

Machine Learning Approaches to Biological Research: Bioimage Informatics and Beyond

Robert F. Murphy

External Senior Fellow, Freiburg Institute for Advanced Studies

Ray and Stephanie Lane Professor of Computational Biology, Carnegie
Mellon University

September 29-October 1, 2009

Outline

- Basic principles and paradigms of supervised and unsupervised machine learning
- Concepts of automated image analysis
- Approaches for creating predictive models from images
- Active learning paradigms for closed loop systems of cycles of experimentation, model refinement and model testing

The Discipline of Machine Learning

Tom M. Mitchell

July 2006
CMU-ML-06-108

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Over the past 50 years the study of Machine Learning has grown from the efforts of a handful of computer engineers exploring whether computers could learn to play games, and a field of Statistics that largely ignored computational considerations, to a broad discipline that has produced fundamental statistical-computational theories of learning processes, has designed learning algorithms that are routinely used in commercial systems for speech recognition, computer vision, and a variety of other tasks, and has spun off an industry in data mining to discover hidden regularities in the growing volumes of online data. This document provides a brief and personal view of the discipline that has emerged as Machine Learning, the fundamental questions it addresses, its relationship to other sciences and society, and where it might be headed.

What is Machine Learning?

- Fundamental Question of Computer Science: How can we build machines that solve problems, and which problems are inherently tractable/intractable?
- Fundamental Question of Statistics: What can be inferred from data plus a set of modeling assumptions, with what reliability?

Fundamental Question of Machine Learning

- How can we build computer systems that automatically improve with experience, and what are the fundamental laws that govern all learning processes?
 - Tom Mitchell

Why Machine Learning?

- Learn relationships from large sets of complex data: Data mining
 - Predict clinical outcome from tests
 - Decide whether someone is a good credit risk
- Do tasks too complex to program by hand
 - Autonomous driving
- Customize programs to user needs
 - Recommend book/movie based on previous likes

Why Machine Learning?

- Economically efficient
- Can consider larger data spaces and hypothesis spaces than people can
- Can formalize learning problem to explicitly identify/describe goals and criteria

Successful Machine Learning Applications

- Speech recognition
 - Telephone menu navigation
- Computer vision
 - Mail sorting
- Bio-surveillance
 - Identifying disease outbreaks
- Robot control
 - Autonomous driving
- Empirical science

Machine Learning Paradigms

- Supervised Learning
 - Classification
 - Regression
- Unsupervised Learning
 - Clustering
- Semi-supervised Learning
 - Cotraining
 - Active learning

Supervised Learning

- Approaches
 - Classification (discrete predictions)
 - Regression (continuous predictions)
- Common considerations
 - Representation (Features)
 - Feature Selection
 - Functional form
 - Evaluation of predictive power

Classification vs. Regression

- If I want to predict whether a patient will die from a disease within six months, that is classification
- If I want to predict how long the patient will live, that is regression

Representation

- Definition of thing or things to be predicted
 - Classification: *classes*
 - Regression: *regression variable*
- Definition of things (*instances*) to make predictions for
 - Individuals
 - Families
 - Neighborhoods, etc.
- Choice of descriptors (*features*) to describe different aspects of instances

Formal description

- Defining X as a set of *instances* x described by *features*
- Given training examples D from X
- Given a *target function* c that maps $X \rightarrow \{0,1\}$
- Given a *hypothesis space* H
- Determine an hypothesis h in H such that $h(x)=c(x)$ for all x in D

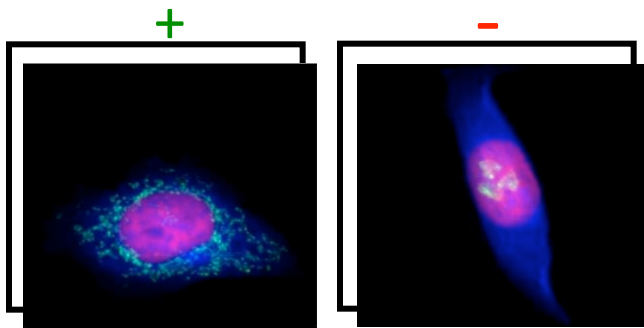
Inductive learning hypothesis

- Any hypothesis found to approximate the target function well over a sufficiently large set of training example will also approximate the target function over other unobserved example

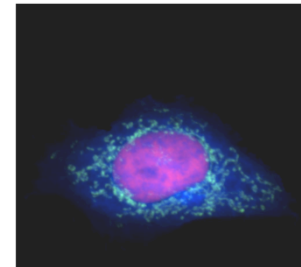
Hypothesis space

- The hypothesis space determines the functional form
- It defines what are allowable rules/functions for classification
- Each classification method uses a different hypothesis space

Simple two class problem



???

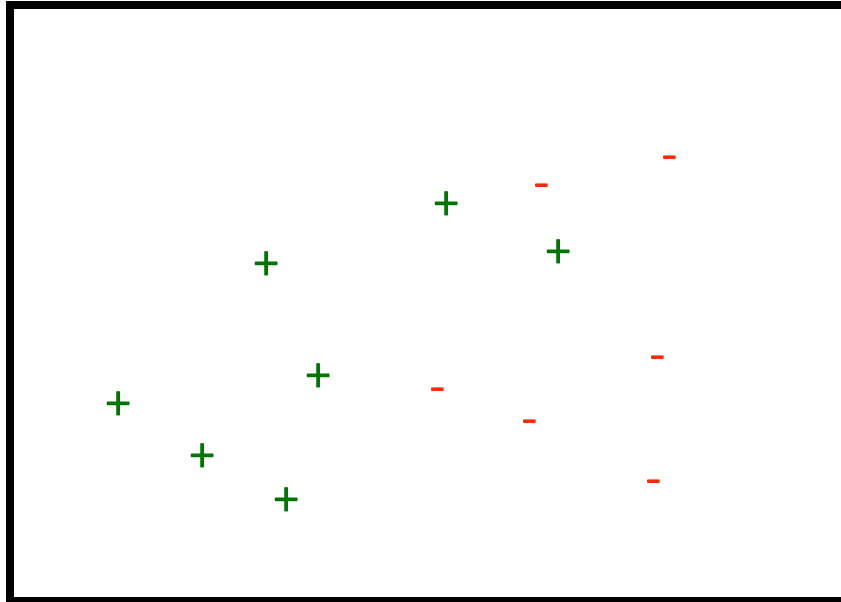


Describe each image by features
Train classifier

k-Nearest Neighbor (kNN)

- In feature space, training examples are

Feature #2
(e.g., roundness)

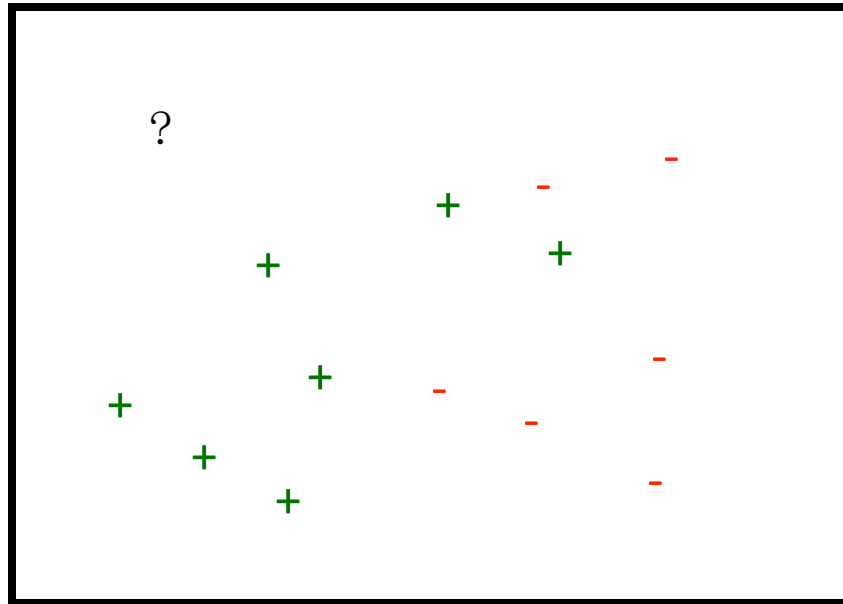


Feature #1 (e.g., 'area')

k-Nearest Neighbor (kNN)

- We want to label ‘?’

Feature #2
(e.g., roundness)

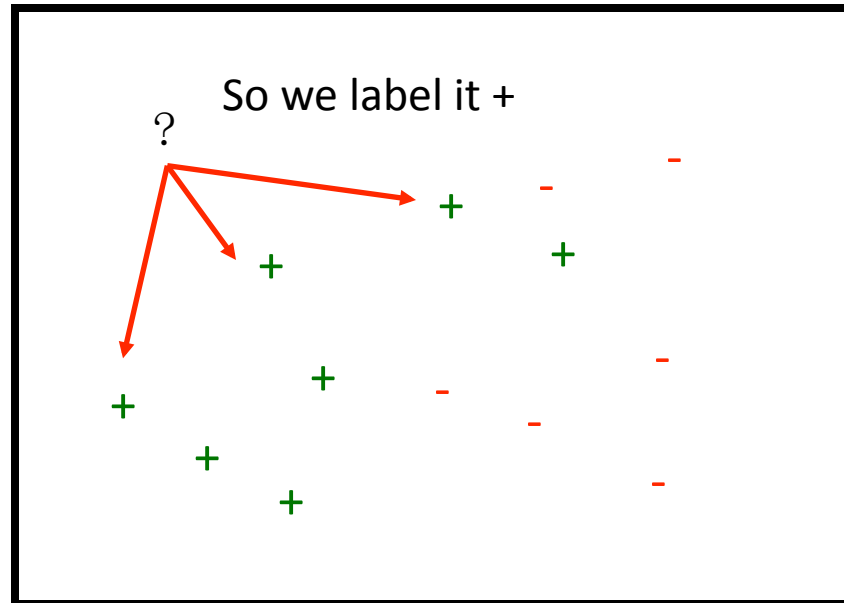


Feature #1 (e.g., 'area')

k-Nearest Neighbor (kNN)

- Find k nearest neighbors and vote

Feature #2
(e.g., roundness)



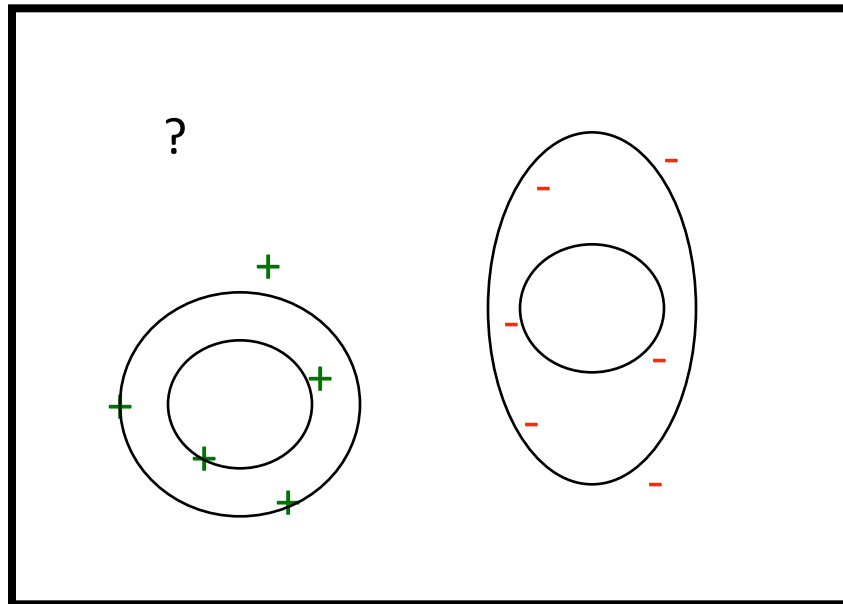
for $k=3$,
nearest
neighbors
are

Feature #1 (e.g., 'area')

Linear Discriminants

- Fit multivariate Gaussian to each class
- Measure distance from ? to each Gaussian

bright.

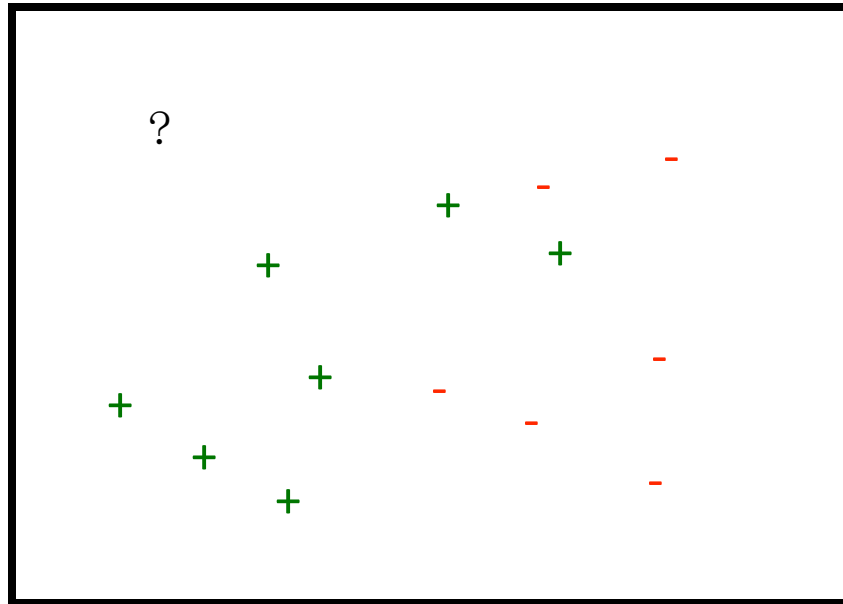


area

Decision trees

- Again we want to label ‘?’

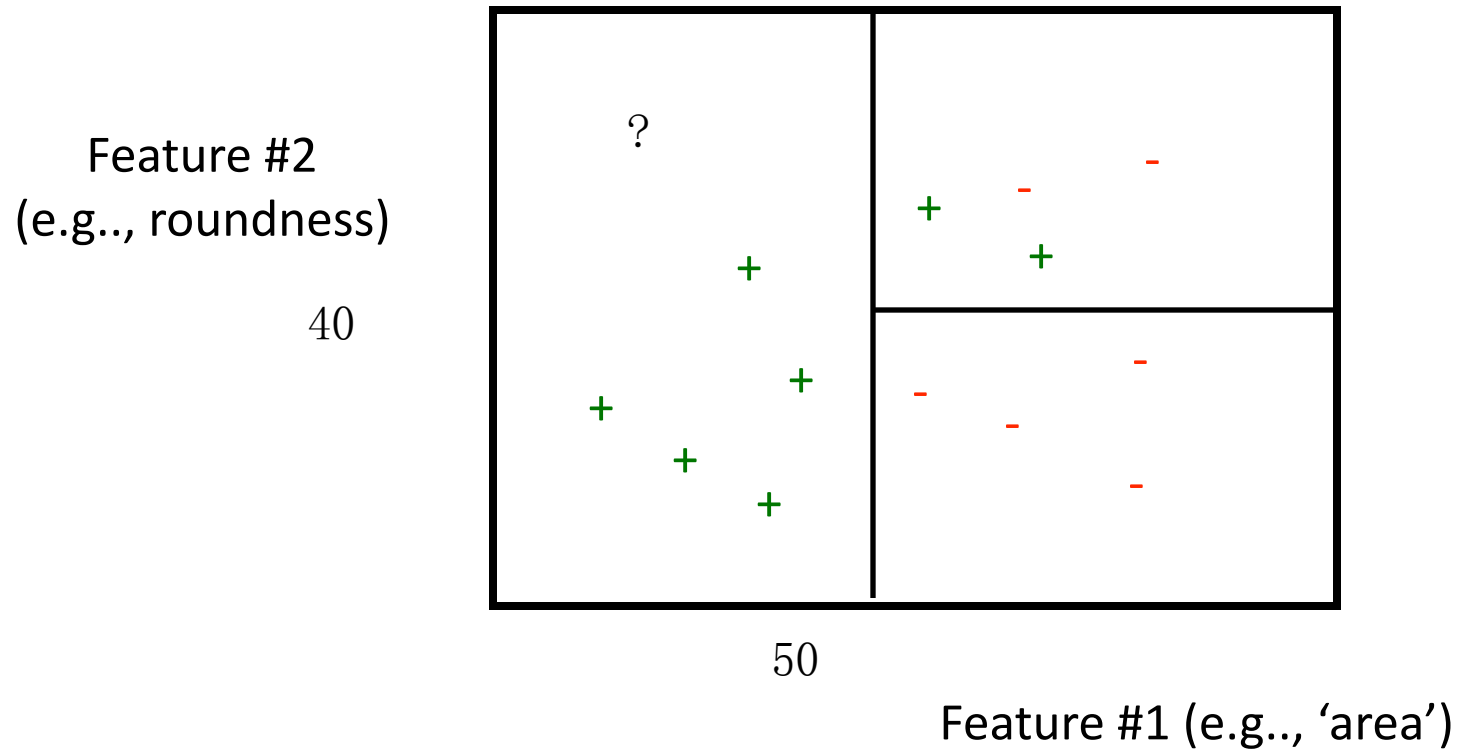
Feature #2
(e.g., roundness)



Feature #1 (e.g., 'area')

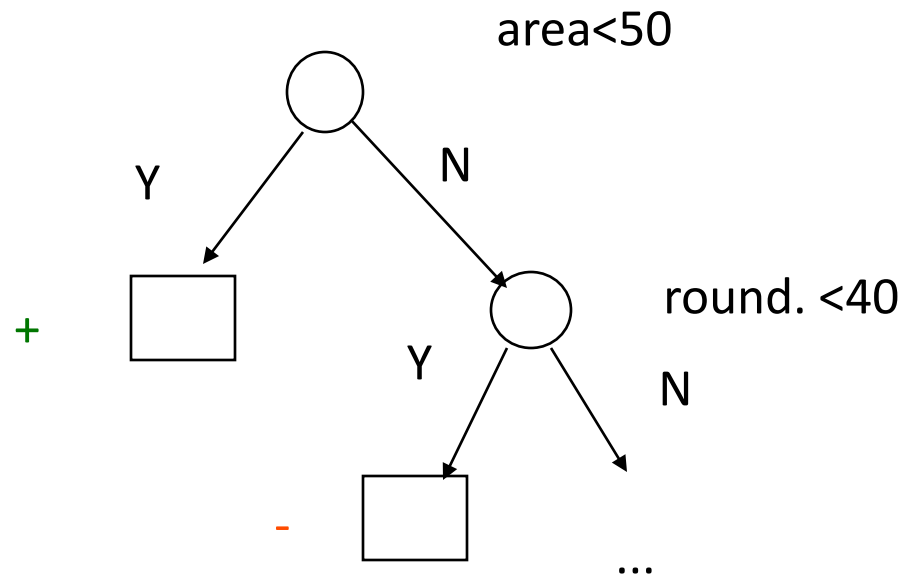
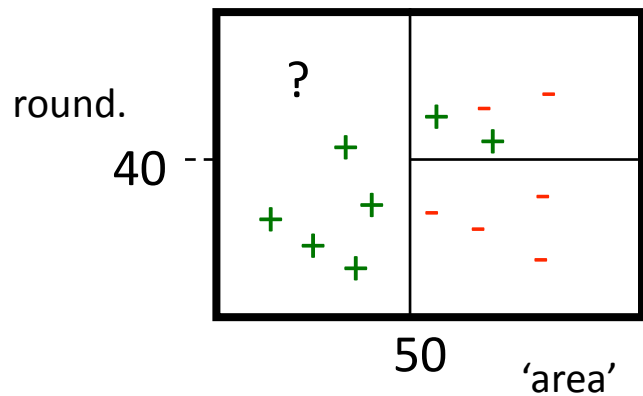
Decision trees

- so we build a decision tree:



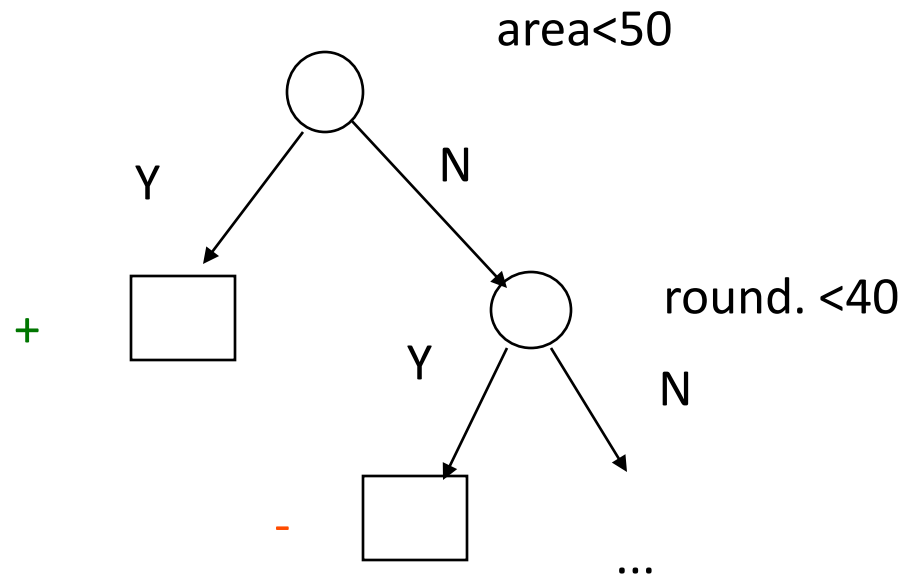
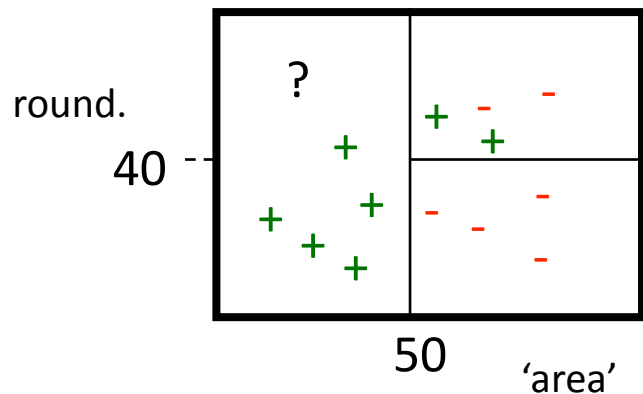
Decision trees

- so we build a decision tree:



Decision trees

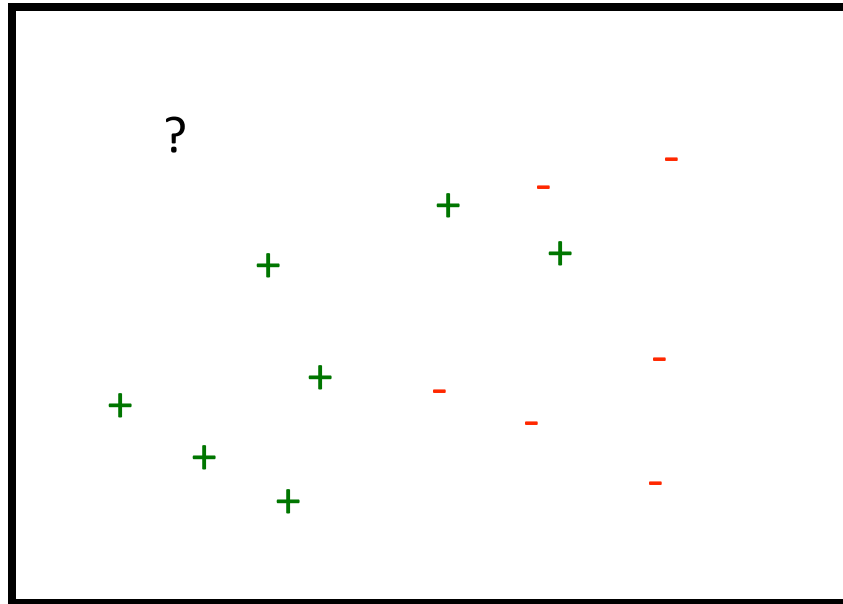
- Goal: split address space in (almost) homogeneous regions



Support vector machines

- Again we want to label ‘?’

Feature #2
(e.g., roundness)

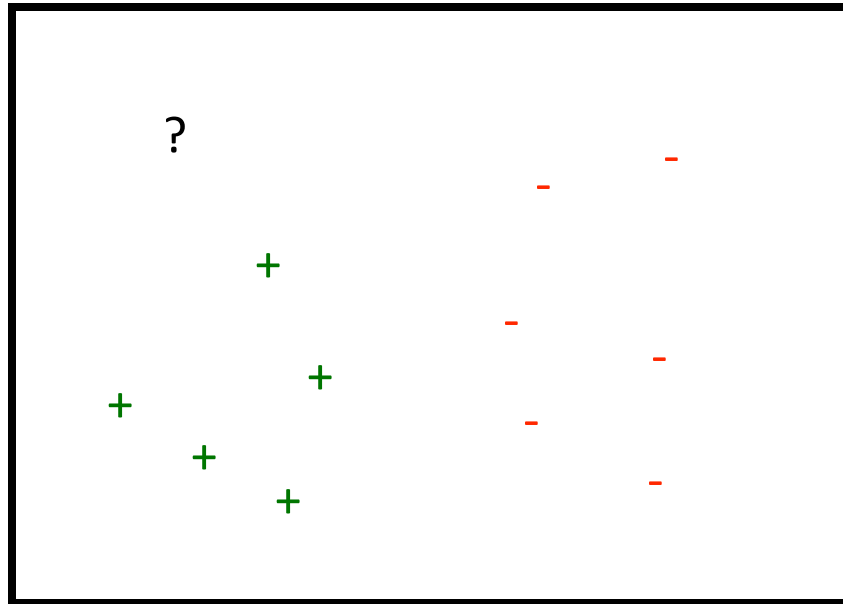


Feature #1 (e.g., 'area')

Support Vector Machines (SVMs)

- Use single linear separator??

round.

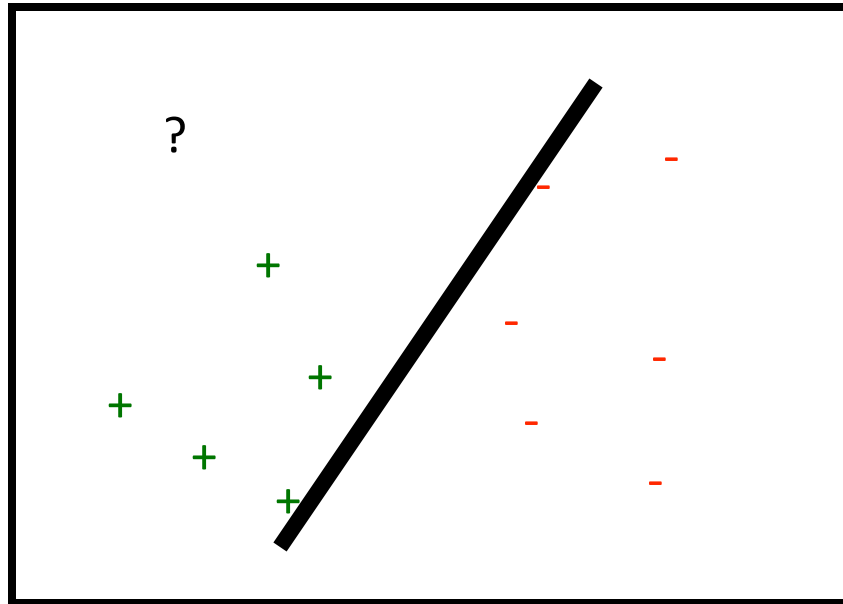


area

Support Vector Machines (SVMs)

- Use single linear separator??

round.

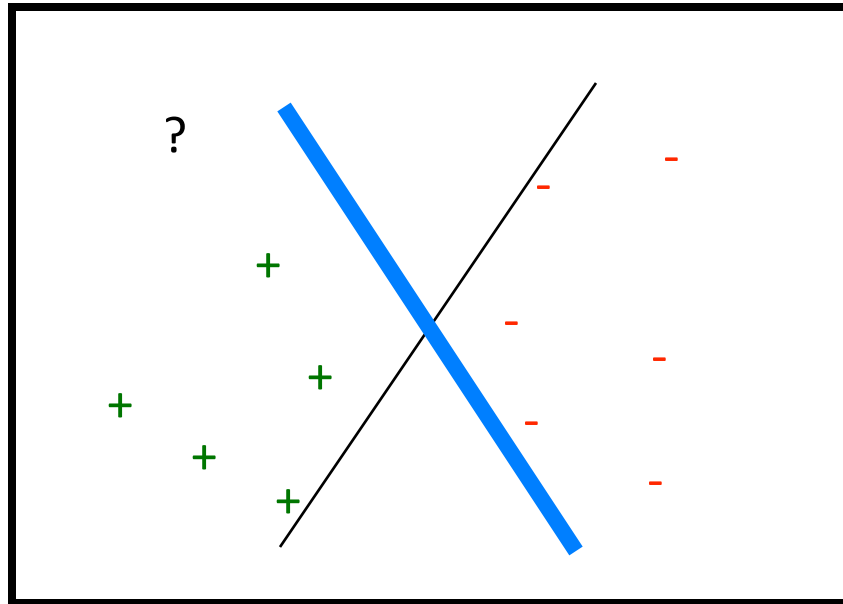


area

Support Vector Machines (SVMs)

- Use single linear separator??

round.

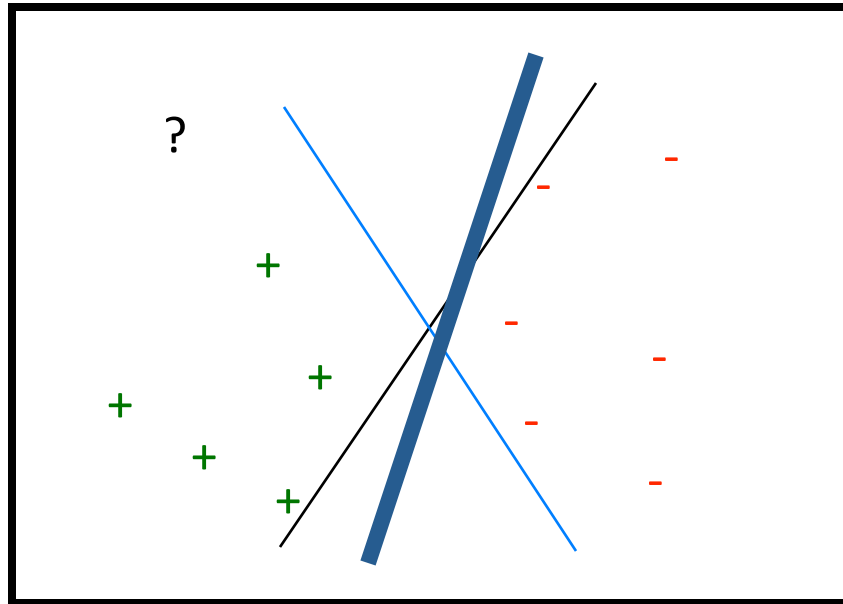


area

Support Vector Machines (SVMs)

- Use single linear separator??

round.

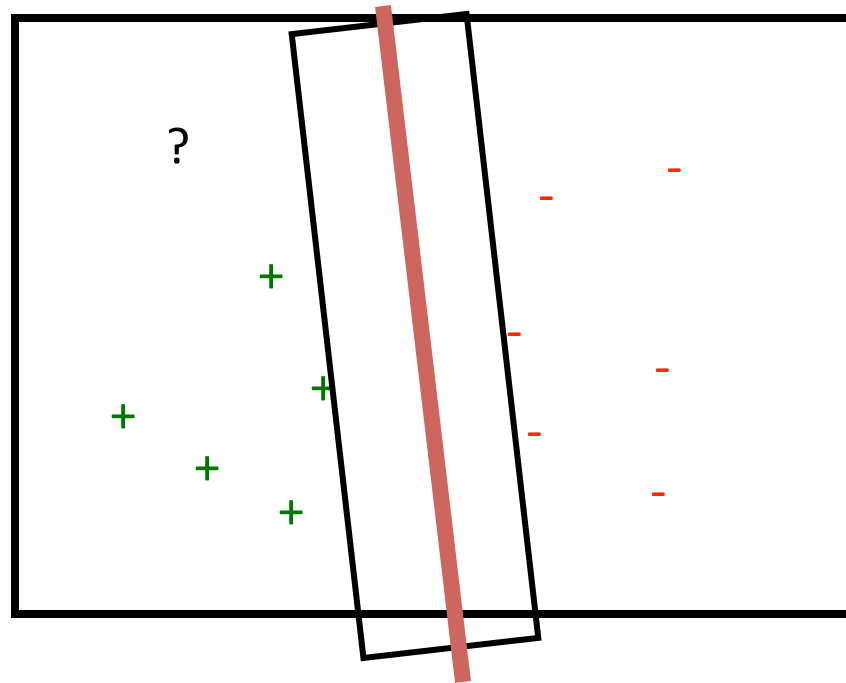


area

Support Vector Machines (SVMs)

- Use single linear separator??

round.

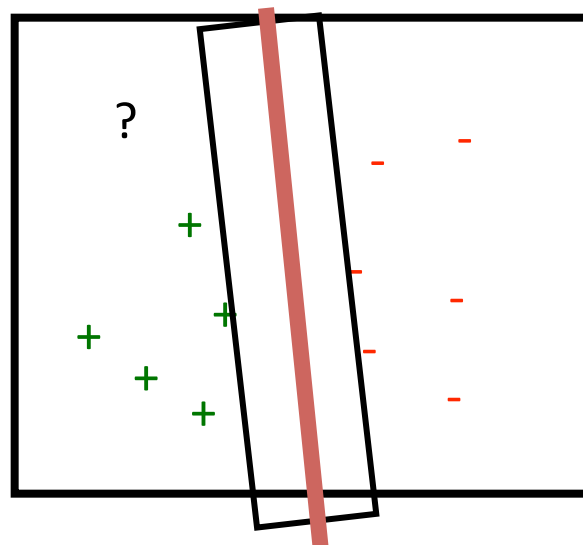


area

Support Vector Machines (SVMs)

- we want to label ‘?’ - linear separator??
- A: the one with the widest corridor!

round.



area

Support Vector Machines (SVMs)

- What if the points for each class are not readily separated by a straight line?
- Use the “kernel trick” – project the points into a higher dimensional space in which we hope that straight lines will separate the classes
- “kernel” refers to the function used for this projection

Support Vector Machines (SVMs)

- Definition of SVMs explicitly considers only two classes
- What if we have more than two classes?
- Train multiple SVMs
- Two basic approaches
 - One against all (one SVM for each class)
 - Pairwise SVMs (one for each pair of classes)
 - Various ways of implementing this

Questions

- What are the hypothesis spaces for
 - kNN classifier
 - Linear discriminants
 - Decision trees
 - Support Vector Machines

Cross-Validation

- If we train a classifier to minimize error on a set of data, have no ability to estimate (generalize) error that will be seen on new dataset
- To calculate *generalizable* accuracy, we use ***n*-fold cross-validation**
- Divide images into n sets, train using $n-1$ of them and test on the remaining set
- Repeat until each set is used as test set and average results across all trials
- Variation on this is called ***leave-one-out***

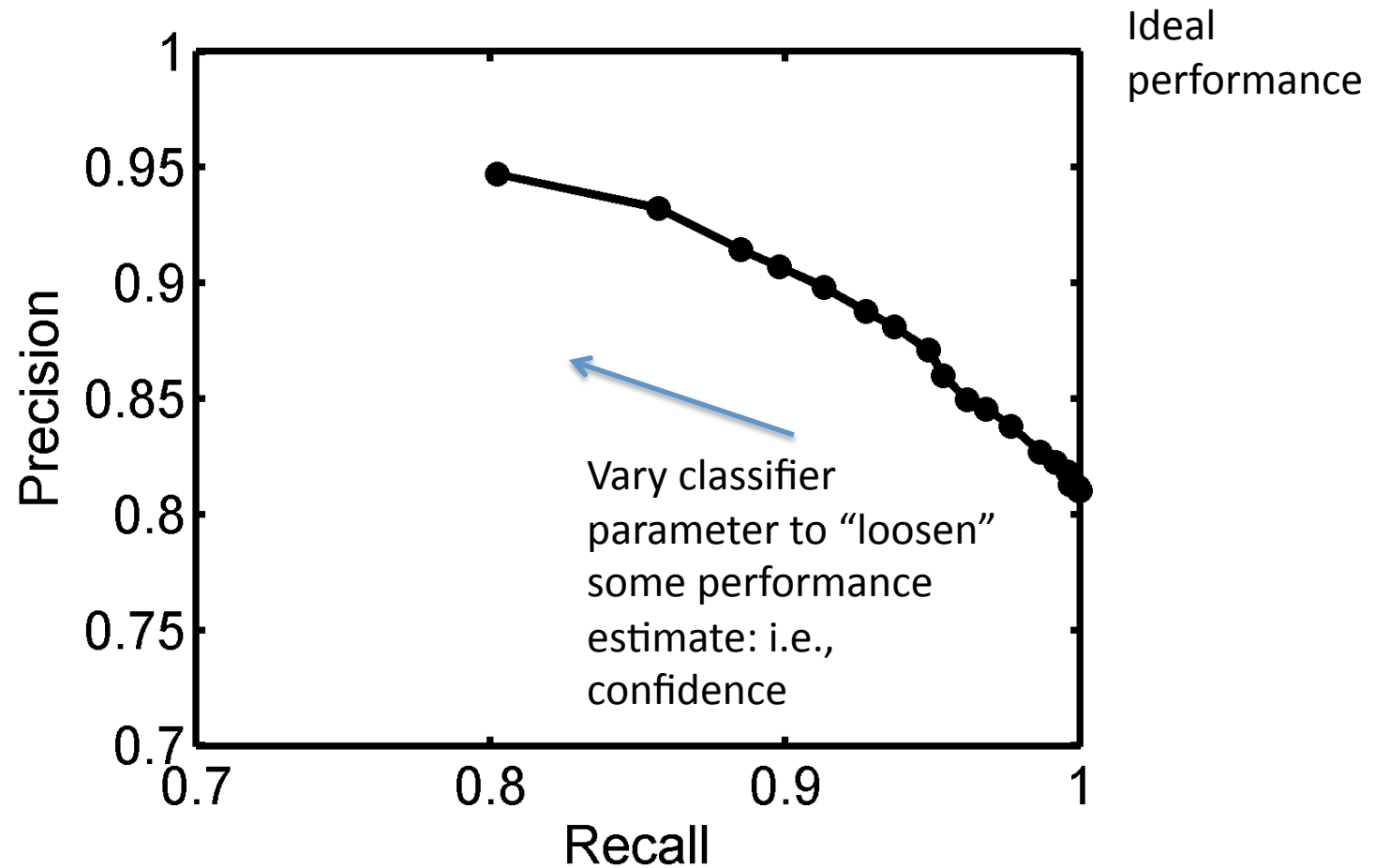
Describing classifier errors

- For binary classifiers (positive or negative), define
 - TP = true positives, FP = false positives
 - TN = true negatives, FN = false negatives
 - Recall = $TP / (TP + FN)$
 - Precision = $TP / (TP + FP)$
 - F-measure = $2 * \text{Recall} * \text{Precision} / (\text{Recall} + \text{Precision})$

Confusion matrix - binary

True \ Predicted	Positive	Negative
Positive	True Positive	False Negative
Negative	False Positive	True Negative

Precision-recall analysis



Describing classifier errors

- For multi-class classifiers, typically report
 - Accuracy = $\frac{\text{\# test images correctly classified}}{\text{\# test images}}$
 - Confusion matrix = table showing all possible combinations of true class and predicted class

Confusion matrix – multi-class

True Class	Output of the Classifier									
	DNA	ER	Gia	Gpp	Lam	Mit	Nuc	Act	TfR	Tub
DNA	98	2	0	0	0	0	0	0	0	0
ER	0	100	0	0	0	0	0	0	0	0
Gia	0	0	100	0	0	0	0	0	0	0
Gpp	0	0	0	96	4	0	0	0	0	0
Lam	0	0	0	4	95	0	0	0	0	2
Mit	0	0	2	0	0	96	0	2	0	0
Nuc	0	0	0	0	0	0	100	0	0	0
Act	0	0	0	0	0	0	0	100	0	0
TfR	0	0	0	0	2	0	0	0	96	2
Tub	0	2	0	0	0	0	0	0	0	98

Overall accuracy = 98%

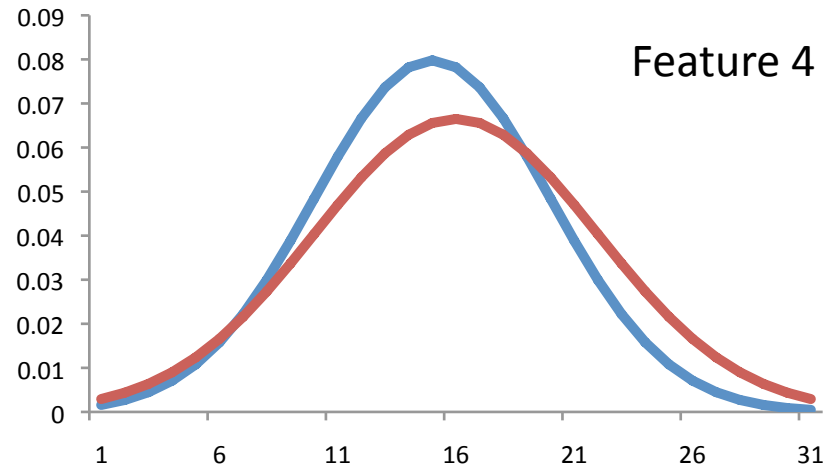
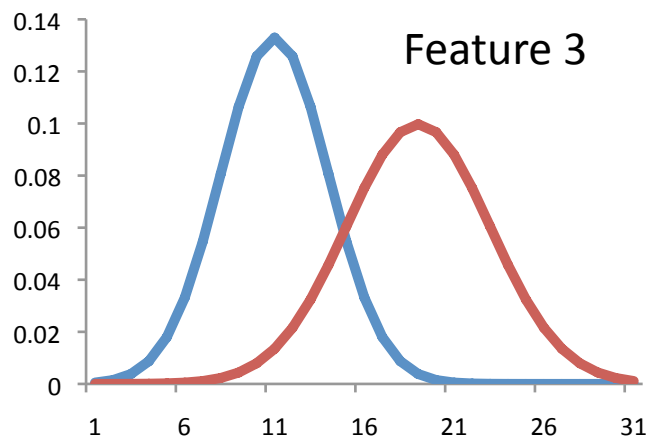
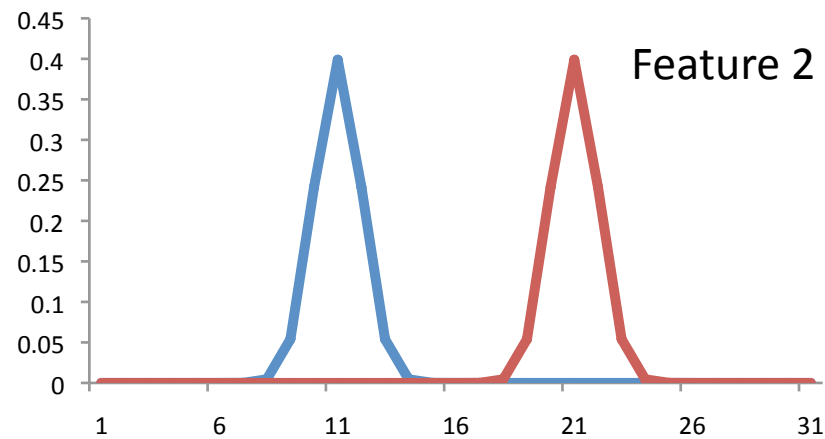
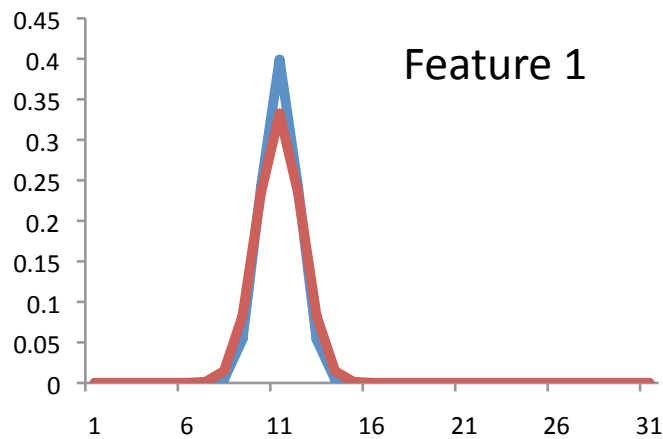
Ground truth

- What is the source and confidence of a class label?
- Most common: Human assignment, unknown confidence
- Preferred: Assignment by experimental design, confidence $\sim 100\%$

Feature selection

- Having too many features can confuse a classifier
- Can use comparison of feature distributions between classes to choose a subset of features that gets rid of uninformative or redundant features

Basic principle of feature selection



red=class 1, blue=class 2

An Introduction to Variable and Feature Selection

Isabelle Guyon

Clopinet

955 Creston Road

Berkeley, CA 94708-1501, USA

ISABELLE@CLOPINET.COM

André Elisseeff

Empirical Inference for Machine Learning and Perception Department

Max Planck Institute for Biological Cybernetics

Spemannstrasse 38

72076 Tübingen, Germany

ANDRE@TUEBINGEN.MPG.DE

Need to consider multivariate distance

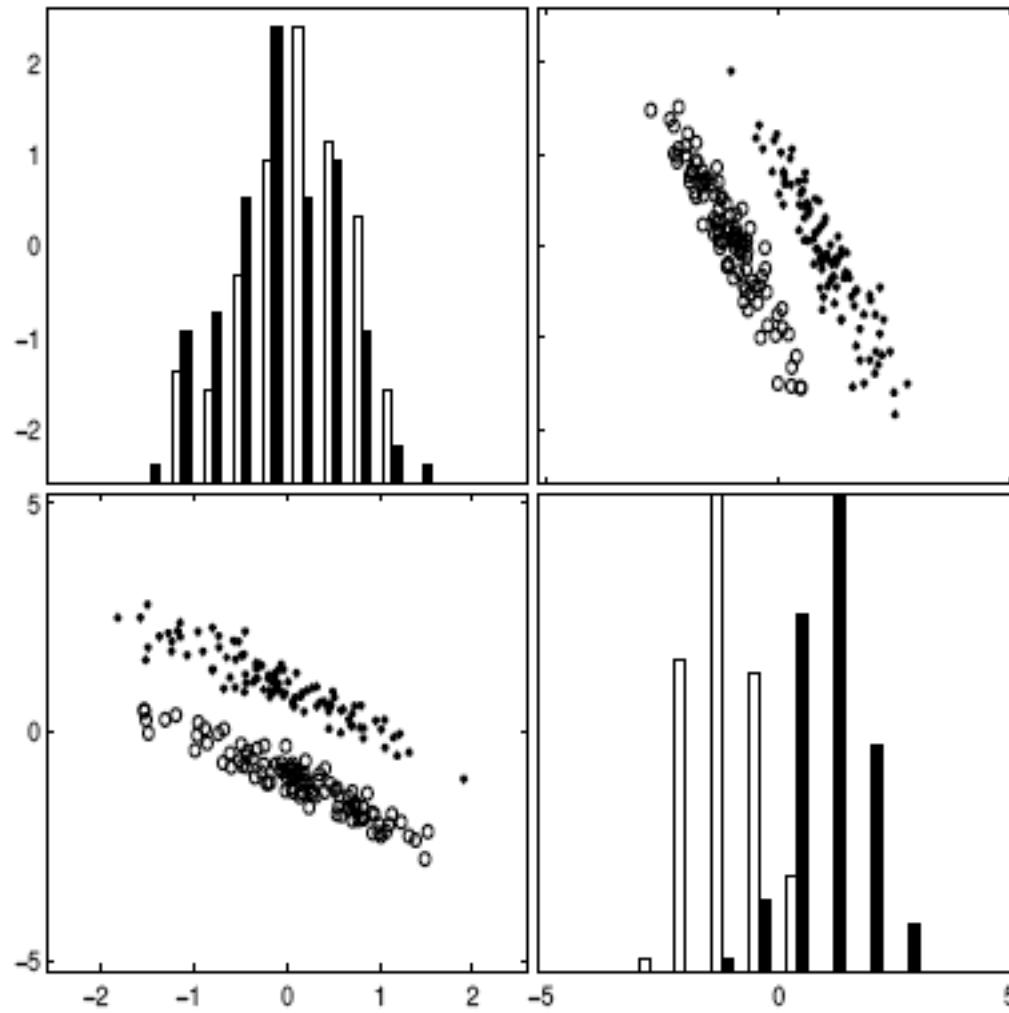


Figure from Guyon & Elisseeff

Bad and Good Covariance

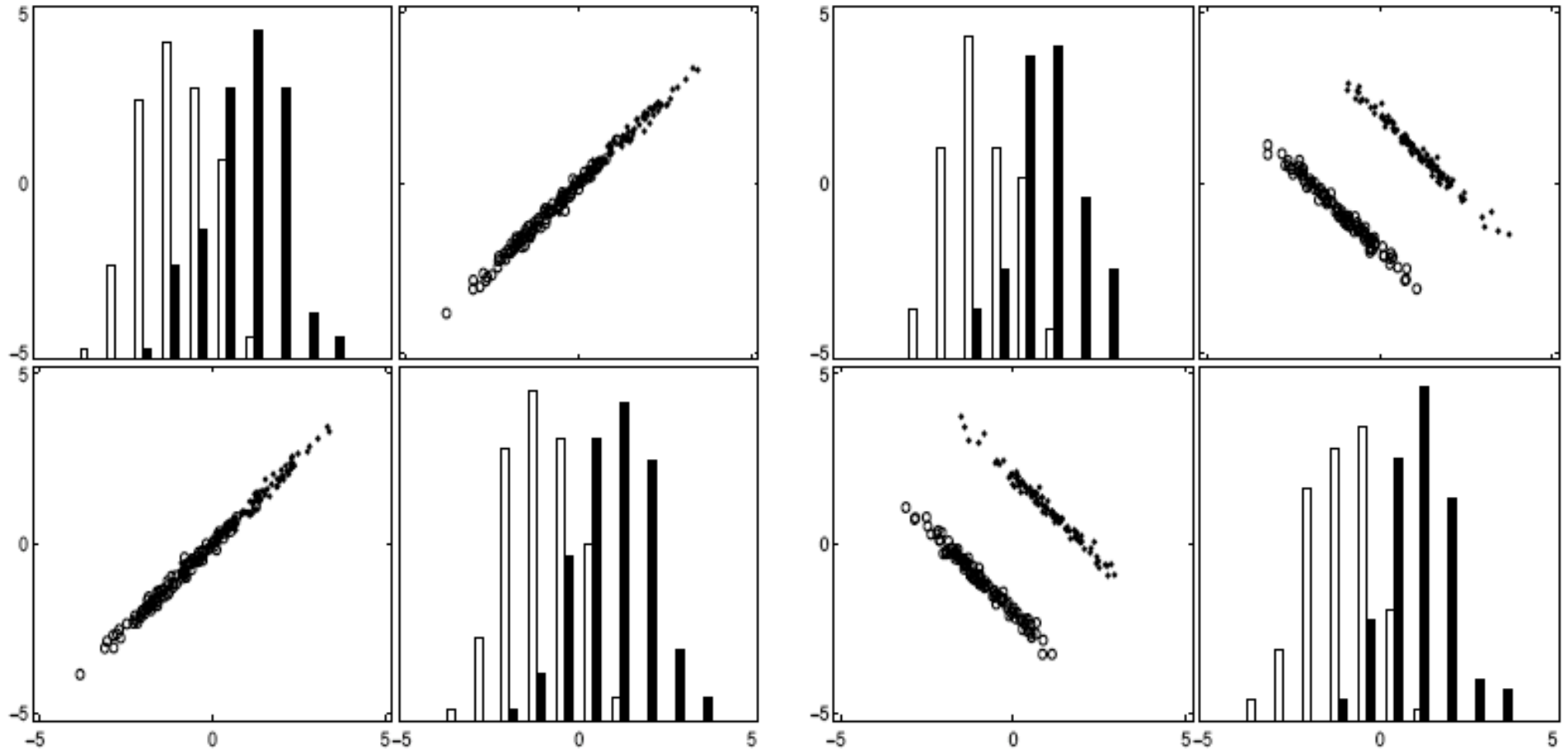


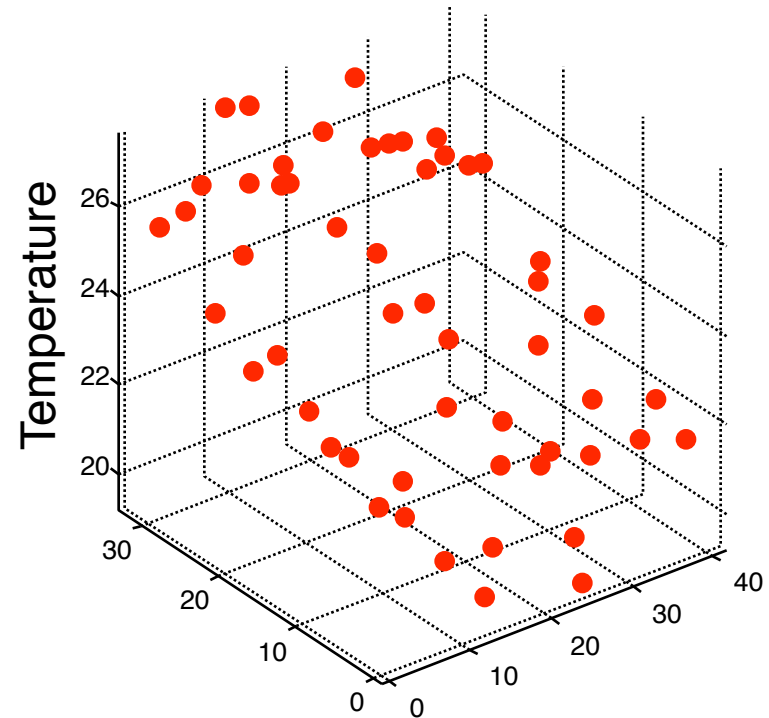
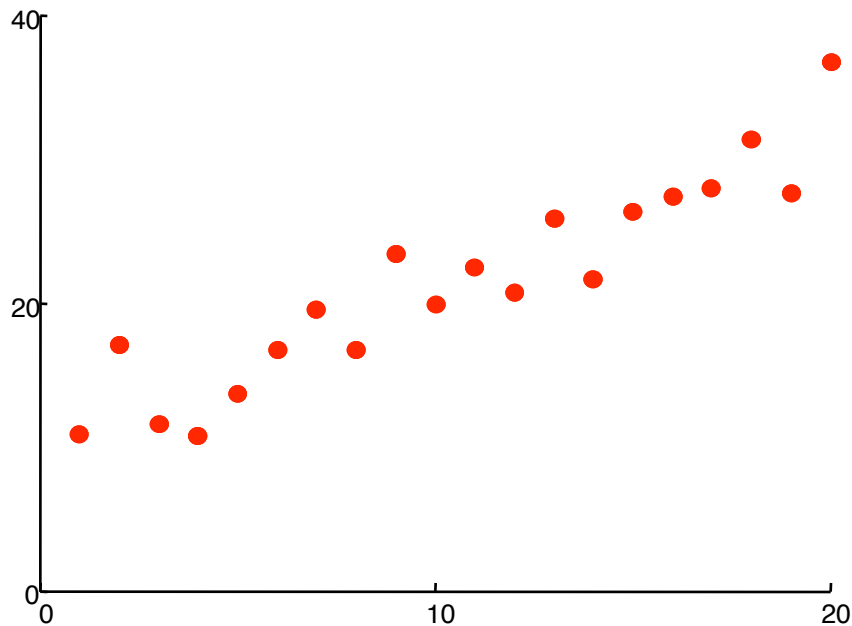
Figure from Guyon & Elisseeff

Feature Selection Methods

- Principal Components Analysis
- Non-Linear Principal Components Analysis
- Independent Components Analysis
- Information Gain
- Stepwise Discriminant Analysis
- Genetic Algorithms

Regression

Linear regression



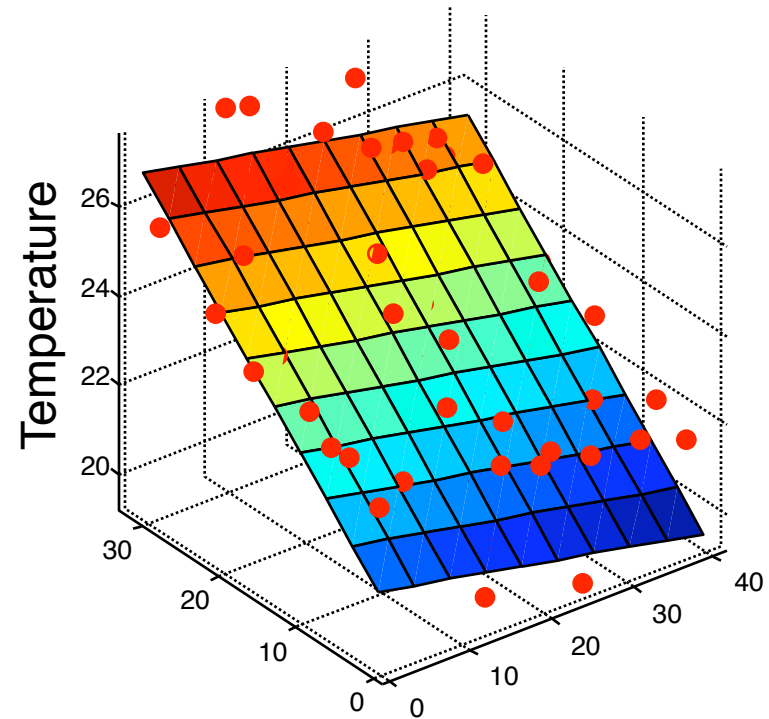
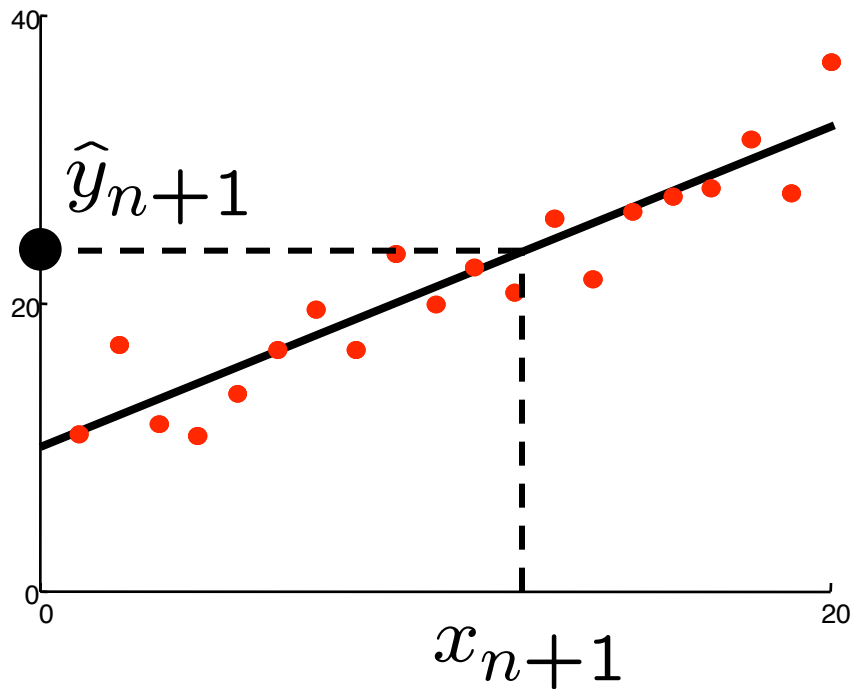
Given examples $(x_i, y_i)_{i=1 \dots n}$

Predict y_{n+1} given a new point x_{n+1}

Slide courtesy Roman Thibaux

[start Matlab demo lecture2.m]

Linear regression

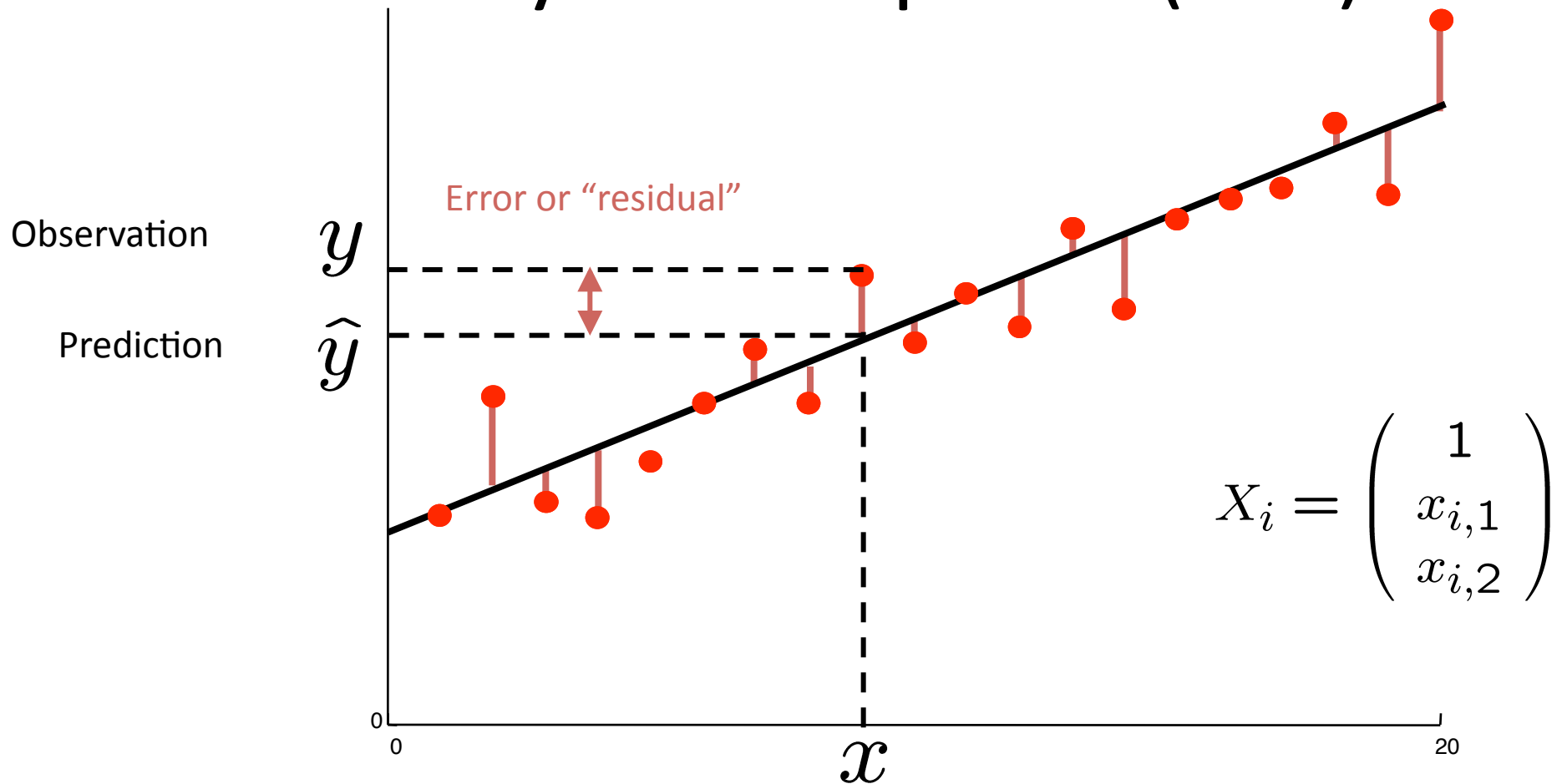


Prediction

$$\hat{y}_i = w_0 + w_1 x_i$$

$$\hat{y}_i = \begin{pmatrix} 1 & x_{i,1} & x_{i,2} \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ w_2 \end{pmatrix}$$

Ordinary Least Squares (OLS)



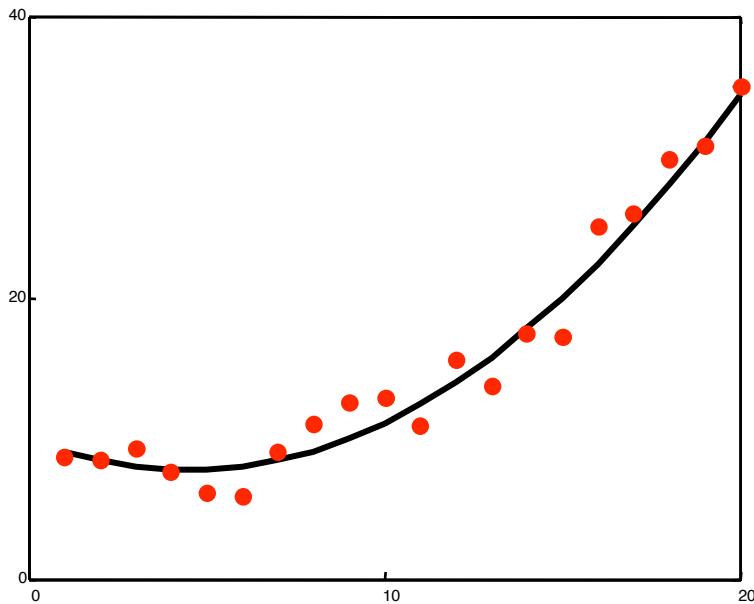
Sum squared error

Slide courtesy Roman Thibaux

$$\sum_i (X_i^\top w - y_i)^2$$

Beyond lines and planes

$$\hat{y}_i = w_0 + w_1x_i + w_2x_i^2$$



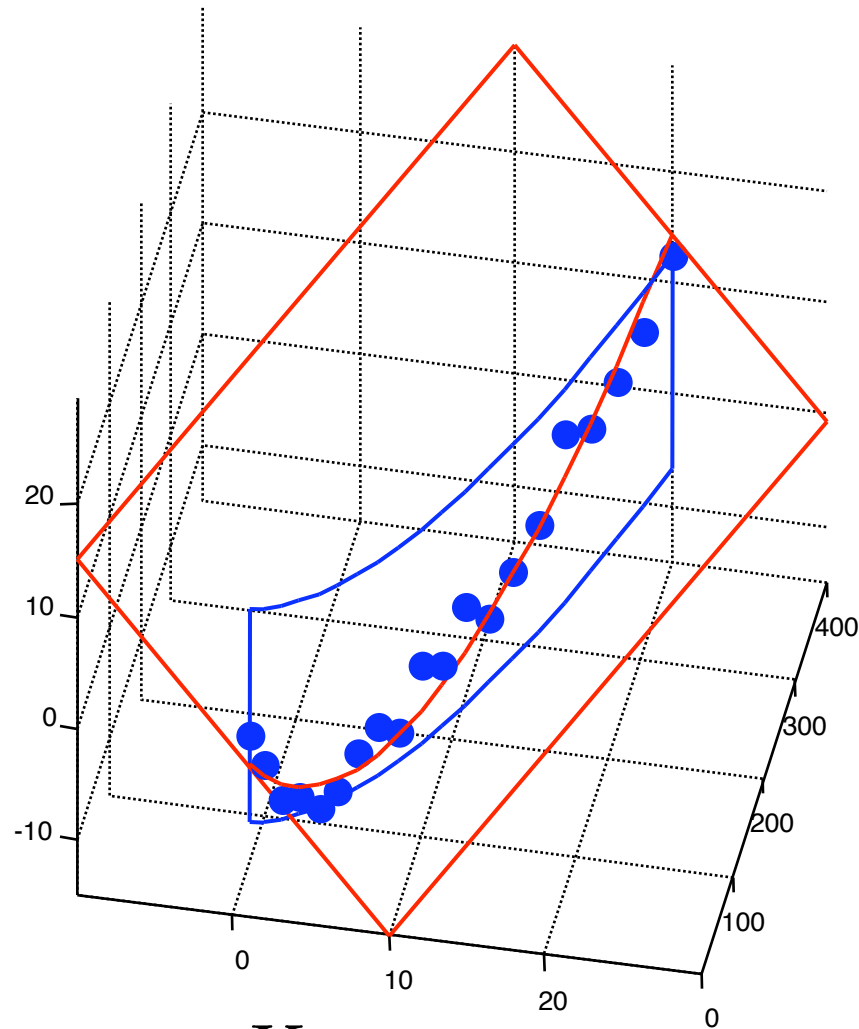
still linear in w

everything is the same with

$$X_i = \begin{pmatrix} 1 \\ x_i \\ x_i^2 \end{pmatrix}$$

Geometric interpretation

$$\hat{y} = w_1 x + w_2 x^2$$



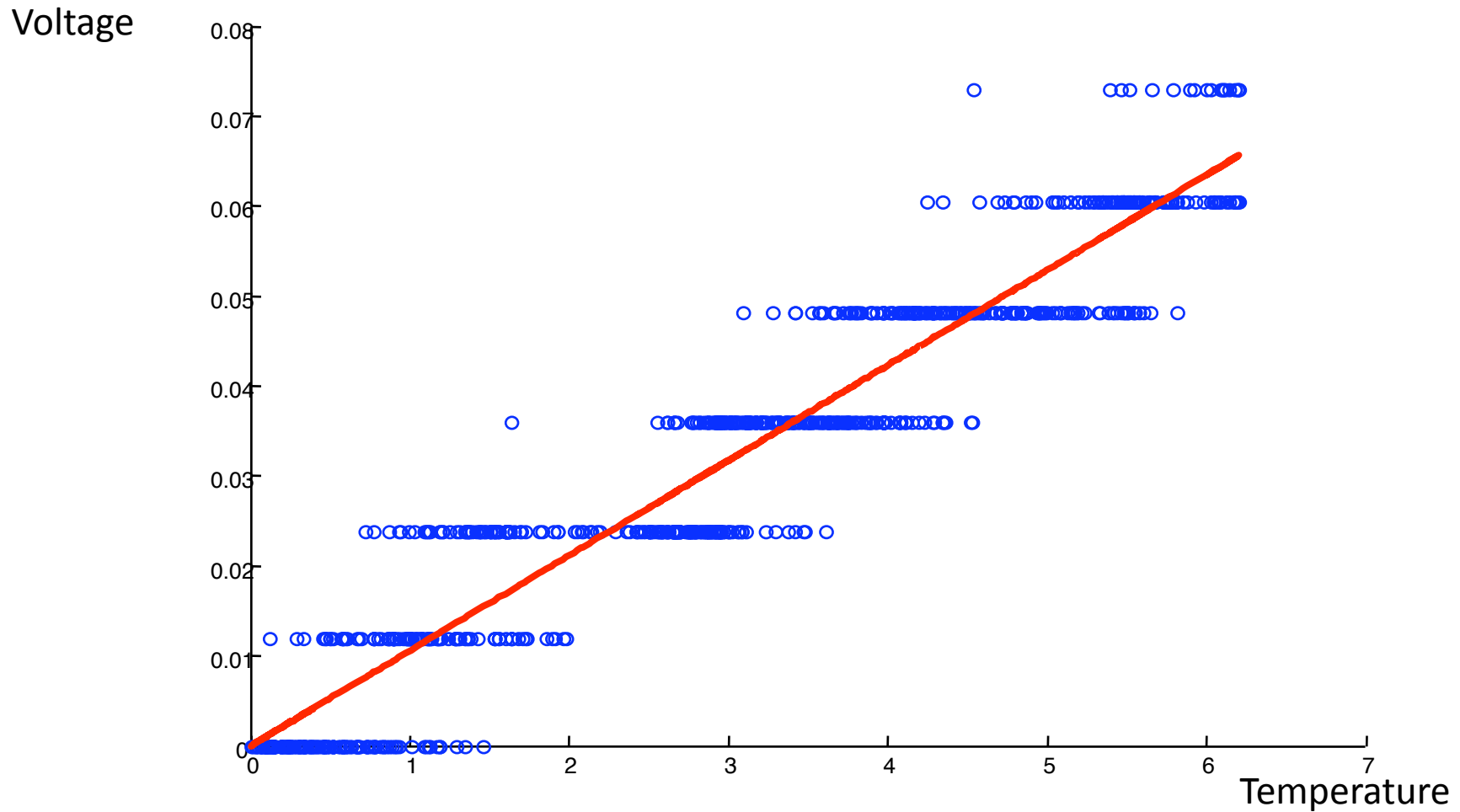
$$X_2 = x^2$$

$$X_1 = x$$

Slide courtesy Roman Thibaux

[Matlab demo]

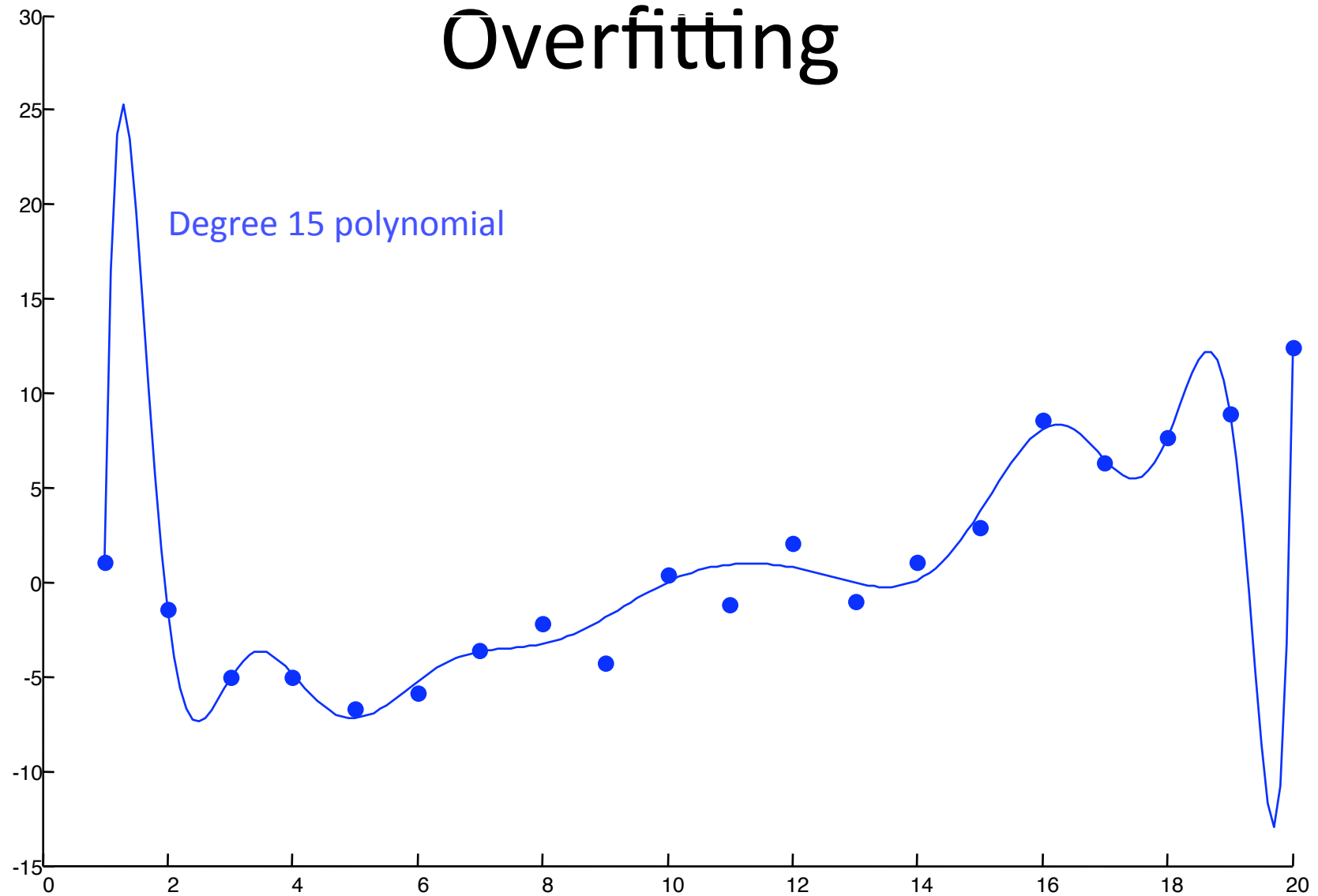
Assumptions vs. Reality



Intel sensor network data

Slide courtesy Roman Thibaux

Overfitting



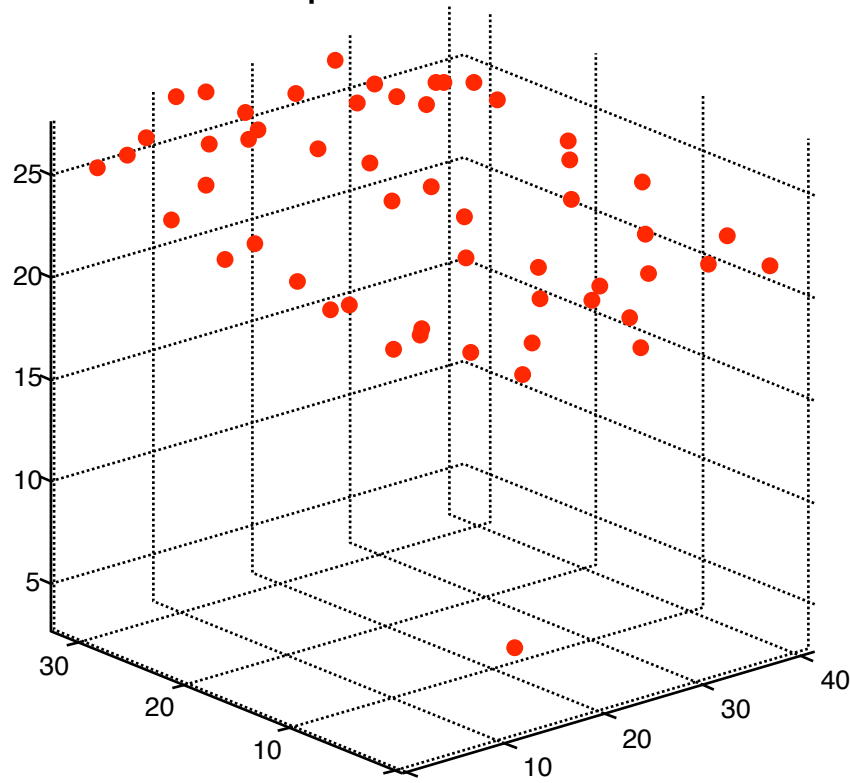
Slide courtesy Roman Thibaux

[Matlab demo]

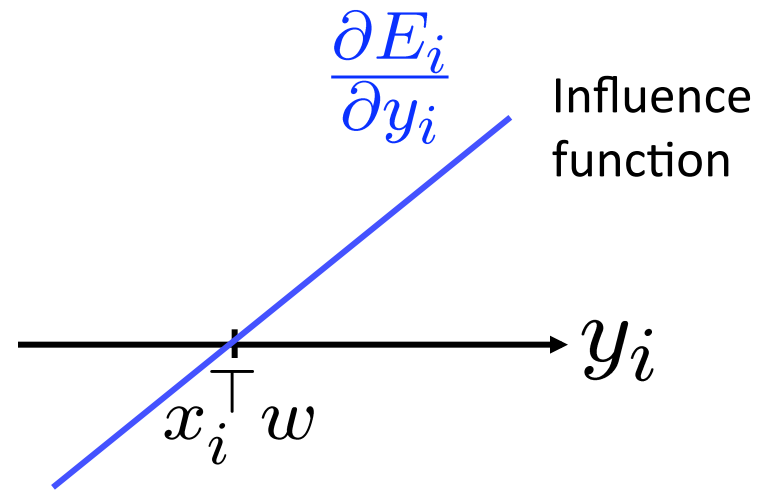
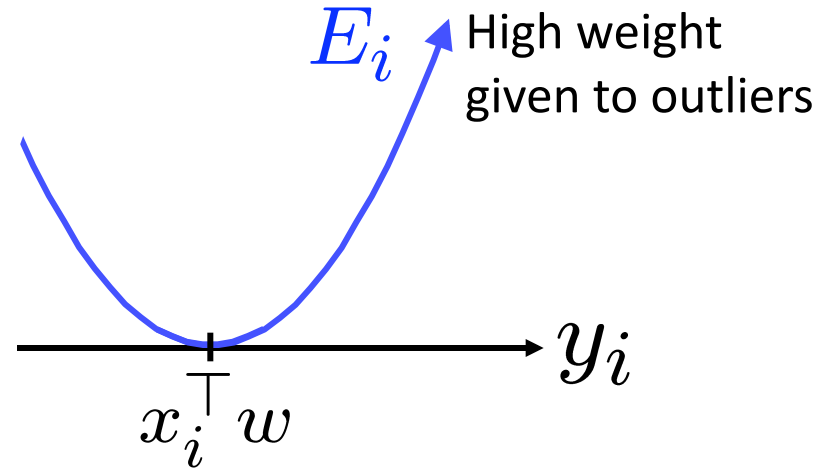
Sensitivity to outliers

$$E = \sum_i (x_i^\top w - y_i)^2 = \sum_i E_i$$

Temperature at noon

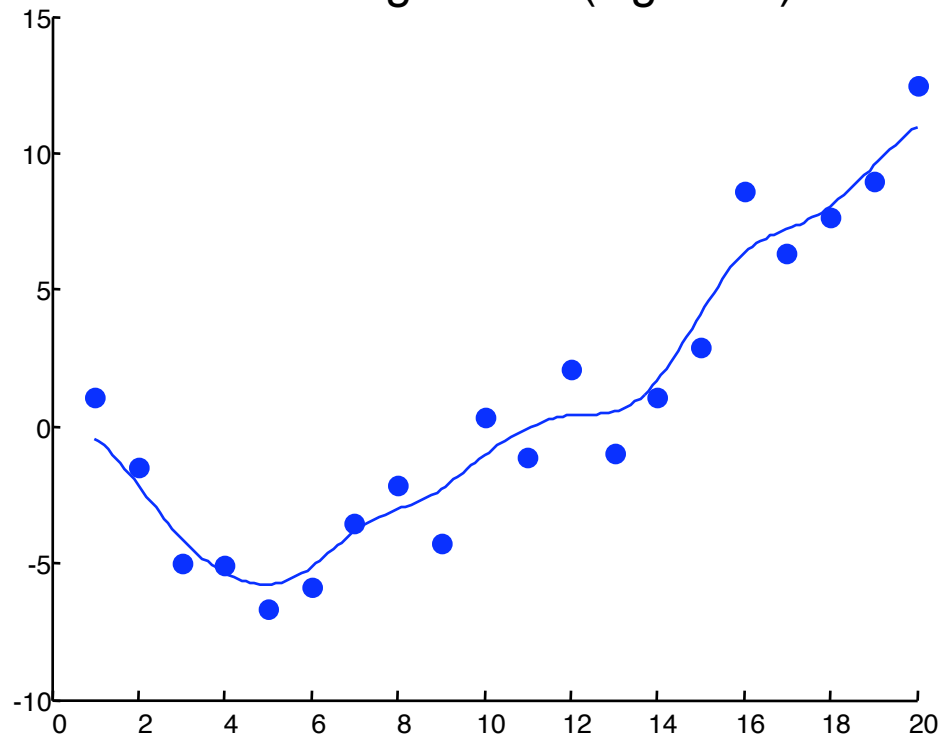


Slide courtesy Roman Thibaux

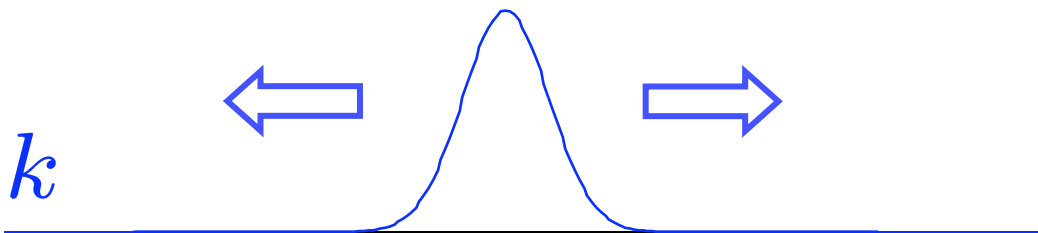


Kernel Regression

Kernel regression (sigma=1)

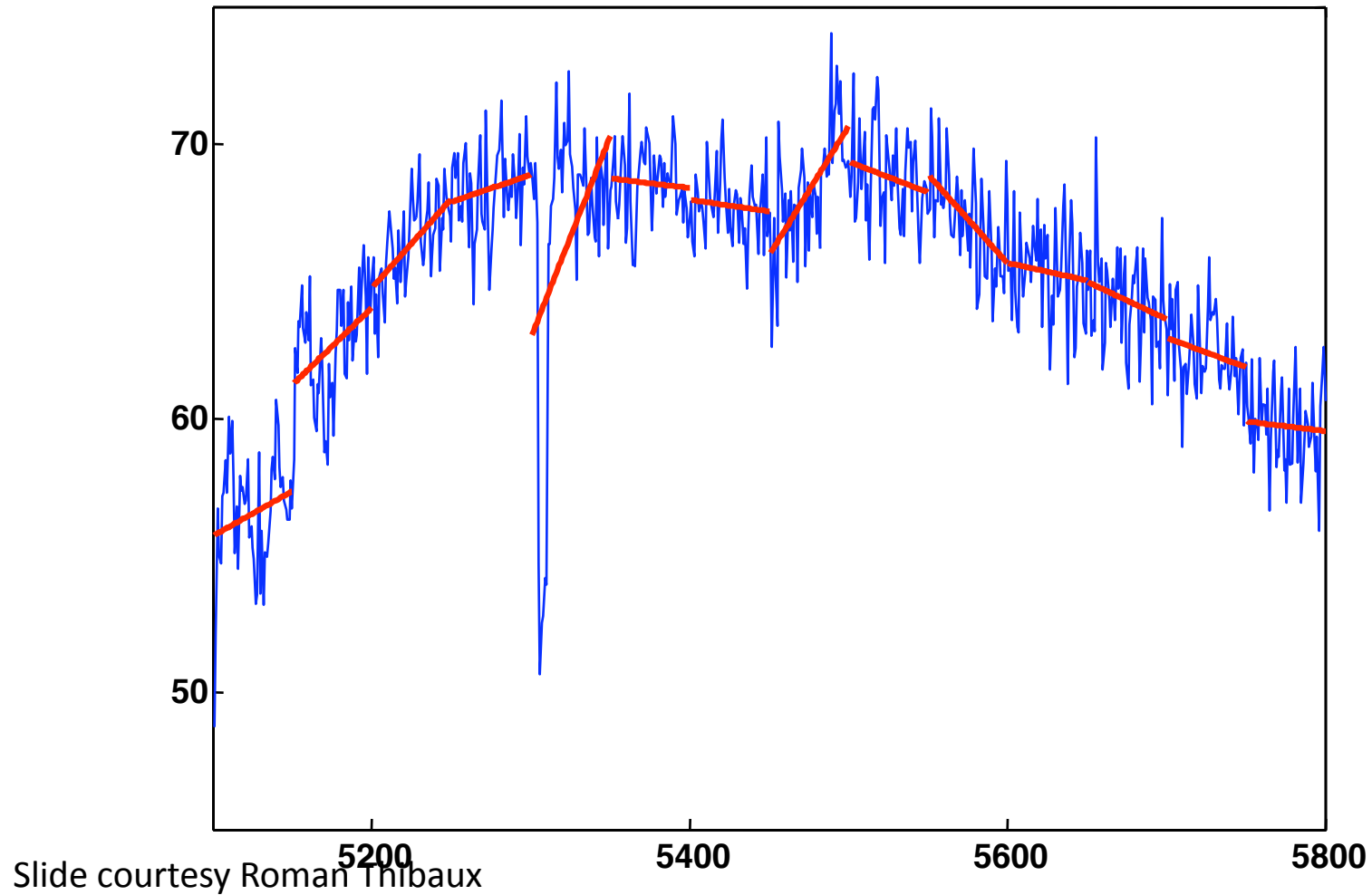


$$\hat{y}(x) = \frac{\sum_i y_i k(x_i - x)}{\sum_i k(x_i - x)}$$



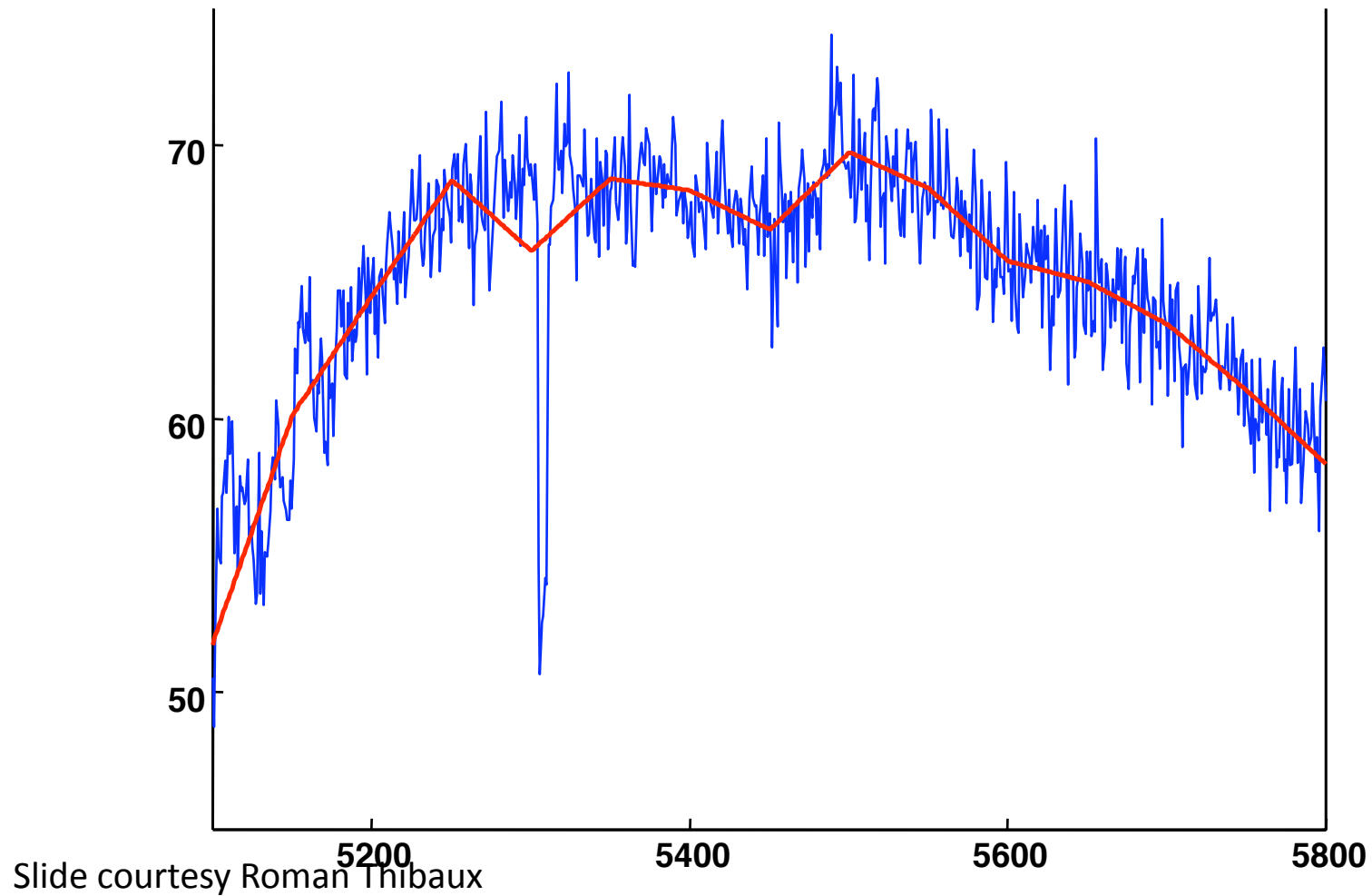
Spline Regression

Regression on each interval



Spline Regression

With equality constraints



Cluster analysis

- Supervised learning (Classification) assumes classes are known
- Unsupervised learning (Cluster analysis) seeks to discover the classes

Formal description

- Given X as a set of *instances* described by *features*
- Given an *objective function* g
- Given a *partition space* H
- Determine a partition h in H such that $h(X)$ maximizes/minimizes $g(h(X))$

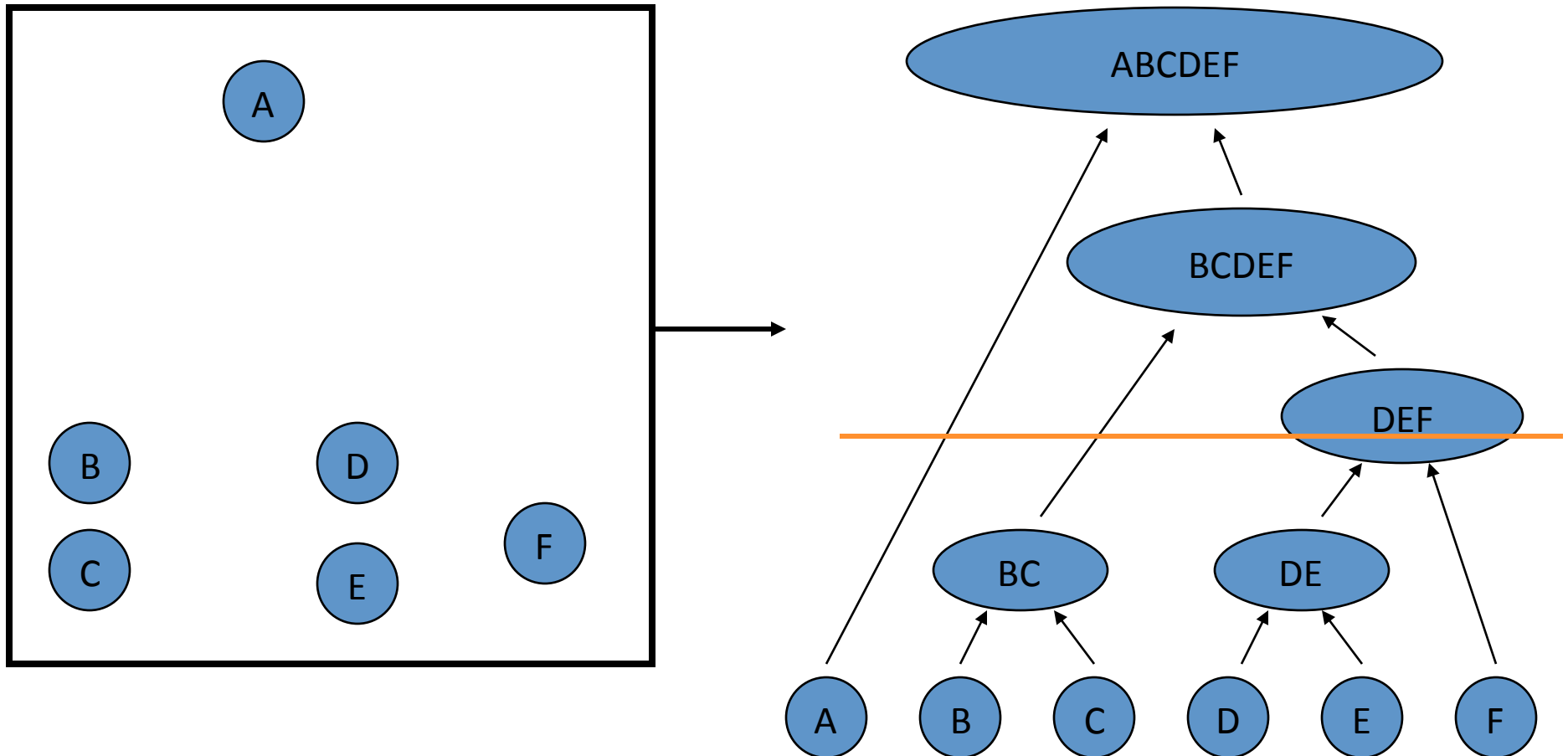
Formal description

- *objective function* g often stated in terms of minimizing a *distance function* d
- Example: Euclidean distance

Hierarchical vs. k -means clustering

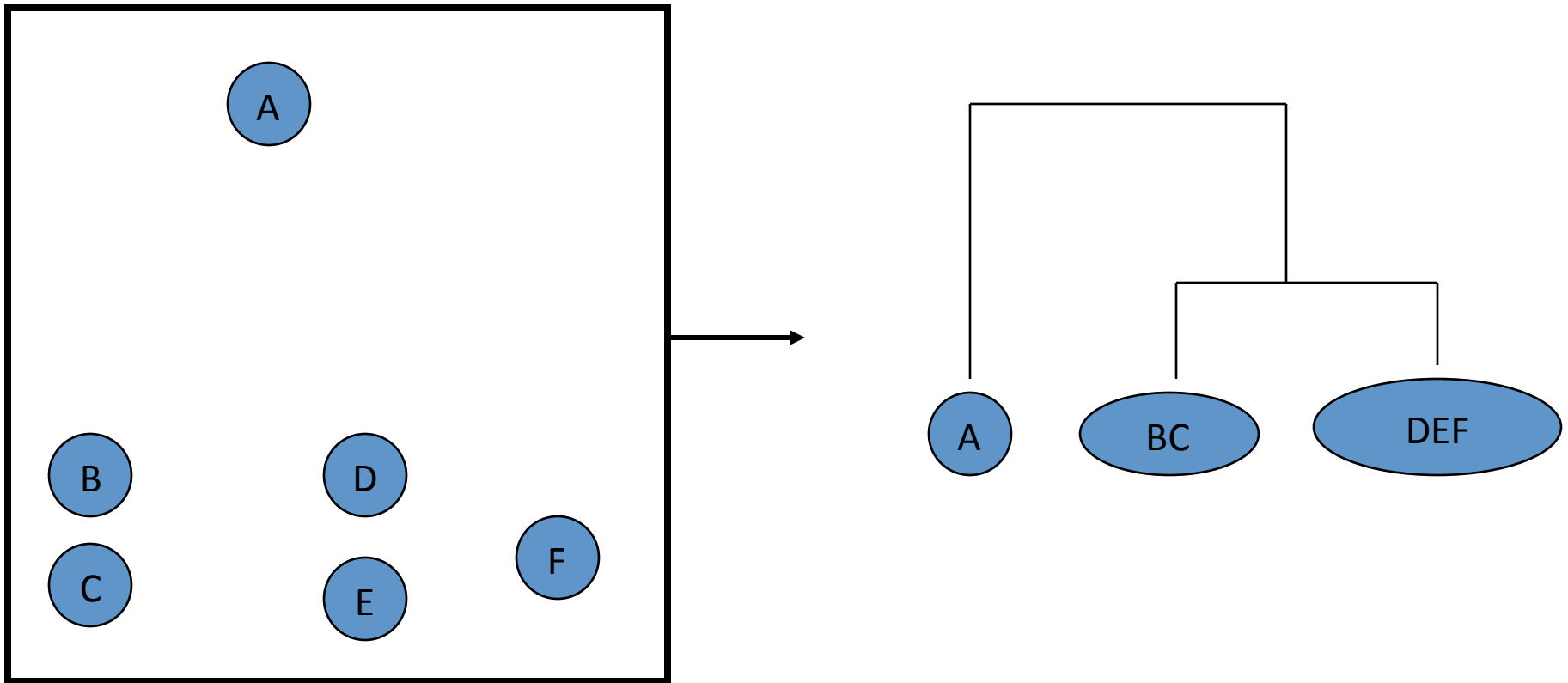
- Two most popular clustering algorithms
- Hierarchical builds tree sequentially from the closest pair of points (wells/cells/probes/conditions)
- k -means starts with k randomly chosen seed points, assigns each remaining point to the nearest seed, and repeats this until no point moves

Hierarchical Clustering



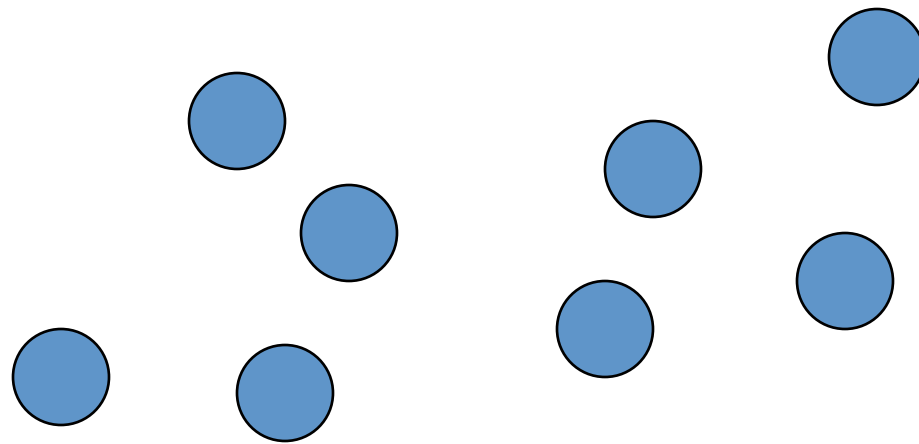
Slide courtesy of Elvira Garcia Osuna

Hierarchical Clustering



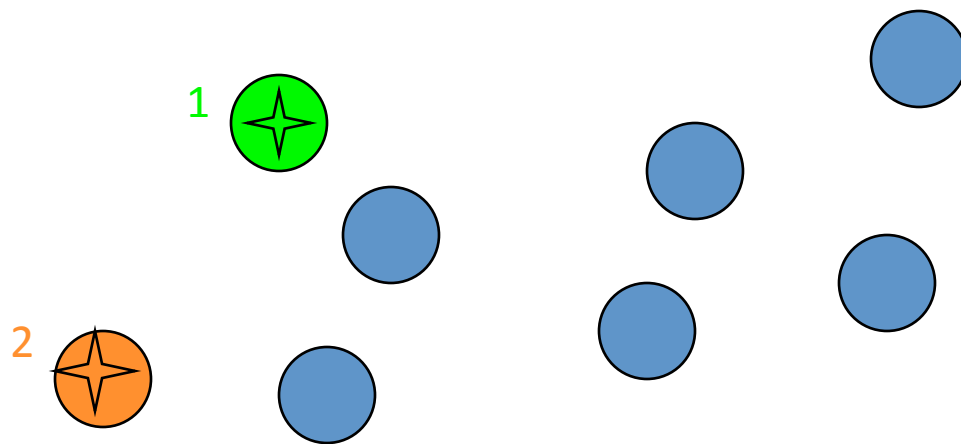
Slide courtesy of Elvira Garcia Osuna

K-means



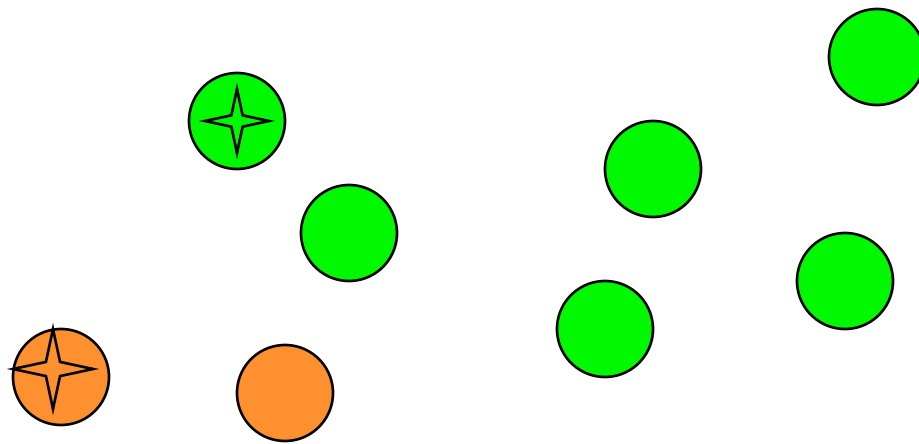
Slide courtesy of Elvira Garcia Osuna

K-means



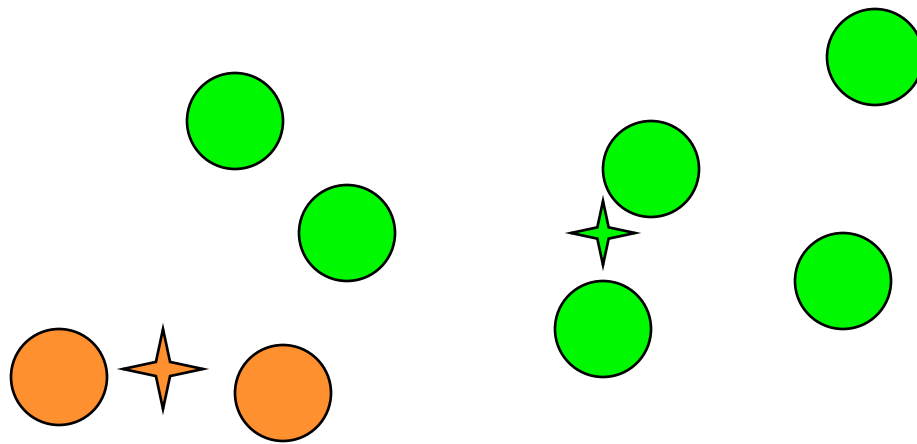
Slide courtesy of Elvira Garcia Osuna

K-means



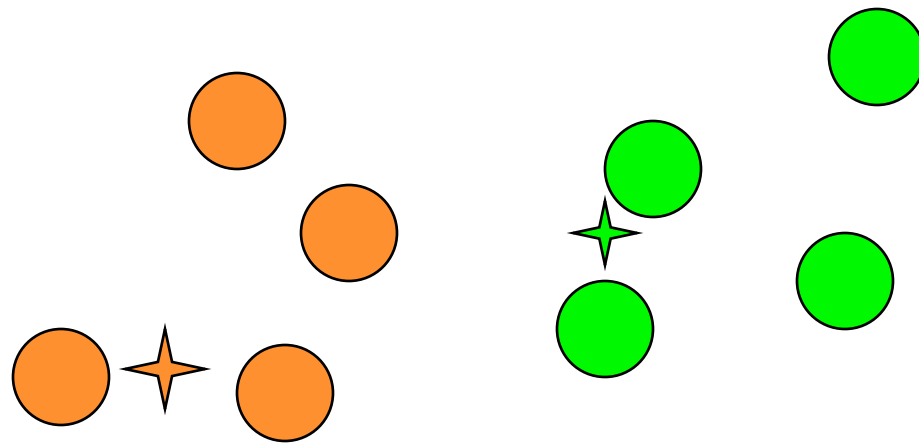
Slide courtesy of Elvira Garcia Osuna

K-means



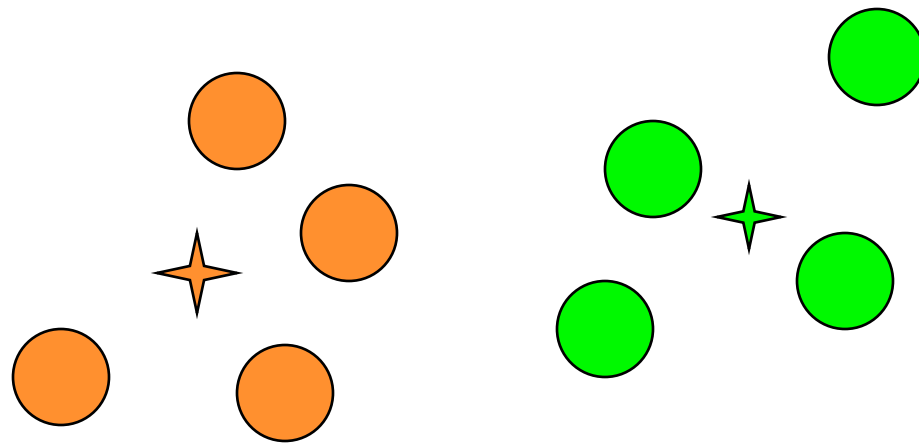
Slide courtesy of Elvira Garcia Osuna

K-means



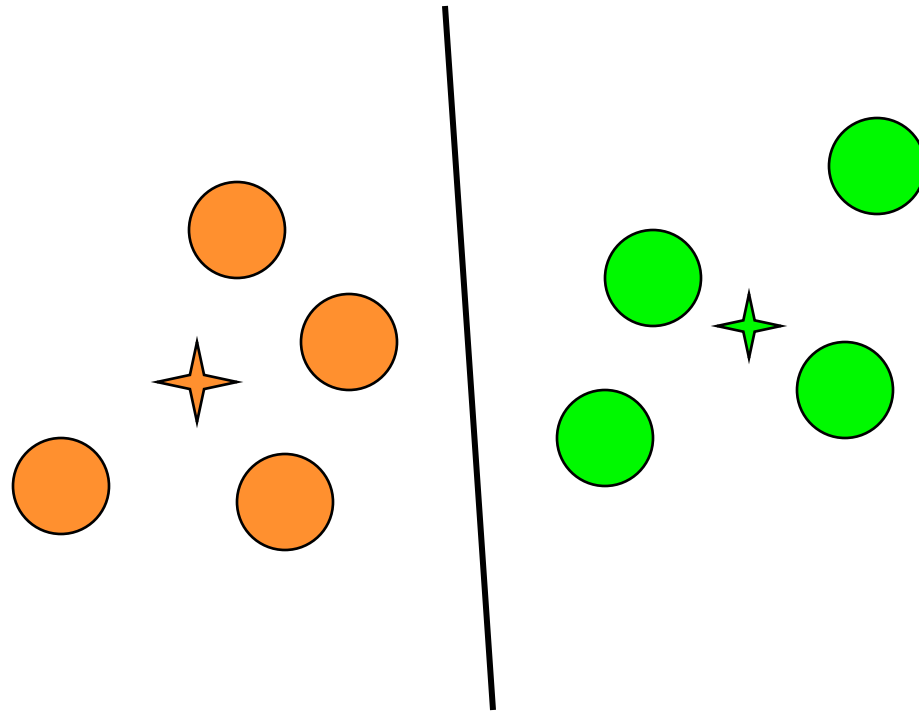
Slide courtesy of Elvira Garcia Osuna

K-means



Slide courtesy of Elvira Garcia Osuna

K-means

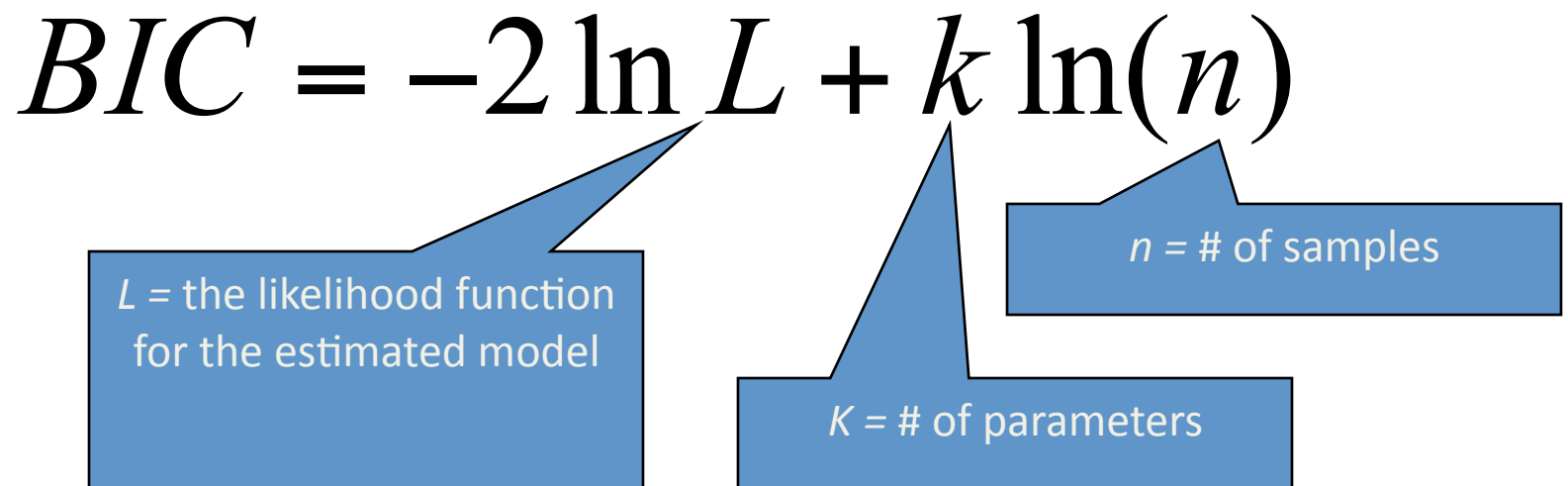


Slide courtesy of Elvira Garcia Osuna

Choosing the number of Clusters

- A difficult problem
- Most common approach is to try to find the solution that minimizes the Bayesian Information Criterion

$$BIC = -2 \ln L + k \ln(n)$$



L = the likelihood function
for the estimated model

K = # of parameters

n = # of samples

Microarray raw data

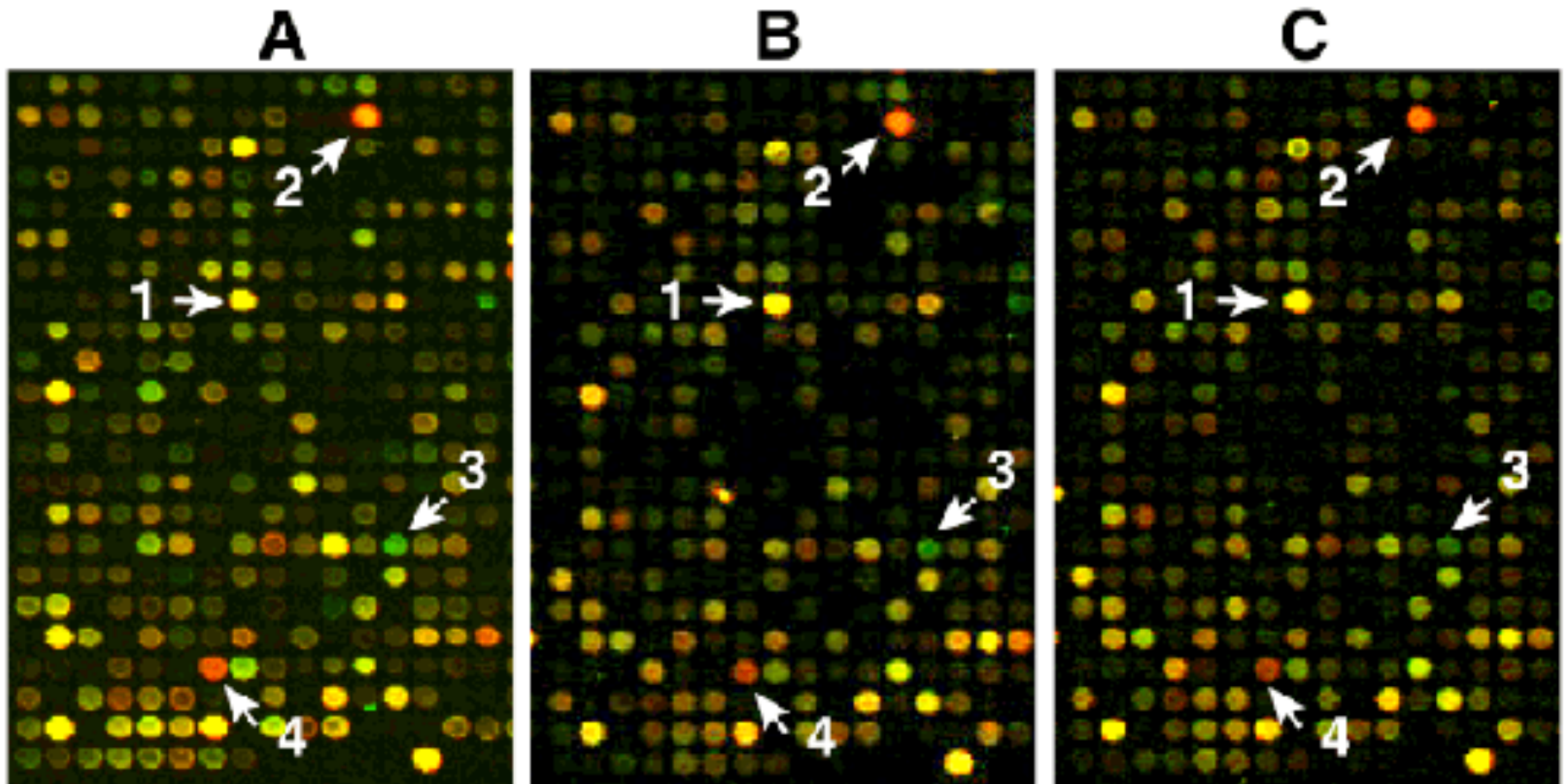
- Label mRNA from one sample with a red fluorescence probe (Cy5) and mRNA from another sample with a green fluorescence probe (Cy3)
- Hybridize to a chip with specific DNAs fixed to each well
- Measure amounts of green and red fluorescence

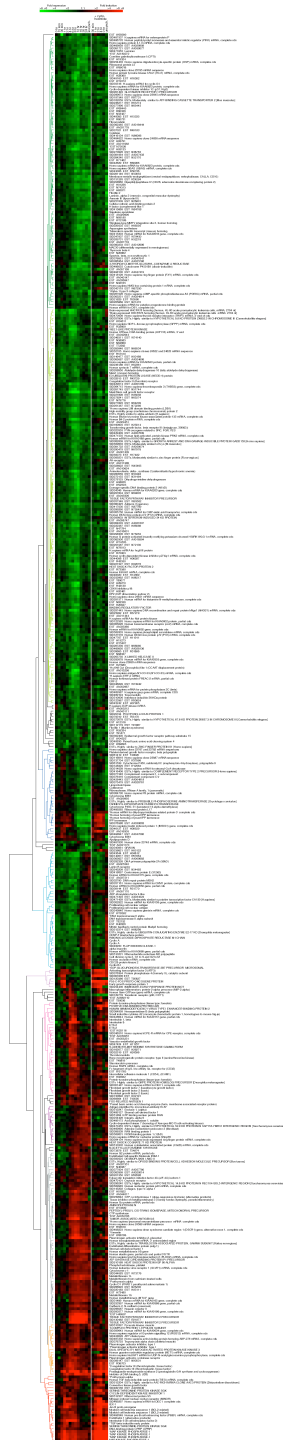
Flash animations:

PCR <http://www.maxanim.com/genetics/PCR/PCR.htm>

Microarray <http://www.bio.davidson.edu/Courses/genomics/chip/chip.html>

Example microarray image





mRNA expression microarray data for 9800 genes (gene number shown vertically) for 0 to 24 h (time shown horizontally) after addition of serum to a human cell line that had been deprived of serum (from <http://genome-www.stanford.edu/serum>)

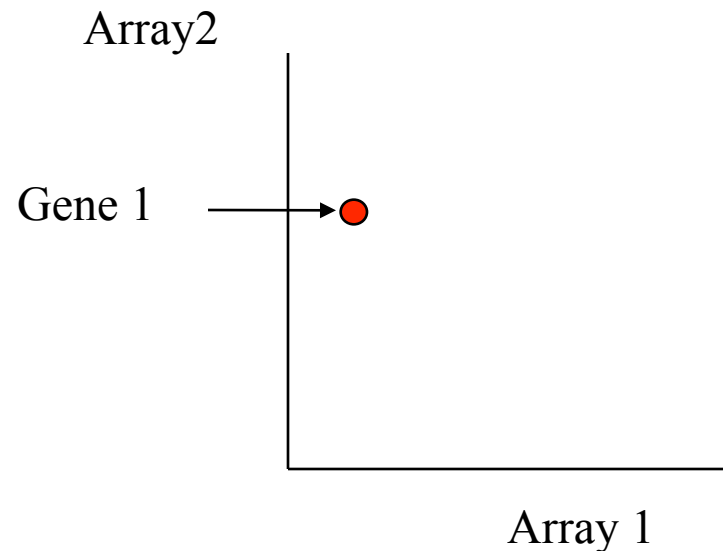
Data extraction

- Adjust fluorescent intensities using standards (as necessary)
- Calculate ratio of red to green fluorescence
- Convert to \log_2 and round to integer
- Display saturated green=-3 to black = 0 to saturated red = +3

Distances

- High dimensionality
- Based on **vector geometry** – how close are two data points?

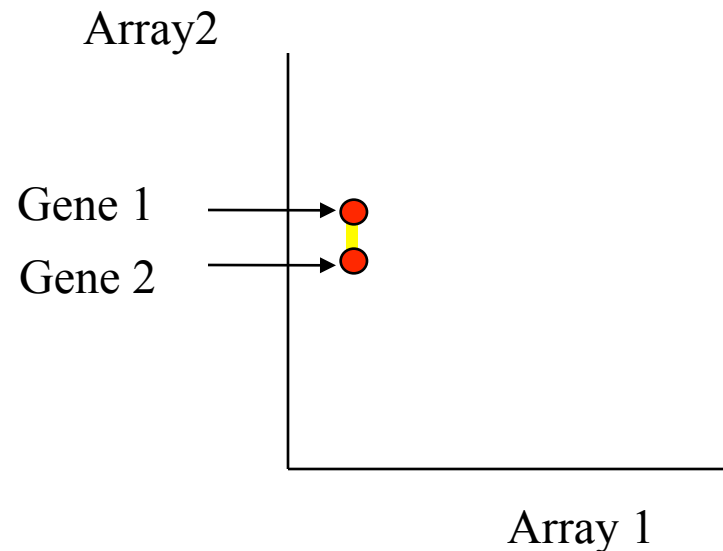
	Array 1	Array 2
Gene 1	1	4
...		



Distances

- High dimensionality
- Based on **vector geometry** – how close are two data points?

	Array 1	Array 2
Gene 1	1	4
Gene 2	1	3
...		

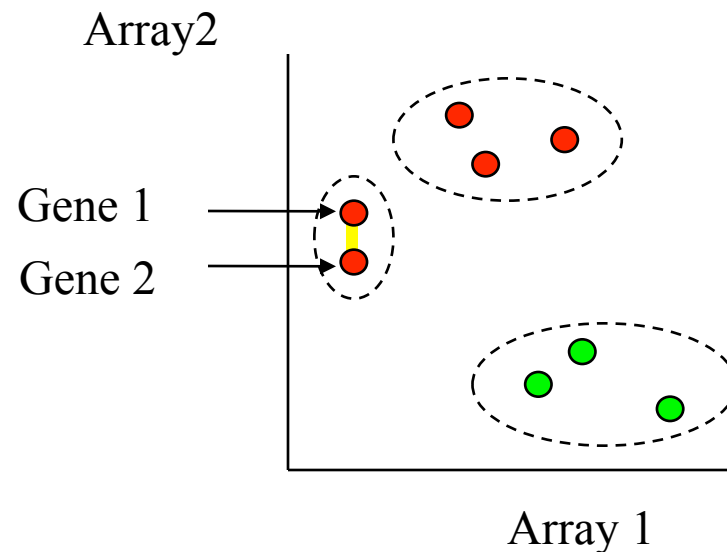


$$\text{Distance}(\text{Gene 1}, \text{Gene 2}) = 1$$

Distances

- High dimensionality
- Based on **vector geometry** – how close are two data points?
- Use distances to determine clusters

	Array 1	Array 2
Gene 1	1	4
Gene 2	1	3
...		



Distance(Gene 1, Gene 2) = 1

General Multivariate Dataset

- We are given values of p variables for n independent observations
- Construct an $n \times p$ matrix **M** consisting of vectors \mathbf{X}_1 through \mathbf{X}_n each of length p

Multivariate Sample Mean

- Define mean vector \mathbf{I} of length p

$$\mathbf{I}(j) = \frac{\sum_{i=1}^n \mathbf{M}(i, j)}{n}$$

matrix notation

or

$$\mathbf{I} = \frac{\sum_{i=1}^n \mathbf{X}_i}{n}$$

vector notation

Multivariate Variance

- Define variance vector σ^2 of length p

$$\sigma^2(j) = \frac{\sum_{i=1}^n (\mathbf{M}(i, j) - \mathbf{I}(j))^2}{n - 1}$$

matrix notation

Multivariate Variance

- or

$$\sigma^2 = \frac{\sum_{i=1}^n (\mathbf{X}_i - \mathbf{I})^2}{n - 1}$$

vector notation

Covariance Matrix

- Define a $p \times p$ matrix **cov** (called the **covariance matrix**) analogous to σ^2

$$\mathbf{cov}(j,k) = \frac{\sum_{i=1}^n (\mathbf{M}(i,j) - \mathbf{I}(j))(\mathbf{M}(i,k) - \mathbf{I}(k))}{n - 1}$$

Covariance Matrix

- Note that the covariance of a variable with itself is simply the variance of that variable

$$\mathbf{cov}(j, j) = \sigma^2(j)$$

Univariate Distance

- The simple distance between the values of a single variable j for two observations i and l is

$$\mathbf{M}(i, j) - \mathbf{M}(l, j)$$

Univariate z-score Distance

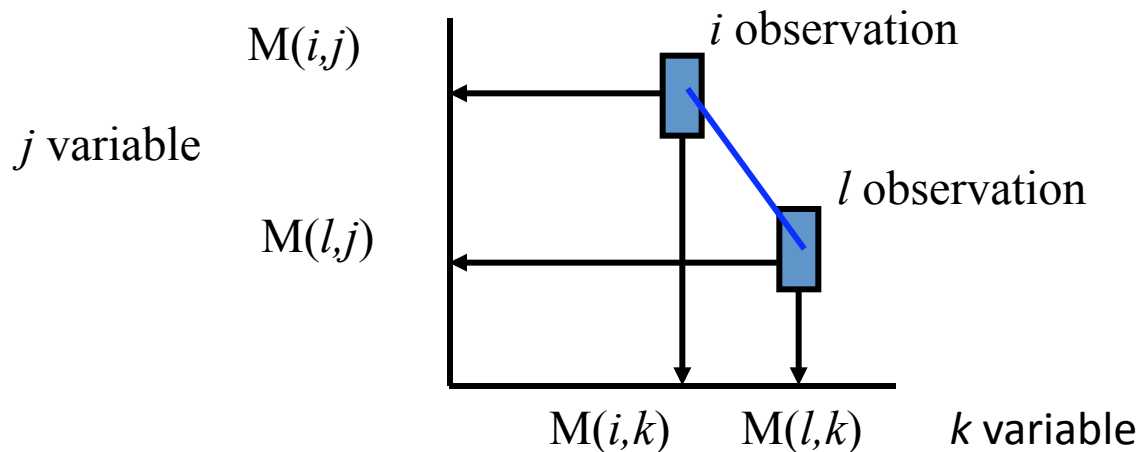
- To measure distance ***in units of standard deviation*** between the values of a single variable j for two observations i and l we define the **z-score distance**

$$\frac{\mathbf{M}(i, j) - \mathbf{M}(l, j)}{\sigma(j)}$$

Bivariate Euclidean Distance

- The most commonly used measure of distance between two observations i and l on two variables j and k is the **Euclidean distance**

$$\sqrt{(\mathbf{M}(i,j) - \mathbf{M}(l,j))^2 + (\mathbf{M}(i,k) - \mathbf{M}(l,k))^2}$$

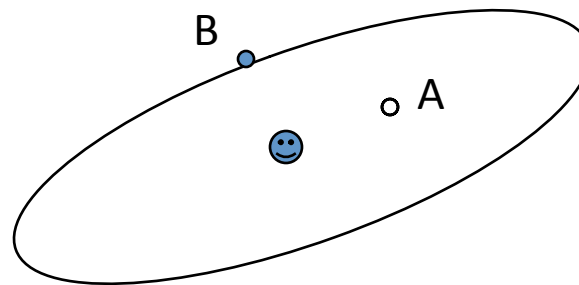


Multivariate Euclidean Distance

- This can be extended to more than two variables

$$\sqrt{\sum_{j=1}^p (\mathbf{M}(i, j) - \mathbf{M}(l, j))^2}$$

Effects of variance and covariance on Euclidean distance



The ellipse shows the 50% contour of a hypothetical population.

Points A and B have similar Euclidean distances from the mean, but point B is clearly “more different” from the population than point A.

Mahalanobis Distance

- To account for differences in variance between the variables, and to account for correlations between variables, we use the Mahalanobis distance

$$D^2 = (\mathbf{X}_i - \mathbf{X}_l) \mathbf{cov}^{-1} (\mathbf{X}_i - \mathbf{X}_l)^T$$

Other distance functions

- We can use other distance functions, including ones in which the weights on each variable are learned
- Cluster analysis tools for microarray data most commonly use Pearson correlation coefficient

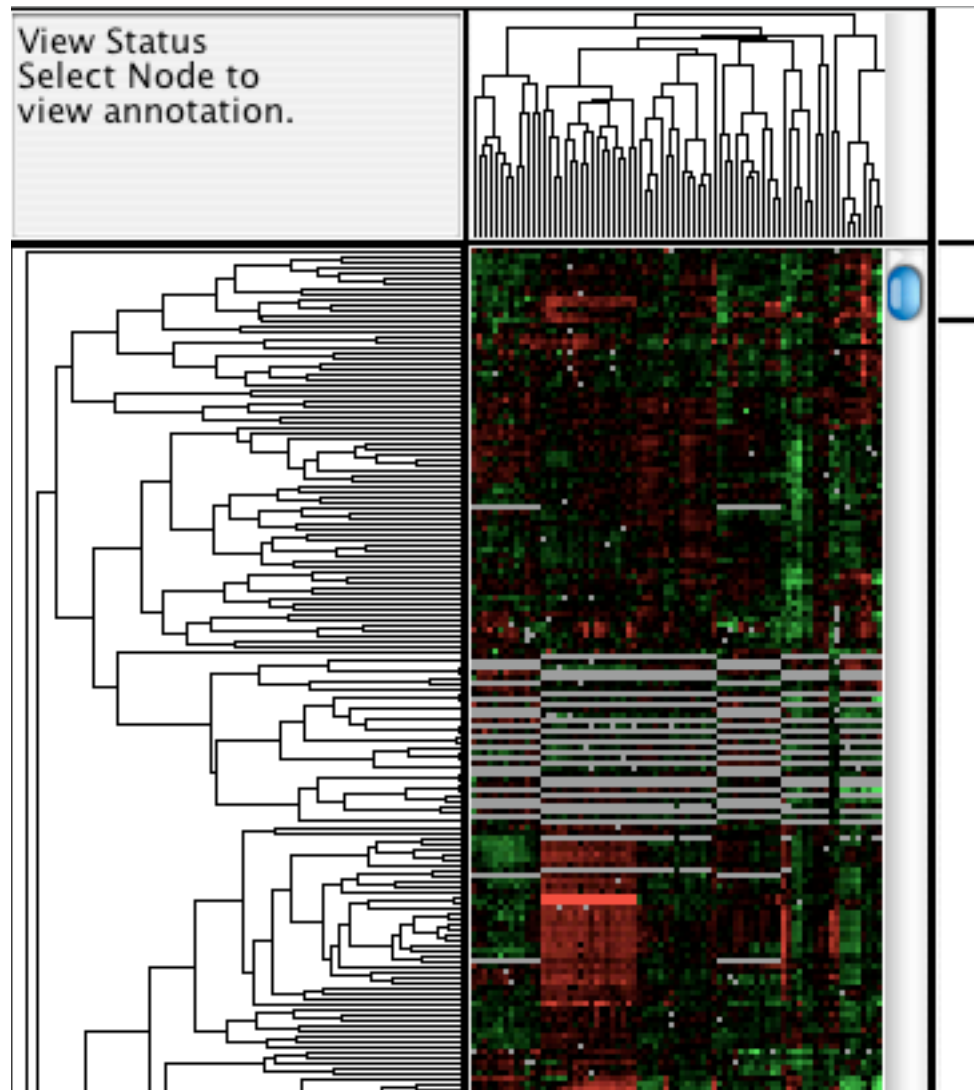
Input data for clustering

- Genes in rows, conditions in columns

YORF	NAME	GWEIGHT	Cell-cycle	Alph	Cell-cycle	Alph	Cell-cycle	Alph
EWEIGHT				1		1		1
YHR051W	YHR051W	CC	1	0.03	0.3		0.37	
YKL181W	YKL181W	PR	1	0.33	-0.2		-0.12	
YHR124W	YHR124W	NE	1	0.36	0.08		0.06	
YHL020C	YHL020C	OP	1	-0.01	-0.03		0.21	
YGR072W	YGR072W	UF	1	0.2	-0.43		-0.22	
YGR145W	YGR145W		1	0.11	-1.15		-1.03	
YGR218W	YGR218W	CF	1	0.24	-0.23		0.12	
YGL041C	YGL041C		1	0.06	0.23		0.2	
YOR202W	YOR202W	HI	1	0.1	0.48		0.86	
YCR005C	YCR005C	CI	1	0.34	1.46		1.23	
YER187W	YER187W		1	0.71	0.03		0.11	
YBR026C	YBR026C	MR	1	-0.22	0.14		0.14	
YMR244W	YMR244W		1	0.16	-0.18		-0.38	
YAR047C	YAR047C		1	-0.43	-0.56		-0.14	
YMR317W	YMR317W		1	-0.43	-0.03		0.21	

Clustering genes and conditions

- Rows and columns can be clustered independently - hierarchical is preferred for visualizing this



Stating Goals vs. Approaches

- Temptation when first considering using a machine learning approach to a biological problem is to describe the problem as automating the approach that you would solve the problem
- “I need a program to predict how much a gene is expressed by measuring how well its promoter matches a template”

Stating Goals vs. Approaches

- “I need a program that given a gene sequence predicts how much that gene is expressed by measuring how well its promoter matches a template”
- “I need a program that given a gene sequence predicts how much that gene is expressed by learning from sequences of genes whose expression is known”

Resources

- Association for the Advancement of Artificial Intelligence
 - <http://www.aaai.org/ALTopics/pmwiki/pmwiki.php/ALTopics/MachineLearning>
- Machine Learning – Mitchell, Carnegie Mellon
 - <http://www.cs.cmu.edu/afs/cs.cmu.edu/user/mitchell/ftp/mlbook.html>
- Practical Machine Learning – Jordan, UC Berkeley
 - <http://www.cs.berkeley.edu/~asimma/294-fall06/>
- Learning and Empirical Inference – Rish, Tesauero, Jebara, Vadpnik – Columbia
 - <http://www1.cs.columbia.edu/~jebara/6998/>