Comparison of discrimination methods for the classification of tumors using gene expression data

Sandrine Dudoit[1], Jane Fridlyand[2] and Terry Speed[2,3]

1. Mathematical Sciences Research Institute, Berkeley
2. Department of Statistics, UC Berkeley
3. Walter and Eliza Hall Institute, Melbourne

Tumor classification

A reliable and precise classification of tumors is essential for successful treatment of cancer.
Current methods for classifying human malignancies rely on a variety of morphological, clinical, and molecular variables.
In spite of recent progress, there are still uncertainties in diagnosis. Also, it is likely that the existing classes are heterogeneous.
DNA microarrays may be used to characterize the molecular variations among tumors by monitoring gene expression profiles on a genomic scale.
This may lead to a more reliable classification of tumors.

Tumor classification, cont'd

There are three main types of statistical problems associated with tumor classification:

1. the identification of new/unknown tumor classes using gene expression profiles - *cluster analysis / unsupervised learning*;
2. the classification of malignancies into known classes - *discriminant analysis / supervised learning*;
3. the identification of "marker" genes that characterize the different tumor classes - *variable selection*.

Gene expression data

Gene expression data on p genes (variables) for n mRNA samples (observations)

$$\begin{array}{cc} & \text{Genes} \\ X_{n\times p} = & \left[\begin{array}{cccc} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{array}\right] & \text{mRNA samples} \end{array}$$

$$\begin{aligned} x_{ij} &= \text{gene expression level of gene } j \text{ in mRNA sample } i \\ &= \begin{cases} \log\left(\frac{\text{Red intensity}}{\text{Green intensity}}\right), \\ \log(\text{Avg. PM} - \text{Avg. MM}). \end{cases} \end{aligned}$$

Gene expression data, cont'd

In some situations, the mRNA samples are known to belong to certain classes ($e.g.$ follicular lymphoma).
Label the classes by $\{1, 2, \ldots, K\}$.
Then, the data for each observation consist of:

$$\begin{aligned}
\mathbf{x}_i &= (x_{i1}, x_{i2}, \ldots, x_{ip}) \\
&\quad \text{- gene expression profile / predictor variables} \\
y_i &= \text{tumor class / response.}
\end{aligned}$$

The prediction problem

Want to predict a **response** y given **predictor variables** $\mathbf{x}$.

Task: construct a prediction function f, such that $f(\mathbf{x})$ is an accurate predictor of y.

E.g. digit recognition for zipcodes;
prediction of binding peptide sequences;
prediction of tumor class from gene expression data.

Predictors

A **predictor** or **classifier** for K tumor classes partitions the space $\mathcal{X}$ of gene expression profiles into K disjoint subsets, $A_1, \ldots, A_K$, such that for a sample with expression profile $\mathbf{x} = (x_1, \ldots, x_p) \in A_k$ the predicted class is k.
Predictors are built from past experience, $i.e.$ from observations which are known to belong to certain classes.
Such observations comprise the **learning set (LS)** $\mathcal{L} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$.
Classifier built from a learning set $\mathcal{L}$:

$$C(\cdot, \mathcal{L}) : \mathcal{X} \to \{1, 2, \ldots, K\}.$$

Predicted class for an observation $\mathbf{x}$: $C(\mathbf{x}, \mathcal{L})$.

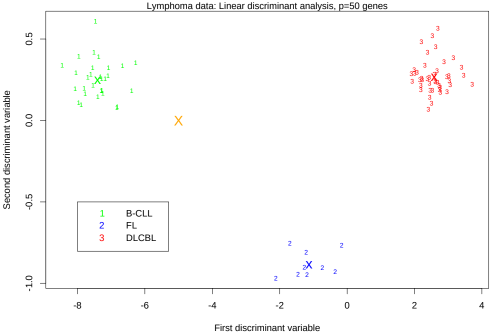
Linear discriminant analysis

Suggested in 1936 by R. A. Fisher, **linear discriminant analysis (LDA)** consists of

1. finding linear combinations $\mathbf{x}\,\mathbf{a}$ of the gene expression profiles $\mathbf{x} = (x_1, \ldots, x_p)$ with large ratios of between-groups to within-groups sum of squares - **discriminant variables**;

2. predicting the class of an observation $\mathbf{x}$ by the class whose mean vector is closest to $\mathbf{x}$ in terms of the discriminant variables.

Linear discriminant analysis, cont'd



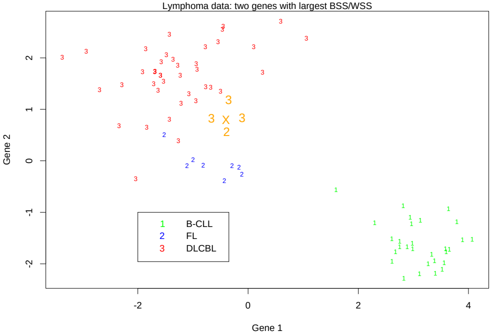
Nearest neighbor classifier

These methods are based on a measure of distance between observations, such as the Euclidean distance or one minus the correlation between two gene expression profiles.
The k **nearest neighbor** rule, due to Fix and Hodges (1951), classifies an observation $\mathbf{x}$ as follows:

1. find the k observations in the learning set that are closest to $\mathbf{x}$;
2. predict the class of $\mathbf{x}$ by majority vote, *i.e.*, choose the class that is most common among those k observations.

The number of neighbors k is chosen by cross-validation.

Nearest neighbor classifier, cont'd

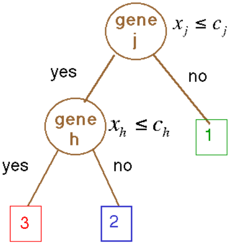


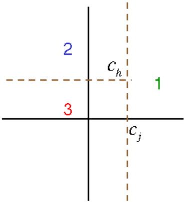
Classification trees

Binary tree structured classifiers are constructed by repeated splits of subsets (nodes) of the measurement space $\mathcal{X}$ into two descendant subsets, starting with $\mathcal{X}$ itself. Each terminal subset is assigned a class label and the resulting partition of $\mathcal{X}$ corresponds to the classifier.
Three main aspects of tree construction: (i) the selection of the splits; (ii) the decision to declare a node terminal or to continue splitting; (iii) the assignment of each terminal node to a class.
Different tree classifiers use different approaches to deal with these three issues. Here, we use **CART - Classification And Regression Trees** - of Breiman *et al.* (1984).

Classification trees, cont'd

New observation: $\mathbf{x} = (x_1, ..., x_p)$





Aggregating predictors

Breiman (1996, 1998) found that gains in accuracy could be obtained by **aggregating predictors** built from perturbed versions of the learning set. In classification, the multiple versions of the predictor are aggregated by **voting**. Let $C(\cdot, \mathcal{L}_b)$ denote the classifier built from the bth perturbed learning set $\mathcal{L}_b$ and let w_b denote the weight given to predictions made by this classifier. The predicted class for an observation $\mathbf{x}$ is given by

$$\mathrm{argmax}_k \sum_b w_b \, I(C(\mathbf{x}, \mathcal{L}_b) = k).$$

Bagging

Breiman (1996).

In the simplest form of **bagging** - bootstrap **aggregating** - perturbed learning sets of the same size as the original learning set are formed by forming non-parametric bootstrap replicates of the learning set, $i.e.$ by drawing at random with replacement from the learning set.

Predictors are built for each perturbed dataset and aggregated by plurality voting $(w_b = 1)$.

Variants on bagging

Parametric bootstrap. Perturbed learning sets are generated according to a mixture of multivariate normal (MVN) distributions.

Convex pseudo-data. Breiman (1996)

Each perturbed learning set is generated by repeating the following n times:

1. select two instances $(\mathbf{x}, y)$ and $(\mathbf{x}', y')$ at random from the learning set;

2. select at random a number v from the interval $[0, d]$, $0 \leq d \leq 1$, and let $u = 1 - v$;

3. define a new instance $(\mathbf{x}'', y'')$ by $y'' = y$ and $\mathbf{x}'' = u\mathbf{x} + v\mathbf{x}'$.

Boosting

Freund and Schapire (1997), Breiman (1998).
The data are **re-sampled adaptively** so that the weights in the re-sampling are increased for those cases most often misclassified.
The aggregation of predictors is done by **weighted voting**.

For a learning set $\mathcal{L} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, let $\{p_1, \ldots, p_n\}$ denote the re-sampling probabilities, initialized to be equal. For bth step of the boosting algorithm (adaptation of `AdaBoost`):

Boosting, cont'd

1. generate a perturbed learning set $\mathcal{L}_b$ of size n by sampling with replacement from $\mathcal{L}$ using $\{p_1, \ldots, p_n\}$;
2. build a classifier $C(\cdot, \mathcal{L}_b)$ based on $\mathcal{L}_b$;
3. run the learning set $\mathcal{L}$ through the classifier $C(\cdot, \mathcal{L}_b)$ and let $d_i = 1$ if the ith case is classified incorrectly and $d_i = 0$ o.w.;
4. define

$$\epsilon_b = \sum_i p_i d_i, \;\; \beta_b = (1 - \epsilon_b)/\epsilon_b \text{ and } w_b = \log(\beta_b)$$

and update the re-sampling probabilities for the $(b+1)$st step by

$$p_i = \frac{p_i \beta_b^{d_i}}{\sum_i p_i \beta_b^{d_i}}.$$

Vote margins

For aggregate classifiers, vote margins assessing the strength of a prediction may be defined for each observation.

The **vote margin** for an observation $\mathbf{x}$ is defined to be

$$M(\mathbf{x}) = \frac{\max_k \sum_b w_b \, I(C(\mathbf{x}, \mathcal{L}_b) = k)}{\sum_b w_b}.$$

When the perturbed learning sets are given equal weights, $i.e.\ w_b = 1$, the vote margin is simply the proportion of votes for the "winning" class, regardless of whether it is correct or not.

Margins belong to $[0, 1]$.

Gene voting, Golub *et al.* (1999)

For binary classification, each gene casts a vote for class 1 or 2, and the votes are aggregated over genes. Gene j's vote for a test set observation $\mathbf{x} = (x_1, \ldots, x_p)$ is given by

$$v_j = a_j(x_j - b_j),$$

where

$$a_j = \frac{\bar{x}_{.j}^{(1)} - \bar{x}_{.j}^{(2)}}{sd_j^{(1)} + sd_j^{(2)}}, \qquad b_j = \frac{1}{2}(\bar{x}_{.j}^{(1)} + \bar{x}_{.j}^{(2)}).$$

Let V_1 and V_2 denote the sum of positive and negative votes, respectively. The predicted class is 1 if $V_1 \geq V_2$ and 2 otherwise. This is a minor variant on the **sample ML discriminant rule** for multivariate normal class densities with constant diagonal covariance matrices.

Datasets - Lymphoma

Study of gene expression in the three most prevalent adult lymphoid malignancies using a specialized cDNA microarray, the Lymphochip (Alizadeh *et al.*, 2000).

- $n = 81$ mRNA samples, three classes:

B-cell chronic lymphocytic leukemia (B-CLL)	29 cases
Follicular lymphoma (FL)	9 cases
Diffuse large B-cell lymphoma (DLBCL)	43 cases

- $p = 4,682$ genes.

Datasets - Leukemia

Study of gene expression in two types of acute leukemias using Affymetrix high-density oligonucleotide arrays (Golub *et al.*, 1999).

- $n = 72$ mRNA samples, three classes:

B-cell acute lymphoblastic leukemia (B-cell ALL)	38 cases
T-cell acute lymphoblastic leukemia (T-cell ALL)	9 cases
Acute myeloid leukemia (AML)	25 cases

- $p = 6,817$ genes.

Data pre-processing

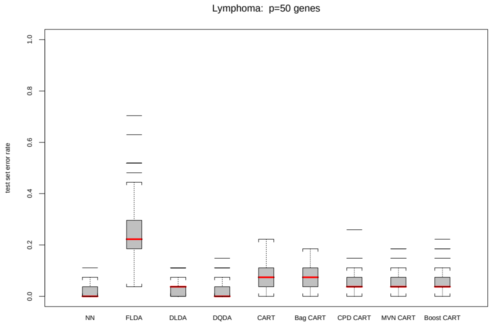
- **Imputation.** k-nearest neighbor imputation, where genes are "neighbors" and the similarity measure between two genes is the correlation in their expression profiles.

- **Standardization.** Standardize observations (arrays) to have mean 0 and variance 1 across variables (genes).

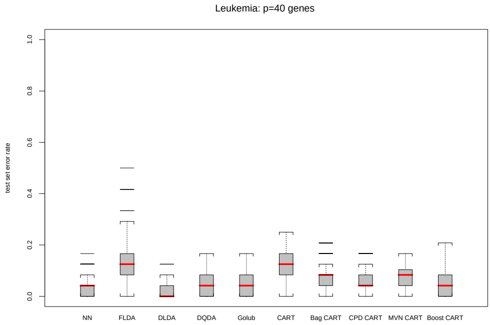
Study design

The original datasets are repeatedly randomly divided into a learning set and a test set, comprising respectively $2/3$ and $1/3$ of the data. For each of $N = 150$ runs:

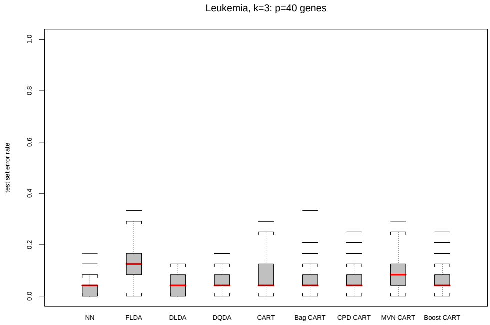
- Select a subset of p genes from the learning set based on their ratio of between to within-groups sum of squares, BSS/WSS. $p = 50$ for lymphoma, $p = 40$ for leukemia.
- Build the different predictors using the learning sets with p genes.
- Apply the predictors to the observations in the test set to obtain test set error rates.



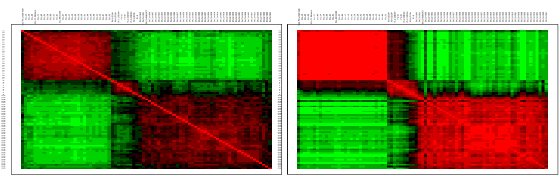
Lymphoma. Boxplots of test set error rates, $p = 50$ genes.



Leukemia, two classes. Test set error rates, $p = 40$ genes.

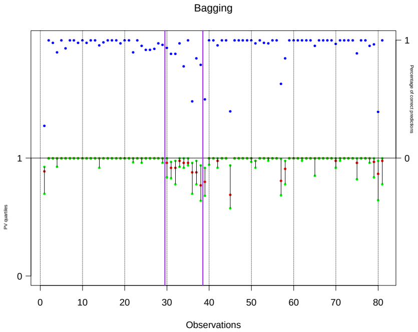


Leukemia, three classes. Test set error rates, $p = 40$ genes.

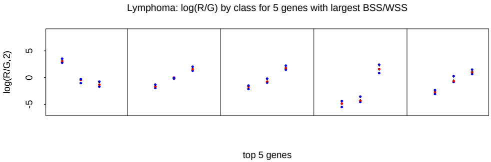


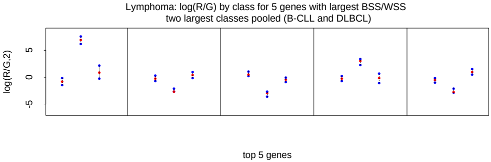
4,682 genes   50 genes

Lymphoma. Images of correlation matrix between 81 mRNA samples.

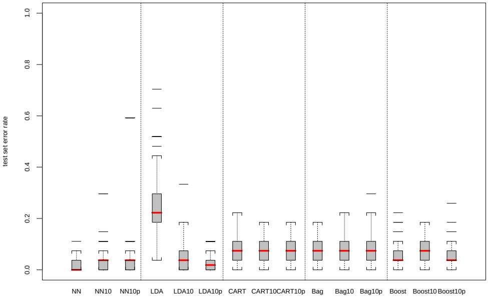


Lymphoma. Proportion of correct predictions (upper panel) and quartiles of vote margins (lower panel) for bagged CART predictors, $p = 50$ genes.





Lymphoma, three classes. Within class IQRs for 5 genes.



Lymphoma. Test set error rates for different gene sets.

Results

- In the main comparison, the nearest neighbor predictor has the smallest error rates, while LDA has the highest error rates.
- Aggregating predictors improves performance, the largest gains being with boosting and bagging with convex pseudo-data.
- For the binary class leukemia data, "diagonal" LDA as in Golub *et al.* performs similarly to nearest neighbors, boosting and bagging with convex pseudo-data.

- For the lymphoma or leukemia datasets, increasing the number of variables to $p = 200$ doesn't affect much the performance of the various predictors.

- A more careful selection of a small number of genes $(p = 10)$ improves the performance of LDA dramatically.

Discussion

- "Diagonal" LDA vs. "correlated" LDA: ignoring correlation between genes helps here.
- Unlike classification trees and nearest neighbors, "diagonal" or "correlated" LDA is unable to take into account gene interactions.
- Although nearest neighbors are simple and intuitive classifiers, their main limitation is that they give very little insight into mechanisms underlying the class distinctions.
- Classification trees are capable of handling and revealing interactions between variables.

- Useful by-product of aggregated predictors: vote margins.

- The relative performance of the different predictors may vary with variable selection.

Open questions

- **Variable selection**. A crude criterion such as BSS/WSS may not identify the genes that discriminate between all the classes and may not reveal interactions between genes.
  Statistical vs. biological significance.

- **Cluster analysis**. Identification and validation of new tumor classes.

Acknowledgments

Biologists: Ash Alizadeh, Pat Brown, Mike Eisen.
Statisticians: Leo Breiman, Sam Buttrey, David Nelson, Mark van der Laan, Jean Yang.