6.034 Introduction to Artificial Intelligence

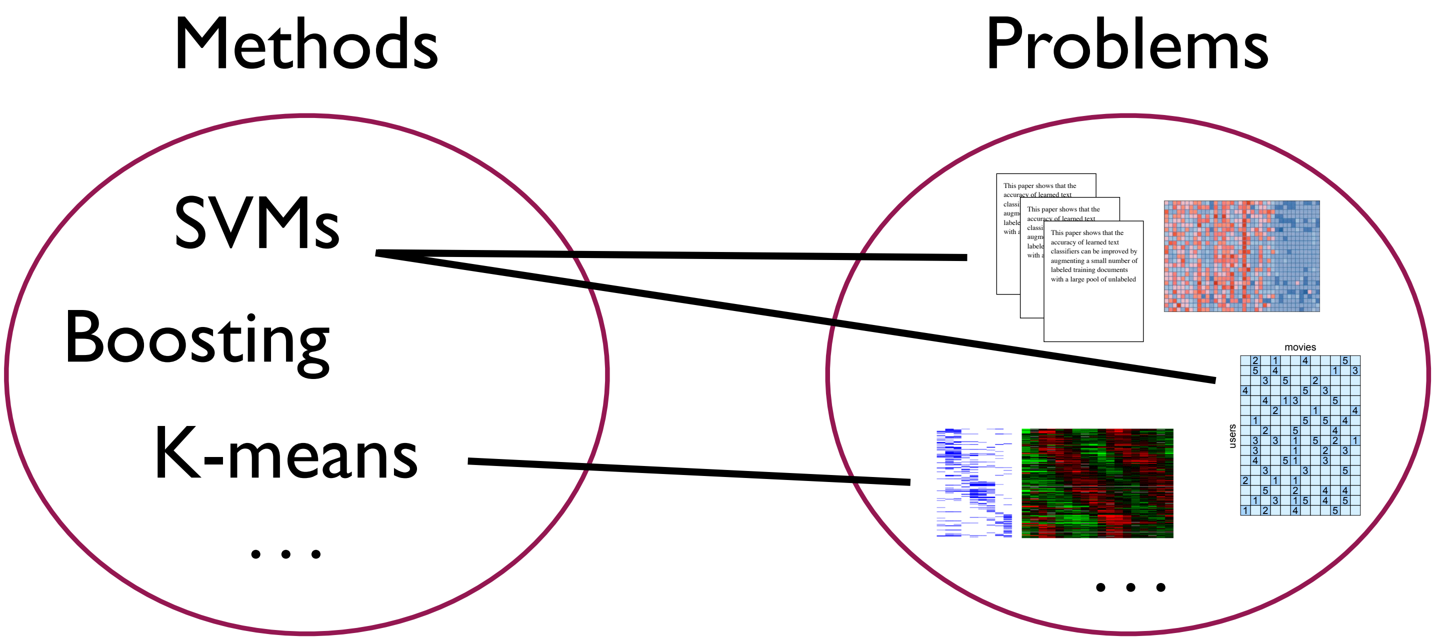
Machine learning and applications

Problems we will cover

- Computational biology
  - cancer classification
  - functional classification of genes
- Information retrieval
  - document classification/ranking
- Recommender systems
  - predicting user preferences (e.g., movies)

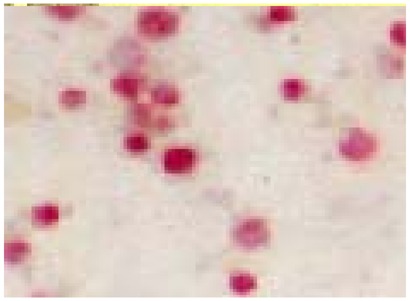
What are we trying to do?

- The goal is to find the right method for the right problem (matching task)



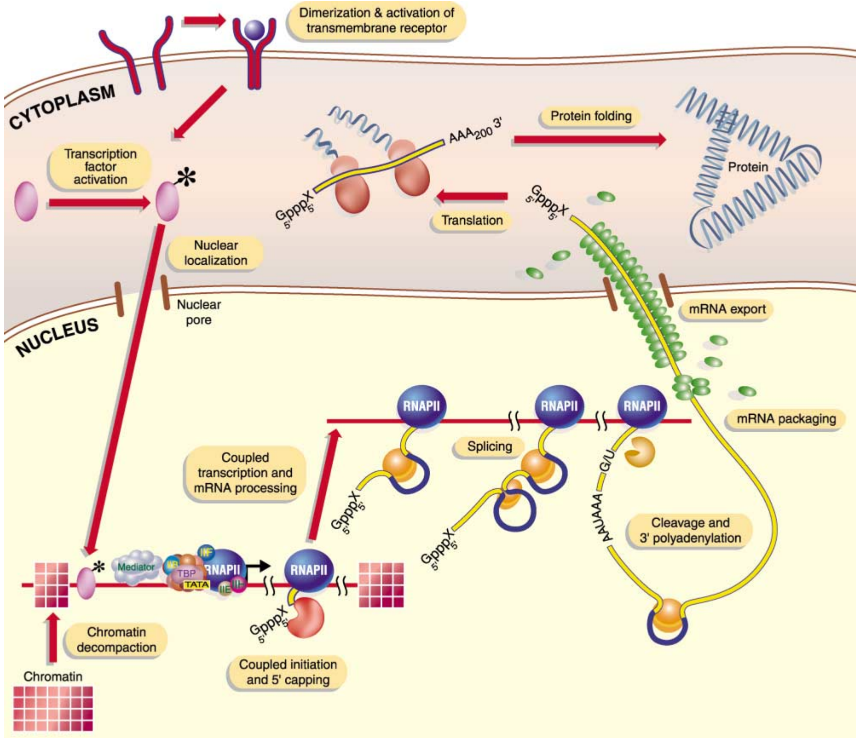
Cancer classification

- We'd like to automatically classify tissue samples according to whether there's evidence of cancer or the type of tumor cells they contain



- What features to extract?
  - visual features due to different types of staining
  - how active different genes are in the cells (gene expression)

Gene expression

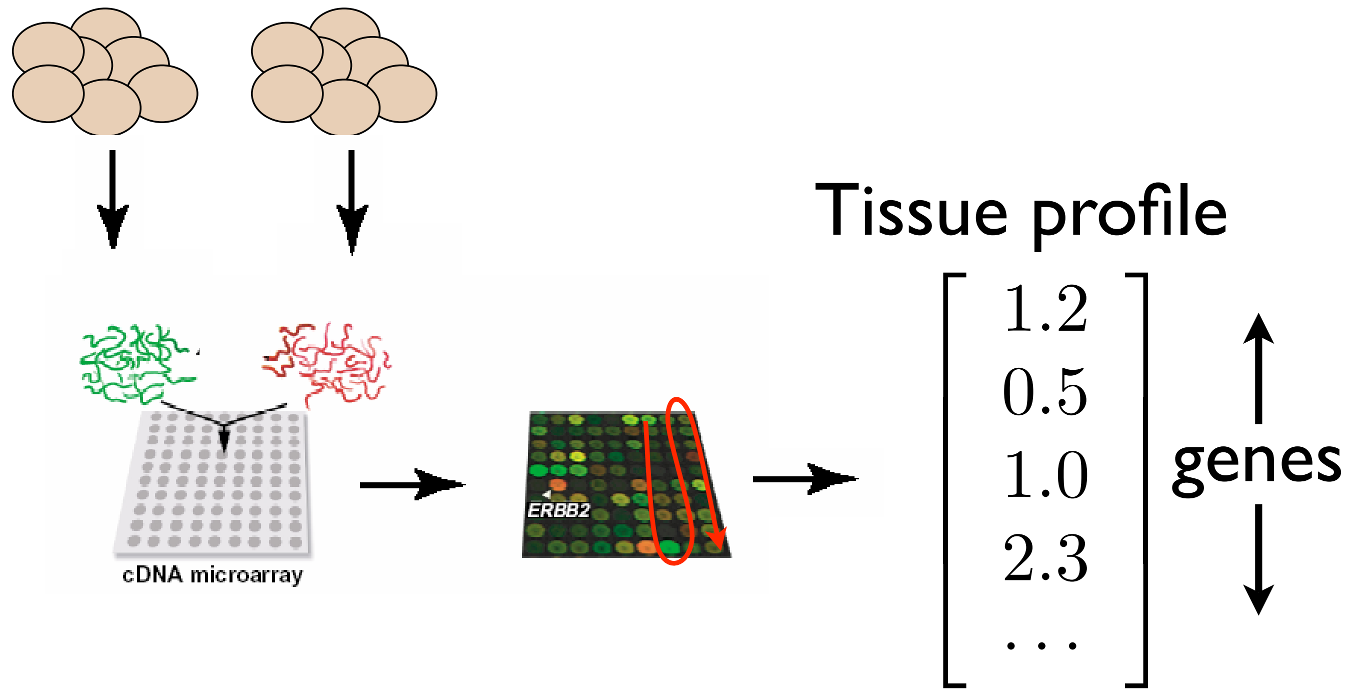


(Orphanides et al. 2002)

Measuring gene expression

- Basic cDNA micro-array technology

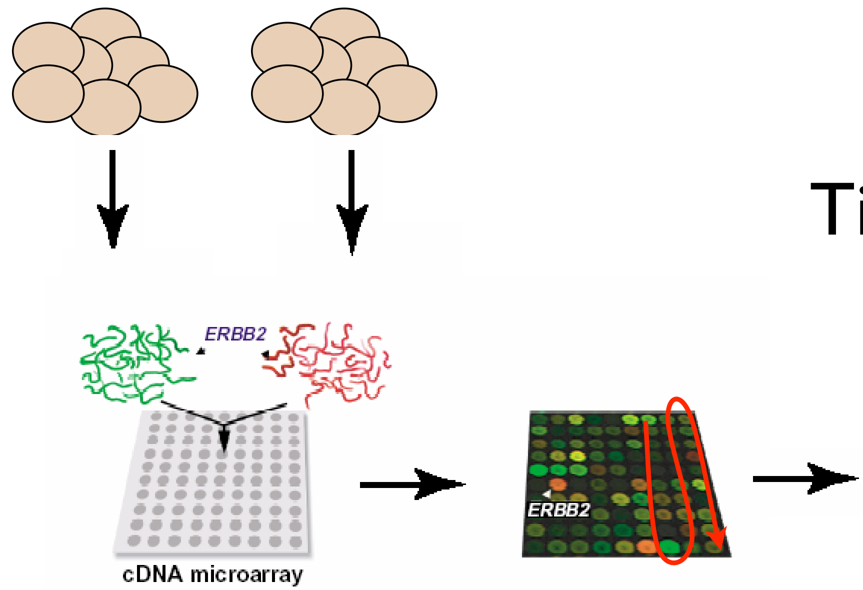
control sample (e.g., tumor)



Measuring gene expression

- Basic cDNA micro-array technology

control sample (e.g., tumor)



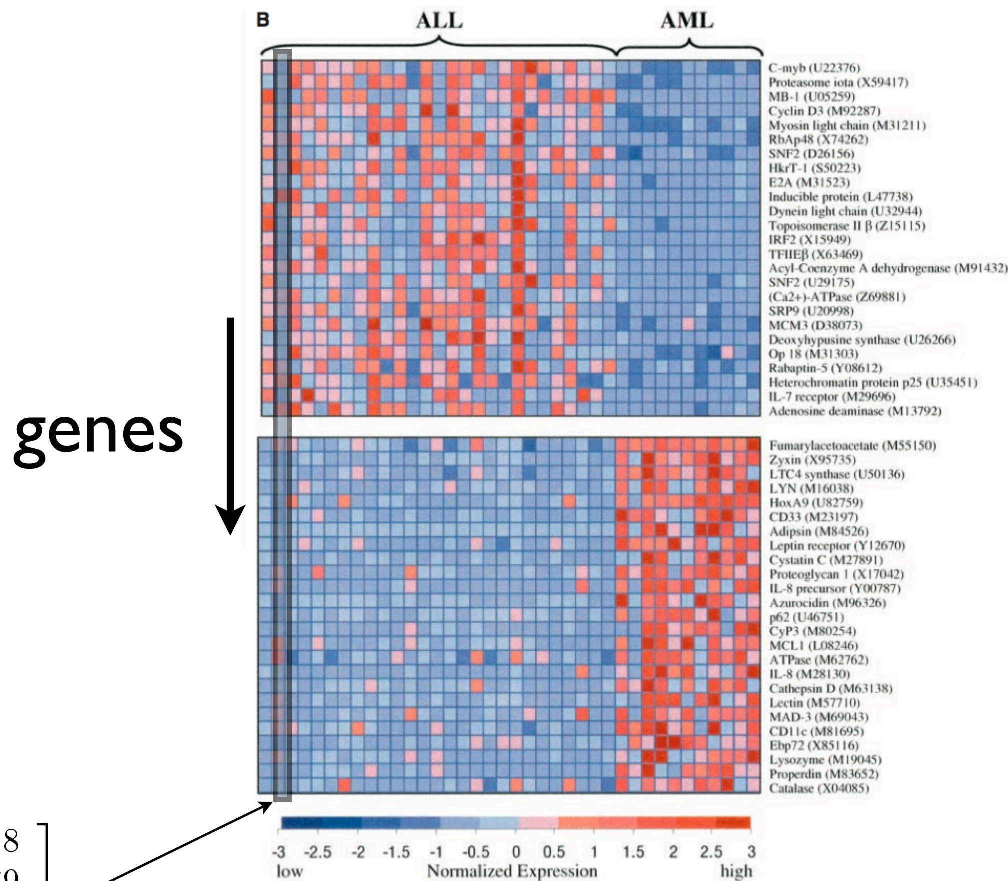
Tissue profile

$$\begin{bmatrix} 0.18 \\ -0.69 \\ 0.00 \\ 0.83 \\ \cdots \end{bmatrix}$$

genes

Cancer classification

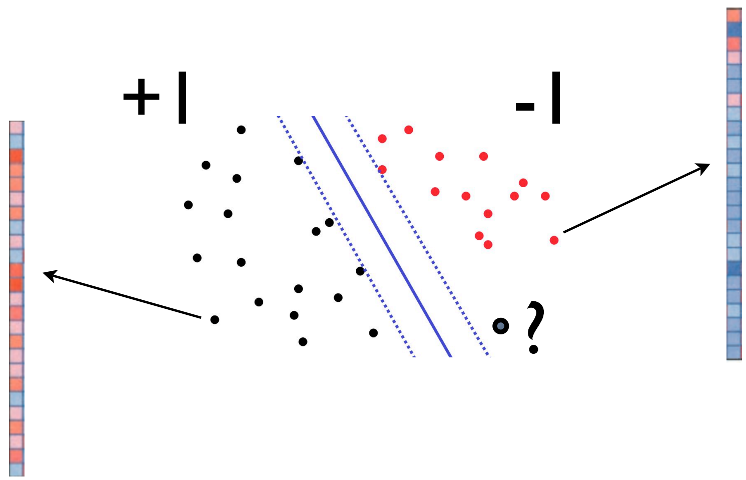
tissues (with known tumor type)



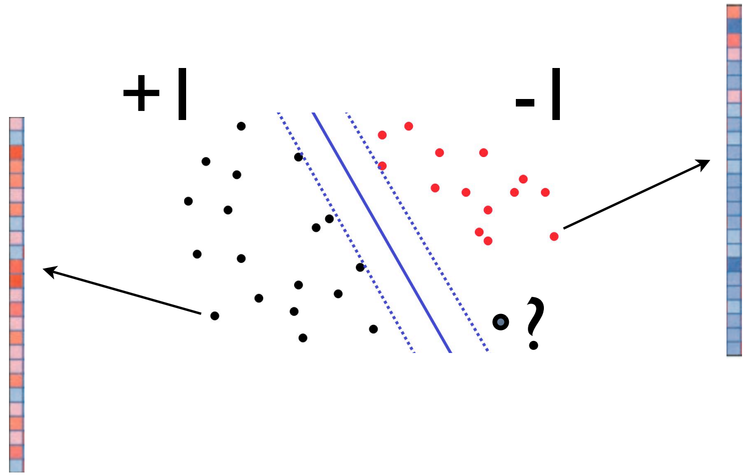
$$\begin{bmatrix} 0.18 \\ -0.69 \\ 0.00 \\ 0.83 \\ \dots \end{bmatrix}$$

(Golub et al. 1999)

Machine learning problem

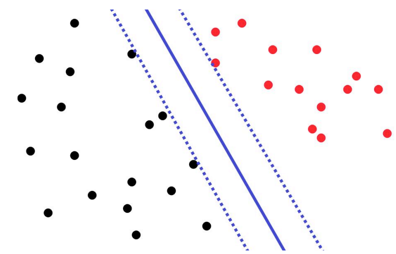


Machine learning problem



- Complicating issues
  - micro-array measurements are very noisy
  - each training example is of very high dimension (e.g., ~ 10,000 genes)
  - there are relatively few labeled tissue samples (only tens per class)
  - some labels may be wrong

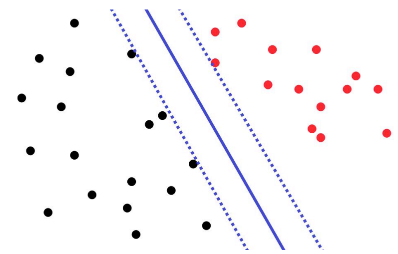
SVM classifiers



$$\hat{y} = \text{sign}\Big(\sum_{i=1}^{n} y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) + w_0 \Big)$$

- $\hat{y}$: Predicted label
- y_i: training label
- α_i: example weight
- w_0: offset
- K: kernel (similarity)
- $\mathbf{x}_i$: training example
- $\mathbf{x}$: new example

SVM training



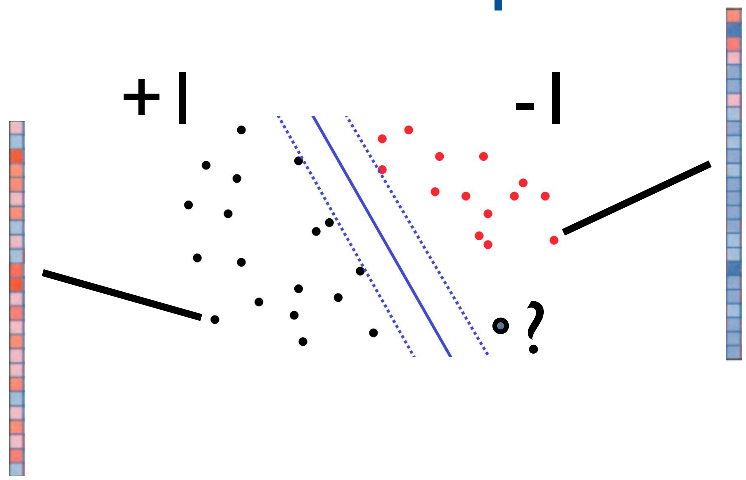
- SVMs are trained by solving a quadratic programming problem

$$\text{minimize} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} y_i y_j \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j)$$

$$\text{subject to} \quad \alpha_i \geq 0, \quad \sum_{i=1}^{n} y_i \alpha_i = 0$$

(where is w_0?)

Back to the problem



- High dimensionality => linear kernel

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j + 1)$$

- Noise in the measurements => feature selection (use only a relevant subset of the genes)
- Outliers => adjust the kernel to increase resistance to outliers

Feature selection / ranking

- We can rank genes according to how much they seem to be related to the classification task

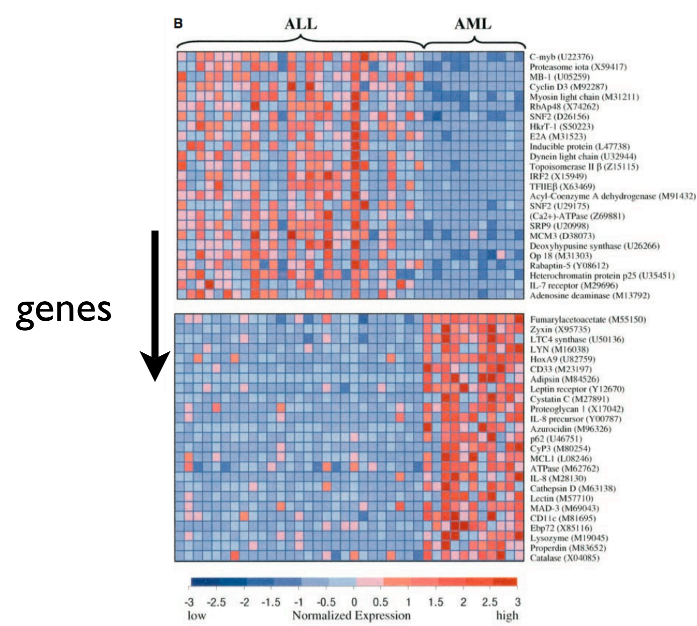
mean value across +1 tissues

mean value across -1 tissues

$$R(\text{gene}_i) = \frac{|\mu_i^+ - \mu_i^-|}{\sigma^+ + \sigma^-}$$

stdv across +1 tissues

stdv across -1 tissues

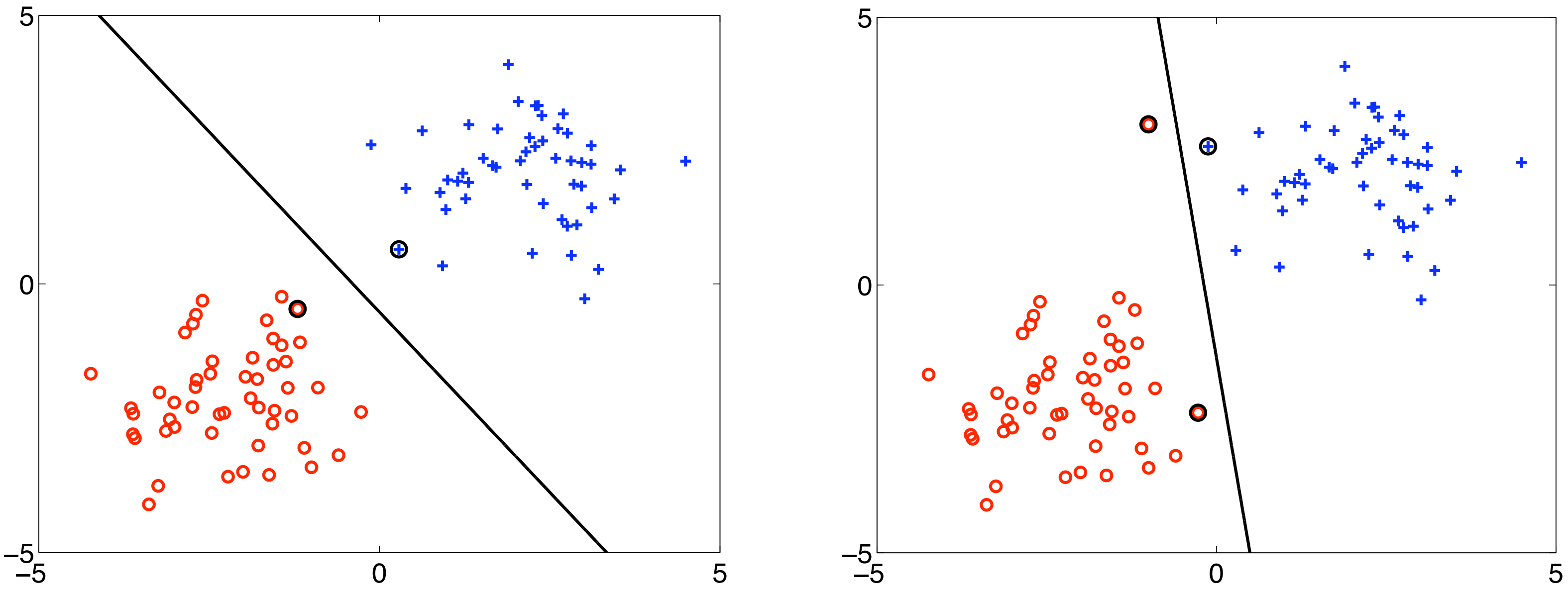


of examples, dimensionality

- Suppose the expression levels of all the 10,000 genes in each tissue sample are drawn at random from some distribution (e.g., normal)
- Based on 5 such expression vectors for each class, can we find a gene that is perfectly correlated with the labels?
- The chance of this happening is 100%
- What if we have had instead 10 such vectors per class? The probability drops to 1%

Dealing with outliers

- We should make the linear decision boundary resistant to outliers (e.g., due to mislabeled samples)

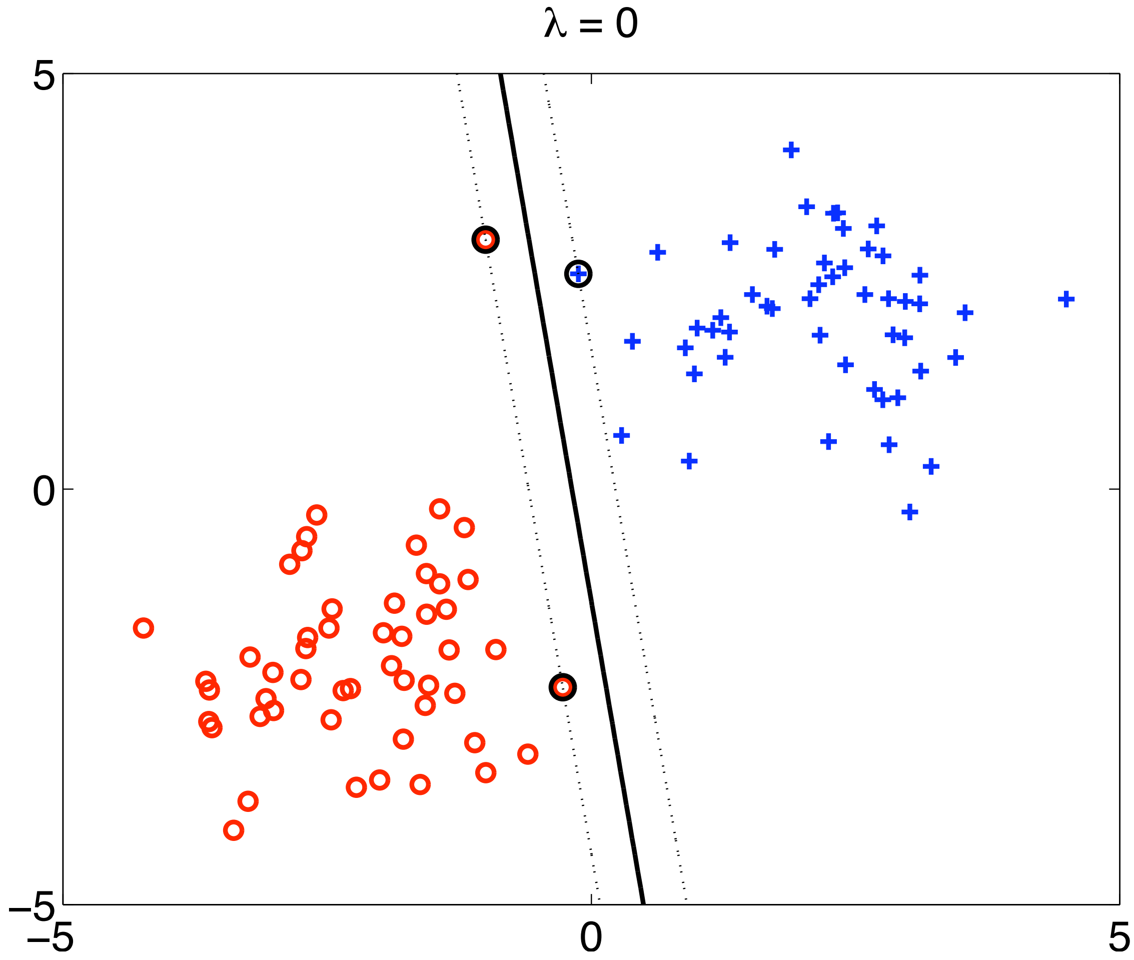


Dealing with outliers

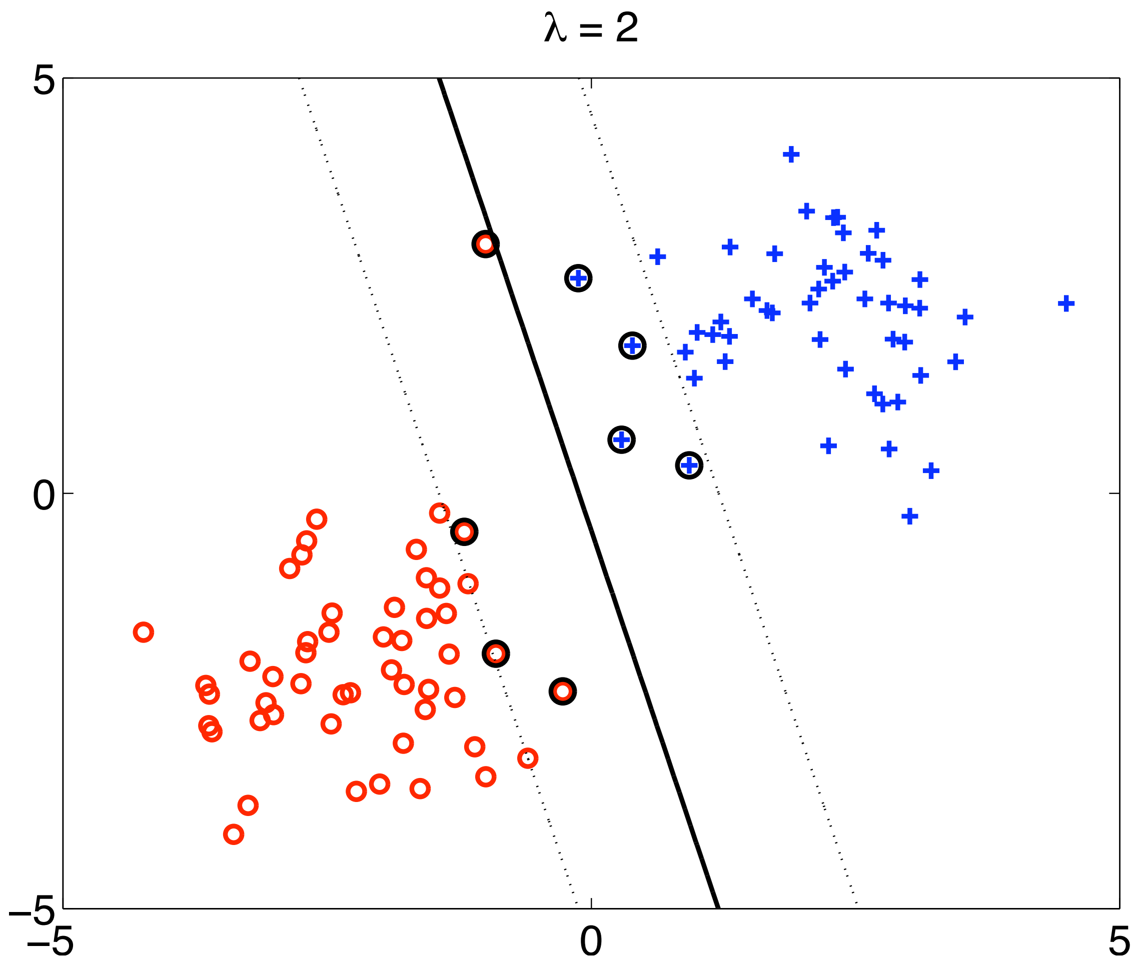
- One way to increase resistance to outliers is to add a diagonal term to the kernel function so that each example appears more similar to itself than before.

$$K \leftarrow \left[\begin{array}{ccc} K(x_1, x_1) + \lambda & \cdots & K(x_1, x_n) \\ \cdots & \cdots & \cdots \\ K(x_n, x_1) & \cdots & K(x_n, x_n) + \lambda \end{array} \right]$$

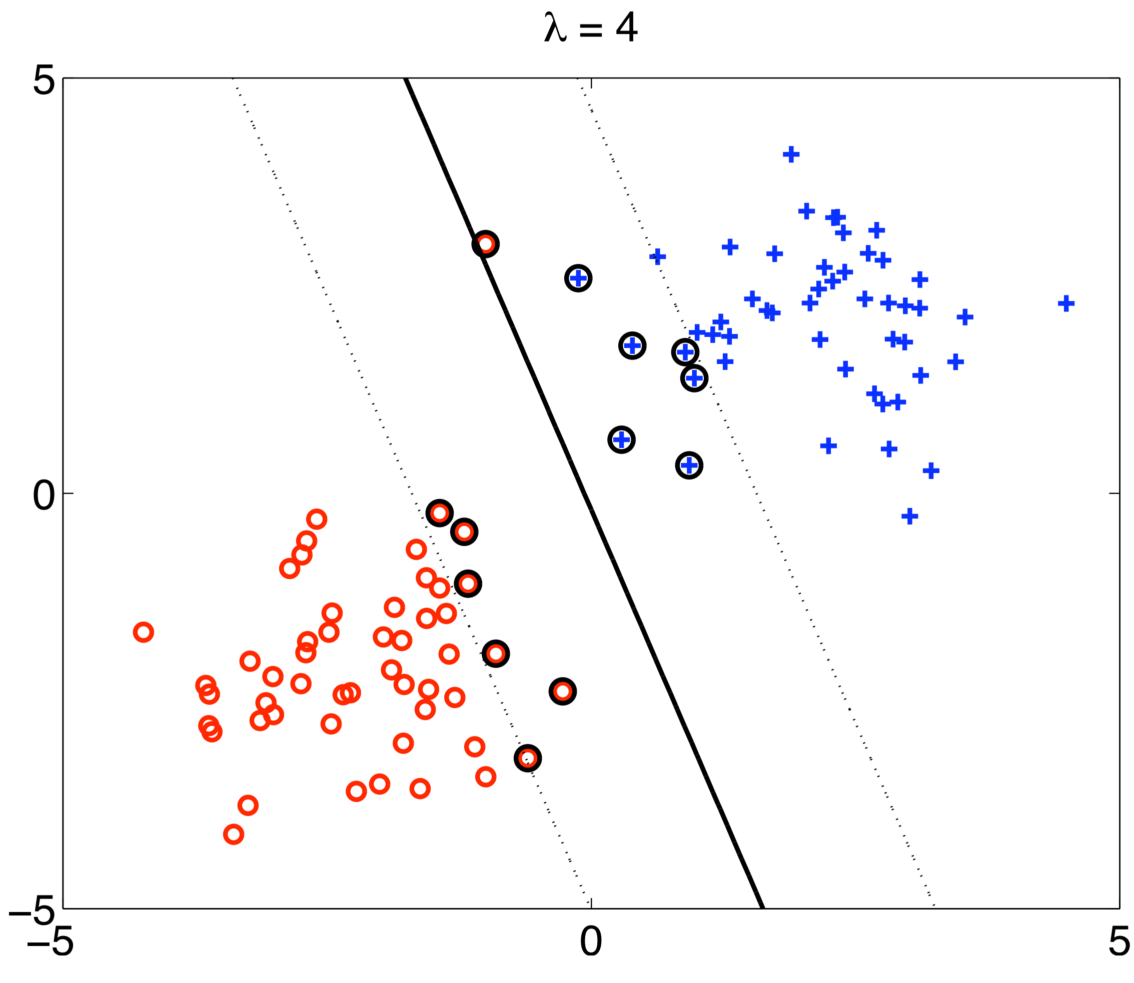
The effect of lambda



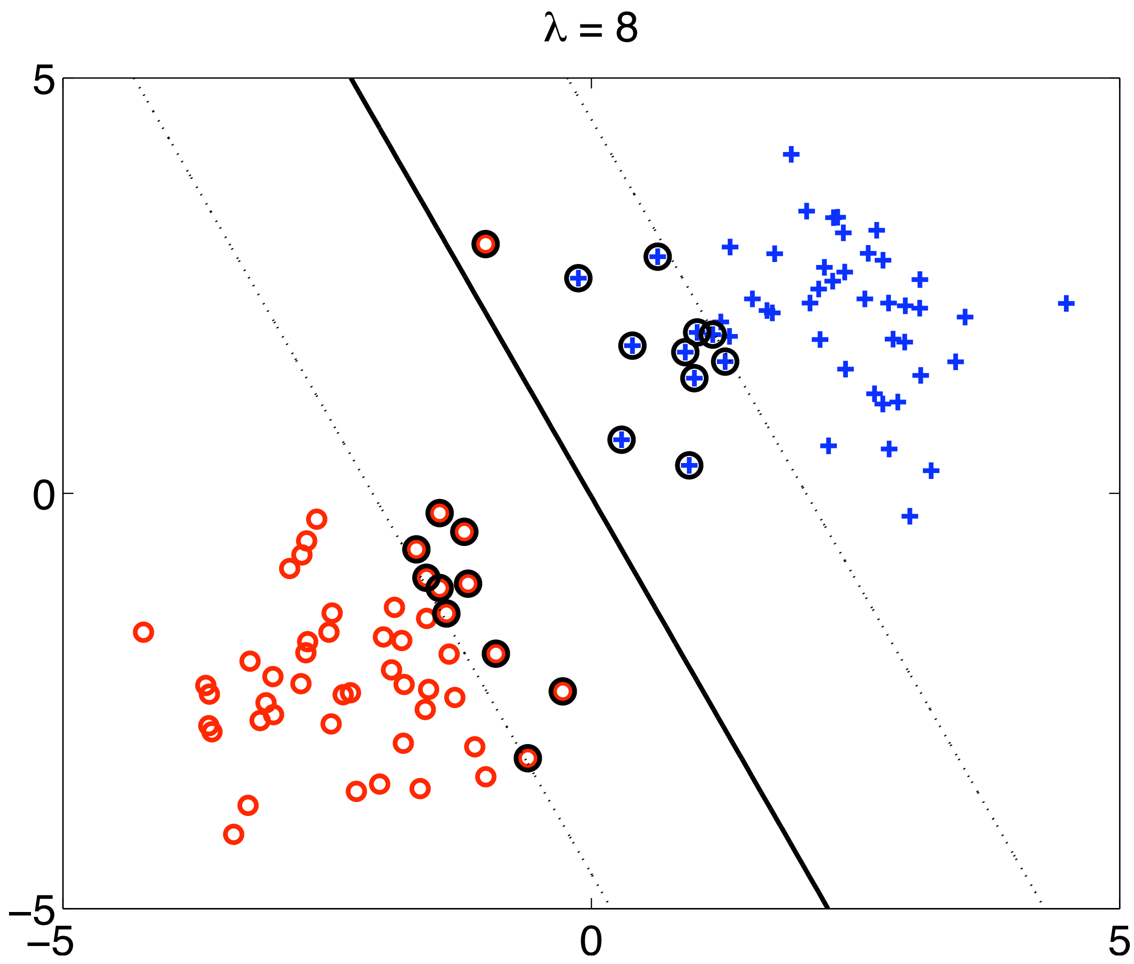
The effect of lambda



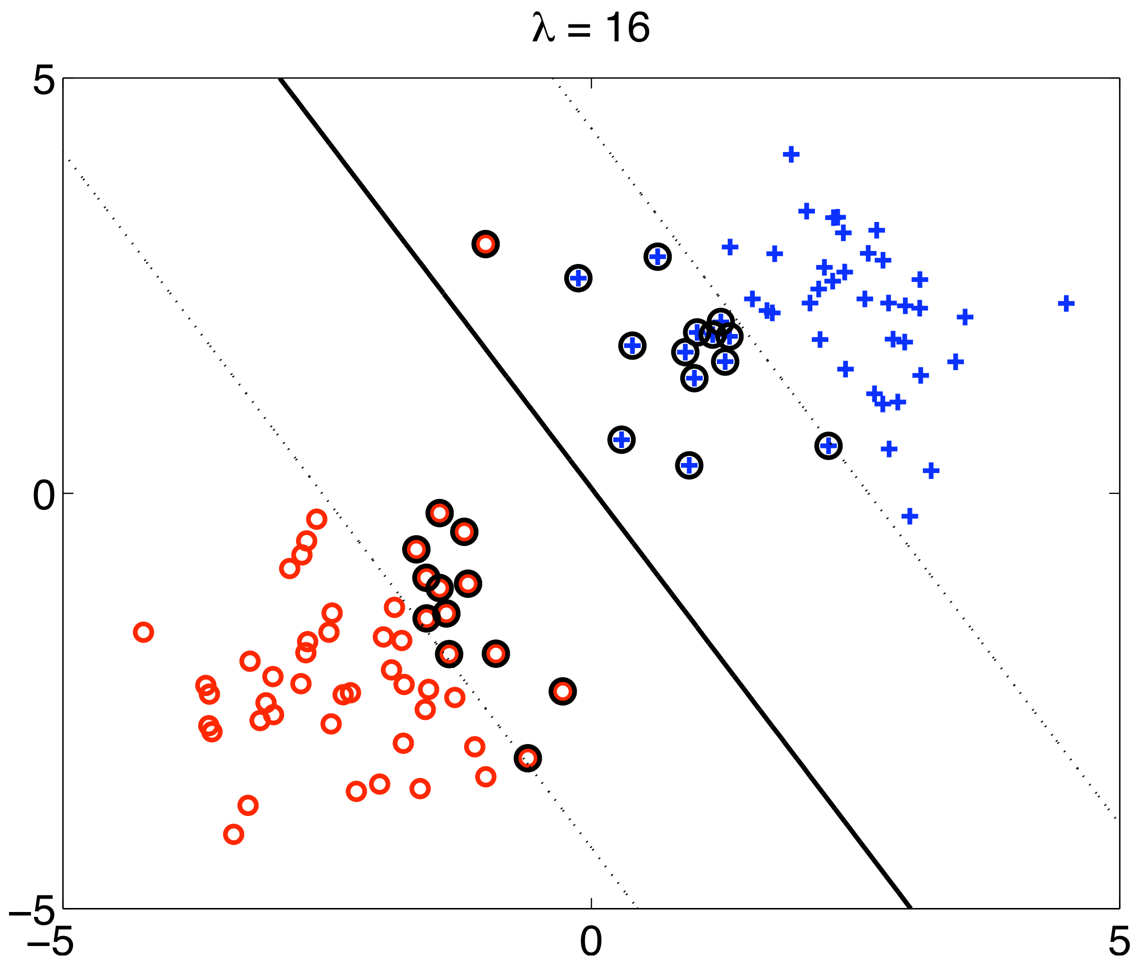
The effect of lambda



The effect of lambda

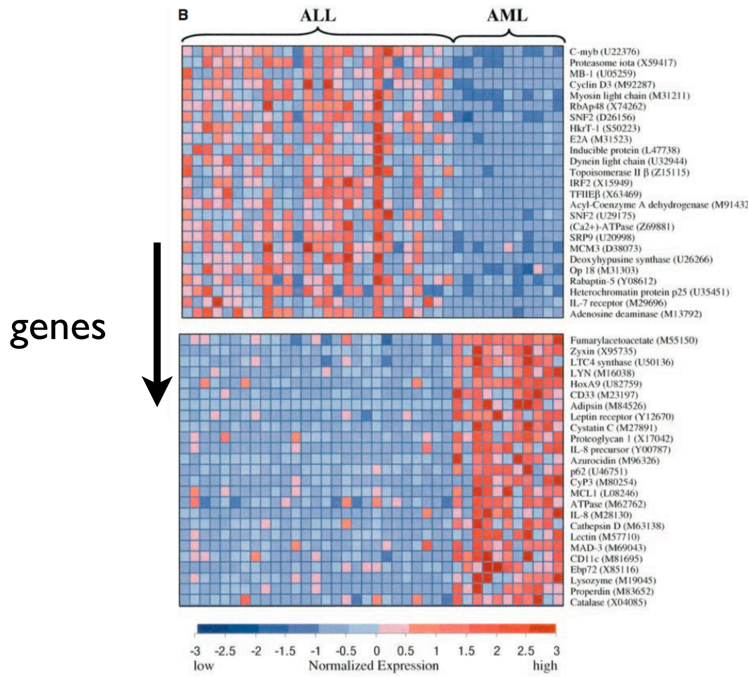


The effect of lambda



Results

- AML vs MML distinction
  - training set: 27 ALL and 11 AML
  - test set: 20 ALL and 14 ALM



- The SVM classifier achieves perfect classification of the test samples

Problems we will cover

- Computational biology
  - cancer classification
  - functional classification of genes
- Information retrieval
  - document classification/ranking
- Recommender systems
  - predicting user preferences (e.g., movies)

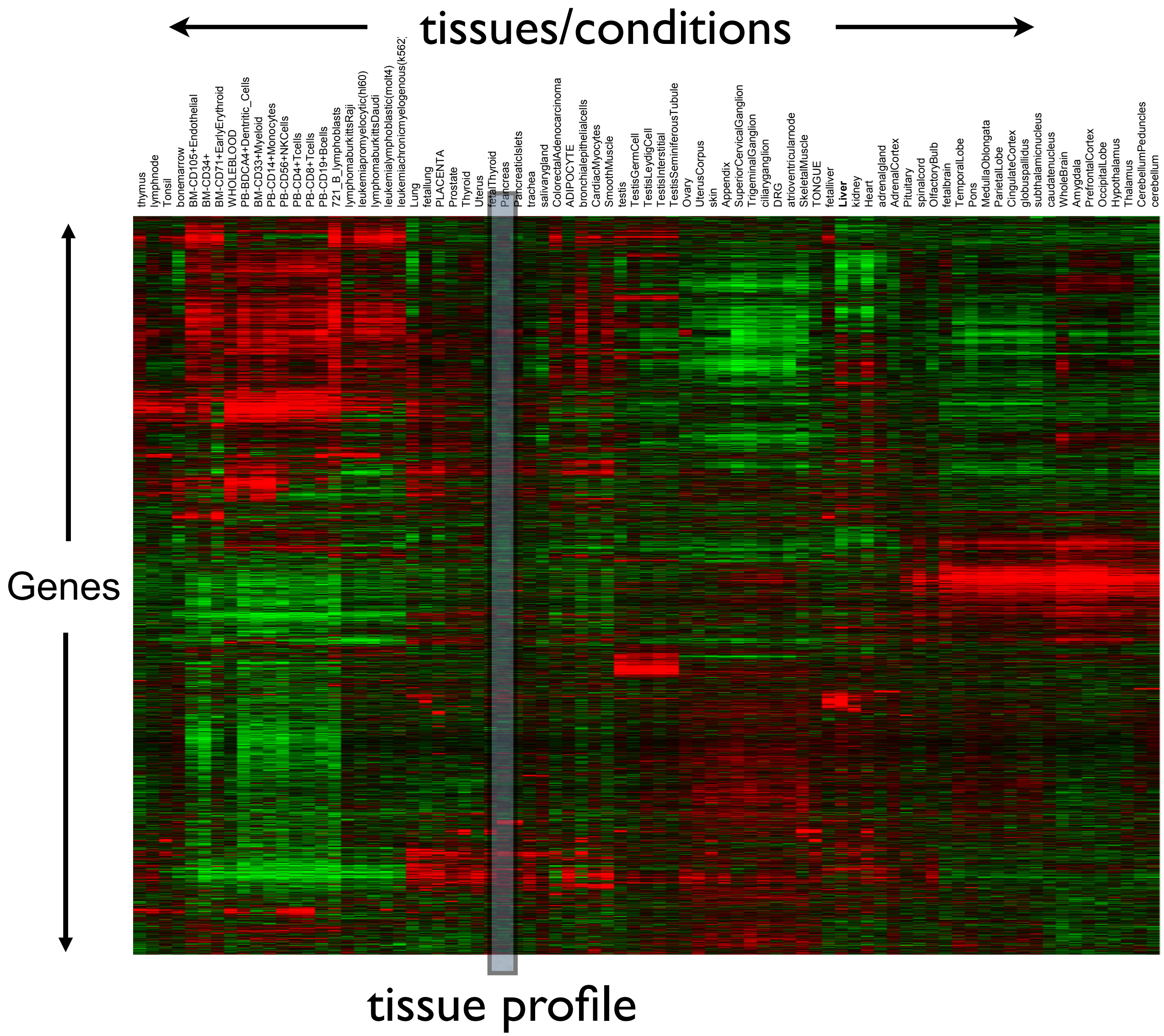
Functional classification of genes

- We don't know what most genes do
- Given known roles for some genes, we would like to predict the function of all the remaining genes

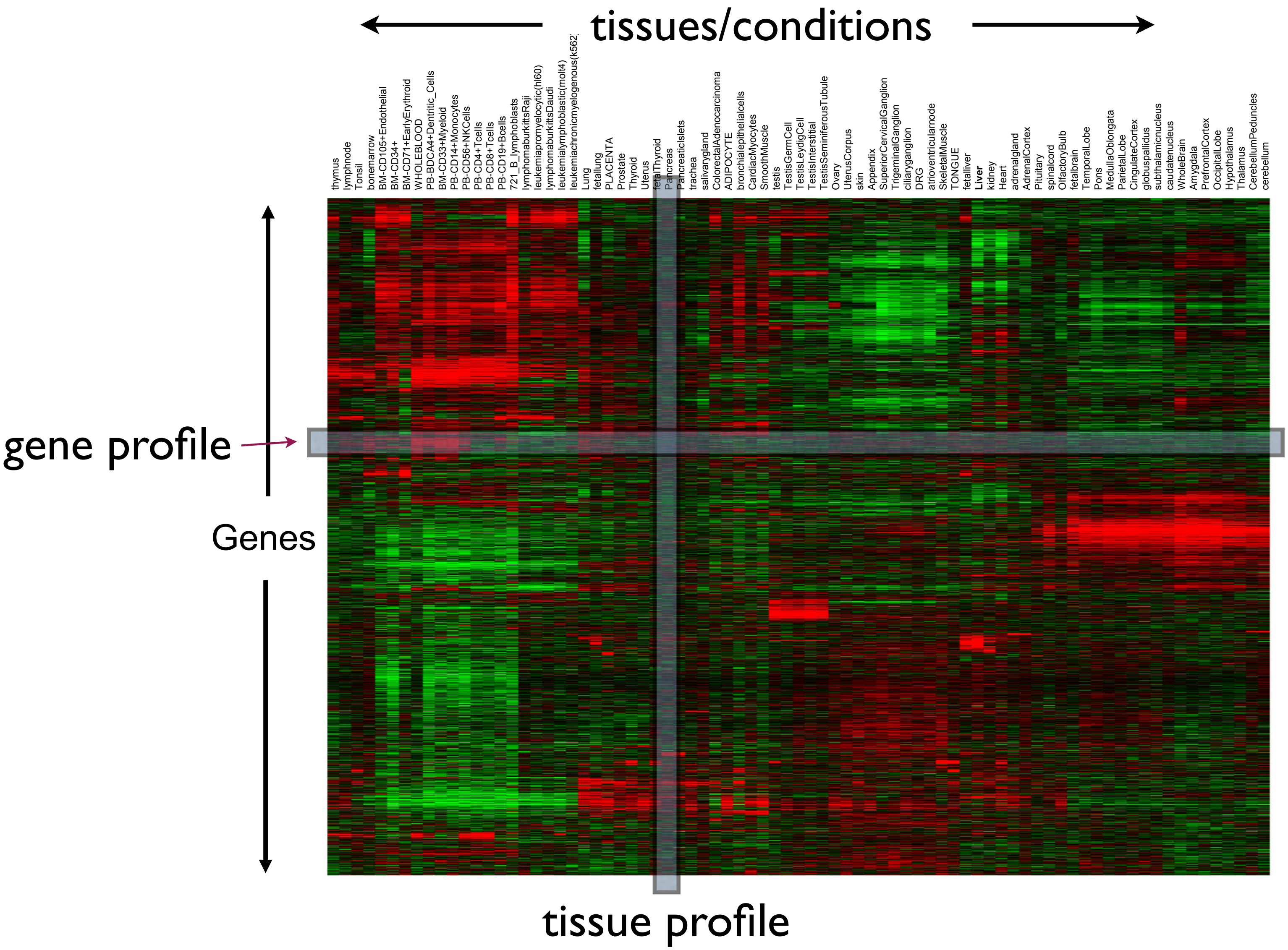
$$\text{ribosomal genes} \left\{ \begin{array}{l} F2N1.3 \\ T18A10.9 \\ F5J6.12 \\ \ldots \end{array} \right.$$

$$\text{unannotated ``genes''} \left\{ \begin{array}{l} \text{YLA003W} \\ \text{YPL037C} \\ \ldots \end{array} \right.$$

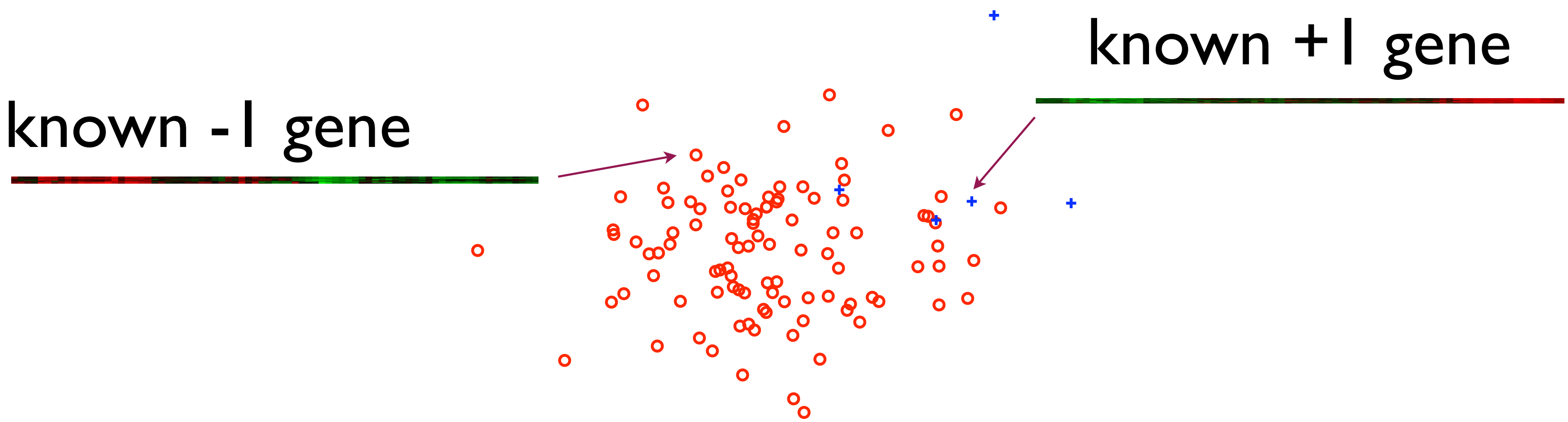
Tissue/gene profiles



Tissue/gene profiles

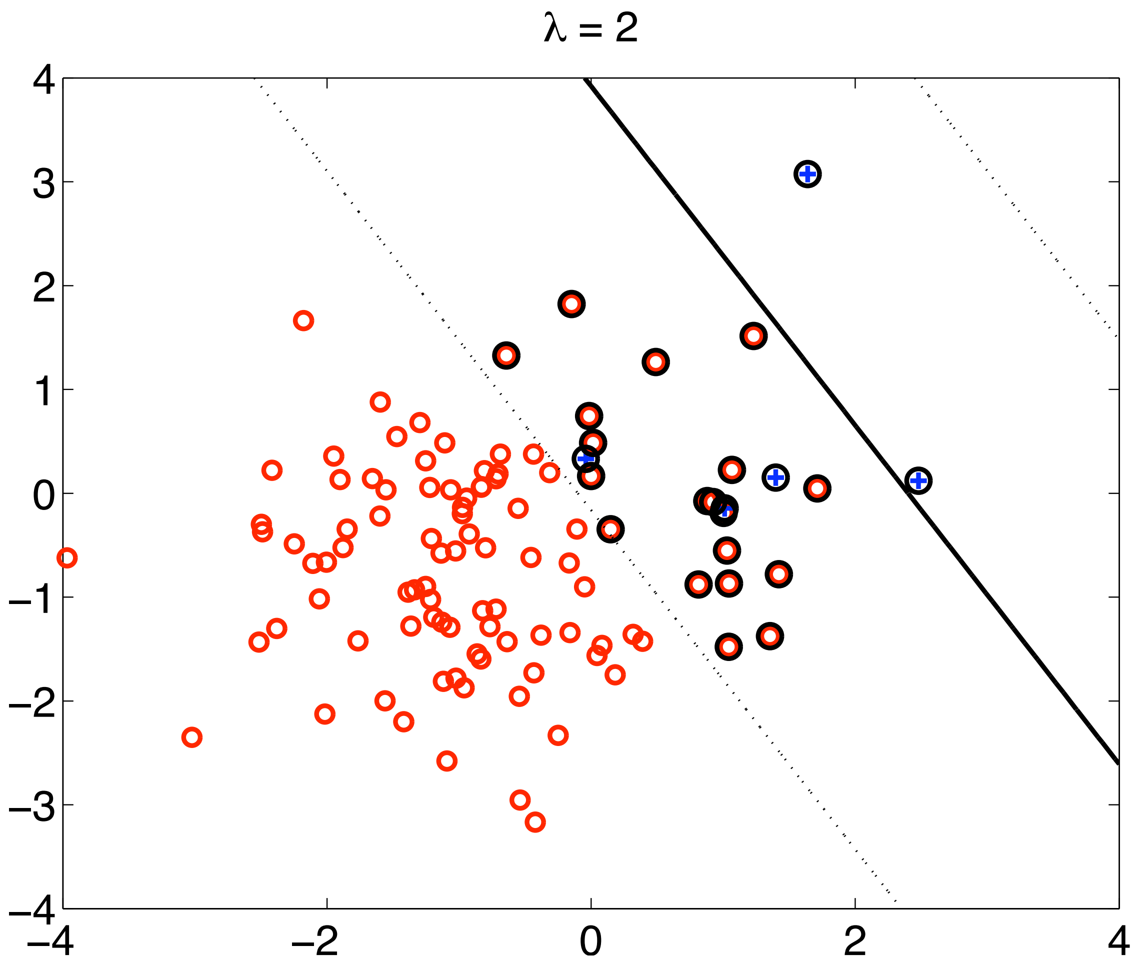


Machine learning problem

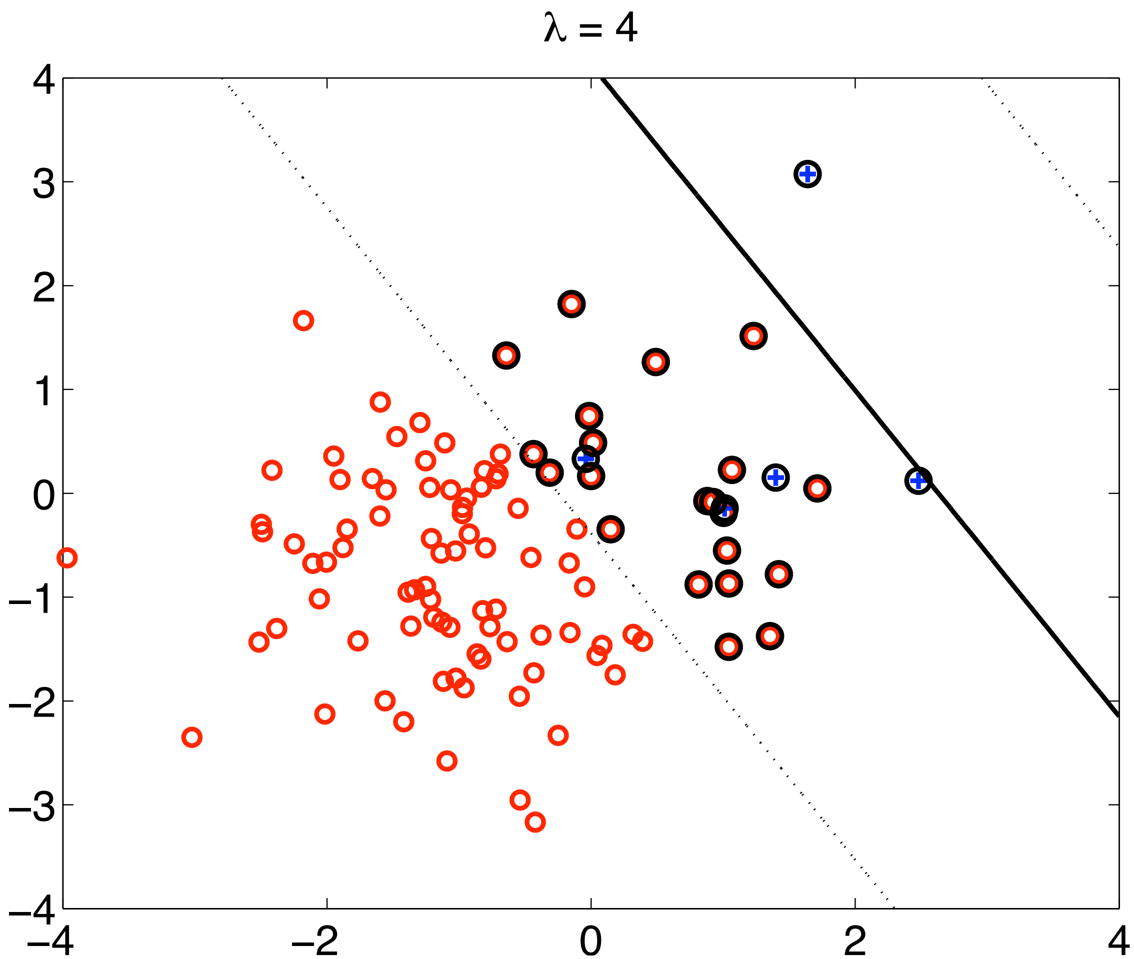


- Dimensionality no longer very high (# of tissue samples/conditions)
- Can use other kernels, e.g., radial basis kernel
- New problem: there are much more negatively labeled genes than positive

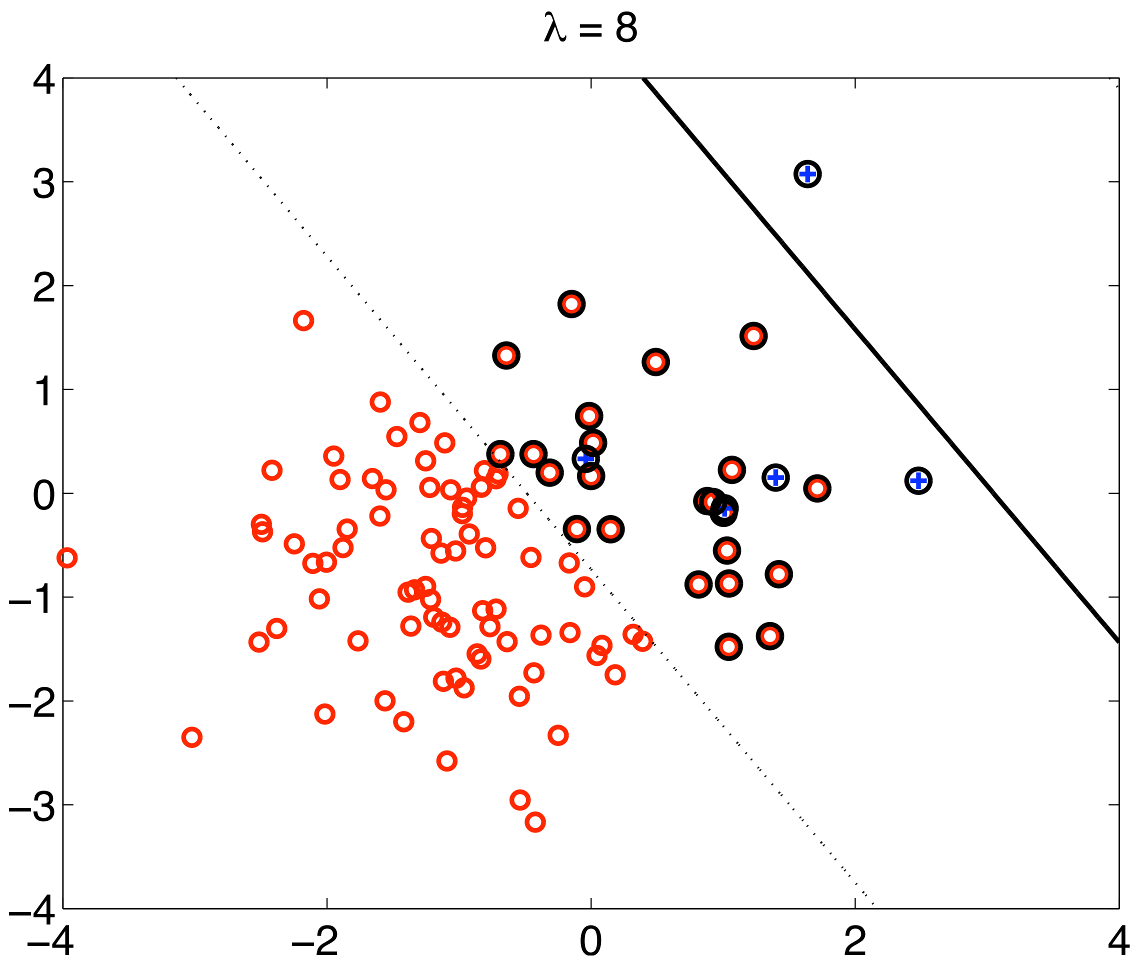
Imbalanced classes



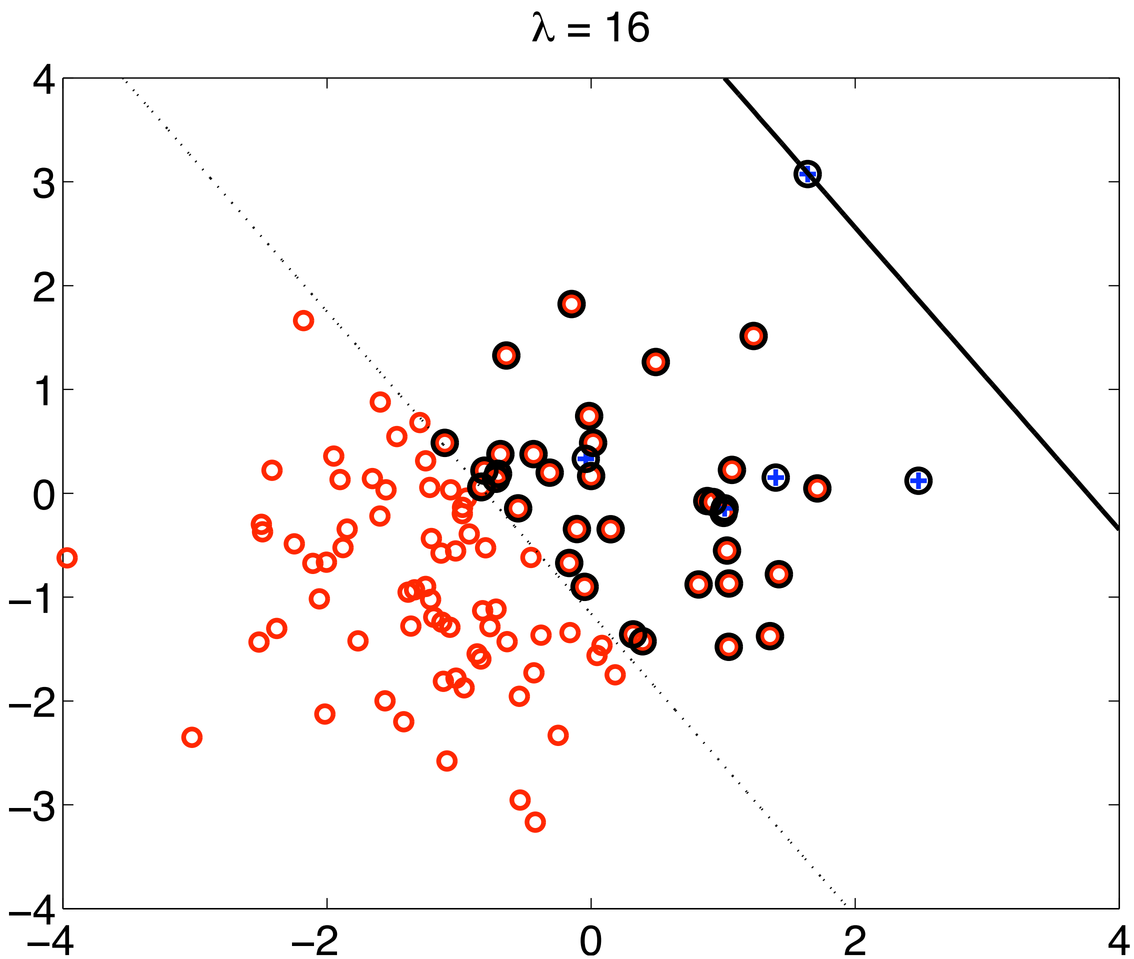
Imbalanced classes



Imbalanced classes



Imbalanced classes



Imbalanced classes

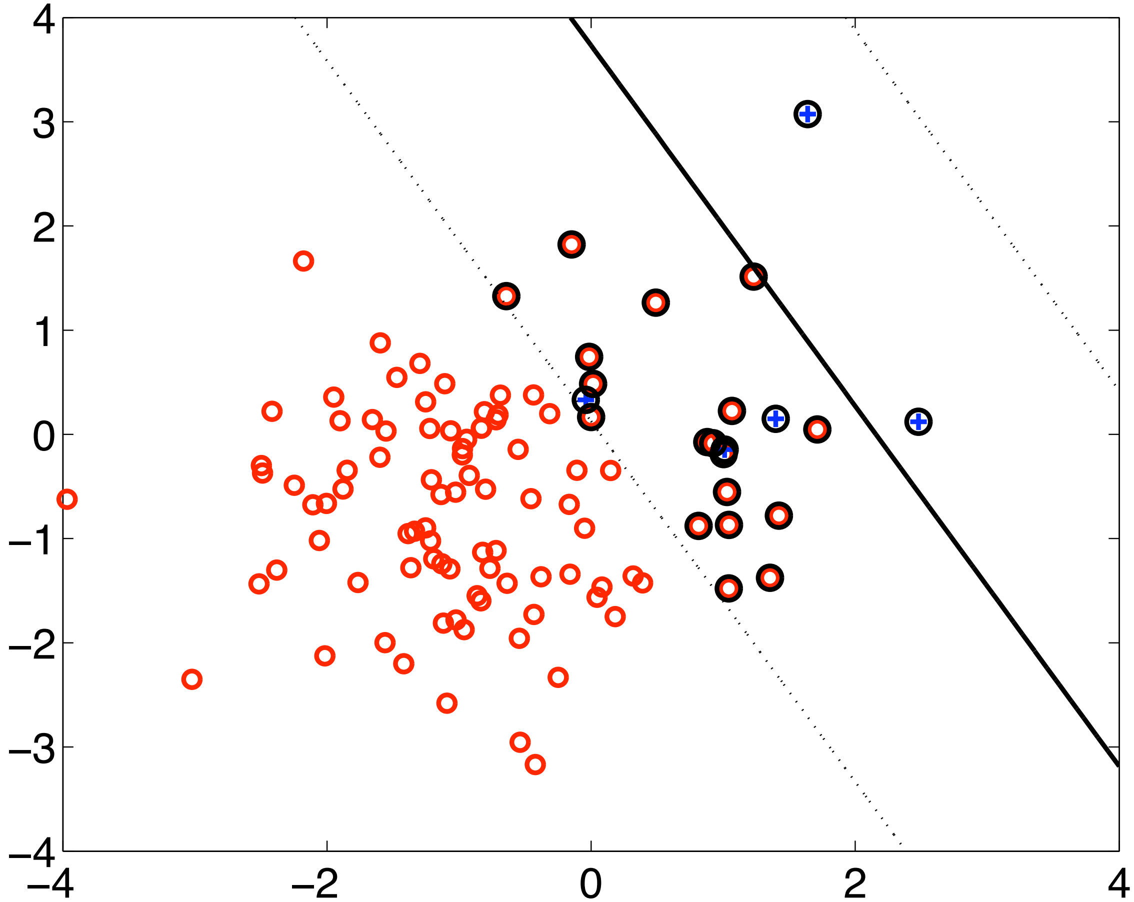
- In order to ensure that the classifier pays attention to the positive class, we increase (proportionally) resistance to negative examples

$$K \leftarrow \begin{bmatrix} K(x_1, x_1) + \lambda(n^+/n) & \cdots & K(x_1, x_n) \\ \cdots & \cdots & \cdots \\ K(x_n, x_1) & \cdots & K(x_n, x_n) + \lambda(n^-/n) \end{bmatrix}$$

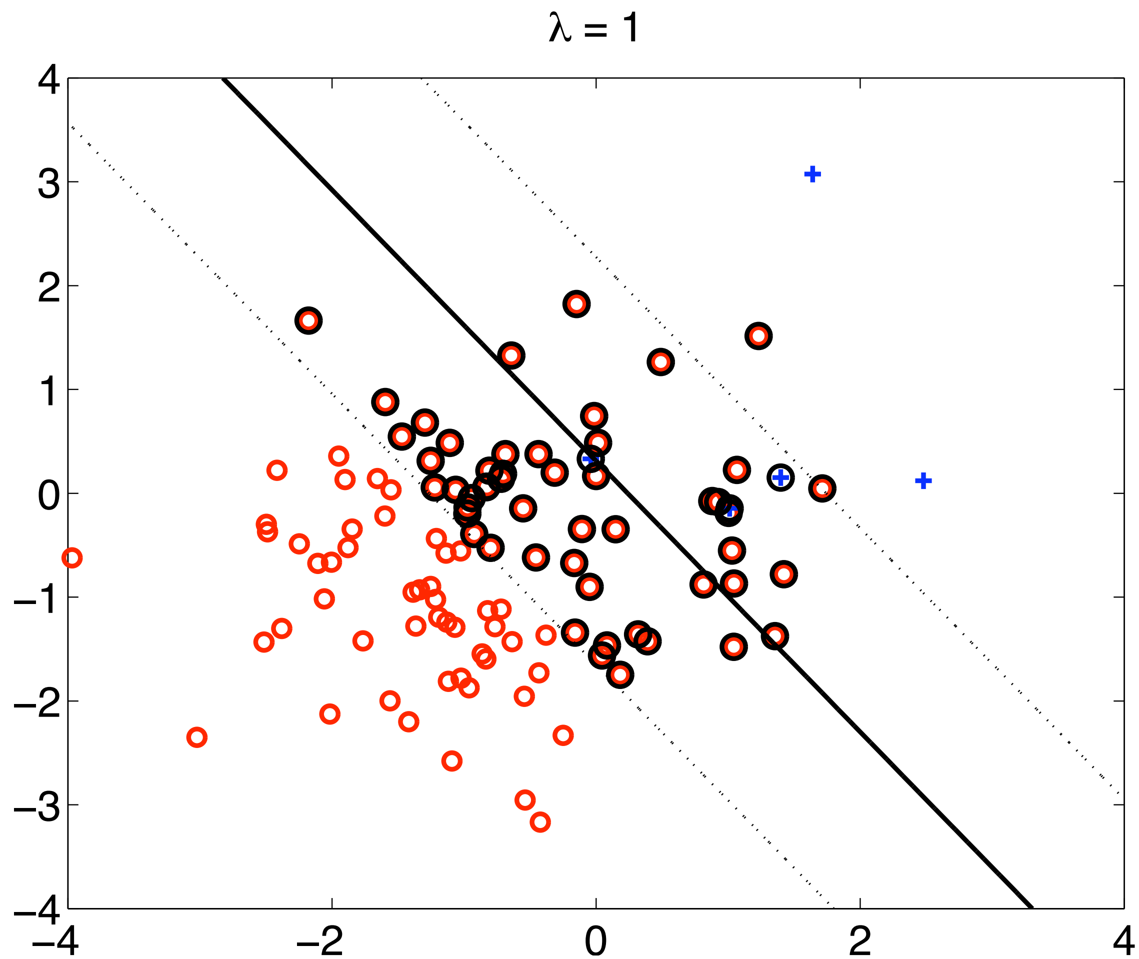
freq. of positive examples (n^+/n)

freq. of negative examples (n^-/n)

Differential resistance



Differential resistance



Functional annotation of genes

- SVMs perform very well (though there are other comparable methods)
- Learning methods can identify incorrectly annotated genes, predict functional roles for uncharacterized genes, as well as guide further experimental effort
- Used in many contexts; based on profiles, text, and/or sequence
  - e.g., understanding developmental roles of genes (lineage specific genes)
  - etc.

Problems we will cover

- Computational biology
  - cancer classification
  - functional classification of genes
- Information retrieval
  - document classification/ranking
- Recommender systems
  - predicting user preferences (e.g., movies)