Introduction to Computational Data Analysis

CX4240, 2019 Spring

Georgia Tech

Lecture 01. Course Overview

Chao Zhang

College of Computing

Georgia Tech

These slides are partially based on slides from Alex Smola, Barnabás Póczos, Ray Mooney, Le Song and others.
Acknowledgments to those who made their course materials freely available online.

Logistics

Time: Monday & Wednesday 4:30-5:45pm

Location: Klaus 1447

Instructor: Chao Zhang (chao.uiuc@gmail.com)

Office Hour: Wednesday 3:30-4:20pm

TA: Hanjun Dai (hanjundai@gatech.edu), Wendi Ren (wren44@gatech.edu)

Office Hour: Monday 3:30-4:20pm

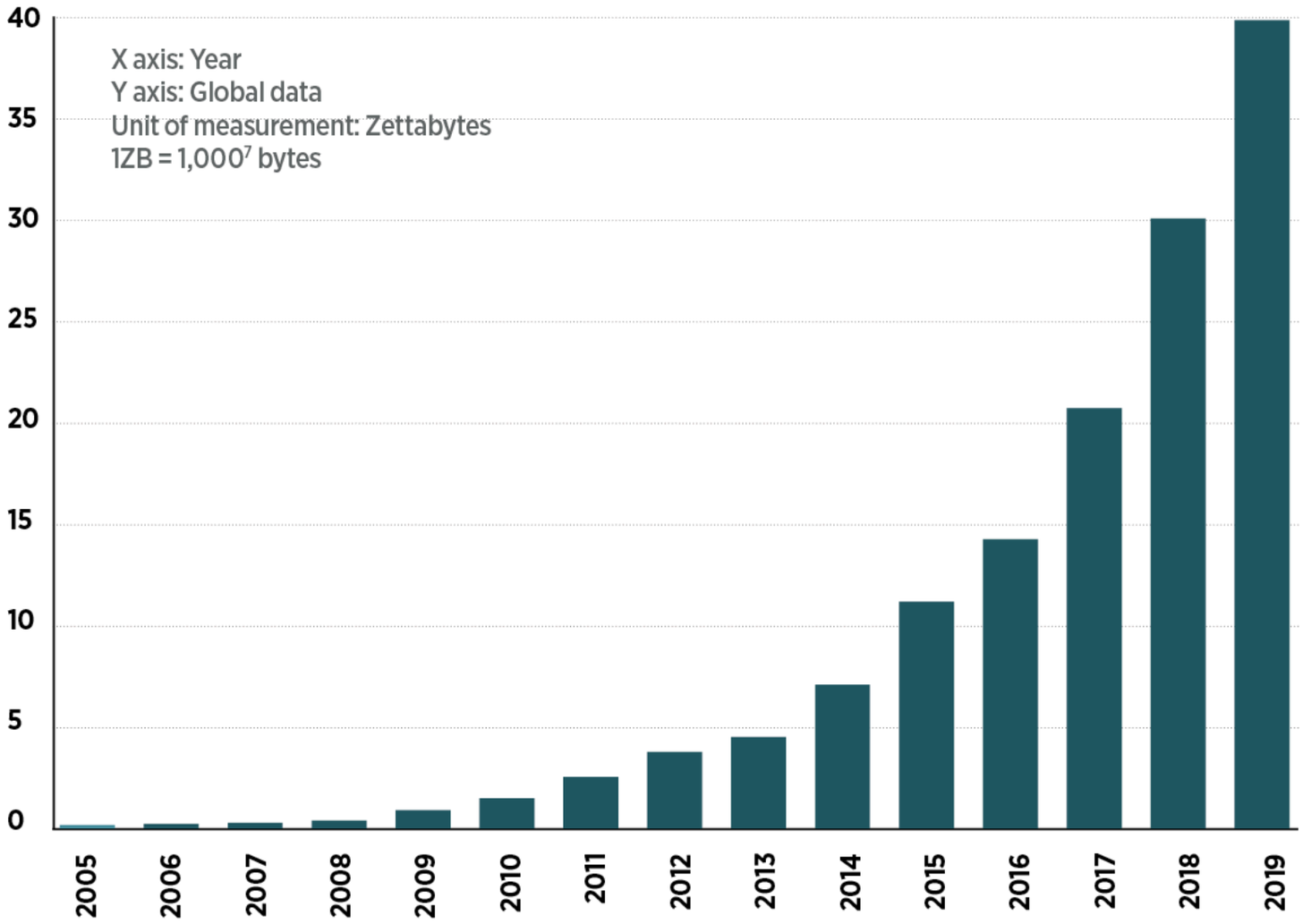
Piazza: https://piazza.com/class/jqeo3f7s5vc426

If you have questions, **ask on Piazza first**. If it's something you do not want to discuss publicly, send an **email** to the teaching staff with **CX4240 in the subject**.

Computational Data Analysis

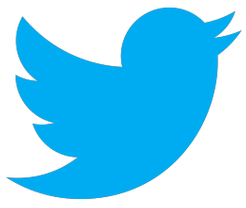
"***We are drowning in information but starved for knowledge.***"

— John Naisbitt



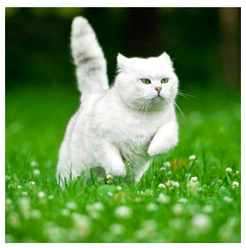
The Booming Age of Data

30 trillion Web pages



500 million tweets per day

2.27 billion monthly active users

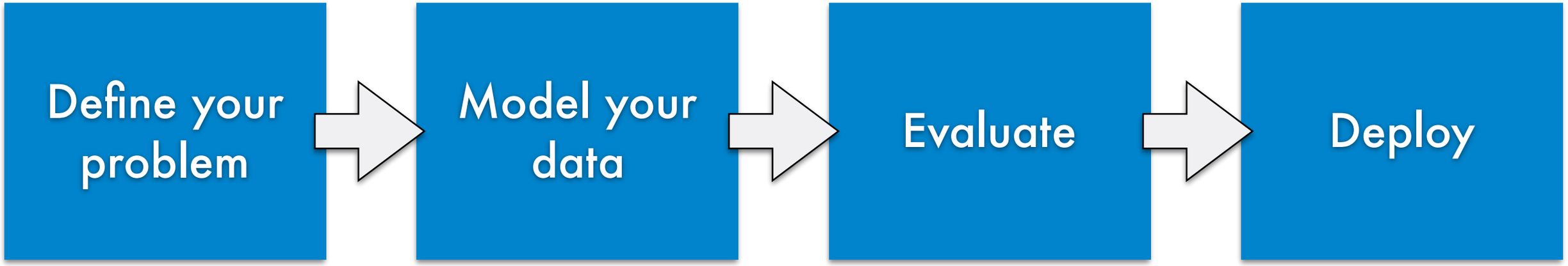


1.8 billion images uploaded to Internet per day

2.9 billion base pairs in human genome

Computational Data Analysis

Computational data analysis is the process of **turning data into actionable knowledge** for **task support** and **decision making**.



Course Objectives

- Introduce to you the **pipeline of computational data analysis**
- Help you understand **major machine learning algorithms**
- Help you learn to **apply tools** for **real data analysis problems**
- Encourage you to **do research** in data science and machine learning

Brief History of Machine Learning

1950s

- Samuel's checker player
- Selfridge's Pandemonium

1960s:

- Neural networks: Perceptron
- Pattern recognition
- Learning in the limit theory
- Minsky and Papert prove limitations of Perceptron

1970s:

- Symbolic concept induction
- Winston's arch learner
- Expert systems and the knowledge acquisition bottleneck
- Quinlan's ID3
- Michalski's AQ and soybean diagnosis
- Scientific discovery with BACON
- Mathematical discovery with AM

Slide Credit: Ray Mooney

Brief History of Machine Learning

1980s:

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

1990s

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

Slide Credit: Ray Mooney

Brief 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
- Learning in robotics and vision

2010s:

- Deep learning
- Reinforcement learning
- Generative models
- Adversarial learning
- Muti-task learning
- Learning in NLP, CV, Robotics, …

Based on Ray Mooney's slide

Syllabus

Part I: Basic math for computational data analysis

- Probability, statistics, linear algebra

Part II: Unsupervised learning for data exploration

- Clustering analysis, dimension reduction, kernel density estimation

Part III: Supervised learning for predictive analysis

- Tree-based models, linear classification/regression, neural networks

Syllabus: Unsupervised Learning

Clustering Analysis

- K-means
- Gaussian mixture model
- Hierarchical clustering
- Density-based clustering

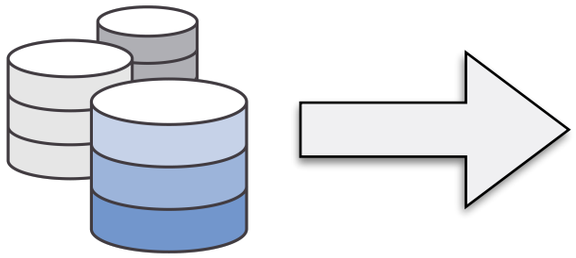
Dimension Reduction

- Principal component analysis

Kernel Density Estimation

- Kernel density estimation
- Mean shift algorithm

Image Clustering



What are the inputs and how to represent them?

What are the desired outputs?

What learning algorithms to choose?

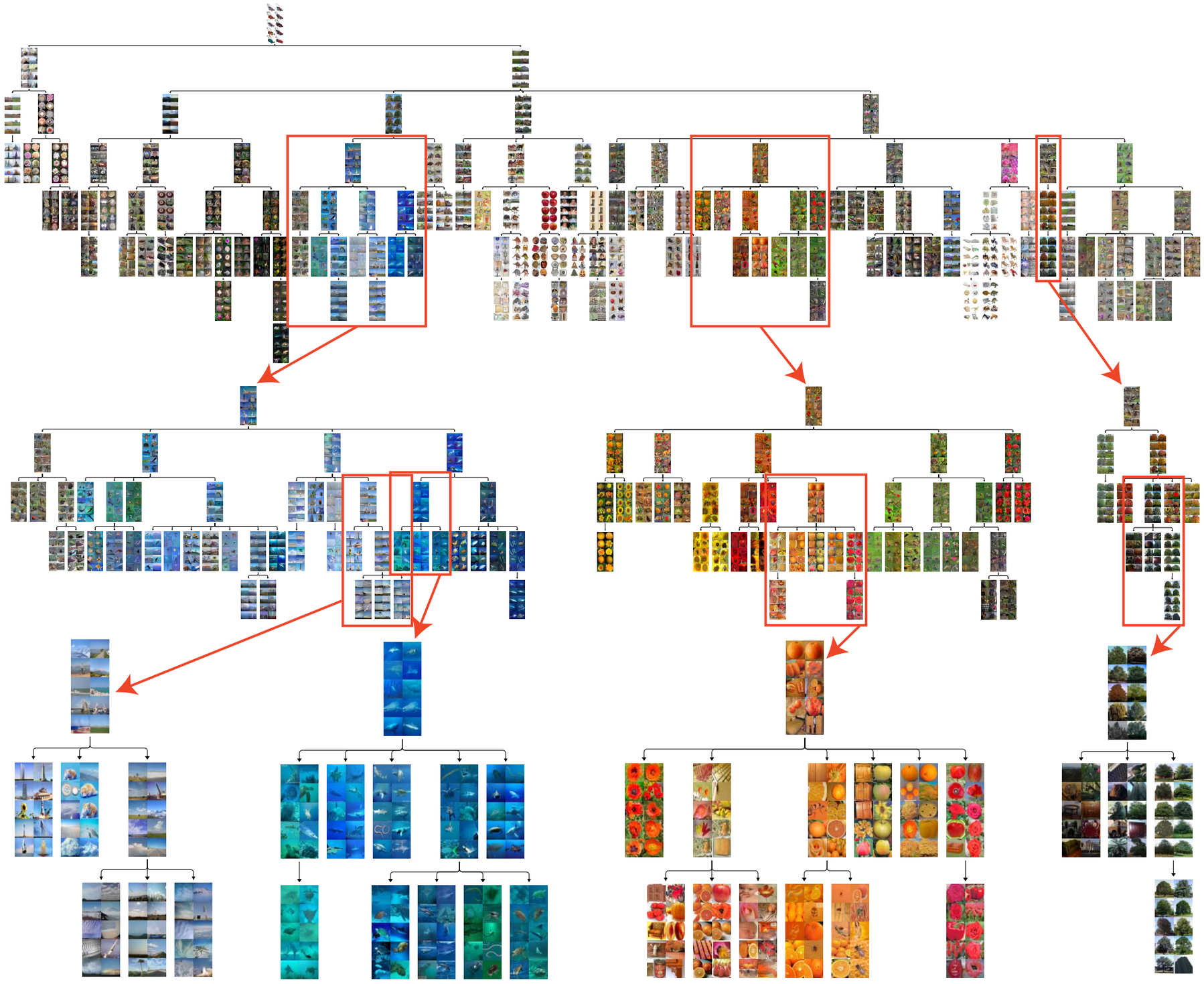


Image credit: Alex Smola

Community Detection in Social Networks

What are the inputs and how to represent them?

What are the desired outputs?

What learning algorithms to choose?

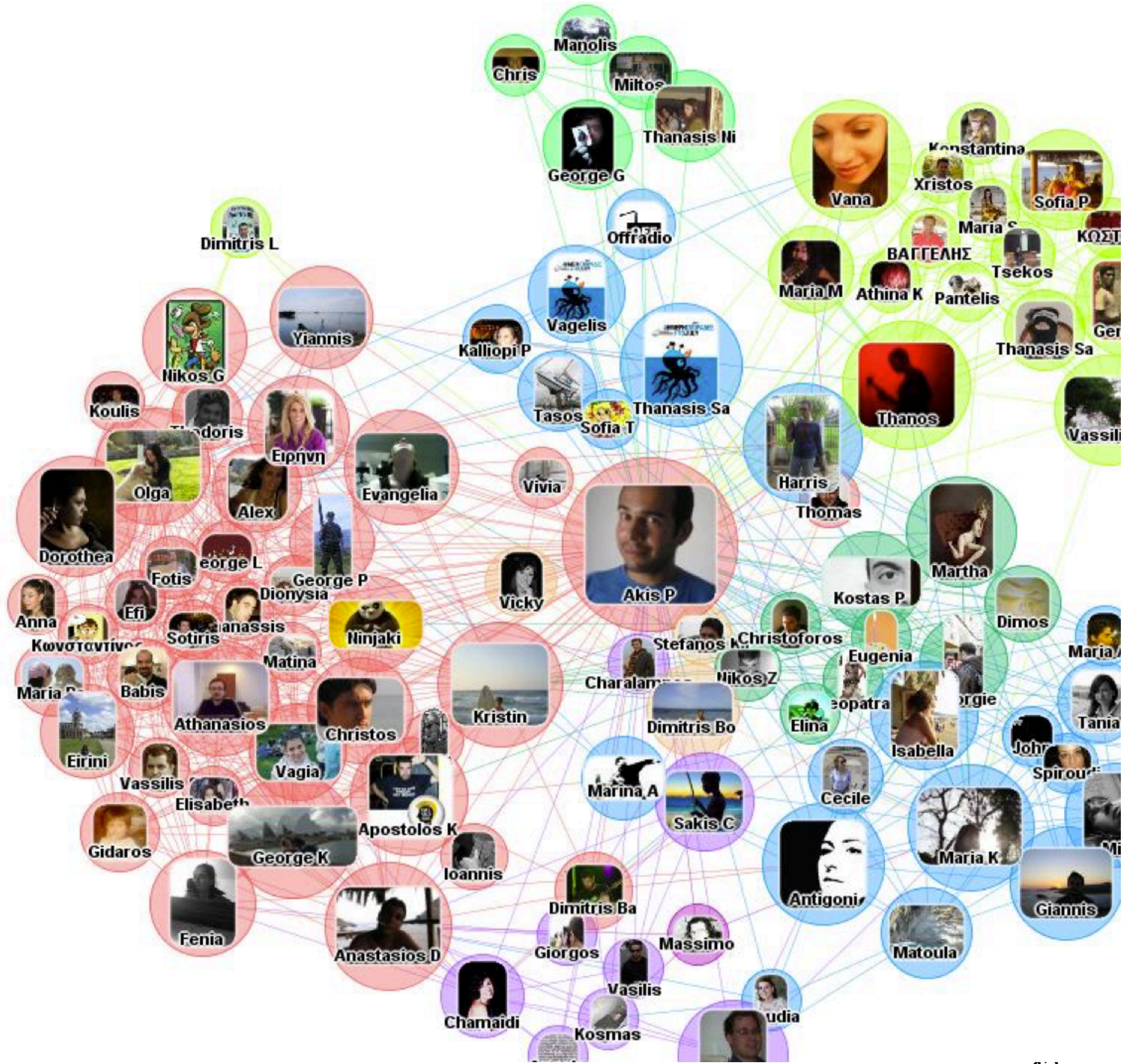
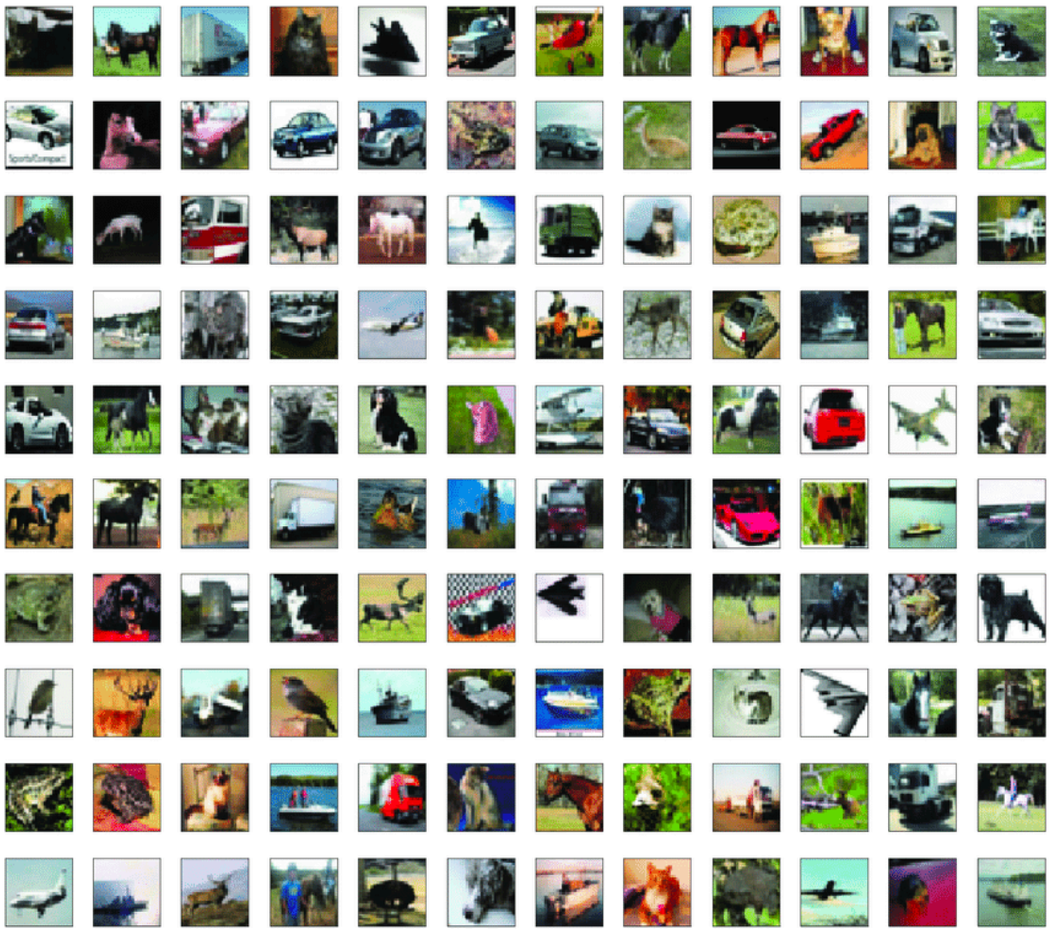
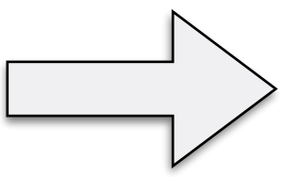
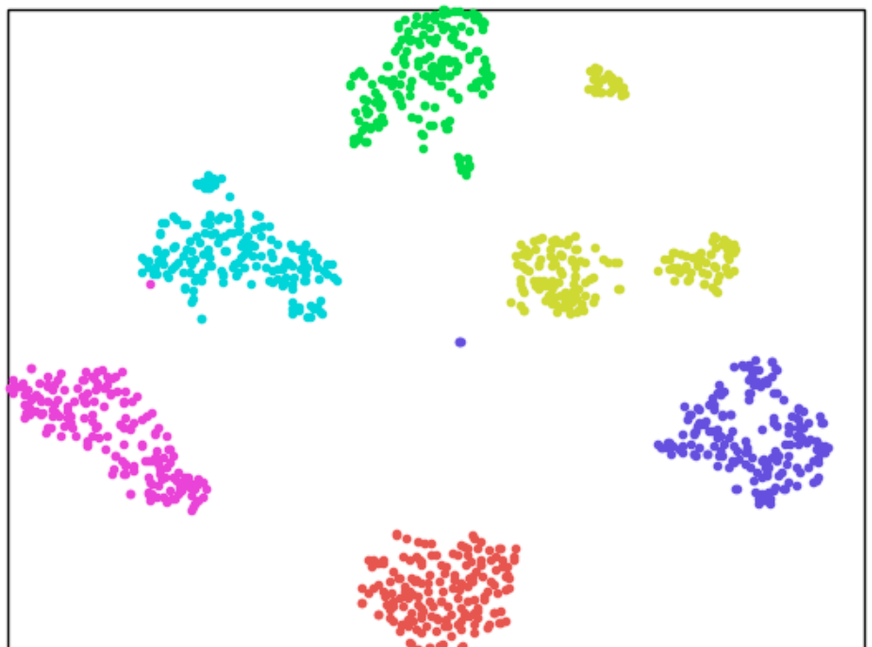


Image credit: Symeon Papadopoulos

Dimension Reduction







What are the inputs and how to represent them?

What are the desired outputs?

What learning algorithms to choose?

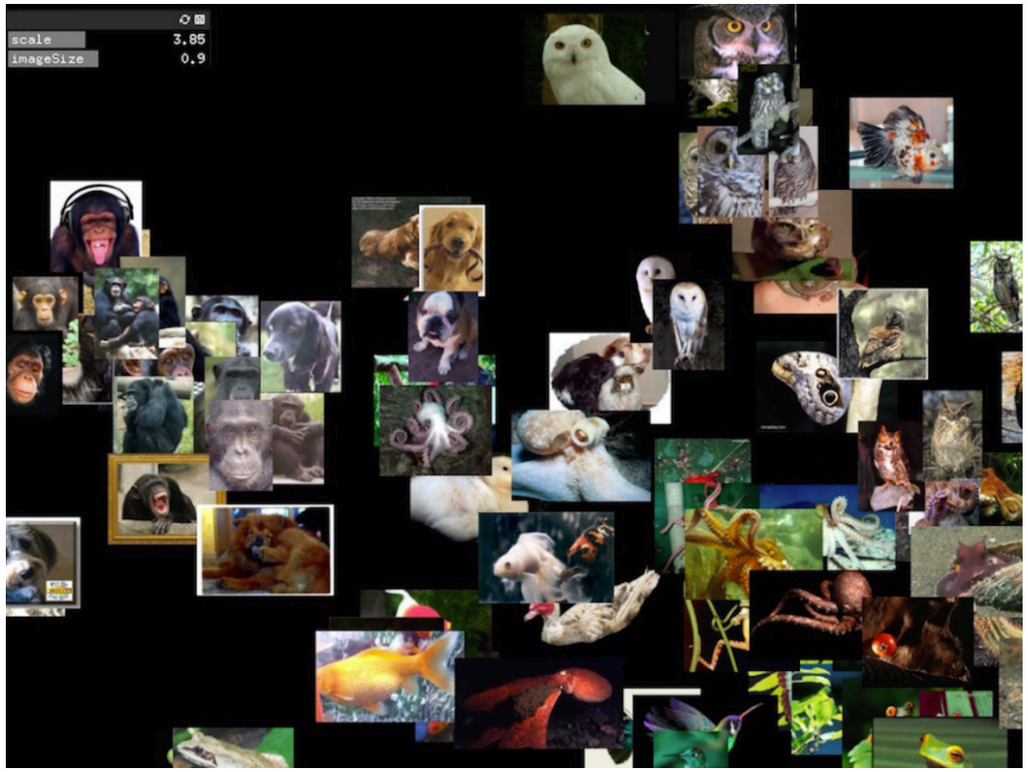
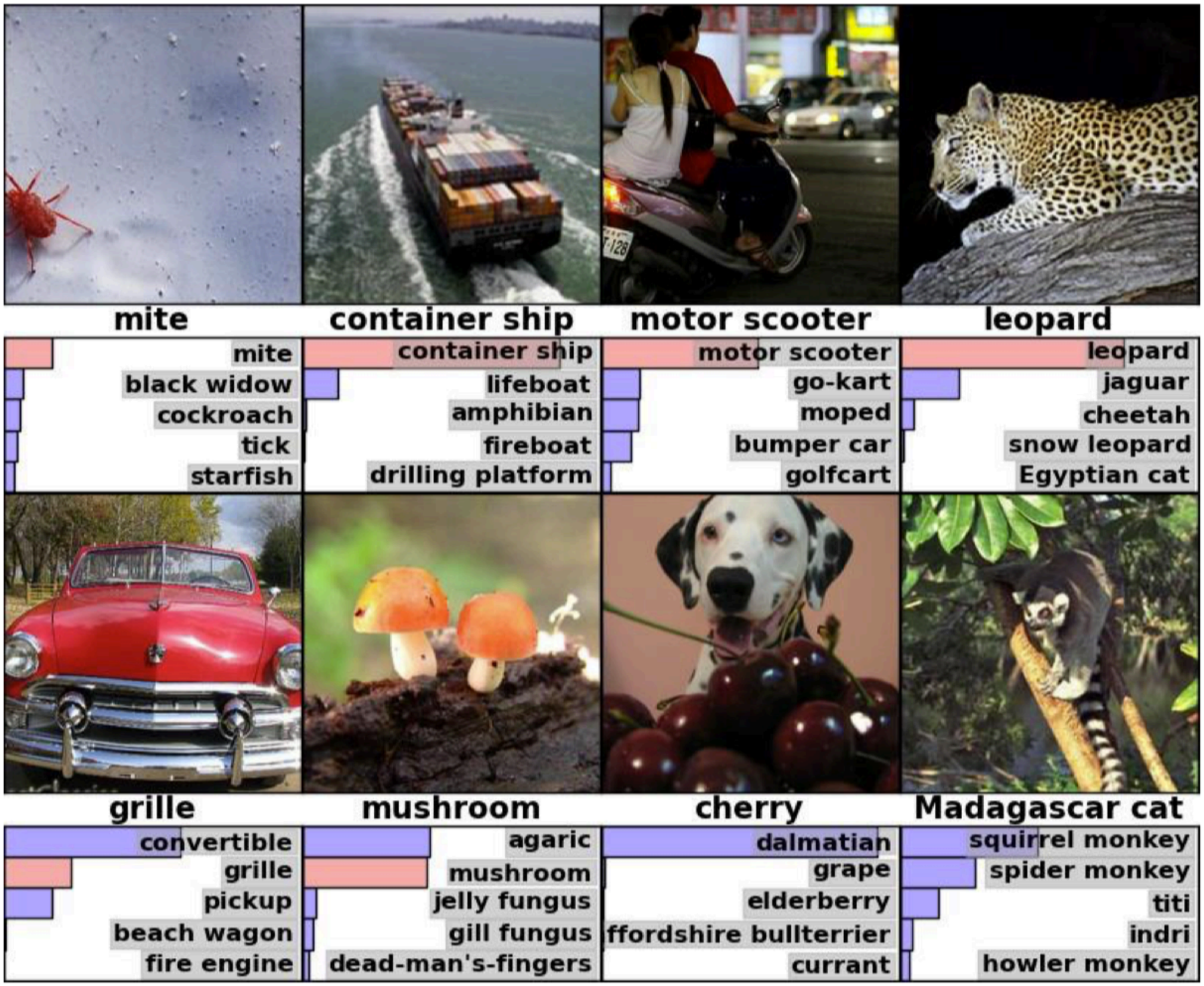


Image credit: Gene Kogan

Syllabus: Supervised Learning

- Tree-based models
  - Decision tree
  - Random forest
- Linear classification/regression models
  - Linear regression
  - Naive Bayes
  - Logistic regression
  - Support vector machine
- Neural networks
  - Feedforward neural networks
  - Convolutional and recurrent neural networks

Image Classification



What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

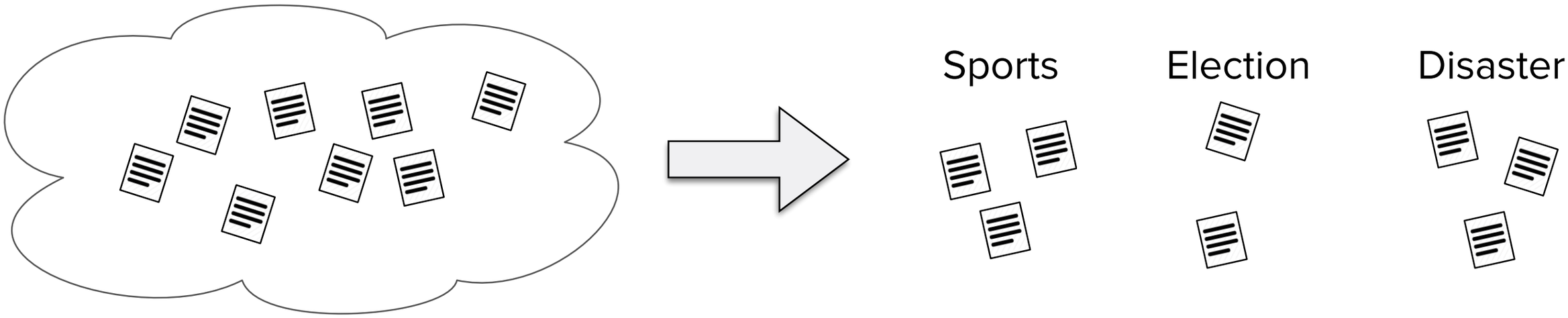
Slide credit: Le Song

News Classification

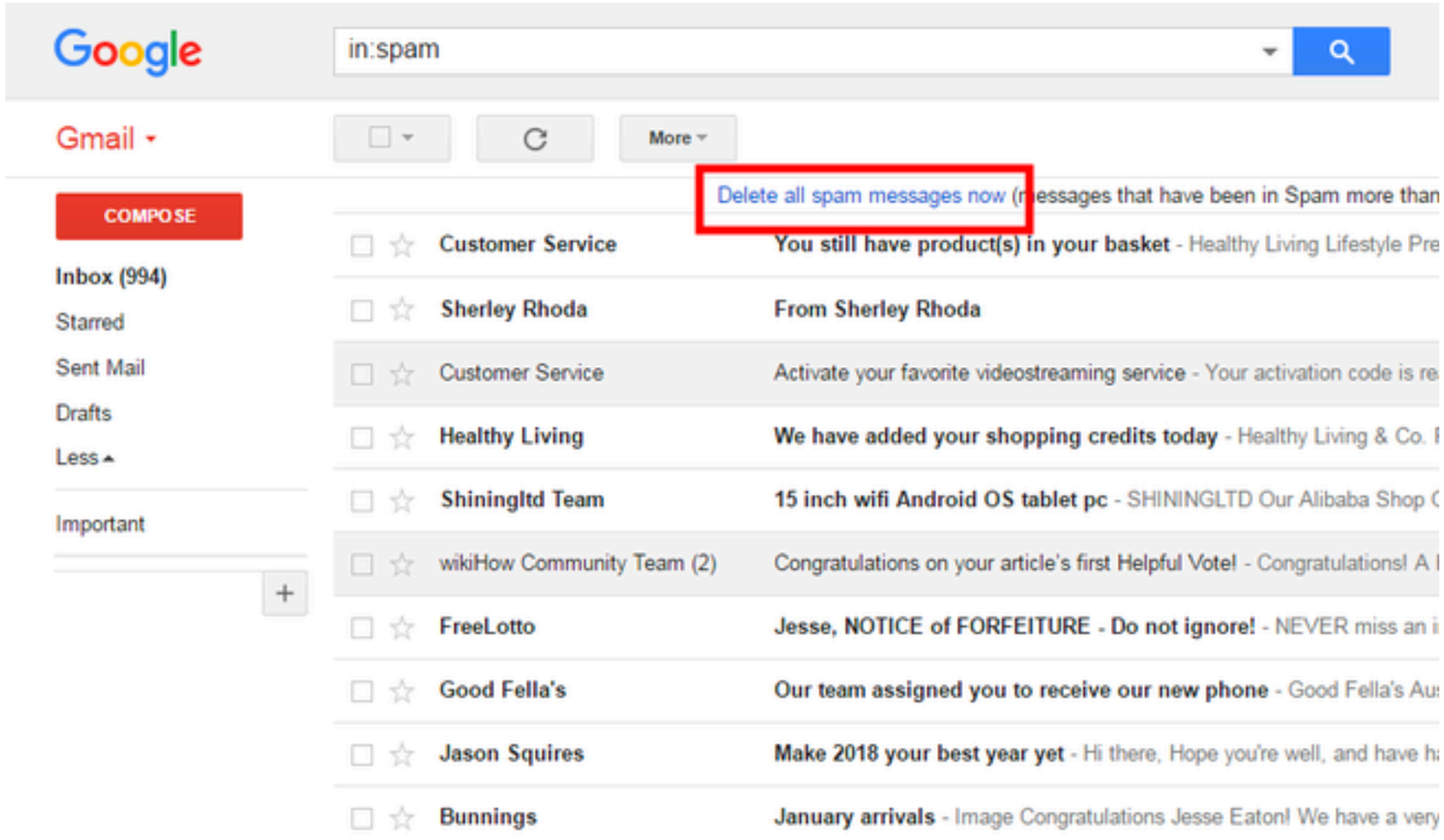
What are the inputs and how to represent them?

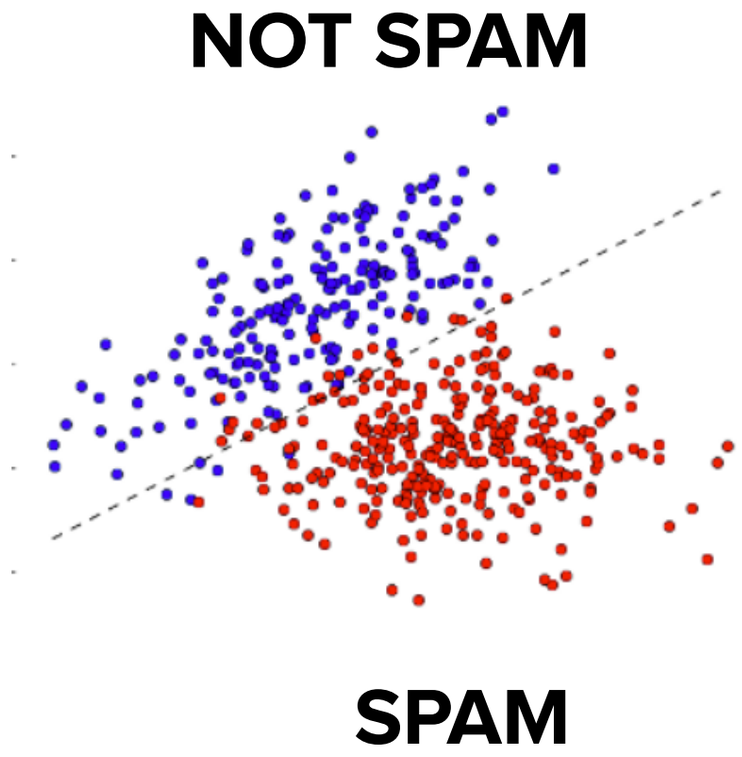
What are the desired outputs?

What learning algorithms to choose?



Spam Detection





What are the inputs and how to represent them?

What are the desired outputs?

What learning algorithms to choose?

Image credit: WikiHow

Stock Price Prediction

Slide credit: Alex Smola

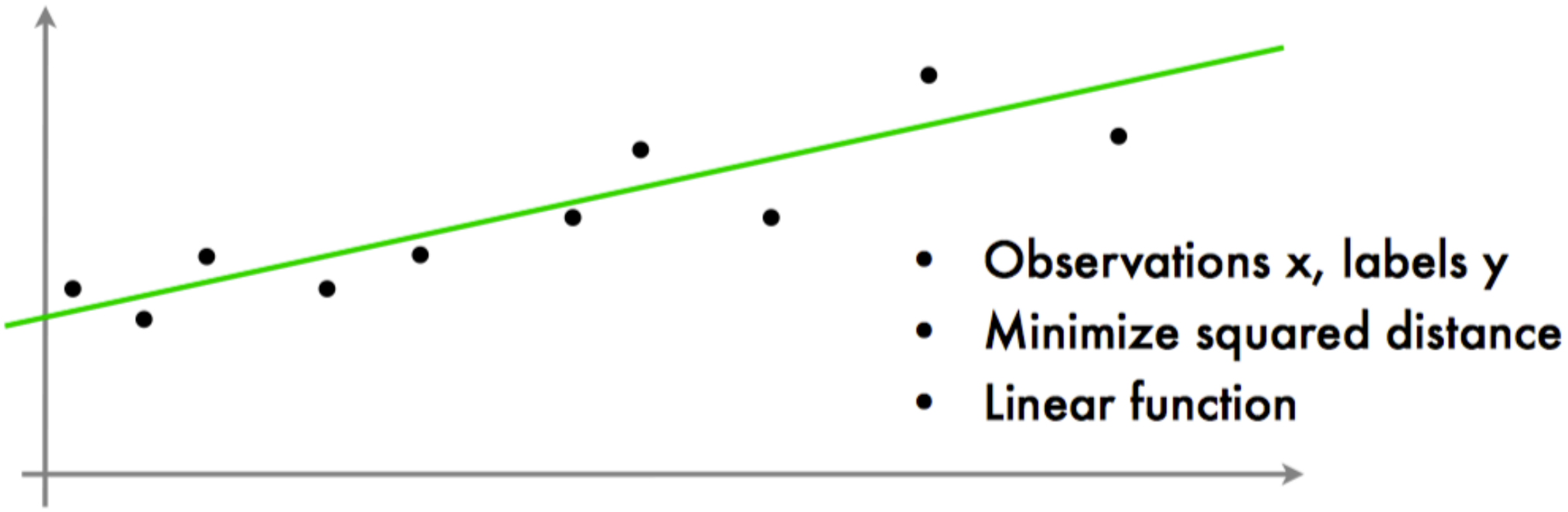
Prerequisites

Basic knowledge in probability, statistics, and linear algebra

Basic programming skills, especially in Python

No background in machine learning is required

Example: Linear Regression



- Observations x, labels y
- Minimize squared distance
- Linear function

$$f(x) = ax + b$$

$$\underset{a,b}{\text{minimize}} \sum_{i=1}^{m} \frac{1}{2}(ax_i + b - y_i)^2$$

$$\partial_a [\ldots] = 0 = \sum_{i=1}^{m} x_i(ax_i + b - y_i)$$

$$\partial_b [\ldots] = 0 = \sum_{i=1}^{m} (ax_i + b - y_)$$

Slide credit: Alex Smola

Example: Linear Regression

- **Optimization Problem**

$$f(x) = \langle a, x \rangle + b = \langle w, (x, 1) \rangle$$

$$\underset{w}{\text{minimize}} \sum_{i=1}^{m} \frac{1}{2} (\langle w, \bar{x}_i \rangle - y_i)^2$$

- **Solving it**

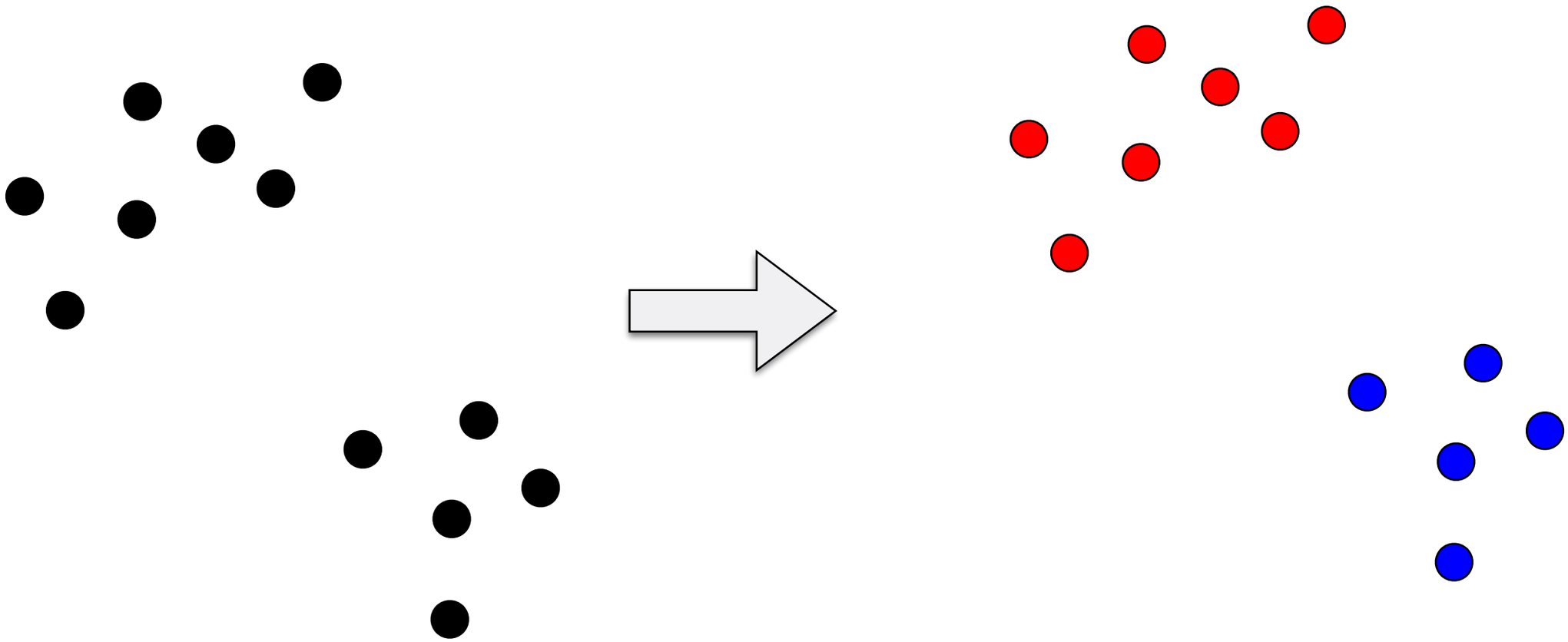
$$0 = \sum_{i=1}^{m} \bar{x}_i (\langle w, \bar{x}_i \rangle - y_i) \iff \left[\sum_{i=1}^{m} \bar{x}_i \bar{x}_i^\top \right] w = \sum_{i=1}^{m} y_i \bar{x}_i$$

only requires a matrix inversion.

Slide credit: Alex Smola

Example: K-Means Clustering

Partition ***n*** points into ***k*** groups such that the points in the same group are “close” to each other.



Example: K-Means Clustering

Given a set of observations $(x_1, x_2, \ldots, x_n)$, where $x_i \in \mathbb{R}^d$

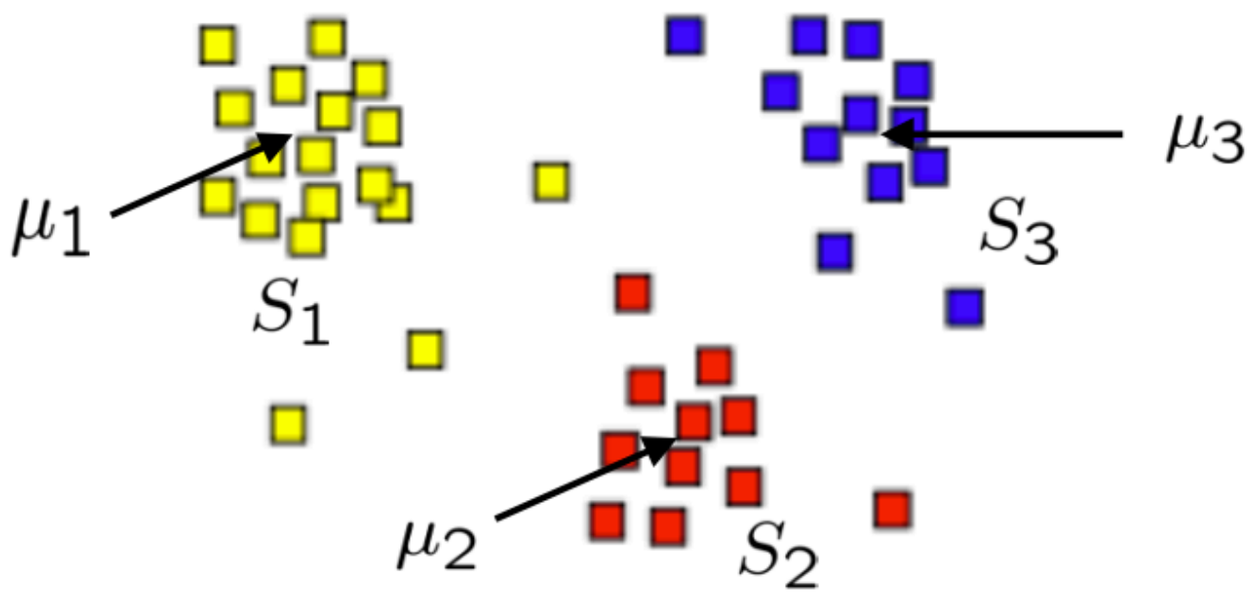
***K*-means clustering problem:**

Partition the *n* observations into *K* sets ($K \leq n$) **S** = {S_1, S_2, …, S_K} such that the sets minimize the within-cluster sum of squares:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{K} \sum_{\mathbf{x}_j \in S_i} \left\| \mathbf{x}_j - \boldsymbol{\mu}_i \right\|^2$$

where μ_i is the mean of points in set S_i.

K=3

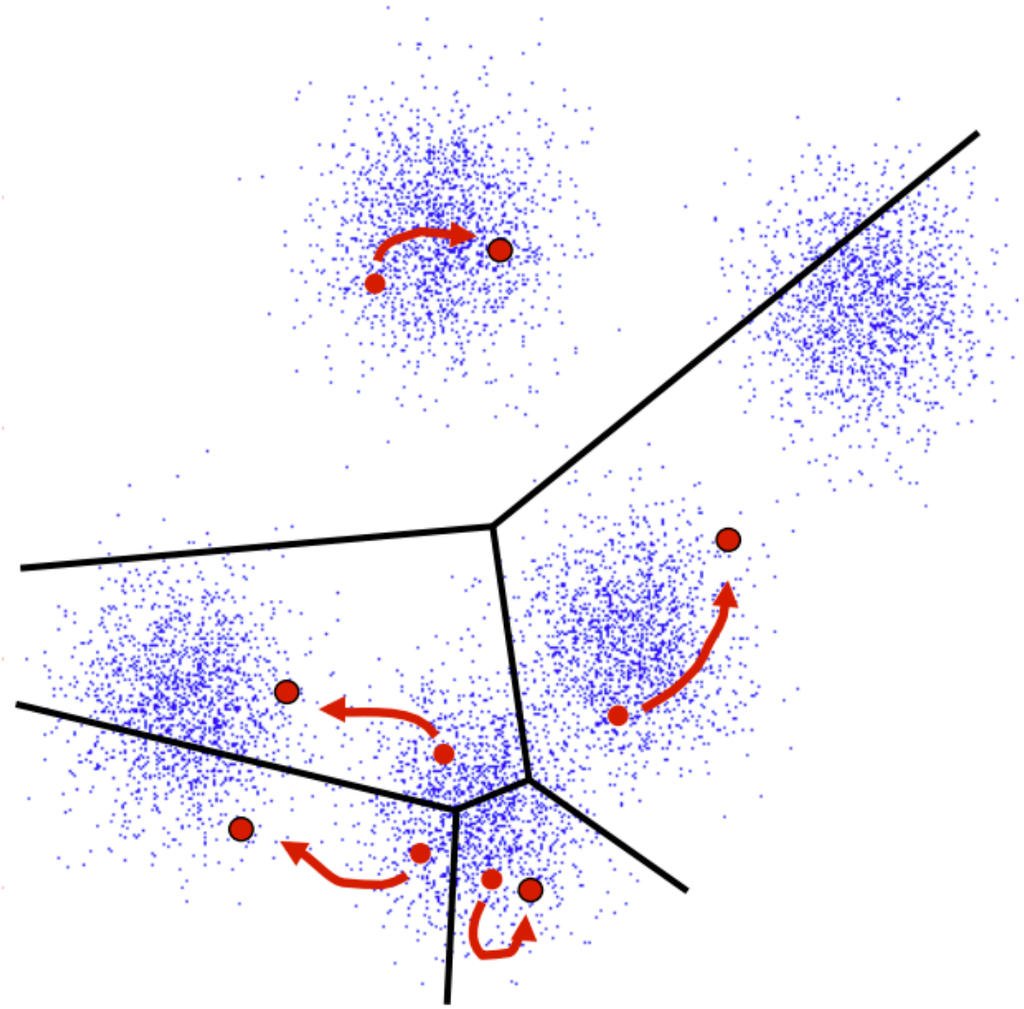


Slide credit: Barnabás Póczos

Example: K-Means Clustering

An iterative clustering algorithm

- Initialize: Pick *K* random points as cluster centers
- Alternate:
  1. Assign data points to closest cluster center
  2. Change the cluster center to the average of its assigned points
- Stop when no points' assignments change

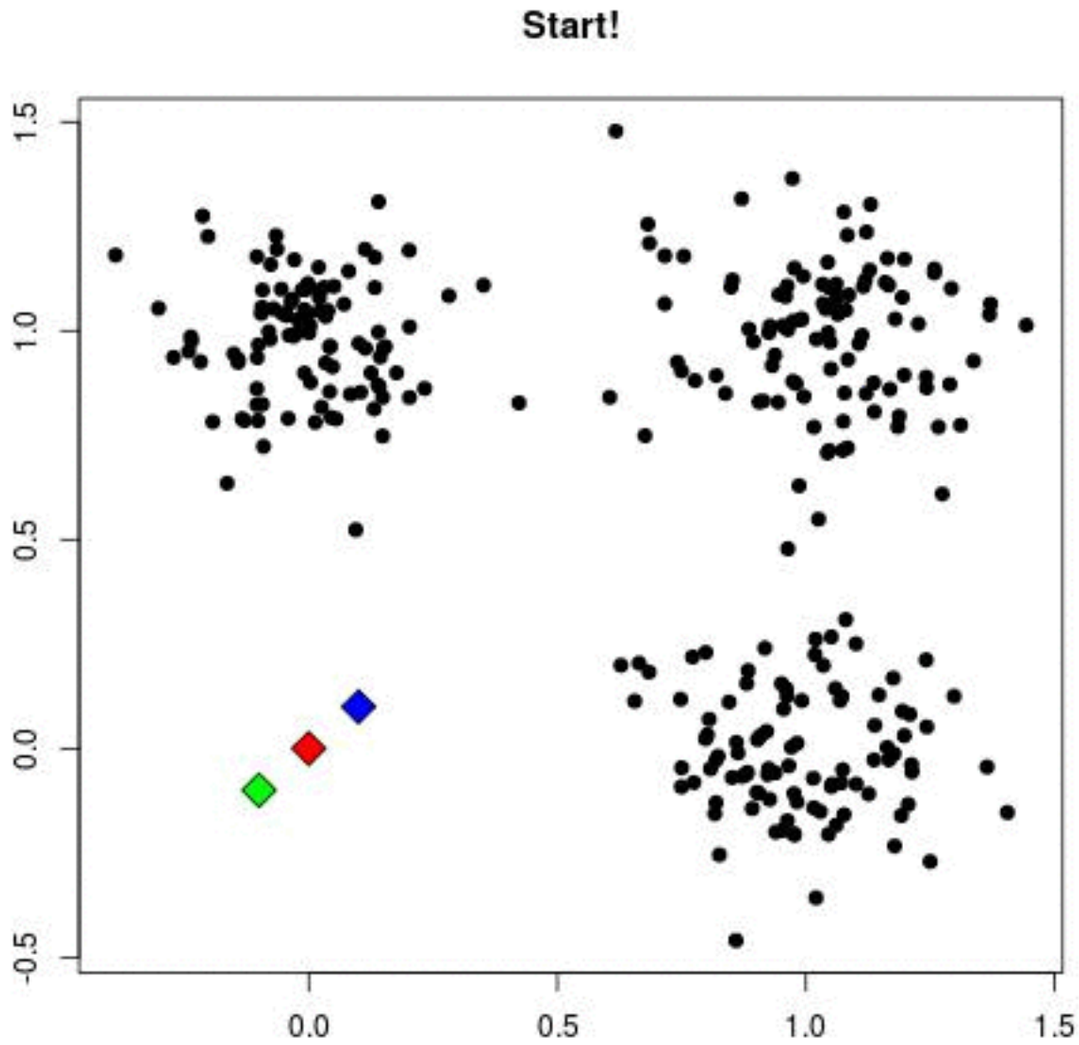


Slide credit: David Sontag

Example: K-Means Clustering

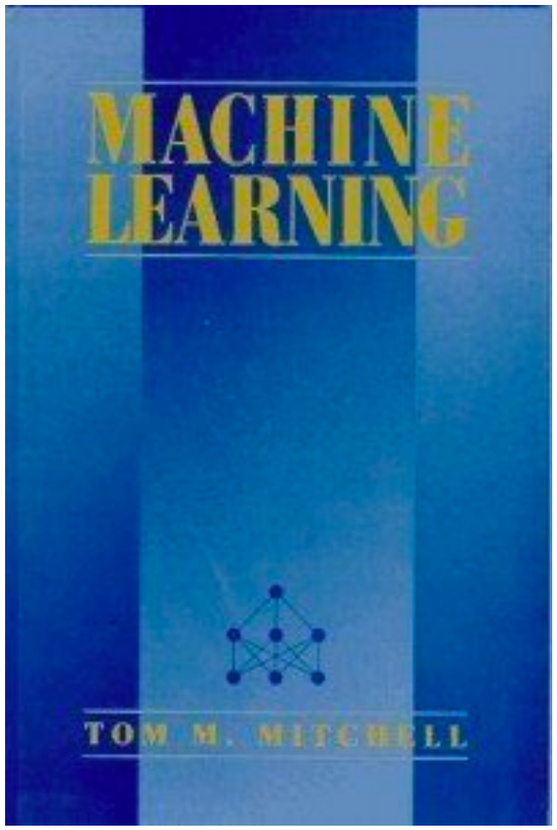
An iterative clustering algorithm

- Initialize: Pick *K* random points as cluster centers
- Alternate:
  1. Assign data points to closest cluster center
  2. Change the cluster center to the average of its assigned points
- Stop when no points' assignments change



Slide credit: David Sontag

Text Books



Machine learning, by Tom Mitchell

Other recommended books:

Data Mining: Concepts and Techniques, by Jiawei Han et al.

The Elements of Statistical Learning, by Trevor Hastie

Deep Learning, by Ian Goodfellow

Pattern recognition and machine learning, by Christopher Bishop

Grading Policy

Assignments (50%)

Four assignments; programming or written analysis

Projects (20%)

2~4 people in a team; require project report (10%) and presentation (10%)

Participation (5%)

Attendance, in-class quizzes; in-class and on-Piazza discussions

Midterm exam (10%)

Open-book; will cover Part I (basic math) and Part II (unsupervised learning)

Final exam(15%)

Open-book; will cover all parts; no make-up exams

Assignments

Four assignments

Each can include written analysis or programming

No-late policy

Assignments received after the due time will receive zero credit

Don’t copy

Any academic misconduct is subject to F grade as well as reporting to the Dean of students.

Projects

- Work on a real-life computational data analysis problem
  - What is the problem? What is your method? How do you evaluate it?
- 2-4 people in a team
- Require a project report as well as a final in-class presentation
- Start your projects early
- Ask for comments and feedbacks from the teaching staff