Introduction

ddebarr@uw.edu

2016-04-07

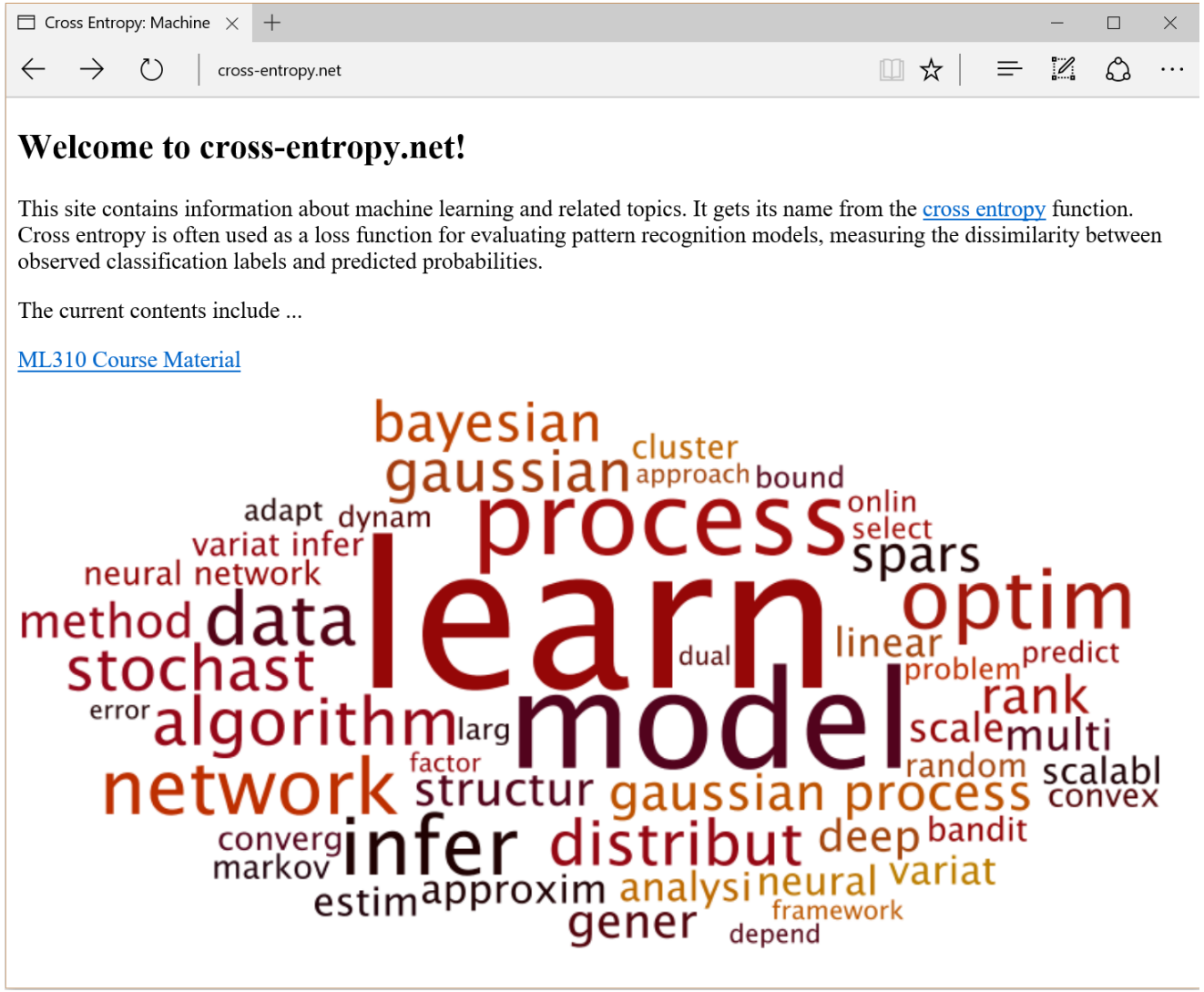
Administrative Stuff

- Pre-requisites: calculus, linear algebra
- Attendance: must attend 80% of classes
- On-site versus online: on-site students can do one online session [licensing]
- Homework: posted by Fri @ 11:59pm; due the next Fri @ 11:59pm
- Grading: must have 80% of homework graded as pass
- External Course Website: http://cross-entropy.net/ML310

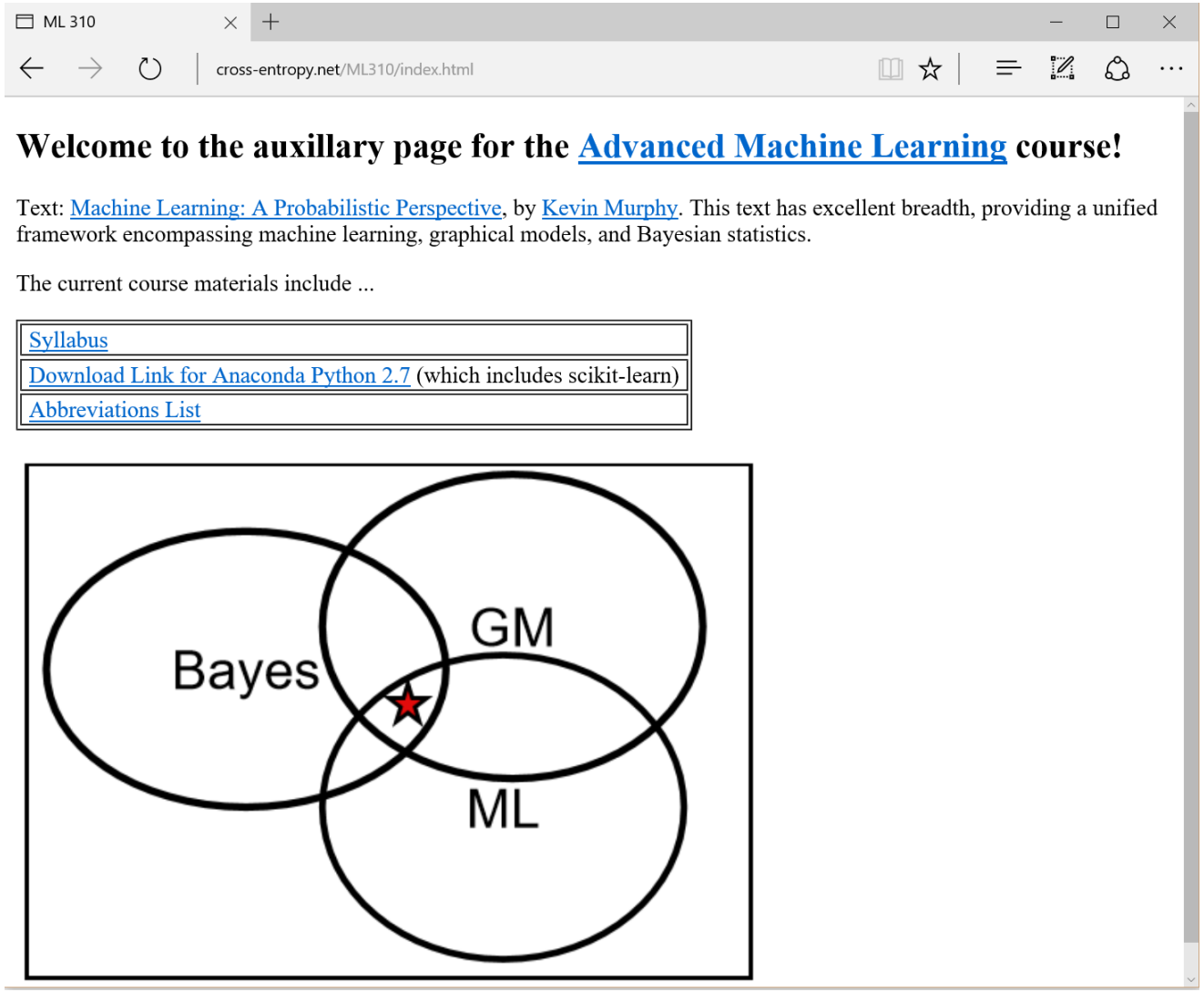
Course Outline

- Apr 7:
  - Chapter 1: Introduction
  - Python and scikit-learn
    - Interfaces
    - Model selection
- Apr 14
  - Chapter 2: Probability
  - Spectral clustering
    - Spectral representation
    - Clustering
- Apr 21
  - Chapter 3: Generative models for discrete data
  - Recommendation systems
    - Collaborative filtering
    - Content filtering
- Apr 28
  - Chapter 4: Gaussian models
  - Natural language processing
    - Bag of words
    - Topic modeling
- May 5
  - Chapter 5: Bayesian statistics
  - Imbalanced classification
    - Weights
    - Sampling
- May 12
  - Chapter 6: Frequentist statistics
  - Graphical models
    - Bayesian networks
    - Conditional random fields
- May 19
  - Chapter 7: Linear regression
  - Semi-supervised learning
    - Self-training
    - Co-training
    - Label propagation
- May 26
  - Chapter 8: Logistic regression
  - Active learning
    - Exploration
    - Exploitation
- Jun 2
  - Chapter 16: Adaptive basis function models
  - Online learning
    - Online gradient descent
    - Bandits
- Jun 9
  - Chapter 28: Deep learning
  - Introduction to deep learning
    - Multi-layer perceptron
    - Representation learning

External Course Website



External Course Website



Facts101?

Most classifiers assume that the input vector $\mathbf{x}$ has a fixed size. A common way to represent variable-length documents in feature-vector format is to use a **bag of words** representation. This is explained in detail in Section 3.4.4.1, but the basic idea is to define $x_{ij} = 1$ iff word j occurs in document i. If we apply this transformation to every document in our data set, we get a binary document $\times$ word co-occurrence matrix: see Figure 1.2 for an example. Essentially the

Co-occurrence matrix:

A co-occurrence matrix or co-occurrence distribution is a matrix or distribution that is defined over an image to be the distribution of co-occurring values at a given offset. Mathematically, a co-occurrence matrix C is defined over an n × m image I, parameterized by an offset (Δx,Δy), as:

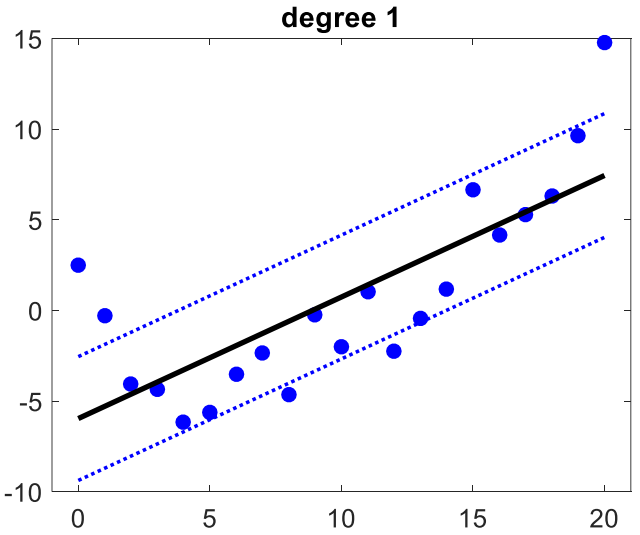
$$C_{\Delta x,\Delta y}(i,j) = \sum_{p=1}^{n} \sum_{q=1}^{m} \begin{cases} 1, & \text{if } I(p,q) = i \text{ and } I(p+\Delta x, q+\Delta y) = j \\ 0, & \text{otherwise} \end{cases}$$

where i and j are the image intensity values of the image, p and q are the spatial positions in the image I and the offset (Δx,Δy) depends on the direction used θ and the distance at which the matrix is computed d. The 'value' of the image originally referred to the grayscale value of the specified pixel, but could be anything, from a binary on/off value to 32-bit color and beyond.

Probability Modeling Tool Kit (PMTK)

- https://www.mathworks.com/store/ [click “Student”?]
- https://github.com/probml/pmtk3 [click “Download Zip”]

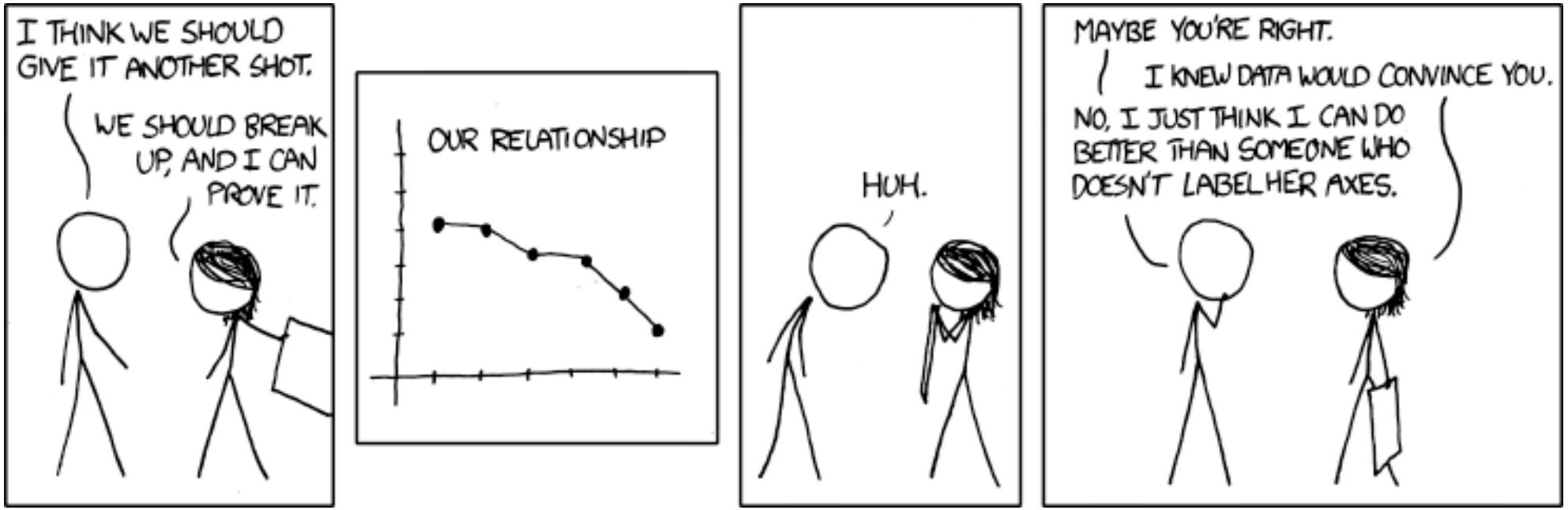
```
Command Window
New to MATLAB? See resources for Getting Started.

    Home License -- for personal use only.  Not for government,
    academic, research, commercial, or other organizational use.

>> cd /projects/pmtk3-master
>> initPmtk3
initializing pmtk3
welcome to pmtk3
>> linregPolyVsDegree
fx >>
```



See Fig 1.7(a)

We interrupt our regularly scheduled broadcast for this important …

Public Service Announcement



https://xkcd.com/833/

Always label your axes!

Agenda

- Machine learning: what and why?
- Supervised learning
- Unsupervised learning
- Some basic concepts in machine learning

Machine Learning Definition

The process of using data to create a model, mapping one or more inputs to one or more outputs.

Supervised Learning Example:

$$\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N} \quad y \in \{0, 1\} \longrightarrow \hat{y} = \hat{f}(\mathbf{x}) = \underset{c=1}{\overset{C}{\operatorname{argmax}}}\, p(y = c|\mathbf{x}, \mathcal{D})$$

Q: What's better than one way to write the probability estimate?

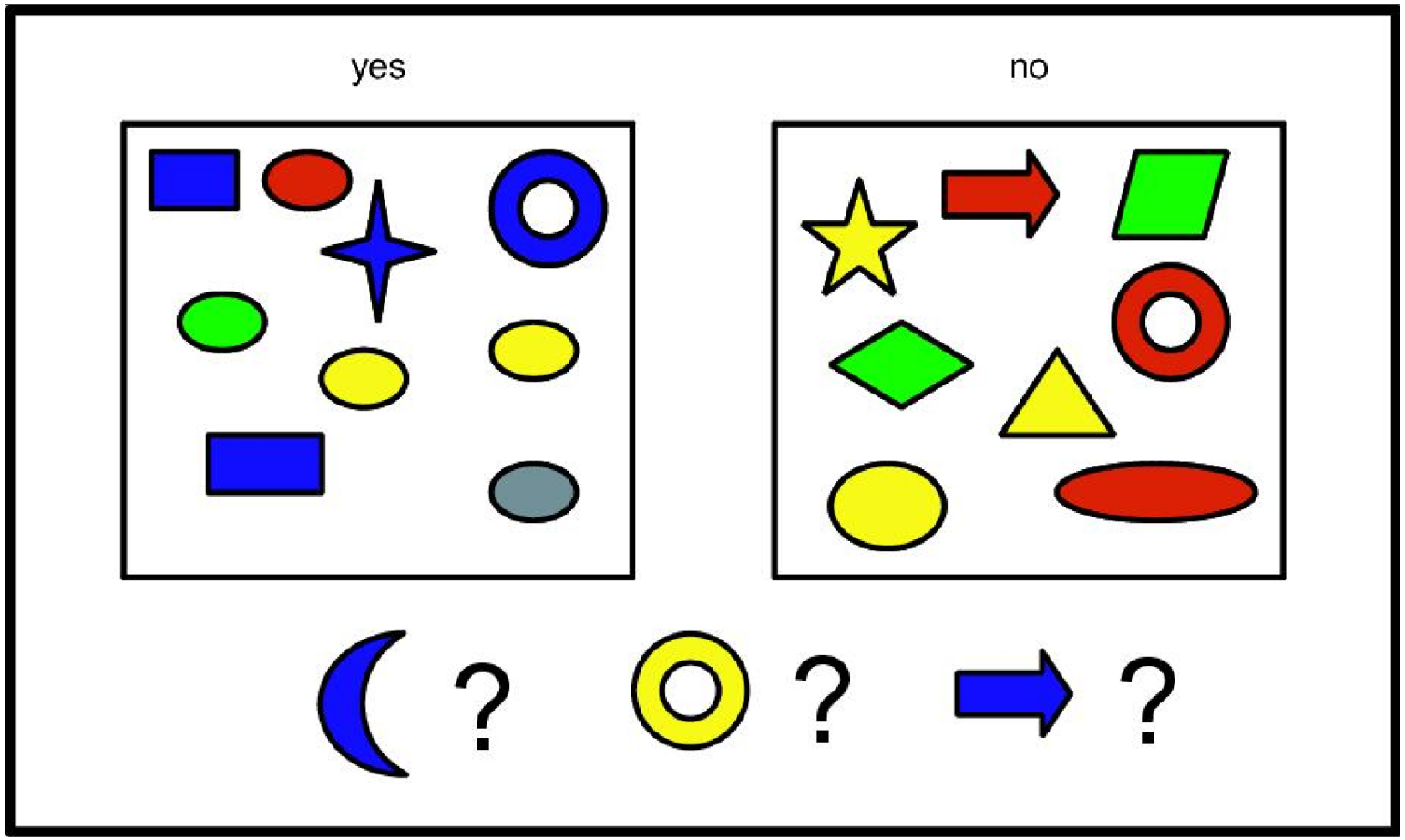
A: Four ways to write it! ☺

$$p(y|\mathbf{x}, \mathcal{D}, M) \quad \equiv \quad p(y|\mathbf{x}, \mathcal{D}) \quad \equiv \quad p(y|\mathbf{x}) \quad \equiv \quad p(y_i|\mathbf{x}_i, \boldsymbol{\theta})$$

Types of Machine Learning

- Supervised Learning
  - Regression
  - Classification
- Unsupervised Learning
  - Clustering
  - Matrix Completion (e.g. Collaborative Filtering and Market Basket Analysis)
- Reinforcement Learning
  - Games

Example Classification Task



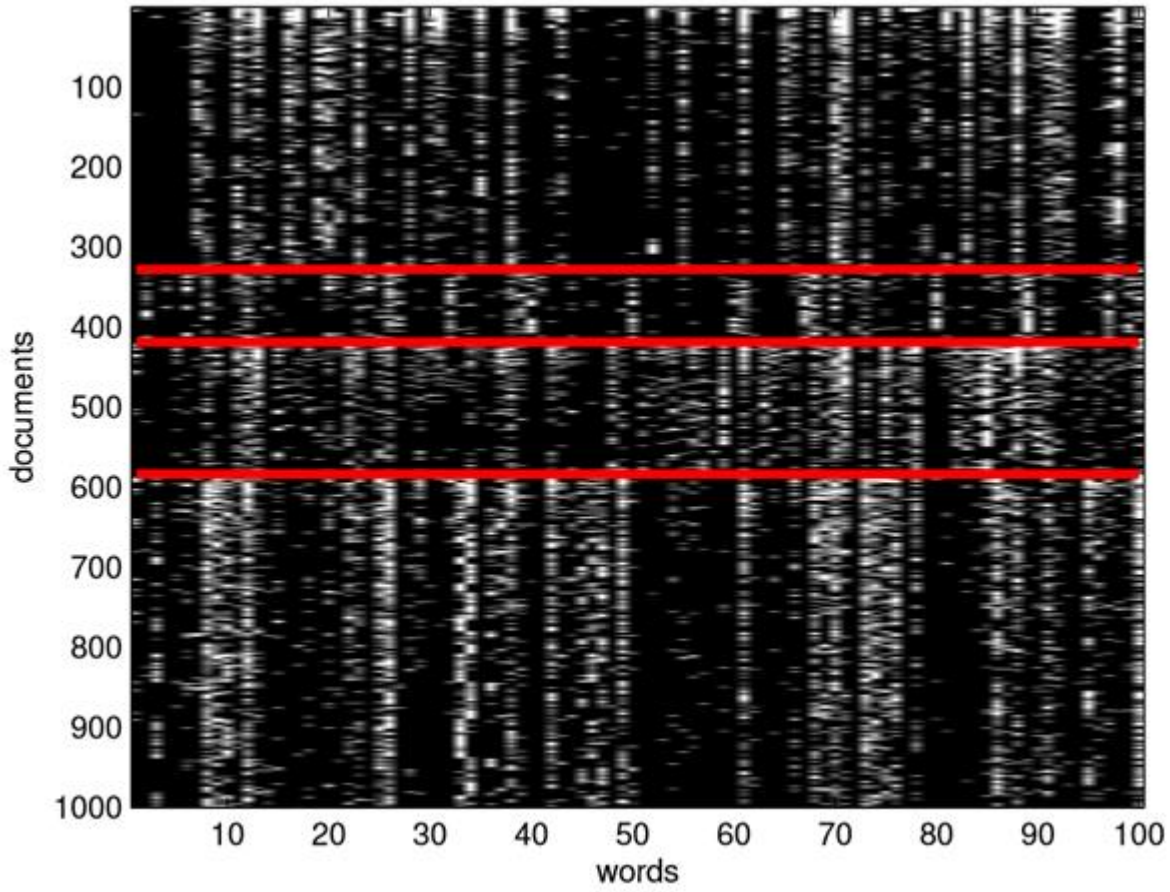
Example Representation

D features (attributes)

N cases

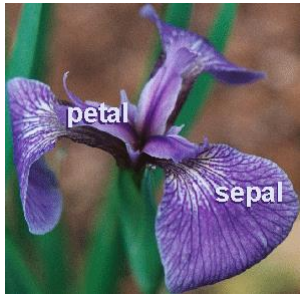
Color	Shape	Size (cm)	Label
Blue	Square	10	1
Red	Ellipse	2.4	1
Red	Ellipse	20.7	0

Document Classification



4 out of 20 newsgroups: comp, rec, sci, talk

Flower Classification

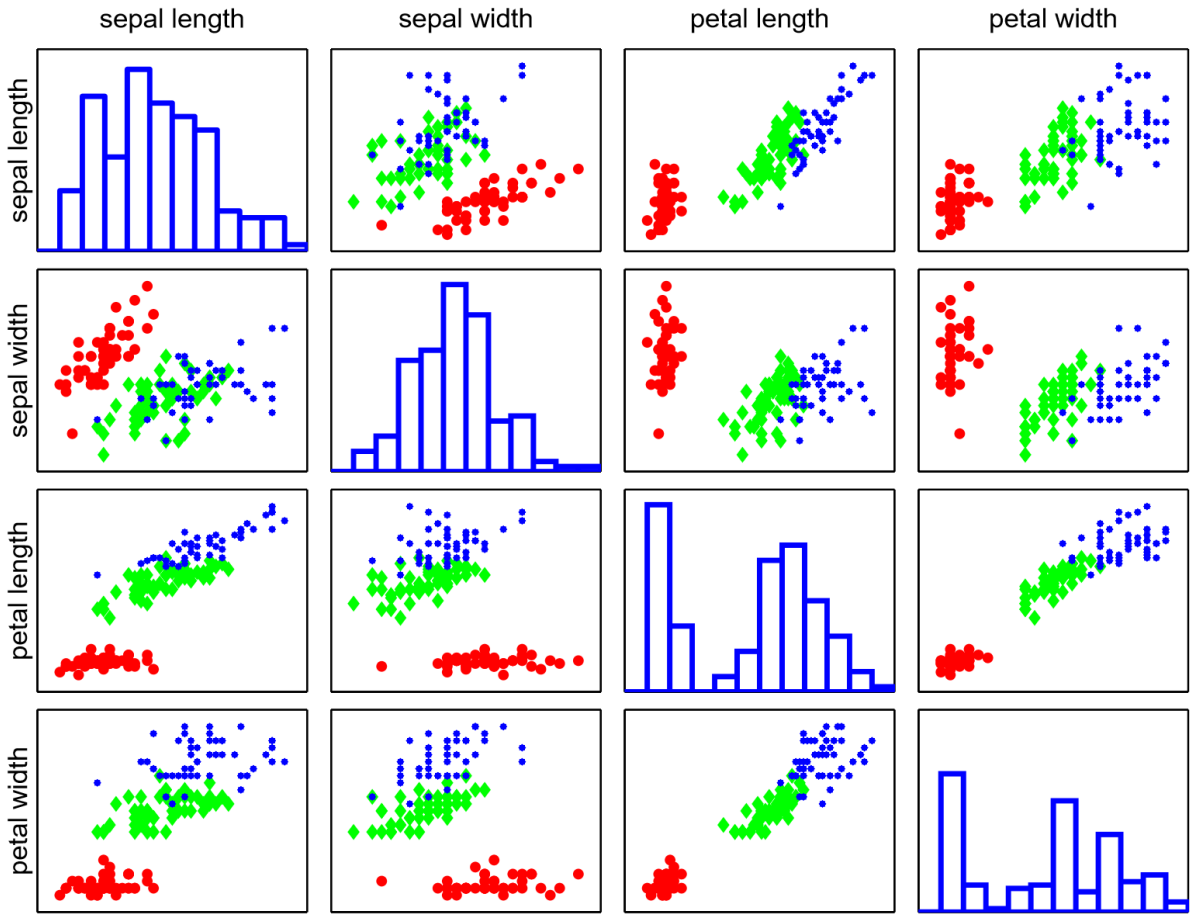


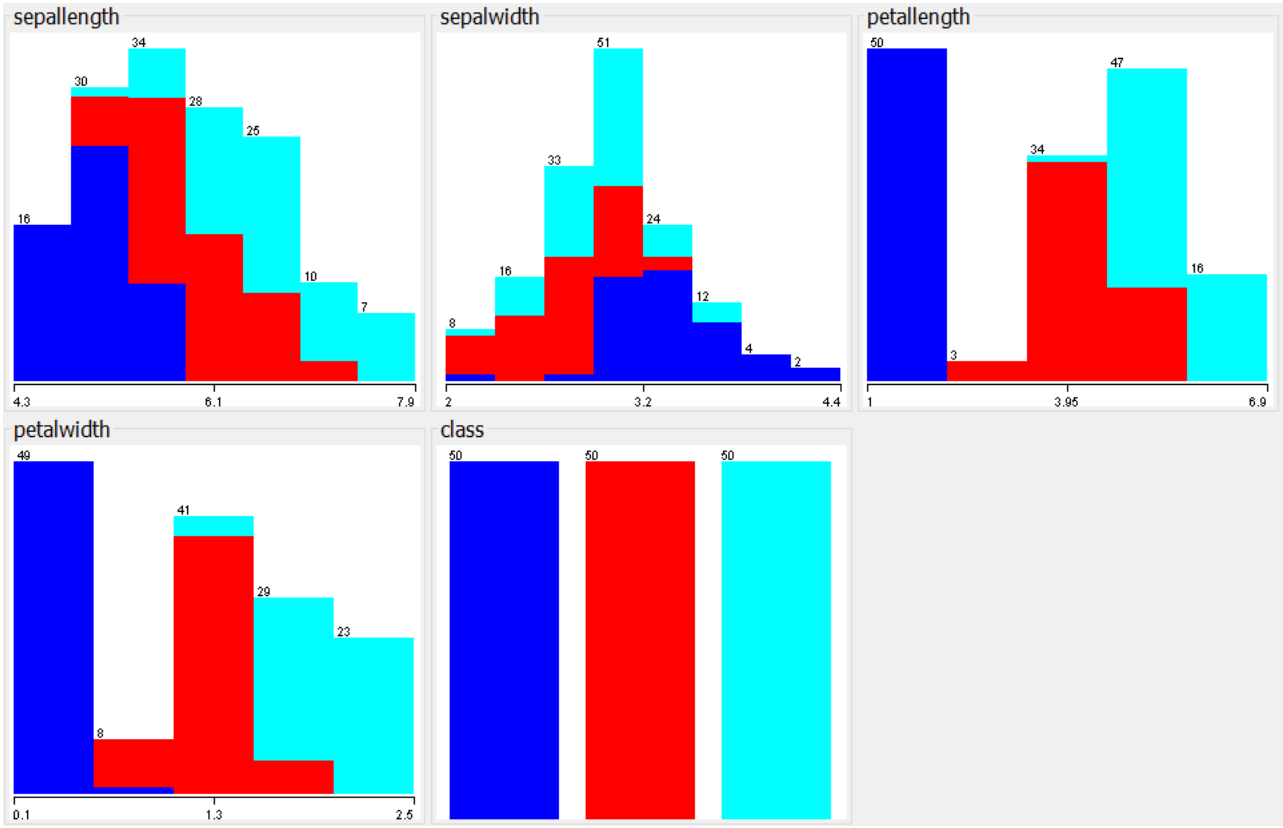
Setosa

Versicolor

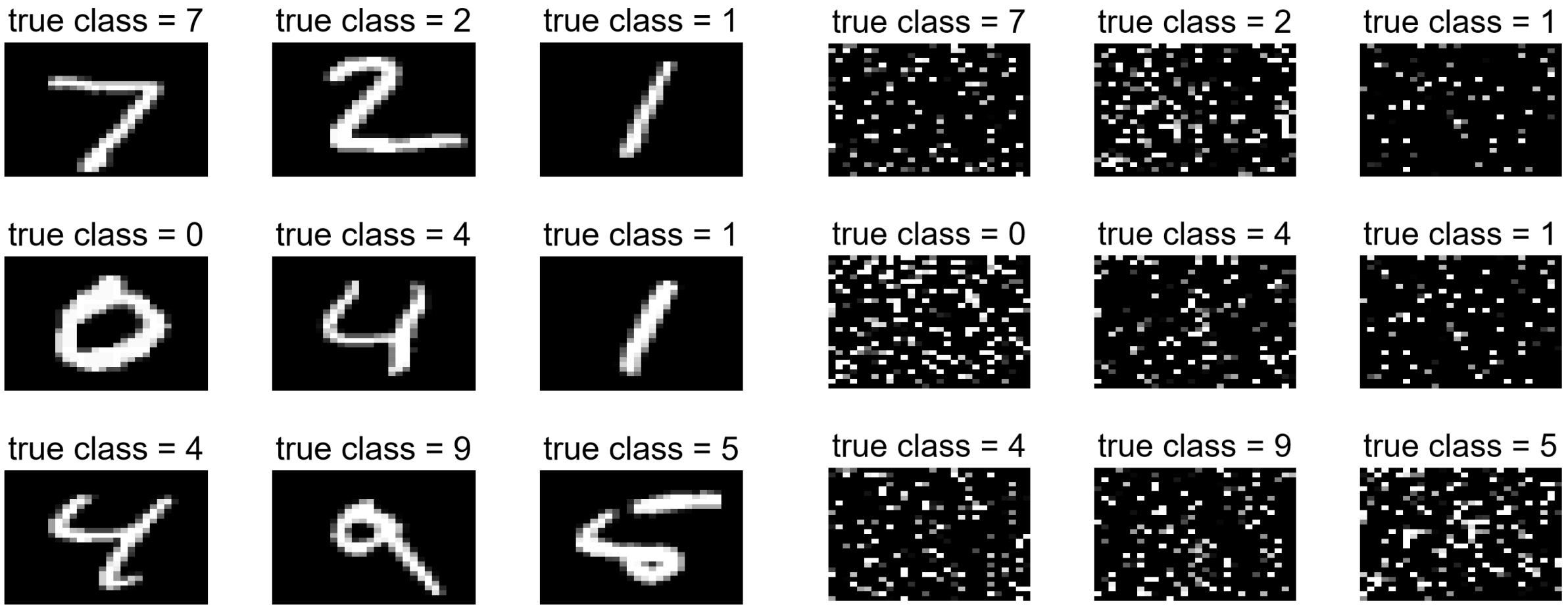
Virginica

Iris: Data Visualization

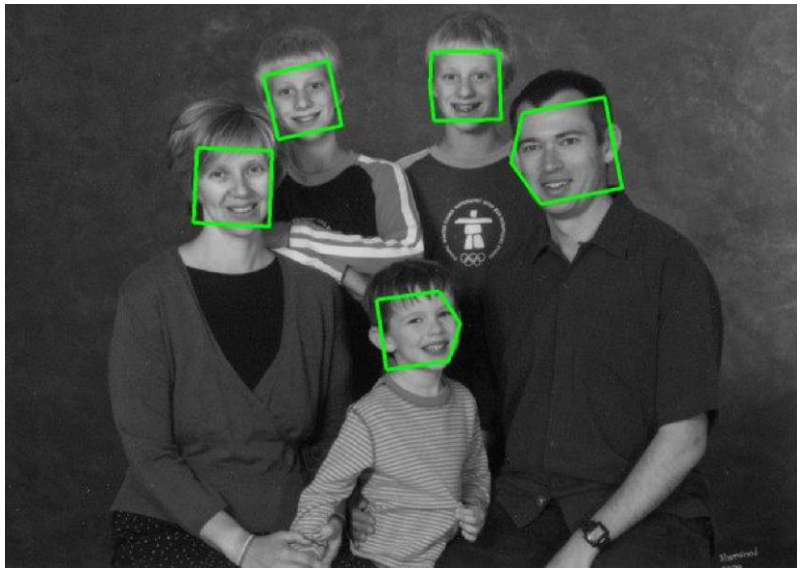




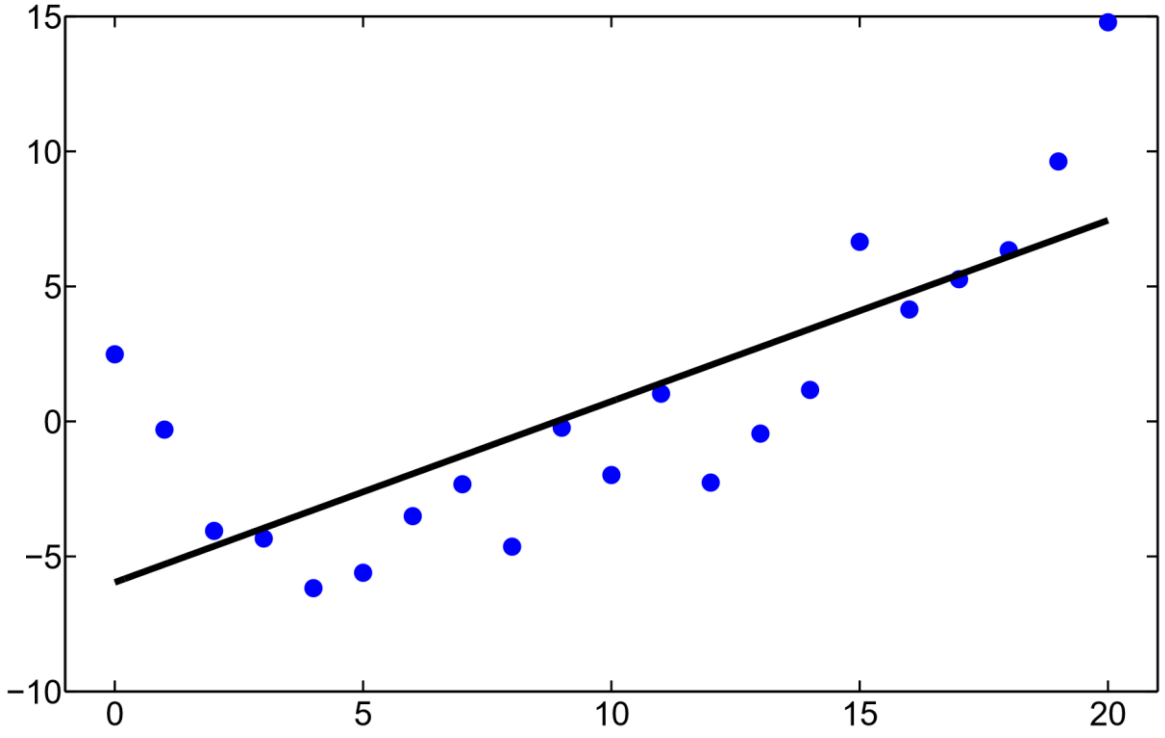
Handwriting Recognition

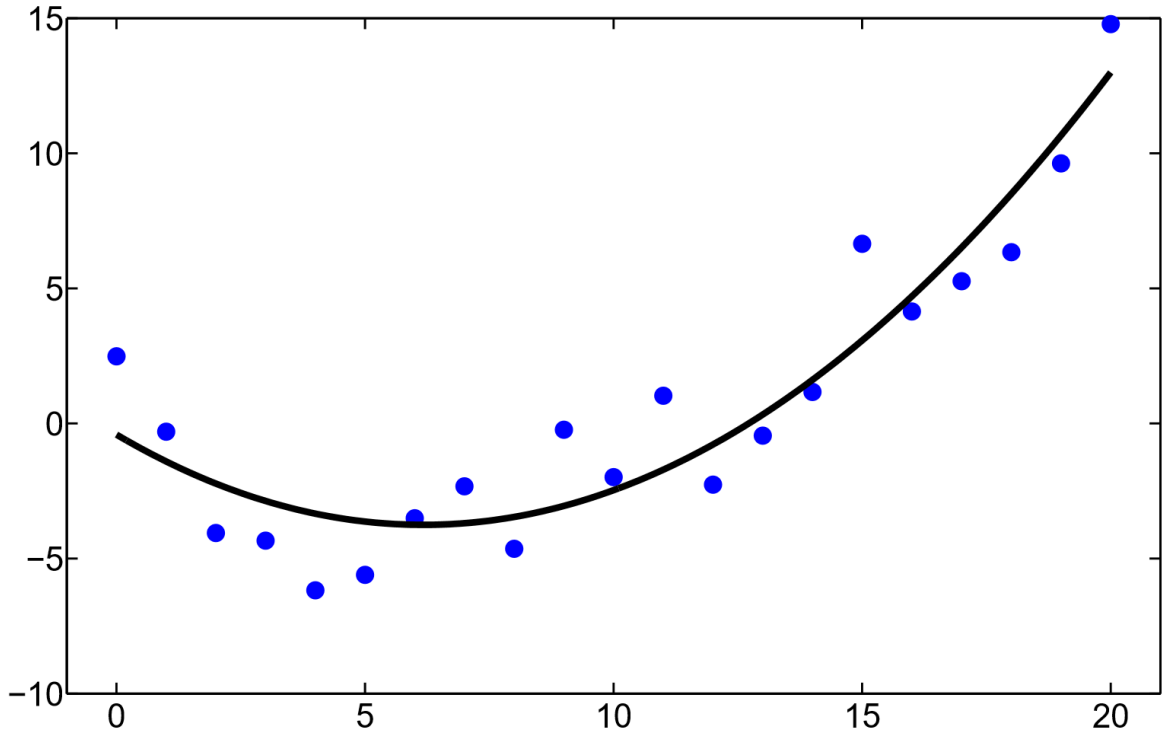


Face Detection

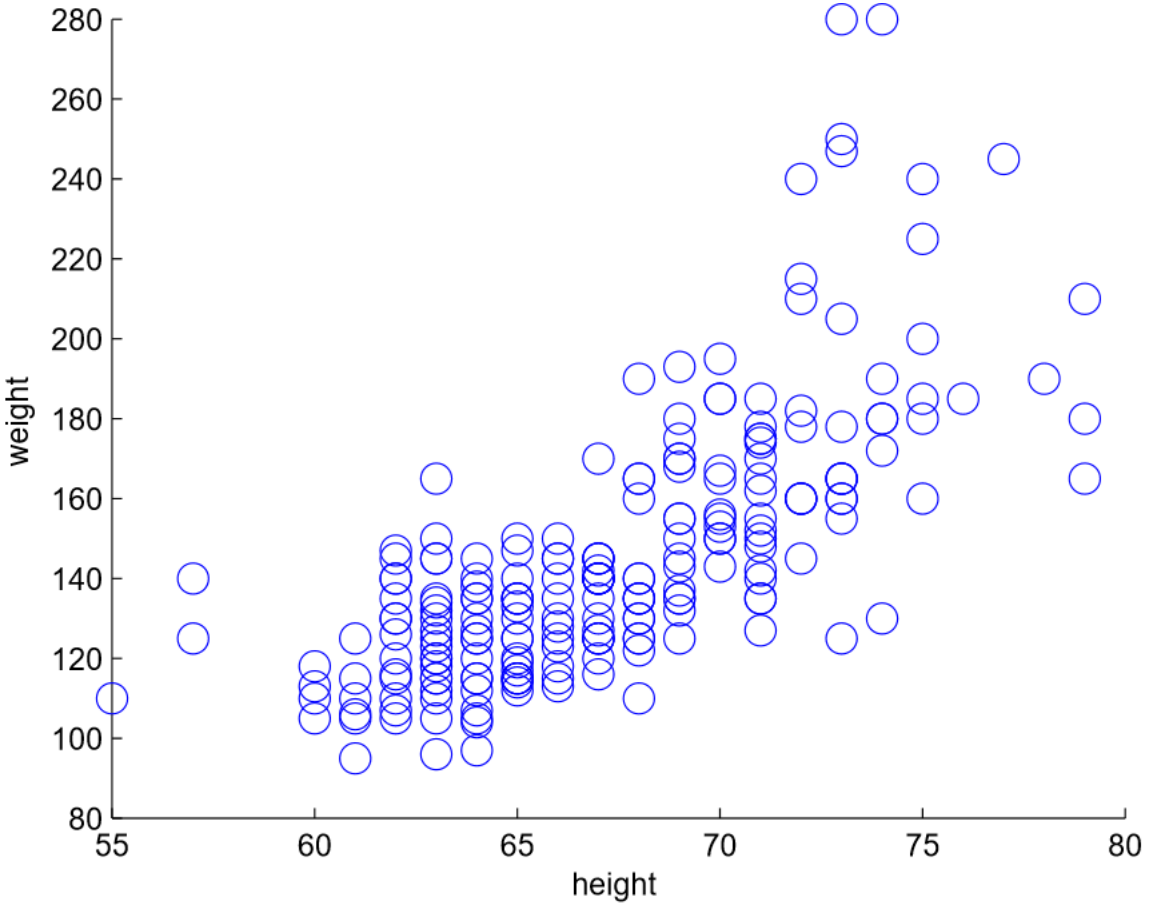


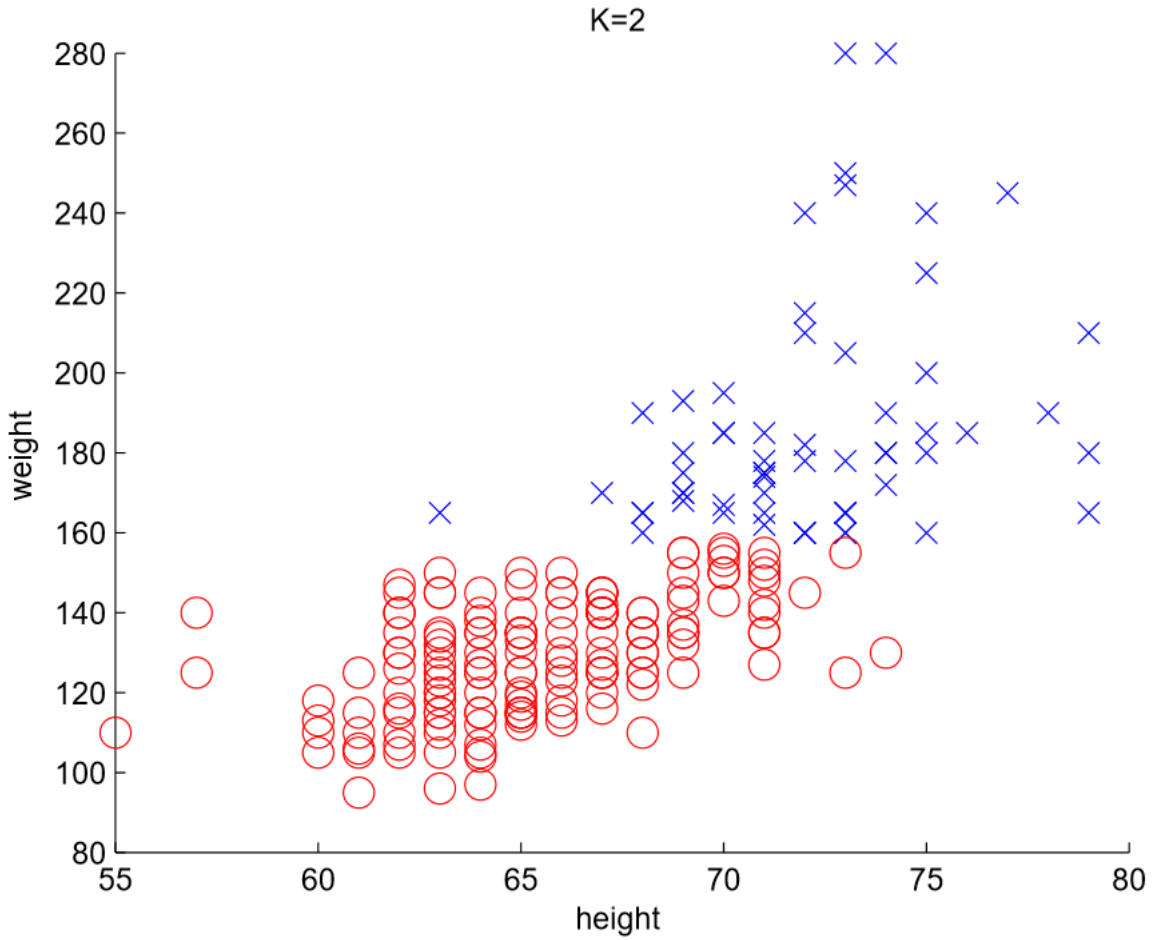
Linear versus Polynomial Regression





Discovering Clusters

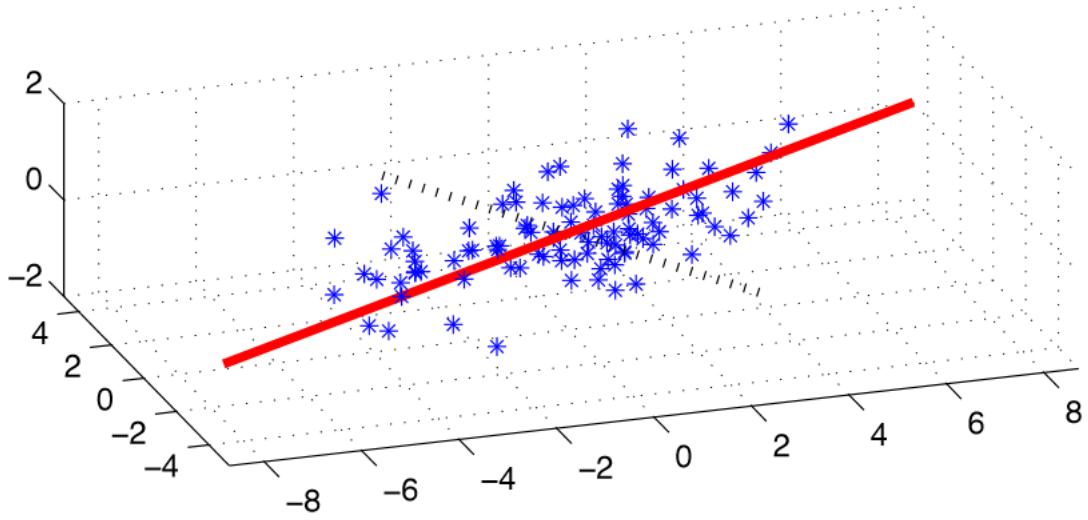


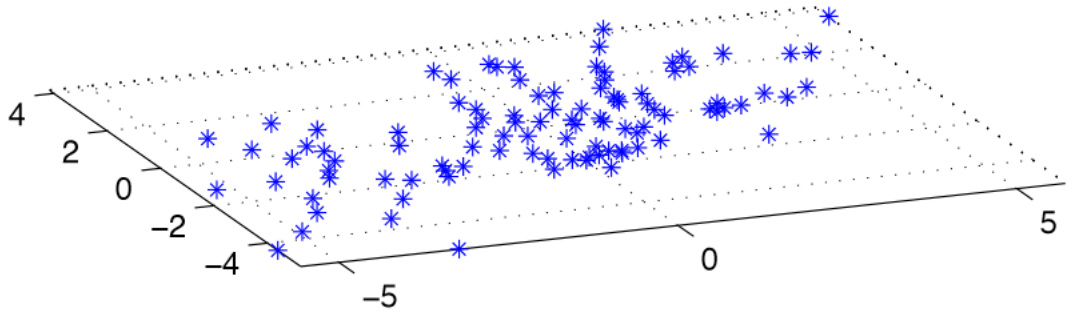


"supervised learning is conditional density estimation,
whereas unsupervised learning is unconditional density estimation"

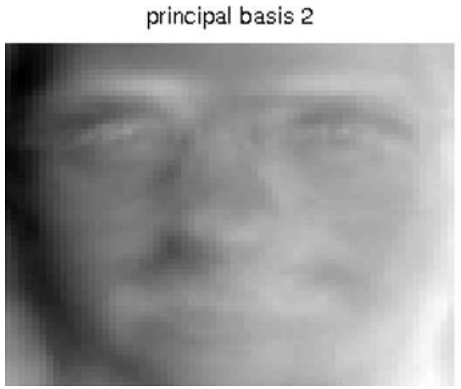
$$p(\mathbf{x}_i|\boldsymbol{\theta})$$

Discovering Latent Factors

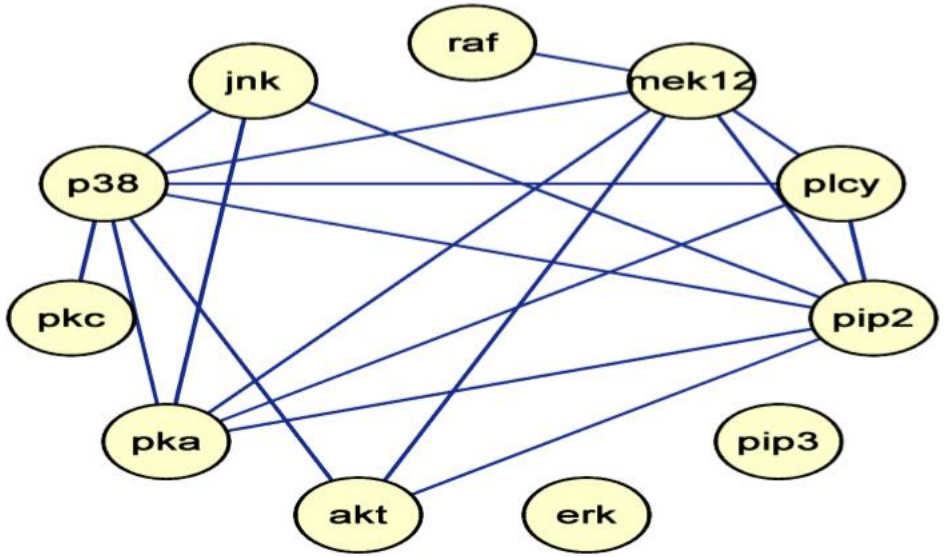




Principal Components for Faces



Discovering Graph Structure



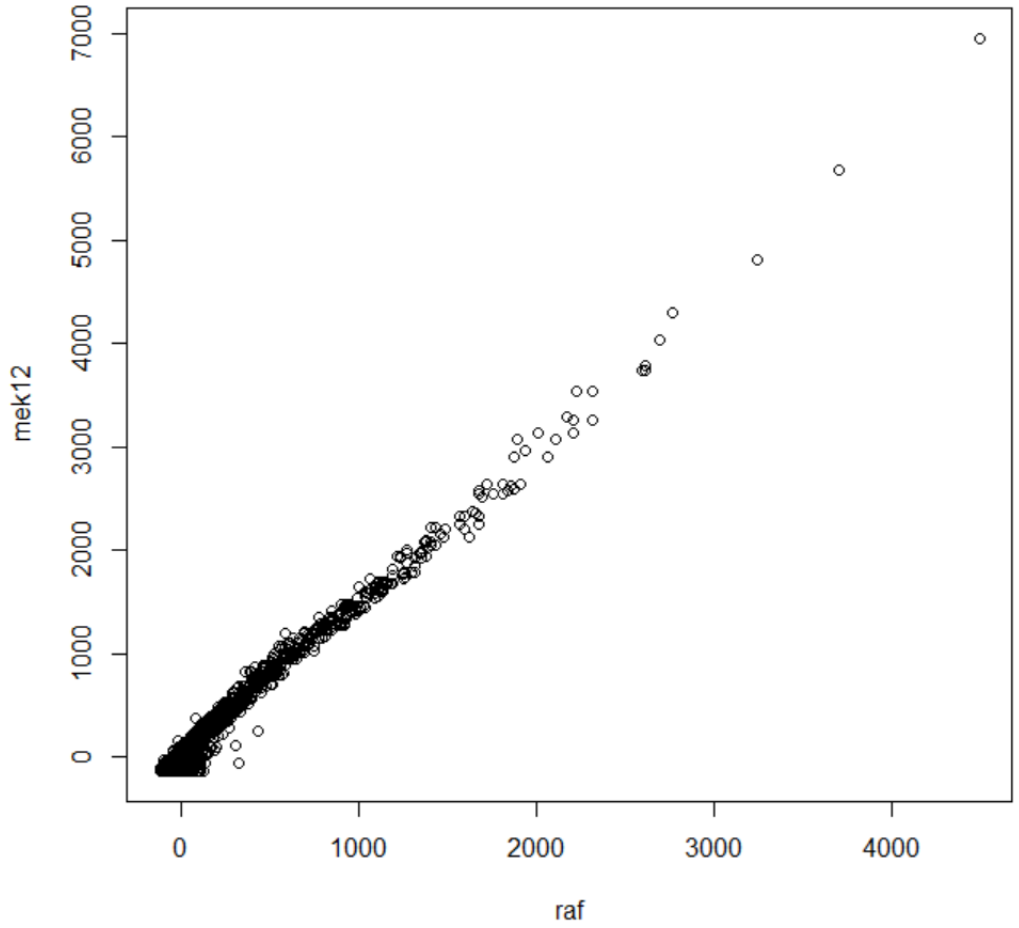
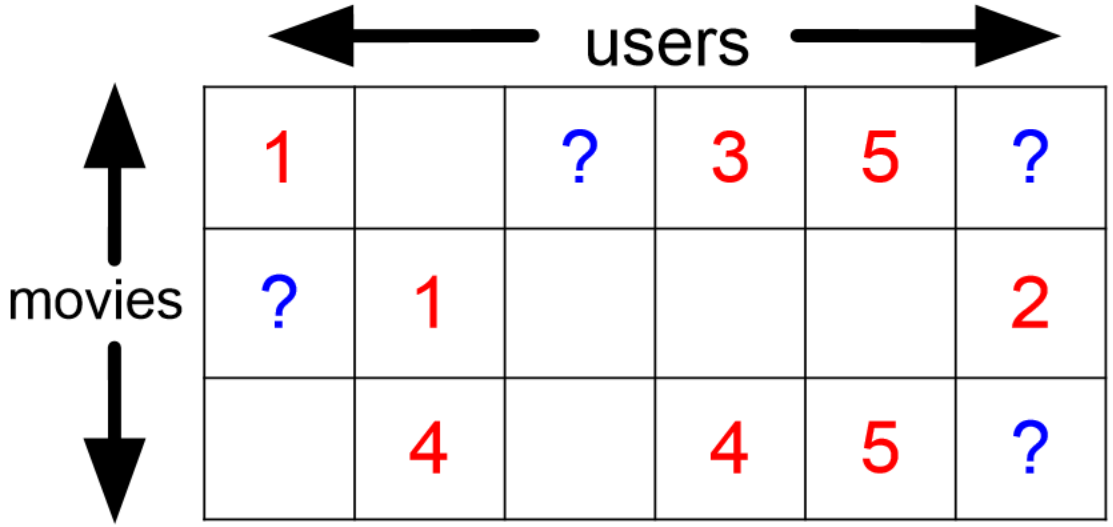


Image Inpainting

Collaborative Filtering

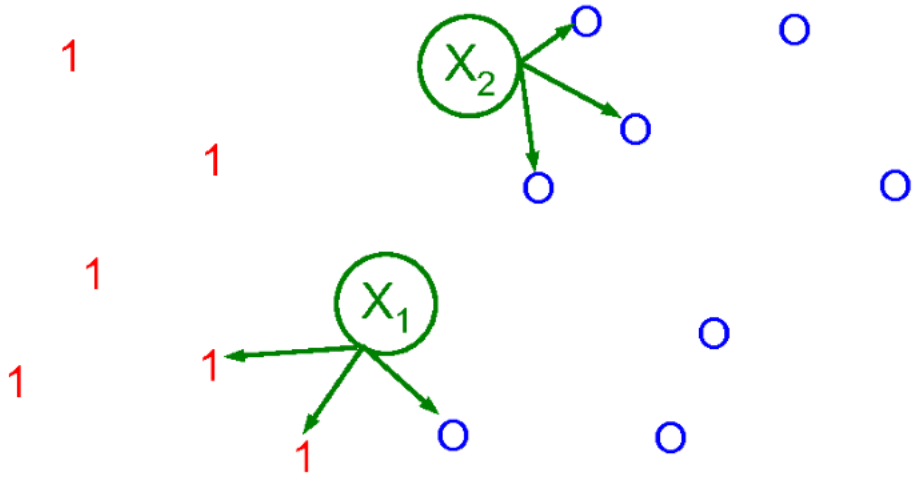


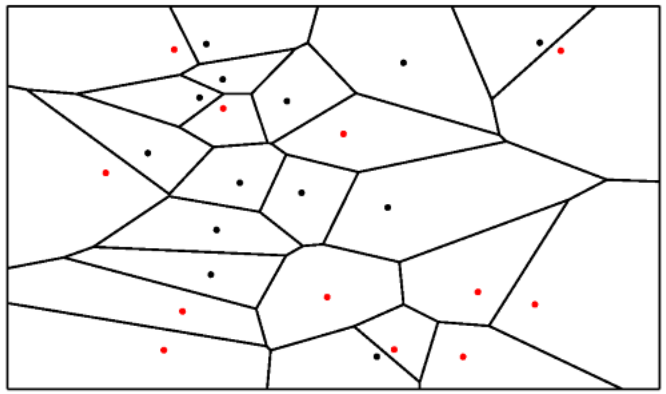
Non-Parametric versus Parametric Model

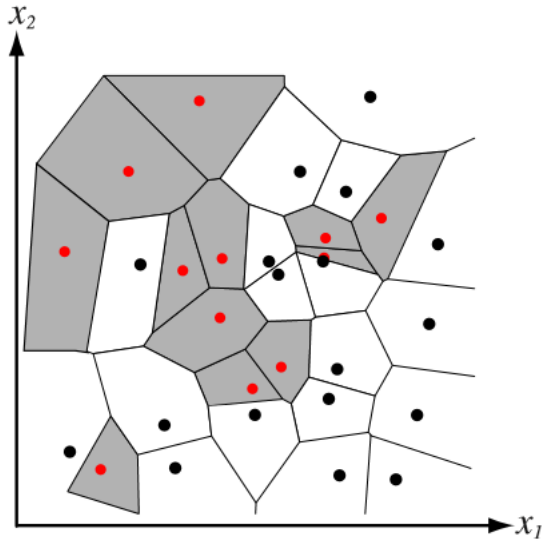
Is the number of parameters fixed?

- “Yes” implies the model is parametric
  - linear regression
  - logistic regression
- “No” implies the model is non-parametric
  - k-nearest neighbor
  - decision tree

Non-Parametric: Nearest Neighbor



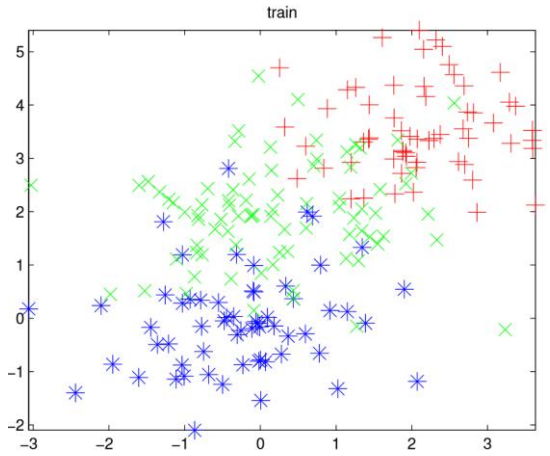


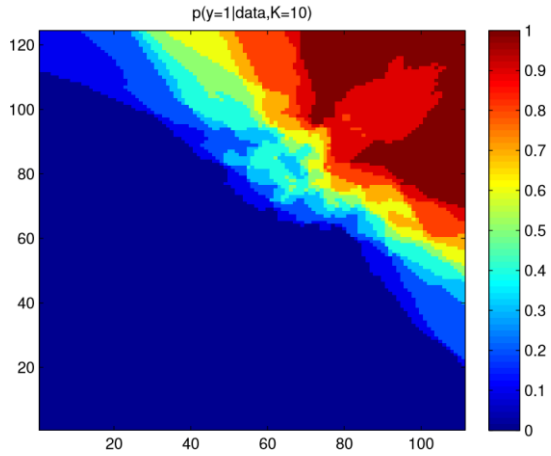


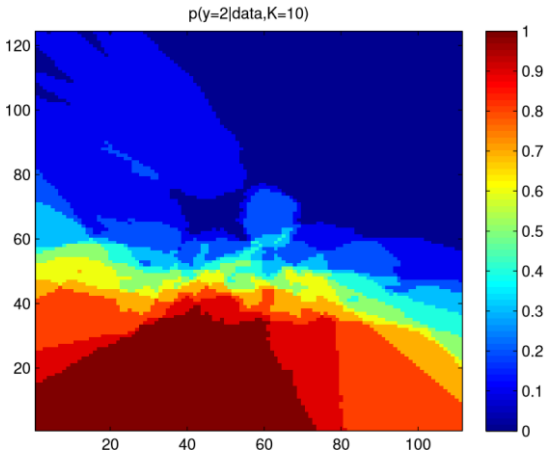
$$p(y = c|\mathbf{x}, \mathcal{D}, K) = \frac{1}{K} \sum_{i \in N_K(\mathbf{x}, \mathcal{D})} \mathbb{I}(y_i = c)$$

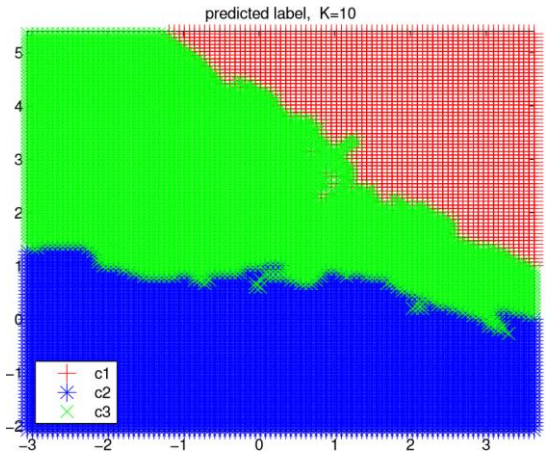
$$\mathbb{I}(e) = \begin{cases} 1 & \text{if } e \text{ is true} \\ 0 & \text{if } e \text{ is false} \end{cases}$$

k Nearest Neighbor: k = 10

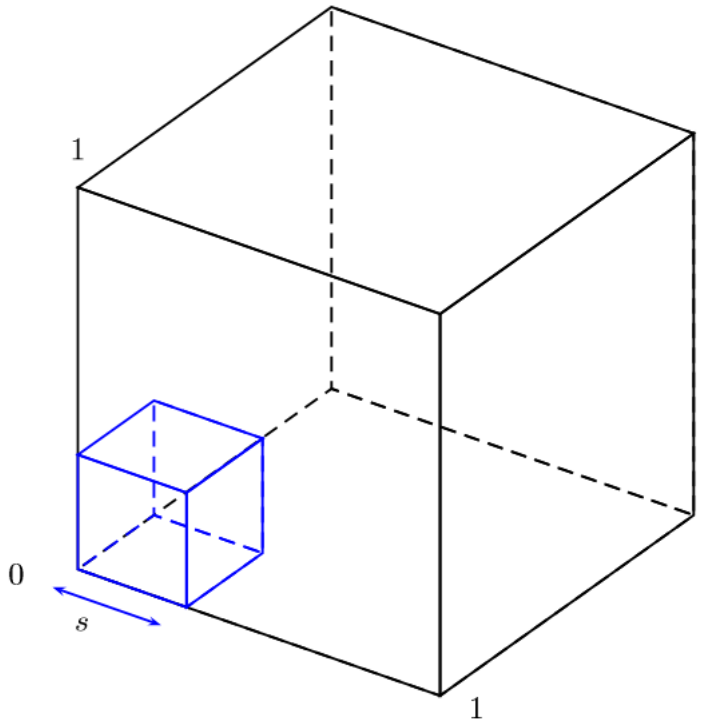


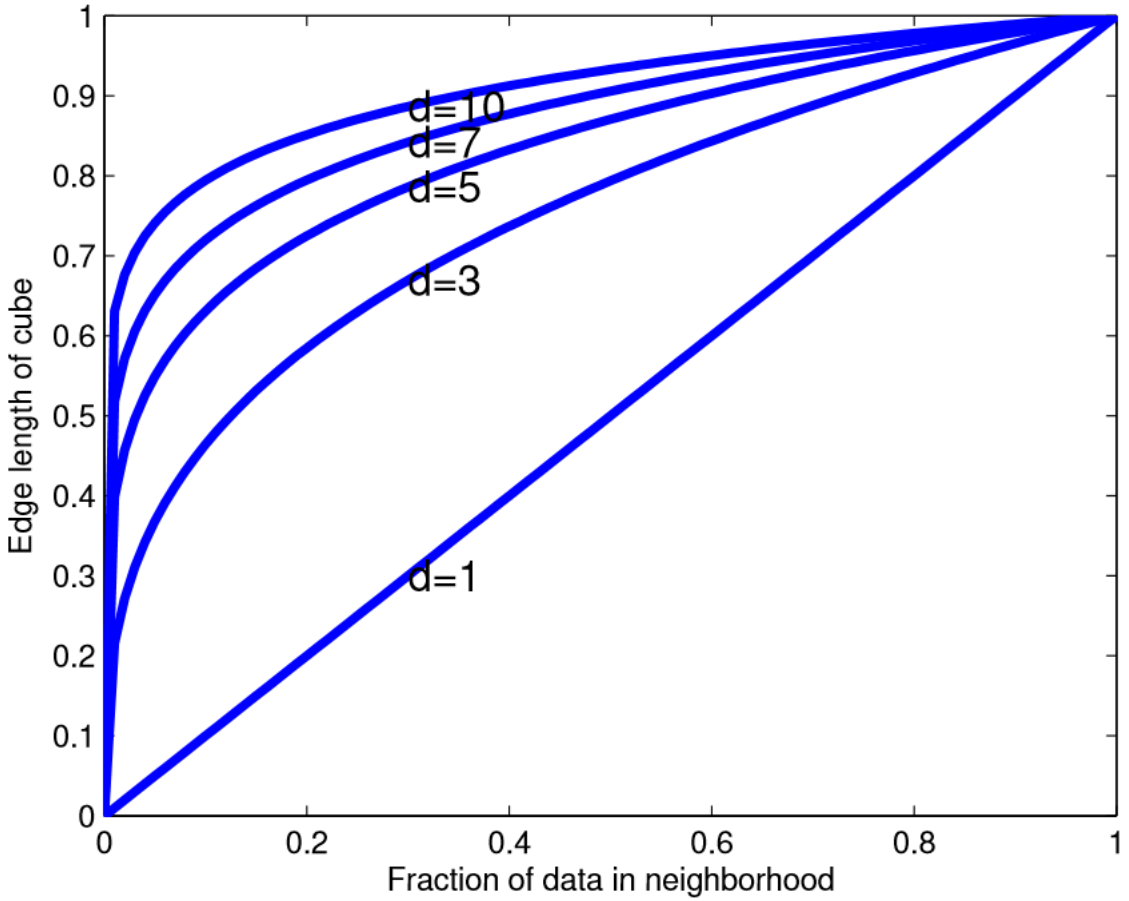






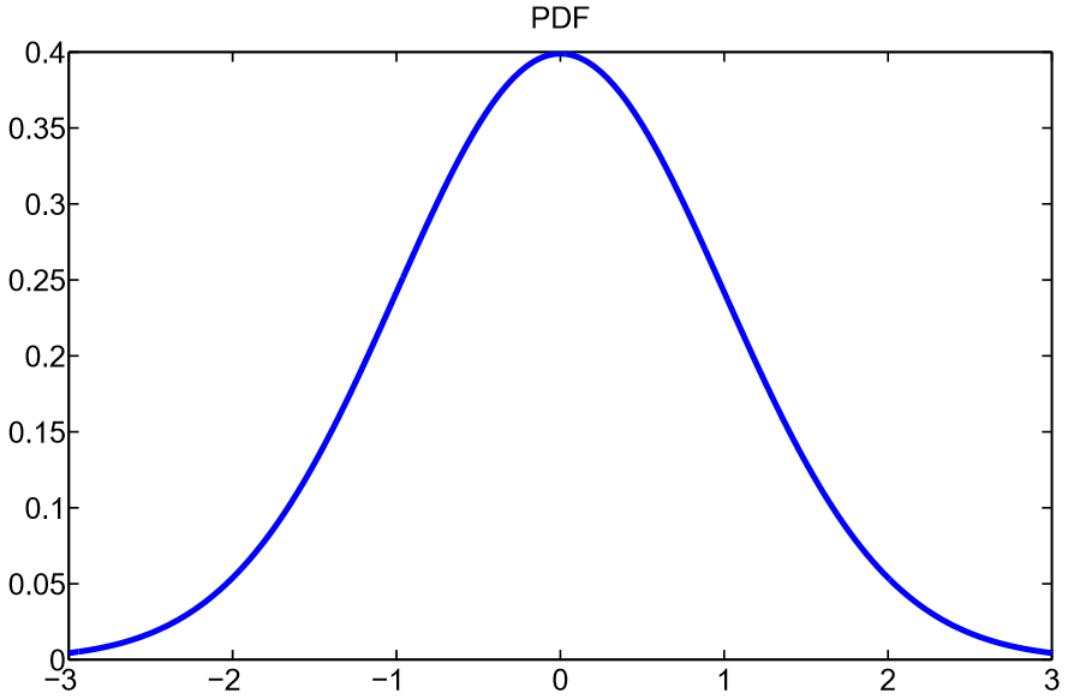
Curse of Dimensionality

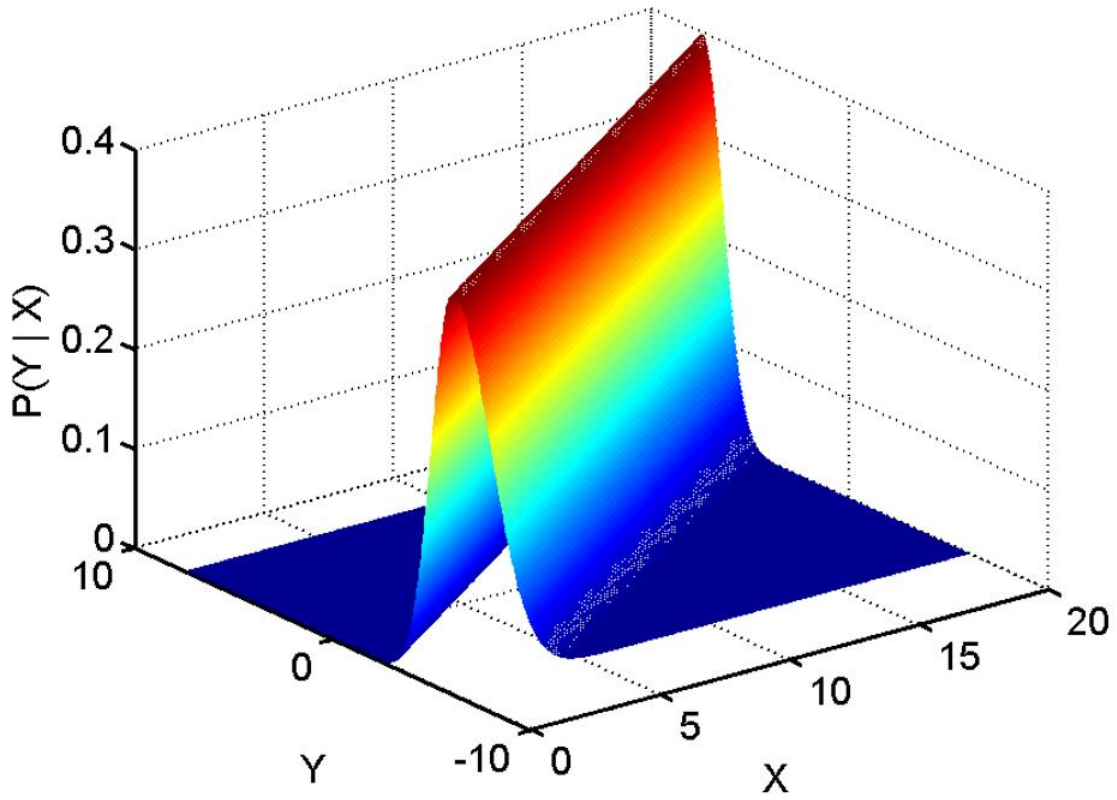




$$e_D(f) = f^{1/D}$$

Parametric: Linear Regression

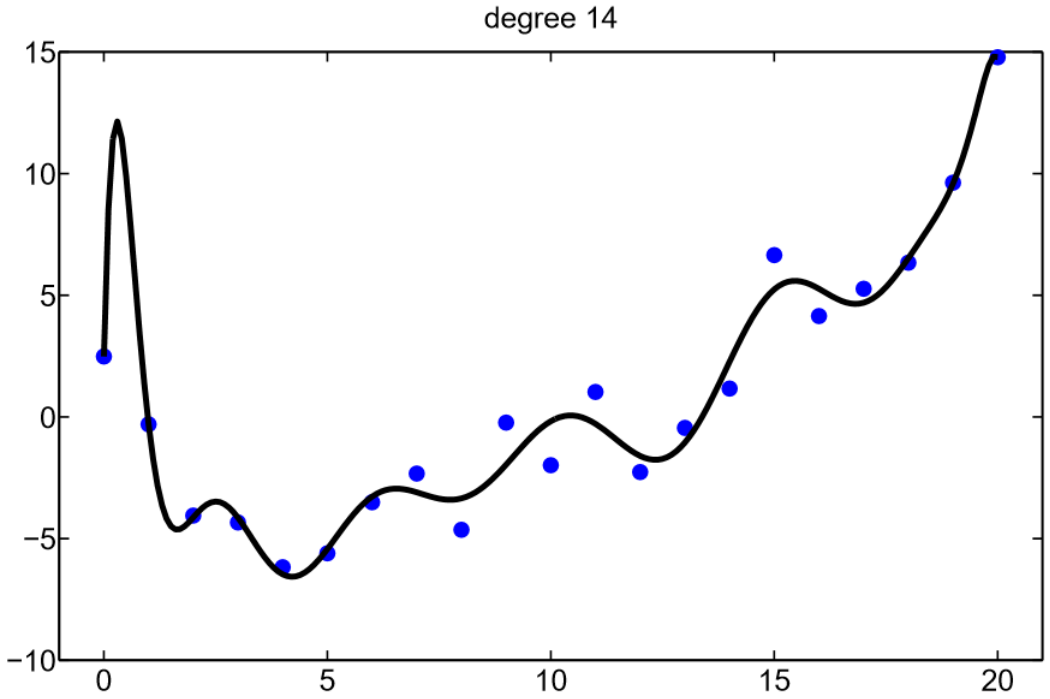


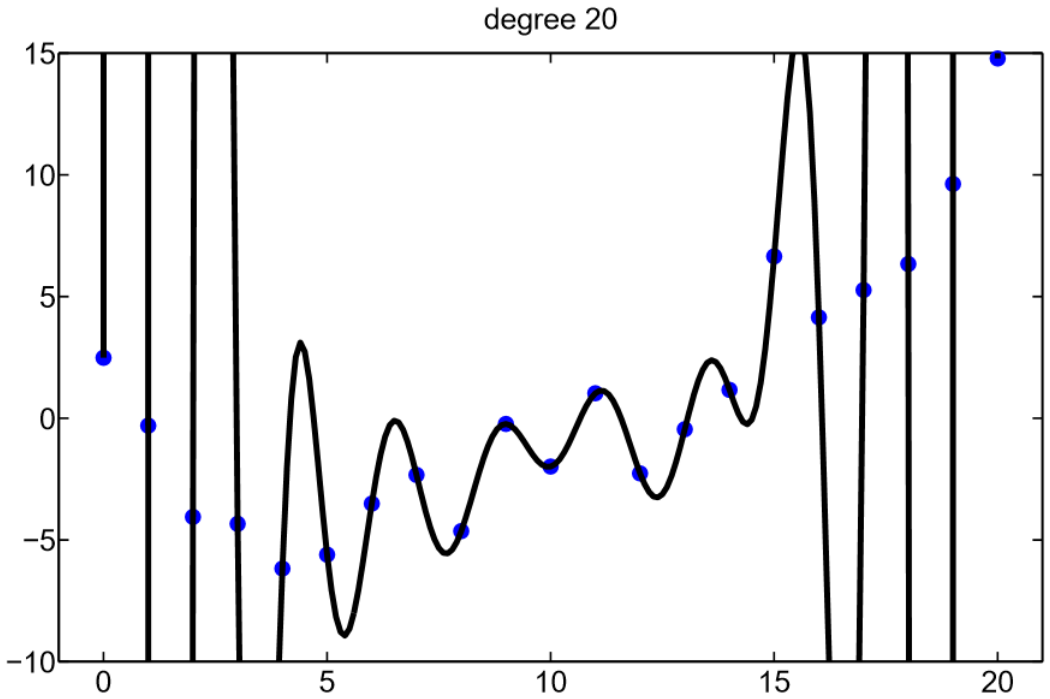


$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + \epsilon = \sum_{j=1}^{D} w_j x_j + \epsilon$$

$$p(y|\mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(y|\mu(\mathbf{x}), \sigma^2(\mathbf{x}))$$

Parametric: Polynomial Regression

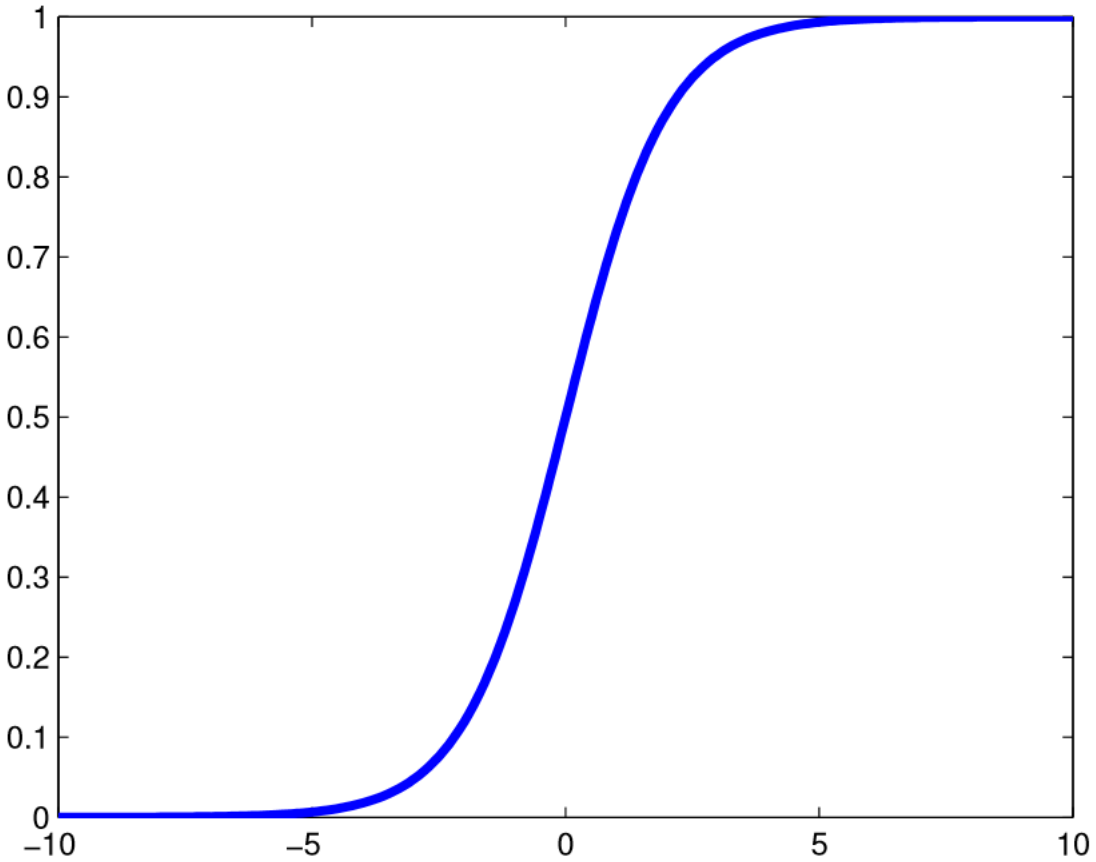


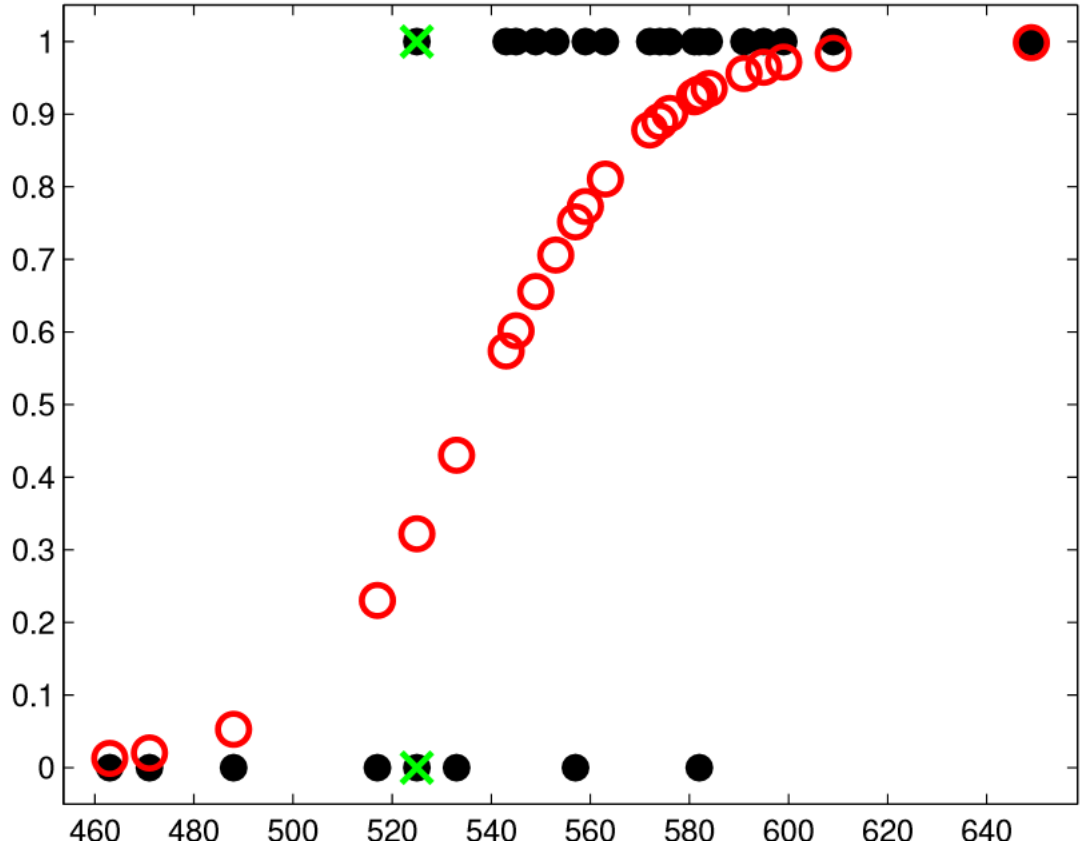


$$p(y|\mathbf{x}, \boldsymbol{\theta}) = \mathcal{N}(y|\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}), \sigma^2)$$

$$\boldsymbol{\phi}(\mathbf{x}) = [1, x, x^2, \ldots, x^d]$$

Parametric: Logistic Regression



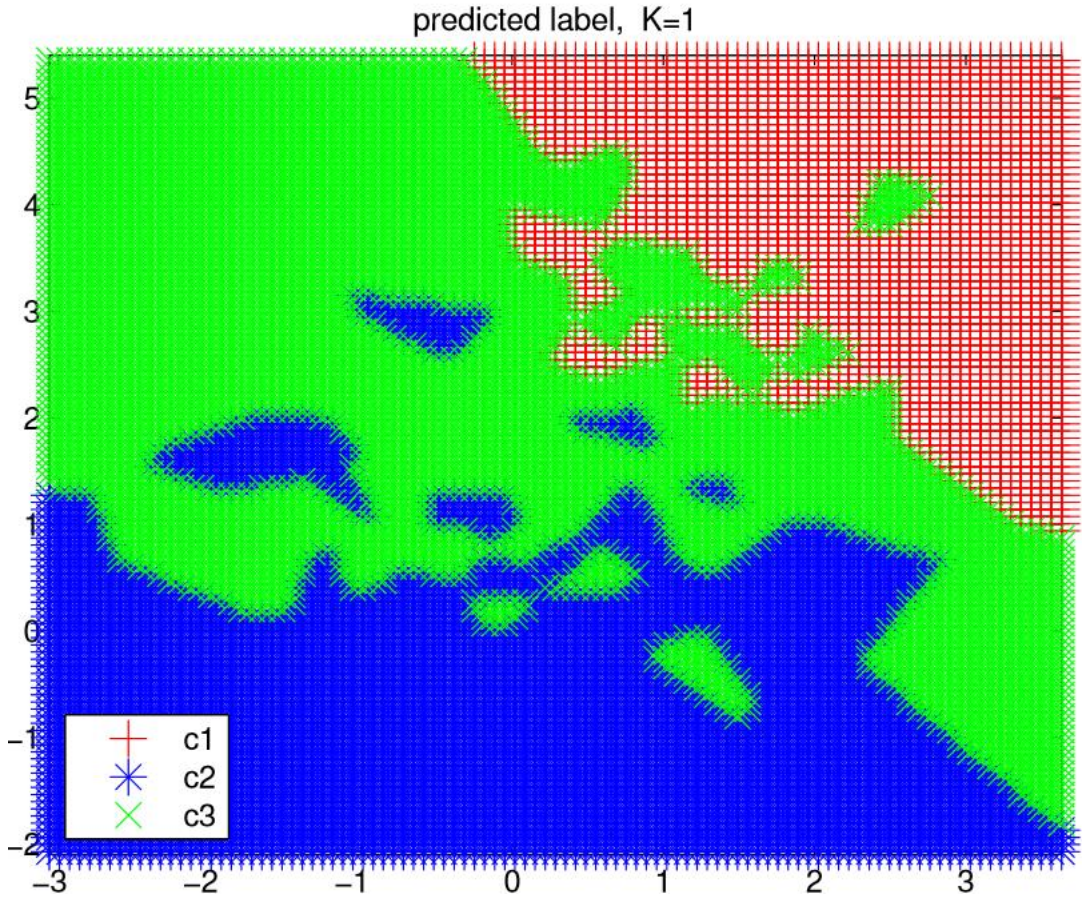


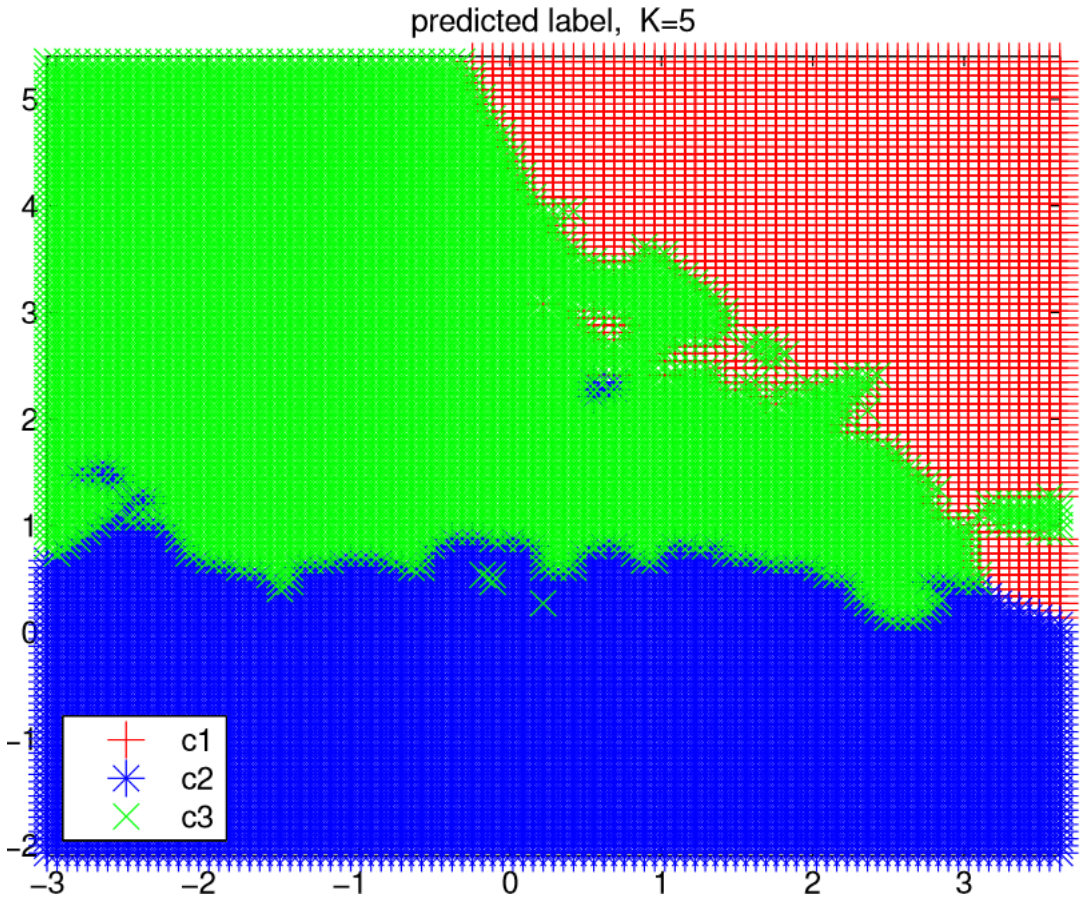
$$p(y|\mathbf{x},\mathbf{w}) = \mathrm{Ber}(y|\mu(\mathbf{x}))$$

$$\mu(\mathbf{x}) = \mathrm{sigm}(\mathbf{w}^T\mathbf{x})$$

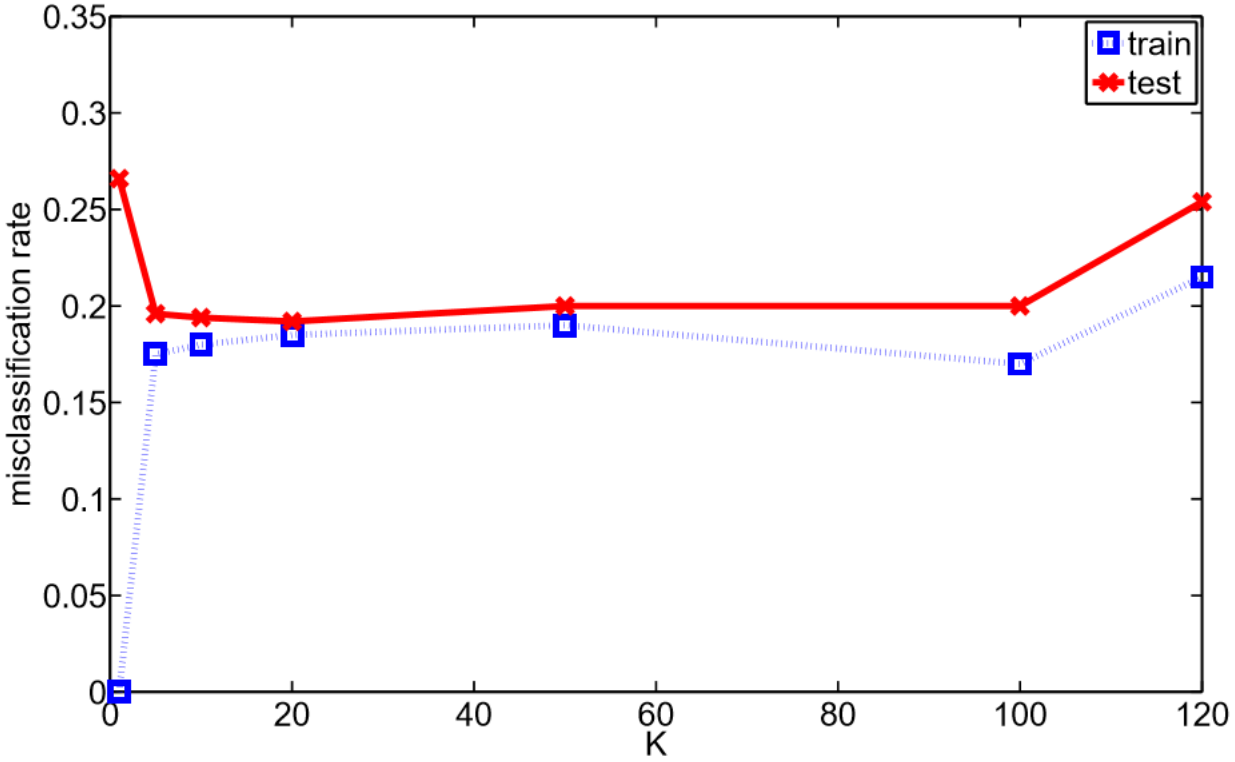
$$\mathrm{sigm}(\eta) \triangleq \frac{1}{1+\exp(-\eta)} = \frac{e^{\eta}}{e^{\eta}+1}$$

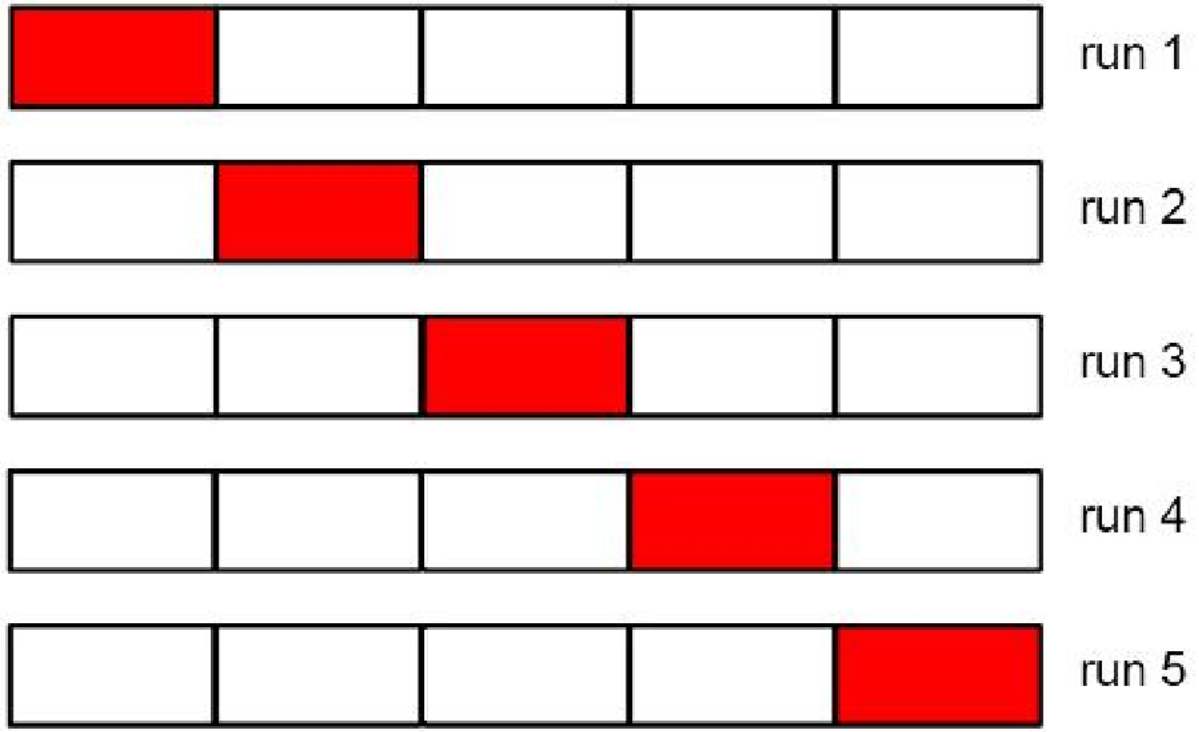
Overfitting





Model Selection





$$\mathrm{err}(f, \mathcal{D}) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(f(\mathbf{x}_i) \neq y_i)$$

No Free Lunch Theorem

- “All models are wrong, but some are useful” – George Box
- Much of machine learning is concerned with devising different models, and different algorithms to fit them
- There is no single best model that works optimally for all kinds of problems

ML v Statistics [Tibshirani]

Glossary

Machine learning	Statistics
network, graphs	model
weights	parameters
learning	fitting
generalization	test set performance
supervised learning	regression/classification
unsupervised learning	density estimation, clustering
large grant = $1,000,000	large grant= $50,000
nice place to have a meeting: Snowbird, Utah, French Alps	nice place to have a meeting: Las Vegas in August

https://twitter.com/ML_Hipster

ML Hipster @ML_Hipster · 31 Aug 2014

Using weighted majority to evaluate MOOCs that teach sequential prediction. It's online learning of online learning for online learning.

74 63

Ask lots of questions! Keep your sense of humor!