Applied Machine Learning

Introduction

Computer Science, Fall 2022
Instructor: Xuhong Zhang

Agenda

- Introduction: what this course is about
- Administrative: resources, grading policy
- Machine Learning Set Up

Agenda

- Introduction: what this course is about
- Administrative: resources, grading policy
- Machine Learning Set Up

What is this course about ?

- Basic theory and practical implementation of machine learning algorithms for real-world applications.
- Topics include data processing and mining, basic machine learning models, advanced machine learning models, model evaluation, generalization

What is Machine Learning ?

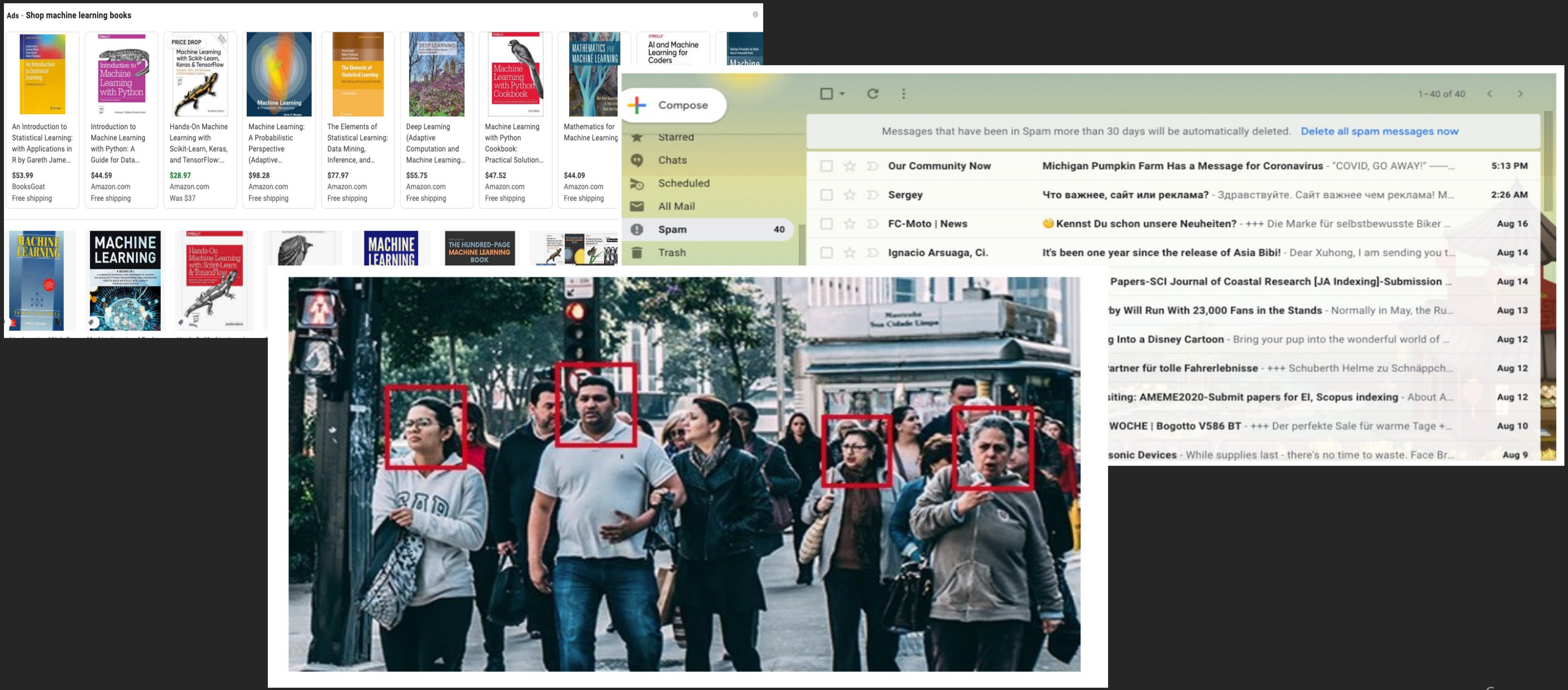
- “Learning is any process by which a system improves performance from experience.”

  - Herbert Simon

- “Machine learning is the study of computer algorithms that improve automatically through experience and by the use of data. It is seen as a part of artificial intelligence.”

  - Wikipedia

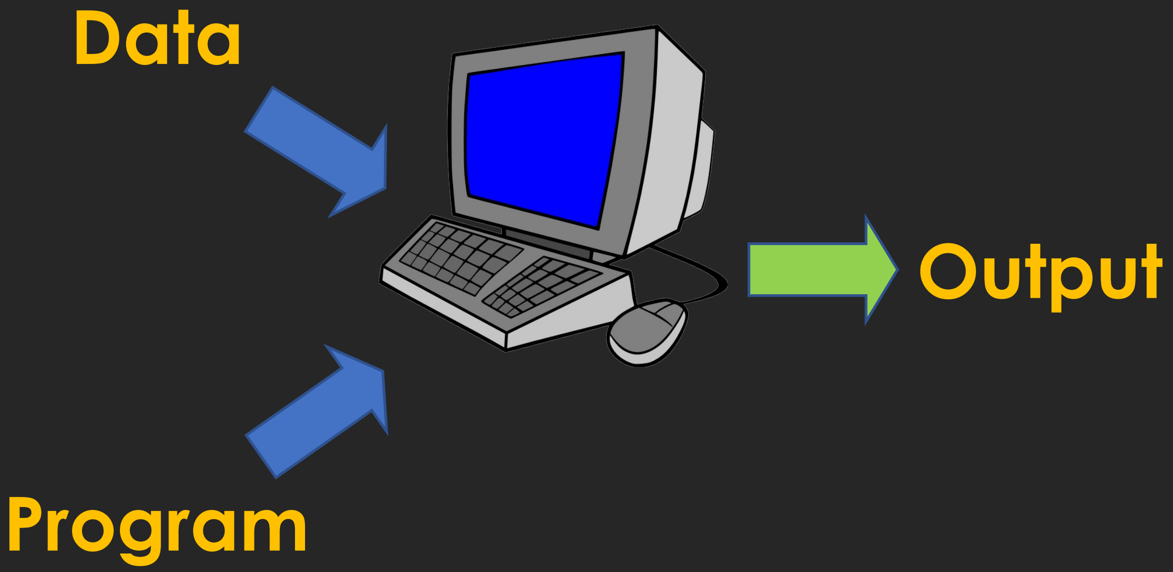
Machine Learning is Everywhere



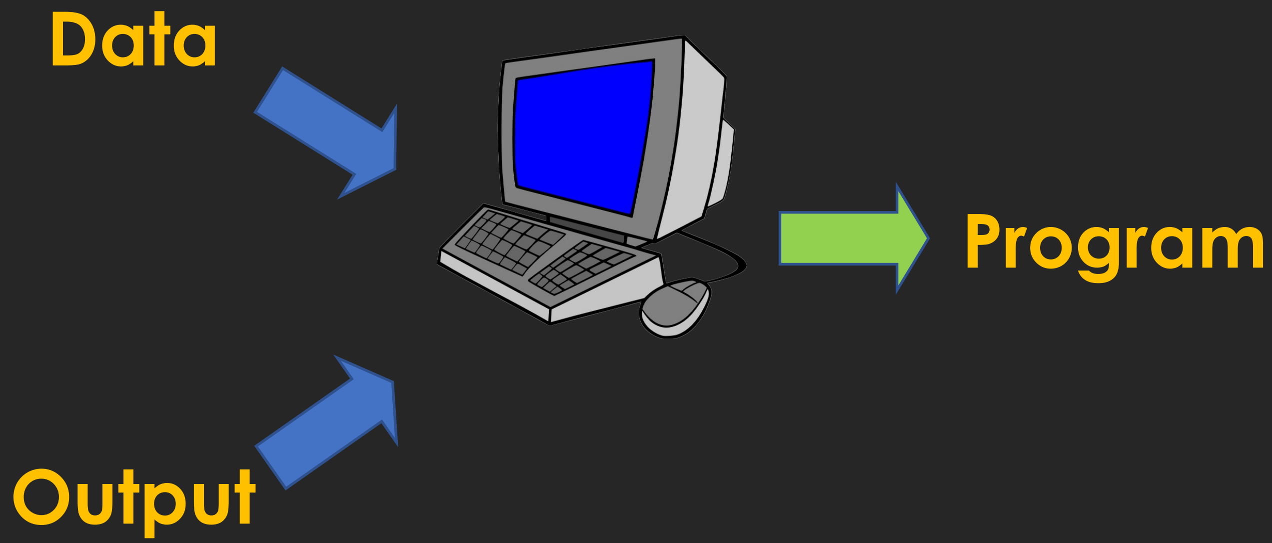
More examples

- Pattern Recognition
  - Handwritten or spoken words
  - Medical imaging analysis
- Pattern Generation
  - Generating images or motion sequences
- Recognizing anomalies
  - Unusual credit card transactions
  - Unusual patterns of sensor readings of automatic driving
- Prediction
  - Future stock prices or housing prices

Traditional CS vs. Machine Learning

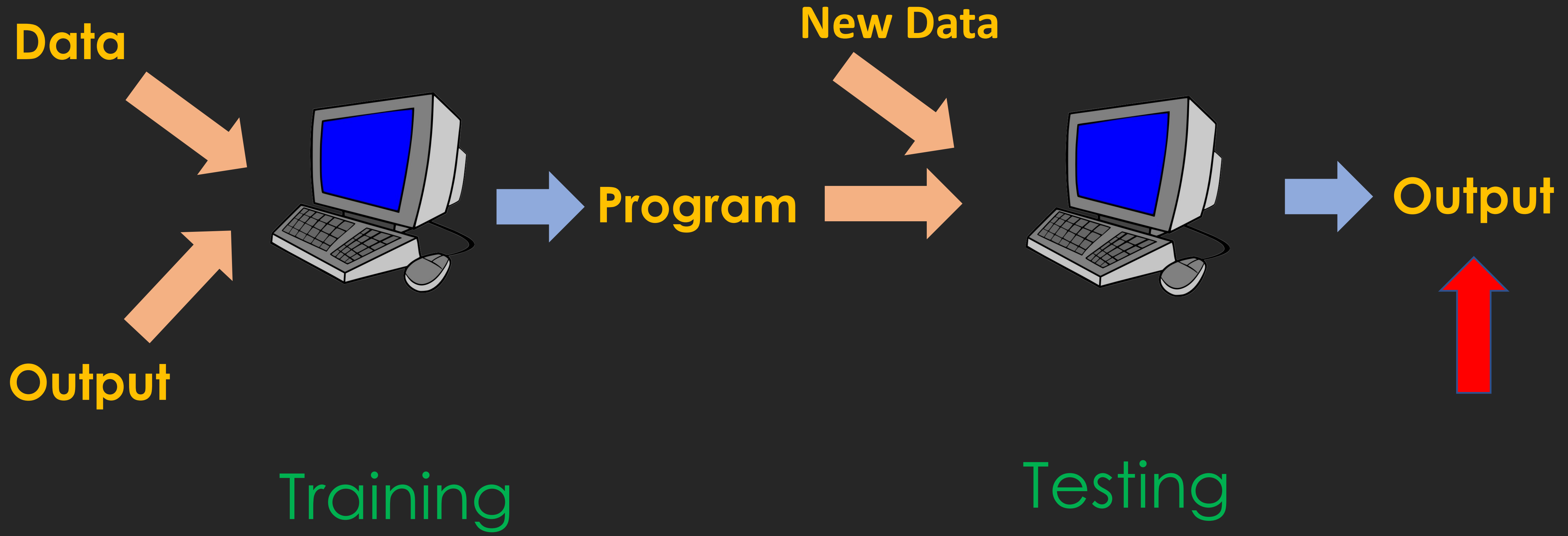


Traditional CS



Machine Learning

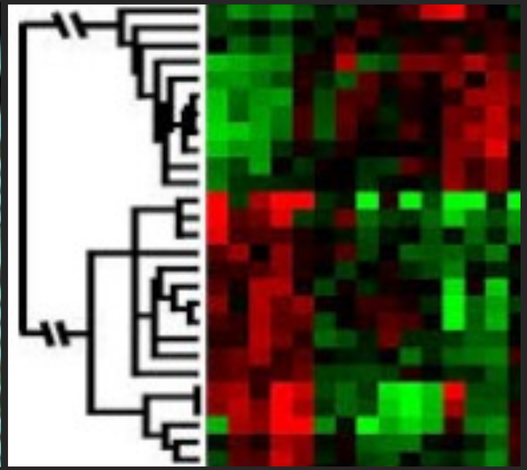
Machine Learning



When is Machine Learning Needed ?

- Human expertise does not exist (navigating on Mars)
- It's hard to explain human's expertise (speech recognition, citation networks)
- Models must be customized (precision medicine)
- Models are based on huge amounts of data (genomics study)



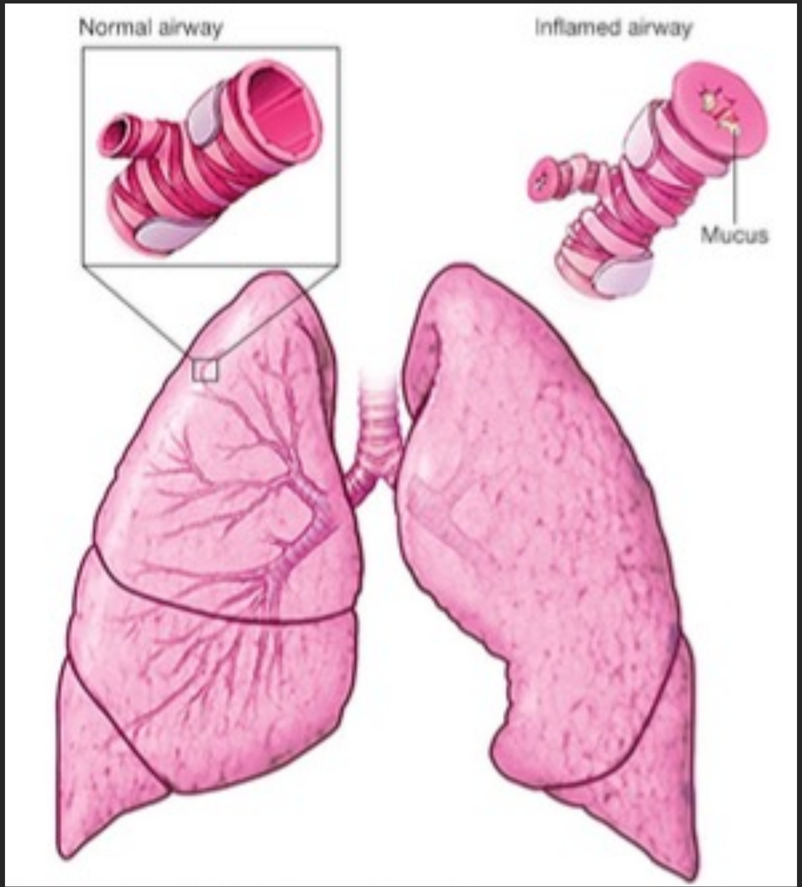
Use with Caution !

"panda"
57.7% confidence

"gibbon"
99.3% confidence

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. and Fergus, R., 2013. Intriguing properties of neural networks.

Use with Caution !



"has Asthma(x)⇒Lower risk(x)"

Trade-off between interpretability and accuracy

Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission Caruana et. al. 2015

Machine Learning vs Statistics

- Machine Learning
  - Data First / Data Driven
  - Prediction Emphasis
- Statistics
  - Model First / Model Driven
  - Inference Emphasis

About this course

- The goal of this course it to help you understand the fundamentals of machine learning
- Provide foundations of machine learning
  - Basic mathematical derivation and implementation
- Cover practical applications of machine learning
  - Use machine learning algorithms for your problems/applications of interest

What this course is *not*

- Focused only on ***applied*** machine learning
  - we are interested in the basic mathematical interpretation of the algorithms
  - be prepared for “some math”
- Focused only on ***theoretical*** machine learning
  - we are also interested in applying algorithms to datasets to get hands-on experience with the algorithms
  - be prepared for some programming-heavy assignments

Agenda

- Introduction: what this course is about
- Administrative: resources, grading policy
- Machine Learning Set Up

Administrative introduction

- Instructor: Xuhong Zhang (zhangxuh@iu.edu)
- Time: MW
- Location: two locations
- Office: Luddy hall, 3012
- Office Hour: Friday Afternoon 4-6pm or by appointment
- TA: Keith Xiao

Logistics

- Final Grade
  - Homework : 40 % (4 homeworks)
  - In-class Programming : 15%
  - Bi-weekly In-Class Quiz : 10 % (First quiz starts Sep 7th—the fifth lecture)
  - Midterm Exam : 15%
  - Final Exam : 20 %

- Homework late submission policy (see canvas)

Logistics

- Lecture slides, course notes, sample code will be provided (Canvas)
- Instruction for submission will be provided (Canvas)
- Discussion/Q&A tools

Logistics

- Form your study group early on !
- For homework, one submission per group (We have a tool to detect code copying and plagiarism)
- Please start on homework early (Warning: cramming does not work !)
- We only accept scripts in ***Python***

- **PLEASE READ THE SYLLABAS !**

Code Copying and Plagiarism

- Copied code will get 0 point for all involved
- Homework will be checked for plagiarism
- Copying from course code is fine
- Copying from online sources (stack overflow, tutorials, etc.) is fine but you have to refer to the source
- You also have to mention your peers if you discuss outside your group
- Plagiarism is not allowed throughout the entire semester

Books

- Pyth Norvig and Russell, *Artificial Intelligence: A Modern Approach*.
- Goodfellow, Bengio and Courville, *Deep Learning*.
- Trevor Hastie, Robert Tibshirani and Jerome Friedman, *The elements of Statistical Learning (Data Mining, Inference, and Prediction)*.

Tentative Schedule

- Holidays (No class)
  - Labor Day: Sep 5
  - Fall Break: Oct 14 – Oct 16
  - Thanksgiving Break: Nov 20 – Nov 27

Previous Projects (20' Fall)

- Hashtag Generator
- DNA and Protein Embedding
- Image Classification with Insufficient Samples
- Food Item Recognition using CNN
- Real Time Object Recognition
- A Stacking Method for Cancer Survival Classification
- Predicting the Recovery Time of Hospitalized Covid-19 Patients

Agenda

- Introduction: what this course is about
- Administrative: resources, grading policy
- Machine Learning Set Up

Defining the Learning Task

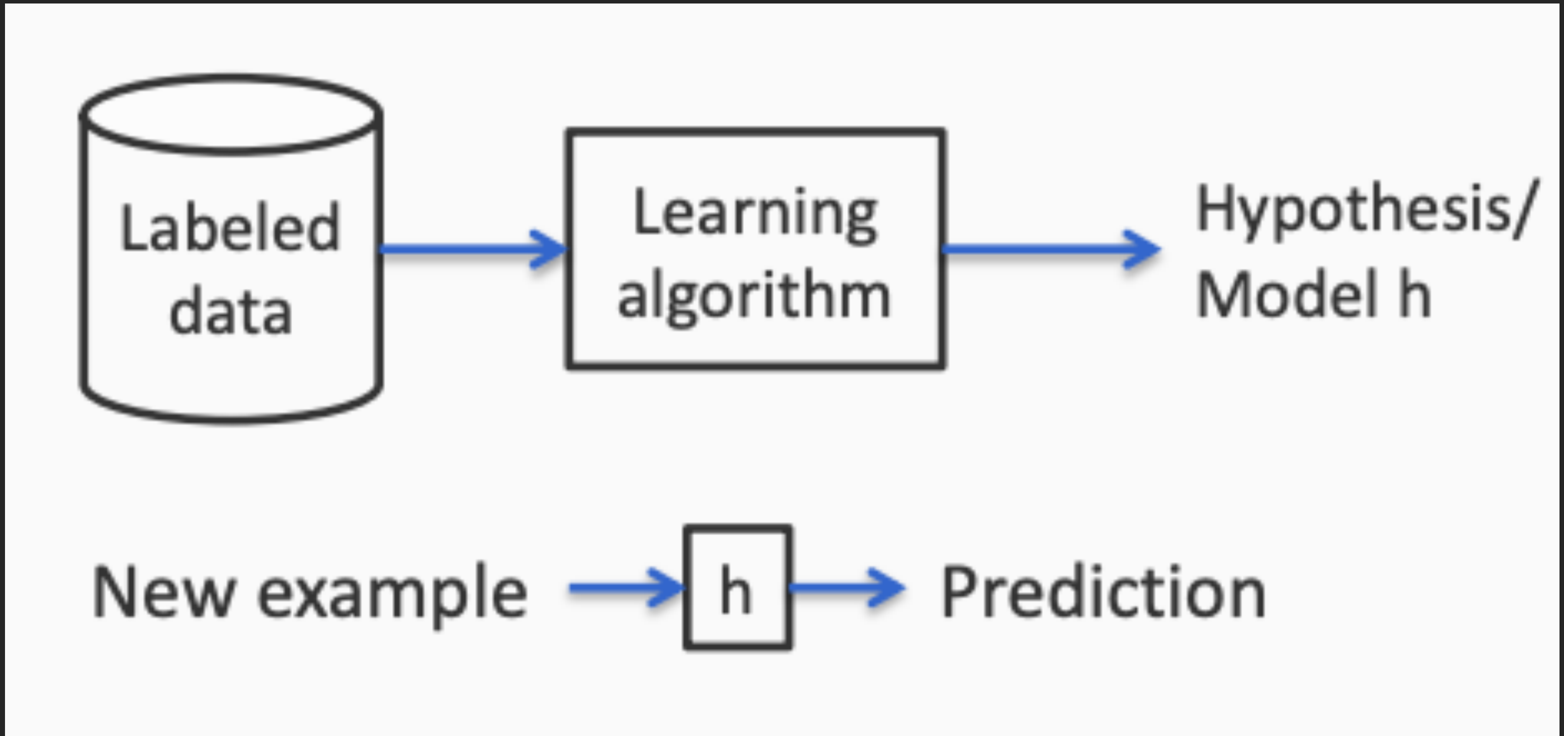
- Improve on task ***T***, with respect to performance metric ***P***, based on experience ***E***
  - ***T***: Categorize email messages as spam or legitimate
  - ***P***: Percentage of email messages correctly classified
  - ***E***: Database of emails, some with human-given labels

  - ***T***: Recognizing hand-written words
  - ***P***: Percentage of words correctly classified
  - ***E***: Database of human-labeled images of handwritten words

Types of Machine Learning

- *Supervised (inductive) Learning*
  - Given: training data + desired outputs (labels)
- *Unsupervised Learning*
  - Given: training data (without desired outputs)
- *Semi-supervised Learning*
  - Given: training data + a few desired outputs
- *Reinforcement Learning*
  - Rewards from sequence of actions

Supervised Learning



Supervised Learning

- Given input-output pairs, learn a function $f(x)$
  - $D = \{(x_i, y_i)_{i=1}^{N},\ (x_i, y_i) \propto p(x, y)\}, iid$
  - $f(x_i) \approx\ y_i$
  - $x_i \in\ \mathbb{R}^d$
  - y_i : categorical---classification
  - y_i : real valued---regression

Supervised Learning

- **Classification**

$$f(x_i) \approx y_i, y \in \{1, \dots, C\}$$

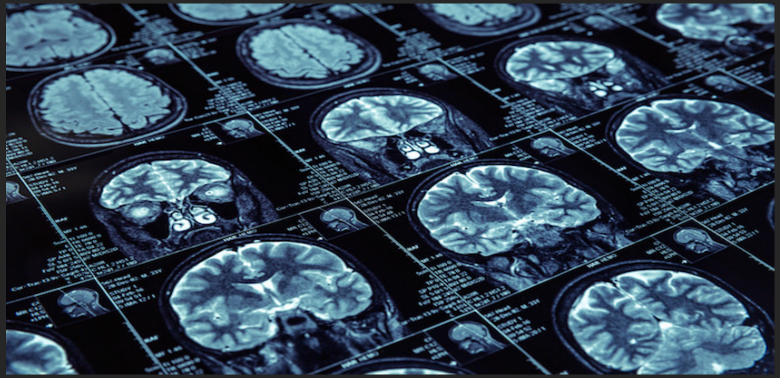
  - $C = 2$: binary classification
  - $C > 2$: multiclass classification

- **Regression**

$f(x_i) \approx y_i$, where y is continuous

Supervised Learning Examples

- Medical Image Learning

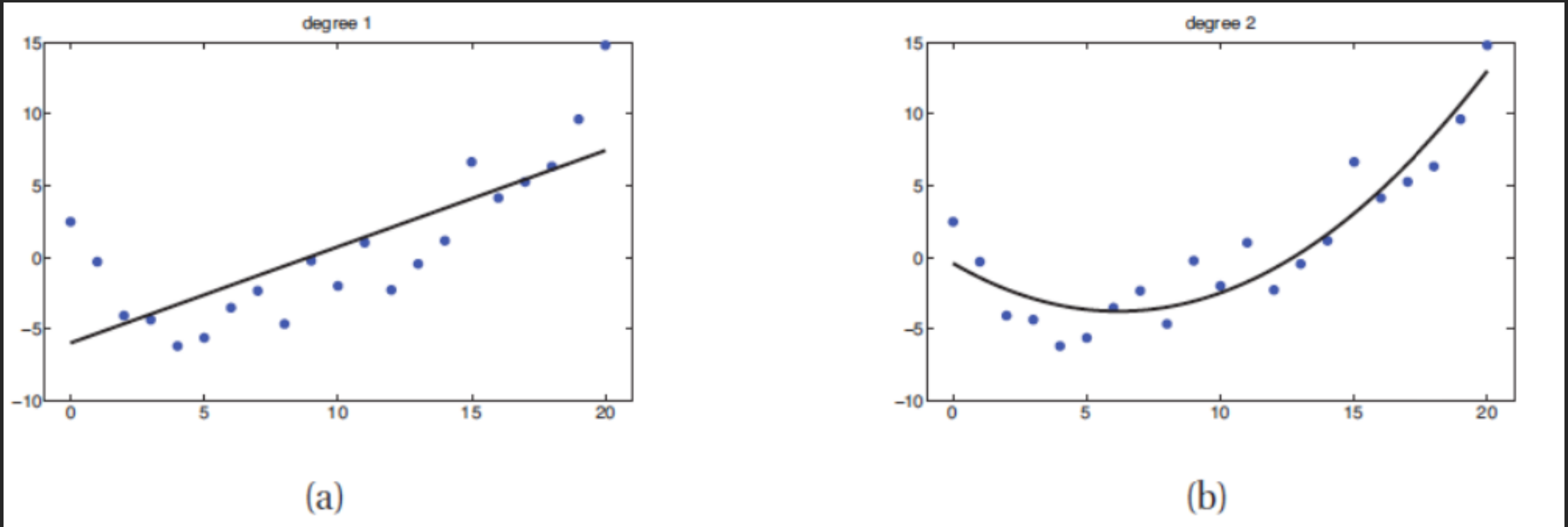


- Iris Type Prediction

(a) (b) (c)

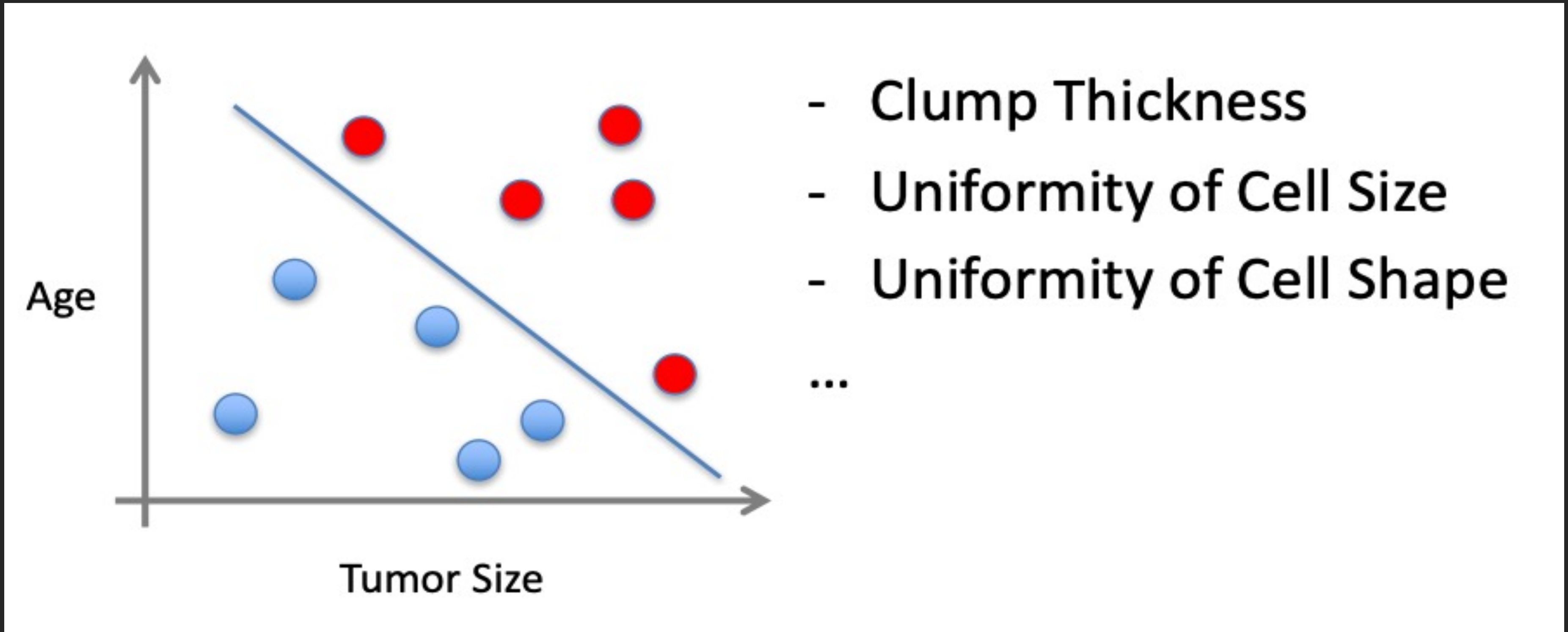
Supervised Learning Examples

- Regression



Supervised Learning Examples

- x can be multi-dimensional
  - Each dimension corresponds an attribute/feature/covariate

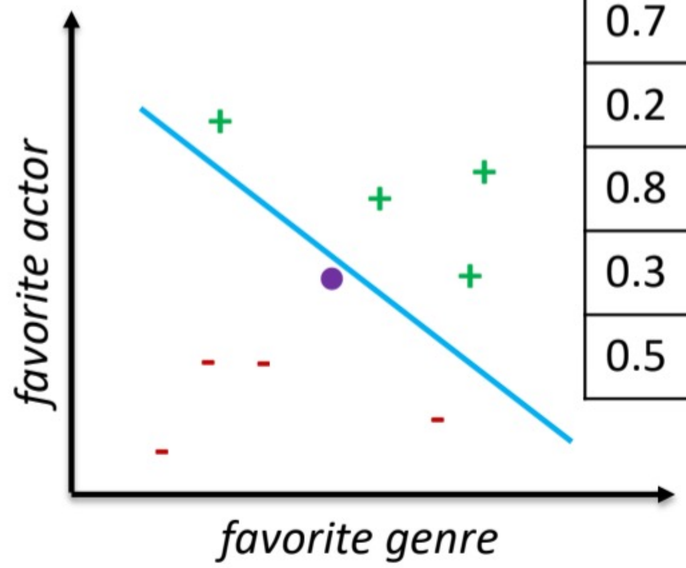


Based on example by Andrew Ng

Supervised Learning Examples

- Problem: predict whether a target user likes a target movie
- Data:
  - Features: percentage of your favorite genre scenes, percentage of scenes where your favorite actor appears
  - Labels: like/dislike

Goal: Learn a linear boundary



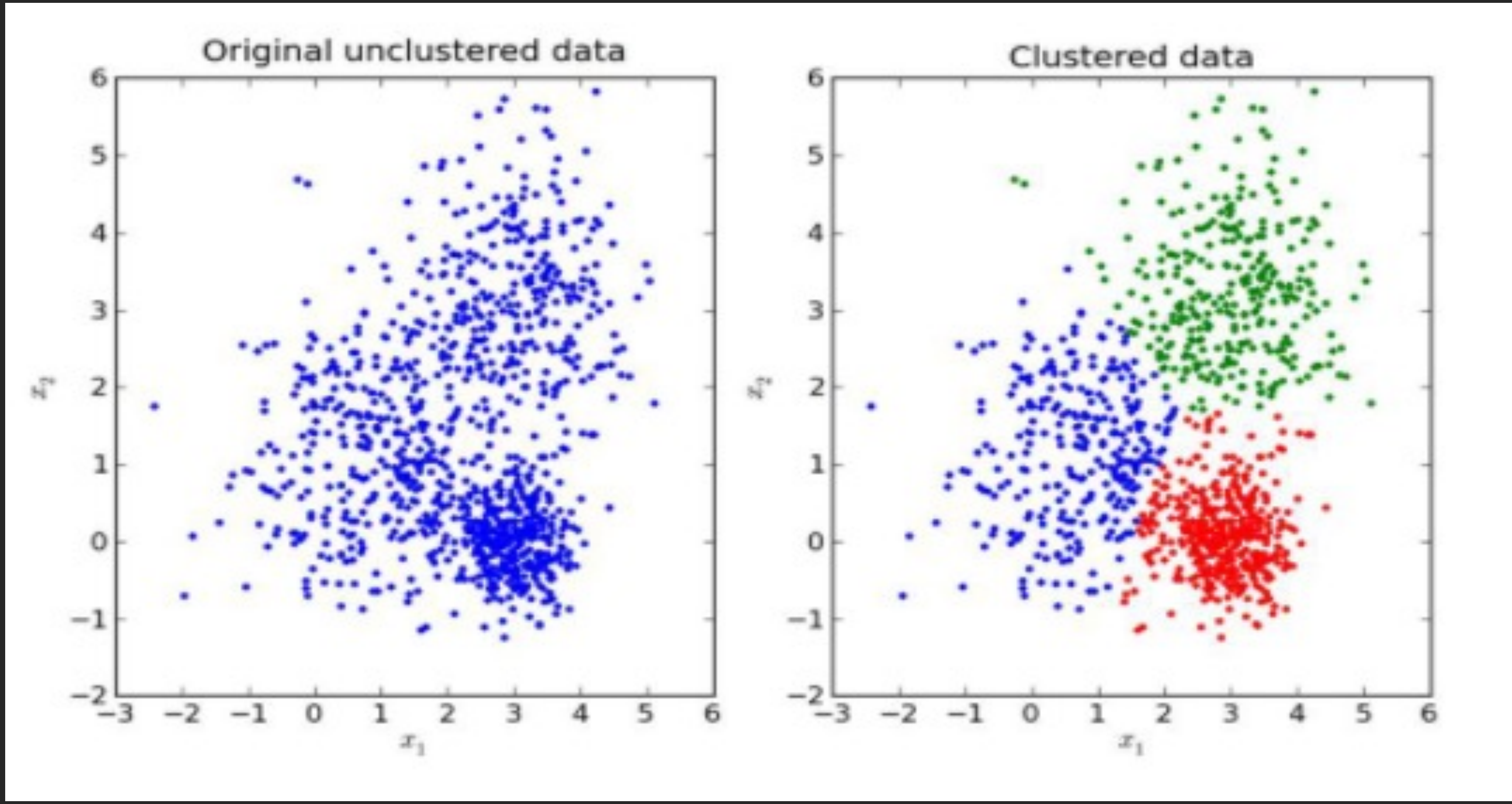
0.7	0.7	Like
0.2	0.2	Dislike
0.8	0.2	Dislike
0.3	0.9	Like
0.5	0.4	?

Unsupervised Learning

- Input Data
  - $D = \{x_i\}_{i=1}^{N},\, x_i \propto p(x),\ iid$
  - Learn about P

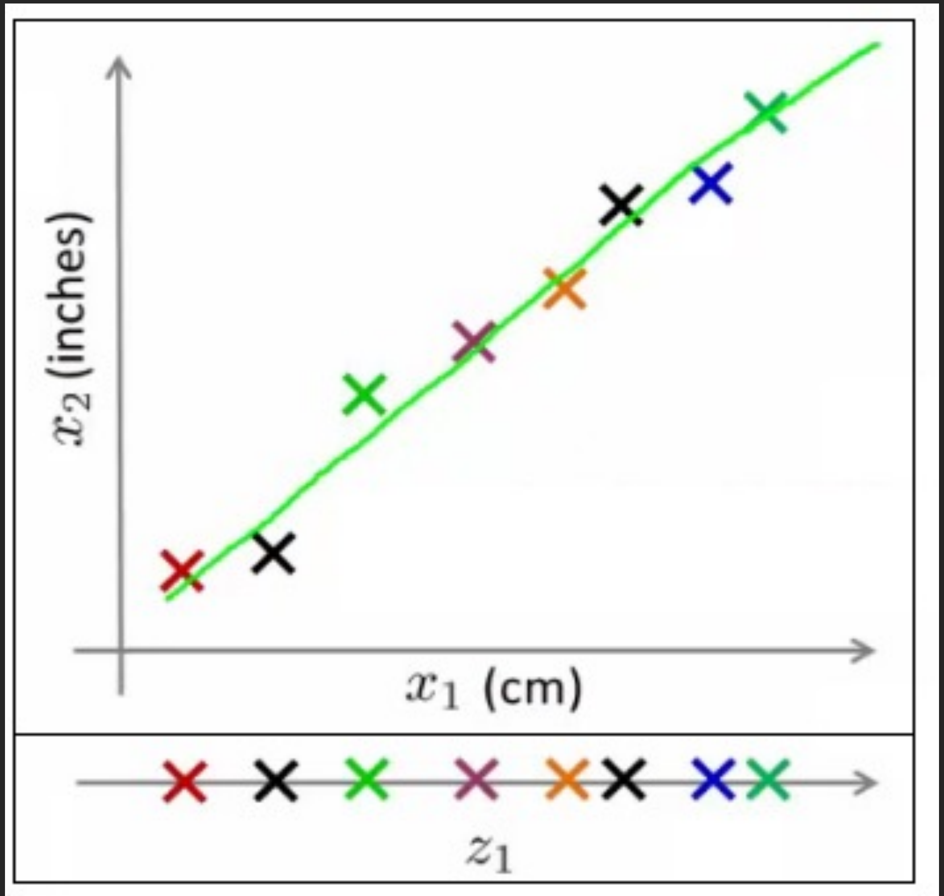
Unsupervised Learning Examples

- Clustering



Unsupervised Learning Examples

- Dimensionality Reduction



Unsupervised Learning Examples

- Topic Modeling

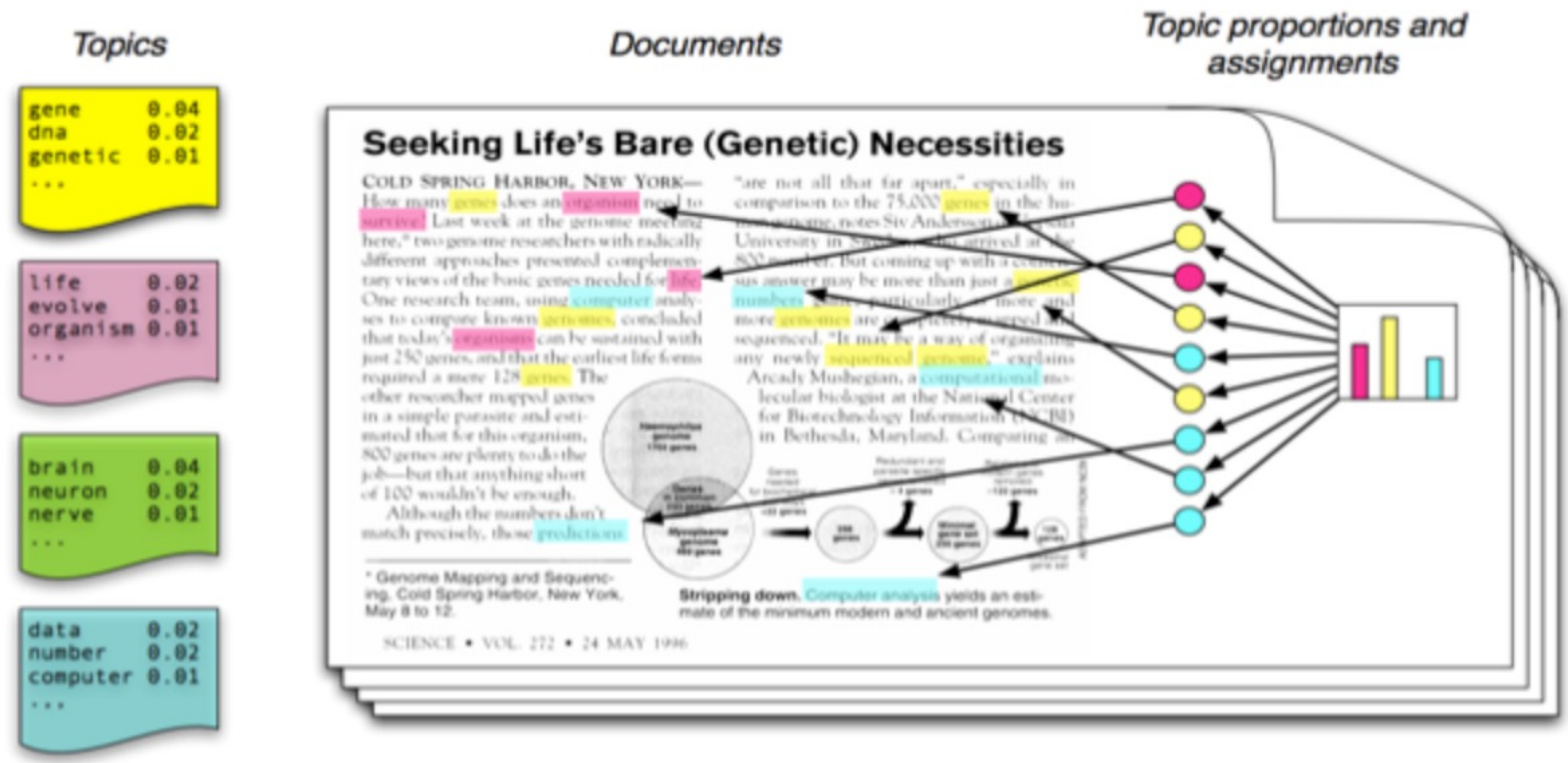
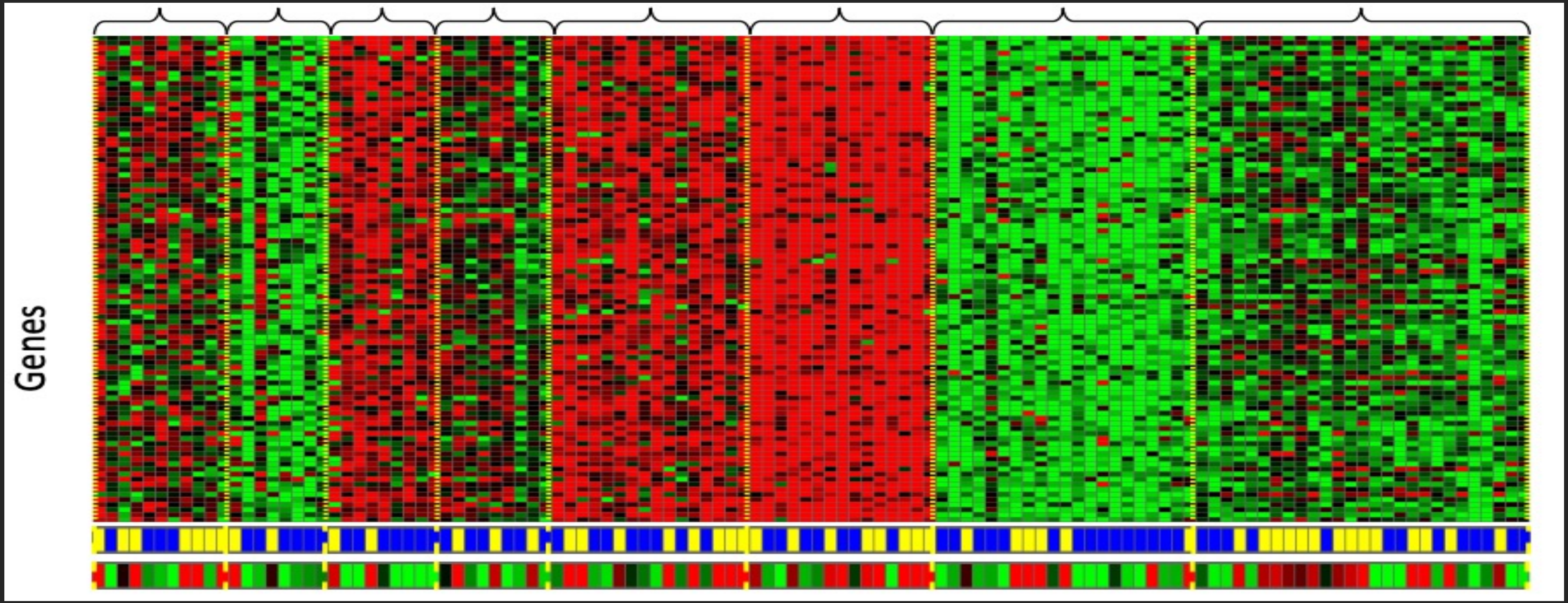


Figure source: Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77-84.

Unsupervised Learning Examples

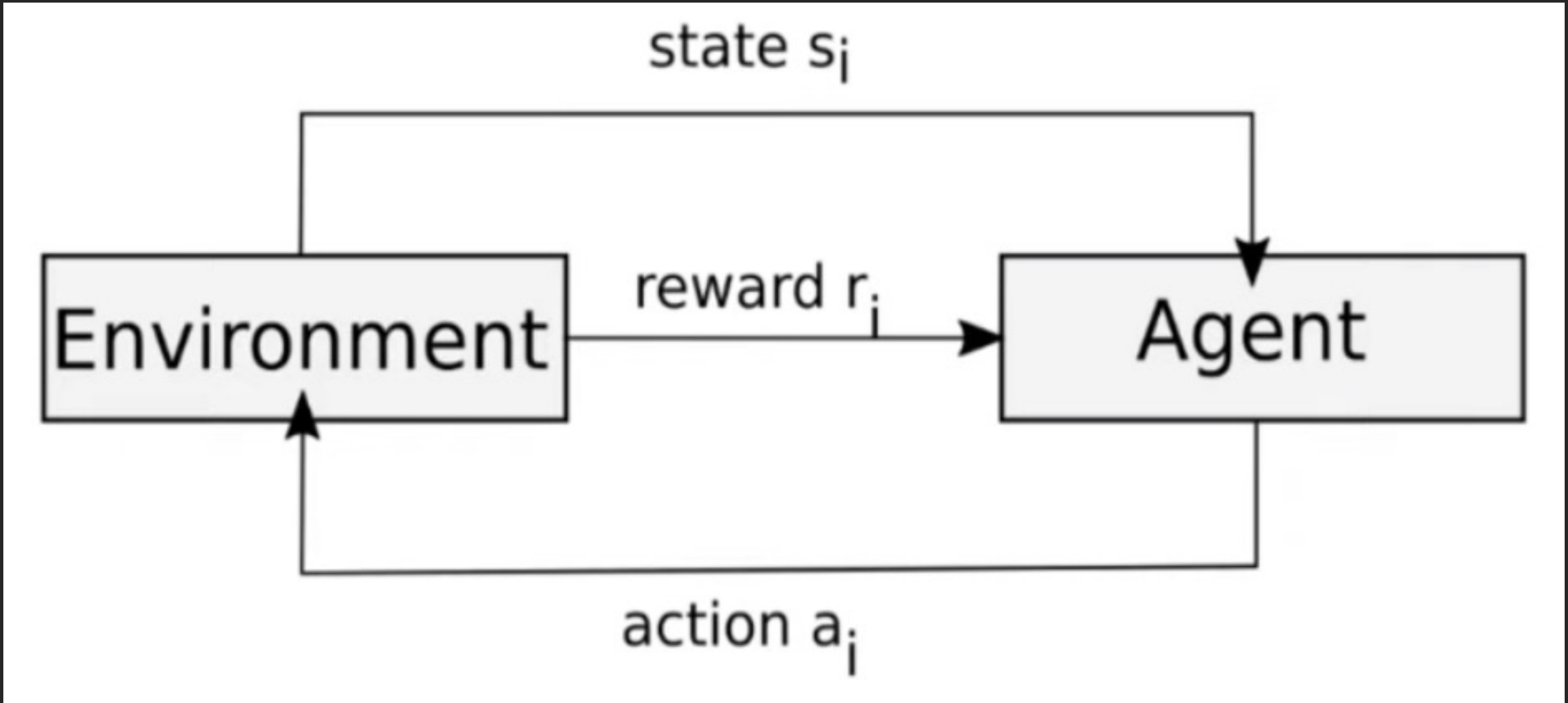
- Genomics application: group individuals by genetic similarity



Individuals

Source: Daphne Koller

Reinforcement Learning



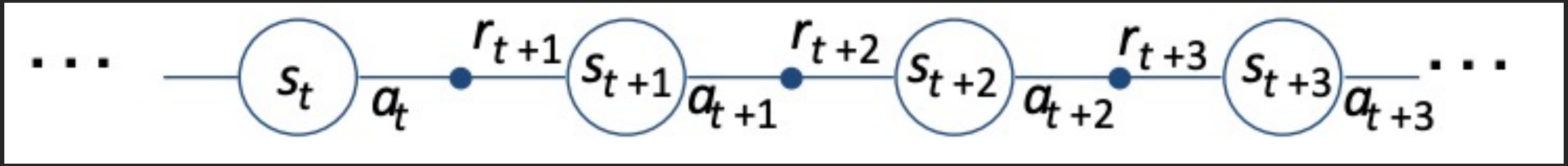
Reinforcement Learning

- Given a sequence of stats and actions with (delayed) rewards, output a policy
  - Policy is a mapping from states to actions that tells you what to do in a given state

- Examples
  - Game playing
  - Robot in a maze

Reinforcement Learning

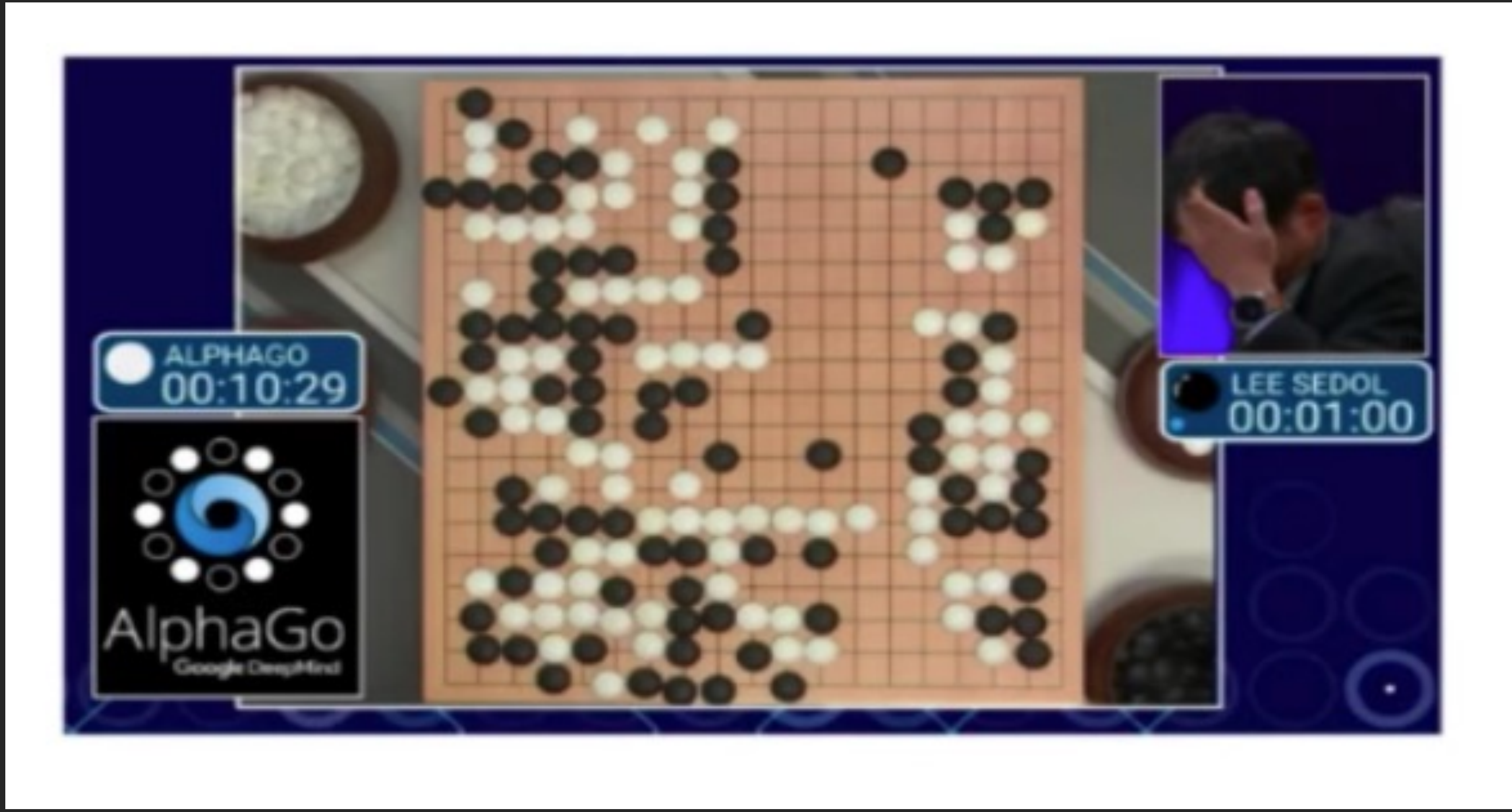
- Agent and environment interact a discrete time steps: $t = 0, 1, \ldots, K$
  - Agent observes state at step t: $S_t \in S$
  - Produces action at step t: $a_t \in A(S_t)$
  - Get resulting reward: $r_{t+1} \in \Re$
  - And resulting next state: S_{t+1}



Slide credit: Sutton & Barto

Reinforcement Learning Examples

- Alpha Go



Reinforcement Learning Examples

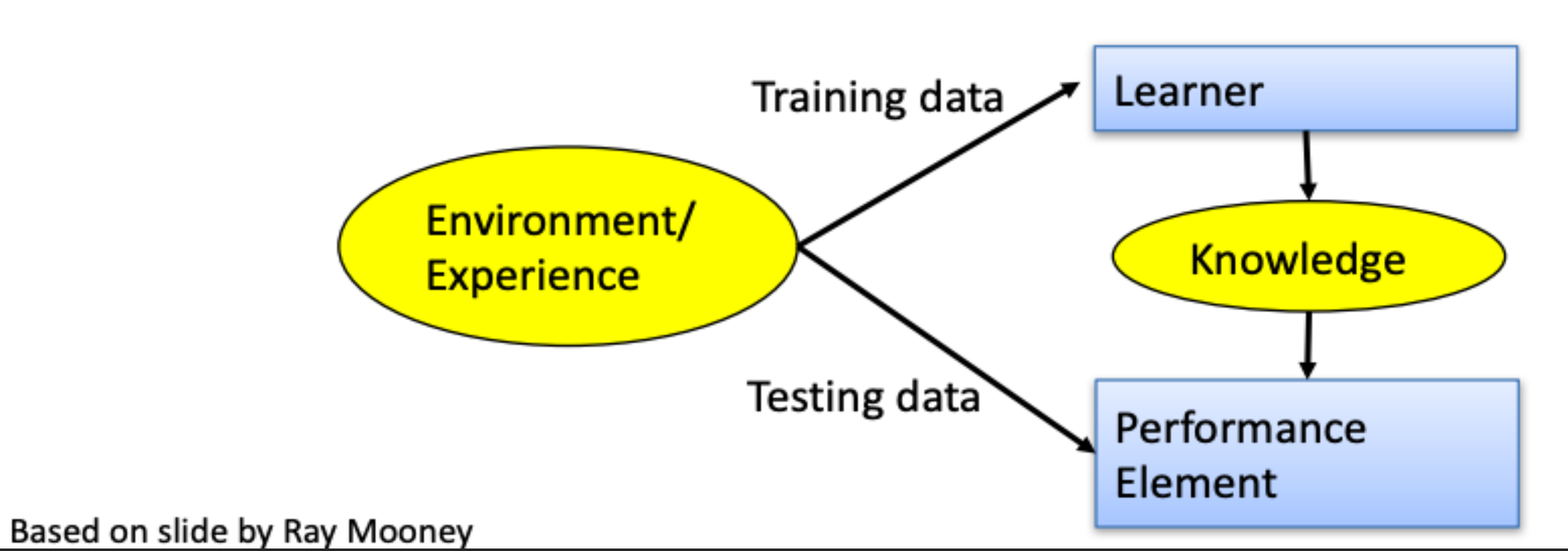
- Self-Driving Car

Other Types

- Semi-supervised
- Active Learning
- Forecasting
- …

How to frame a learning task

- 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



Training vs. Test Distribution

- We generally assume that the training and test examples are independently drawn from the *same* overall distribution of data
  - We call this “i.i.d” which stands for “independent and identically distributed”
- 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
  - *Representation*
  - *Optimization*
  - *Evaluation*

Slide Credit: Pedro Domingos

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*
  - *Markov networks*

Slide credit: Ray Mooney

Various Search/Optimization Algorithms

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

Slide credit: Ray Mooney

Machine Learning in Practice

Loop

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

Slide credit: Pedro Domingos

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.

History of Machine Learning

✓**1960s**

- ***Neural networks: Perceptron***
- Pattern recognition
- Learning in the limit theory

✓**1980s**

- ***Advanced decision tree and rule learning***
- Explanation-based learning (EBL)
- Learning and planning and problem solving
- Utility problem
- Analogy
- Resurgence of neural networks (connectionism, backpropagation)
- Valiant's PAC learning Theory

Slide credit: Ray Mooney

History of Machine Learning

✓1990s

- Data mining
- Adaptive software agents and web applications
- ***Text mining***
- ***Reinforcement Learning (RL)***
- Inductive Logic Programming (ILP)
- ***Ensembles: Bagging, Boosting, and Stacking***
- ***Bayes Net Learning***

Slide credit: Ray Mooney

History of Machine Learning

✓**2000s**

- ***Support vector machines & kernel methods***
- ***Graphical models***
- Statistical relational learning
- ***Transfer learning***
- Sequence labeling
- Collective classification and structured outputs
- Computer Systems Applications (Compilers, Debugging, Graphics, Security)
- E-mail management
- Personalized assistants
- ***Learning in robotics and vision***

Slide credit: Ray Mooney

History of Machine Learning

✓**2010s**

- ***Deep learning systems***
- ***Learning for big data***
- ***Bayesian methods***
- Multi-task & lifelong learning
- Applications to vision, speech, social networks, learning to read, etc.
- …

Slide credit: Ray Mooney

Sidebar: Ethical Considerations

- Privacy
- Fairness and bias
- Benefit vs. Harm
- …

Basic Concepts (1)

- **Parametric** vs. **non-parametric** models
  - Parametric: all the parameters are in finite-dimensional parameter spaces
  - Non-parametric: all the parameters are in infinite-dimensional parameter spaces. The model structure is not specified a priori but is instead determined from the data.

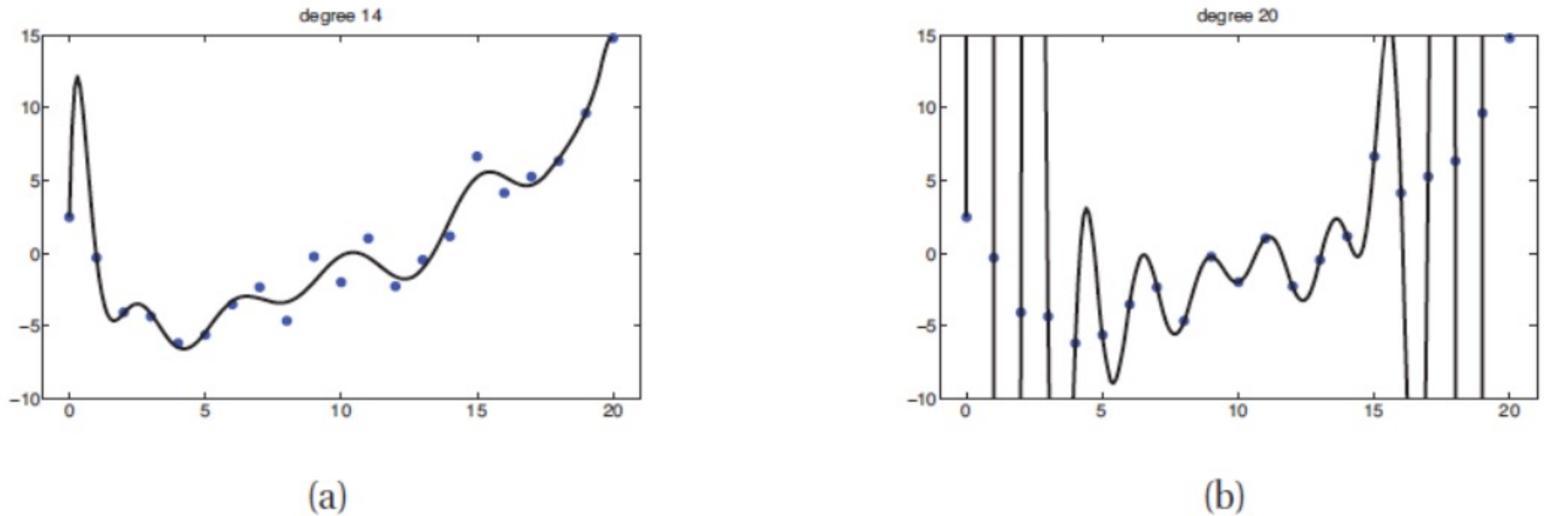
Basic Concepts (1)

- Parametric model examples
  - Exponential family
  - Poisson family
  - …

- Non-parametric model examples
  - K-nearest neighbor
  - Kernel density estimation
  - …

Basic Concepts (2)

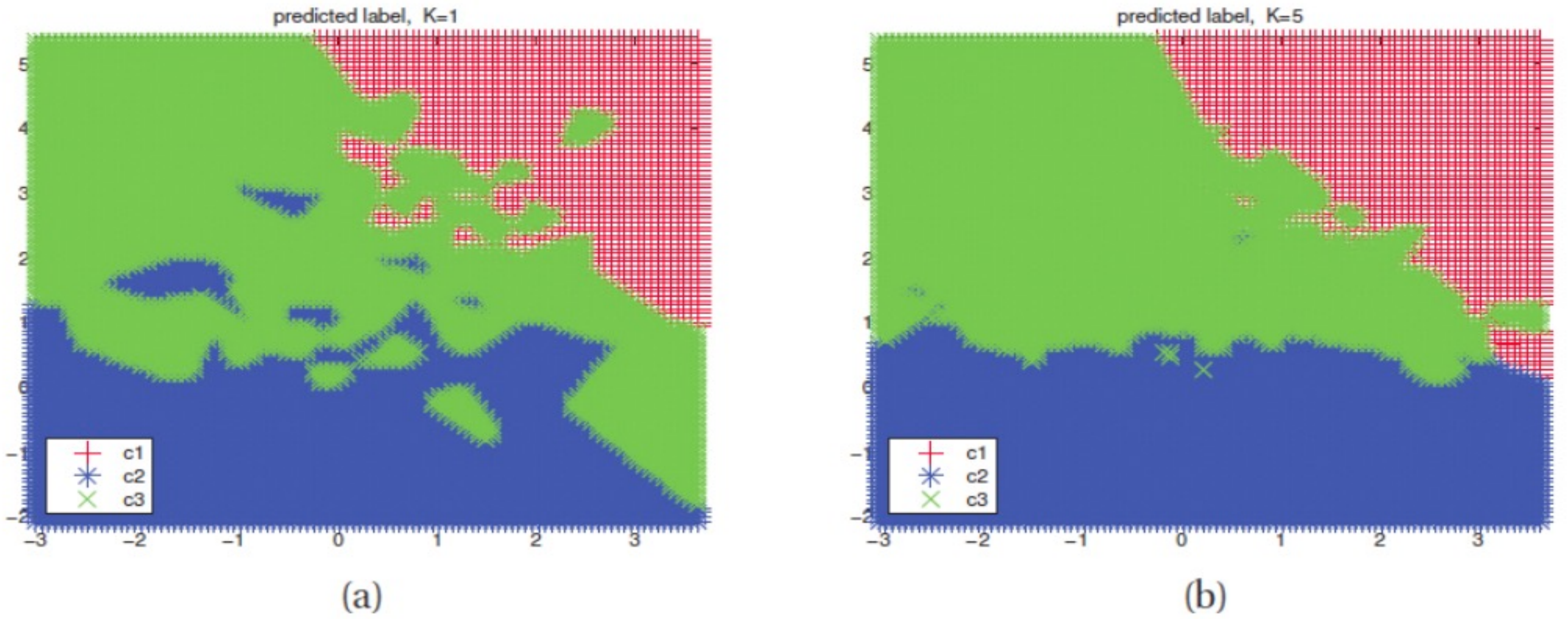
- Overfitting



Polynomial of degrees 14 and 20 fit by least squares to 21 data points. Figure generated by linewfPolyVsDegree in Matlab.

Basic Concepts (2)

- Overfitting



Prediction surface for KNN on the training data. (a) K = 1. (b) K = 5. Figure generated by knnClassifyDemo in Matlab.

Basic Concepts (3)

- Generalization
  - For supervised learning, we not only learn

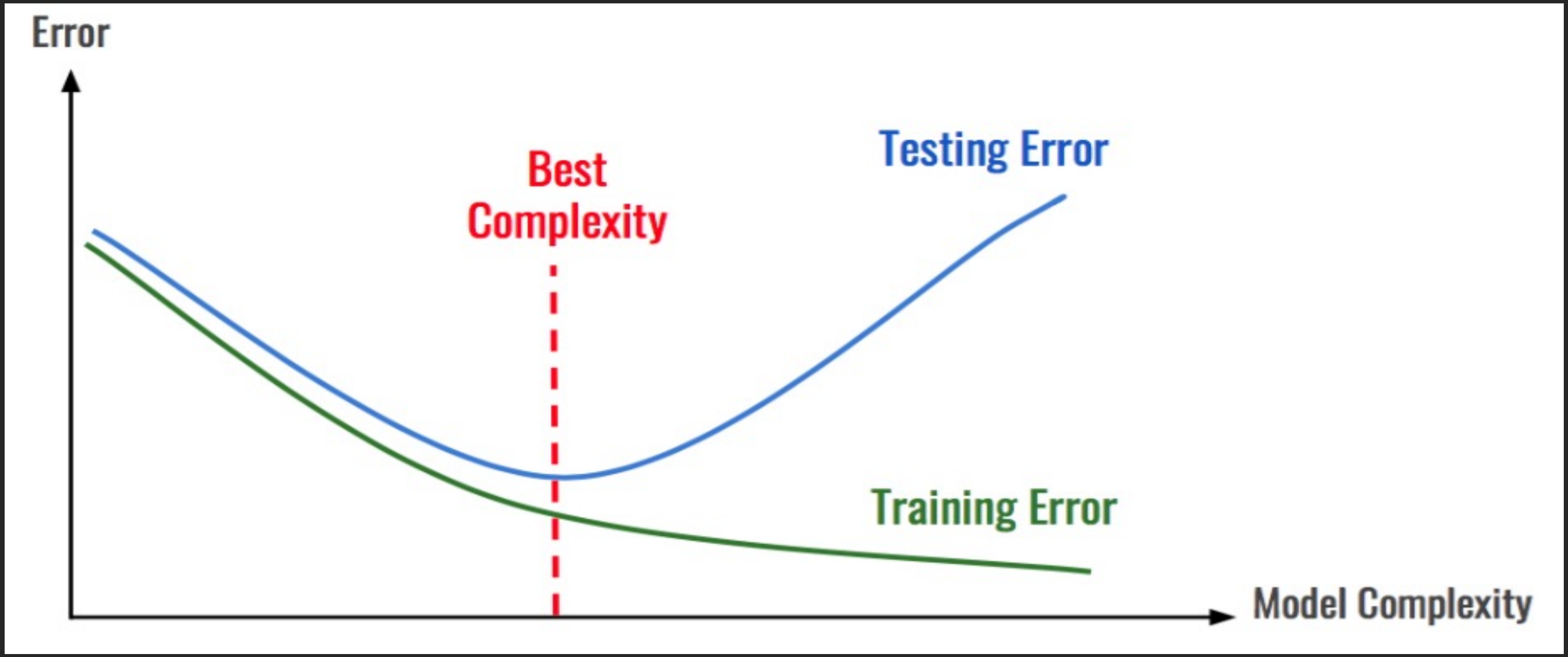
$$f(x_i) \approx y_i$$

  - More important, we want

$$f(x_{new}) \approx y_{true}$$

Basic Concepts (4)

- Model Selection



Knowing Your Goal and Your Data

- What question(s) am I trying to answer? Do I think the data collected can answer that question?
- What is the best way to phrase my questions(s)?
- Have I collected enough data to represent the problem I want to solve?
    - Plot your data !!

Knowing Your Goal and Your Data

- What features of the data did I extract, and will these enable the right predictions?
- How can I measure success in my application?
- Can I interpret the model and the process to someone else?