การพัฒนาระบบอัจฉริยะแบบ AIoT (part 2)

ผศ.ดร. ภัทรวิทย์ พลพินิจ

วิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยขอนแก่น

Outline

- Introduction
  - Introduction to AI / ML (This slide)
- Coding
  - Coding in Python
  - Choosing and Building Models
  - Evaluate Model
- Assignment
- Conclusion

Objective

- Have a fundamental uniderstanding of AI and ML
- Learn how to use necessary tools to code ML in python
- Have experience building ML models(regression and classification)
- Can choose appropriate model for a real-world example

What is Artificial Intelligence?

- **Artificial Intelligence (AI)** is an area of computer science that emphasizes the creation of intelligent machines that works and react likes humans.

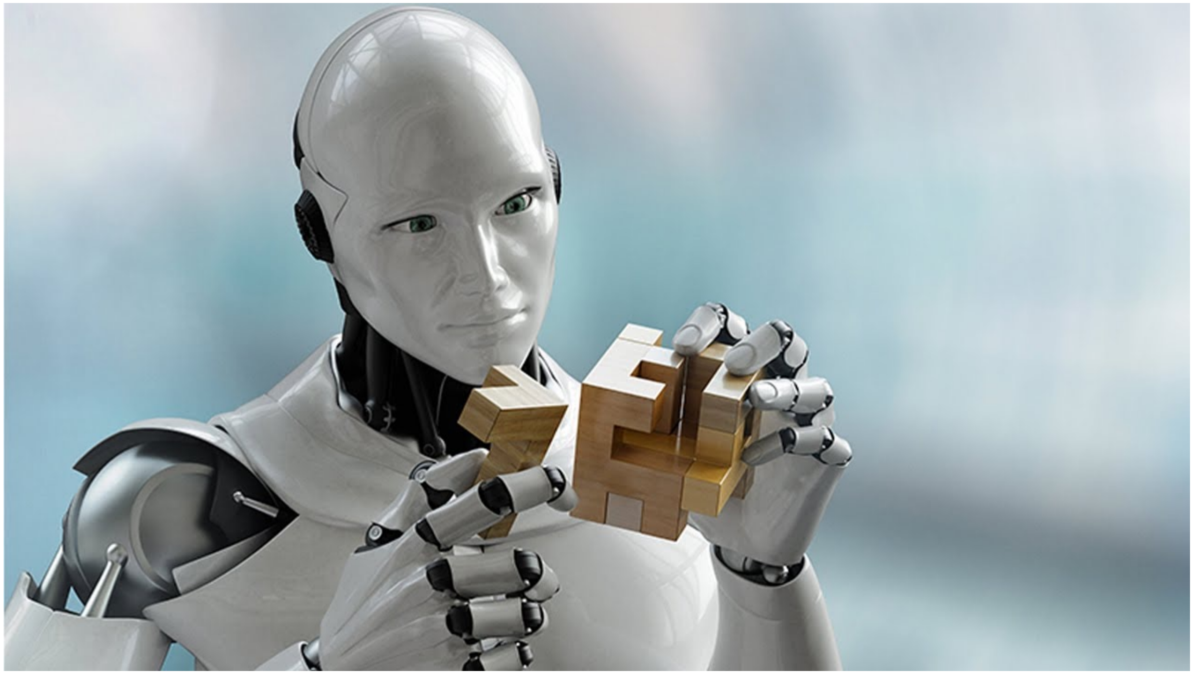


Photo from youtube titled "Artificial Intelligence Robots Development Until 2019 - Machine Learning Robot Ep. 06"

AI Early History

- In 1950 English mathematician **Alan Turing** write a landmark paper titled "Computing Machinery and Intelligence" that asked the question: "Can machine think?"

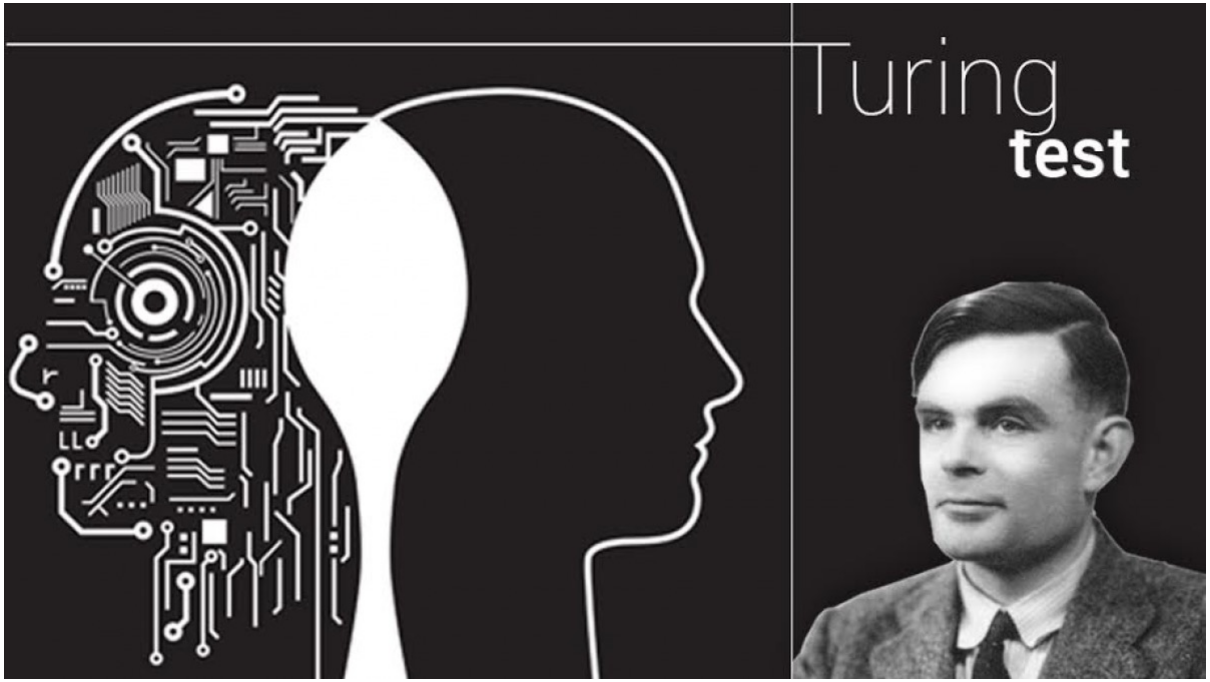


Photo from article titled turing-test-key-contribution-field-artificial-intelligence

AI and Machine Learning and Deep Learning

- **Deep learning** and **machine learning** are sub-fields of **artificial intelligence**, and deep learning is a sub-field of machine learning.

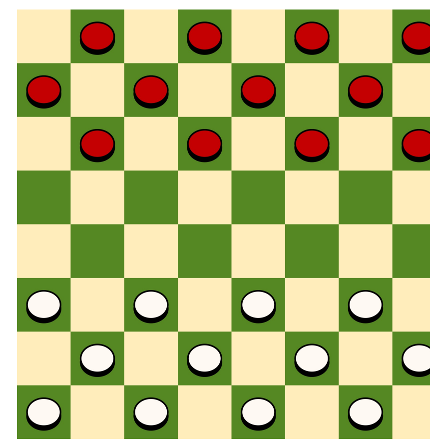
Photo from ShaperAI's article titled "AI terminology a victim of AI's own success"

Definition of Machine Learning

Arthur Samuel (1959): Machine Learning is the field of study that gives the computer the ability to learn without being explicitly programmed.

A. L. Samuel*

Some Studies in Machine Learning Using the Game of Checkers. II—Recent Progress



Photos from Wikipedia

Definition of Machine Learning

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.

- Experience (data): games played by the program (with itself)
- Performance measure: winning rate
- A learning task is given by <P,T,E>

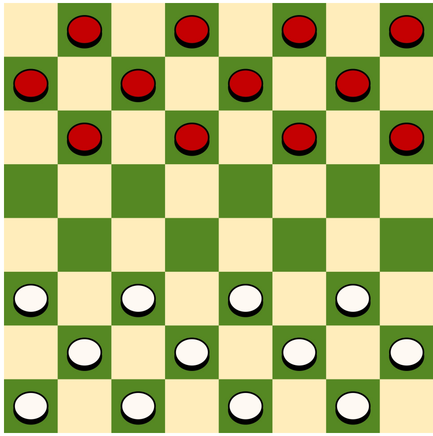
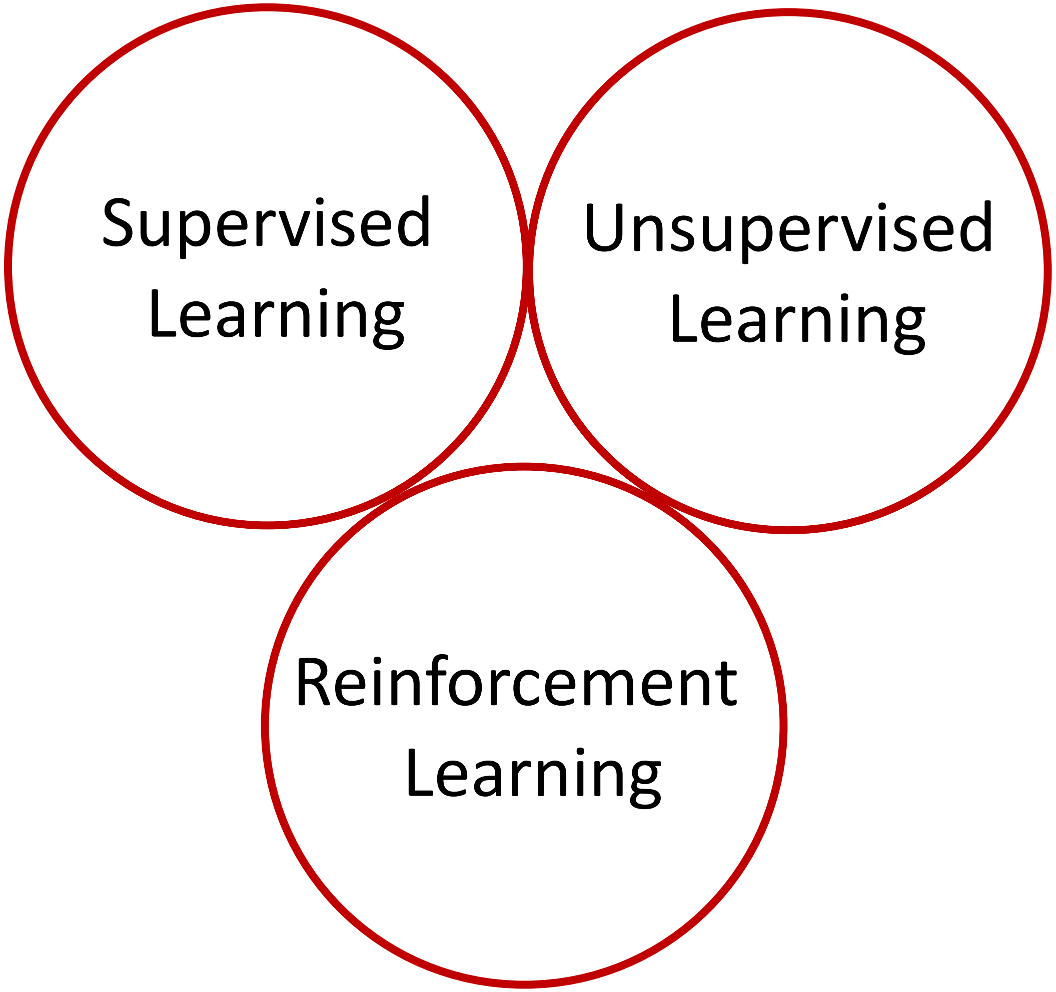


Image from Tom Mitchell’s homepage

Taxonomy of Machine Learning



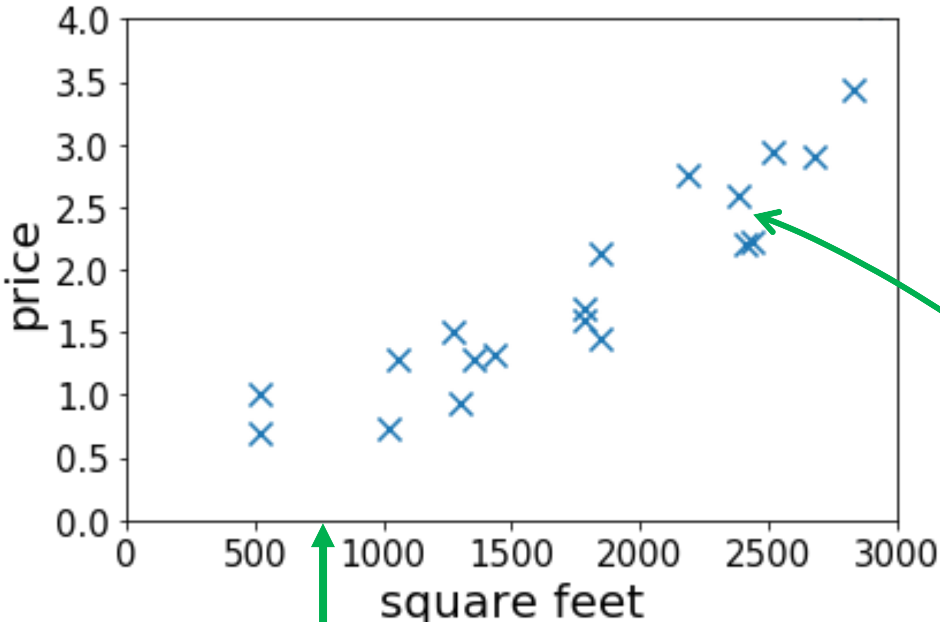
Supervised Learning

Housing Price Prediction (Regression)

- Given: a dataset that contains n samples

$$\left(x^{(1)}, y^{(1)}\right), \dots \left(x^{(n)}, y^{(n)}\right)$$

- Task: if a residence has x square feet, predict its price?



15th sample
$(x^{(15)}, y^{(15)})$

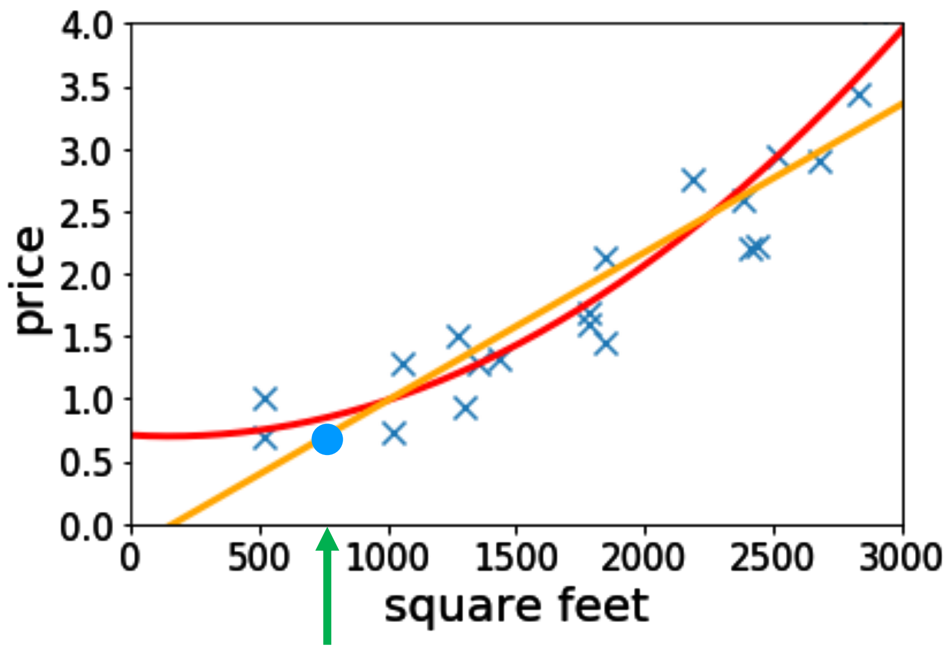
$x = 800$
$y = ?$

Housing Price Prediction (Regression)

- Given: a dataset that contains n samples

$$\left(x^{(1)}, y^{(1)}\right), \ldots \left(x^{(n)}, y^{(n)}\right)$$

- Task: if a residence has x square feet, predict its price?



$x = 800$

$y = ?$

- Try fitting linear/quadratic functions to the dataset

More Features

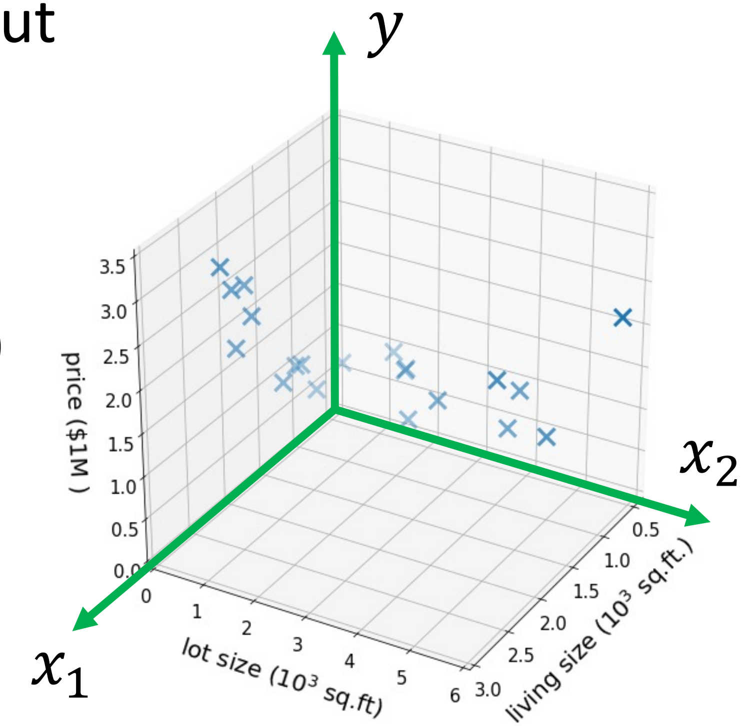
- Suppose we also know the lot size
- Task: find a function that maps

$$\underbrace{\text{(size, lot size)}}_{\substack{\text{features/input}\\ x \in \mathbb{R}^2}} \rightarrow \underbrace{\text{price}}_{\substack{\text{label/output}\\ y \in \mathbb{R}}}$$

- Dataset: $\left(x^{(1)}, y^{(1)}\right), \dots, \left(x^{(n)}, y^{(n)}\right)$

where $x^{(i)} = \left(x_1^{(i)}, x_2^{(i)}\right)$

- "Supervision" refers to $y^{(1)}, \dots, y^{(n)}$



High-dimensional Features

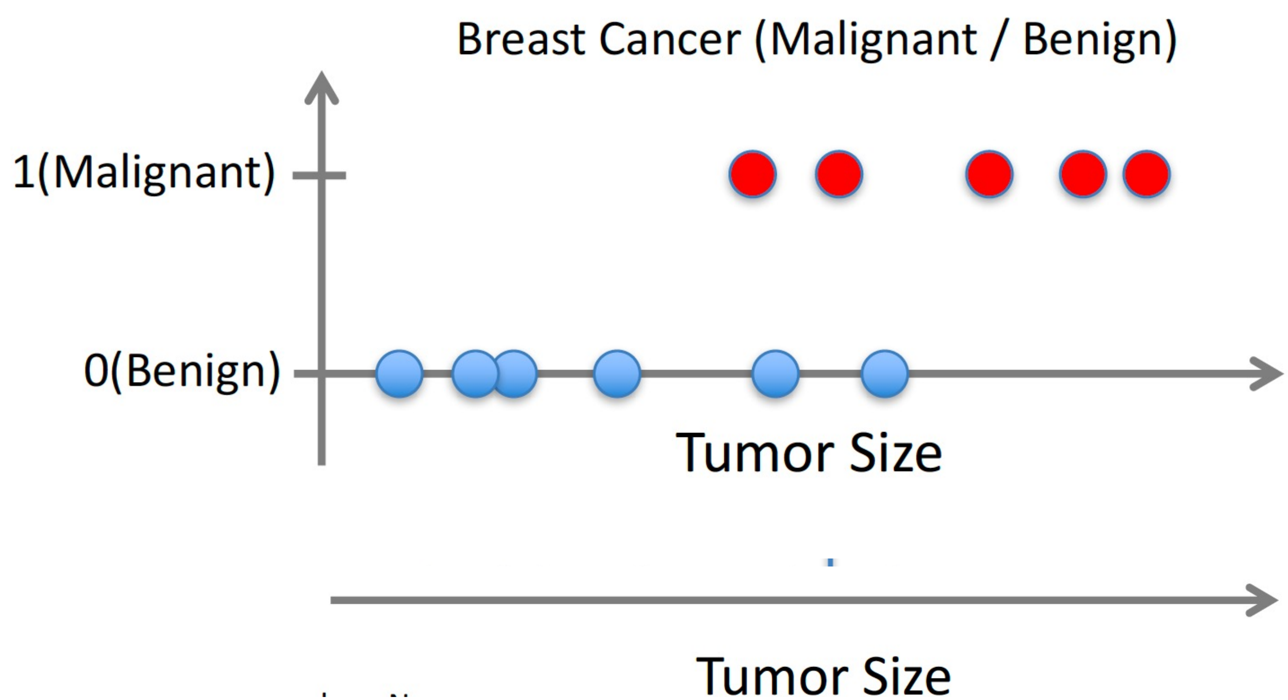
- $x \in \mathbb{R}^d$ for large d
- E.g.,

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ \vdots \\ \vdots \\ x_d \end{bmatrix} \begin{array}{l} \text{--- living size} \\ \text{--- lot size} \\ \text{--- \# floors} \\ \text{--- condition} \\ \text{--- zip code} \\ \quad\quad\vdots \\ \\ \end{array} \longrightarrow \quad y \text{ --- price}$$

- This can go to infinite dimensional features

Breast Cancer Determination (Classification)

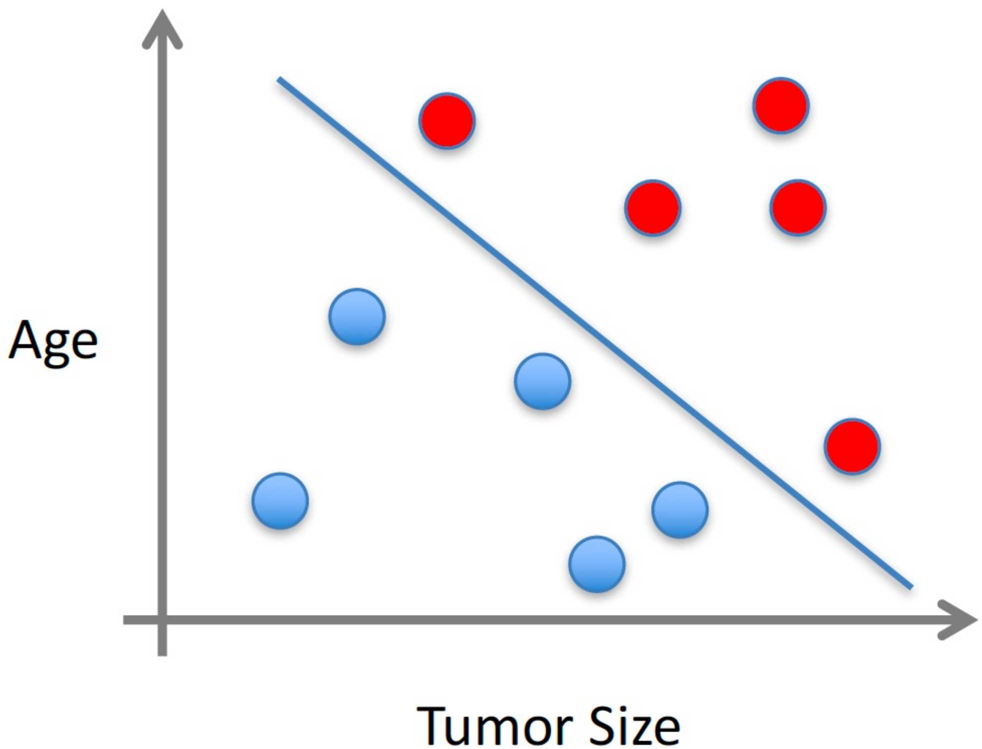
- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- Learn a function f(x) to predict y given x
  - y is categorical == classification



Breast Cancer Determination (Classification)

Higher Dimension Features

- x can be multi-dimensional
  - Each dimension corresponds to an attribute



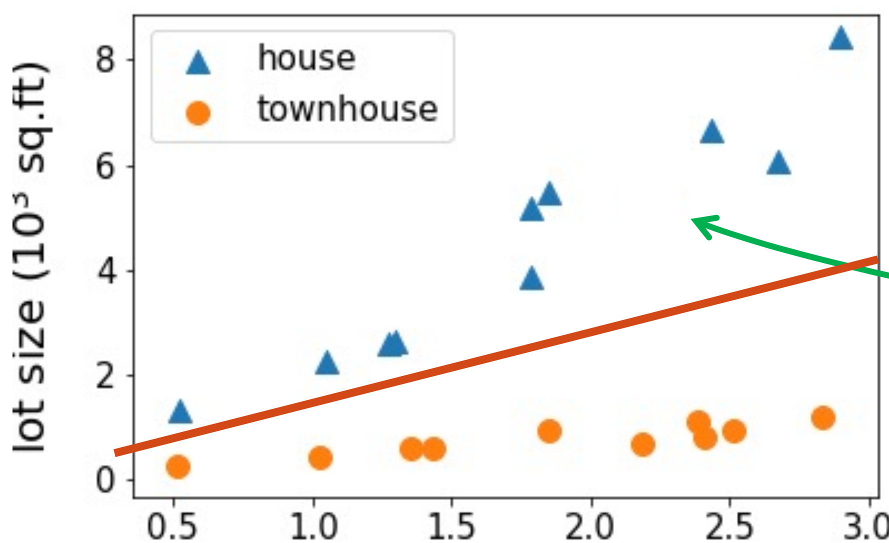
- Clump Thickness
- Cell Size
- Cell Shape

…

Regression vs Classification

- regression: if $y \in \mathbb{R}$ is a continuous variable
  - e.g., price prediction
- classification: the label is a discrete variable
  - e.g., the task of predicting the types of residence

(size, lot size) $\rightarrow$ house or townhouse?



$y =$ house or townhouse?

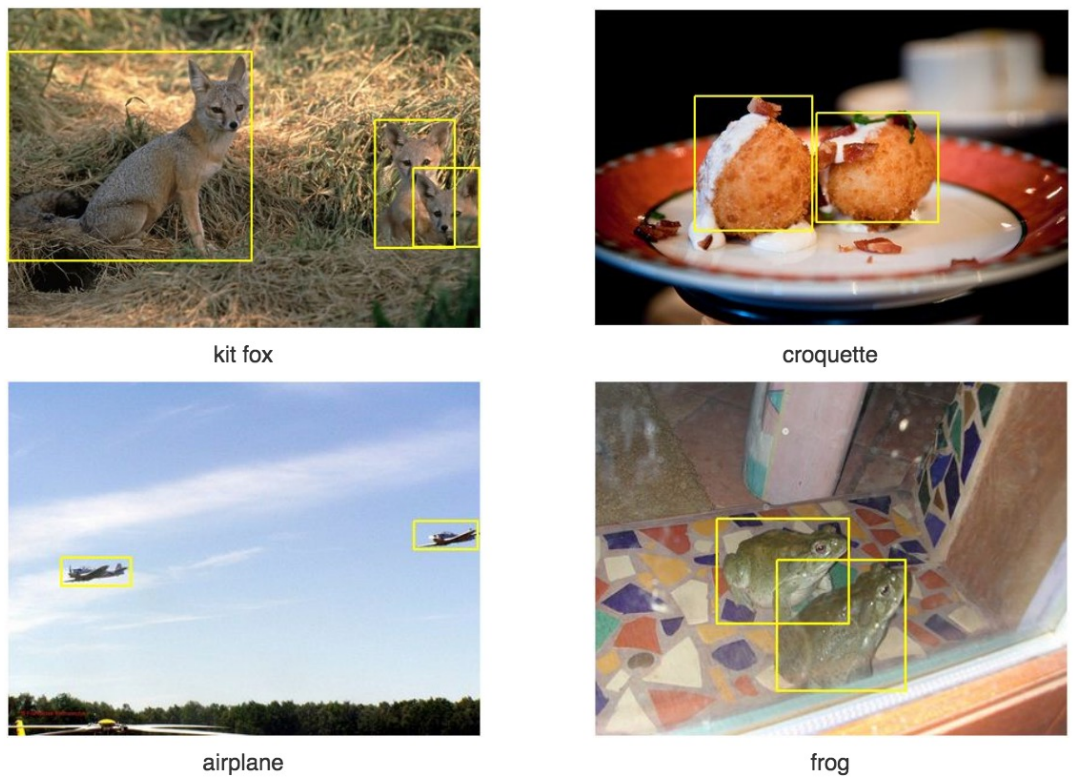
Supervised Learning in Computer Vision

- Image Classification
  - x = raw pixels of the image, y = the main object

ImageNet Large Scale Visual Recognition Challenge. Russakovsky et al.'2015

Supervised Learning in Computer Vision

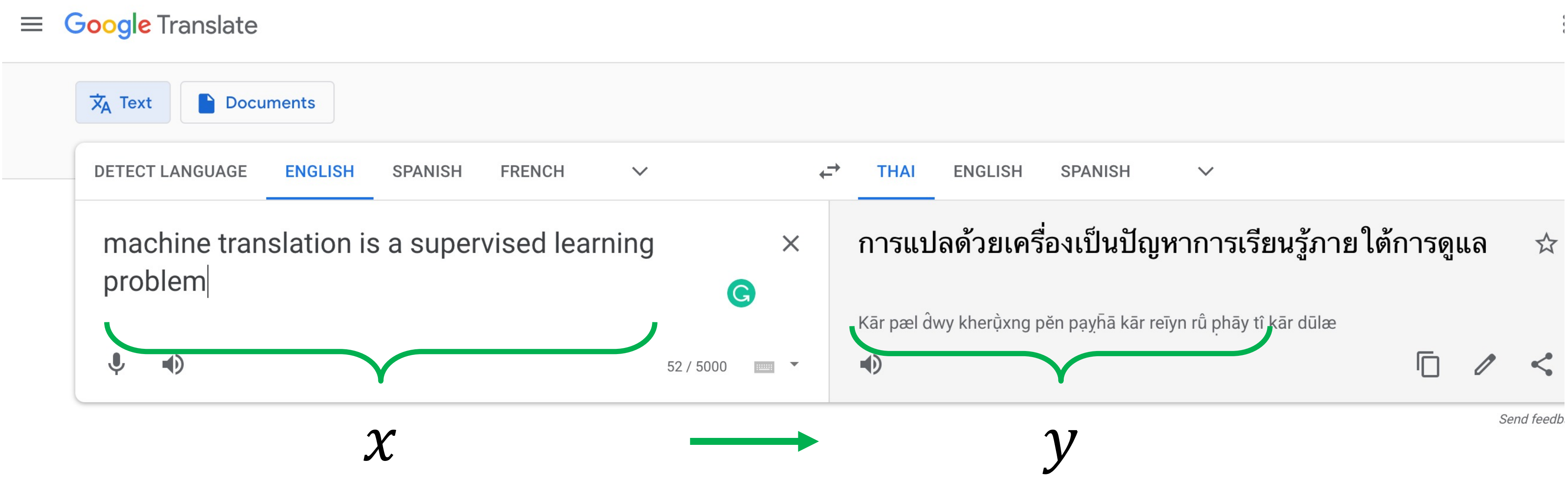
- Object localization and detection
  - $x =$ raw pixels of the image, $y =$ the bounding boxes



ImageNet Large Scale Visual Recognition Challenge. Russakovsky et al.'2015

Supervised Learning in Natural Language Processing

- Machine translation



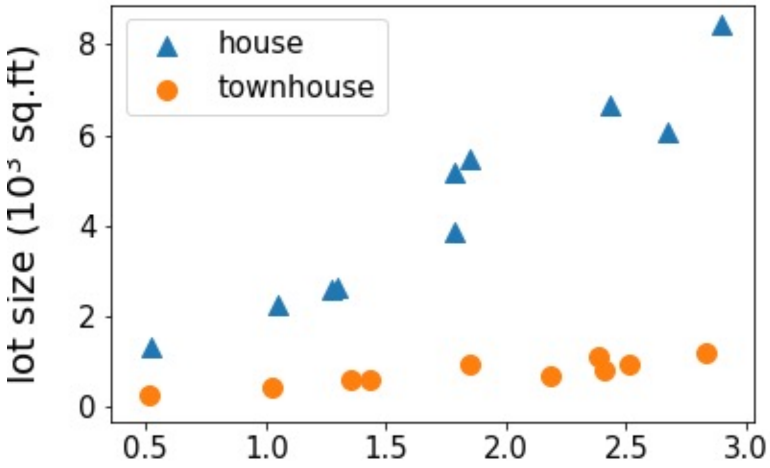
- Note: this will not be covered here. However, you should explore the topic more own your own as this is a very useful and important field.

Unsupervised Learning

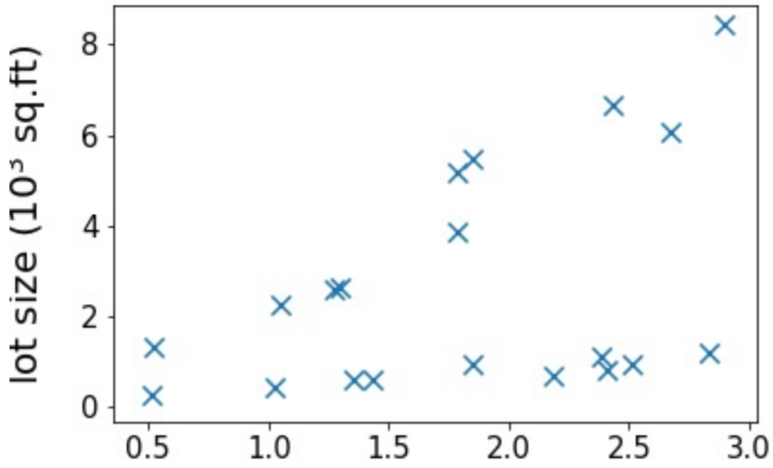
Unsupervised Learning

- Dataset contains no labels: $x^{(1)}, \ldots x^{(n)}$
- Goal (vaguely-posed): to find interesting structures in the data

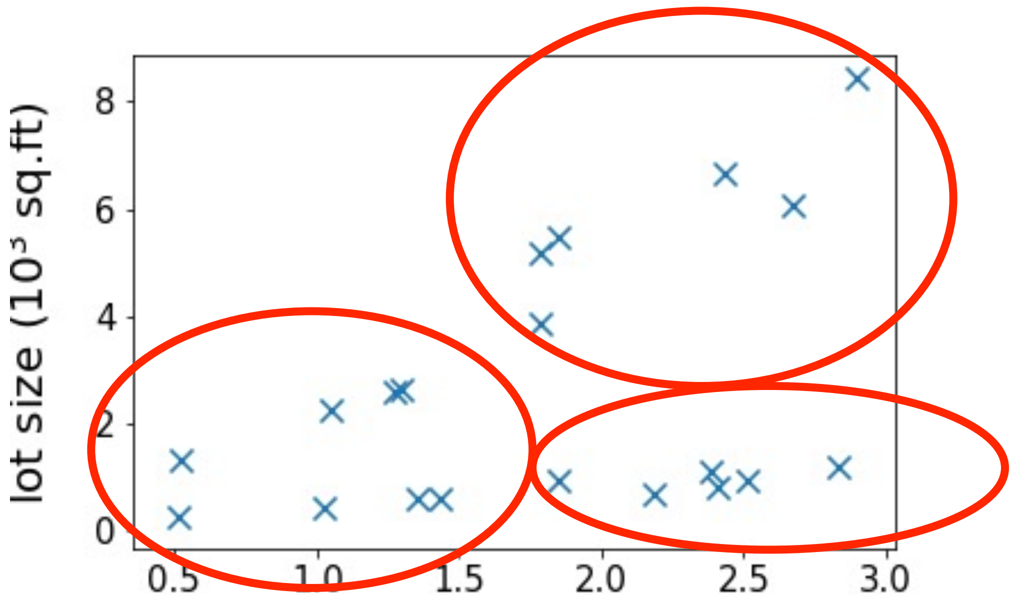
supervised



unsupervised

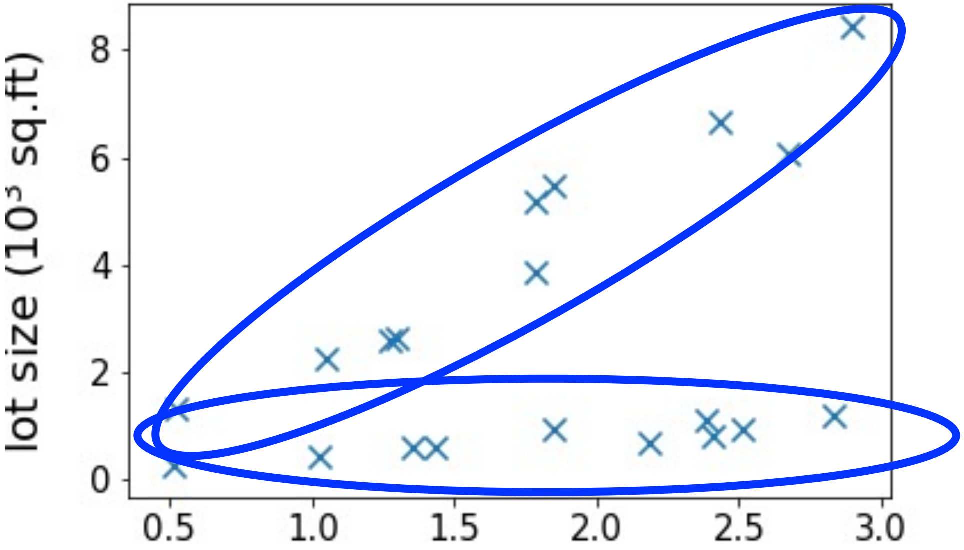


Clustering

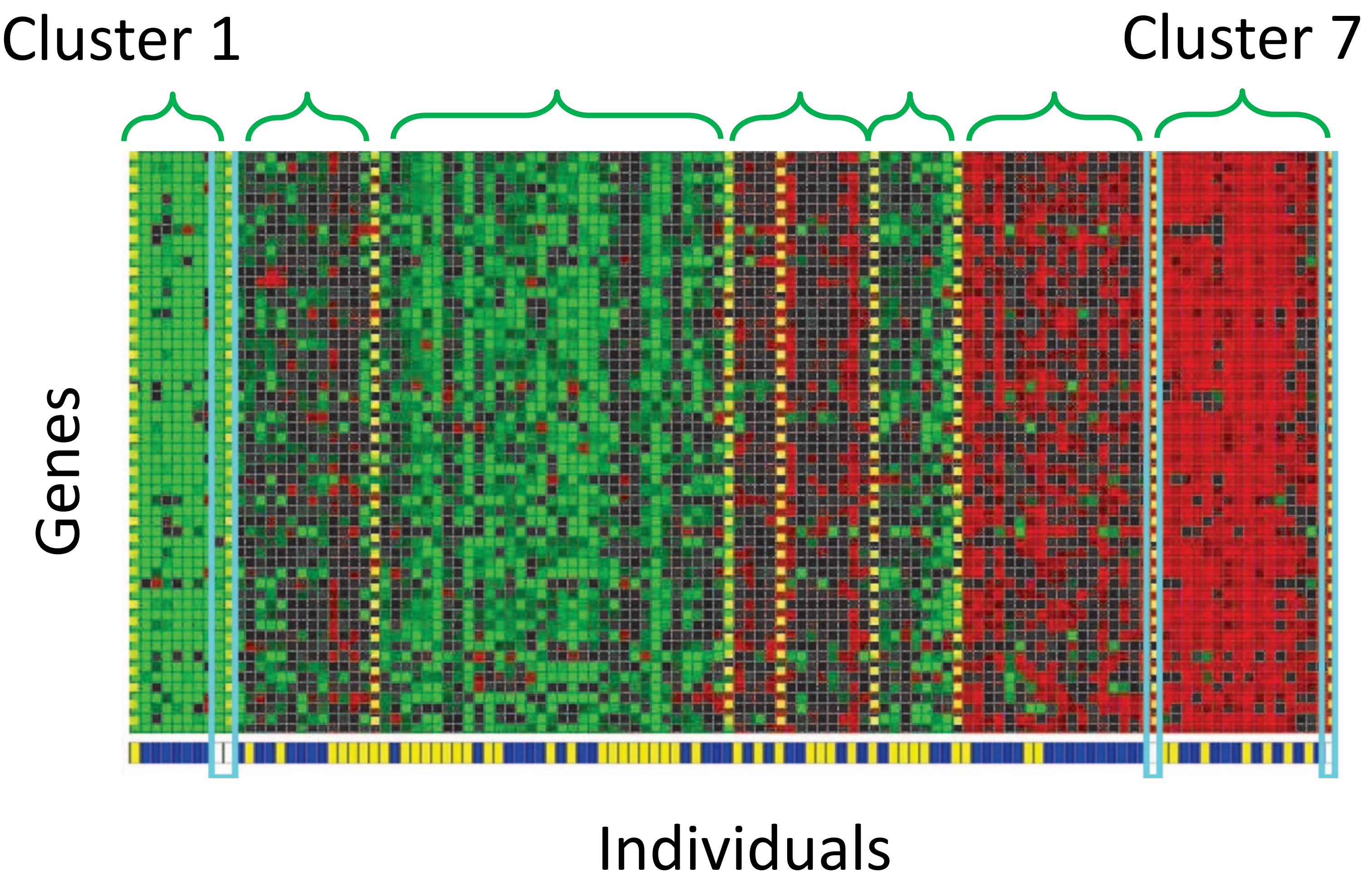


Clustering

- Example : k-mean clustering, mixture of Gaussians



Clustering Genes



Identifying Regulatory Mechanisms using Individual Variation Reveals Key Role for Chromatin Modification. [Su-In Lee, Dana Pe'er, Aimee M. Dudley, George M. Church and Daphne Koller. '06]

Latent Semantic Analysis (LSA)

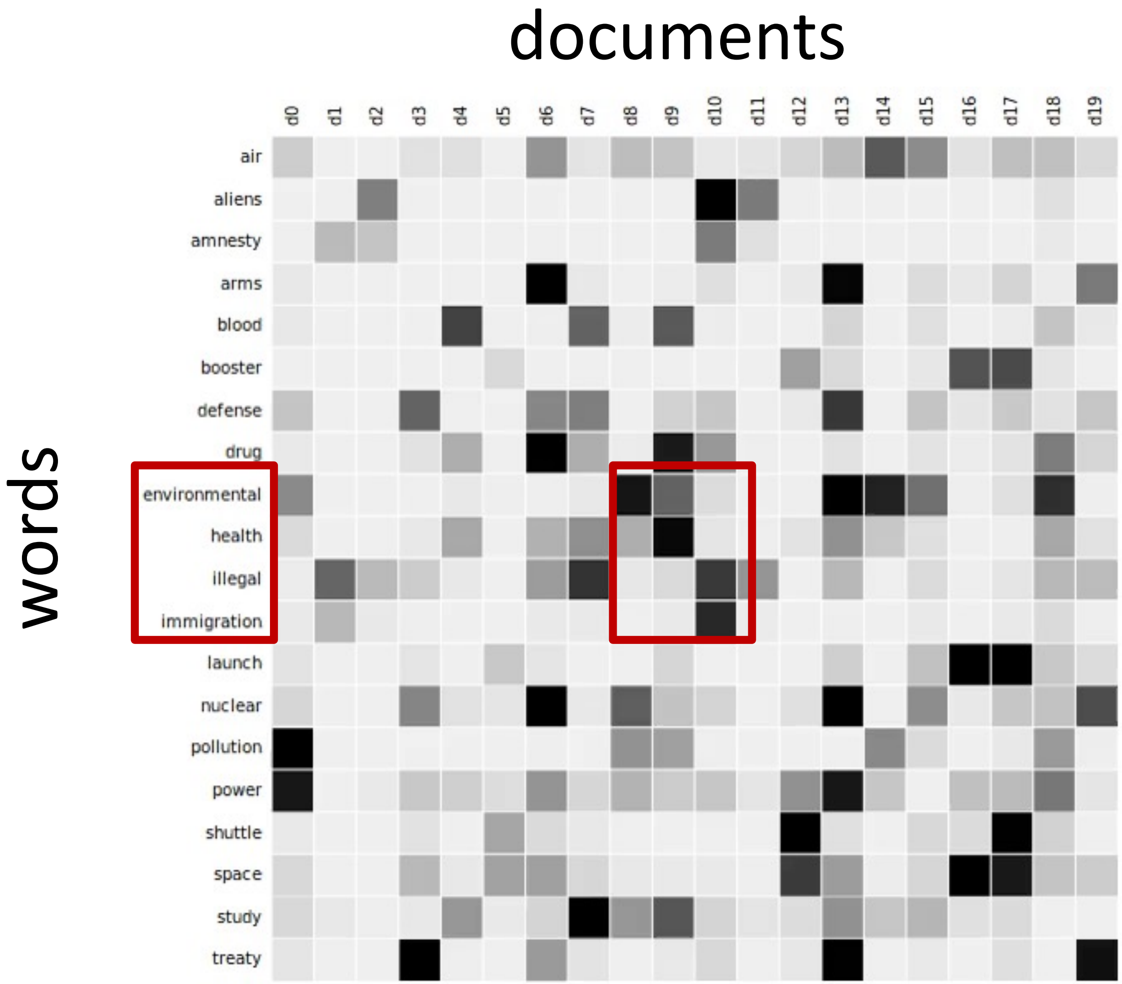
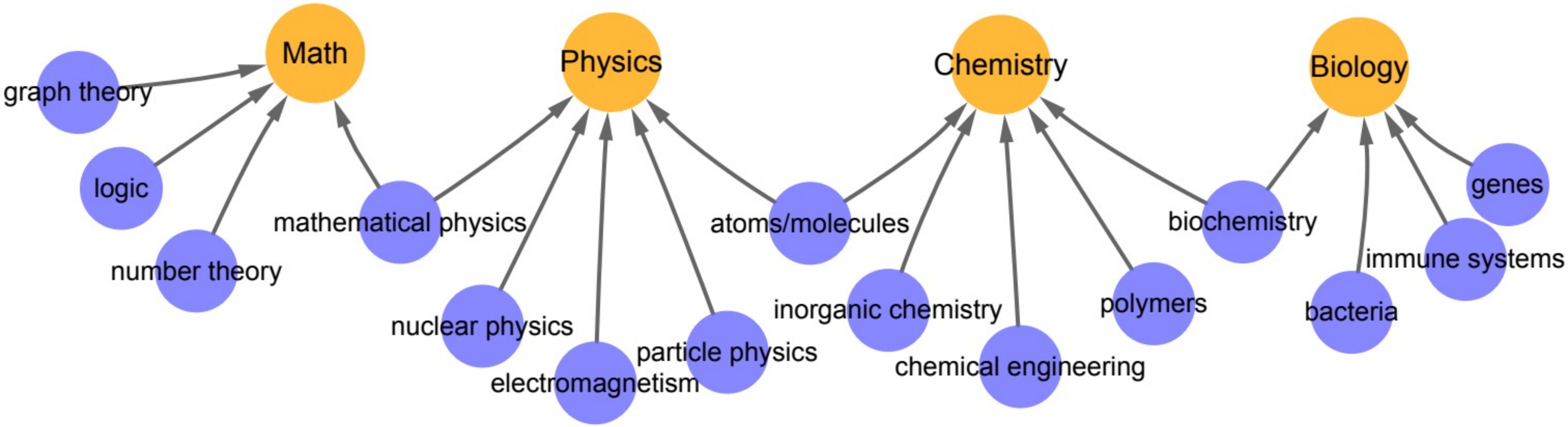


Image credit: https://commons.wikimedia.org/wiki/File:Topic_detection_in_a_document-word_matrix.gif

Clustering Words with Similar Meanings (Hierarchically)



	logic deductive propositional semantics	graph subgraph bipartite vertex	boson massless particle higgs	polyester polypropylene resins epoxy	acids amino biosynthesis peptide
tag	*logic*	*graph theory*	*particle physics*	*polymer*	*biochemistry*

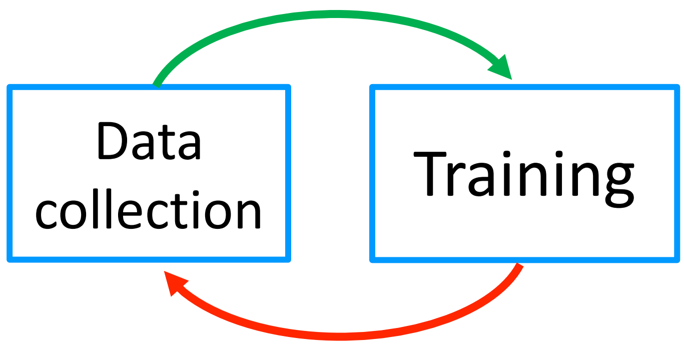
[Arora-Ge-Liang-M.-Risteski, TACL'17,18]

Reinforcement Learning

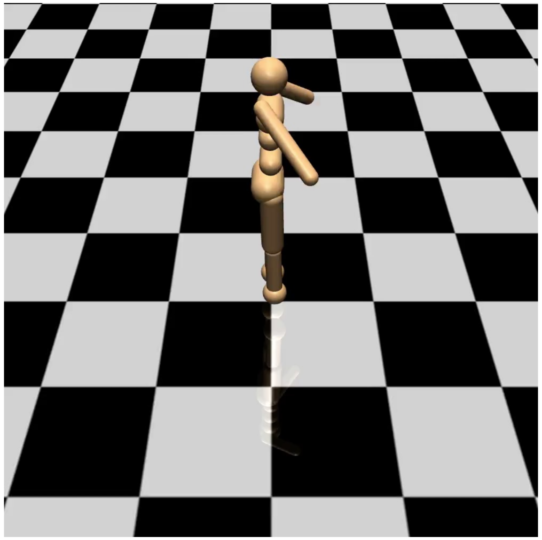
Reinforcement Learning

- The algorithm can collect data interactively

Try the strategy and collect feedbacks



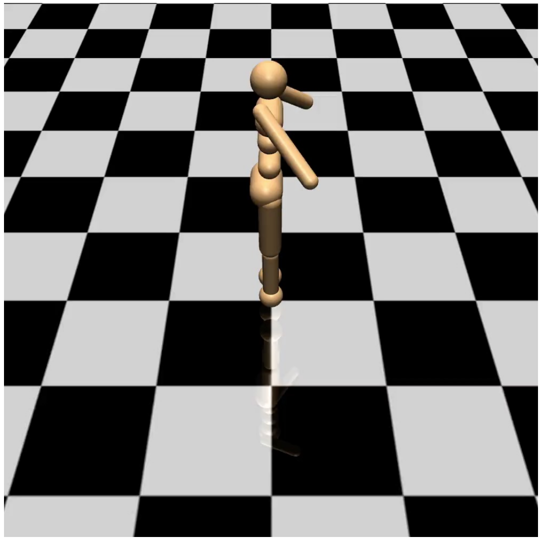
Improve the strategy based on the feedbacks learning to walk to the right



Iteration 10

[Luo-Xu-Li-Tian-Darrell-M.'18]

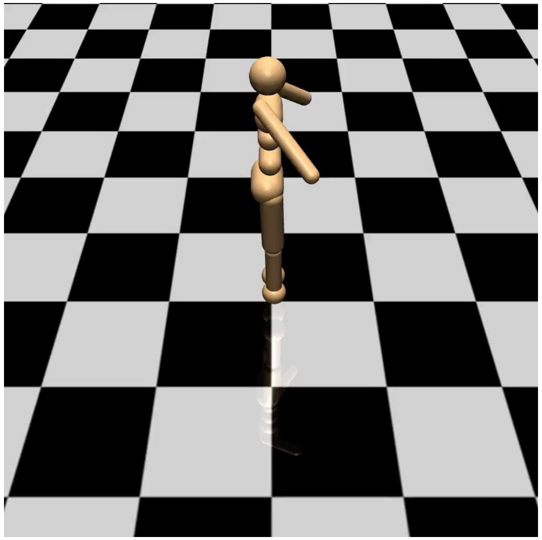
learning to walk to the right



Iteration 20

[Luo-Xu-Li-Tian-Darrell-M.'18]

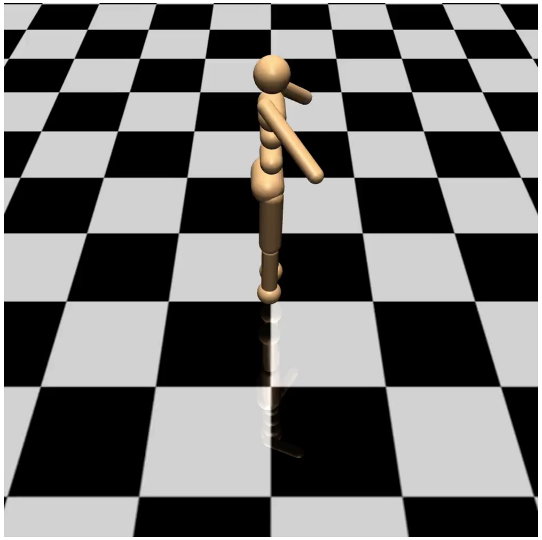
learning to walk to the right



Iteration 80

[Luo-Xu-Li-Tian-Darrell-M.'18]

learning to walk to the right

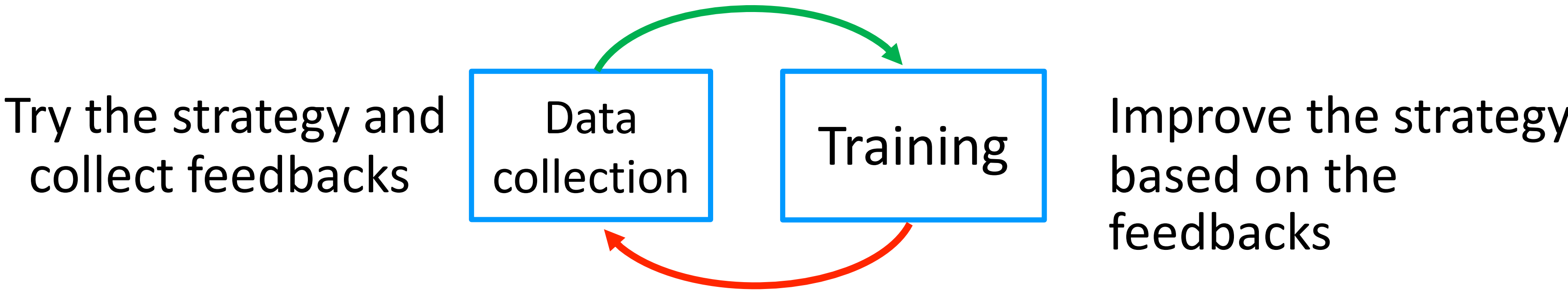


Iteration 210

[Luo-Xu-Li-Tian-Darrell-M.’18]

Reinforcement Learning

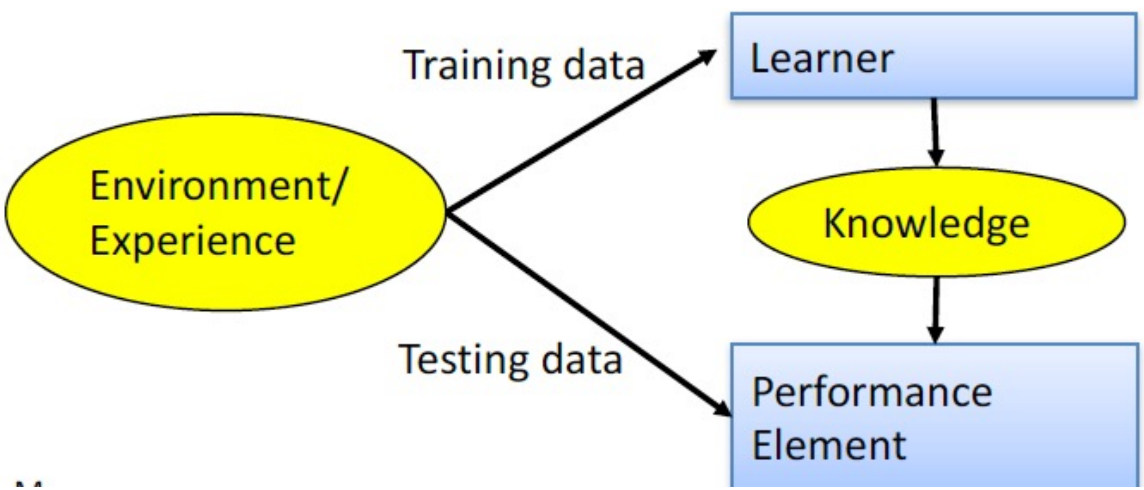
- The algorithm can collect data interactively



Framing a Learning Problem

Designing a Learning System

- Choose the training experience
- Choose exactly what is to be learned
  - i.e. the target function
- Choose how to represent the target function
- Choose a learning algorithm to infer the target function from the experience



Training vs. Test Distribution

- We generally assume that the training and test examples are **independently** drawn from the same overall distribution of data
- If examples are not independent, requires **collective classification**
- If test distribution is different, requires **transfer learning**

Machine Learning in a Nutshell

- Tens of thousands of machine learning algorithms
  - Hundreds new every year

- Every ML algorithm has three components:
  1. Representation
  2. Optimization
  3. Evaluation

Various Function Representations

- Numerical functions
  - Linear regression
  - Neural networks
  - Support vector machines
- Symbolic functions
  - Decision trees
  - Rules in propositional logic
  - Rules in first-order predicate logic
- Instance-based functions
  - Nearest-neighbor
  - Case-based
- Probabilistic Graphical Models
  - Naïve Bayes
  - Bayesian networks
  - Hidden-Markov Models (HMMs)
  - Probabilistic Context Free Grammars (PCFGs)
  - Markov networks

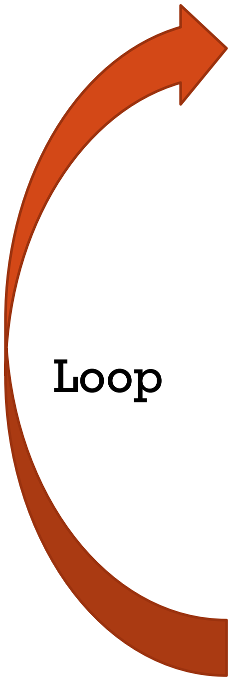
Search/Optimization Algorithms

- Gradient descent
  - Perceptron
  - Backpropagation
- Dynamic Programming
  - MM Learning
  - PCFGs Learning
- Divide and Conquer
  - Decision tree induction
  - Rule learning
- Evolutionary Computation
  - Genetic Algorithms (GAs)
  - Genetic Programming (GP)
  - Neuro-evolution

Evaluation

- Accuracy
- Precision and recall
- Squared error
- Likelihood
- Posterior probability
- Cost / Utility
- Margin
- Entropy
- K-L divergence
- etc.

Machine Learning in Practice



- Understand domain, prior knowledge, and goals
- Data integration, selection, cleaning, pre-processing, etc.
- Learn models
- Interpret results
- Consolidate and deploy discovered knowledge

Lessons Learned about Learning

- Learning can be viewed as using direct or indirect experience to approximate a chosen target function.

- Function approximation can be viewed as a search through a space of hypotheses (representations of functions) for one that best fits a set of training data.

- Different learning methods assume different hypothesis spaces (representation languages) and/or employ different search techniques.

Questions?

Thank you!