Classification

Jian Ma

February 19, 2025

02-510/02-710 — Computational Genomics (Spring 2025)

Carnegie Mellon University

Today

- What is classification
- Different classifier types
- Cross validation
- Performance metrics

What is classification?

- **Classification** is the problem of identifying to which of a set of categories a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known.
  - Spam filtering
  - Disease diagnosis based on biomarkers

- Classification is considered a type of **supervised learning**.

What is classification?

- The individual observations are analyzed into a set of quantifiable properties
  - categorical (e.g. “A”, “B”, “AB” or “O”, for blood type)
  - ordinal (e.g. “large”, “medium” or “small”)
  - integer-valued (e.g. the number of occurrences of a word in an email)
  - real-valued (e.g. a measurement of blood pressure)

- An algorithm that implements classification, especially in a concrete implementation, is known as a **classifier**.

- The term “classifier” sometimes also refers to the mathematical function, implemented by a classification algorithm, that maps input data to a category.

Steps / Questions in classification

- Feature transformation
  - How do we encode the information (genomic signals, sequence, image)?
- Model / classifier specification
  - What type of classifier should we use?
- Model / classifier estimation
  - How do we learn the parameters of the classifier?
- Feature selection
  - Do we really need all the features? Can we use a smaller number and still achieve the same (or better) results?

Types of classifiers

- We can divide the large variety of classification approaches into roughly two main types

- 1. Generative:
  - build a generative statistical model
  - e.g., mixture models …

- 2. Discriminative
  - directly estimate a decision rule / boundary
  - e.g., SVM, logistic regression …

Golub et al.
***Science* 1999**

Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring

T. R. Golub,[1,2*†] D. K. Slonim,[1†] P. Tamayo,[1] C. Huard,[1] M. Gaasenbeek,[1] J. P. Mesirov,[1] H. Coller,[1] M. L. Loh,[2] J. R. Downing,[3] M. A. Caligiuri,[4] C. D. Bloomfield,[4] E. S. Lander[1,5*]

Although cancer classification has improved over the past 30 years, there has been no general approach for identifying new cancer classes (class discovery) or for assigning tumors to known classes (class prediction). Here, a generic approach to cancer classification based on gene expression monitoring by DNA microarrays is described and applied to human acute leukemias as a test case. A class discovery procedure automatically discovered the distinction between acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) without previous knowledge of these classes. An automatically derived class predictor was able to determine the class of new leukemia cases. The results demonstrate the feasibility of cancer classification based solely on gene expression monitoring and suggest a general strategy for discovering and predicting cancer classes for other types of cancer, independent of previous biological knowledge.

Molecular classification of cancer: class discovery and class prediction by gene expression monitoring
TR Golub, DK Slonim, P Tamayo, C Huard... - ..., 1999 - science.sciencemag.org
Although cancer classification has improved over the past 30 years, there has been no general approach for identifying new cancer classes (class discovery) or for assigning tumors to known classes (class prediction). Here, a generic approach to cancer classification based on gene expression monitoring by DNA microarrays is described and applied to human acute leukemias as a test case. A class discovery procedure automatically discovered the distinction between acute myeloid leukemia (AML) and acute lymphoblastic
☆ 99 Cited by 12430 Related articles All 82 versions Import into BibTeX

Golub et al.

- 38 test samples (27 ALL; 11 AML)
- Each gene was initially compared to an idealized expression pattern: 11111111111111000000000000000000 for class 1 and similarly 00000000000000000000111111111111 for the second class.

- The actual selection was done by setting:

$$p(g,c) = \frac{\mu_1(g) - \mu_2(g)}{\sigma_1(g) + \sigma_2(g)}$$

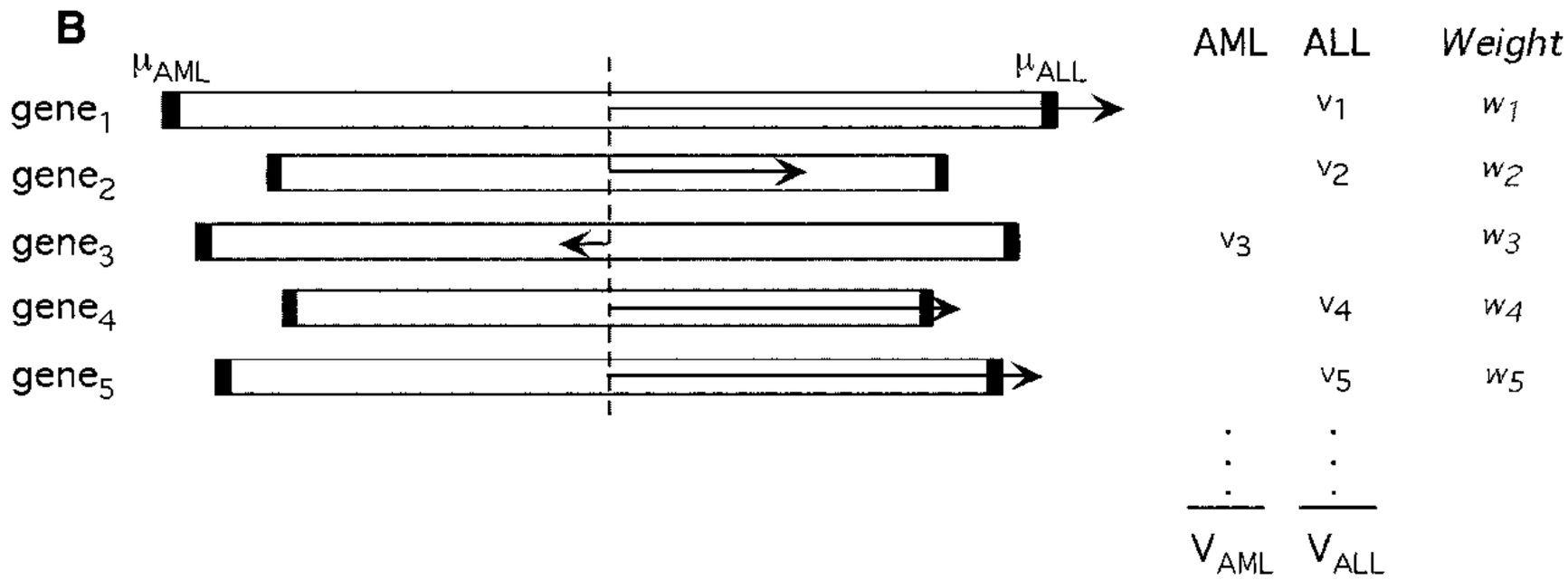
- Large values of $|p(g,c)|$ indicate strong correlation between the gene and the classes, and the sign of $|p(g,c)|$ depends on the class in which this gene is expressed.

Weighted voting

- Use a subset of the selected genes (50)
- Set $a_g = p(g, c)$ and $b_g = (\mu_1(g) + \mu_2(g))/2$
- Given a new sample x, we set the vote of gene g to:

$$v_g = a_g(x_g - b_g)$$

- A positive value is a vote for class 1 and a negative for the second class
- Reflects the deviation of the expression level in the sample from the average within the two classes


B
μ_{AML}
μ_{ALL}
$gene_1$
$gene_2$
$gene_3$
$gene_4$
$gene_5$
AML
ALL
Weight
v_1
w_1
v_2
w_2
v_3
w_3
v_4
w_4
v_5
w_5
V_{AML}
V_{ALL}

Voting strength

- The votes are summed for each of the two classes.

- The decision is made by using “prediction strength” (*PS*):

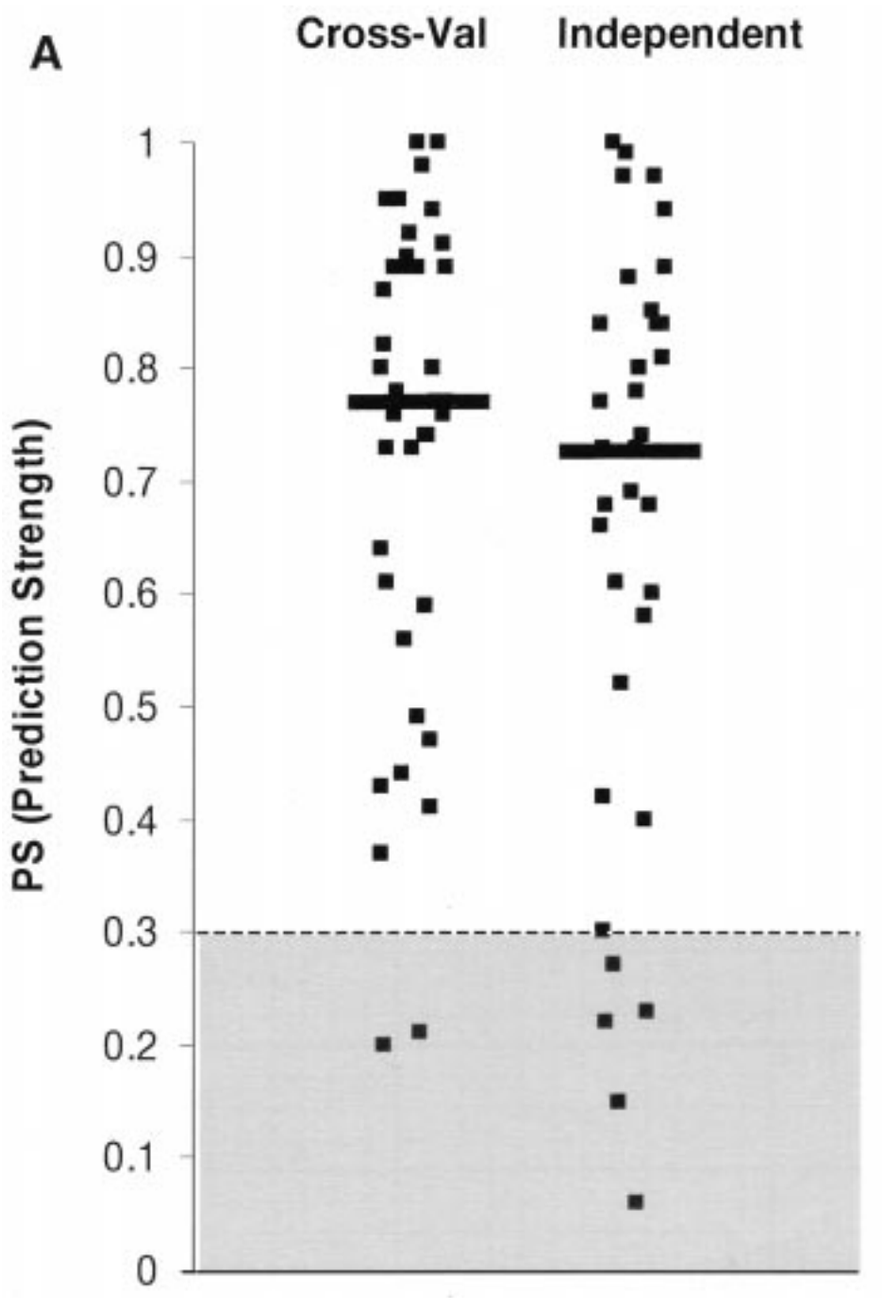
$$PS = \frac{v_{win} - v_{lose}}{v_{win} + v_{lose}}$$

- *PS* determines our confidence in the classification result.

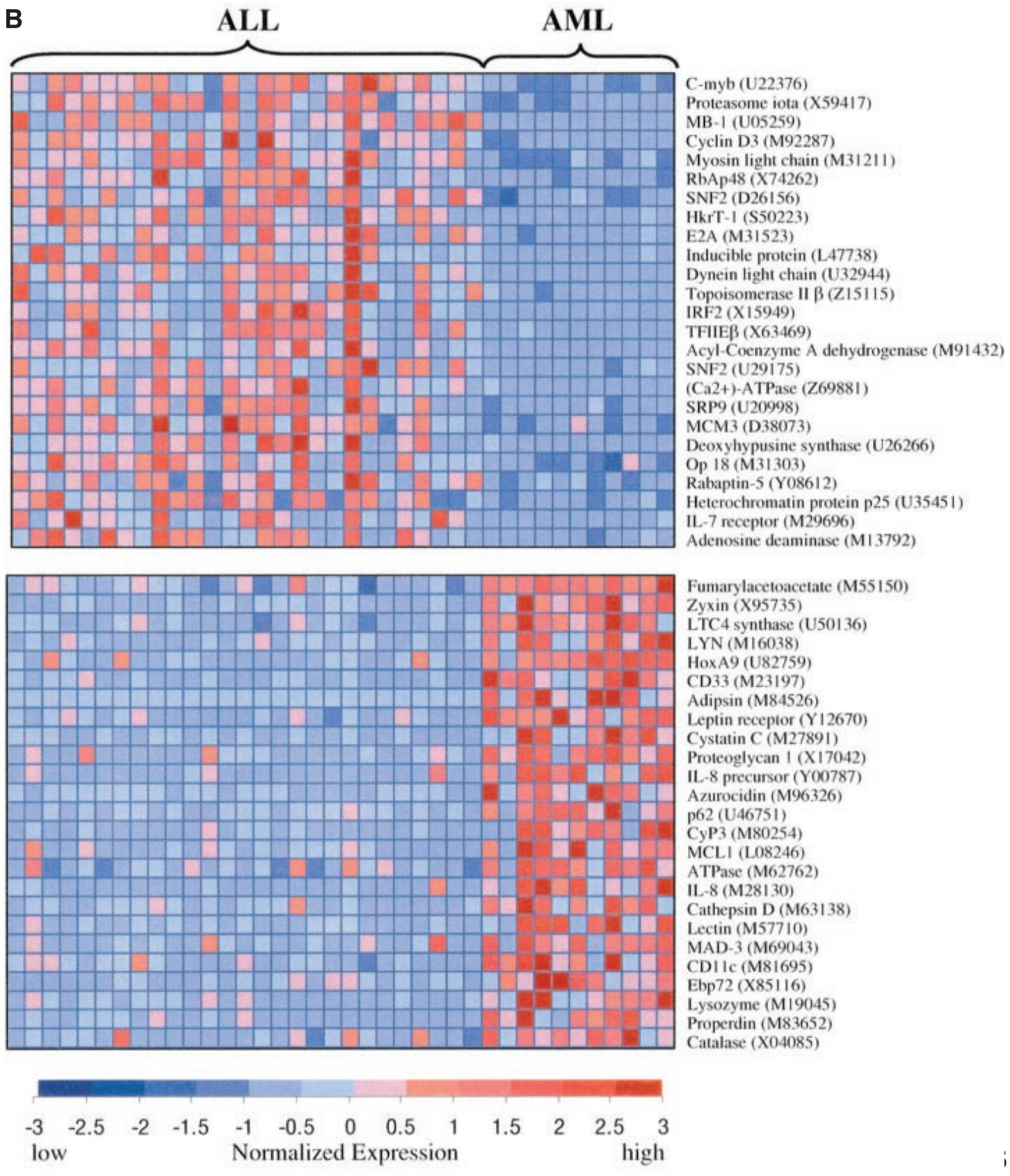
Testing the classifier

- Cross validation
- Independent test set: 34 samples:
  - 20 ALL
  - 14 AML
- 29 of 34 had a classification value higher than the threshold and all were predicted correctly.

Classification results



Selected genes



Generative classifiers: Bayes classification

- A mixture of two Gaussians, one Gaussian per class choice of class:

$$X \in \text{class } 1 \Rightarrow X \sim (\mu_1, \sigma_1)$$

$$X \in \text{class } 0 \Rightarrow X \sim (\mu_0, \sigma_0)$$

- where X corresponds to, e.g., a tissue sample (expression levels across the genes)

- Three basic problems we need to address:
  - decisions
  - estimation
  - variable (feature) selection

Decision: Bayesian classifiers

- Given a probabilistic model and an unlabeled data vector *X*, we can use Bayes rule to determine the class:

$$p(class = 1|X) = \frac{P(X|class = 1)P(class = 1)}{P(X|class = 1)P(class = 1) + P(X|class = 0)P(class = 0)}$$

- We compute $p(class = 1|X)$ and $p(class = 0|X)$ and choose the class with the highest probability

- The method can be easily extended to multiple classes

Decision boundary

- Given a probabilistic model and an unlabeled data vector *X*, we can use Bayes rule to determine the class:

$$p(class = 1|X) = \frac{P(X|class = 1)P(class = 1)}{P(X|class = 1) + P(X|class = 0)}$$

- Using Bayes classifiers, the decision comes down to the following (log) likelihood ratio:

$$\log \frac{p(X|\mu_1, \sigma_1)p(class = 1)}{p(X|\mu_0, \sigma_0)p(class = 0)} > 0 \Rightarrow class = 1$$

Decision boundaries

- Equal covariances

$$X \sim (\mu_1, \Sigma); class = 1$$

$$X \sim (\mu_0, \Sigma); class = 0$$

- The decision rule is linear

- Unequal covariances

$$X \sim (\mu_0, \sigma_1); class = 1$$

$$X \sim (\mu_0, \sigma_0); class = 0$$

- The decision rule is quadratic

Estimation

- Suppose we are given a set of labeled tissue samples

$$x_1, ..., x_k \Rightarrow class = 1$$

$$x_{k+1}, ..., x_n \Rightarrow class = 0$$

- We can estimate the two Gaussians separately

- For example, using MLE we get:

$$P(class = 1) = k/n$$
$$\mu_1 = \text{sample mean of } x_1, ..., x_k$$
$$\sigma_1 = \text{sample covariance of } x_1, ..., x_k$$

- And similarly for the other classes

Golub et al.

- Leukemia classification problem

- 7,120 ORFs (expression levels)
- 38 labeled training examples
- 34 testing examples

- Our mixture model (assume equal class priors)

$$X \sim (\mu_1, \Sigma); class = 1$$
$$X \sim (\mu_0, \Sigma); class = 0$$

- Problems?

Naive Bayes classifiers

- This full covariance model is too complex, we need to constrain the covariance matrices
- The simplest constraint we can use is a diagonal covariance matrix instead of a full covariance
- When using such a matrix we make the (implicit) assumption that the genes are independent given the class labels
- In other words, we assume that:

$$p(X|class = 1) = \prod_i p(X_i|class = 1)$$

$$X_i \sim N(\mu_i^1, \sigma_i^2)$$

where X_i is the expression value for gene *i*

Naive Bayes classifiers

- Lets further assume equal variance for a specific gene across the two sets of samples (that is, noise is independent of the sample condition)

- As a result, we need to only estimate class-conditional means and a common variance for each gene

Feature selection

- Test which genes are predictive of the class distinction

- Why is this important?

- H_0 is that a gene is not predictive of the class label
- H_1 is that a gene can predict the class label

$$H_0 = X_1 \sim N(\mu, \sigma^2), X_2 \sim N(\mu, \sigma^2)$$
$$H_1 = X_1 \sim N(\mu_1, \sigma^2), X_2 \sim N(\mu_2, \sigma^2)$$

- We can use a likelihood ratio test for this purpose.

  Let X_i^t denote the observed expression levels for gene *i*

- The parameter estimates are computed from the available populations in accordance with the hypothesis

Gene selection

- We rank the genes in the descending order of the test statistics $T(X_i)$
- How many genes should we include?
- We can use various multiple hypothesis correction methods, e.g., FDR

Types of classifiers

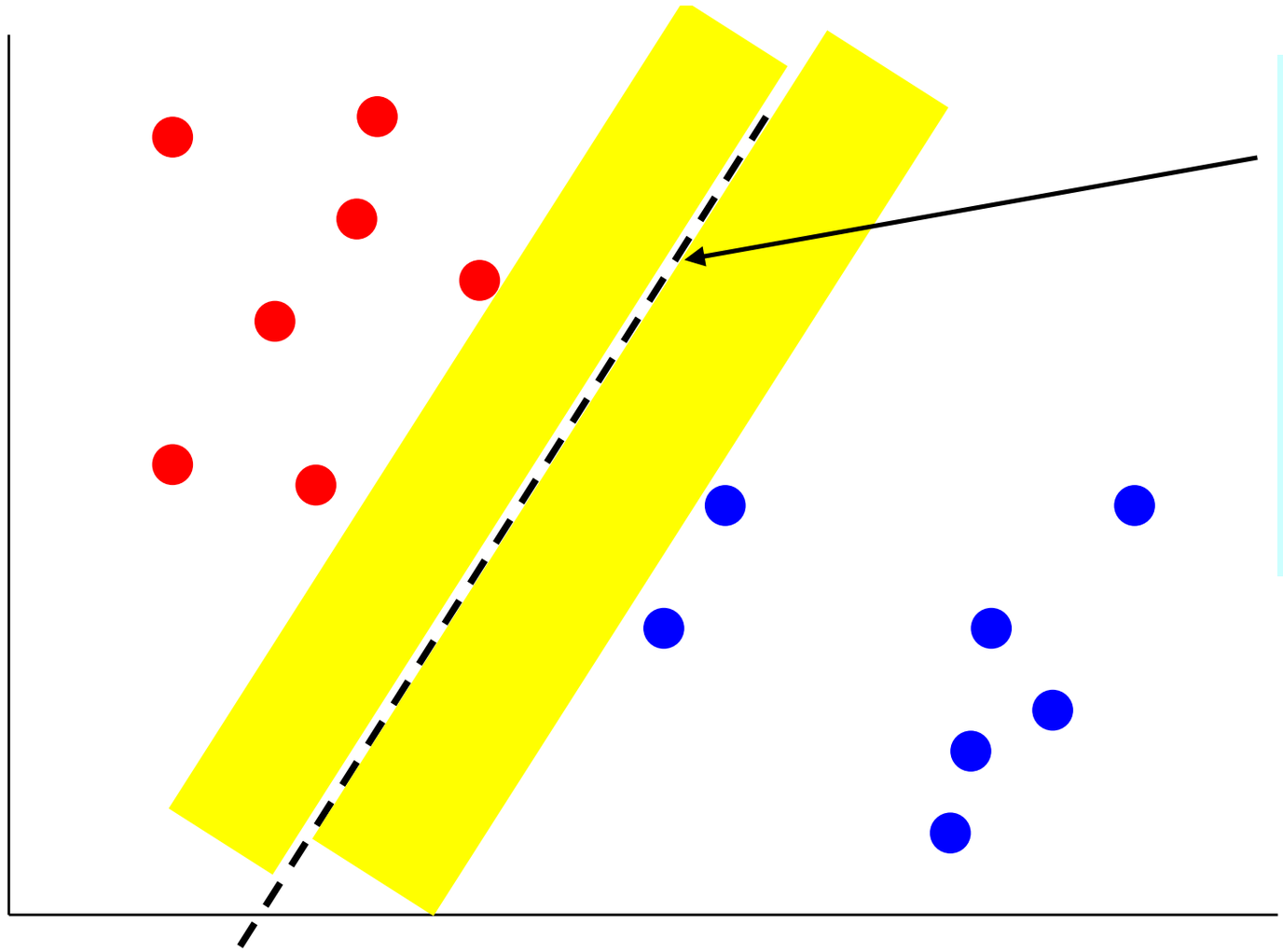
- We can divide the large variety of classification approaches into roughly two main types

- 1. Generative:
  - build a generative statistical model
  - e.g., mixture models …

- 2. Discriminative
  - directly estimate a decision rule / boundary
  - e.g., SVM, logistic regression …

SVM: A max margin classifier

- Instead of fitting all points, focus on boundary points
- Learn a boundary that leads to the largest margin from both sets of points



From all the possible boundary lines, this leads to the largest margin on both sides

SVM for non-linearly separable data

- SVM optimizes the following:

$$\min_w \frac{W^T W}{2} + \sum_{i=1}^{n} C\varepsilon_i$$

subject to the following constraints:

For all x_i in class +1

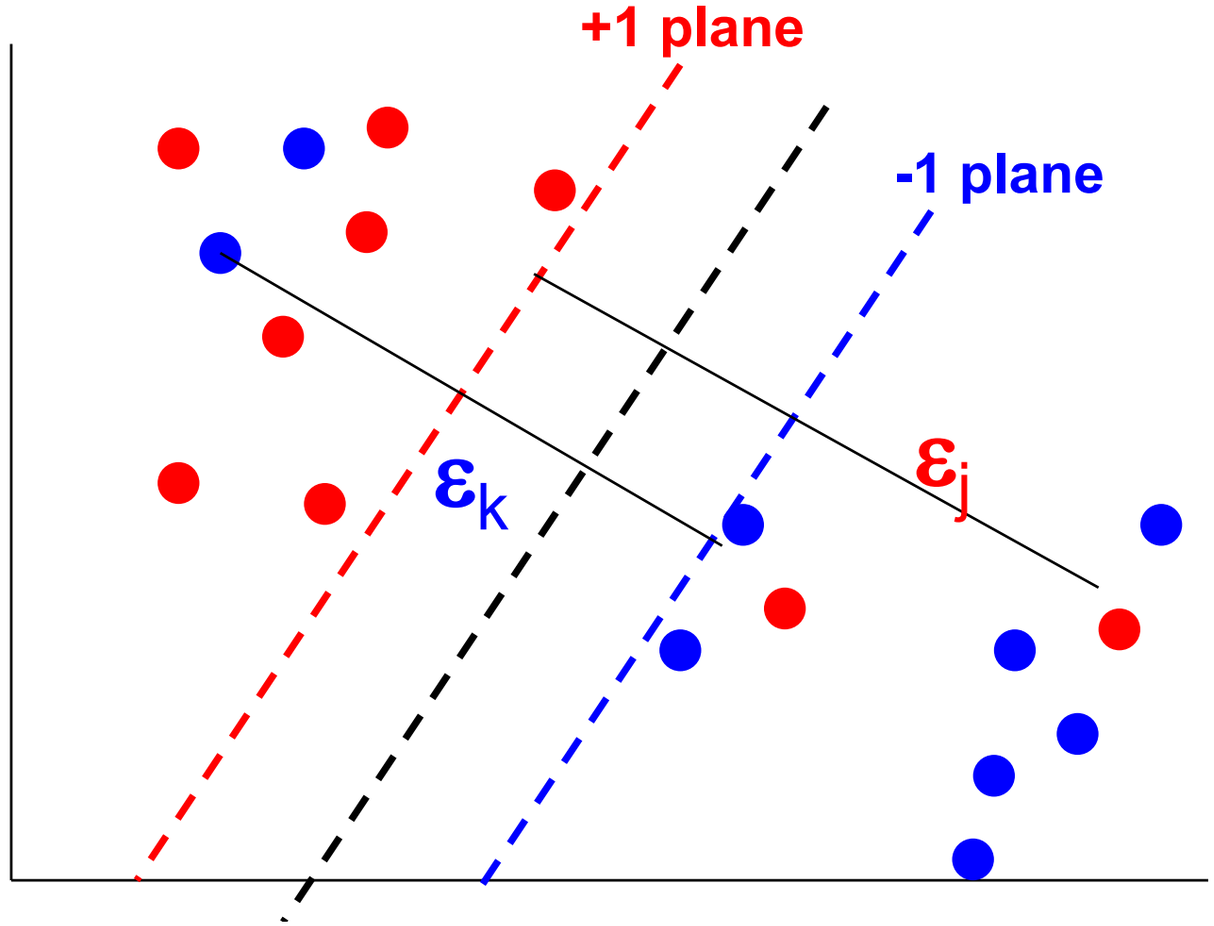
$$W^T X + b \geq 1 - \varepsilon_i$$

For all x_i in class -1

$$W^T X + b \leq -1 + \varepsilon_i$$

For all i

$$\varepsilon_i \geq 0$$



Other classifiers

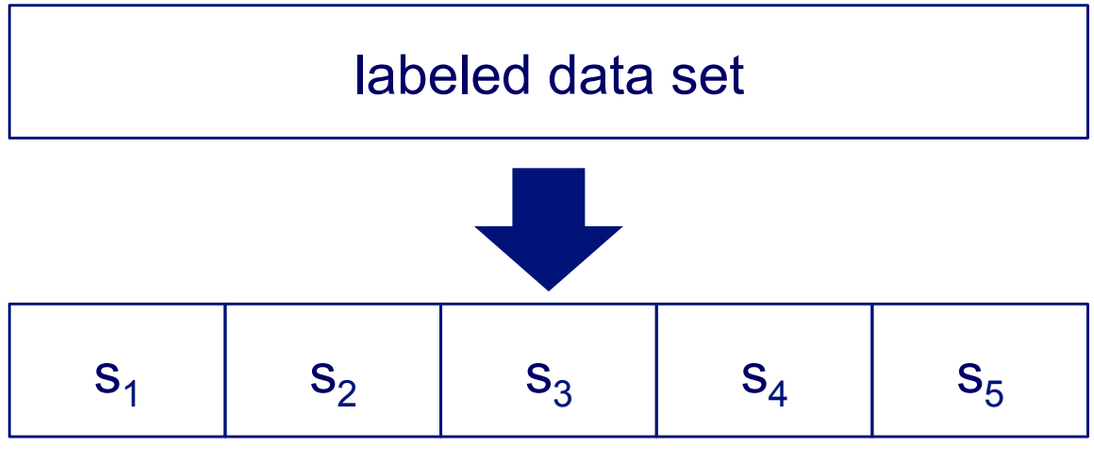
- We will discuss some of these in the context of genomic applications later in the semester —
- Logistic regression
- Random forest
- Gradient boosted trees
- Deep neural networks
- …

Limitations of using a single training/testing

- We may not have enough data to make sufficiently large training and test sets
  - A larger test set gives us more reliable estimate of accuracy (i.e. a lower variance estimate)
  - but … a larger training set will be more representative of how much data we actually have for learning process
- A single training set doesn't tell us how sensitive accuracy is to a particular training sample

Cross validation

- Partition data into *n* subsamples
- Iteratively leave one subsample out for the test dataset, train on the rest
- *n* = 10 is common choice



iteration	train on	test on
1	s_2 s_3 s_4 s_5	s_1
2	s_1 s_3 s_4 s_5	s_2
3	s_1 s_2 s_4 s_5	s_3
4	s_1 s_2 s_3 s_5	s_4
5	s_1 s_2 s_3 s_4	s_5

Cross validation

- 10-fold cross validation is common, but smaller values of *n* (e.g., 5) are often used when learning takes a lot of time
- In leave-one-out cross validation, n=# instances
- CV makes efficient use of the available data for testing
- Note that whenever we use multiple training sets, as in CV and random resampling, we are evaluating a learning method as opposed to an individual learned model
- *Internal cross validation* — Instead of a single validation set, we can use cross-validation within a training set to select a model

Confusion matrix for 2-class problem

predicted class	actual class: positive	actual class: negative
positive	true positives (TP)	false positives (FP)
negative	false negatives (FN)	true negatives (TN)

$$\text{arruracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

- However, accuracy may not be useful measure in cases where there is a large class skew

Other metric

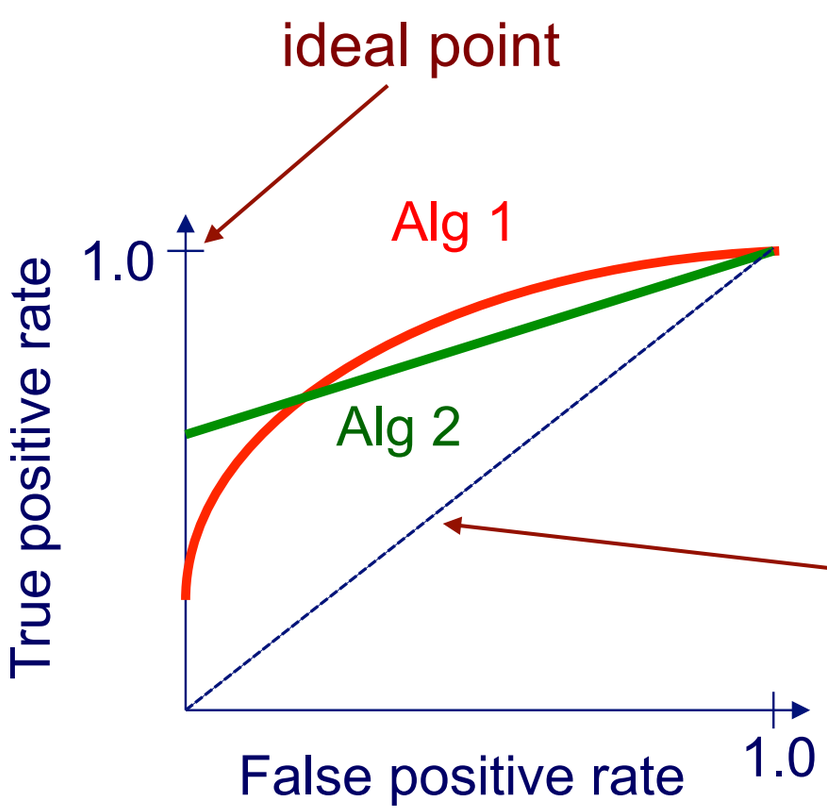
predicted class	actual class: positive	actual class: negative
positive	true positives (TP)	false positives (FP)
negative	false negatives (FN)	true negatives (TN)

$$\text{true positive rate (recall)} = \frac{TP}{TP + FN}$$

$$\text{false positive rate} = \frac{FP}{TN + FP}$$

ROC curves

- A Receiver Operating Characteristic (ROC) curve plots the TP-rate vs. the FP-rate as a threshold on the confidence of an instance being positive is varied



Different methods can work better in different parts of ROC space. This depends on cost of false + vs. false -

expected curve for random guessing

Other metric

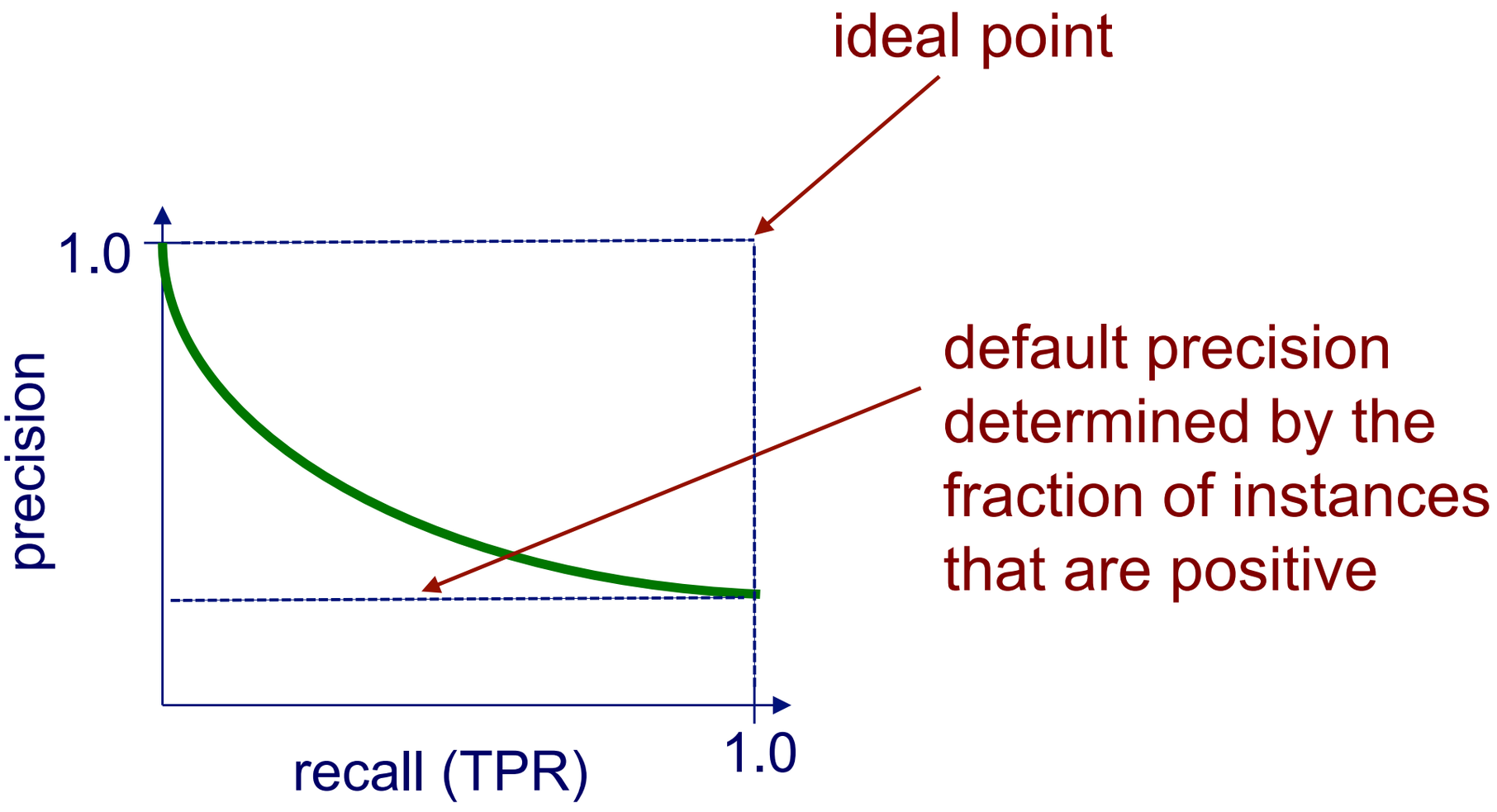
predicted class \ actual class	positive	negative
positive	true positives (TP)	false positives (FP)
negative	false negatives (FN)	true negatives (TN)

$$\text{true positive rate (recall)} = \frac{TP}{TP + FN}$$

$$\text{precision} = \frac{TP}{TP + FP}$$

Precision / recall curve

- A precision/recall curve plots the precision vs. recall (TP-rate) as a threshold on the confidence of an instance being positive is varied



How do we get one ROC/PR curve when we do cross validation?

- Approach 1
  - Make assumption that confidence values are comparable across folds
  - Pool predictions from all test sets
  - Plot the curve from the pooled predictions
- Approach 2 (for ROC curves)
  - Plot individual curves for all test sets
  - View each curve as a function
  - Plot the average curve for this set of functions

Comparison between ROC and PR curves

- Both approaches
  - Allow predictive performance to be assessed at various levels of confidence
  - Assume binary classification tasks
  - Sometimes summarized by calculating area under the curve

- ROC curves
  - Insensitive to changes in class distribution (ROC curve does not change if the proportion of positive and negative instances in the test set are varied)
  - Can identify optimal classification thresholds for tasks with differential misclassification costs

- Precision / Recall curves
  - Show the fraction of predictions that are false positives
  - Well suited for tasks with lots of negative instances

Questions to ask yourself
(before submitting your paper)

- Is my held-aside test data really representative of going out to collect new data?
  - Even if your methodology is fine, someone may have collected features for positive examples differently than for negatives – should be randomized
  - Example: samples from cancer processed by different people or on different days than samples for normal controls

- Did I repeat my entire data processing procedure on every fold of cross-validation, using only the training data for that fold?
  - On each fold of cross-validation, did I ever access in any way the label of a test case?
  - Any preprocessing done over entire data set (feature selection, parameter tuning, threshold selection) must not use labels

- Have I modified my algorithm so many times, or tried so many approaches, on this same data set that I (the human) am overfitting it?
  - Have I continually modified my preprocessing or learning algorithm until I got some improvement on this data set?
  - If so, I really need to get some additional data now to at least test on

Summary

- What is classification
- Different classifier types
- Cross validation
- Performance metrics
- Homework: spend time to play with **scikit-learn (http://scikit-learn.org/)**