Supervised clustering of genes

Report

Author(s): Dettling, Marcel; Bühlmann, Peter Lukas

Publication date: 2002

Permanent link: https://doi.org/10.3929/ethz-a-005081613

Rights / license: In Copyright - Non-Commercial Use Permitted
Research

Supervised clustering of genes
Marcel Dettling and Peter Bühlmann

Address: Seminar für Statistik, Eidgenössische Technische Hochschule (ETH) Zürich, 8092 Zürich, Switzerland.

Correspondence: Marcel Dettling. E-mail: dettling@stat.math.ethz.ch

Abstract

Background: We focus on microarray data where experiments monitor gene expression in different tissues and where each experiment is equipped with an additional response variable such as a cancer type. Although the number of measured genes is in the thousands, it is assumed that only a few marker components of gene subsets determine the type of a tissue. Here we present a new method for finding such groups of genes by directly incorporating the response variables into the grouping process, yielding a supervised clustering algorithm for genes.

Results: An empirical study on eight publicly available microarray datasets shows that our algorithm identifies gene clusters with excellent predictive potential, often superior to classification with state-of-the-art methods based on single genes. Permutation tests and bootstrapping provide evidence that the output is reasonably stable and more than a noise artifact.

Conclusions: In contrast to other methods such as hierarchical clustering, our algorithm identifies several gene clusters whose expression levels clearly distinguish the different tissue types. The identification of such gene clusters is potentially useful for medical diagnostics and may at the same time reveal insights into functional genomics.

Background

Microarray technology allows the measurement of expression levels of thousands of genes simultaneously and is expected to contribute significantly to advances in fundamental questions of biology and medicine. We focus on the case where the experiments monitor the gene expression of different tissue samples, and where each experiment is equipped with an additional categorical outcome variable, describing, for example, a cancer type. An important problem in this setting is to study the relation between gene expression and tissue type. While microarrays monitor thousands of genes, it is assumed that only a few underlying marker components of gene subsets account for nearly all of the outcome variation - that is, determine the type of a tissue. The identification of these functional groups is crucial for tissue classification in medical diagnostics, as well as for understanding how the genome as a whole works.

As a first approach, unsupervised clustering techniques have been widely applied to find groups of co-regulated genes on microarray data. Hierarchical clustering [1,2] identifies sets of correlated genes with similar behavior across the experiments, but yields thousands of clusters in a tree-like structure. This makes the identification of functional groups very difficult. In contrast, self-organizing maps [3] require a prespecified number and an initial spatial structure of clusters, but this may be hard to come up with in real problems. These drawbacks were improved
by a novel graph theoretical clustering algorithm [4], but as with all other unsupervised techniques, it usually fails to reveal functional groups of genes that are of special interest in tissue classification. This is because genes are clustered by similarity only, without using any information about the experiment’s response variables.

We focus here on supervised clustering, defined as grouping of variables (genes), controlled by information about the \( Y \) variables, that is, the tumor types of the tissues. Previous work in this field encompasses tree harvesting [5], a two-step method which consists first of generating numerous candidate groups by unsupervised hierarchical clustering. Then, the average expression profile of each cluster is considered as a potential input variable for a response model and the few gene groups that contain the most useful information for tissue discrimination are identified. Only this second step makes the clustering supervised, as the selection process relies on external information about the tissue types. An interesting supervised clustering approach that directly incorporates the response variables \( Y \) in the grouping process is the partial least squares (PLS) procedure [6,7], a tool often applied in the chemometrics literature, in which a supervised manner constructs weighted linear combinations of genes that have maximal covariance with the outcome. PLS has the drawback that the fitted components involve all (usually thousands of) genes, which makes them very difficult to interpret.

Here we present a promising new method for searching functional groups, each made up of only a few genes whose consensus expression profiles provide useful information for tissue discrimination. Like PLS, it is a one-step approach that directly incorporates the response variables \( Y \) into the grouping process, and is thus an algorithm for supervised clustering of genes. Because of the combinatorial complexity when clustering thousands of genes, we rely on a greedy strategy. It optimizes an empirical objective function that quickly and efficiently measures the cluster’s ability for phenotype discrimination. Inspired by [8], we choose Wilcoxon’s test statistic for two unpaired samples [9], refined by a novel second criterion, the margin function. Our supervised algorithm can be started with or without initial groups of genes, and then clusters genes in a stepwise forward and backward search, as long as their differential expression in terms of our objective function can be improved. This yields clusters typically made up of three to nine genes, whose coherent average expression levels allow perfect discrimination of tissue types. In an empirical study, the clusters show excellent out-of-sample levels allow perfect discrimination of tissue types. In an empirical study, the clusters show excellent out-of-sample discrimination. This is because genes are clustered by similarity only, without using any information about the experiment’s response variables.

We first describe our new algorithm for supervised clustering of gene-expression data and then apply the procedure to eight publicly available microarray datasets and test the results for their predictive potential, stability and relevance.

Results and discussion

Algorithm for supervised clustering of genes

This section presents an algorithm for supervised learning of similarities and interactions among predictor variables for classification in very high dimensional spaces, and hence is predestinated for searching functional groups of genes on microarray expression data.

The partitioning problem

Our basic stochastic model for microarray data equipped with categorical response is given by a random pair

\[
(X, Y) \text{ with values } \mathbb{R}^p \times Y
\]

where \( X \in \mathbb{R}^p \) denotes a log-transformed gene-expression profile of a tissue sample, standardized to mean zero and unit variance. \( Y \) is the associated response variable, taking numeric values in \( Y = \{0, 1, \ldots, K-1\} \). A usual interpretation is that \( Y \) codes for one of \( K \) cancer types. For simplicity, and a concise description of the algorithm, we first assume that \( K = 2 \), so that the response is binary. A generalization of the setting for multicategorical response (\( K > 2 \)) is given below.

To account for the fact that not all \( p \) genes on the chip, but rather a few functional gene subsets, determine nearly all of the outcome variation and thus the type of a tissue, we model the conditional probability as

\[
P[Y = 1|X] = f(X_{C_1}, X_{C_2}, \ldots, X_{C_q}),
\]

where \( f(\cdot) \) is a nonlinear function mapping from \( \mathbb{R}^q \) to \([0, 1]\), \( \{C_1, \ldots, C_q\} \) with \( q \ll p \) are functional groups or clusters of genes which form a disjoint and usually incomplete partition of the index set: \( \bigcup_{i=1}^q C_i \subset \{1, \ldots, p\} \) and \( C_i \cap C_j = \emptyset \), \( i \neq j \).

Finally, \( X_{C_i} \in \mathbb{R} \) denotes a ‘representative’ expression value of gene cluster \( C_i \). There are many possibilities to determine such group values \( X_{C_i} \), but as we would like to shape clusters that contain similar genes, a simple linear combination is an accurate choice (see [5,10]):

\[
X_{C_i} = \frac{1}{|C_i|} \sum_{g \in C_i} \alpha_g X_g \text{ with } \alpha_g \in \{-1,1\}.
\]

Because of the use of log-transformed, mean-centered and standardized expression data, we, as a novel extension, allow the contribution of a particular gene \( g \) to the group value \( X_{C_i} \) also to be given by its ‘sign-flipped’ expression value \(-X_g\).
This means that we treat under- and overexpression symmetrically, and it prevents the differential expression of genes with different polarity (that is, one with low expression for class 0 and the other with low expression for class 1) from canceling out when they are averaged. But even by using such simple cluster expression values as in Equation 2, finding a partition of the index set \( \{1, \ldots, p\} \) into subsets or clusters \( \{C_1, \ldots, C_p\} \) that virtually determine the probability structure is still highly non-trivial and the design of a procedure that reveals the exact partition according to Equation 1 is too ambitious. Thus, we have developed a computationally intensive procedure that approximately solves Equation 1 and empirically yields good results.

**Clustering with scores and margins**

A practical heuristic for gene clustering is the *cluster affinity search technique* (CAST) [4]. Our approach is algorithmically similar and also relies on growing the cluster incrementally by adding one gene after the other. Subsequent cleaning steps help us to remove spurious genes that were incorrectly added to the cluster at earlier stages. As in CAST, we repeat growth and removal until the cluster stabilizes, and then start a new cluster.

The main, and very important, difference is that we do not augment (or shorten) the cluster by the gene that suits best (or least) into the current cluster in terms of an unsupervised similarity measure, but base our strategy for supervised clustering of genes on adding (or removing) the gene that improves the differential expression of the current cluster most, according to an empirical objective function for the representative group values from Equation 2. To be more explicit, we assume now that we are given \( n \) independent and identically distributed realizations

\[
(x_{i1}, y_{i1}), \ldots, (x_{in}, y_{in}), \text{ with } x_j \in \mathbb{R}^p \text{ and } y_j \in \{0,1\},
\]

of the random vector \((X, Y)\), whose expression profiles \(x_j\) are centered to mean zero and scaled to unit variance. The objective function needs to be a quantitative and efficiently computable measure of a cluster’s ability to discriminate the tissues. As we aim for subsets of genes with accurate separation in binary problems, we rely on Wilcoxon’s test statistic for two unpaired samples [9], which has been also applied as a nonparametric rank-based score function for genes in [8].

The score of a single gene \(i\) is computed from its \(n\)-dimensional vector of observed values \( \xi_i = (x_{i1}, \ldots, x_{in}) \),

\[
\text{Score}(\xi_i) = s(\xi_i) = \frac{1}{2} \sum_{j \in N_k} \sum_{l \in N_k} 1_{[y_l = y_j]} \left( x_{il} - x_{lj} \right),
\]

where \( x_{ij} \) is the expression value of gene \(i\) for tissue \(j\) and \(N_k\) represents the set of the \(n_k\) tissues in \(\{1, \ldots, n\}\) being of type \(k \in \{0,1\}\). The score uses information about the type of the tissues and is thus a criterion for supervised clustering. It can be interpreted as counting, for each experiment having response value \(0\), the number of tissues from class 1 that have smaller expression values, and summing up these quantities.

Computing the score for a gene cluster \(C_i\) goes likewise via its observed representative values \( \hat{\xi}_i = (x_{i1}, \ldots, x_{in}) \). Viewing the score as Wilcoxon’s test statistic, it allows the ordering of genes and clusters according to their potential significance for tissue discrimination. If the expression values of a particular gene or cluster yield exact separation of the classes, the expression values for all tissue samples having response \(0\) are uniformly lower than the ones belonging to class \(1\) or vice versa. In the former case, the score function returns its minimal value \( s_{\min} = 0 \), in the latter case the maximum score \( s_{\max} = n_0 n_1 \) is assigned.

We rely on the use of log-transformed, mean-centered and standardized gene-expression data and thus need to prevent the averaging of two discriminatory genes with different polarity (that is, one with low expression for class \(0\) and the other with low expression for class \(1\)) canceling out the differential expression of their mean. Therefore, we aim for low expression values pointing to class \(0\) for all genes, which is achieved by using the sign-flipped expression \( \hat{\xi}_i \) for all genes \(i \in \{1, \ldots, p\}\),

\[
\hat{\xi}_i = \alpha_i \hat{\xi}_i = \begin{cases} (x_{i1}, \ldots, x_{in}), & \text{if } s(\xi_i) \leq s_{\max}/2, \\ (-x_{i1}, \ldots, -x_{in}), & \text{if } s(\xi_i) > s_{\max}/2. \end{cases}
\]

The sign-flip is equivalent to setting \(\alpha_i = -1\) in Equation 2 for all genes that tend to have lower expression values for the tissues of type \(1\) than for tissues of type \(0\). After the sign-flip, the scores of all individual genes \(i\) in the expression matrix are equal to

\[
\hat{s}(\hat{\xi}_i) = \min(s(\xi_i), s_{\max} - s(\xi_i)),
\]

and as all genes now have the same polarity, we can safely average them to compute group expression values. It is important to notice that the biological interpretation is not impeded by the sign-flips. Nevertheless, for interpretative purposes, the information about them should be recorded.

During the clustering process, we typically come across different gene or cluster expression vectors that have equal score (often zero) and hence the same quality according to our objective function. This is due to the discrete range of the score function. To achieve uniqueness in the decisions in which gene or cluster is optimal, we need a refinement of our objective function. We thus introduce the margin function, a continuous and real-valued measure for the strength of tissue discrimination of a sign-flipped gene-expression vector \( \hat{\xi}_i \), where low expression values point towards the tissues of class \(0\),

\[
\text{Margin}(\hat{\xi}_i) = m(\hat{\xi}_i) = \min_{l \in N_0} (x_{il}) - \max_{j \in N_1} (x_{lj}).
\]

where \(N_0, N_1\) and \(x_{ij}\) are as in Equation 4. The margin function is positive if, and only if, the score is zero and \(\hat{\xi}_i\) then
perfectly separates the tissues; otherwise it is negative. It measures the size of the gap between the lowest expression value from tissues with response 1, and the highest gene expression corresponding to class 0. The larger this gap, and hence the value of the margin function, the easier and clearer the discrimination of the two classes. The computation of the margin is again likewise for clusters via $\xi_C$. Whenever various gene or cluster expression profiles have equal scores, their quality is judged by the margin function. Our objective function thus has two components. The score function is regarded as highest priority criterion to achieve uniqueness.

The algorithm

Our clustering algorithm is detailed below.

1. Start with the entire $p \times n$ expression matrix $X$. Its rows are genes, and its columns are observations of two different tissue types, having zero mean and unit variance.

2. Determine the score of every gene $i$, that is, every $n$-dimensional row of observed expression values $\xi_i = (x_{i1},..., x_{in})$ in $X$ as in Equation 4. Flip the sign of each gene expression vector $\xi_i$ that has score $s(\xi_i) > s_{\text{max}}/2$ by multiplying it with (-1),

$$\tilde{\xi}_i = \alpha_i^{*} \xi_i = \begin{cases} 
\xi_i, & \text{if } s(\xi_i) \leq s_{\text{max}}/2, \\
-\xi_i, & \text{if } s(\xi_i) > s_{\text{max}}/2.
\end{cases}$$

This operation changes the score to $s(\tilde{\xi}_i) = \min(s(\xi_i), s_{\text{max}} - s(\xi_i))$.

3. Composition of the starting values

(a) If no initial cluster $C$ is given, identify the gene $i^*$ having the lowest score $s(\tilde{\xi}_i)$. If more than one is found, the gene $i^*$ with the largest margin $m(\tilde{\xi}_i)$ as in Equation 6 is chosen. Set the initial cluster mean $\overline{\xi}_C$ equal to the expression vector $(\tilde{\xi}_i)$ of the chosen gene.

(b) If an initial cluster $C$ is given, average the expression of the genes therein,

$$\overline{\xi}_C = \frac{1}{|C|} \sum_{g \in C} \xi_g = \frac{1}{|C|} \sum_{g \in C} (x_{g1},..., x_{gn}).$$

4. Forward search

Average the current cluster expression profile $\overline{\xi}_C$ with each individual gene $i$,

$$\xi_{C,i} = \frac{1}{|C| + 1} \left( \overline{\xi}_C + \sum_{g \in C} \xi_g \right), \quad i = 1,.., p.$$ 

Identify the winning gene $i^*$ as $\arg\min_i s(\xi_{C,i})$; that is, the gene that leads to the lowest score. If not unique, identify the winning gene $i^*$ as the one that optimizes score and margin; that is, $i^* = \arg\min_i s(\xi_{C,i})$ as well as $i^* = \arg\max_i m(\xi_{C,i})$.

5. Repeat step 4 until the identified gene $i^*$ is no longer accepted to enter the cluster. This is said to happen if the score of the updated cluster expression vector $\xi_{C,i}$ worsens, that is, $s(\xi_{C,i}) > s(\xi_{C,i})$, or if the score remains unchanged and the margin deteriorates, that is, $s(\xi_{C,i}) = s(\xi_{C,i})$ as well as $m(\xi_{C,i}) < m(\xi_{C,i})$.

6. Backward search

Exclude each gene $i$ of the current cluster $C$ separately, and average the expression vectors of the remaining genes,

$$\xi_{C,i} = \frac{1}{|C| - 1} \left( \sum_{g \in C \setminus \{i\}} \xi_g \right), \quad i \in C.$$

Compute score and margin of each $\xi_{C,i}$. Identify (as in step 4) that gene $i^*$ whose exclusion optimizes the score, or if not unique, optimizes score and margin.

7. Repeat step 6 until the exclusion of the identified gene $i^*$ is (according to the formulation in step 5) no longer accepted.

8. Repeat steps 4-7 until the cluster converges and the objective function is optimal.

9. If more than one cluster $C$ is desired, discard the genes in the former clusters from $X$ and restart the algorithm at step 3 with the reduced, sign-flipped expression matrix.

The algorithm begins with the sign-flip operation described in Equation 5 to bring all genes to the same polarity. The clustering process can be started with or without initial gene clusters. If none are given, we start the procedure with the single gene that optimizes the objective function. Otherwise, the representative value of the starting cluster is determined. We then proceed by constructing the cluster incrementally. By searching among all genes, we merge and average the current cluster with one single gene, such that the augmented cluster optimizes our objective function, that is, has the lowest score or (in case of ‘ties’) the largest margin. The merging process is repeated until the objective function can no longer be improved. To remove spurious elements out of the current cluster, we then continue with a backward pruning stage, where genes are excluded step by step so that the objective function is optimized by every single removal. This cleaning stage aims to root out genes that were wrongly added to the cluster before. Accordingly, the forward and backward stages are repeated until the cluster converges, that is, when no further improvement of the objective function by adding or removing single genes is possible.

If one wishes to have more than $q = 1$ cluster for a binary class distinction, the genes forming the first cluster are discarded from the expression matrix, and the clustering process is restarted, again with or without an initial cluster. The algorithm’s computations are feasible for dimensions $p$.
and sample sizes $n$ which are clearly beyond today’s common orders and hence also applicable for microarray experiments in the future. The computing time for searching $q = 5$ clusters in the binary leukemia dataset with $n = 72$ observations and $p = 3,571$ genes on a Linux PC with an Intel Pentium IV 1.6 GHz processor is about 5 seconds only. Software for the supervised clustering algorithm is available free as an R-Package at [11].

In summary, our cluster algorithm is a combination of variable (gene) selection for cluster membership and formation of a new predictor by possible sign-flipping and averaging the gene expressions within a cluster as in Equation 2. The cluster membership is determined with a forward and backward searching technique that optimizes the predictive score and margin criteria in Equations 4 and 6, which both involve the supervised response variables from the data.

**Generalization for multiclass problems**

Here we explain the extension of the supervised clustering algorithm to multicategory ($K > 2$) problems, where the response comprises more than two tissue types. We recommend comparing each response class separately against all other classes. This one-against-all approach for reduction to $K$ binary problems is very popular in the machine-learning community, as many algorithms are solely designed for binary response. It works by defining

$$Y^{(k)} = \begin{cases} 1, & \text{if } Y = k, \\ 0, & \text{else} \end{cases}$$

and running $K$ times the supervised clustering algorithm on $(x_n y_1^{(k)}, \ldots, x_n y_n^{(k)})$ as explained above. The interpretation is that we, as in Equation 1, model the conditional probability for discrimination of the $k$th class versus all the other classes.

**Numerical results**

**Data**

**Leukemia dataset.** This dataset contains gene expression levels of $n = 72$ patients either suffering from acute lymphoblastic leukemia (ALL, 47 cases) or acute myeloid leukemia (AML, 25 cases) and was obtained from Affymetrix oligonucleotide microarrays. For more information see [14]; the data are available at [15]. Following exactly the protocol in [16], we preprocess the data by thresholding, filtering, a logarithmic transformation, and standardization, so that they finally comprise the expression values of $p = 3,571$ genes.

**Breast cancer dataset.** This dataset, described in [17], monitors $p = 7,129$ genes in 49 breast tumor samples. The data were obtained by applying the Affymetrix technology and are available at [18]. We thresholded the raw data with a floor of 100 and a ceiling of 16,000 before applying a base 10 logarithmic transformation. Finally, each experiment was standardized to zero mean and unit variance. The response variable describes the status of the estrogen receptor (ER). According to [17], two samples failed to hybridize correctly and were excluded from their analysis. In five cases, two different clinical tests for determination of the ER status yielded conflicting results. These five plus another four randomly chosen samples were also separated from the rest of the data, so that a dataset of $n = 38$ samples remained, of which 18 were ER-positive and 20 ER-negative.

**Colon cancer dataset.** In this dataset, expression levels of 40 tumor and 22 normal colon tissues for 6,500 human genes are measured using the Affymetrix technology. A selection of 2,000 genes with highest minimal intensity across the samples has been made in [19]. The data are available at [20]. As for all other datasets, we process these data further by carrying out a base 10 logarithmic transformation and standardizing each tissue sample to zero mean and unit variance across the genes.

**Prostate cancer dataset.** The raw data are available at [15] and comprise the expression of 52 prostate tumors and 50 non-tumor prostate samples, obtained using the Affymetrix technology. We use normalized and thresholded data as described in [21]. We also excluded genes whose expression varied less than fivefold relatively, or less than 500 units absolutely, between the samples, leaving us with the expression of $p = 6,033$ genes. Finally, we applied a base 10 logarithmic transformation and standardized each experiment to zero mean and unit variance across the genes.

**SRBCT dataset.** This was described in [22] and contains gene-expression profiles for classifying small round blue-cell tumors of childhood (SRBCT) into four classes (neuroblastoma, rhabdomyosarcoma, non-Hodgkin lymphoma, Ewing family of tumors) and was obtained from cDNA microarrays. A training set comprising 63 SRBCT tissues, as well as a test set consisting of 20 SRBCT and 5 non-SRBCT samples are
available at [23]. Each tissue sample is associated with a thoroughly preprocessed expression profile of $p = 2,308$ genes, already standardized to zero mean and unit variance across genes.

**Lymphoma dataset.** This dataset is available at [24] and contains gene-expression levels of the $K = 3$ most prevalent adult lymphoid malignancies: 42 samples of diffuse large B-cell lymphoma (DLBCL, class 0), 9 observations of follicular lymphoma (FL, class 1), and 11 cases of chronic lymphocytic leukemia (CLL, class 2). The total sample size is $n = 62$, and the expression of $p = 4,026$ well-measured genes, preferentially expressed in lymphoid cells or with known immunological or oncological importance is documented. More information on these data can be found in [25]. We imputed missing values and standardized the data as described in [16].

**Brain tumor dataset.** This dataset, presented in [26], contains $n = 42$ microarray gene expression profiles from $K = 5$ different tumors of the central nervous system, that is, 10 medulloblastomas, 10 malignant gliomas, 8 atypical teratoid/rhabdoid tumors (AT/RTs), 8 primitive neuro-ectodermal tumors (PNETs) and 4 human cerebella. The raw data were originated using the Affymetrix technology and are publicly available at [15]. For data preprocessing, we followed the protocol in the supplementary information to [26]. After thresholding, filtering, a logarithmic transformation and standardization of each experiment to zero mean and unit variance, a dataset comprising $p = 5,597$ genes remained.

**National Cancer Institute (NCI) dataset.** This comprises gene-expression levels of $p = 5,244$ genes for $n = 61$ human tumor cell lines which can be divided in $K = 8$ classes: seven breast, five CNS, seven colon, six leukemia, eight melanoma, nine non-small-cell lung carcinoma, six ovarian and nine renal tumors. A more detailed description of the data can be found at [27] and in [28]. We work with preprocessed data as in [16].

**Results from the supervised clustering algorithm**

In this section we briefly describe the results obtained by applying the supervised clustering algorithm to the above datasets. Generally, the output of the clustering procedure is very promising. In all eight datasets we analyzed, comprising a total of 24 binary class distinctions, the average cluster expression $x_C$ always perfectly discriminates the two response classes (in multiclass problems, this is one class against the rest). Hence, the scores of all clusters are equal to zero. Moreover, the clusters have strongly positive margins, indicating that the different tissue types are clearly separated. As an example, Figure 1 shows impressively how well the average cluster expression vectors $x_C^{(1)}$ and $x_C^{(2)}$ discriminate between the three response classes of the lymphoma dataset. It is intuitively clear from Figure 1 that our cluster expression vectors $x_C$ are very suitable as predictor variables for the tissue types and they indeed allow for error-free classification on the training data and also yield good results on independent test datasets.

**Permutation test**

This section is concerned with assessing relevance and addresses the question of whether or not the promising output of the clustering procedure is a noise artifact. For this purpose, we explore quality measures of clusters generated from random-noise gene-expression data and compare them to the results obtained with the original data. As the distributions of the score function $s(\cdot)$ and the margin function $m(\cdot)$ on noise are not known, we rely on simulations. Let $(y_1, \ldots, y_n)$ be the original set of responses. Then,

$$(y_1^{*l}, \ldots, y_n^{*l})$$

is a ‘shuffled’ set of responses, constructed from the original response set by a random permutation for each $l = 1, \ldots, L$. We then allocate an element of the permuted response to each of the (fixed) gene-expression profiles $x_l$ giving us independent and identically distributed pairs

$$(x_l, y_1^{*l}), (x_l, y_2^{*l}), \ldots, (x_l, y_n^{*l}) \text{ for each } l = 1, \ldots, L$$

as in Equation 3. The supervised clustering procedure is then applied $L = 1,000$ times on such data with randomly permuted responses. For every permuted set of responses, a
single cluster \((q = 1)\) was formed on the entire dataset and both its final score \(s^{(0)}\) and margin \(m^{(0)}\) were recorded (Tables 1,2).

We explored the empirical distribution of the scores and margins from permuted data to judge whether the clusters found on the original datasets are of better quality than we would expect by chance. The results given in Figure 2 and in Tables 1 and 2 for a representative selection of data (see the legend to Table 1 for details of data selection) are very satisfactory. As outlined above, the scores \(s^{(0)}\) on the original datasets altogether are equal to zero, with clearly positive margins \(m^{(0)}\). The parameters on the randomly permuted data are worse: the final score \(s^{(0)}\) reached the minimal value of zero in 11% to 98% of the shuffling trials in different datasets (for example, 41% in Figure 2). These frequencies represent a non-significant result in our permutation test for the score function. However, this is not very troubling, as the final margins \(m^{(0)}\) for the permuted data were at best slightly positive, not indicating a clear separation of the randomly shuffled response classes. Values in the range of the margin in the original data were never achieved with any of the permuted data. This corresponds to a \(p\)-value of zero in the permutation test for our entire objective function consisting of score and margin. We thus can surely reject the hypothesis that the clusters found on the original data by our supervised algorithm are irrelevant and just a noise artifact. Moreover, we observed that the clusters from permuted data were much larger in size, clearly exceeding the typical size of between three to nine genes from non-permuted data. For example, permuted data gave a mean cluster size of 12.5 genes and a standard deviation (SD) of 3.2 for the AML/ALL distinction on the leukemia dataset.

The fact that the score has highly non-significant \(p\)-values is at first sight surprising. The reason for this is that the cluster expression values \(x_{Cj}\) in Equation 2 are highly dependent among the samples \(j = 1,\ldots,n\) via the responses \(y_j\) in the supervisedly estimated cluster \(C = C(y_{1},\ldots,y_{n})\) and the sign coefficients \(\alpha_{g}\). This strong interdependence causes the unusual phenomenon that the null-distribution, assuming no association between the expression values \(X\) and the response \(Y\), has a substantial probability to score zero. The margin statistics in Equation 6 has much better power properties than the score.

### Predictive potential

In this section, we will evaluate the predictive potential of the supervised clustering algorithm’s output to see if it could successfully reveal functional groups of genes. A predictor or classifier for \(K\) different tissue types is a function \(C(\cdot)\) that assigns a class label \(\hat{y}\), based on an observed feature vector \(x\). More precisely, the classification rule here will be based on average cluster expression values \(x = (x_{C1}^{q_1},\ldots,x_{C_{K}^{q_{K}}}^{q_{K}})\) as \(K \cdot q\) features

\[
\hat{y} = C(x) = C(x_{C1}^{q_1},\ldots,x_{C_{K}^{q_{K}}}^{q_{K}}) \in \{0,\ldots,K-1\}
\]

In practice, the classifier is built from a learning set of tissues whose class labels are known. Subsequently it can be used to predict the class labels of new tissues with unknown outcome. There are various methods to build classification rules based on past experience and we restrict here on two relatively simple methods that are well suited for our purpose.

### Nearest-neighbor classification

An easy to implement and, compared to more sophisticated methods, impressively competitive classifier for microarray data is the \(k\)-nearest-neighbor rule [29]. It is based on a distance function \(d(\cdot,\cdot)\)

| Margin statistics |
|-------------------|
| Margins \(m^{(0)}\) from the original datasets, as well as maximal, median and minimal margins \(m^{(0)}\) from 1,000 permuted replicates , for leukemia data (AML/ALL distinction), breast cancer data (ER-positive/ER-negative distinction), prostate data (tumor/normal distinction), colon data (tumor/normal distinction), SRBCT data (distinction of the Ewing family of tumors versus three other tumor types), lymphoma data (distinction of DLBCL versus FL and CLL), brain tumor data (separation of atypical teratoid/rhabdoid tumors (AT/RTs) against 4 other tumor types) and NCI data (distinction of leukemia against seven other cancers). |

| Scores |
|-------|
| \(s^{(0)}\) \(\min(s^{(0)})\) \(\max(s^{(0)})\) \(\text{Number of } (s^{(0)} = 0)/L\) |
| Leukemia | 0 | 0 | 279 | 0.41 |
| Breast Cancer | 0 | 0 | 43 | 0.91 |
| Prostate | 0 | 0 | 566 | 0.17 |
| Colon | 0 | 0 | 164 | 0.11 |
| SRBCT | 0 | 0 | 148 | 0.26 |
| Lymphoma | 0 | 0 | 78 | 0.67 |
| Brain | 0 | 0 | 11 | 0.98 |
| NCI | 0 | 0 | 13 | 0.95 |

Scores \(s^{(0)}\) from the original dataset, maximal and minimal scores \(s^{(0)}\) from \(L = 1,000\) permuted replicates, and proportion of shuffled bootstrap trials where score 0 was achieved. The selection of data was as in Table 1.
for pairs \( \mathbf{x} \) and \( \mathbf{x}' \) of feature vectors. As we consider standardized gene-expression data here, the Euclidean distance function

\[
d(\mathbf{x}, \mathbf{x}') = \sqrt{\sum_{i=1}^{K \cdot q} (x_i - x'_i)^2}
\]

is a reasonable choice. Then, for each new feature vector, the \( k \) closest feature vectors from the tissues in the learning data are identified and the predicted class is given by majority vote of the associated responses of these \( k \) closest neighbors. We found a choice of \( k = 1 \) neighbors to be appropriate, but more data-driven approaches via cross-validation for the determination of \( k \) would be possible.

**Aggregated trees.** Another approach that proved to be very fruitful in our setting is as follows: When knowing conditional probabilities \( p_k(\mathbf{x}) = P[Y^{(k)}=1|\mathbf{X} = \mathbf{x}] \), which specify how likely it is that a tissue with feature vector \( \mathbf{x} \) belongs to the \( k \)th or one of the other classes, the classifier function is

\[
\hat{y} = C(\mathbf{x}) = \arg \max_{k \in \{0, \ldots, K-1\}} p_k(x_{C^1}, \ldots, x_{C^q}),
\]

meaning that a tissue is assigned to the class with highest probability. In practice, of course, we have to rely on estimated probabilities \( \hat{p}_k(\mathbf{x}) \), and does not take into account any of the useful information about the remaining \( (q - 1) \) input variables \( x_{C^2}, \ldots, x_{C^q} \). To improve the tree-based probability estimates, we design a novel technique based on plurality voting with classification trees, called aggregated trees. The idea is to fit \( q \) trees, one each with the \( q \) cluster expression profiles (components of the feature vector \( \mathbf{x} \)) that have been found by our supervised algorithm for a particular binary class distinction. Each tree casts a weighted vote \( \hat{p}_{ki}(x_{Ci}) \), \( i = 1, \ldots, q \), for response class \( k \) against the rest. Averaging then yields

\[
\hat{p}_k(\mathbf{x}) = \frac{1}{q} \sum_{i=1}^{q} \hat{p}_{ki}(x_{Ci}),
\]

as estimated conditional probabilities, which can be plugged into Equation 7 for maximum-likelihood classification.

**Empirical study.** Because, except for the leukemia and SRBCT data, no genuine test sets are available, our empirical study for exploring the classification potential is based on random divisions into learning and test set as well as leave-one-out cross-validation. For the latter, we set aside the \( i \)th tissue and carry out cluster identification and classifier fitting by considering only the remaining \( (n - 1) \) data points. We then honestly predict \( \hat{y}_i \), the class label of the \( i \)th tissue sample and repeat this process for all data we have. Each observation is held out and predicted exactly once. We can determine the test-set error by calculating the fraction of predicted class labels which differ from the true class labels. Results for the nearest-neighbor and the aggregated tree classifier and varying number of clusters \( q \) are given in Table 3.
Aggregated trees 49.18% 47.54% 39.34% 29.51% 21.31% 21.31% 19.67%
Nearest neighbor 40.98% 40.98% 36.07% 29.51% 24.59% 27.87% 26.23%

NCI
Aggregated trees 42.86% 23.81% 21.43% 19.05% 14.29% 11.90% 11.90%
Nearest neighbor 30.95% 23.81% 19.05% 16.67% 19.05% 16.67% 16.67%

Brain
Aggregated trees 3.23% 1.61% 1.61% 1.61% 0.00% 0.00% 0.00%
Nearest neighbor 3.23% 1.61% 1.61% 1.61% 0.00% 0.00% 0.00%

Lymphoma
Aggregated trees 3.17% 0.00% 0.00% 0.00% 0.00% 0.00% 0.00%
Nearest neighbor 0.00% 0.00% 0.00% 0.00% 0.00% 0.00% 0.00%

SRBCT
Aggregated trees 27.42% 29.03% 19.35% 19.35% 16.13% 17.74% 17.74%
Nearest neighbor 27.42% 22.58% 22.58% 19.35% 16.13% 17.74% 19.35%

Colon
Aggregated trees 13.73% 13.73% 6.86% 8.82% 6.86% 5.88% 5.88%
Nearest neighbor 13.73% 7.84% 4.90% 6.86% 4.90% 4.90% 5.88%

We observe that the error estimates obtained from random splitting are on a slightly higher level than the ones from leave-one-out cross-validation. We also see that introducing some redundancy for the discrimination process by using additional clusters, that is, increasing \( q \), yields better performance; but of course, a too large value of \( q \) would exhibit overfitting.

**Comparison with classification using single genes.**

Does the use of averaged cluster expression profiles from our supervised algorithm improve the classification results compared to non-averaged, individual genes? To answer this important question, we also classified our datasets with exactly the same genes that were contained in the clusters, but did not average them. Instead of \( q \) average expression profiles, we then have roughly five times as many single genes as predictor variables. Misclassification rates from repeated random splitting are given in Table 5. We observe that the aggregated tree classifier yields in 54 of 56 cases better results with cluster averages than with individual genes as input. Also the nearest-neighbor classifier is in 43 out of 56 cases better when used in conjunction with clusters than with single genes. Note that since the events are not independent, we cannot use a binomial test for the null hypothesis of equal performance between clusters and single genes. An analysis of score and margin of the individual genes that were used in the clusters shows that most of them are not the strongest individually for predicting the tissue types, that is, they individually often only have mediocre scores and margins, but have very good predictive power as a group. So far, we gained evidence that our algorithm really identifies functional groups of genes whose
average expression level has high explanatory power for the response classes.

Comparison with other studies. We now classify the breast cancer validation sample of [17], which contains four randomly chosen tissues plus five instances where two different clinical tests for determination of the ER status yielded conflicting results. We choose the nearest neighbor method with $q = 3$ clusters to be our classifier for the validation sample, as it had the best predictive potential on the training data. Our predictions, shown in Table 6, always agree with the class label provided on the Proc Natl Acad Sci USA supporting information website [32], which corresponds to the outcome of the immunoblot assay method.

Not only the results on the validation sample are very convincing, but the cross-validation on the $n = 38$ training tissues is also error free. This is different from the results in [17] with precedent feature selection, singular value decomposition and Bayesian binary regression, where 7 of 9 tissues in the validation sample and 36 of 38 tissues in the training sample were accurately predicted. Moreover, our result confirms that the breast cancer expression matrix contains a strong signal for discriminating the ER status.

We next used our method to classify the original 34 test samples in the leukemia dataset. We applied the supervised clustering algorithm on the $n = 38$ training data, where we also fit the best predictor from our random splitting study (aggregated trees with $q = 20$ clusters as input features) as classifier for the independent sample. Our predictions turned out to be error-free, a result which can be directly compared to [14], where 29 of 34 observations were classified correctly by a weighted voting scheme. With support vector machines, results ranging between 30 to 32 correct classifications were reported [33]. Moreover, a full leave-one-out cross-validation on the $n = 38$ training data (results not shown) resulted in perfect classification for various $q$ values; also, the performance for cross-validation on the entire dataset with $n = 72$ observations is competitive, compared, for example, to [34].

The SRBCT data contains an additional test set of 20 SRBCT and 5 non-SRBCT samples. We first classified the 20 SRBCT tissues with the best classifier from the random splitting study on the $n = 63$ training samples, the nearest-neighbor method with $q = 3$ clusters as input. The predictions turned out to be error-free, approving the perfect classification with artificial neural networks and principal components as in

Table 4

| Misclassification rates based on random splitting |
|-----------------------------------------------|
| Leukemia                                      |
| Nearest neighbor $q = 1$                       |
| Aggregated trees $q = 1$                       |
| Breast                                        |
| Nearest neighbor $q = 1$                       |
| Aggregated trees $q = 1$                       |
| Prostate                                      |
| Nearest neighbor $q = 1$                       |
| Aggregated trees $q = 1$                       |
| Colon                                         |
| Nearest neighbor $q = 1$                       |
| Aggregated trees $q = 1$                       |
| SRBCT                                         |
| Nearest neighbor $q = 1$                       |
| Aggregated trees $q = 1$                       |
| Lymphoma                                      |
| Nearest neighbor $q = 1$                       |
| Aggregated trees $q = 1$                       |
| Brain                                         |
| Nearest neighbor $q = 1$                       |
| Aggregated trees $q = 1$                       |
| NCI                                           |
| Nearest neighbor $q = 1$                       |
| Aggregated trees $q = 1$                       |

Misclassification rates for out-of-sample classification with $q$ gene clusters as features, based on $N = 100$ random divisions into learning set (two thirds of the data) and test set (one third of the data).
Table 5

Benchmark misclassification rates

| Cancer Type | q = 1 | q = 2 | q = 3 | q = 5 | q = 10 | q = 15 | q = 20 |
|-------------|-------|-------|-------|-------|--------|--------|--------|
| Leukemia    | 6.33% | 4.79% | 4.50% | 4.08% | 3.67%  | 3.75%  | 3.79%  |
| Aggregated trees | 8.50% | 6.04% | 4.54% | 3.92% | 4.83%  | 6.79%  | 8.46%  |
| Breast      | 1.08% | 0.83% | 0.92% | 1.17% | 1.33%  | 1.50%  | 1.58%  |
| Aggregated trees | 5.42% | 2.50% | 1.83% | 2.42% | 4.17%  | 5.42%  | 8.33%  |
| Prostate    | 13.24%| 10.68%| 9.15% | 8.44% | 7.76%  | 8.18%  | 7.85%  |
| Aggregated trees | 25.47%| 21.29%| 18.56%| 17.44%| 16.65% | 17.65% | 18.94% |
| Colon       | 23.40%| 21.95%| 20.15%| 18.90%| 16.65% | 16.25% | 15.70% |
| Aggregated trees | 30.95%| 29.70%| 30.20%| 31.20%| 33.55% | 34.15% | 34.90% |
| SRBCT       | 1.76% | 0.86% | 0.81% | 1.05% | 1.19%  | 1.43%  | 1.48%  |
| Aggregated trees | 4.38% | 2.00% | 2.62% | 3.95% | 6.48%  | 6.95%  | 8.43%  |
| Lymphoma    | 2.43% | 2.29% | 1.76% | 1.05% | 0.81%  | 0.81%  | 0.86%  |
| Aggregated trees | 4.38% | 2.81% | 2.10% | 1.00% | 0.81%  | 1.05%  | 1.24%  |
| Brain       | 30.79%| 29.07%| 29.50%| 27.57%| 28.50% | 28.00% | 27.50% |
| Aggregated trees | 40.14%| 35.29%| 34.64%| 33.50%| 34.36% | 34.79% | 35.29% |
| NCI         | 39.63%| 34.89%| 32.84%| 31.95%| 30.66% | 29.74% | 28.95% |
| Aggregated trees | 56.58%| 49.53%| 44.84%| 42.42%| 39.21% | 39.05% | 37.79% |

Benchmark misclassification rates for out-of-sample classification with the very same but non-averaged genes from q clusters as features, based on N = 100 random divisions into learning set (two thirds of the data) and test set (one third of the data).

[22], as well as the correct diagnosis obtained with multiclass-support vector machines in [35]. As aggregated trees and the one-nearest-neighbor classifier with q = 3 clusters as features, based on error rate of 4.90%, which corresponds to five misclassifications in a total of 102 samples. The colon cancer dataset has already been considered by various authors, for example in [34], with classifiers based on single genes such as nearest neighbors and boosting in a cross-validation study. Our method does not clearly improve their results, although it seems to have an edge over them. However, we could not achieve a cross-validation error rate of 9.68%, as reported in [33] with support vector machines. The error rates on the lymphoma, brain tumor and NCI data provide evidence that

The five remaining microarray studies do not contain genuine test sets and we thus compare our error rates from cross-validation and random splitting against the literature. The classification of tumor versus normal prostate tissue has already been considered by various authors, for example in [21]. After precedent feature selection, an accuracy of “greater than 90%” was obtained, a result that can be beaten by our

Table 6

Classification of the breast cancer validation sample

| Tumor | 14 | 31 | 33 | 44 | 45 | 46 | 47 | 48 | 49 |
|-------|----|----|----|----|----|----|----|----|----|
| Status| Neg?| Neg?| Neg?| Neg| Neg| Pos?| Pos?| Pos| Pos |
| Prediction| Neg| Neg| Neg| Neg| Neg| Pos| Pos| Pos| Neg |

The sample is classified with q = 3 cluster expression profiles based on the training sample with 38 tumors as features and aggregated trees as predictor. The status of the tumors is according to the information provided on the Proc NOD Acad Sci USA website [32]. The question mark means that two clinical tests yielded conflicting results. Displayed here is the outcome of the immunoblot assay method.
our method, based on a one-against-all approach, does a good job in multiclass problems as well. On the lymphoma data we observe perfect classification, thus achieve the non-to-improve benchmark. On the brain tumor data, our minimal cross-validation error rate of 11.90% is superior to the 16.67% obtained in [26] with a weighted voting algorithm. Many more misclassifications occur on the NCI than on the other datasets, due to the large number of classes and their heterogeneity. However, when comparing our predictions to the results in a broad evaluation of classifiers on the NCI data [16], they prove to be very valuable. We consistently obtained mean error rates of less than 30% with random splitting, the optimum is 26.50% using aggregated trees with $q = 20$ clusters, whereas the best median error rates reported in [16] are in a range around 35% and higher (Table 7).

In summary, our predictions from simple classifiers based on the supervised clustering’s output can easily keep up with sophisticated methods that are based on single genes, and as Table 7 shows, our supervised clusters beat the best reported results from the literature in four out of eight datasets. On three further datasets, we achieve the benchmark of perfect classification. The success of our method may be because the averaging of genes according to Equation 2 has a variance-reducing effect and yields more stable and accurate features for classification. As well as its good predictive potential, the cluster structure provided by our method is very accessible for biological interpretation and can be beneficial for functional genomics.

**Stability**

The stability of the gene clusters detected by our supervised clustering algorithm is a critical issue. The output is much more useful for functional genomics if it remains unchanged for ‘similar’ input data. We use the bootstrap as a tool for assigning statistical significance, see [36]. We assume $n$ pairs of observations $(x_i, y_i)$ with binary response $y_i \in \{0,1\}$, from which we form a resampled gene expression dataset

$$(x_1, y_1)^*,...,(x_n, y_n)^*$$

of length $n$ by drawing with replacement from the original data pairs. We can then apply our supervised algorithm to extract clusters $C_1^*,...,C_q^*$ out of these resampled data. For an empirical study, we generated $L = 1,000$ resampled gene-expression datasets of size $n$ to explore the compositional variability of the first cluster $C_1^*$ in eight binary problems as detailed in the caption of Table 8.

We first analyze the variability in cluster size. The results, summarized in Table 8, show surprising stability across the eight different datasets. We observe that quite small clusters, typically made up of three to nine genes, were found. The SD in cluster size was fairly low in all eight datasets. As a next, and more difficult, step, we try to explore the compositional variability of the clusters. To give a rough overview which proportion of genes is actively present in the clustering process, we assess a confidence level to each individual gene $i$, which measures how likely it is to be clustered,

$$\pi_i = \frac{N_i}{L} = \frac{1}{L} \sum_{l=1}^{L} 1_{[\text{gene } i \in C_l^*]} \quad i = 1,...,p, \quad (8)$$

where $N_i$ is the number of the $L$ clusters that contain gene $i$. The numerical results given in Table 9 show that except for the colon tumor data, only a minority of genes ever entered a cluster. Also for the prostate and leukemia data this proportion was somewhat bigger, but still most of the genes never took part in the clustering process. More important, only a very small part of the genes is used frequently, that is more than 50 times in the 1,000 clusters. We conjecture that our supervised algorithm discriminates phenotypes with a small core of genes only, and in this sense it is reasonably stable.

We continue by assessing confidence levels to pairs of genes which gives a clue about pairwise interactions. We count the number $N_{ij}$ of clusters $C_l^*$ found with our bootstrapped gene expression datasets that both contain the genes $i$ and $j$, and then divide by the number of replicates $L$,

$$\pi_{ij} = \frac{N_{ij}}{L} = \frac{1}{L} \sum_{l=1}^{L} 1_{[\text{gene } i \in C_l^* \cap \text{gene } j \in C_l^*]} \quad i,j = \{1,...,p\}. \quad (9)$$

These confidence levels not only give an idea how likely the pairs are, but also provide information for functional genomics, as we can now analyze whether pairs of genes preferentially enter clusters simultaneously or not. The number of hits $N_i$ for individual genes $i$ follows a binomial$(L, \pi_i)$ distribution (given the data) , and for pairs

### Table 7

|             | Leukemia | Breast | Prostate | Colon | SRBCT | Lymphoma | Brain   | NCI<sup>≈</sup> |
|-------------|----------|--------|----------|-------|-------|----------|---------|---------------|
| Supervised clustering | 1.39%    | 0.00%  | 4.90%    | 16.13%| 0.00% | 0.00%    | 11.90%  | 26.50%        |
| Literature  | 1.39%    | 5.26%  | 9.80%    | 9.68% | 0.00% | ?        | 16.67%  | ≈35%          |

Best leave-one-out cross validation error-rates from our supervised clustering procedure compared to best reported results from the literature where directly comparable, references are given in the main text. *The mean error-rate on the NCI data is based on random divisions into training and test set, and compared against the median error-rate obtained under the same framework in [16].
joint probability. There were no attraction or repulsion between genes, the independence, we can test the hypothesis with

\[
\sum_{i,j} n_{ij} \left( p_{ij} - p_{i} p_{j} \right)
\]

where \( n_{ij} \) is the marginal counts of hits between the two genes and \( p_{ij} \) is the joint probability of the marginal probabilities. By calibrating the observed number of hits \( n_{ij} \) with the binomial distribution under independence, we can test the hypothesis

\[
H_0: \pi_{ij} = \pi_i \pi_j
\]

and compute the associated \( p \)-values. Low \( p \)-values indicate significant pairs of genes. Moreover, we also distinguish between two genes which are attracting (with \( n_{ij} \) larger than expected under the null hypothesis), and which are repelling (with \( n_{ij} \) lower than expected under \( H_0 \)). We implemented an empirical analysis based on \( L = 1,000 \) bootstrap replicates, for pairs made up of the five genes with the highest confidence levels \( \pi_i \) in the discrimination of lymphoma class 0 (DLBCL) from the other two phenotypes. Numerical results are summarized in Table 10.

Table 8

| Cluster size | Mean  | SD   | Min | Max |
|--------------|-------|------|-----|-----|
| Leukemia     | 5.855 | 2.910| 1   | 23  |
| Breast cancer| 4.344 | 2.062| 1   | 13  |
| Prostate     | 6.327 | 2.373| 2   | 17  |
| Colon        | 6.642 | 2.733| 2   | 20  |
| SRBCT        | 4.739 | 1.816| 1   | 14  |
| Lymphoma     | 5.485 | 2.679| 1   | 16  |
| Brain        | 6.094 | 2.751| 1   | 19  |
| NCI          | 6.174 | 2.930| 1   | 20  |

Table 9

| Number and proportion of genes used in the various clusters |
|-----------------------------------------------------------|
| Active genes    | \( \sum_{i,j} |n_{ij}| \) | \( \sum_{i,j} |n_{ij}| p_{ij} \) | \( \sum_{i,j} |n_{ij}| p_{ij}^2 \) | \( \sum_{i,j} |n_{ij}| p_{ij}^3 \) | \( \sum_{i,j} |n_{ij}| p_{ij}^4 \) |
|-----------------|---------------|----------------|----------------|----------------|----------------|
| Leukemia        | 624           | 17.474%        | 18             | 0.504%         |                |
| Breast cancer   | 128           | 1.803%         | 9              | 0.130%         |                |
| Prostate        | 949           | 15.730%        | 16             | 0.265%         |                |
| Colon           | 1028          | 51.400%        | 12             | 0.600%         |                |
| SRBCT           | 68            | 2.946%         | 11             | 0.477%         |                |
| Lymphoma        | 279           | 6.930%         | 19             | 0.472%         |                |
| Brain           | 345           | 6.164%         | 21             | 0.375%         |                |
| NCI             | 227           | 4.329%         | 23             | 0.439%         |                |

Number and proportion of genes that ever have been used in the first cluster \( C^* \) (first two columns), as well as number and proportion of genes that have been used for cluster \( C^* \) in more than 50 out of the 1000 bootstrap trials (last two columns). The selection of data is identical to Table 8.

It is now tempting to extend this kind of analysis from pairs to tuples of third and higher orders. But estimating higher-order interactions will become very unreliable because of the limited amount of sample size \( n \).

Additional modifications

Our supervised clustering procedure can be understood as a generic method and allows alteration of various details according to the users’ choice and specific demands. We also tried to improve the supervised clustering procedure ourselves with additional modifications, the most important of which are described here. The averaging of the gene expression in Equation 2 is specified by the arithmetic mean plus sign-flips, a very simple linear combination of genes, as it is impracticable to repeatedly optimize a general linear combination such as

\[
\mathbf{X}_{i} = \sum_{g \in C_i} \beta_g \mathbf{x}_g \text{ with } \sum_{g} |\beta_g| = 1
\]

during the clustering process. But theoretically, once the cluster algorithm has done its work, we could try to improve the discriminatory power of the actual cluster by numerically optimizing a weighted linear combination as above with respect to score and margin. In practice, we recognized that the numerical optimization was very difficult. If we started it with equal weights, they only changed slightly, and the objective function (this is, the margin) did not improve much. Because of this we favor the more simple method.

Since the margin function in Equation 6 is not scale-invariant, we also considered clustering with an adjusted margin. This means that we optimized the quotient of margin and within-group variation for a gene-expression vector \( \mathbf{X}_i = (x_{i1}, \ldots, x_{in}) \),

\[
\text{Adjusted margin } (\xi) = \frac{\text{Margin } (\xi)}{\sqrt{s_0^2/n_0 + s_1^2/n_1}}
\]
Our algorithm, as described above, yields disjoint clusters of genes. To account for the fact that genes may function in multiple pathways, one could modify it as follows. First, run the clustering algorithm on the data, producing a first cluster; second, compute a probability estimate for $P[Y = 1|X]$ for a two-class problem, for example, with probability-based classification methods or in a logistic model; third, reweight the data with weights as in the Real AdaBoost algorithm [37]; then return to the first step but now with reweighted data. Doing the loop $q$ times produces $q$ clusters, which are allowed to be non-disjoint.

We also explored the improvement of the supervised clustering algorithm by biasing it towards larger clusters. Specifically, we did not stop the forward search when score and/or margin first worsened, but continued as long as the objective function remained within a factor of the best. Our intention was that the objective function could improve again and reach even better values. As soon as the objective function once dropped below the tolerance (a factor times the best ever achieved value), we stopped the forward search and continued the algorithm with the cluster that yielded the best parameters ever. Although our first guess was that the biasing could result in larger clusters with clearer separation, it rarely ever had any effect in practice.

Conclusions

We have proposed an algorithm for supervised clustering of genes from microarray experiments. Our procedure is potentially useful in the context of medical diagnostics, as it identifies groups of interacting genes that have high explanatory power for given tissue types, and which in turn can be used to accurately predict the class labels of new samples. At the same time, such gene clusters may reveal insights into biological processes and may be valuable for functional genomics.

In summary, our algorithm tries to cluster genes such that the discrimination of different tissue types is as simple as possible. It builds the clusters incrementally and relies on a fast, stepwise strategy that allows exhaustive searches among thousands of genes. More specifically, the aim is to identify sparse linear combinations of genes whose average expression level is uniformly low for one response class and uniformly high for the other class(es).

In empirical studies, the average cluster-expression profiles showed superior classification potential compared to other techniques where unclustered genes had been used. The clusters showed reasonable stability and there are several reasons that point towards their biological significance. They do not only contain the genes that are individually good, but groups of genes whose consensus expression profile is best with respect to the objective function. The predictive potential of the very same, unaveraged genes cannot keep up with the prediction potential of the corresponding cluster means. And, finally, an application of our algorithm to randomly
permuted data shows that the identified structure is more than just a noise artifact.

An important task that remains to be addressed in future research is the generalization of the supervised clustering algorithm to quantitative response variables and to censored survival data. The fundamental idea of supervised clustering can be pursued again, but needs alternative objective functions that rank individual genes and gene clusters on the basis of their explanatory power for non-categorical response variables.

Acknowledgements
We thank Jane Fridlyand for providing the preprocessed NCI data. Software is available at [11].

References
1. Weinstein J, Myers T, O’Connor P, Friend H, Fornace Jr A, Kohn K, Fojo T, Bates S, Rubinstein L, Anderson N, et al.: An information-intensive approach to the molecular pharmacology of cancer. Science 1997, 273:343-349.
2. Eisen M, Spellman P, Brown P, Botstein D: Cluster analysis and display of genome-wide expression patterns. Proc Natl Acad Sci USA 1998, 95:14863-14868.
3. Tamayo P, Slonim D, Mesirov J, Zhu Q, Kitareewan S, Dmitrovsky E, Lander E, Golub T: Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. Proc Natl Acad Sci USA 1999, 96: 2907-2912.
4. Ben-Dor A, Shamir R, Yakhini Z: Clustering gene expression patterns. J Comput Biol 1999, 6:281-297.
5. Hastie T, Tibshirani R, Botstein D, Brown P: Supervised harvesting of expression trees. Genome Biol 2001, 2:research0003.1-0003.12.
6. Nguyen D, Rocke D: Tumor classification by partial least squares using microarray gene expression data. Bioinformatics 2002, 18:39-50.
7. Geladi P, Kowalski B: Partial least squares regression: a tutorial. Analit Chim Acta 1986, 185:1-17.
8. Park P, Pagano M, Bonetti M: A nonparametric scoring algo- rithm for identifying informative genes from microarray data. Proc Symp Biocomput 2001, 652-663.
9. Wilcoxon F: Individual comparisons by ranking methods. Biometrics 1945, 1:80-83.
10. Hastie T, Tibshirani R, Eisen M, Alizadeh A, Levy R, Staudt L, Botstein D, Brown P: Gene shaving as a method of identifying distinct sets of genes with similar expression patterns. Genome Bio 2000, 1:research0003.1-0003.21.
11. Supervised clustering of genes [http://stat.ethz.ch/~dettling/supercluster.html]
12. Hastie T, Tibshirani R: Classification by pairwise coupling. Annls Statist 1998, 26:451-471.
13. Allwein E, Schapire R, Singer Y: Reducing multiclass to binary: a unifying approach for margin classifiers. J Machine Learn Res 2000, 1:13-141.
14. Golub T, Slonim D, Tamayo P, Huard C, Gassenbeek M, Coller H, Loh M, Downing J, Caligiuri M, Bloomfield C, Lander E: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 1999, 286:531-538
15. Whitehead Institute Center for Genomic Research: cancer genomics [http://www-genome.wi.mit.edu/cancer]
16. Dudoit S, Fridlyand J, Speed T: Comparison of discrimination methods for the classification of tumors using gene expression data. J Am Stat Assoc 2002, 97:77-87.
17. West M, Blanchette C, Dressman H, Huang E, Ishida S, Spang R, Zuzan H, Olson J, Marks J, Nevins J: Predicting the clinical status of human breast cancer by using gene expression profiles. Proc Natl Acad Sci USA 2001, 98:11462-11467.
18. Duke DNA Microarray Center [http://mgm.duke.edu/genome/dna_micro/work/]