Extracting Predictive Gene Groups from Microarray Data and Combining them with Clinical Variables

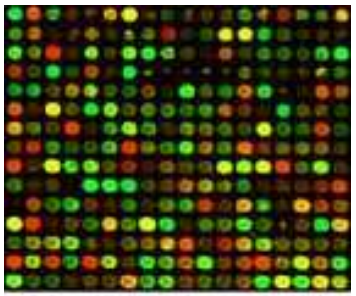
Marcel Dettling
Seminar für Statistik
ETH Zürich, Switzerland

dettling@stat.math.ethz.ch
http://stat.ethz.ch/~dettling

Duke University, Durham
November 10th, 2003

Microarray Gene Expression Data

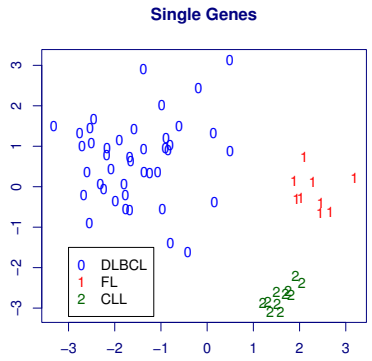
- Microarray technology & preprocessing steps $\rightarrow$ **gene expression matrix**



$$\rightarrow (x_{ij}) = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & & \vdots \\ x_{p1} & x_{p2} & \dots & x_{pn} \end{pmatrix}$$

- n experimental tissues, showing expression levels or activities of p genes.
- Typically between 2'000-20'000 genes, but only 20-200 experiments
- Experiments are grouped into several populations, e.g. different cancer types or tumor classes. This information is given as a categorical response vector $y = (y_1, y_2, \dots, y_n)$

Our Goal: Finding Predictive Gene Groups

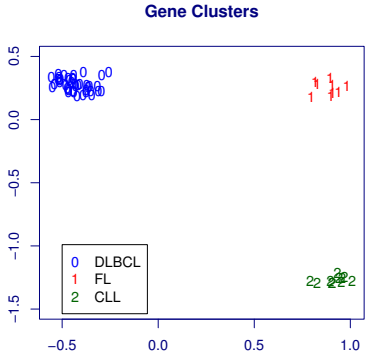


We are searching groups of genes that:

- have moderate size (between 3–50 genes)
- are strongly associated with the response
- allow clear discrimination of populations
- represent functional units of the genome

These groups are possibly useful for:

- accurate prediction in medical diagnostics/prognostics
- gaining insight into biological & regulatory processes
- a huge "statistical" dimensionality reduction



Predictive Gene Grouping: Toy Example

The unsupervised situation:

A patient is suffering from a certain cancer type. Given his alphabetic "gene expression profile" below: Which group of "genes" tells us the cancer type?

1	2	3	4	5	6	7	8	9	10	11	12	13	14
a	b	u	r	e	t	z	e	k	a	s	n	o	t

The supervised situation:

Assume that we know the patient is suffering from **BREAST** cancer. Which group of "genes" contains the information about the cancer type?

1	2	3	4	5	6	7	8	9	10	11	12	13	14
a	**B**	u	**R**	e	t	z	**E**	k	**A**	**S**	n	o	**T**

The group is $\{2, 4, 8, 10, 11, 14\}$. It could have been found unsupervised, but:

$\rightarrow$ Using supervised information about the outcome of the experiments makes it much easier to reveal predictive gene groups

$\rightarrow$ This is what we call supervised clustering

A Brief Overview about Supervised Clustering

Important: Supervised clustering . . .
- is very different from classical, unsupervised hierarchical clustering
- does not primarily find groups of co-expressed genes, but identifies groups providing relevant predictive information

Previous work sharing similarities:

- Supervised Harvesting of Expression Trees (Hastie et al., 2001)
  2-step approach: Unsupervised Clustering plus supervised logistic modeling
- Partial Least Squares (Rocke et al., 2002)
  Predictive meta-components involving all genes, no gene selection
- Supervised Clustering of Genes (MD and Bühlmann, 2002)
  First direct, "fully" supervised 1-step approach
- Simultaneous Gene Clustering and Classification (Jörnsten and Yu, 2003)
  Based on MDL principle, another "fully" supervised 1-step-approach
- Pelora (MD and Bühlmann, 2003)
  Improved 1-step approach, allows useful extensions. To be presented. . .

Supervised Clustering - Theory

Definition:
Supervised Clustering is defined as grouping of variables (genes) by using information from both the X and Y variables, i.e. from both the gene expression data and the response vector

Mathematics:
The data are iid realizations of random variables (X, Y), and we assume that a few marker groups of genes determine a tissue's type.

$$P[Y = 1|X] = f\left(X_{\mathcal{C}_1}, X_{\mathcal{C}_2}, \ldots, X_{\mathcal{C}_q}\right), \qquad q \ll p$$

Representative value:
The centroid is chosen to be the representative value for the groups $\mathcal{C}_1, \ldots, \mathcal{C}_q$

$$X_{\mathcal{C}} = \frac{1}{|\mathcal{C}|} \sum_{g \in \mathcal{C}} X_g$$

Rough Idea for Finding the Gene Groups

Use a statistical optimization criterion, i.e. find the unknown groups by minimizing the log-likelihood (the refinement is shown later. . .)

$$\ell_{x,y}(\mathcal{C}) = -\sum_{j=1}^{n} y_j \log(p_{\mathcal{C}}(x_j)) + (1 - y_j)(\log(1 - p_{\mathcal{C}}(x_j))$$

where $p_{\mathcal{C}}(X) = P[Y = 1|X, \mathcal{C}_1, \ldots, \mathcal{C}_q]$ are model-based conditional probabilities

Why is this difficult?

- Selection and optimization for possibly non-disjoint groups $\mathcal{C}_1, \ldots, \mathcal{C}_q$
  $\rightarrow$ huge combinatorial complexity

- Toy example: In a dataset of 5'000 genes, find 1 cluster of 10 genes
  $\rightarrow$ there are $2 \cdot 10^{30}$ candidates, even a fast PC that checks 10^6 candidates per second still has $3.2 \cdot 10^{16}$ years to finish

- The "real" structure we are looking for is even much more complex

$\rightarrow$ exhaustive search among all partitions of the gene index set is infeasible

Supervised Clustering - Practice (I)

A simplified search heuristic to generate "promising" clustering solutions

Greedy forward algorithm:

1. Start from scratch, check all genes and pick the best single gene

2. The first gene remains fixed. Augment the cluster by checking all genes and picking the one such that the centroid of these 2 genes is optimal

3. The current clustering remains fixed. Grow the cluster in a stepwise fashion by adding one gene after the other, such that the chosen genes lead to optimal centroids

4. If the cluster cannot be improved by adding genes, run a pruning step to remove spurious genes from the cluster

5. If the current cluster can no longer be improved by 3) and 4), terminate it, and start a new cluster

$\rightarrow$ To implement steps 1) $-$ 5), we need a statistical optimization criterion S

Supervised Clustering - Practice (II)

We optimize the ℓ_2-penalized, negative log-likelihood for 2-class problems

$$S_{x,y}(\theta, \mathcal{C}) \;=\; -\sum_{j=1}^{n} \left(y_j \cdot \log p_{\theta,\mathcal{C}}(x_j) + (1 - y_j) \cdot \log(1 - p_{\theta,\mathcal{C}}(x_j)) \right) + \lambda \theta^T P \theta$$

$p_{\theta,\mathcal{C}}(x_j)$ are conditional class probabilities from penalized logistic regression

$$\log \left(\frac{p_{\theta,\mathcal{C}}(x)}{1 - p_{\theta,\mathcal{C}}(x)} \right) = \theta_0 + \sum_{i=1}^{q} \theta_i x_{\mathcal{C}_i}$$

$S_{x,y}(\theta, \mathcal{C})$ measures the gene cluster's ability to group the sample populations.
Minimize this criterion over the parameter vector θ and clustering partition $\mathcal{C}$:

a) Generate a "promising" clustering partition $\mathcal{C}^*$ by our search heuristic

b) Minimize $S_{x,y}(\theta, \mathcal{C}^*)$ by a Newton-like, fast 2-step MLE approximation

Parameter estimation & computing the clustering criterion go hand in hand
$\rightarrow$ reduced computational effort

Remarks to the Algorithm

- We call this algorithm PELORA, as the criterion is based on conditional probabilities from PEnalized LOgistic REgression Analysis

- Pelora has a built-in classifier, working via its class probabilities $p_{\mathcal{C}}(x)$

  $$G(x) = 1_{[p_{\mathcal{C}}(x) > 1/2]}$$

  or with another threshold, depending on the costs of misclassification

- No univariate gene preselection by t-tests or related methods is necessary. The variable selection happens in a multivariate approach

- The penalty parameter λ improves clustering and predictive performance. The choice is not too difficult, as the output depends "smoothly" on λ

- The gene clusters $\mathcal{C}_1, \ldots, \mathcal{C}_q$ can either be overlapping (to capture genes operating in multiple pathways), or disjoint

- The number of clusters q can be chosen by cross validation to optimize prediction, or by other empirical approaches from the literature

Evaluation of the Algorithm

We are requiring: Clusters should . . .

1) have good predictive performance for test data
   $\longrightarrow$ cross validation studies

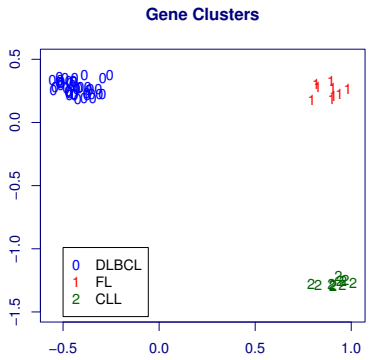
2) be more than random structure
   $\longrightarrow$ check this by permutation tests

3) be relatively stable under slight input variation
   $\longrightarrow$ bootstrapping for assessing stability

4) be biologically relevant (capture true pathways)
   $\longrightarrow$ difficult, needs close collaboration with biologists

An empirical analysis on more than 15 (mostly) publicly available datasets shows "good" results (details follow . . .)

Typical Output, Permutations & Bootstrap

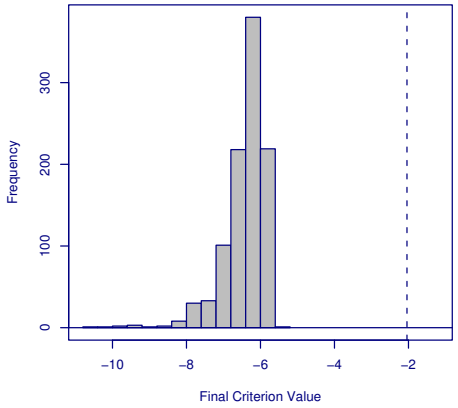


Typical output on more than 15 datasets

- The clusters contain between 15–20 genes
- The populations are very clearly separated
- Error-free classification of training data
- Usually, 10 gene clusters are enough

Permutation test on microarray datasets

- No clearly separating clusters on noise data
- Final criterion on permuted data is always worse
- With p-value 0: clusters are non-random structure
- Bootstrapping: clusters are reasonably stable



Predictive Potential for Test Data

50 splits	Breast	Estro	Nodal	Colon	Prostate	Lymph
Pelora	35.69%	11.50%	27.88%	15.71%	8.94%	0.76%
1-NN	35.77%	15.38%	43.25%	15.90%	12.82%	0.67%
SVM	36.54%	11.12%	36.88%	17.62%	8.35%	0.48%

Prediction scheme:
- Cross validation using 50 random splits into balanced training and test sets
- Supervised clustering & single gene preselection repeated on all training sets

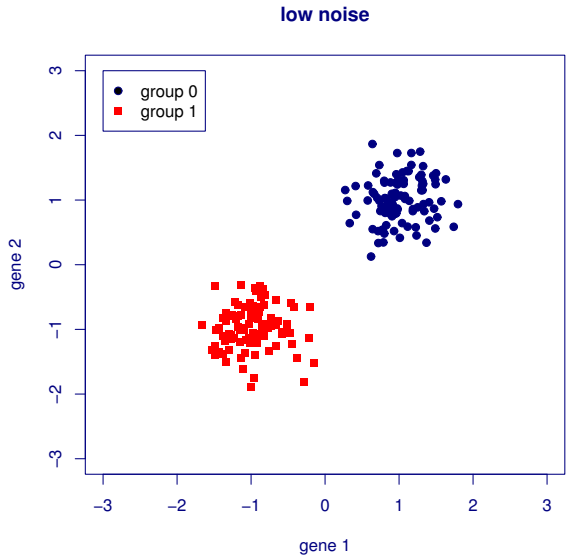
Details about the classifiers:
- Pelora: $q = 10$ clusters for the built-in penalized logistic regression classifier
- 1-NN & SVM: Use the 200 "best" single genes according to t-test statistic

Conclusions:
- Pelora can certainly keep up with (even sophisticated) single gene classifiers
- Better interpretability of the prediction model compared to SVM-like tools
- Improvement mainly on difficult problems with high misclassification risk

Why does Pelora work? (I)

- Consider a single gene g with differential expression in populations 0 and 1
- The expression values arise from a normal distribution, $x_g^{(k)} = \mathcal{N}(\mu_g^{(k)}, \sigma_g^2)$
- Let $\mu_g^{(0)} = -1$ and $\mu_g^{(1)} = 1$ be the expected expression values

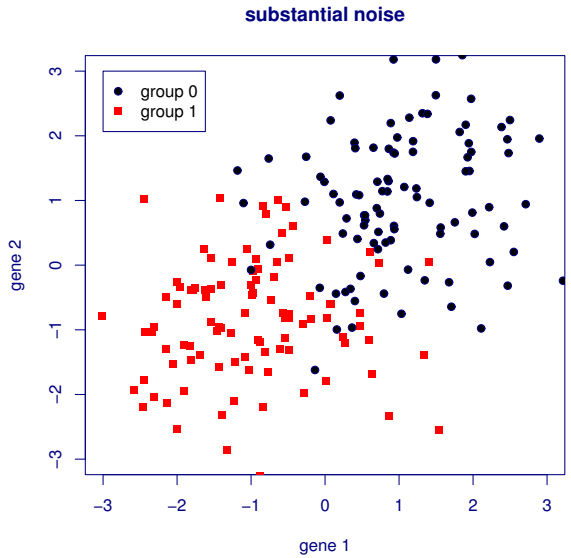
Situation 1: Low noise, $\sigma_g^2 = 1/9$



- $|\mu_g^{(0)} - \mu_g^{(0)}| = 2$ is large compared to the noise level $\sigma_g^2 = 1/9$
- classification is "easy", even with single genes

Why does Pelora work? (II)

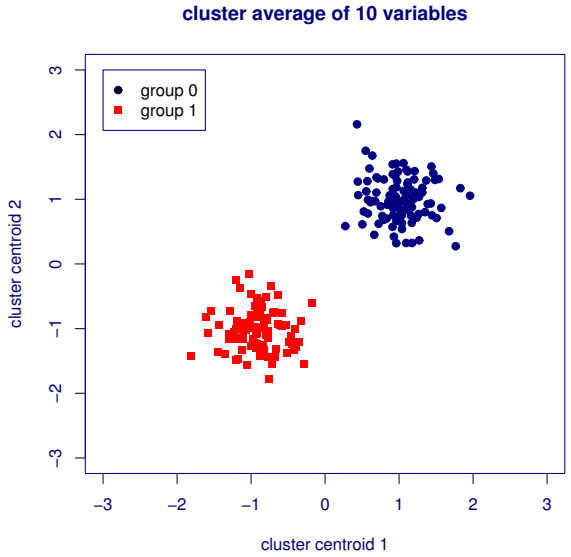
Situation 2: Substantial noise, $\sigma_g^2 = 1$



- $|\mu_g^{(0)} - \mu_g^{(0)}| = 2$ is small compared to the noise level $\sigma_g^2 = 1$
- perfect classification with single genes is more difficult
- **Grouping and averaging of genes can help!!!**

Why does Pelora work? (III)

Situation 3: Substantial noise $\sigma_g^2 = 1$, but using 2 clusters of 10 genes



- Noise level for the clusters drops to $\sigma_{\mathcal{C}}^2 = 1/10 \cdot \sigma_g^2 = 1/10$
- $|\mu_{\mathcal{C}}^{(0)} - \mu_{\mathcal{C}}^{(1)}| = 2$ is again large compared to the noise level $\sigma_{\mathcal{C}}^2 = 1/10$
- The art of the business is picking the right genes in large expression datasets

Why does Pelora work? (IV)

Summary of the last 3 slides:

- Pelora does a sort of multivariate variable selection and grouping
- Averaging of differentially expressed noisy genes reduces the noise variance

Another view:

- Pelora classifies with a sparse, strongly regularized linear logistic model
- Variable selection forces the coefficients for most genes to be zero
- Grouping of the variables causes the coefficients for many genes to be equal
- Yields a model with few parameters, but still using not too few genes

Classification of Rhabdomyosarcoma (RMS)

- In collaboration with the Children's Hospital, University of Zurich
- Rhabdomyosarcoma are the most common soft tissue sarcoma in children
- RMS are subdivided into 2 different histological subtypes: ARMS and ERMS

ARMS	vs.	**ERMS**
– around 30% of the cases		– around 70% of the cases
– has a poor prognosis		– better prognosis
– Subgroups: pARMS/nARMS		– without known subgroups

- Discrimination between nARMS and ERMS is very crucial for successful treatment, but proved to be difficult with microarray data.
- Supervised Clustering with Pelora identified marker groups, that allowed for a classification with more than 93% of accuracy.
- Huge difference in transcriptional activity of cell lines and tumor biopsies
- An analysis of the gene groups yields "some interesting insights".

Risk Groups in Acute Lymphoblastic Leukemia

- In collaboration with the Children's Hospital, University of Zurich
- Currently, ALL-patients are classified into 3 risk groups via "minimal residual disease check" (MRD) after initial treatment

Questions:
1) Is risk assignment possible based on gene expression data?
2) Does it pay off to use microarrays, or is MRD sufficient?

→ If using Pelora, the answer for 1) is "rather yes". For an implementation in daily clinical life, a study with many more patients would be necessary.

→ For answering 2), we need methodology to compare the predictive ability of gene expression data and clinical predictors. This can e.g. be done with Pelora and is presented next...

Combining Expression Data & Clinical Variables

Work in progress - collaboration with Corinne Dahinden

Goal: Combine predictive information from different sources

Two aspects:

1) Build a prediction model incorporating expression data and clinical variables

2) Where does the predictive information come from? Do statistical inference with respect to individual variables or groups of variables

For 1) we require:

- A simple, clear and interpretable prediction model
- Good predictive ability of the combined model
- Possible approaches are:
  - CART (an old classic, not so predictive and rather unstable)
  - Pre-validated microarray predictor (Tibshirani et al., 2002)
  - Forward variable selection and grouping with Pelora. To be presented...

Including Clinical Variables in Pelora

Assume that we are given p-dimensional gene expression data and m clinical variables for all n experiments, such that we have now a random triple

$$(X, U, Y) \in R^p \times R^m \times \{0, 1\}$$

for each experiment. The model which is fitted by Pelora is then

$$logit(P[Y = 1|X, U, \mathcal{C}_1, \ldots, \mathcal{C}_q]) = \theta_0 + \sum_{i=1}^{q} \theta_i x_{\mathcal{C}_i} + \sum_{s=1}^{r} \theta_s u_s$$

Pelora does gene grouping and clinical variable selection simultaneously. Algorithmically, fitting such a combined model works as follows:

- When a new cluster is started, check whether a single gene or a clinical variable can provide the most valuable additional predictive information
  - if it's a gene, choose it and start to build a gene cluster
  - if it's a clinical variable, choose it, proceed & search the next predictor

Important:

- There is no grouping or averaging for clinical variables. Neither among each other, nor with gene expression data
- Pelora performs a selection of clinical variables. Not all of them are incorporated into the final prediction model.

Application to Breast Cancer Data

- Public dataset, published in Nature 2002 by van't Veer et al.
- Goal is discrimination of breast cancer leading (or not leading) to distant metastases within 5 years
- Sample size is $n = 78$, number of genes is $p = 5'408$, $m = 6$ clinical variables are given, i.e.
  - Tumor grade
  - Estrogen receptor status
  - Progesterone receptor status
  - Tumor size
  - Patient age
  - Angioinvasion
- Fitted model on a toy dataset with 1141 manually chosen genes

$$\begin{aligned} logit(P[Y = 1|X, U, \mathcal{C}]) &= \theta_0 + \theta_1(tumor\ grade) + \theta_2 x_{\mathcal{C}_2} + \theta_3(patient\ age) \\ &\quad + \theta_4 x_{\mathcal{C}_4} + \theta_5 x_{\mathcal{C}_5} + \theta_6 x_{\mathcal{C}_6} + \theta_7 x_{\mathcal{C}_7} + \theta_8(angioinv) \\ &\quad + \theta_9 x_{\mathcal{C}_9} + \theta_{10} x_{\mathcal{C}_{10}} \end{aligned}$$

Statistical Inference (I)

Approach 1: Check significance of model coefficients $\theta_0, \ldots, \theta_{10}$

Out-of sample re-estimation of model parameters by bootstrapping yields

$$\widehat{\theta}^{(b)} = (\widehat{\theta}_0^{(b)}, \ldots, \widehat{\theta}_q^{(b)})$$

for each set b. Quantify the significance by inverted $(1-\alpha)$-bootstrap CI's

$$[2 \cdot \theta_j - q_{j,(1-\frac{\alpha}{2})}; 2 \cdot \theta_j - q_{j,\frac{\alpha}{2}}],$$

This results in the following p-values for the 10 predictors chosen by Pelora:

predictor	0	1	2	3	4
variable	intercept	tum. grade	cluster	pat. age	cluster
p-value	0.012	0.000	0.000	0.000	0.136
predictor	5	6	7	8	9
variable	cluster	cluster	cluster	angioinv	cluster
p-value	0.084	0.008	0.146	0.024	0.022

Statistical Inference (II)

Approach 2: Investigate differences in the predictive potential

Estimate the out-of-sample prediction error for classification models $\mathcal{M}_1$ (e.g. Pelora without clinical variables) and $\mathcal{M}_2$ (e.g. Pelora with clinical variables)

$$err(\mathcal{M}) = E_{\mathcal{M}}[1_{[Y_{new} \neq G_{\mathcal{M}}(X_{new})]}]$$

by using k-fold cross validation or leave-one-out bootstrap, yielding $\widehat{err}(\mathcal{M})$. To quantify differences in predictive ability of $\mathcal{M}_1$ and $\mathcal{M}_2$, we need

$$\widehat{Var}(\widehat{err}(\mathcal{M}))$$

Formulas are available, but "open" issues are dependency corrections, simulation results and results on theoretical properties. This is work in progress...

Example: van't Veer breast cancer data with all 5'408 genes
$\mathcal{M}_1$ = fit 10 clusters with Pelora, without clinical variables
$\mathcal{M}_2$ = fit 10 clusters with Pelora, including clinical variables

$\rightsquigarrow$ Leave-one-out CV yields better results with $\mathcal{M}_1$, though the improvement is not significant with a p-value of 0.32.

Conclusions

Pelora . . .
- is a supervised algorithm for selection and grouping of genes
- yields gene clusters, whose centroids try to make the discrimination of different tissue populations as simple as it can be
- can be extended for selecting a combined model with gene expression data and clinical predictor variables

Pelora is (potentially) beneficial for . . .
- medical diagnostics, because it identifies groups of interacting genes with excellent predictive potential, usable as tumor markers
- functional genomics, as its clusters can give a clue on pathways, gene interaction and gene regulation
- quantifying whether the relevant predictive information comes mainly from gene expression data or traditional clinical factors

Outlook & Extensions of Pelora:
- extension to continuous response variables is possible in the same framework

Availability: Software for Pelora is available as R-package, contact

dettling@stat.math.ethz.ch