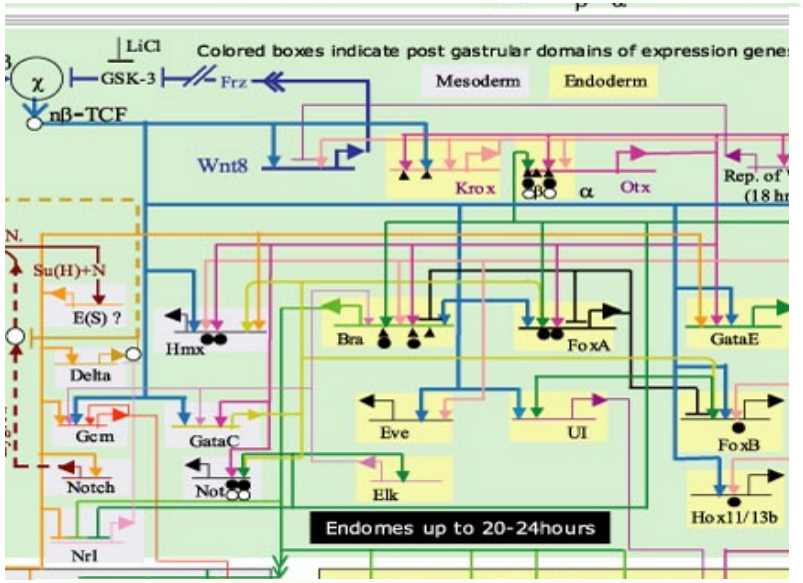
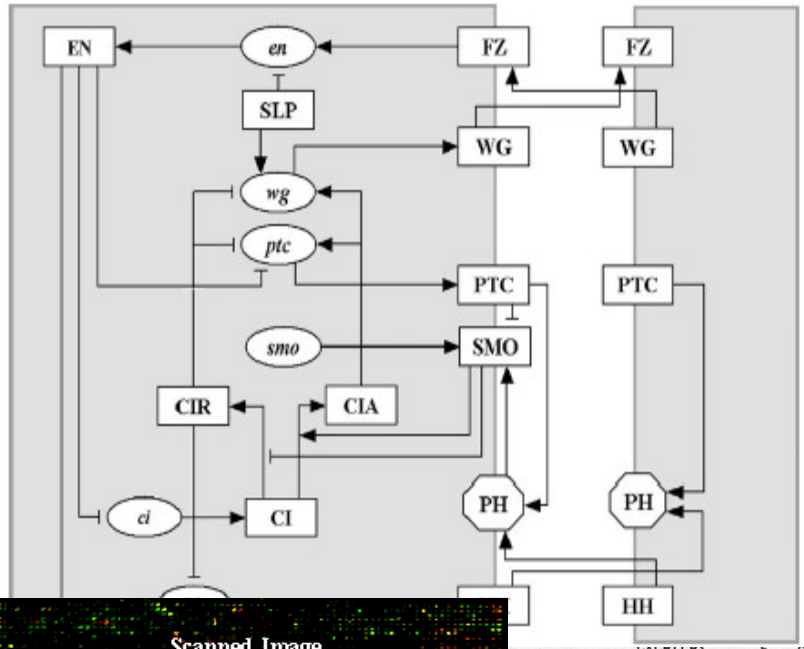
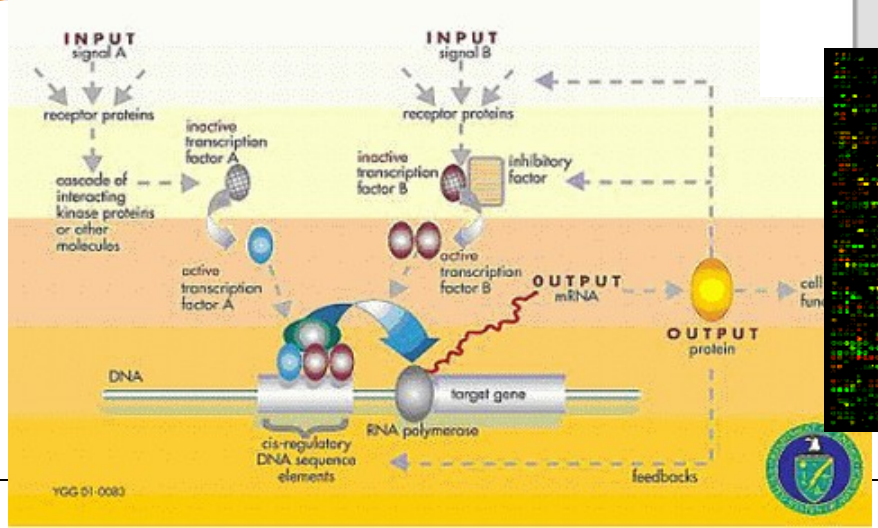
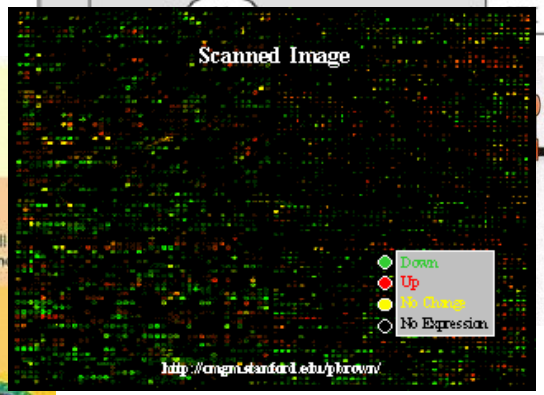
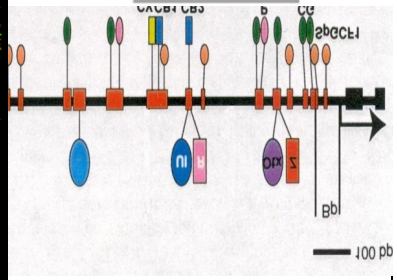
ECS 234: Data Analysis: Classification











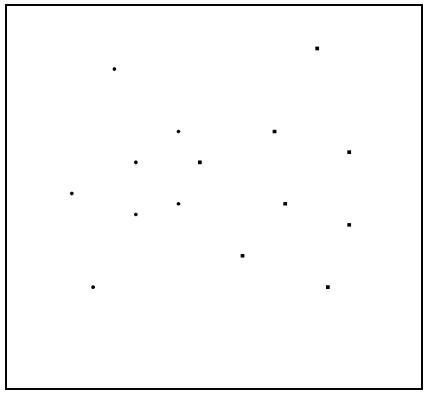
Classification vs. Clustering

- Classification:
  - Goal: Placing objects (e.g. genes) into meaningful classes
  - Supervised
- Clustering:
  - Goal: Discover meaningful classes
  - Unsupervised

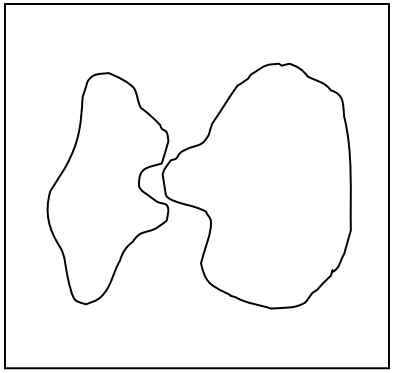
Classification vs. Clustering

- Classification
  - Needs meta-data
  - Can detect “weaker” patterns, but may be biased
- Clustering
  - No need for extra information
  - Patterns need to be strong in order to be discovered

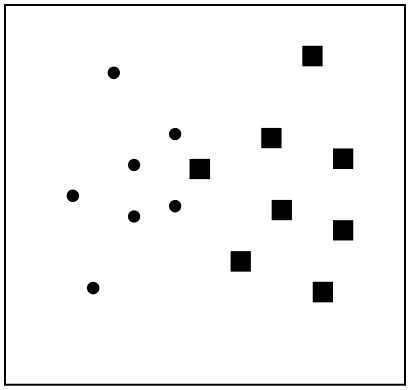
Classification



Data

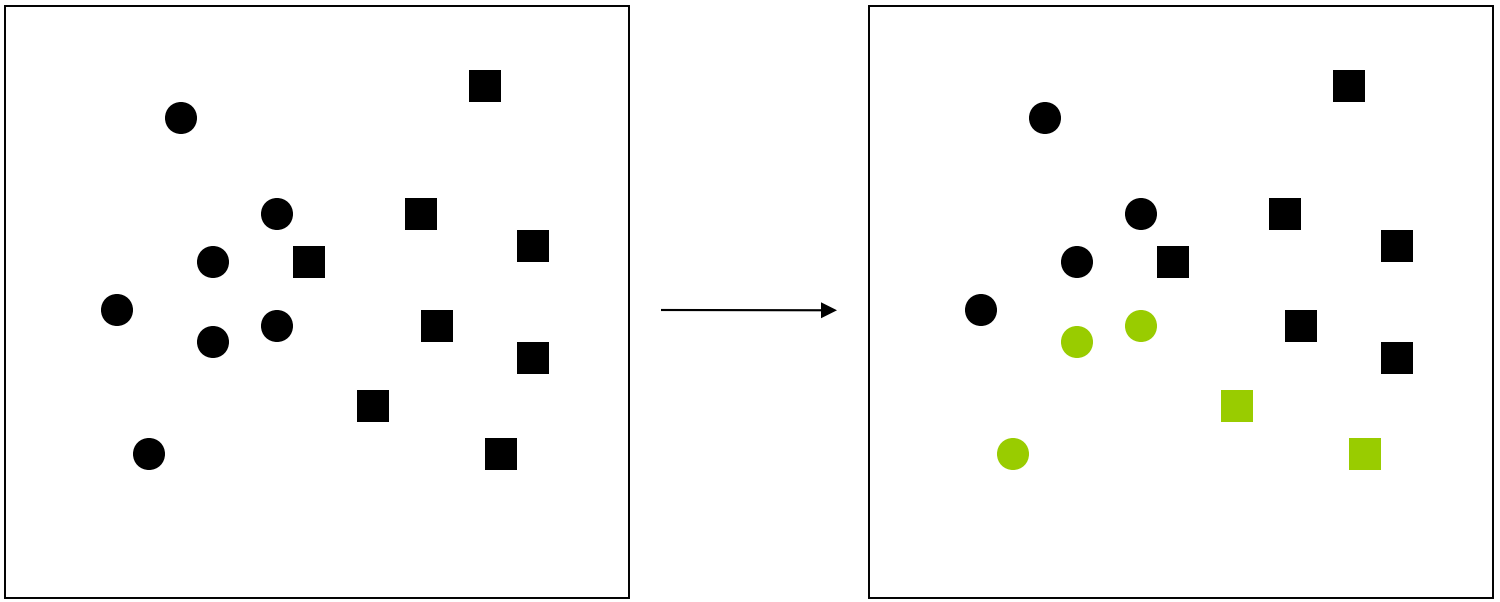


Classes
(knowledge)



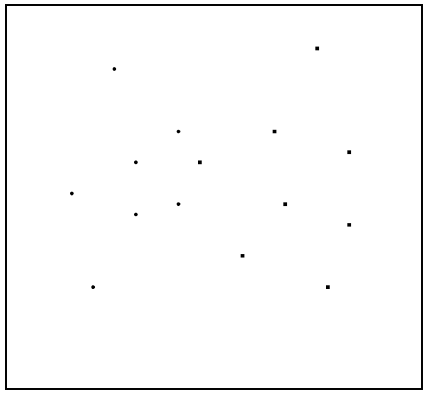
Classification

Supervised learning step

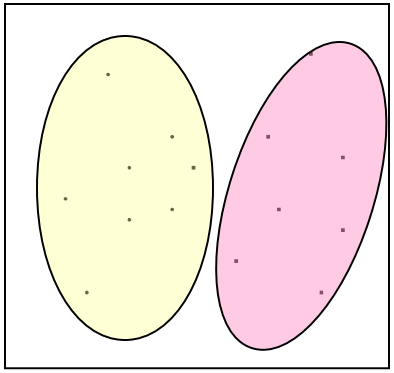


More a priori knowledge helps in identifying weaker patterns in data

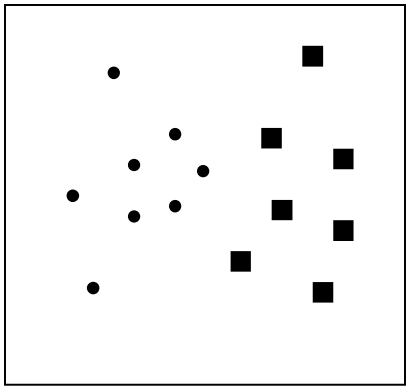
Clustering



Data



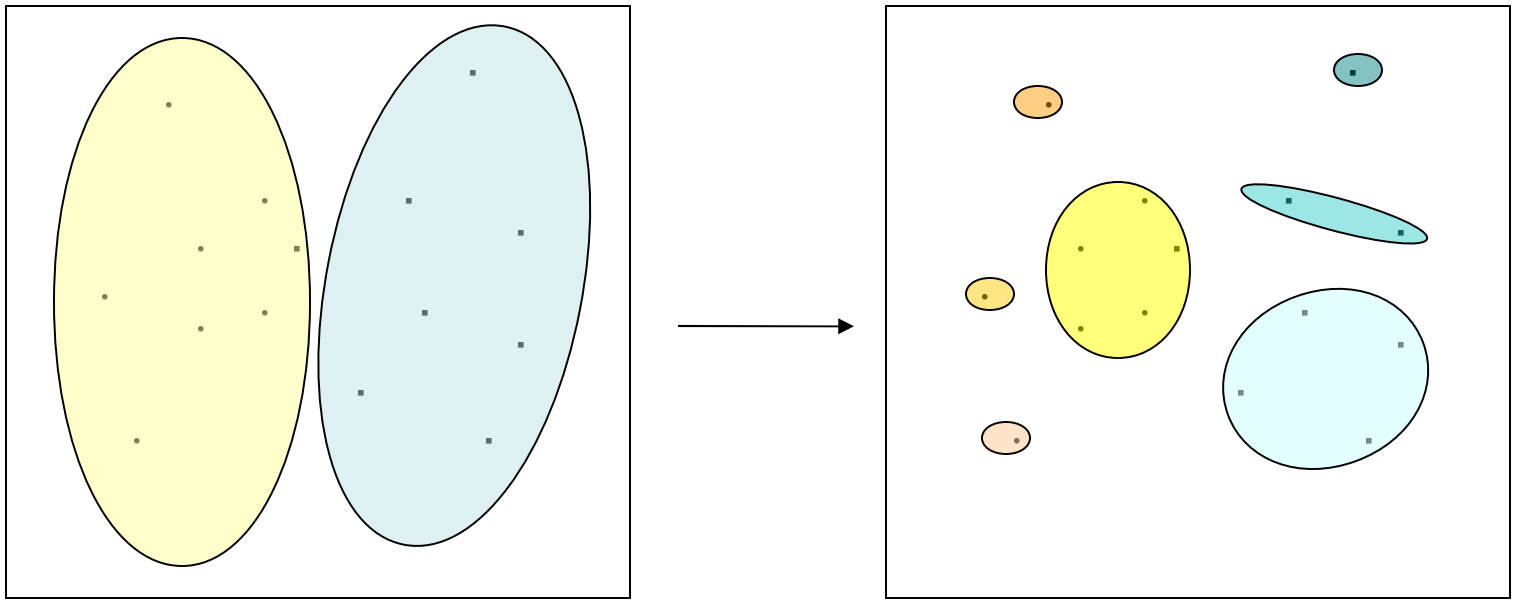
Distance based
cluster assignment



Clustering

Unsupervised learning step

Clustering



Further clustering reveals only very strong signals

Learning Methods in Computational Functional Genomics

Supervised (Classification)

(a) Single Feature
- Naïve bayes classifier

(b) Multiple Features
- Nearest Neighbor
- Decision Trees
- Gaussian Processes
- Neural Nets
- Support Vector Machines

Unsupervised (Clustering)

(a) Single Feature
- Nearest Neighbor
- Agglomerative Clustering (hierarchical)
- Partitional Clustering
  - K-Means
  - SOM

(b) Multiple Features
- Plaid Models
- Biclustering

Classification

- Linear nearest neighbor model
- Support Vector Machines

Molecular Classification of Cancer

(Golub et al, Science 1999)

Overview: General approach for cancer classification based on gene expression monitoring

The authors address both:

- Class Prediction (Assignment of tumors to known classes)
- Class Discovery (New cancer classes)

Cancer Classification

- Helps in prescribing necessary treatment
- Has been based primarily on morphological appearance
- Such approaches have limitations: similar tumors in appearance can be significantly different otherwise
- Needed: better classification scheme!

Cancer Data

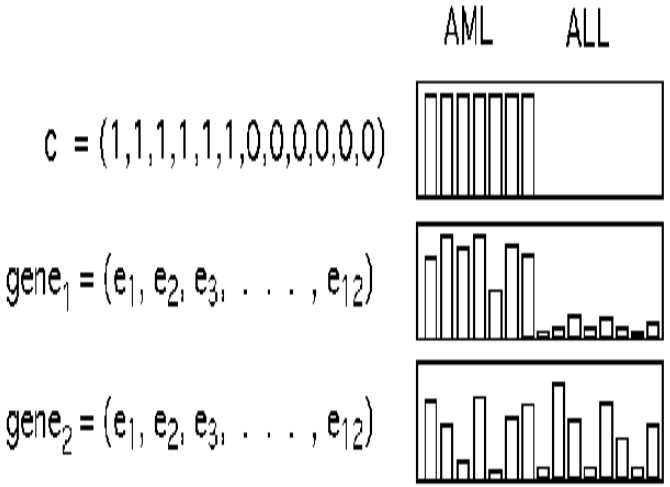
- Human Patients; Two Types of Leukemia
  - Acute Myeloid Leukemia
  - Acute Lymphoblastic Leukemia
- Oligo arrays data sets (6817 genes):
  - Learning Set, 38 bone marrow samples, 27 ALL, 11 AML
  - Test Set, 34 bone marrow samples, 20 ALL, 14 AML

Classification Based on Expression Data

1. Selecting the most informative genes
   - Class Distinctors
   - Used to predict the class of unclassified genes
2. Class Prediction (Classification)
   - Given a new gene, classify it based on the most informative genes
3. Class Discovery (Clustering)
   - Using Self Organizing Maps discover new classes of genes

1. Selecting "Class Distinctor" Genes

The goal is to select a number of genes whose expression profiles correlate significantly well with an idealized class distinction, c



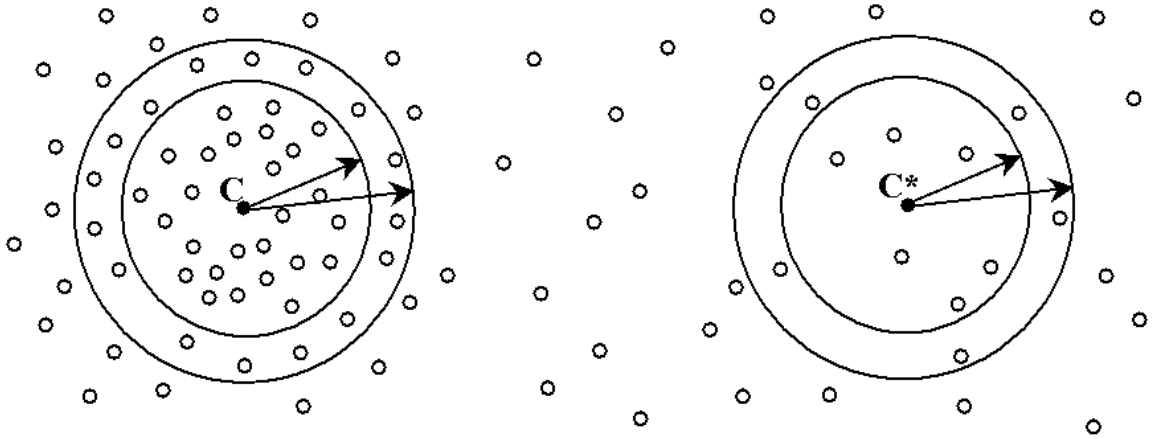
The class distinction is indicative of the two classes, and is uniformly high in the first (1=AML), and uniformly low for the second (0=ALL)

The correlation is calculated as: $P(g,c) = (\mu_1 - \mu_2)/(\sigma_1 - \sigma_2)$

Where μ_i's and σ_i's are the means and standard deviations of the log of expression levels of gene *g* for the samples in class AML and ALL.

Sufficient Information for Class Distinction?

To test whether there are informative genes based on c, the significance of having highly correlated gene patterns to c was assessed by neighborhood analysis



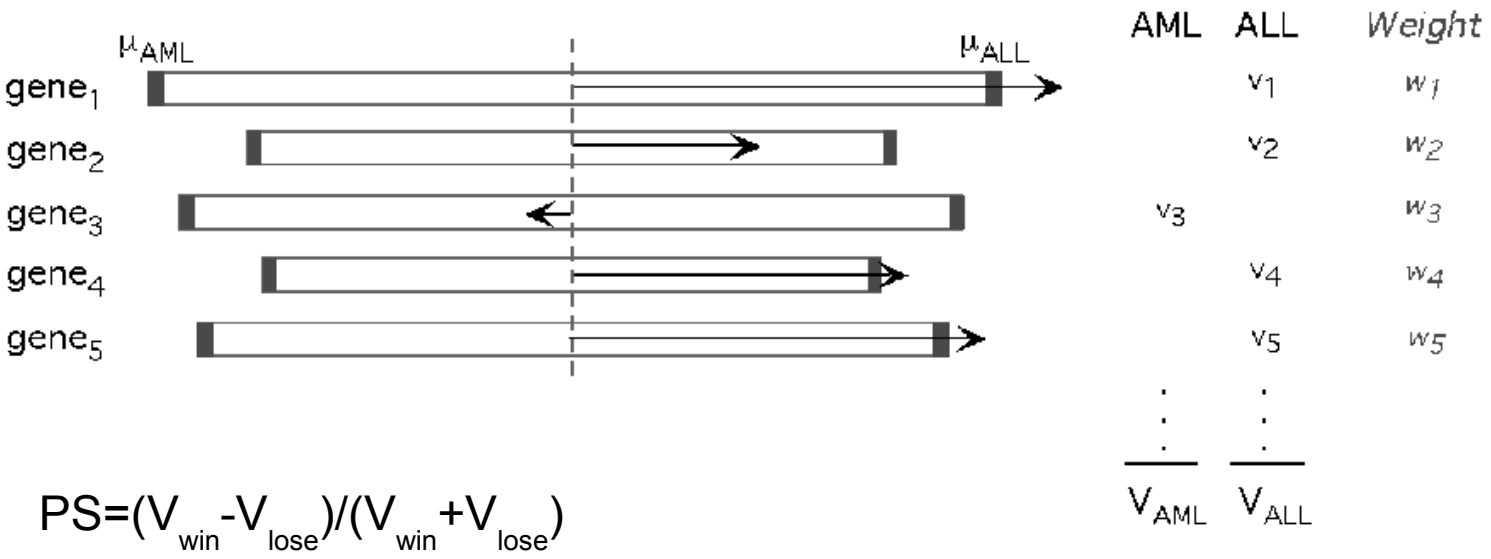
Neighborhood analysis showed that 1100 genes were more highly correlated with the AML-ALL class distinction than would be expected by chance

Selecting Informative Genes

- Large values of |P(g,c)| indicate strong correlation
- Select 50 significantly correlated, 25 most positive and 25 most negative ones
- Selecting the top 50 could be possibly bad:
  - If AML gene are more highly expressed than ALL
  - Unequal number of informative genes for each class

2. Class Prediction

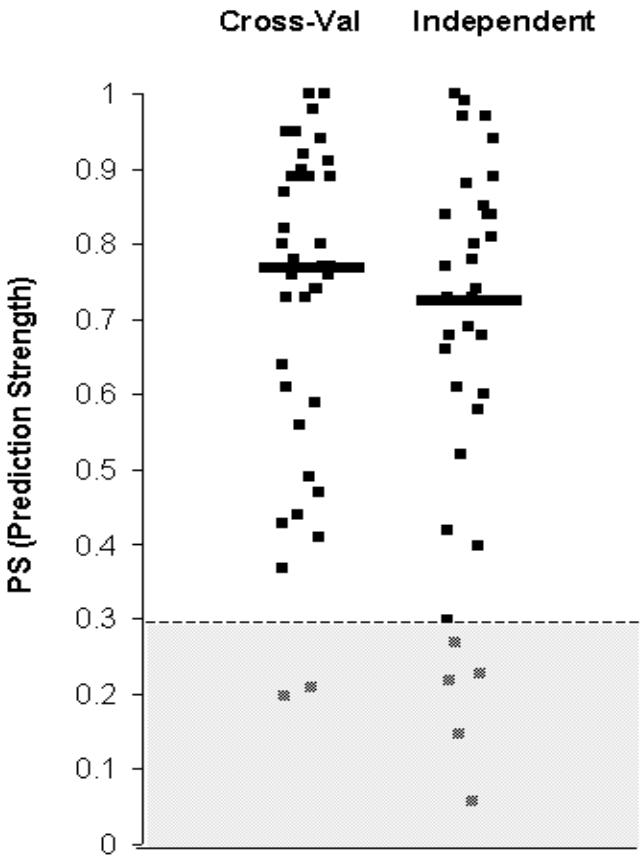
- Given a sample, classify it in AML or ALL
- Method:
  - Each of the fixed set of informative genes makes a prediction
  - The vote is based on the expression level of these genes in the new sample, and the degree of correlation with c
  - Votes are summed up to determine
    - The winning class and
    - The prediction strength (ps)



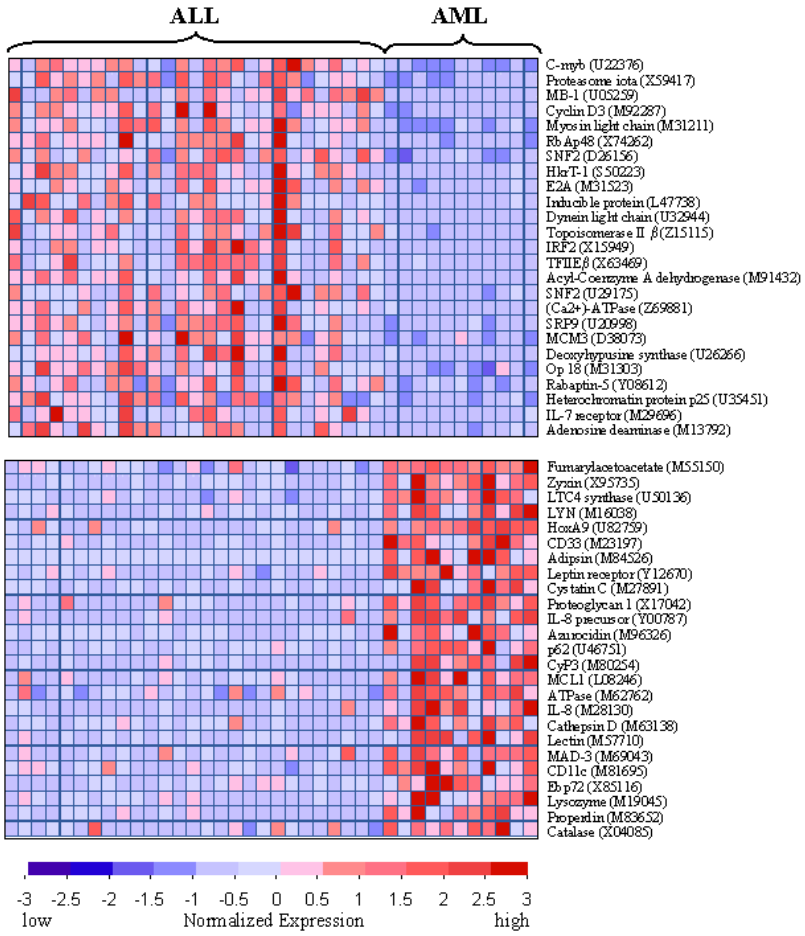
$$PS=(V_{win}-V_{lose})/(V_{win}+V_{lose})$$

Validity of Class Predictions

- Leave-one-out Cross Validation with the initial data
- Validation on an independent data set (test)



List of Informative Genes



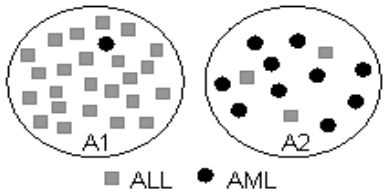
3. Class Discovery

- What if the AML-ALL class distinction was not known before hand? Could we discover it automatically?

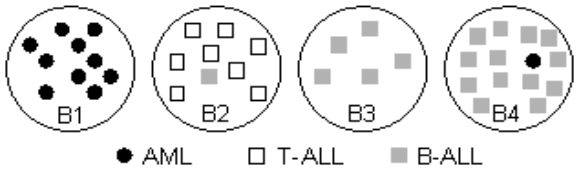
- Golub et al used a SOM clustering to discover two classes, and finer subclasses

Finer Classes

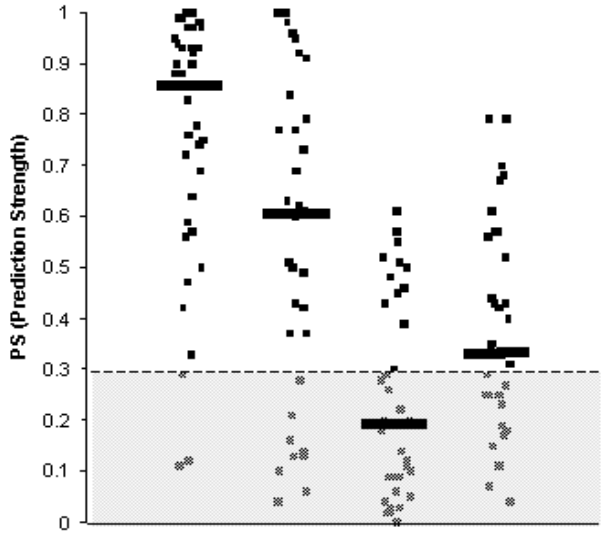
A



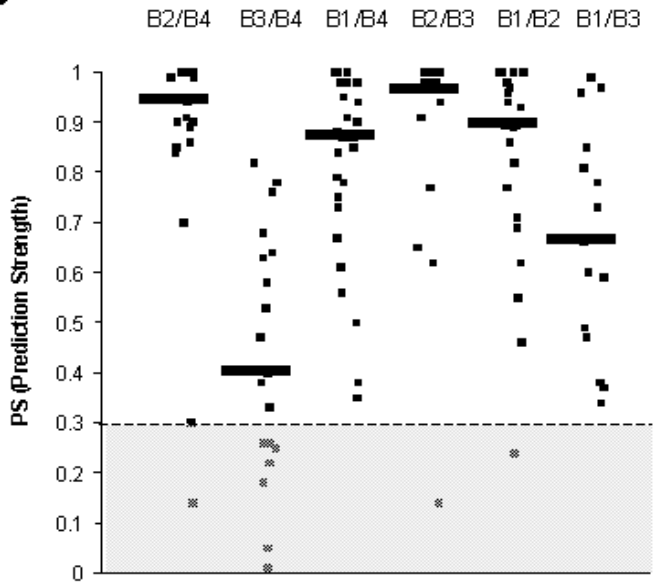
C



B



D



Conclusions

- Linear nearest-neighbor discriminators are quick, and identify strong informative signals well
- Easy and good biological validation

But

- Only gross differences in expression are found. Subtler differences cannot be detected
- The most informative genes may not be also biologically most informative. It is almost always possible to find genes that split samples into two classes

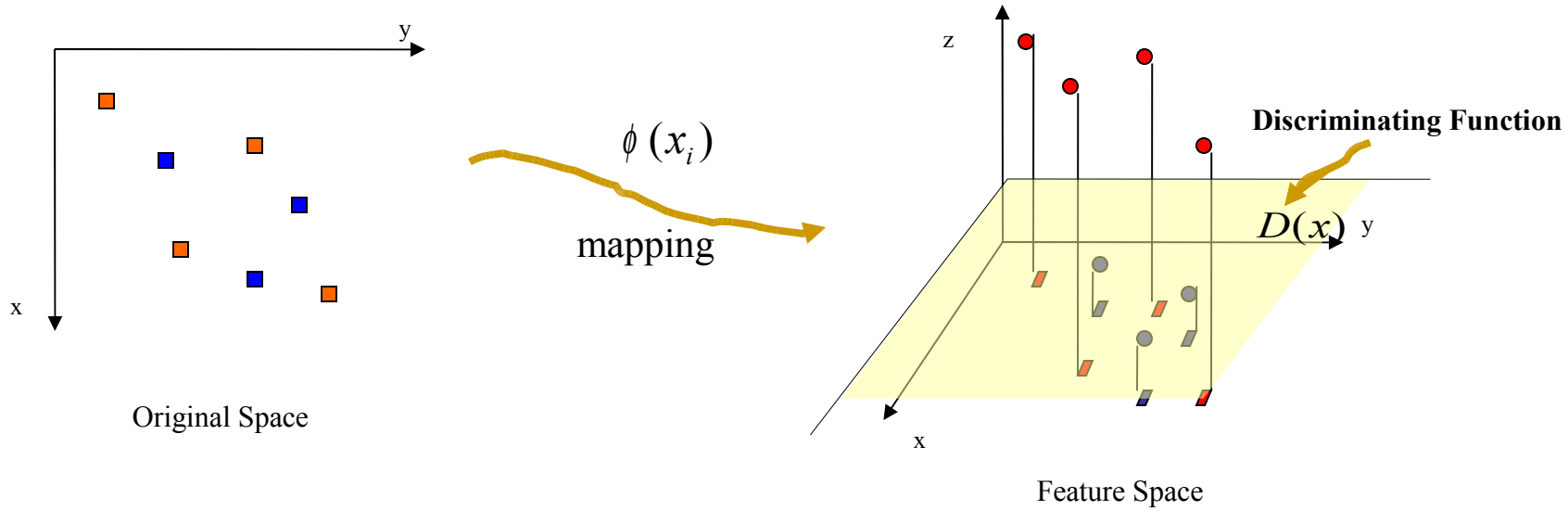
Support Vector Machines

- Inventor: V. N. Vapnik, late seventies
- Area of Origin: Theory of Statistical Learning
- In short: AI + Statistics
- Have shown promissing results in many areas:
  - OCR
  - Object recognition
  - Voice recognition
  - Biological sequence data analysis

Kernel Methods Basics

KM can be used as classifiers for data classes with complex discrimination boundaries

Kernel Functions map the data to higher dimensions where the discrimination boundary is simpler



Linear Learning Machines

Binary classification problem

- Given: n training pairs, $(\langle x_i \rangle, y_i)$, where $\langle x_i \rangle = (x_{i1}, x_{i2}, \ldots, x_{ik})$ is an input vector, and $y_i = +1/-1$, is the corresponding classification into two classes H_+ and H_-
- Out: A label y for a new vector x, as a function of the training pairs

$$y = D(x, ((\langle x_1 \rangle, y_1), (\langle x_2 \rangle, y_2), \ldots, (\langle x_n \rangle, y_n)))$$

Linear Discriminator Function

The classification of new examples, *x*, is based on all the previous ones, weighted by:

- λ_i, measuring the importance of example *i*, and
- The kernel $K(x_i,x)$, measuring the similarity of new example *x* to the training x_i

$$y = D(x) = \sum_i y_i \lambda_i K(x_i, x)$$

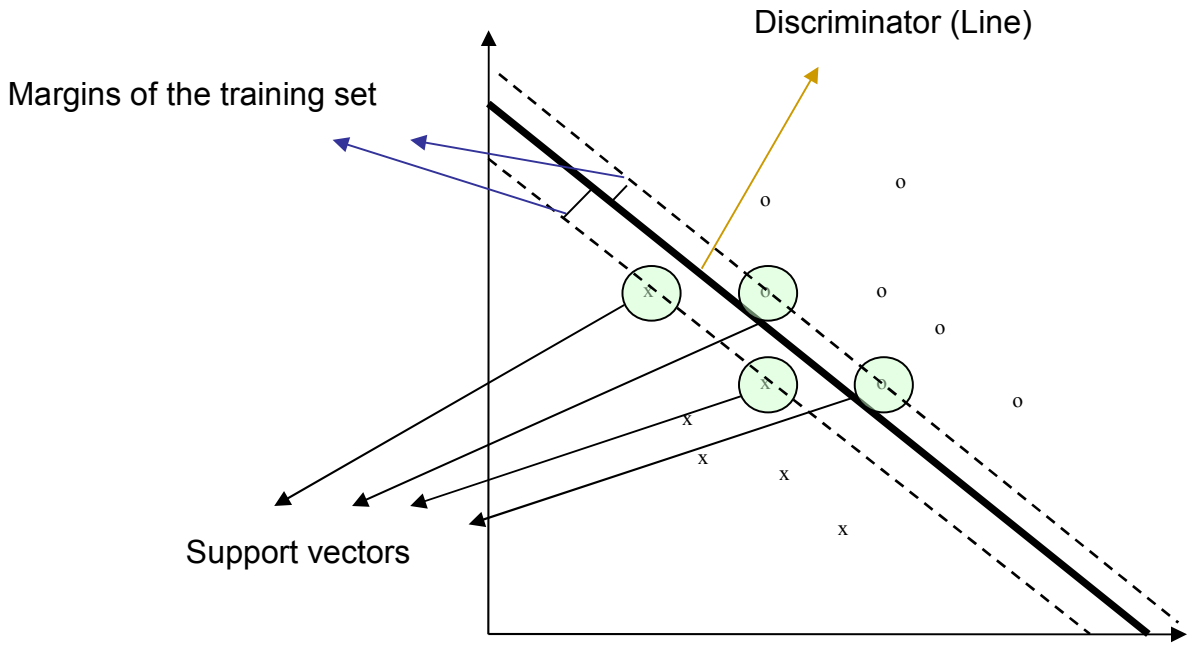
Linear Classification

- Learn the class labels, y_i, on the training set
  - The Perceptron algorithm
  - Optimization: 0,1 Integer program
  - Many possible consistent classifiers
- Classify a new example, x, based on which side of the classifier line it is on

$$y = D(x, ((< x_i >, y_i), ...,)) = << y > \cdot x > + b$$

$$= \sum_{i=1}^{n} y_i x_i + b$$

Discriminators and Support Vectors



Goal: To find good discriminators by maximizing the margins

Non-Linear Case

- Notice that the data during training appears only as a dot product
- Kernel functions, $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$
- Thus, the original data can be mapped, with a suitable mapping ϕ, to a space in which the discrimination task is easier
- All we need is such a decomposable Kernel function K

Possible Kernel Functions

Polynomial kernels : $(1+ x_i \cdot x_j)^m$

Radial Basis Kernel : $e^{-\frac{|xi-xj|^2}{2\sigma^2}}$

Neural Network Kernel : $\tanh(\mu x_i^t x_j + \kappa)$

Practical Considerations When Training the SVMs

- Computationally expensive to compute the Kernel function for each pair of elements
- Solution: Use only part of the data, preferably the part that contributes most to the decision boundary
- How do we do that? Heuristics

Using SVMs to Classify Genes Based on Microarray Expression

"Knowledge-based analysis of microarray gene expression data by using support vector machines", Brown et al., PNAS 2000

A method of functionally classifying genes based on DNA Microarray expression data based on the theory of SVMs.

Method

- A training data set
  - (1) genes that are known to have the same function, f, and
  - (2) genes that are known to have a different function than f
- Such a training set can be obtained from publicly available data sources
- Use the SVM machinery on the above and predict known and new examples, and compare to other classification methods

Data

- Yeast genes
- Training data
  - 2467 genes
  - 79 hybridiztion exp.
- Test Data
  - 6221 genes (including all above)
  - 80 hybridization exp. (65 from above + 15 others)
- Functional classifications
  - Five functional classes from MYGD

Kernels and Other Methods

- Kernels used
  - Polynomial, degrees 1, 2, and 3
  - Radial
- Compared to four other methods
  - Parzen windows
  - Fisher's linear discriminant
  - Two decision tree learners
- Tested false positives, false negatives, true positives, true negatives, and overall perf.

Results

-The SVMs outperform the other methods.

-Unannotated genes were predicted to be in functional classes

- Some functional classes cannot be predicted with SVMs possibly because they have little to do with gene expression

Table 1. Comparison of error rates for various classification methods

Class	Method	FP	FN	TP	TN	S(M)
TCA	D-p 1 SVM	18	5	12	2,432	6
	D-p 2 SVM	7	9	8	2,443	9
	D-p 3 SVM	4	9	8	2,446	12
	Radial SVM	5	9	8	2,445	11
	Parzen	4	12	5	2,446	6
	FLD	9	10	7	2,441	5
	C4.5	7	17	0	2,443	−7
	MOC1	3	16	1	2,446	−1
Resp	D-p 1 SVM	15	7	23	2,422	31
	D-p 2 SVM	7	7	23	2,430	39
	D-p 3 SVM	6	8	22	2,431	38
	Radial SVM	5	11	19	2,432	33
	Parzen	22	10	20	2,415	18
	FLD	10	10	20	2,427	30
	C4.5	18	17	13	2,419	8
	MOC1	12	26	4	2,425	−4
Ribo	D-p 1 SVM	14	2	119	2,332	224
	D-p 2 SVM	9	2	119	2,337	229
	D-p 3 SVM	7	3	118	2,339	229
	Radial SVM	6	5	116	2,340	226
	Parzen	6	8	113	2,340	220
	FLD	15	5	116	2,331	217
	C4.5	31	21	100	2,315	169
	MOC1	26	26	95	2,320	164
Prot	D-p 1 SVM	21	7	28	2,411	35
	D-p 2 SVM	6	8	27	2,426	48
	D-p 3 SVM	3	8	27	2,429	51
	Radial SVM	2	8	27	2,430	52
	Parzen	21	5	30	2,411	39
	FLD	7	12	23	2,425	39
	C4.5	17	10	25	2,415	33
	MOC1	10	17	18	2,422	26
Hist	D-p 1 SVM	0	2	9	2,456	18
	D-p 2 SVM	0	2	9	2,456	18
	D-p 3 SVM	0	2	9	2,456	18
	Radial SVM	0	2	9	2,456	18
	Parzen	2	3	8	2,454	14
	FLD	0	3	8	2,456	16
	C4.5	2	2	9	2,454	16
	MOC1	2	5	6	2,454	10
HTH	D-p 1 SVM	60	14	2	2,391	−56
	D-p 2 SVM	3	16	0	2,448	−3
	D-p 3 SVM	1	16	0	2,450	−1
	Radial SVM	0	16	0	2,451	0
	Parzen	14	16	0	2,437	−14
	FLD	14	16	0	2,437	−14
	C4.5	2	16	0	2,449	−2
	MOC1	6	16	0	2,445	−6

References and Further Reading

- Golub et al., Molecular Classification of Cancer, Science, 1999
- Brown et al., PNAS, 2000
- Kohane, Kho, Butte. Microarrays for an Integrative Genomics, MIT Press, 2003
- Baldi, Hatfield. DNA Microarrays and Gene Expression, Cambridge, 2002
- Cristianini and Shawe-Taylor, An Introduction to Support Vector Machines, Cambridge, 2000
- Shamir, Analysis of Gene Expression Data, Tel Aviv University, 2002, Lecture 7. *http://www.math.tau.ac.il/~rshamir/ge/02/scribes/lec07.pdf*