

CLASSIFICATION IN MICROARRAY EXPERIMENTS

- ➔ Statistical foundation of Classification
- ➔ Overview of different classifiers
 - The Bayes rule
 - Maximum likelihood
 - Linear and quadratic discriminant analysis
 - Some issues in classification
- ➔ Example
- ➔ Final Remarks

GOAL

- ➔ Classification of biological samples and prediction of clinical outcomes (disease diagnosis)
- ➔ It is not a new subject in the statistical literature, but it is a new area of application, with interesting and new challenges from both the methodological and computational sides (microarray: very large and complex multivariate datasets)
- Tumor classification using gene expression profiling
 - ① Identification of new tumor classes (*unsupervised learning*)
 - ② Classification of malignancies into known classes (*supervised learning*)
 - ③ Identification of genes that characterize classes (*feature selection*)
- Current methods for classification:
 - Essential for successful diagnosis and treatment of cancer
 - Clinical, morphological and molecular variables
 - Uncertainties still exist in diagnosis
 - Existing classes may be heterogeneous (molecularly distinct diseases)

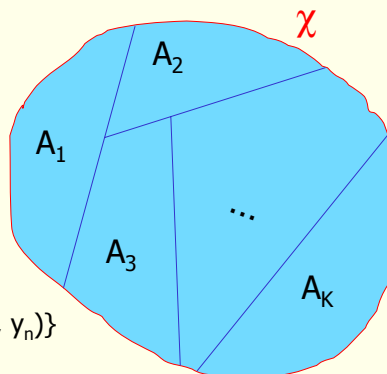
CLASSIFICATION

- K predefined (unordered) classes (c_1, c_2, \dots, c_K)
- Each object: Response variable $Y \in \{1, 2, \dots, K\}$
 Predictor variables $\mathbf{X} = (X_1, X_2, \dots, X_G) \in \chi$

➔ Classifier (Predictor):

$C: \chi \rightarrow \{1, 2, \dots, K\}$

$\mathbf{x} = (x_1, x_2, \dots, x_G) \in A_K \rightarrow \hat{y} = k$



➔ Learning set (LS): $\mathcal{L} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$

$C(\cdot; \mathcal{L})$: Classifier built (trained) from past experience

Gene expression data from DNA microarray experiment (Cancer data; 3 classes)

	individual	Diagnostic	Gene 1	Gene 2	...	Gene G
{	1	A	x_{11}	x_{21}	...	x_{G1}
	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
{	n_1	A	x_{1n1}	x_{2n1}	...	x_{Gn1}
{	$n_1 + 1$	B	x_{11}	x_{21}	...	x_{G1}
	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
{	$n_1 + n_2$	B	x_{1n2}	x_{2n2}	...	x_{Gn2}
{	$n_1 + n_2 + 1$	C	x_{11}	x_{21}	...	x_{G1}
	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
{	$n_1 + n_2 + n_3$	C	x_{1n3}	x_{2n3}	...	x_{Gn3}

➔ $\mathbf{X} = (x_{gi}), g = 1, \dots, G$ and $i = 1, \dots, n$

- Expression levels (x_{gi}): highly processed data

{ absolute values (e.g. oligonucleotide)
 relative values (reference cDNA array)

“small n, large p paradigm”

{ $n \cong 100$
 $p \cong 5,000 - 10,000$

Classification as a Statistical Decision Theory Problem

- $\pi_k = \Pr(Y = k)$: Proportion of objects of class k in the population.
- $p_k(\mathbf{x}) = p(\mathbf{x} | Y = k)$: Conditional density (multivariate distribution).
- ➔ **Loss Function $L(h, l)$** : Loss incurred if an object of class h is erroneously classified as belonging to class l .
- ➔ **Risk Function $R(L)$** : Expected loss when C (classifier) is used to classify.

$$R(C) = E[L(Y, C(\mathbf{X}))] = \sum_k E[L(Y, C(\mathbf{X})) | Y = k] \pi_k$$

$$= \sum_k \int L(Y, C(\mathbf{X})) p_k(\mathbf{x}) \pi_k d\mathbf{x}$$

- ➔ Typically: $\begin{cases} L(h, h) = 0 \\ L(h, l) = 1, h \neq l \end{cases} \Rightarrow \text{Risk} = \text{misclassification rate}$

$$\Pr[C(\mathbf{X}) \neq Y] = \sum_k \int_{C(\mathbf{X}) \neq k} p_k(\mathbf{x}) \pi_k d\mathbf{x}$$

If π_k and $p_k(\mathbf{x})$ are known, it is possible to define an optimal classifier, which minimizes the risk function.

This situation gives an upper bound on performance of classifiers.

The Bayes Rule:

$$p(k | \mathbf{x}) = \frac{\pi_k p_k(\mathbf{x})}{\sum_l \pi_l p_l(\mathbf{x})}$$

$$C_B(\mathbf{X}) = \arg \max_k p(k | \mathbf{x})$$

- ① The Bayes rule minimizes the risk function or misclassification rate under a symmetric loss function (Bayes Risk)
- ② For a general loss function L , the classification rule that minimizes the risk function is:

$$C_B(\mathbf{X}) = \arg \min_l \sum_{l=1}^K L(h, l) p(h | \mathbf{x})$$

- ③ If $L(h, l) = L_h I(h \neq l)$, the Bayes rule is: $C_B(\mathbf{X}) = \arg \max_k L_k p(k | \mathbf{x})$

Issues:

→ Parametric and nonparametric estimators of $p(k|\mathbf{x})$.

- **Density estimation approach:** $p_k(\mathbf{x})$ and π_k are estimated separately for each class; Bayes theorem is applied to estimate $p(k|\mathbf{x})$.

Examples: { Discriminant analysis (Gaussian)
Learning vector quantization
Bayesian belief networks
Naïve Bayes methods

- **Direct function estimation:** $p(k|\mathbf{x})$ are estimated directly based on function estimation methodology such as regression.

Examples: { Logistic regression
Neural networks
Classification trees
Projection pursuit
Nearest neighbor classifier

Maximum likelihood:

$$\text{ML rule: } C_{\text{ML}}(\mathbf{X}) = \arg \max_k p_k(\mathbf{x})$$

Note: If $\pi_k = 1/K$, ML rule amounts to maximizing the class posterior probabilities $p(k|\mathbf{x})$ (= Bayes rule)

→ Fisher Linear Discriminant Analysis (FLDA) (Barnard, 1935; Fisher, 1936)

- FLDA is based on finding linear combinations $\mathbf{x}\mathbf{a}$ of the $1 \times G$ feature vectors $\mathbf{x} = (x_1, \dots, x_G)$ with large ratios (R) of between-groups to within-groups sums of squares.

Linear combination:

$$\mathbf{a}'\mathbf{X} \rightarrow R = \frac{\mathbf{a}'\mathbf{B}\mathbf{a}}{\mathbf{a}'\mathbf{W}\mathbf{a}} \quad \left\{ \begin{array}{l} \mathbf{a}' \ (1 \times G) \\ \mathbf{X} \ (G \times n) \end{array} \right.$$

where **B** and **W**: $G \times G$ matrices of between and within-groups sum of squares and cross-products.

- Extreme values of R are obtained from the eigenvalues and eigenvectors of $\mathbf{W}^{-1}\mathbf{B}$.

☞ $\mathbf{W}^{-1}\mathbf{B}$ has at most $s = \min(K-1, G)$ nonzero eigenvalues:

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_s$$

with corresponding eigenvectors v_1, v_2, \dots, v_s .

☞ Discriminant variables: $\mathbf{x}v_l$ ($l = 1, 2, \dots, s$)

$$a = v_1 \text{ maximizes } R$$

- Suppose $\mathbf{x} = (x_1, \dots, x_G)$

☞ Squared Euclidian Distance: $d_k^2(\mathbf{x}) = \sum_{l=1}^s [(\mathbf{x} - \bar{\mathbf{x}}_k)v_l]^2$

☞ Predicted class: $C(\mathbf{x}; \mathcal{L}) = \operatorname{argmin}_k d_k(\mathbf{x})$

$\bar{\mathbf{x}}_k = (\bar{x}_{k1}, \dots, \bar{x}_{kG})$
(from learning set \mathcal{L})

Summary of FLDA:

- ① Feature selection or dimensionality reduction (s discriminant variables)
- ② Classification (based on distances from class means)

➔ FLDA is a nonparametric method; parametric settings also available

Quadratic
Discriminant
Analysis
(QDA)

$$\mathbf{X} | y = k \sim N(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

$$C(\mathbf{x}) = \operatorname{argmin}_k \left\{ \underbrace{(\mathbf{x} - \boldsymbol{\mu}_k) \boldsymbol{\Sigma}_k^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)'}_{\text{Squared Mahalanobis distance}} + \log |\boldsymbol{\Sigma}_k| - 2 \log \pi_k \right\}$$

Squared Mahalanobis distance

- When $\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$ (same covariance matrix):

$$C(\mathbf{x}) = \operatorname{argmin}_k (\mathbf{x} - \boldsymbol{\mu}_k) \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}_k)' = \operatorname{argmin}_k (\boldsymbol{\mu}_k \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k' - 2\mathbf{x} \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k')$$

Linear Discriminant Analysis (LDA)

Others:

- When $\Sigma_k = \mathbf{D}_k = \text{diag}(\sigma_{k1}^2, \dots, \sigma_{kG}^2)$:

$$C(\mathbf{x}) = \arg \min_k \sum_{g=1}^G \left\{ \frac{(x_g - \mu_{kg})^2}{\sigma_{kg}^2} + \log \sigma_{kg}^2 \right\}$$

Diagonal Quadratic Discriminant Analysis (DQDA)

- When $\Sigma_k = \mathbf{D} = \text{diag}(\sigma_1^2, \dots, \sigma_G^2)$:

$$C(\mathbf{x}) = \arg \min_k \sum_{g=1}^G \frac{(x_g - \mu_{kg})^2}{\sigma_g^2}$$

Diagonal Linear Discriminant Analysis (DLDA)

- Simplest case: $\Sigma_k = I_G$ Observations classified based on their Euclidian distance from class means

➔ Linear Discriminant Methods:

• Advantages:

- ① Simple and intuitive rule
- ② Estimated Bayes rule
(for Gaussian class conditional densities and constant π_k)
- ③ Easy to implement
- ④ Good performance in practice

• Limitations:

- ① Linear or even quadratic discriminant boundaries may not be flexible enough
- ② Features may have mixture distributions within classes
- ③ For large number of features, performance may degrade rapidly
(due to over-parameterization and high variance parameter estimators)

➔ Some Issues in Classification

- ① Feature selection
 - ② Standardization
 - ③ Distance function
 - ④ Imputation of missing data
 - ⑤ Performance assessment
-

• Feature Selection

- ☞ Very important in microarray datasets; large number of genes are likely to be uninformative
- ☞ **Explicitly** (prior to building classifiers); filtering
- ☞ **Implicitly** (part of the classifier building procedure); wrapper methods (machine learning)

• Standardization

- ☞ Transformation of variables and/or observations (location and scale transformations)
- ☞ Choice of transformation and distance function should be made jointly.

Example:
$$x_{gi}^* = \frac{x_{gi} - \bar{x}_g}{s_g} \quad (\text{unit-less variables})$$

- ☞ With microarray data:

Standardization of observations (array): **Normalization step**
(arrays have mean zero and variance 1; loess also)

• **Distance Functions** (Source: Dudoit and Fridlyand, 2003)

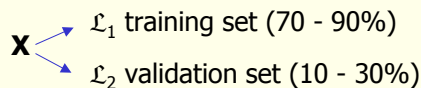
Name	Formula
Euclidean metric	$d_E(\mathbf{x}_i, \mathbf{x}_j) = \left\{ \sum_g w_g (x_{gi} - x_{gj})^2 \right\}^{1/2}$
Unstandardized	$w_g = 1$
Standardized by s.d.	$w_g = 1/s_g^2$
Standardized by range	$w_g = 1/R_g^2$
	Unstand. Euclidean metric: $S = I_G$
Mahalanobis metric	$d_{Ml}(\mathbf{x}_i, \mathbf{x}_j) = \left\{ (\mathbf{x}_i - \mathbf{x}_j) S^{-1} (\mathbf{x}_i - \mathbf{x}_j)^T \right\}^{1/2}$
Manhattan metric	$d_{Mn}(\mathbf{x}_i, \mathbf{x}_j) = \sum_g w_g x_{gi} - x_{gj} $ $\left\{ \begin{array}{l} \lambda=1: \text{Manhattan} \\ \lambda=2: \text{Euclidean} \end{array} \right.$
Minkowsky metric	$d_{Mk}(\mathbf{x}_i, \mathbf{x}_j) = \left\{ \sum_g w_g x_{gi} - x_{gj} ^\lambda \right\}^{1/\lambda}$
Canberra metric	$d_C(\mathbf{x}_i, \mathbf{x}_j) = \sum_g (x_{gi} - x_{gj}) / (x_{gi} + x_{gj}) $
One-minus-Pearson-correlation	$d_{corr}(\mathbf{x}_i, \mathbf{x}_j) = 1 - \frac{\sum_g (x_{gi} - \bar{x}_i)(x_{gj} + \bar{x}_j)}{\left\{ \sum_g (x_{gi} - \bar{x}_i)^2 \sum_g (x_{gj} + \bar{x}_j)^2 \right\}^{1/2}}$

• **Performance Assessment**

- ☞ Bias, variance, and error rates



- ☞ Resubstitution estimation (same data set is used to build the classifier and to assess its performance)
- ☞ Leave-one-out cross-validations
- ☞ Monte Carlo cross-validation



Limitation: Reduces effective sample size for training purposes (problem in microarray datasets; n is small)

- ☞ Others: Fold cross-validation; Leave-one-out; out-of the bag estimation, etc.

EXAMPLE

(Pomeroy et al., 2002)

➔ Data set:

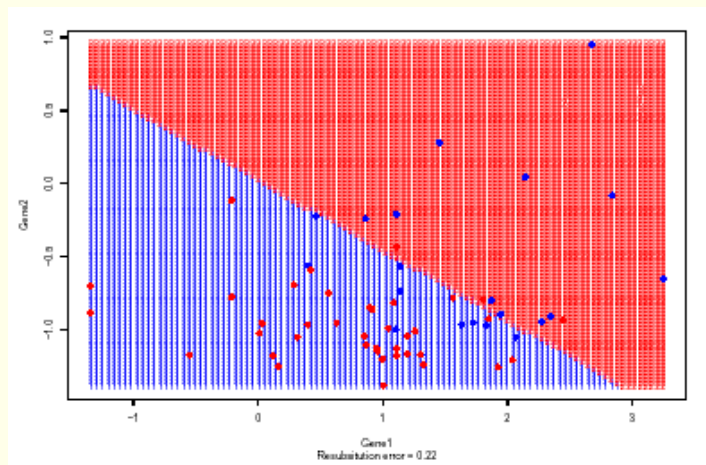
- $n = 60$ medulloblastoma (MD) samples $\left\{ \begin{array}{l} 39 \text{ survivors} \\ 21 \text{ non-survivors} \end{array} \right.$
- Affymetrix: 7,129 human probe sequences (including 5,920 known human sequences + 897 ESTs)
- www-genome.wi.mit.edu/mpr/CNS

➔ Data processing:

- ① **Thresholding:** floor 100; ceiling 16,000
- ② **Filtering:** exclusion of genes $\max/\min \leq 5$ or $\max - \min \leq 500$ (across 60 samples)
- ③ **Logarithm transformation**
- ④ **Standardization:** arrays with mean 0 and variance 1 across genes

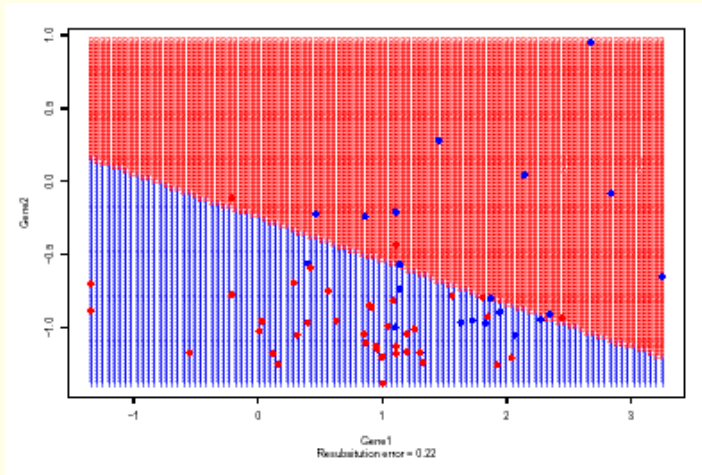
$$4459 \mathbf{X}_{60} = (x_{gi})$$

EXAMPLE: Linear Discriminant Analysis



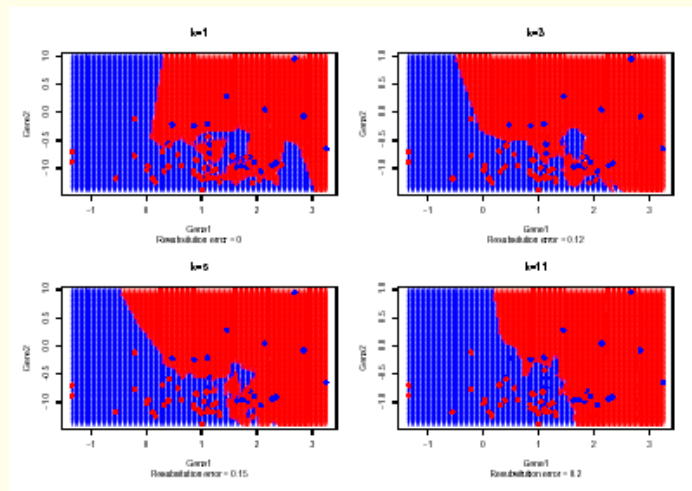
Brain tumor survival data set. LDA partition for the two genes with the largest absolute t-statistics. (Dudoit and Fridlyand, 2003)

EXAMPLE: Quadratic Discriminant Analysis



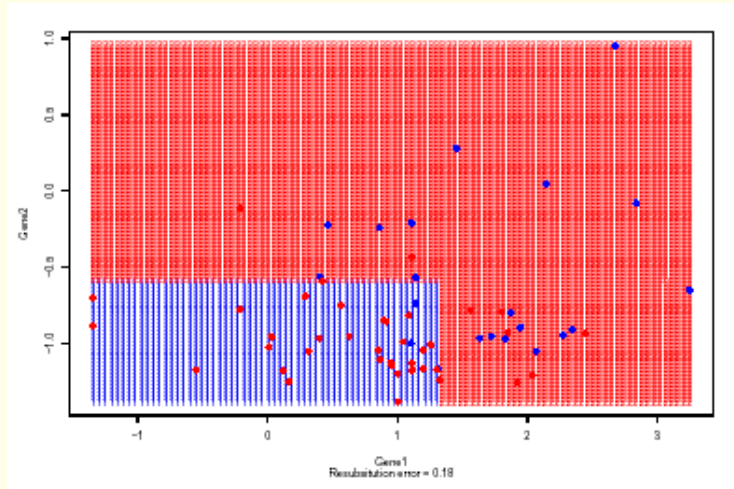
Brain tumor survival data set. QDA partition for the two genes with the largest absolute t-statistics. (Dudoit and Fridlyand, 2003)

EXAMPLE: Nearest Neighbor Classifier



Brain tumor survival data set. k -NN partitions ($k = 1, 3, 5, 11$) for the two genes with the largest absolute t-statistics. (Dudoit and Fridlyand, 2003)

EXAMPLE: Classification Tree



Brain tumor survival data set. CART (10-fold CV) partition for the two genes with the largest absolute t-statistics. (Dudoit and Fridlyand, 2003)

FINAL REMARKS

- ➔ Simple methods such as nearest neighbor and naïve Bayes classification, are competitive with more complex approaches, such as aggregated classification trees or support vector machines (Dudoit and Fridlyand, 2003)
- ➔ Screening of genes to $G = 10$ to 100 is advisable; univariate *vs.* multivariate screenings
- ➔ Models may include other predictor variables (such as age and sex)
- ➔ Outcomes may be continuous (e.g., blood pressure, cholesterol level, etc.)