419, pp. 1140-1144
DOI: 10.1126/science.aar6404



nature › articles › article

Article | Published: 23 December 2020

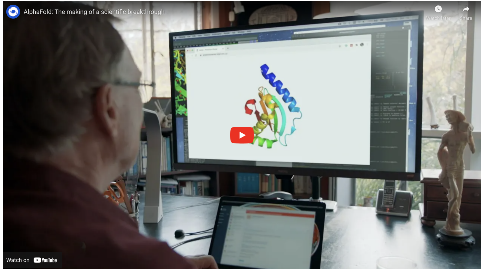
Mastering Atari, Go, chess and shogi by planning with a learned model

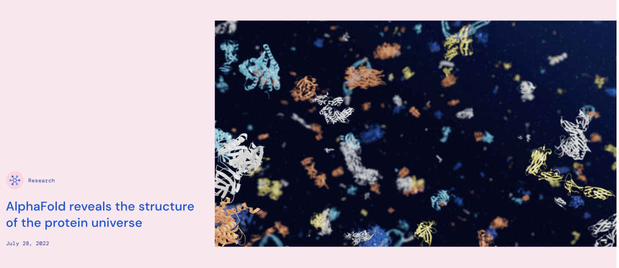
Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap & David Silver

Nature **588**, 604–609(2020) | Cite this article

- From AlphaGo to AlphaGo Zero to AlphaZero to MuZero

From AlphaGo to AlphaFold





- Providing a solution to a 50-year-old grand challenge in biology

Generating Realistic Images and Art from Natural Language