CS540 Machine learning
Lecture 1
Introduction

Outline

- Administrivia
- Overview
- Supervised learning
- Unsupervised learning
- Other kinds of learning

Administrivia

- Class web page
  www.cs.ubc.ca/~murphyk/Teaching/CS540-Fall08
- Join groups.google.com/group/cs540-fall08
- Office hours: Fri 10.30-11.30am
- Midterm: Tue Oct 14
- Final project due Fri Dec 5th
- weekly homeworks
- Grading
  - Midterm (open-book):30%
  - Final project: 50%
  - Weekly Assignments: 20%

Homeworks

Weekly homeworks, out on Tue, due back on Tue

- Collaboration policy:
  - You can collaborate on homeworks if you write the name of your collaborators on what you hand in; however, you must understand everything you write, and be able to do it on your own (eg. in the exam!)
- Sickness policy:
  - If you cannot do an assignment or an exam, you must come see me in person; a doctor's note (or equivalent) will be required.

Workload

- This class will be quite time consuming.
- Attending lectures: 3h.
- Weekly homeworks: about 6h.
- Weekly reading: about 6h.
- Total: 15h/week.

- If this is too time consuming, and/or you don’t have the pre-reqs, why not take CS340, the ugrad ML class, this Fall? (Can still get grad credit!)

Pre-requisites

- You should know
  - Basic multivariate calculus e.g.,

$$\nabla_{\mathbf{X}} \mathbf{x}^T \mathbf{a} = \mathbf{a}$$

  - Basic linear algebra e.g.,

$$A\vec{u}_i = \lambda_i \vec{u}_i$$

  - Basic probability/ statistics e.g.

$$Cov(X, Y) = E\left[(X - EX)(Y - EY)\right] = E[XY] - E[X]E[Y]$$

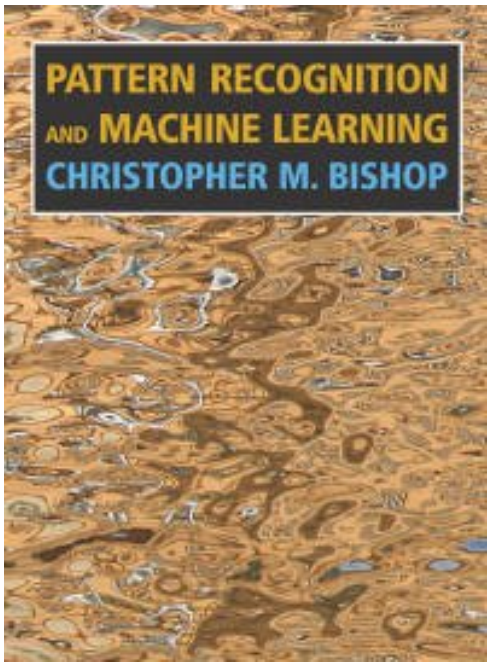
  - Basic data structures and algorithms (e.g., trees, lists, sorting, dynamic programming, etc)

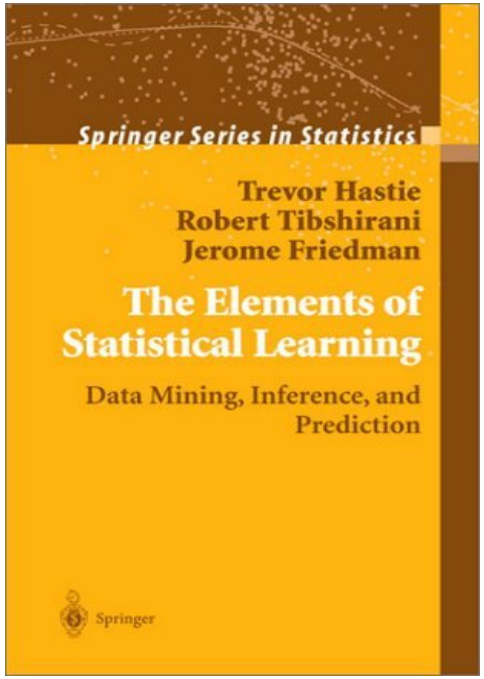
Textbook

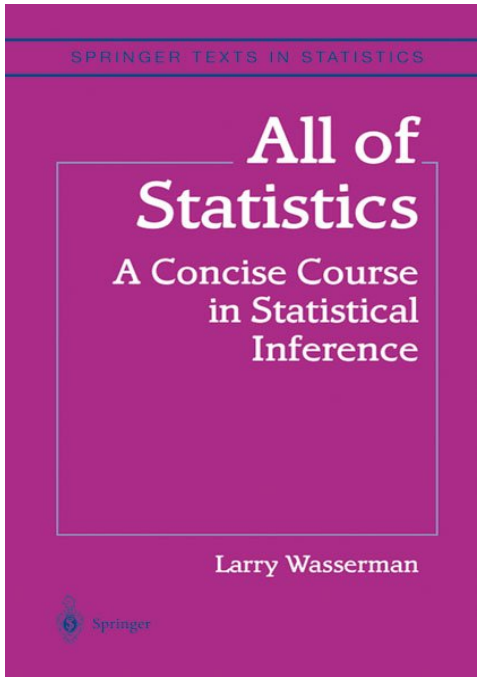
- “Machine learning: a probabilistic approach”
- Draft copies available from Copiesmart in the UBC Village (next to Macdonald’s) for about $35
- pdf online for color pictures/ easy searching – please do not distribute by email!
- See whiteboard for secret password
- Extra credit (up to 5% of your grade) for finding errors (5 points) or typos (1 point) – consult list of typos on book webpage before sending me your list (one email per chapter).
- Please bring your book to every class.

Other good books

If you want a book that is already “debugged”, see one of these







Matlab

- Matlab is a mathematical scripting language widely used for machine learning (and engineering and numerical computation in general).
- Everyone should have access to Matlab via their CS account. If not, ask for a CS guest account.
- You can buy a student version for $170 from the UBC bookstore. Please make sure it has the Stats toolbox.
- Matt Dunham has written an excellent Matlab tutorial which is on the class web site – please study it carefully!

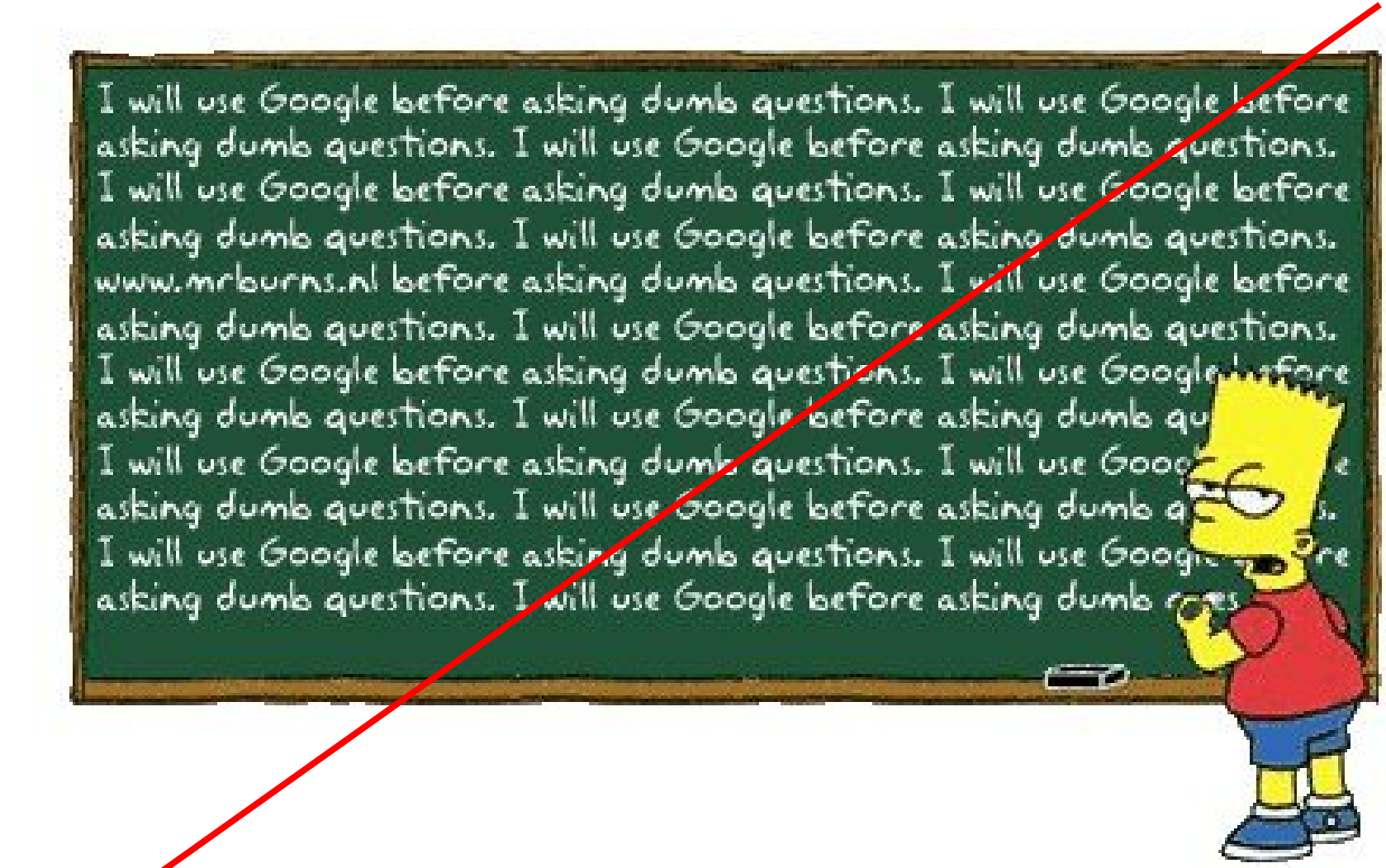
BLT

- Bayesian Learning Toolkit (BLT) is a Matlab package I am currently developing to go along with my book.
- It uses the latest object oriented features of Matlab 2008a and will not run on older versions.

Learning objectives

- By the end of this class, you should be able to
  - Understand basic principles and techniques of machine learning and its connection to other fields
  - Create suitable statistical models for any given problem
  - Derive the algorithm (equations etc) needed to learn and apply the model
  - Implement the algorithm in reasonably efficient Matlab
  - Demonstrate your skills by doing a reasonably challenging project

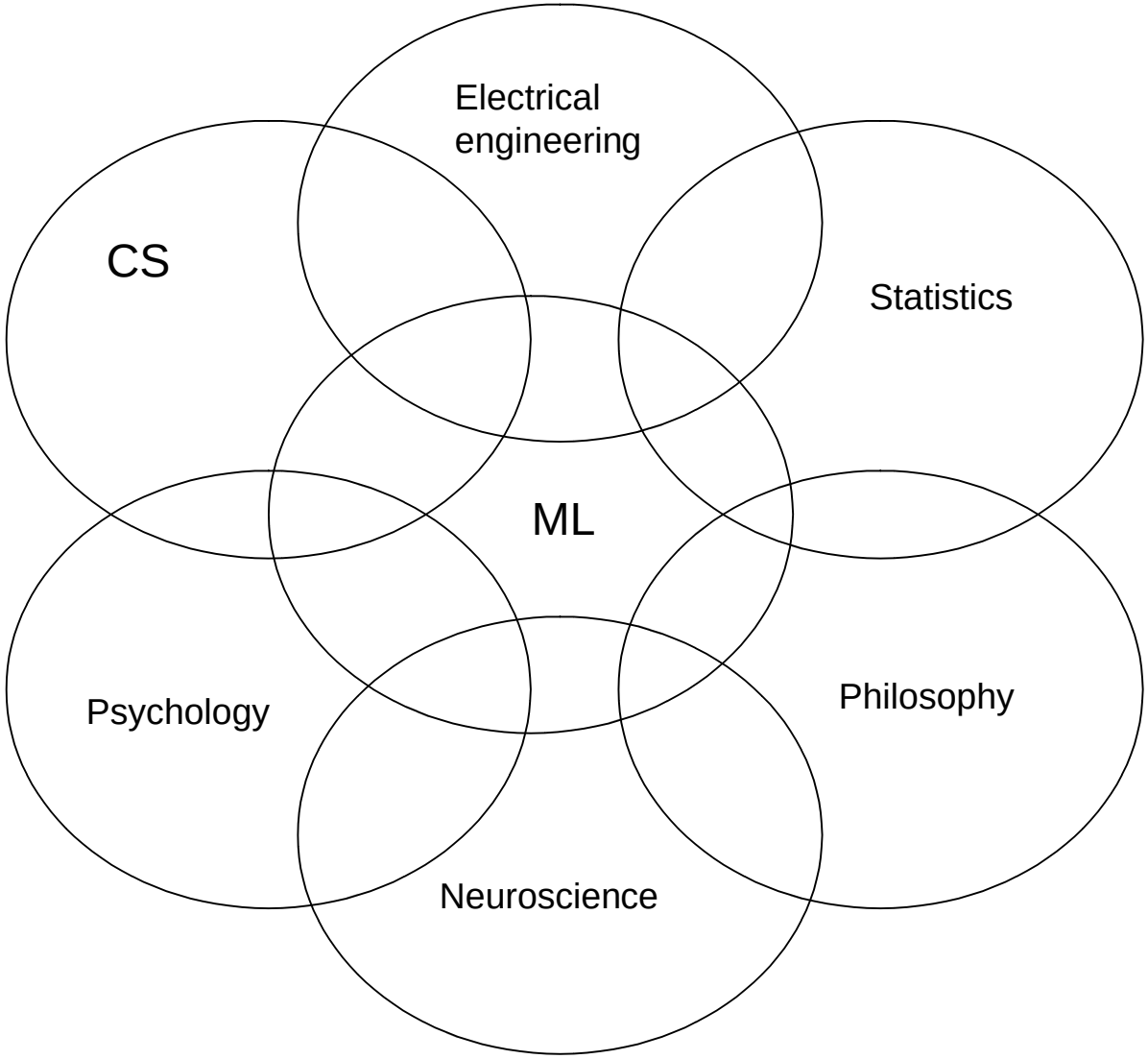
Ask questions early and often!



Outline

- Administrivia
- Overview
- Supervised learning
- Unsupervised learning
- Other kinds of learning

What is machine learning?



What is machine learning?

- ``Learning denotes changes in the system that are adaptive in the sense that they enable the system to do the task or tasks drawn from the same population more efficiently and more effectively the next time." -- Herbert Simon
- Closely related to
  - Statistics (fitting models to data and testing them)
  - Data mining/ exploratory data analysis (discovering patterns in data)
  - Adaptive control theory (learning models online and using them to achieve goals)
  - AI (building intelligent machines by hand)

Types of machine learning

- Supervised Learning
  - Predict output from input
- Unsupervised Learning
  - Find patterns in data
- Reinforcement Learning
  - Learn how to behave in novel environments (eg robot navigation)
  - not covered in this class – see e.g., CS422

Why “Learn” ?

- Machine learning is programming computers to optimize a performance criterion using example data or past experience.
- There is no need to “learn” to calculate payroll
- Learning is used when:
  - Humans not in loop (navigating on Mars)
  - Humans are unable to explain their expertise (speech recognition)
  - Solution changes in time (routing on a computer network)
  - Solution needs to be adapted to particular cases (user biometrics)

Outline

- Administrivia
- Overview
- Supervised learning
- Unsupervised learning
- Other kinds of learning

Supervised learning

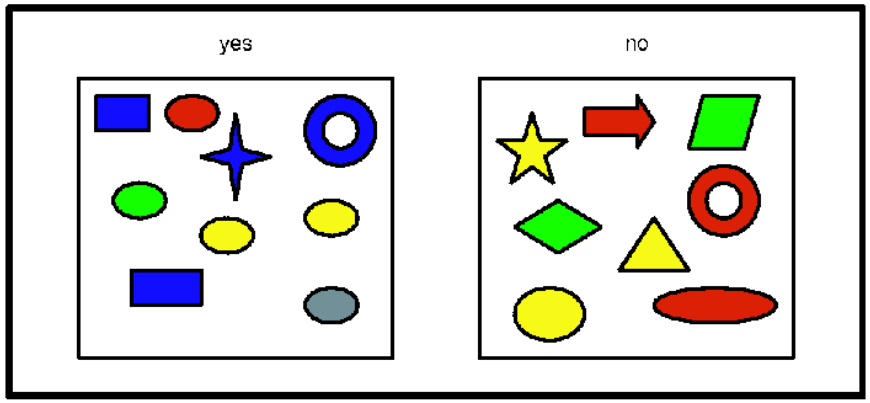
- Learning a mapping f from input x to output y:

$$\hat{y} = f(\mathbf{x})$$

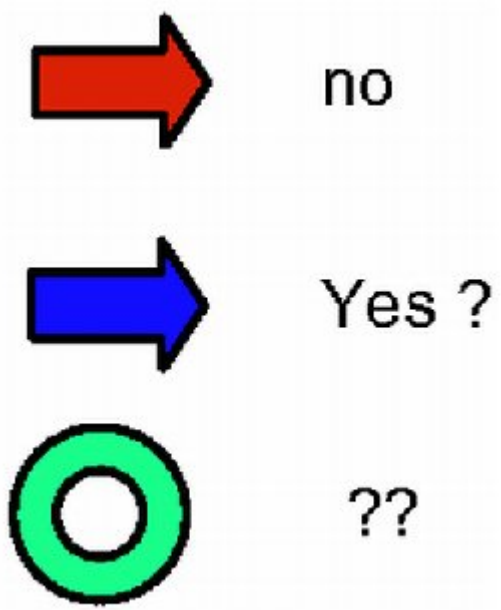
- If y 2 {1,…,C}, this is called classification
- If y 2 R, this is called regression

Binary classification

Training data

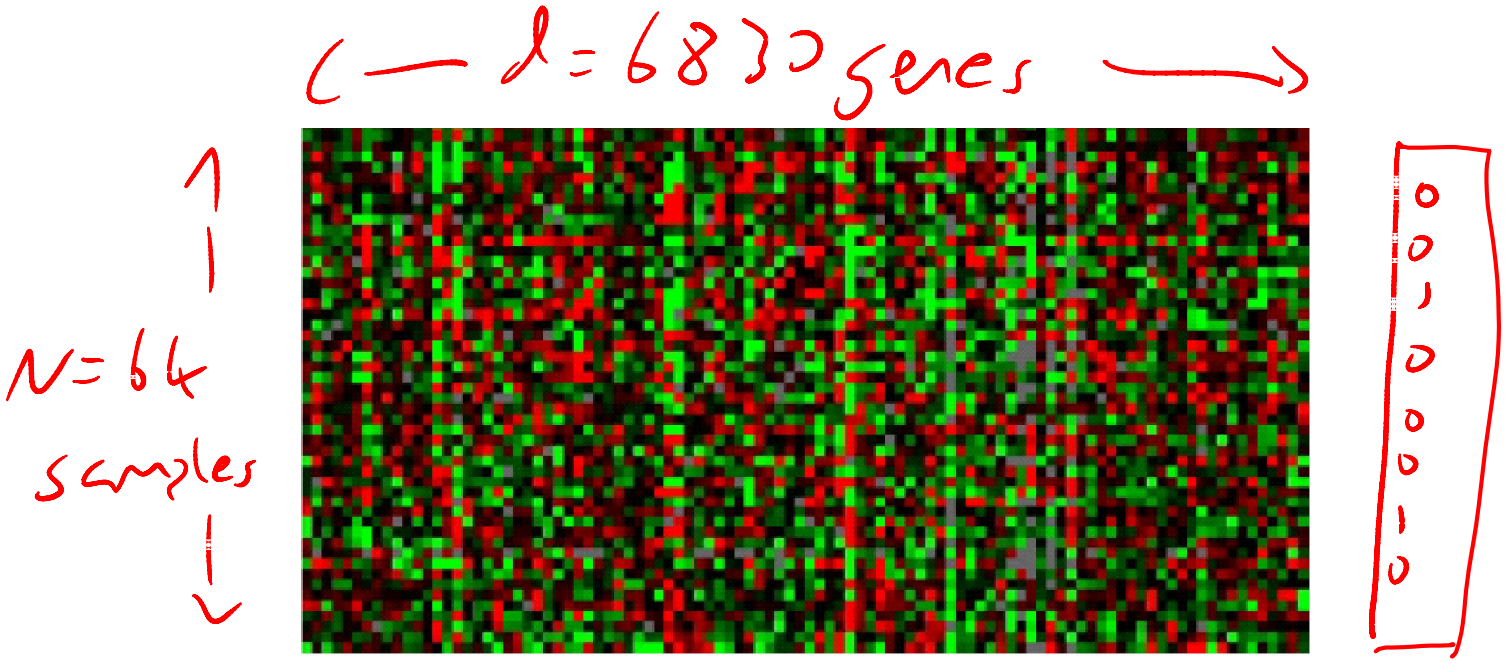


Testing data



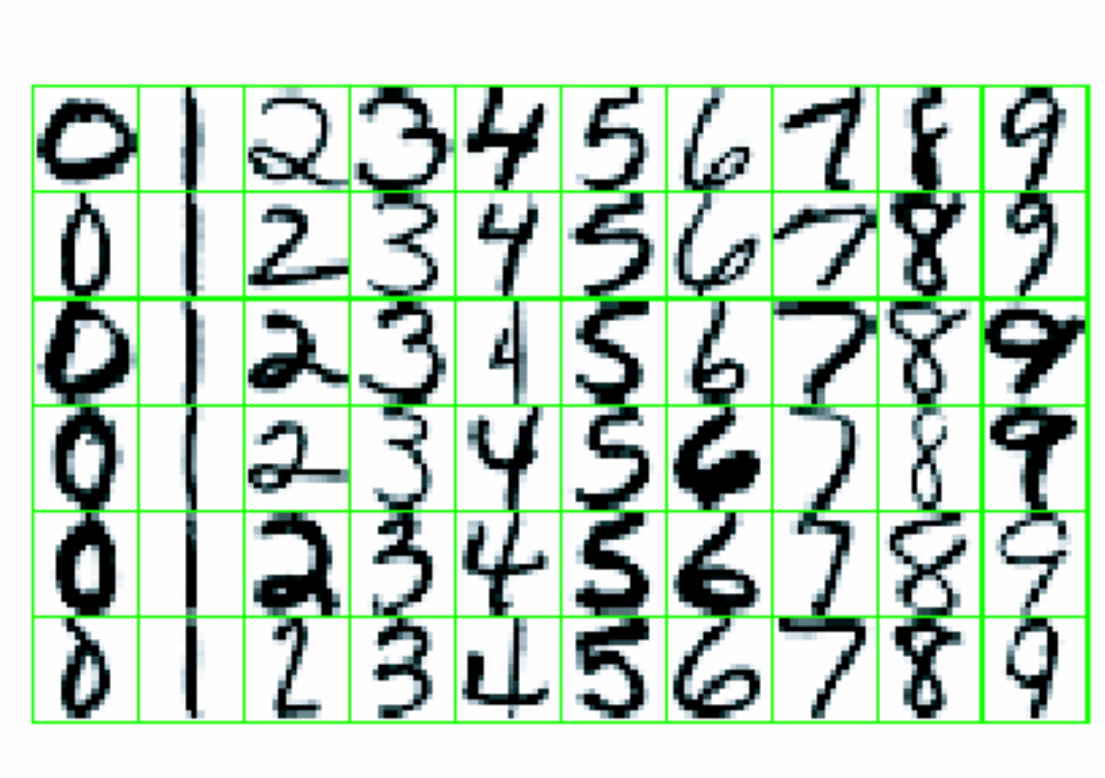
X y

Classifying gene microarray data



Handwritten digit recognition

- x 2 $R^{16 \pounds 16}$, y 2 {0,...,9}

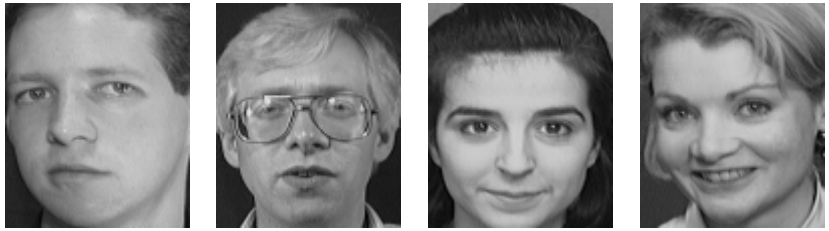


Face Recognition

Training examples of a person

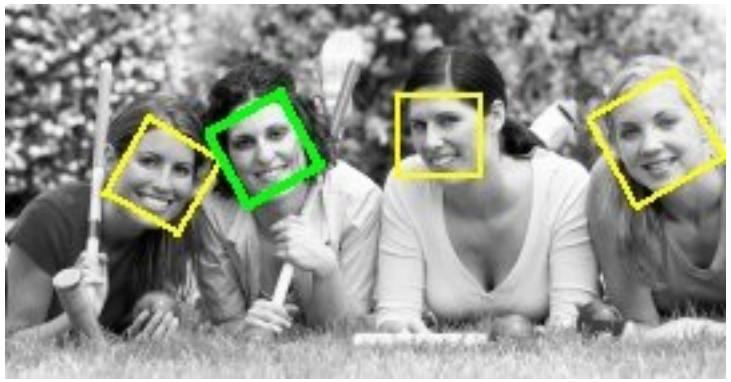
Possibly no negative examples

Test images



Face detection

http://demo.pittpatt.com

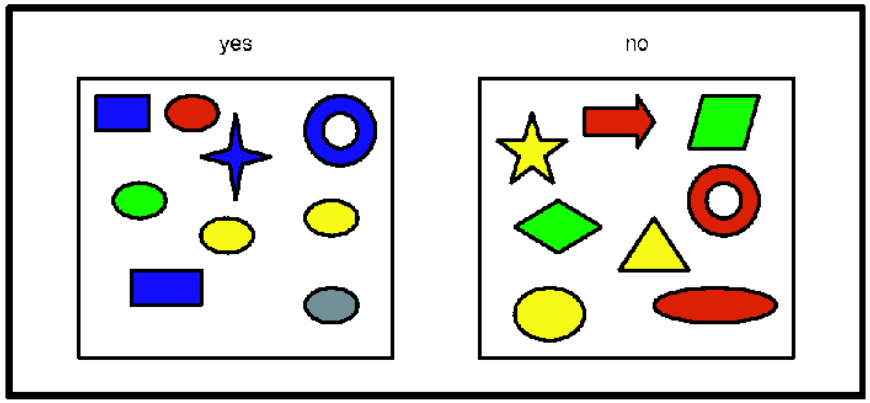


Car detection

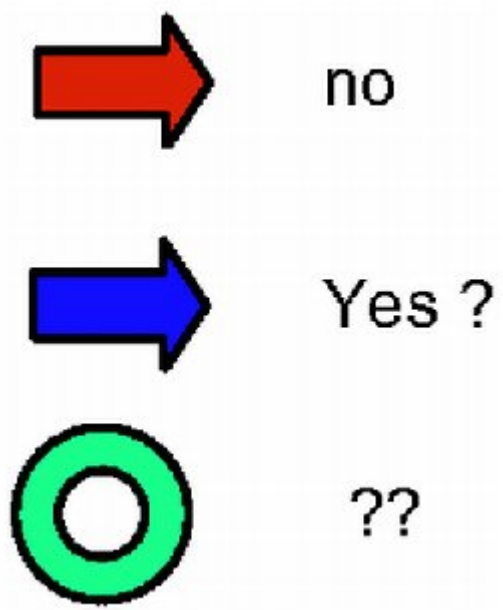
Jordan Reynolds, UBC, 2004

Probabilistic output

Training data



Testing data

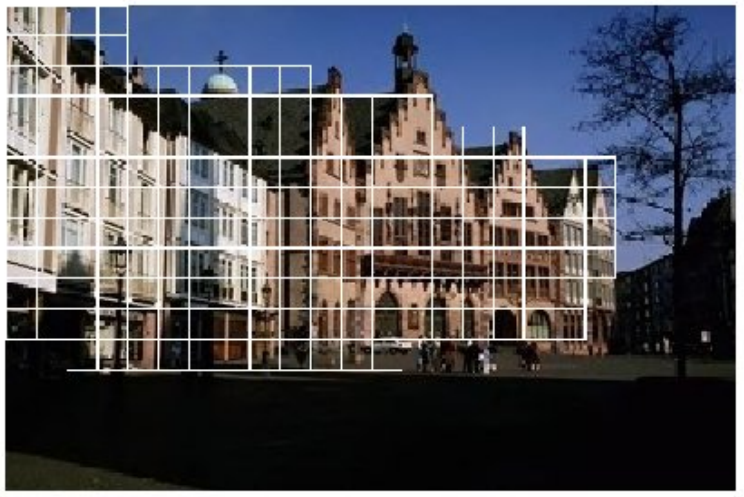


$$\hat{y} = \arg \max_{c \in \{1,\ldots,C\}} p(y = c|\mathbf{x})$$

x y

Structured output classification

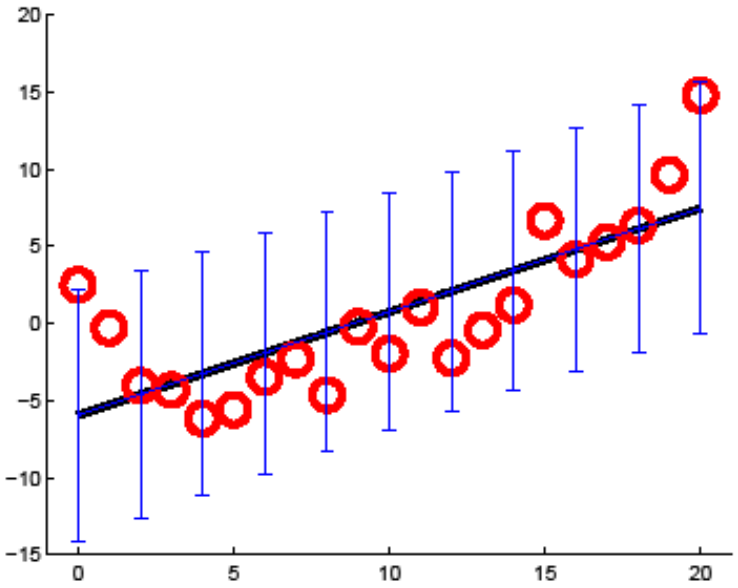
- Predict multiple output labels, which may be correlated
- Here we use a conditional random field (CRF)

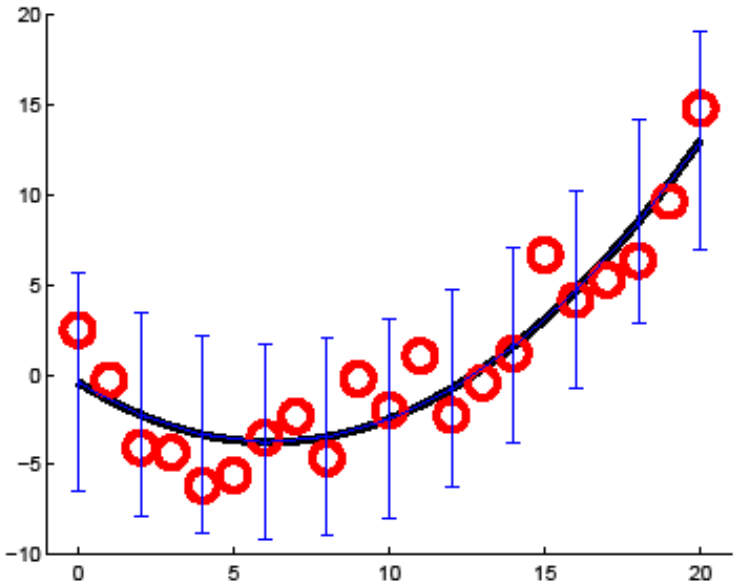


Regression

$$f(x) = w_0 + w_1 x + w_2 x^2 + \cdots + w_d x^D$$

$$f(x) = \mathbf{w}^T \boldsymbol{\phi}(x) = \sum_{j=1}^{d} w_k \phi_j(\mathbf{x})$$



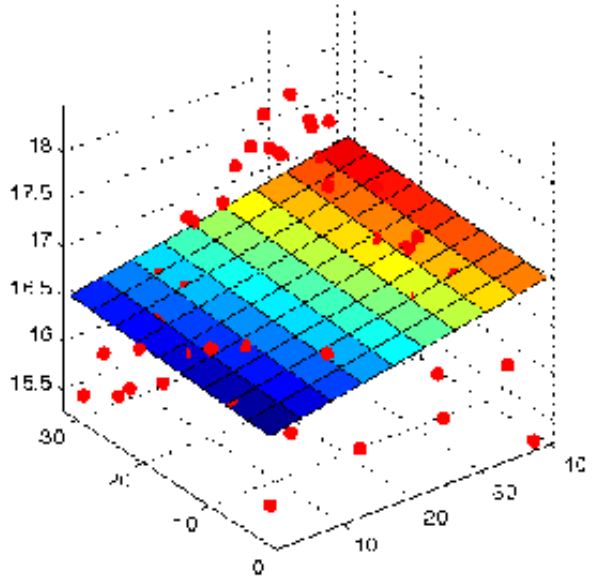


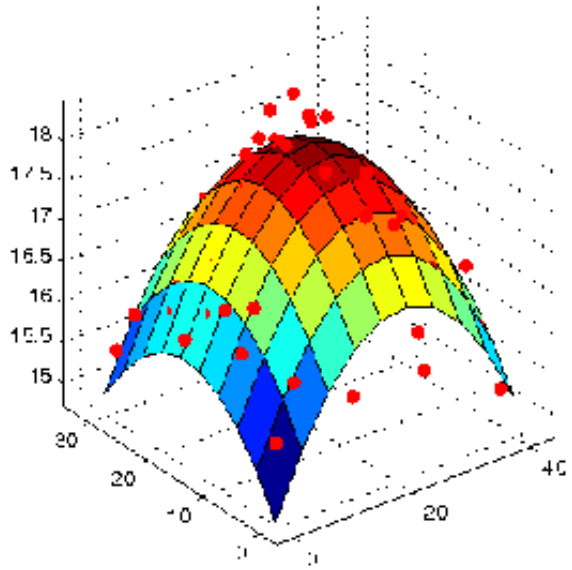
Line denotes posterior mode arg $\max_y$ p(y|x)

Error bars denote 95% credible interval $p(y \in I|\mathbf{x}) = 0.95$

Regression

$$f(x) = \mathbf{w}^T \boldsymbol{\phi}(x) = \sum_{j=1}^{d} w_k \phi_j(\mathbf{x})$$





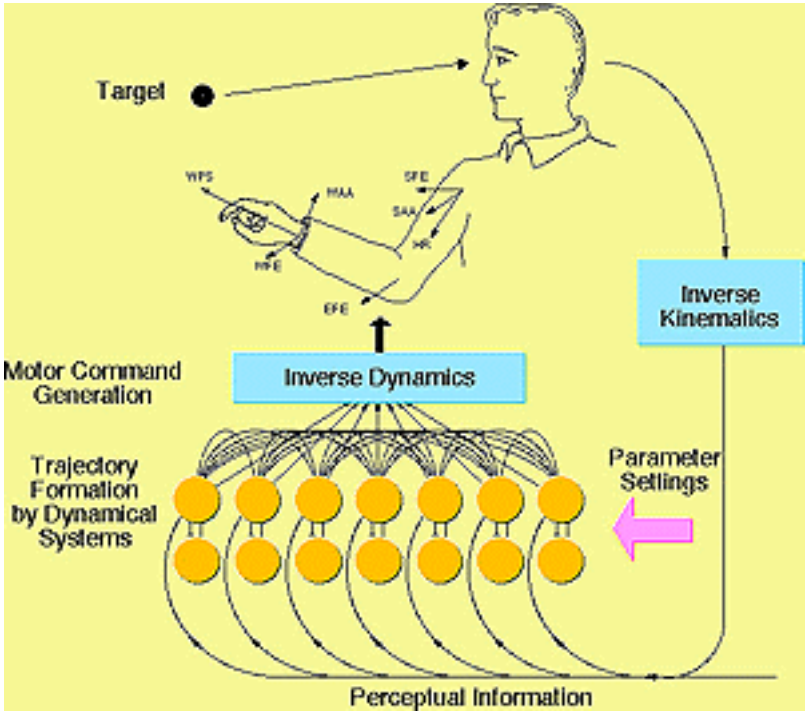
$$f(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2$$

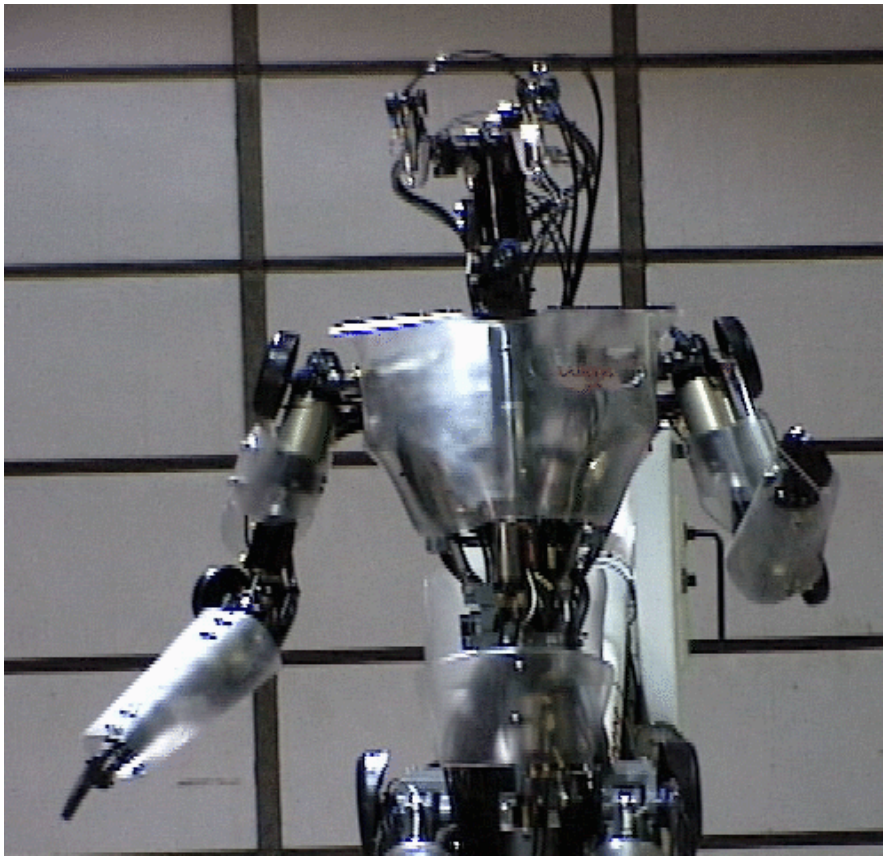
$$f(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2$$

$$f(\mathbf{x}) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2 + w_5 x_1 x_2$$

Interaction term

Regression for control



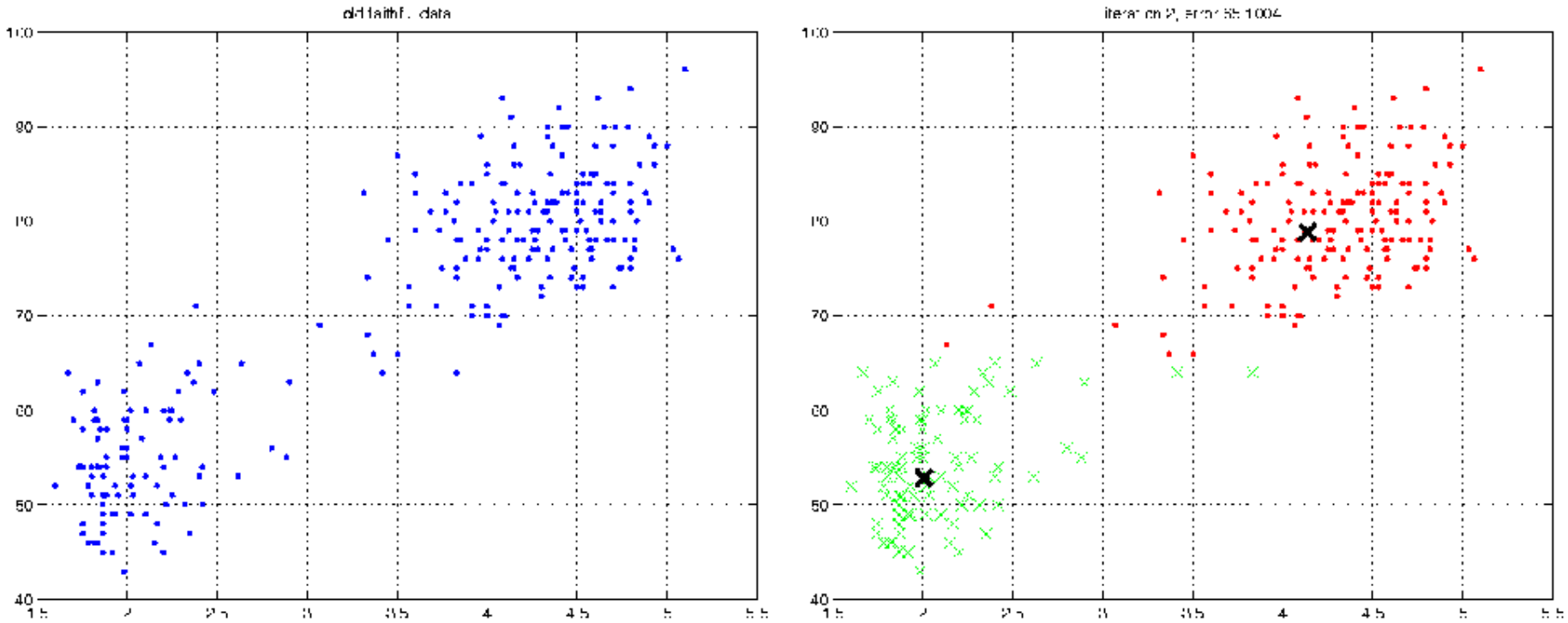


http://www-clmc.usc.edu/Research/HumanoidRobotics

Outline

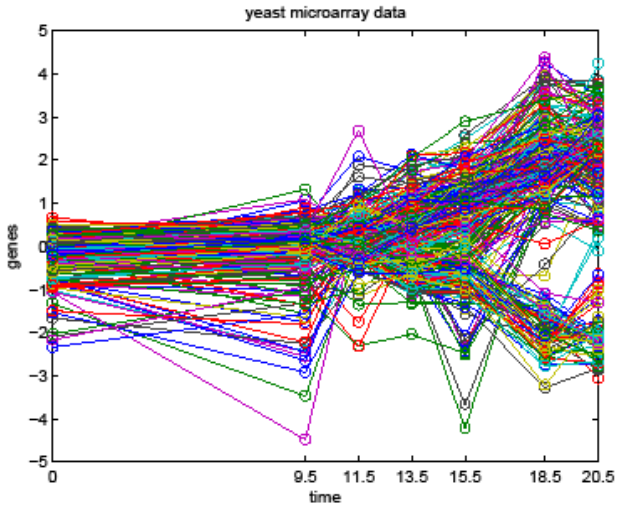
- Administrivia
- Overview
- Supervised learning
- Unsupervised learning
- Other kinds of learning

Clustering

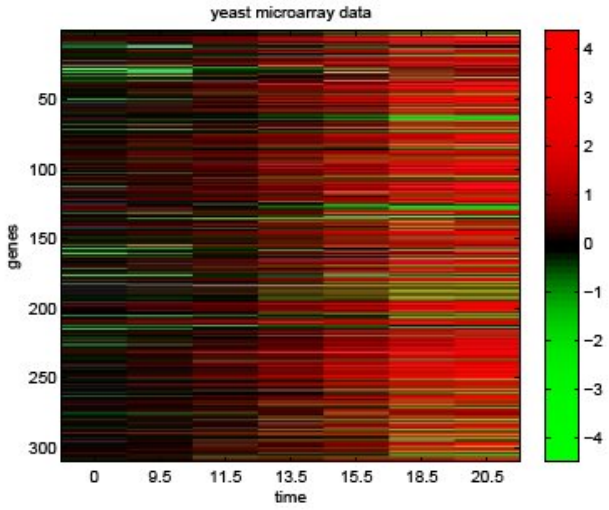


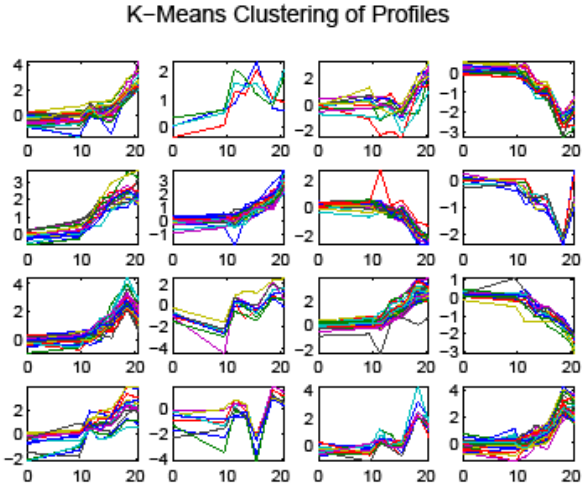
K-means after 2 iterations

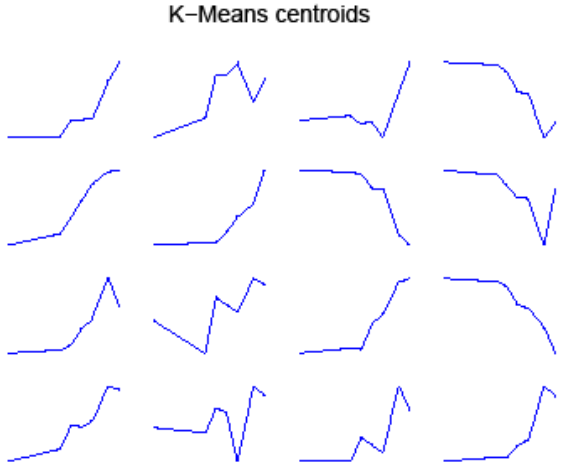
Clustering genes



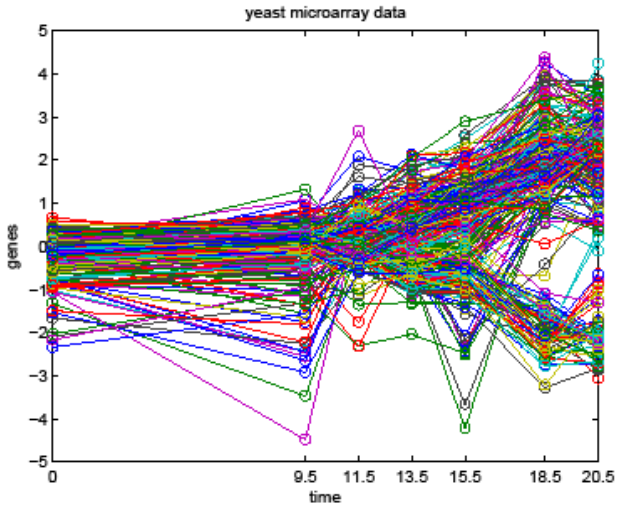
310x7

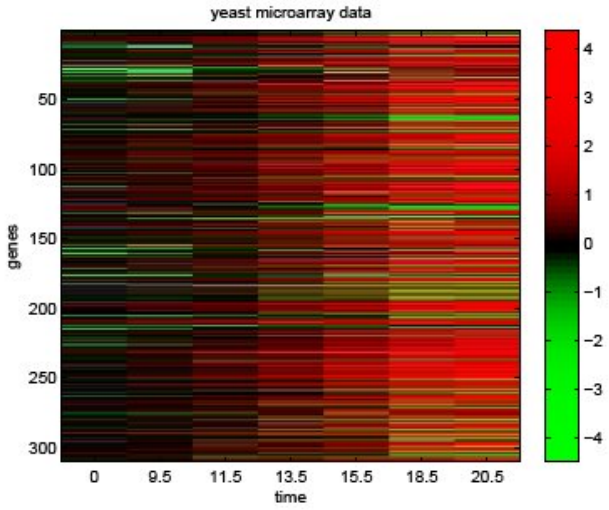


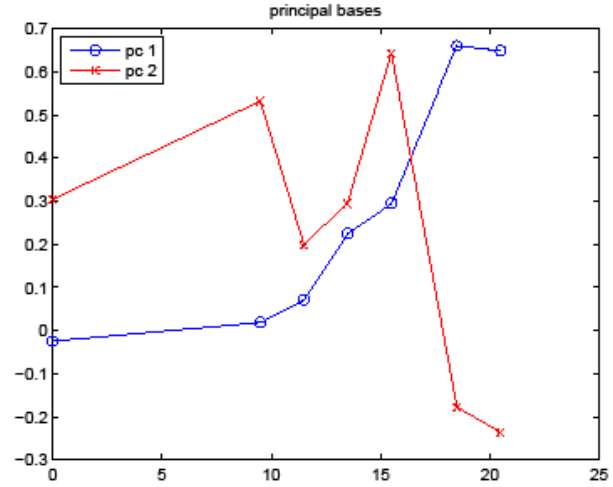


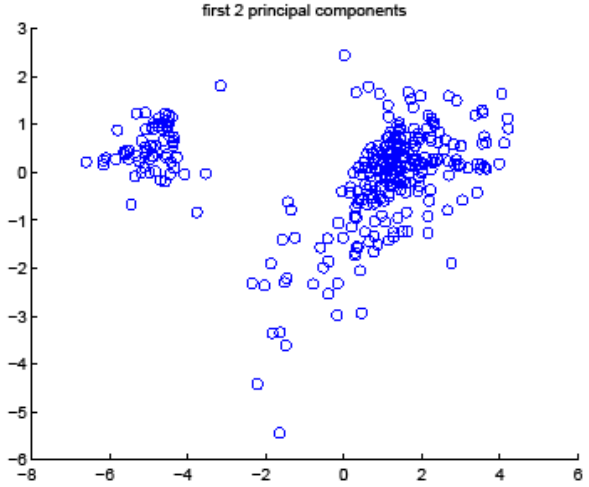


PCA



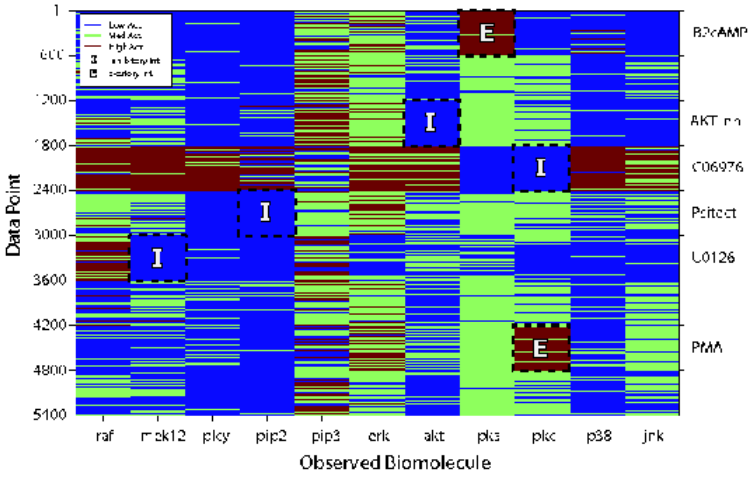




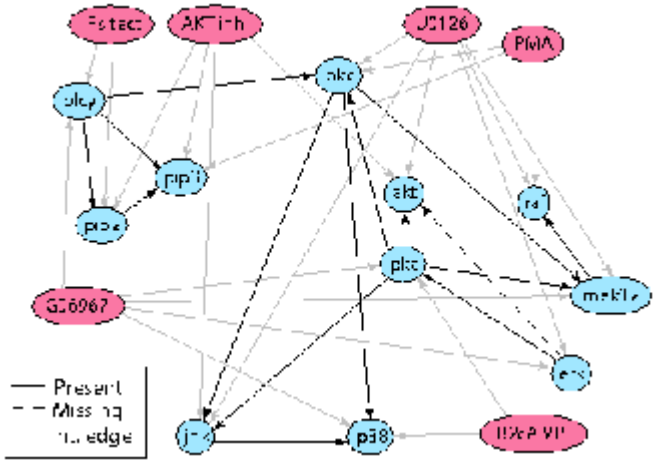


Principal components analysis

Learning graph structures



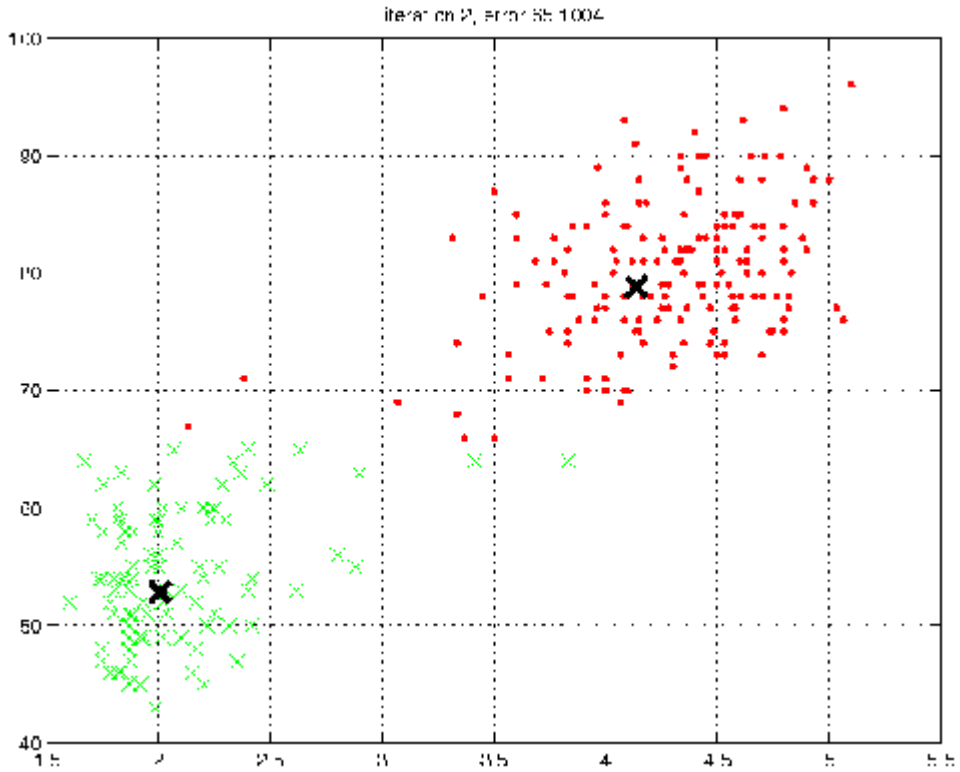
Protein phosphorylation data



DAG model

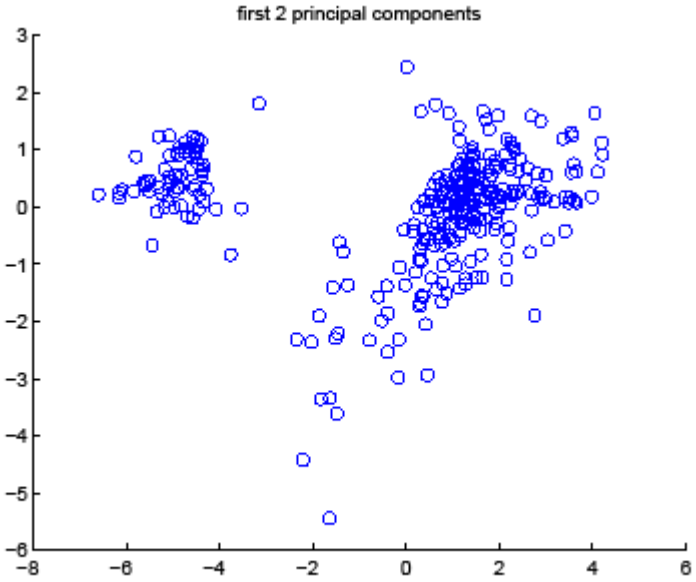
See Stat521A Spring 2009

Assessing unsupervised learning



2 clusters or 3?

Assessing unsupervised learning



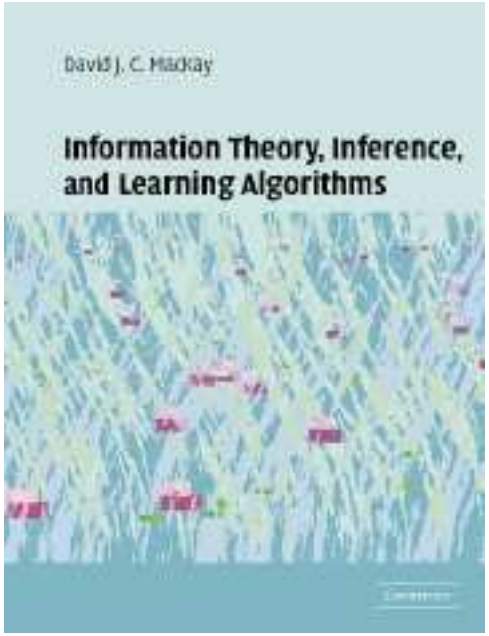
2 dimensions or more?
Linear subspace or something else?

Density estimation

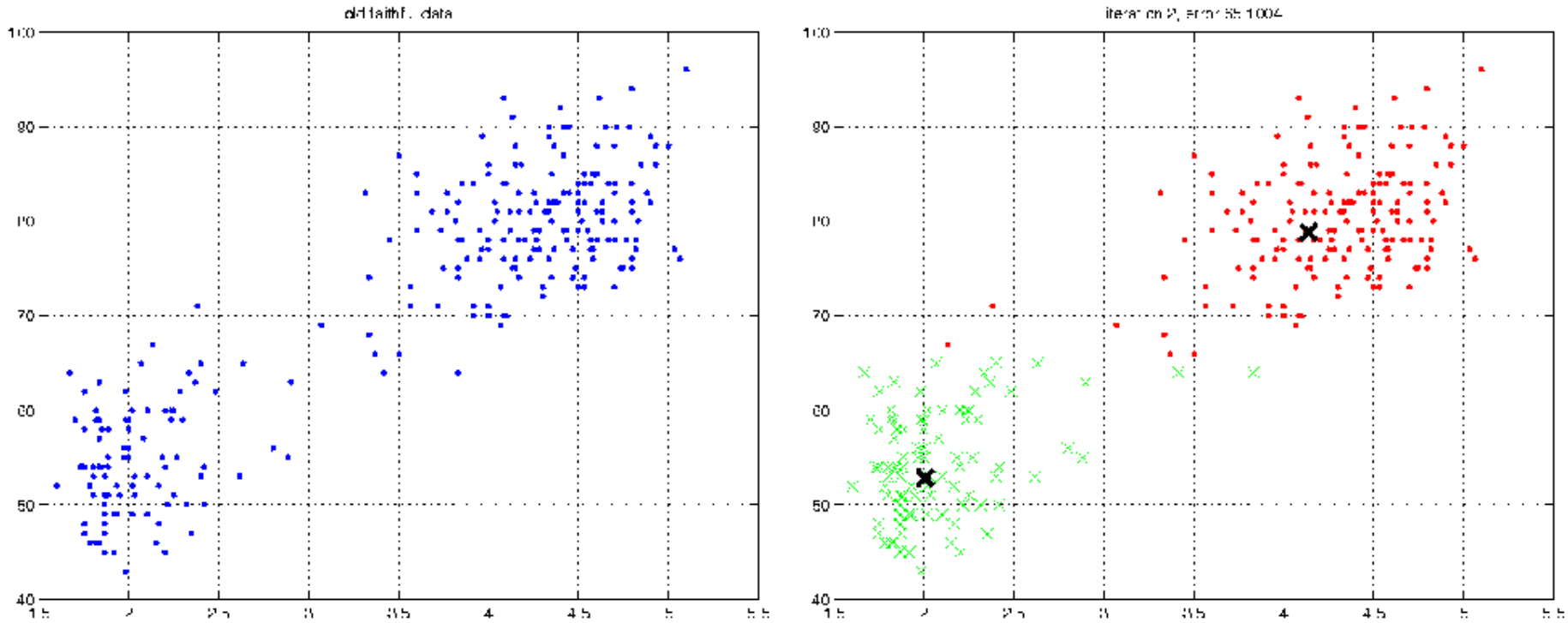
- Can formalize unsupervised learning as learning a model of p(y) instead of p(y|x)
- Model should assign high probability to future data
- If we generate from the model, it should look like the observed data
- If we have too many clusters, it will overfit (see next lecture)
- If we have too few clusters, it will underfit (see next lecture)
- Choosing K is an example of model selection

Data compression

- In the information theory chapter, we show that finding a good data compression scheme relies on building an accurate probabilistic model of the data.
- Frequent data vectors get assigned short code-words (fewer bits required).
- Infrequent data vectors can be given long code-words.
- See Mackay's book



Vector quantization

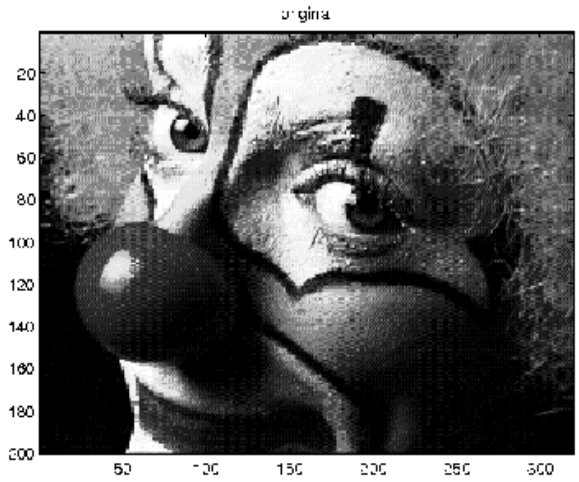


Replace each x_i 2 R^2 with a codeword z_i in {1,..,K}

This is an index into the codebook m_1, m_2, …, m_K in R^2

K-means minimizes the distortion

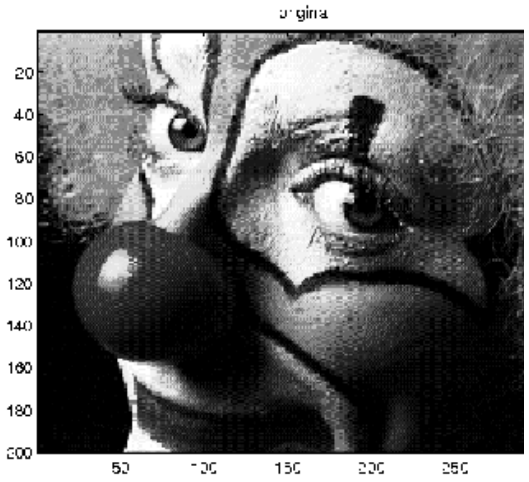
$$J = \frac{1}{n}\sum_{i=1}^{n} ||\mathbf{x}_i - \text{decode}(\text{encode}(\mathbf{x}_i))||^2 = \frac{1}{n}\sum_{i=1}^{n} ||\mathbf{x}_i - \mathbf{m}_{z_i}||^2$$

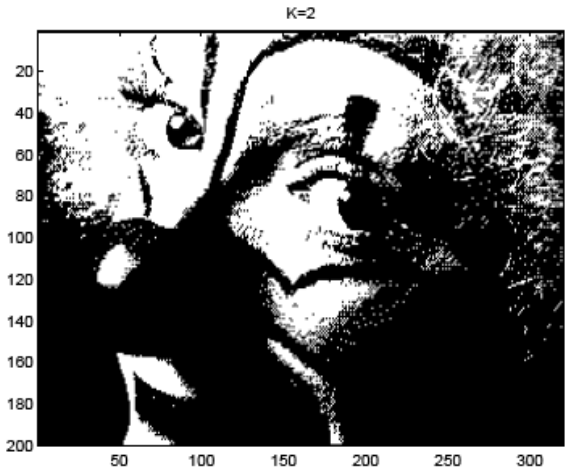


Original K=2 K=4

K-means minimizes the distortion

$$J = \frac{1}{n}\sum_{i=1}^{n} ||\mathbf{x}_i - \text{decode}(\text{encode}(\mathbf{x}_i))||^2 = \frac{1}{n}\sum_{i=1}^{n} ||\mathbf{x}_i - \mathbf{m}_{z_i}||^2$$





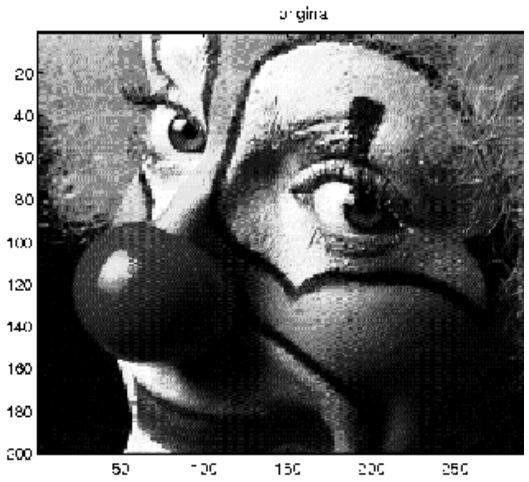
Original

K=2

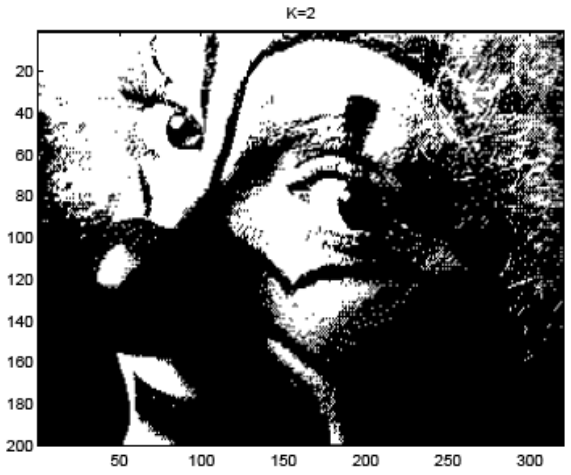
K=4

K-means minimizes the distortion

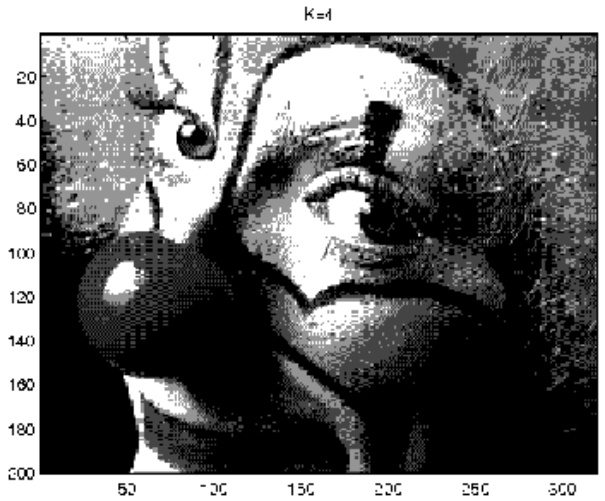
$$J = \frac{1}{n}\sum_{i=1}^{n} ||\mathbf{x}_i - \text{decode}(\text{encode}(\mathbf{x}_i))||^2 = \frac{1}{n}\sum_{i=1}^{n} ||\mathbf{x}_i - \mathbf{m}_{z_i}||^2$$



Original



K=2



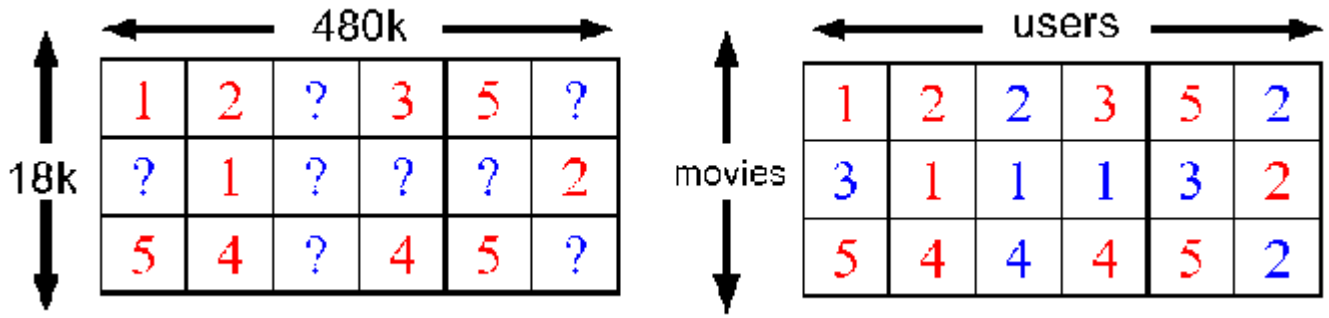
K=4

Outline

- Administrivia
- Overview
- Supervised learning
- Unsupervised learning
- Other kinds of learning

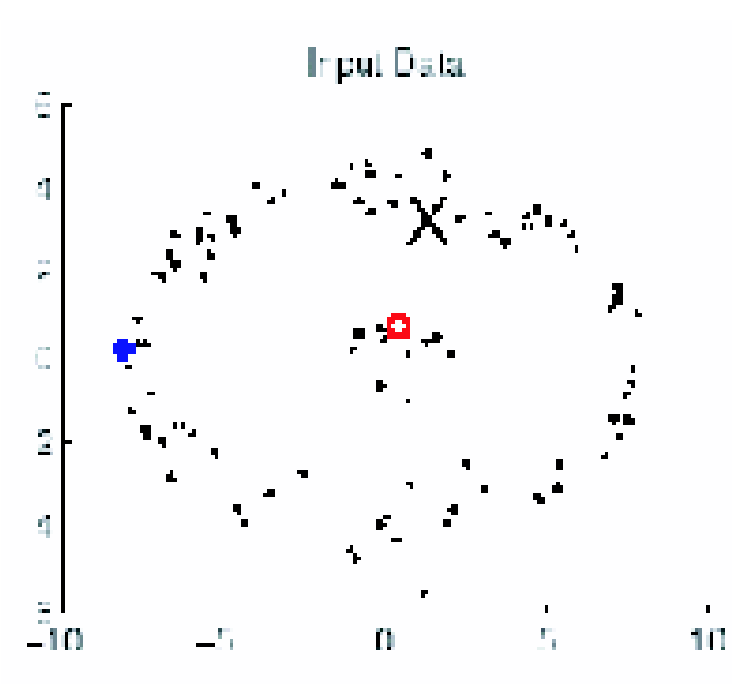
Collaborative filtering

www.netflixprize.com $1M USD

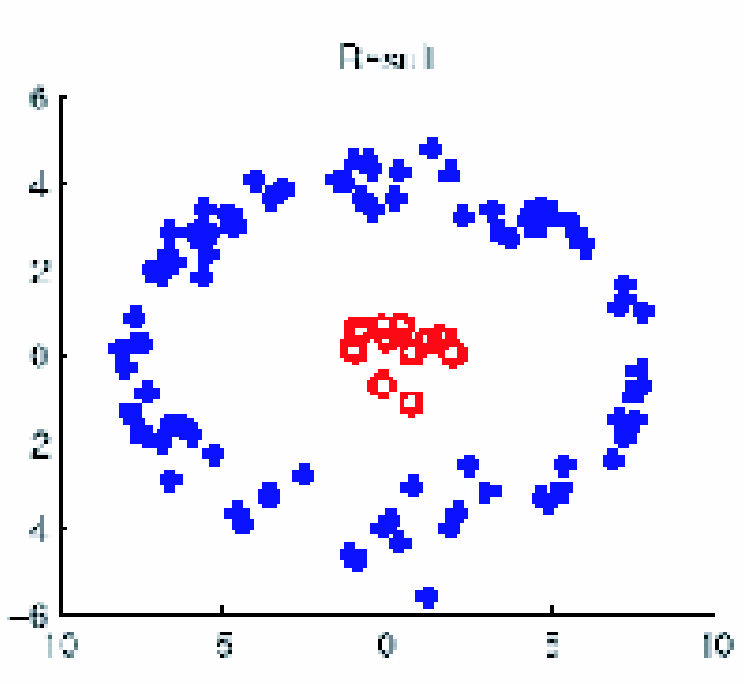


$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(R(u_i, m_i) - \hat{R}(u_i, m_i))^2}$$

Semi-supervised learning

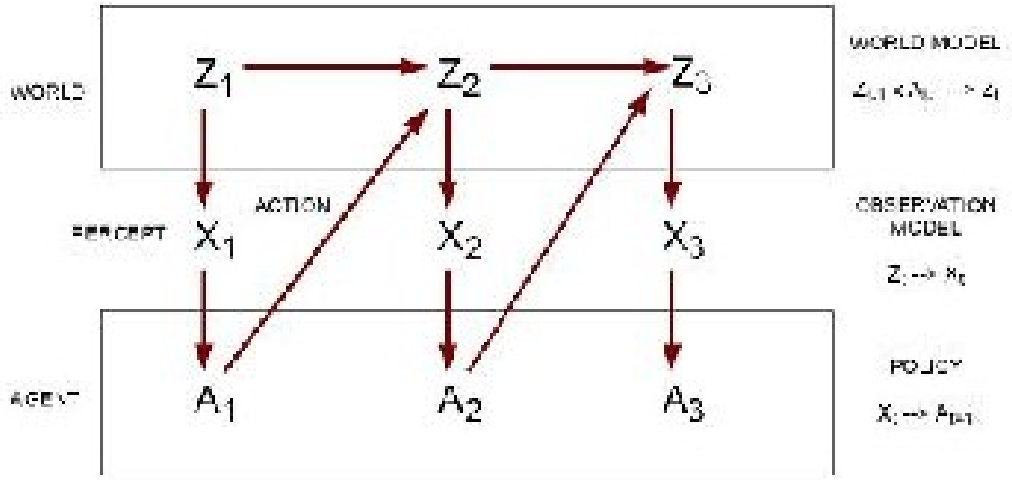


2 labeled, 1000s unlabeled



Propagate y labels
to “similar” x’s

Reinforcement learning



Search over actions to maximize expected utility:
- Predict effects of actions using probabilistic model
- Use utility theory to decide which outcome is best

-RL tries to learn a controller that simulates the above behavior

See CS322 and CS502