Overabundance Analysis and Class Discovery in Gene Expression Data[*]

Amir Ben-Dor[†]
Agilent Laboratories
3500 Deer Creek Road, MS 25U-5
Palo Alto, CA 94304
and
University of Washington
amir_ben-dor@agilent.com

Nir Friedman[‡]
School of Computer Science & Engineering
Hebrew University
Jerusalem, 91904, ISRAEL
nir@cs.huji.ac.il

Zohar Yakhini
Agilent Laboratories
MATAM Bdg 30
Haifa 31905, Israel
and
Technion
zohar_yakhini@agilent.com

Abstract

Recent studies (Alizadeh et al. 2000, Bittner et al. 2000, Golub et al. 1999) demonstrate the discovery of disease subtypes from gene expression data. In this paper, we propose a principled and systematic approach to address the computational problem of partitioning the set of sample tissues into statistically meaningful classes. We start by describing a method, called *overabundance analysis*, for assessing how informative a given expression data set is with respect to a partition of the samples. As we show, in several published expression datasets, an overabundance of genes separating known classes is observed. Then, we use this method as the foundation to a novel approach to *class discovery*. In this approach, we search for partitions that have statistically significant overabundance score. We evaluate the performance of our approach on synthetic data, where we show it can recover planted partitions. Finally, we apply it to several published tumor expression datasets, and show that we find several highly pronounced partitions.

[*] A preliminary version of this paper appeared in *Fifth Annual International Conference on Computational Molecular Biology*, 2001, with the title “Class Discovery in Gene Expression Data”.

[†] Contact author.

[‡] Nir Friedman was supported by ISF grant 244/99, Israeli Ministry of Science grant 2008-1-99, and an Alon fellowship.

1 Introduction

An important application of gene expression profiling technologies, such as array-based hybridization assays, is to study the differences, at the molecular level, between cell types. Such studies collect expression level measurements of thousands of genes in multiple samples. Samples are labeled by properties of their source. For example, (Alon et al. 1999) report a data set comparing gene expression in normal colon tissues to that in colon carcinomas, (Golub et al. 1999, Alizadeh et al. 2000, Bittner et al. 2000) report data sets of tissues from different types of leukemia, lymphoma and melanoma, respectively.

Two cell types with dramatically different biological characteristics (e.g., a normal cells and tumor cells from the same tissue) are expected to also have different gene expression profiles. It is important, however, to realize that the majority of the active cellular mRNA is not effected by these differences. That is, a dramatic biological difference does have a gene expression level manifestation, but the set of genes that is involved can be rather small. Thus, most of the genes measured in these experiments are irrelevant to the distinction between the cell classes. However, other genes play major roles in the biochemical pathways that underly those distinctions, and thus are differentially expressed in the distinct cell classes. A central task of the data analysis phase is to identify potentially relevant genes based on their expression profiles. For example, Golub et al. (1999) use the *separation score* of a gene (Slonim et al. 2000) to measure whether a gene is differentially expressed in two classes of samples. Ben-Dor, Bruhn, Friedman, Nachman, Schummer & Yakhini (2000), develop the *Threshold Number of Misclassification* ($TNoM$) score to measure differential expression. In (Ben-Dor, Friedman & Yakhini 2000) we introduced the *Conditional Entropy* ($INFO$) score. Both scoring methods are reviewed in details in Section 2.

By evaluating such scores for all genes in a dataset we can order the genes by their relevance and identify the ones that are deemed most relevant to the distinction being studied. This, however, does not suffice. To avoid inaccurate conclusions, it is crucial to assess the *statistical significance* of the observed gene relevance scores. This is usually done by formulating a *null-hypothesis*, which models the situation where the expression profiles are independent of the classification of interest. We then compute the *p-value*, the probability of observing this score (or a better one) under the null-hypothesis. One approach for evaluating p-values is via a *permutation test*, that consists of randomly permuting the tissues labels (i.e., their class membership), and then computing scores under the new partition. Repeating this process allows one to estimate significance levels of different scores. For example, in Slonim et al. (2000), a permutation test was performed 400 times, and the 95% and 99% significance levels (p-values of < 0.05 and < 0.01, respectively) were estimated. Sampling based methods are, however, computationally intensive, and can only examine a limited range of p-values. Naive sampling cannot detect p-values of magnitude 10^{-15}, such as the ones we find in the leukemia data set. Therefore, sampling based methods have limited utility in highlighting and distinguishing statistical significance levels.

Efficient methods for computing *exact* p-values at low orders of magnitude are, therefore, instrumental in analyzing expression data. In Section 3, we develop a closed form formula for the distribution of the $TNoM$ score and an efficient dynamic programming scheme for calculating the exact distribution of the $INFO$ score, under the uniform null model (permutations of the tissue labels are uniformly drawn).

Relevance p-values are important for several aspects of the data analysis task. Many expression datasets contain *missing values*. As a result, each gene is evaluated with respect to a different (and potentially unique) subset of the samples. Thus, it is not possible to directly compare the scores of two genes from the same data set and same partition. However, by comparing the p-values of these scores, we account for the different patterns of missing values for each gene. By the same virtues, p-values provide a common scale for comparing individual gene relevance scores across different datasets, different scoring methods, and different partitions of the same data.

Importantly, as noted above, p-values allow the identification of highly significant genes. Expression differences in genes with extremely low p-values are likely to have biological, mechanistic or protocol reasons. These genes for which the latter two options can be ruled out, are interesting subjects for further investigation and are expected to give deeper insight into the biology of the different cell types.

Once we compute p-values of the scores for all the genes in a data set, we can also examine the *global* pattern that emerges. Instead of searching for few genes with particularly small p-values, we can examine the distribution of p-values for genes in the dataset. Then we can compare the observed distribution of relevance scores in the dataset to the distribution expected under a null model. This allows us to highlight an *overabundance* of informative genes in the dataset, and as a result to assess the global statistical support for the partition of the set of samples. Indeed, as we demonstrate below, several examples of biologically meaningful differences between cell types yield high overabundance of informative genes. Such analysis was instrumental in a melanoma gene expression study reported in (Bittner et al. 2000). The authors applied relevance scores and p-values to statistically validate a putative cutaneous melanoma subtype and to select differentially expressed genes. In Section 4, we propose methods for *quantifying* informative genes overabundance. Such methods enable statistical evaluation of putative classifications of the set of samples.

Overabundance analysis allows us to compare the support of different partitions of the same data set. This provides a statistical foundation for the task of *class discovery*. Recent studies (Alizadeh et al. 2000, Bittner et al. 2000, Golub et al. 1999, Slonim et al. 2000) demonstrate the discovery of putative disease sub-types from gene expression data. Alizadeh *et al* (2000) discover a putative subclass of DLBCL, a type of lymphoma. Bittner *et al* (2000) suggest a putative sub-class of cutaneous melanoma. In both cases the findings were further biologically validated.

Recent examples of class discovery (Alizadeh et al. 2000, Bittner et al. 2000) involved the application of supervised and unsupervised clustering methods. For example, Bittner *et al* (2000) applied several clustering methods on expression profiles of melanoma tumors. One of these methods discovered a classification that was then verified by other means to capture a meaningful distinction in melanoma tumor behaviors. Alizadeh *et al* (2000) used a more complex protocol. They collected experssion profiles of DLBCL (a type of lymphoma) samples, as well as samples of healthy T-Cell and B-Cell in different development stages, and samples of other types of lymphoma. Then they applied an agglomerative clustering over genes to find genes with similar behaviour, across the different types of samples. From the resulting hierarchy they manually selected specific subsets of genes. These were named according to the samples they were active in (e.g., “Germinal Center B-Cell” genes were up-regulated in samples of B-Cells in the germinal center). Finally, they applied clustering to discover a partition of the DLBCL samples, by restricting a sample clustering procedure to examine a particular subset of genes they have selected in this manner.

Such methodology of class discovery suffers from the need for manual intervention. This intervention was required since typical clustering procedures used in gene expression analysis (Alon et al. 1999, Ben-Dor et al. 1999, Eisen et al. 1998, Sharan & Shamir 2000, Tamayo et al. 1999, Tavazoie et al. 1999) attempt to find groups of samples such that the *overall* expression profiles are similar within clusters and different between clusters. In practice, however, dramatic phenotypical differences might effect only a relatively small subset of the mRNA transcripts. Such differences are “washed out” by uniform measures of similarity (such as the Pearson correlation used by many clustering procedures). For example, the classification discovered by Alizadeh *et al* (2000) is not apparent when tissues are clustered using all the genes. In this particular example, a set of relevant genes was identified based on other considerations and prior hypotheses about potential sources of differences between DLBCL subtypes.

In the current work we take a direct unsupervised approach to class discovery. The process we develop consists of two components. We start by defining a figure of merit to putative partitions of the set of samples. We are guided by the fact that biologically meaningful partitions of the samples are typically manifested by a large overabundance of genes that are differentially expressed in the different sample classes. That is, the number of genes that sharply separate two biologically meaningful classes is extremely higher than that expected for a random partition of the data. Therefore, reasoning in reverse, we seek partitions of the samples for which we observe an overabundance of informative genes using the overabundance score of Section 4. In Section 5 we consider an alternative score that measures how well we can classify tissues according to the putative partition. This score is based on a cross-validation procedure that learns the classifier from some of the tissues, and evaluate prediction on others. We use the *naive Bayesian Classifer* with the *leave-one-out cross validation* (LOOCV) test as specific embodiment of this process.

Once we define a figure or merit, we apply heuristic search methods, such as *simulated annealing*, exploring the space of all possible partitions of the set of samples. As described in Section 6, this process is iterated to find several different partitions. In Sections 7 and 8 we assess the performance of these methods applied to both simulated data (where we can know the “true” classification) and actual biological data.

2 Informative Genes

We start with some definitions. Assume that we are given a *data set* D, consisting of M vectors $\langle x_1, \ldots, x_M \rangle$. Each *tissue* or *expression pattern*, x_i, is a vector in $\mathbf{R}^N$ that describes the expression values of N genes/clones in a particular biological sample. A *labeling* for D is a vector $l = \langle l_1, \ldots, l_M \rangle$, where the *label* l_i associated with x_i is either $-$ (negative sample), $+$ (positive sample), or 0 (control sample). Control samples can be ignored when genes are scored for relevance.

Consider an expression data with a known classification of tissues (typically based on histological measurements, pathological analysis, or genetic level information). In order to highlight the genes whose function underlie the molecular level differences between the different tissue classes we are interested in scoring the genes according to their relevance to the distinction between the different tissue classes.

In the literature several methods for scoring genes have been proposed. *Parametric* scores make assumptions about the form of the distribution of the scores within each group. For example, the *t-test* score (Alon et al. 1999, Schummer et al. 1999) compare the hypothesis that each group has a different mean to the hypothesis that they have the same mean. This test assumes that both groups of expression values have the same variance. The *separation score* of Golub *et al* (1999) attempts to measure the difference between the distributions of expression values in the two groups. This score estimates a Gaussian distribution for each group and then measures the distance between them in terms of standard deviations.

Non-parametric methods do not make distributional assumptions about the expression levels. As such they are more robust. These include *Wilcoxon* test (DeGroot 1989), the $TNoM$ score of Ben-Dor *et al* (2000), and the $INFO$ score of Ben-Dor *et al* (2000).

We now briefly describe the scores used in this work. Assume that a tissues are labeled as positive, and b are labeled as negative. Let g be a gene we want to score for relevance with respect to the positive vs. negative partition. Intuitively, g is relevant to the tissue partition if it is either over-expressed in the positive tissues (compared to the negative tissues) or vice-versa.

To formalize this notion of relevance, we consider how g's expression levels in the positive tissues interlace with its expression levels in the negative tissues. To do so, we order the tissues according to the expression levels of g. let $\pi = \langle \pi_1, \ldots, \pi_{a+b} \rangle$ be the permutation of the tissues induced by the expression levels of g. That is, g expression level is minimal in π_1, and maximal in π_{a+b}. The *rank vector*, v, of g, is a $\{-,+\}$ vector of length $a+b$, where v_i is the label of the tissue π_i.

For example, if g's expression levels in the positive tissues are $\{10, 20, 30, 50, 60, 70, 110, 140\}$, and g's expression levels in the negative tissues are $\{40, 80, 90, 100, 120, 130, 150\}$, then

$$v = \langle +, +, +, -, +, +, +, -, -, -, +, -, -, +, - \rangle. \tag{1}$$

Note that the rank vector v captures the essence of the differential expression profile of g. If g is under-expressed in the positive class, then the positive entries of v are concentrated in the left hand side of the vector, and the negative entries are concentrated at the right hand side. Similarly, for the opposite situation. In the latter case we can partition v into a prefix x, consisting of mostly $-$, and a suffix y, consisting of mostly $+$ (or vice versa). On the other hand, if g is not informative with respect to the given labeling, the $+$ and $-$ in v are interleaved, and there no good partition of v into homogeneous prefix and suffix. The $TNoM$ and $INFO$ scores, described below are two natural ways to quantify the relevance (or information level) of a rank vector based on its most homogeneous partition.

The $TNoM$ score of v corresponds to the partition that best divides v into a homogeneous prefix and a homogeneous suffix. Formally, the $TNoM$ score of a rank vector v is defined as

$$TNoM(v) = \min_{x;y=v} \min([\#_-(x) + \#_+(y)], [\#_+(x) + \#_-(y)]) \tag{2}$$

where $\#_s(x)$ is the number of times a symbol s appears in the vector x. Thus, for each partition $x; y$ of v, we first consider the classification that labels x as positive and y as negative. In this case the number of misclassifications is $\#_-(x) + \#_+(y)$. Then, we consider the opposite classification, where the number of misclassifications is $\#_+(x) + \#_-(y)$. Finally, we return the partition for which the best classification makes the smaller number of misclassifications.

For example, for the rank vector v in Equation 1, the best partition of v into two parts is

$$v = \langle +, +, +, -, +, +, + \rangle ; \langle -, -, -, +, -, -, +, - \rangle, \tag{3}$$

and thus, $TNoM(v) = 1 + 2 = 3$. Note that the partition of v is equivalent to choosing a threshold expression level, and counting the number of induced misclassifications (and hence the name *Threshold Number of Misclassification* $TNoM$).

The $TNoM$ score does not distinguish a rule that makes k one-sided errors (e.g., all the errors are tissues of class $+$ that are predicted as $-$) and a rule that makes $k/2$ errors of the first kind and

$k/2$ error the second kind. This distinction is important, since a rule that makes only one-sided errors is performing quite badly on one of the classes. We now describe an approach that attempts to make such finer distinctions.

Similar to the $TNoM$ score, the $INFO$ score of Ben-Dor, Friedman & Yakhini (2000) measures the level of homogeneity of the partitions of the rank vector of g. However, instead of counting misclassified sampled (as done in $TNoM$), $INFO$ score uses the information theoretic notion of conditional entropy (Cover & Thomas 1991). Let x be a $\{-,+\}$ vector, and let p denote the fraction of positive entries in x. The *entropy* of x, is defined as

$$H(x) = -p\log(p) - (1-p)\log(1-p).$$

The entropy measures the *information* in the vector x. This quantity is non-negative, and equal to 0 if and only if $p = 0$ or $p = 1$. That is, if x is homogeneous. The maximal value of $H(x)$ is 1 when x is composed of an equal number of positive and negative labels. Thus, $H(x)$ is an information-theoretic measure of (non-)homogeneity.

The $INFO$ score of v is defined to be the minimal weighted sum of the entropies of a prefix-suffix division. That is,

$$INFO(v) = \min_{x;y=v}\left\{\frac{|x|}{|v|}\cdot H(x) + \frac{|y|}{|v|}\cdot H(y)\right\},$$

where $|\cdot|$ is the length of the vector. This is the *conditional entropy* of the rank vector given the partition of the samples into two groups (these in x and these in y). The conditional entropy is a non-negative quantity. It is equal to 0 if and only if the division of v in to two groups is perfect. That is, x contains only '+' and y contains only '-' (or vice versa).

For example, if we consider rank vector v of Equation 1, the best partition for the $INFO$ score happens to be the same as in Equation 3. There we get

$$\begin{aligned} INFO(v) &= \frac{7}{15}\cdot H(\frac{6}{7}) + \frac{8}{15}\cdot H(\frac{1}{4}) \\ &= \frac{7}{15}\cdot 0.5917 + \frac{8}{15}\cdot 0.8113 = 0.7088 \end{aligned}$$

3 Computing p-values

When scoring a gene for how relevant it is with respect to a given partition of the set of samples it is important to evaluate the result against a null model. To this end we want to compute the probability of this gene (with the given expression values) being so relevant for a uniformly randomly drawn partition of the samples. This number is the *p-value* corresponding to the scoring method in effect and the given score level s. Genes with very low p-values are very rare in random data and their relevance to the studied phenomenon is therefore likely to have biological, mechanistic or protocol reasons.

We always compute p-values in the context of a null hypothesis. In the usual parametric tests, the null-hypothesis is about the distribution of expression values of g. For example, the null hypothesis underlying the t-test assumes that the expression values of g are sampled from a normal distribution (with unknown mean and variance).

In the context of gene expression data, we do not necessarily want to make assumptions about the distribution. Instead, we only assume that the assignment of labels to tissues is independent of the gene expression data. More precisely, under the null model we assume that the tissue labels are randomly and uniformly permuted.

Formally, let $\{-,+\}^{(n,p)}$ denote all labelings with $n\,'-'$ entries and $p\,'+'$ entries. A scoring method $\mathcal{S}$ (e.g., $TNoM$) is a function that takes a rank vector $v \in \{-,+\}^{(n,p)}$ and returns a score as described in Section 2. Let V be a random labeling drawn uniformly over $\{-,+\}^{(n,p)}$. The p-value of a score level s is then

$$pVal(s) = Prob(\mathcal{S}(V) \leq s) \tag{4}$$

Ben-Dor, Bruhn, Friedman, Nachman, Schummer & Yakhini (2000) describe a dynamic programming procedure for computing p-values for the $TNoM$ score, and Ben-Dor, Friedman & Yakhini (2000) describe a stochastic simulation procedure for computing p-values for the $INFO$ score. Here we describe how to efficiently compute *exact* p-values for both the $TNoM$ and $INFO$ scores. Such efficient procedures enable the class discovery methods descried in Section 6.

3.1 TNoM p-Values

The combinatorial character of $TNoM$ makes it amenable to rigorous calculations. Ben-Dor *et al.* (2000) describe a dynamic programing procedure for computing p-values for the TNoM score. In this section, we use the reflection principle, in a repeated manner, to develop a closed form formula of the $TNoM$ scores in $\{-,+\}^{(n,p)}$. (It is easy to see how to extend these results to p-value computations when we also have unlabeled samples.)

Let $v \in \{-,+\}^{(n,p)}$ be a rank vector. We can think of v as defining a rectilinear path in the plane, denoted π_v. Specifically, at the i'th step we progress one unit on the x-axis, and either add or subtract a unit on the y-axis depending on the i'th label in v. Formally, π_v is a function $\pi_v : \{1, 2, \ldots, M\} \longrightarrow$ **Z** where $\pi_v(i) = \#_+(v_{1:i}) - \#_-(v_{1:i})$. Here, for $v = \langle v_1, v_2, \ldots, v_M \rangle$ we use $v_{i:j}$ $(i \leq j)$ to denote $\langle v_i, v_{i+1}, \ldots, v_j \rangle$.

The correspondence $v \mapsto \pi_v$ is one to one and onto from $\{-,+\}^{(n,p)}$ to the set of rectilinear paths that start at $(0,0)$ and end at (M,C), where $M = n+p$ and $C = p-n$. We denote by $\Pi\left[(x_s,y_s) \longrightarrow (x_e,y_e)\right]$ the collection of all rectilinear, unit step paths starting at (x_s,y_s) and ending at (x_e,y_e).

The following lemma characterizes the set of paths that corresponds to a set of vectors $v \in \{-,+\}^{(n,p)}$ with a given $TNoM$ score.

Proposition 3.1 *Let $v \in \{-,+\}^{(n,p)}$. Then, $TNoM(v) \leq s$ if and only if there is an i such that $\pi_v(i) \geq p-s$ or $\pi_v(i) \leq s-n$.*

Proof:

Recall that the $TNoM(v)$ is defined by minimizing over the errors made by possible classifications over the rank vector . We start by considering the errors made by a specific classification. Let $v \in \{-,+\}^{(n,p)}$, and suppose we partition v into the vectors $v_{1:i}$ and $v_{i+1:M}$. There are two classifications we can make over this partition.

- If we classify $v_{1:i}$ as '+' and $v_{i+1:M}$ as '-', then the number of errors of the rule is:

$$\begin{aligned} \#_-(v_{1:i}) + \#_+(v_{i+1:M}) &= \#_-(v_{1:i}) + p - \#_+(v_{1:i}) \\ &= p - \pi_v(i) \end{aligned}$$

- If we classify $v_{1:i}$ as '-' and $v_{i+1:M}$ as '+', then the number of errors is:

$$\begin{aligned} \#_+(v_{1:i}) + \#_-(v_{i+1:M}) &= \#_+(v_{1:i}) + n - \#_-(v_{1:i}) \\ &= \pi_v(i) + n \end{aligned}$$

Thus, we can rewrite (2) as

$$\begin{aligned} \mathit{TNoM}(v) &= \min_i \min([\#_-(v_{1:i}) + \#_+(v_{i+1:M})], [\#_+(v_{1:i}) + \#_-(v_{i+1:M})]) \\ &= \min_i \min([p - \pi_v(i)], [n + \pi_v(i)]) \end{aligned}$$

The claim follows immediately. ∎

Let s be a score level of interest. Set $A = p - s$ and $B = n - s$. (Note that we are only interested in s for which both $A \geq 0$ and $B \geq 0$, since otherwise the p-value is 1.) By Proposition 3.1 we have

$$\mathit{Prob}\,(\mathit{TNoM}(L) \leq s) = \nu(A,B) \cdot \binom{M}{p}^{-1} \tag{5}$$

where

$$\nu(A,B) = \left| \left\{ \pi \in \Pi\,[(0,0) \longrightarrow (M,C)] : \max_i \pi(i) \geq A \text{ or } \min_i \pi(i) \leq -B \right\} \right|$$

That is, $\nu(A,B)$ is the number of paths that start at $(0,0)$, terminate at (M,C), and visit either the $y = A$ or the $y = -B$ line (or both). To compute $\nu(A,B)$ we use the *repeated reflection principle*. We start by classifying paths. An *alternating A/B pattern* is a word $w \in (AB)^* \cup B(AB)^* \cup (BA)^* \cup A(BA)^*$, for example $ABABA$. Let $|w|$ denote the *length* of a pattern, e.g $|ABABA| = 5$. A path $\pi \in \Pi\,[(0,0) \longrightarrow (M,C)]$ is said to *visit the pattern* w if there is a set of indices $t_1 \leq t_2 \leq \ldots \leq t_{|w|}$ such that if the j'th symbol in the pattern is A, then $\pi(t_j) = A$, and if the j'th symbol of the pattern is B, then $\pi(t_j) = -B$. For an alternating A/B pattern w we set

$$\Lambda(w) = |\{\pi \in \Pi\,[(0,0) \longrightarrow (M,C)] : \pi \text{ visits } w\}|\,.$$

Lemma 3.2

$$\nu(A,B) = \sum_{w:|w| \leq \lceil \frac{M}{A+B} \rceil} (-1)^{|w|+1} \Lambda(w).$$

Proof: This is an inclusion/exclusion like counting process. Explicitly: we count the paths that do venture out of the strip between A and $-B$. First we count all paths that cross A: $\Lambda(A)$. We add the number of paths that cross $-B$ to get $\Lambda(A) + \Lambda(B)$. We have double counted all paths that visit both A and $-B$. So, we subtract $\Lambda(AB) + \Lambda(BA)$. Now, however, paths that visit both AB and BA are subtracted twice. So, we add $\Lambda(ABA) + \Lambda(BAB)$. We continue this process to obtain the stated formula. Of course we only need to consider patterns up to length $M/(A+B)$ since it takes at least $A + B$ steps to go from $y = A$ to $y = -B$ or vice versa. ∎

Now it only remains to count the number paths that visit a particular pattern.

Lemma 3.3 *Consider an alternating A/B pattern $w = w(1)w(2)\ldots w(l)$. Set*

$$t(w) = \begin{cases} 2\sum_{i=1}^{l} w(i) & \textit{if } w(l) = A \\ -2\sum_{i=1}^{l} w(i) & \textit{if } w(l) = B \end{cases}$$

We then have

$$\Lambda(w) = |\Pi\left[(0, t(w)) \longrightarrow (M, C)\right]| = \binom{M}{(C - t(w) + M)/2}.$$

Proof: The first equation follows by repeated reflection. The second is a simple calculation of the number of paths connecting any point $(0, t(w))$ to (M, C). For illustration purposes assume $w = A$. To any path $\pi \in \Pi\left[(0,0) \longrightarrow (M, C)\right]$ that goes through the line $y = A$ we can match, in a one to one and onto manner, a path $\rho(\pi) \in \Pi\left[(0, 2A) \longrightarrow (M, C)\right]$ by reflecting the part that is to the left of the first visit to $y = A$, across the $y = A$ line. Therefore, in this case:

$$\Lambda(w) = |\Pi\left[(0, 2A) \longrightarrow (M, C)\right]|$$

To deal with the general case we repeatedly reflect across the lines $y = A$ and $y = -B$. For more on repeated reflection see (Feller 1970). ▮

The combination of Eq. (5), Lemma 3.2 and Lemma 3.3 yields a closed form formula for computing the distribution of the $TNoM$ score, in $\{-,+\}^{(n,p)}$.

3.2 INFO p-Values

Consider $v \in \{-,+\}^{(n,p)}$. Recall that

$$\mathit{INFO}(v) = \min_i \frac{i}{M} H(v_{1:i}) + \frac{M-i}{M} H(v_{i+1:M}).$$

To compute the p-value for a score level s we examine the number of paths that *do not* achieve score s. This collection of paths can be characterized as paths that are bounded within a certain region of the plane.

Let $v \in \{-,+\}^{(n,p)}$ be a rank vector, and let π_v be the corresponding path from $(0,0)$ to (M, C). Recall that $\pi_v(i) = \#_+(v_{1:i}) - \#_-(v_{1:i})$. It is easy to see, that this implies that $\#_+(v_{1:i}) = \frac{\pi_v(i)+i}{2}$ and similarly, $\#_+(v_{i+1:M}) = \frac{2p-\pi_v(i)-i}{2}$. Let

$$C(i, y) = \frac{i}{M} H\left(\frac{i+y}{2i}\right) + \frac{M-i}{M} H\left(\frac{2p-y-i}{2(M-i)}\right).$$

This is the conditional entropy of a labels vector v for which $\pi_v(i) = y$, achieved by dividing v into the prefix with the first i symbols in v and the suffix with the last $M - i$ symbols in v. Note that this quantity does not depend on how the path π_v reaches the point (i, y).

As $\mathit{INFO}(v) > s$ if and only if $C(i, \pi_v(i)) > s$ for all $i = 0, \ldots, M$. we want to count the number of paths that are restricted to the grid defined by

$$\mathcal{R}(s) = \{(i, y) : C(i, y) > s\}.$$

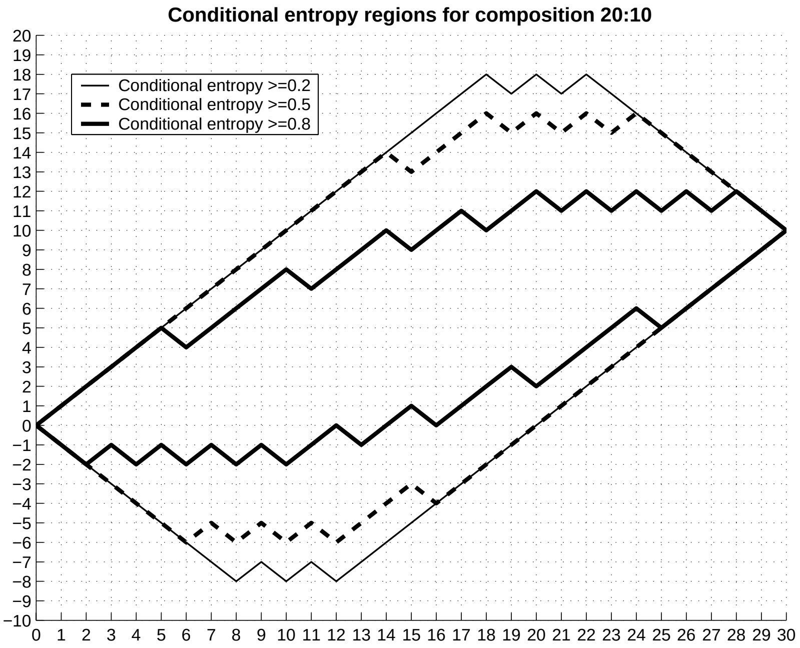


Figure 1: Conditional entropy regions, $\mathcal{R}(s)$ for some example score levels. The probability of having a rank vector with $\mathit{INFO}(v) \le s$ is obtained by counting paths that venture out of the corresponding region $\mathcal{R}(s)$.

Figure 1 depicts example conditional entropy regions of this form.

To count the paths that don't venture out of $\mathcal{R}(s)$ we now apply a dynamic programming scheme. Fix s. For $0 \le i \le M$ and $-n \le y \le p$ define

$$\Gamma(i, y; s) = |\{\pi \in \Pi\left[(0,0) \longrightarrow (i,y)\right] : (j, \pi(j)) \in \mathcal{R}(s) \text{ for all } 0 \le j \le i\}|.$$

Then we have that $\Gamma(0,0;s) = 1$ and $\Gamma(0,y;s) = 0$ for $y \neq 0$, and the recursion rule

$$\Gamma(i, y; s) = \begin{cases} \Gamma(i-1, y+1; s) + \Gamma(i-1, y-1; s) & (i,y) \in \mathcal{R}(s) \\ \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

The entries $\Gamma(i, y; s)$ can be computed by dynamic programing procedure that fills the $O(M^2)$ possible entries. Once we compute $\Gamma(M, C; s)$, we set

$$\mathit{Prob}(\mathit{INFO}(L) \le s) = 1 - \Gamma(M, C; s) \cdot \binom{M}{p}^{-1}.$$

4 Overabundance analysis

Consider a set of p positively labeled samples, and n negatively labeled samples. Let l denote the vector of labels for the samples and let N denote the number of genes profiled for each tissue. In

Section 2 we described how to compute relevance scores ($TNoM$ and $INFO$) for each gene with respect to l. In Section 3 we described efficient procedures for evaluating the significance levels (p-values), for those score methods. More specifically, for each score level s, we can compute the probability p_s that a gene will attain this score (or better) assuming that the tissues labels are randomly permuted.

In this section, we describe two methods that utilize the individual gene p-values in computing a signficance score for the entire collection of expression profiles. We term this approach *overabundance analysis*. For each possible relevance score s attained by the scoring method of choice (e.g., $TNoM$), we can compare the actual number of genes in the dataset attaining a score of s or better to the number of genes expected to have a score of s or better under random labeling (which is $N \cdot p_s$).

Assume now that the given labeling vector l indeed captures a biologically meaningful classification of the tissues (say tumor samples vs. normal samples, or samples treated with two different drugs). Even without a complete understanding of the molecular basis of the difference between the two tissue classes we expect to observe much more informative genes than would be expected at random. Indeed, examining data sets with biologically meaningful classifications, we find an over-abundance of significantly informative genes (Ben-Dor, Friedman & Yakhini 2000, Bittner et al. 2000). Figure 2 contrasts the expected number of genes with particular p-value to the actual number of genes for the $TNoM$ and $INFO$ scores in several published datasets.

The general trend is clear. The number of genes with small scores is much higher than expected. For example, in the Leukemia data set (Golub et al. 1999), there are 3 genes with $TNoM$ score 3 (p-value $7.8 \cdot 10^{-15}$) while the expected number is $5.5 \cdot 10^{-11}$. Moreover, there are 294 genes with $TNoM$ score 15 or less, while the expected number is roughly 1.

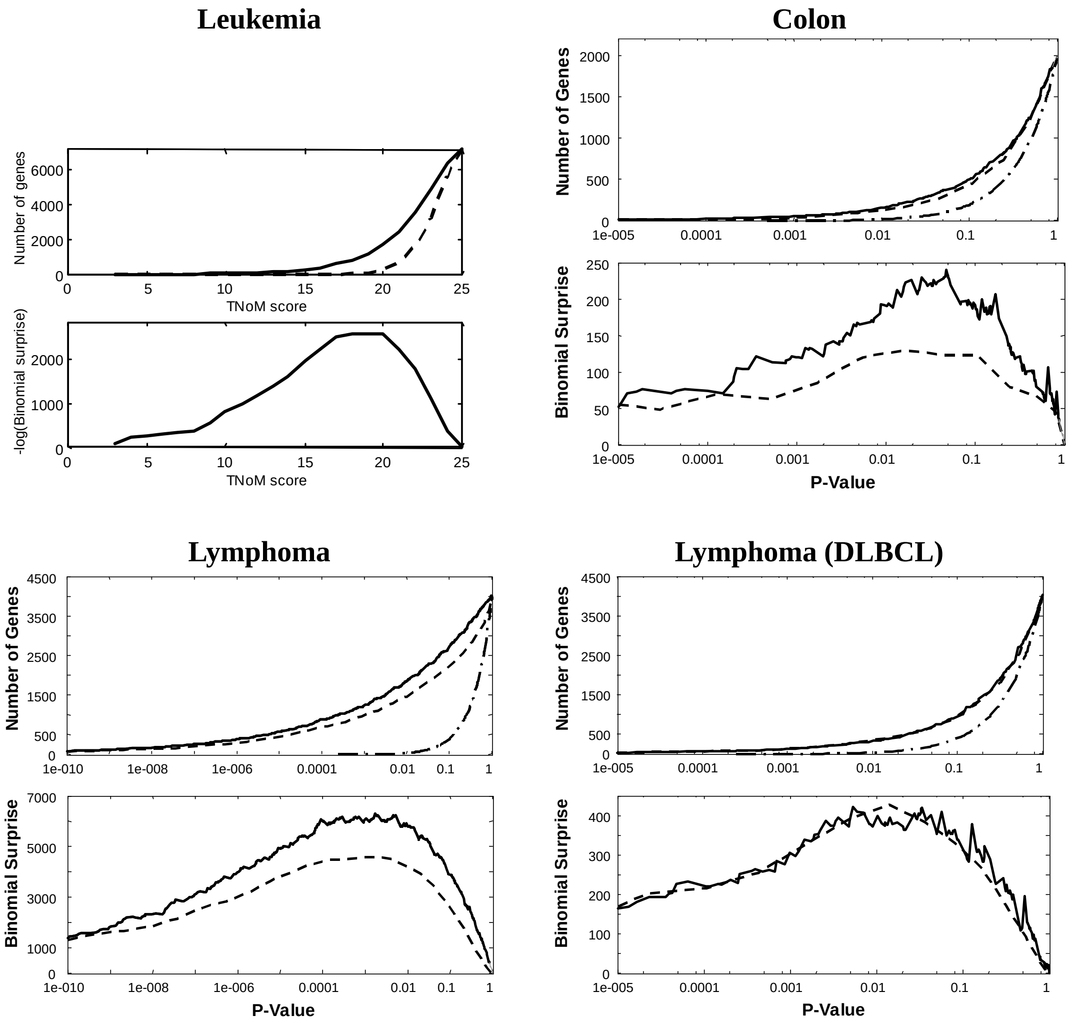
This is an overabundance of informative genes, meaning that the expression profiles carry information relevant to the biological classification. Our next step is to *quantify* the statistical significance of the this statement. This quantification is important for two types of situations.

- Consider a biologically meaningful classification (e.g., two subtypes of cancer, as in the case of the Leukemia data set). Then, we want to ascertain whether gene expression patterns reflect that classification. The examples we discuss above show that this is the case without doubt. In other classifications, when there are fewer tissue samples, or more subtle signal, the situation might not be obvious. Using standard methods (e.g., using Bonferroni bounds), we can determine whether a single gene is significant for the classification. Our aim, however, is to take into account the global patterns. That is, the behavior of all the genes. An overabundance of informative genes is an indication of statistical significance, even if no single gene is Bonferroni significant.

- Consider a putative classification, as in Bittner *et al* (2000), that might correspond to a real biological distinction. Clearly, the ultimate test for a putative classification is a biological validation test (as described therein). However, statistics is a tool for evaluating classifications before planning further experiments. Thus, we want to develop statistical scores that measure the significance of suggested partitions.

We now present a method for scoring a proposed partition of the tissues.

The formal model is as follows. Let $\mathcal{L} = \{-, +\}^{(n,p)}$ denote the set of possible labeling of the samples with the same mixture of positive and negative labels as in l. Fixing a gene scoring


Leukemia

Colon

Lymphoma

Lymphoma (DLBCL)

Figure 2: Comparison of the number of significant genes in actual dataset to expected number under the null-hypothesis (random labels). The x-axis denotes the p-value of the score. In the top part of each graph, the y-axis is the number of genes, in the bottom part, the y-axis is the negative logarithm of the probability of the observed number of genes given the binomial model, which is exactly the surprise score (see below). Data sets: Leukemia from Golub *et al* (1999); Colon from Alon *et al* (1999); and Lymphoma from Alizadeh *et al* (2000).

method (e.g. $TNoM$), let $s_1 < s_2 < \cdots < s_k$ denote the possible relevance scores. As described in Section 3, we can compute the probability, p_i of attaining score s_i, if the labeling is chosen at random from $\mathcal{L}$. We define $q_i(l)$ to be the fraction of genes that have a score of exactly s_i under the labeling l. We now ask how rare it is to observe the distribution $Q(l) = \{q_i(l)\}$ when we expect to see the distribution $P = \{p_i\}$. To answer this question we need to make an additional simplifying assumption

Ind-Genes: The expression levels of the genes are independent

This assumption implies that knowing the scores of any subset of genes under a random labeling l adds no information about the scores of other genes under the same labeling. It is clear that this assumption oversimplifies things: expression levels of genes can be correlated, and thus their scores on random labeling are not independent. Nonetheless, as in many modeling situations, this simplifying assumption allows for efficient computations, and might capture the essence of the results.

Intuitively, we consider each score level s_i independently. We compare the actual number of genes with a score less than or equal to s_i to the expected number. The most striking overabundance observed will determine the surprise score of the labeling l. Formally, we first define the *surprise* at s_i. Let $q_{\leq i}(l)$ denote the observed fraction of genes with a score of s_i or better (for the labeling l). Let $p_{\leq i}$ denote the expected fraction of genes with a score s_i or better (for a labeling chosen at random from $\mathcal{L}$). That is

$$q_{\leq i}(l) = \sum_{j \leq i} q_j(l), \ \text{ and } \ p_{\leq i} = \sum_{j \leq i} p_j.$$

Under the independence assumption, the number of genes with a score of s_i or better has a binomial distribution, with parameters $p_{\leq i}$ and N. We define the *surprise* at score s_i with respect to l to be

$$\text{Surprise}(s_i, l) = -\log Prob(Bin(p_{\leq i}, N) \geq N q_{\leq i}(l))$$

We can compute the surprise score directly, or using a Chernoff bound on the tail probability of the binomial distribution. Of course, at different score values we get different surprise values. However, we can find the score s_i at which the observed number is most surprising:

$$\text{Max-Surprise}(l) = \max_i \text{Surprise}(s_i, l).$$

This quantity is employed to evaluate the labeling vector l. This matches our intuition that a good labeling should have a surprisingly large number of informative genes.

5 Classification Scores

In the previous sections we studied scores that evaluate the quality of putative label vectors by measuring significant deviations from the distribution of scores we expect under null-hypothesis models. An alternative approach is to seek classes which are predictable based on the gene expression measurements.

In (Ben-Dor, Bruhn, Friedman, Nachman, Schummer & Yakhini 2000, Slonim et al. 2000) the problem of predicting tissue classification is examined. As demonstrated there, for actual labelings in real-life data sets, it is possible to *train* a classifier that has good predictive accuracy. More precisely, a classification algorithm is a function f_D that depends on a data set D of patterns and sample labels. Given a new query $x \in \mathbf{R}^N$, this function returns a predicted label $\hat{l} = f_D(x)$. Good predictive accuracy means that predicted labels match the "true" label of the query. Several classification methods were applied to gene expression data (Ben-Dor, Bruhn, Friedman, Nachman, Schummer & Yakhini 2000, Ben-Dor, Friedman & Yakhini 2000, Slonim et al. 2000). For completeness we briefly review the method used here.

The *naive Bayesian classifier* (Duda & Hart 1973, Friedman 1997, Friedman et al. 1997) is based on a probabilistic approach to the problem. We start by estimating the probability of each label (e.g., $-$ or $+$) given gene g's expression level. We model this distribution by a *decision stump*: we learn a threshold t, and make one prediction if $x_g < t$ and another if $x_g > t$. The threshold t is chosen as in the $INFO$ score, and the conditional distribution for x_g above (resp. below) the threshold is estimated from the proportions of $+$ and $-$ labels for samples where g's expression level is above (resp. below) t. Then, assuming that expression patterns of genes are independent given the labeling (this is the "naive" assumption) and using Bayes rule, we get that

$$\begin{aligned}
\log \frac{P(+ \mid x)}{P(- \mid x)} &= \log \frac{P(+)}{P(-)} + \log \frac{P(x \mid +)}{P(x \mid -)} \\
&= \log \frac{P(+)}{P(-)} + \sum_g \log \frac{P(x_g \mid +)}{P(x_g \mid -)} \\
&= \log \frac{P(+)}{P(-)} + \sum_g \left(\log \frac{P(+ \mid x_g)}{P(- \mid x_g)} - \log \frac{P(+)}{P(-)} \right).
\end{aligned}$$

Where the first step is an application of Bayes rule, the second depends on the independence assumption, and the third step is again an application of Bayes rule. If $\log\frac{P(+|x)}{P(-|x)}$ is positive we predict $+$, otherwise we predict $-$.

A key issue we need to address is how to evaluate the accuracy of a classification method applied to a given labeled data set and labeling. We follow standard methodology and use *leave one out cross validation* (LOOCV) to estimate the prediction accuracy of a classification method on new examples. This procedure iterates on the samples in the data set. In each iteration, it removes a single sample and trains the classification procedure on the remaining data. The trained classifier is then applied to the held-out sample and the predicted label is compared to the true label. The fraction of errors thus committed in the entire process, is our estimate of the error rate of the classification procedure.

A final issue is *feature selection*. As (Ben-Dor, Bruhn, Friedman, Nachman, Schummer & Yakhini 2000, Slonim et al. 2000) show, predictions based on an informative subset of genes are more accurate than these that are based on all genes. In our procedure we employ a simple, but surprisingly effective, procedure to select genes. Given a training data, we compute the score s that attains max surprise. We then focus on genes that have this score or better. The learned classifier is then based on these genes only. We stress that in each LOOCV iteration this procedure is applied on the data set without the held-out sample. Thus, in each iteration a different score will attains maximum surprise and a different set of genes is selected. In such a situation the LOOCV estimate is the estimate of the performance of the combined classification procedure that uses the training data for feature selection and then learns a classifier based on the selected genes.

Now suppose we are given a (putative) labeling of samples in our training data. The intuition we outlined above suggests that if the labeling captures a “true” phenomenon in the data, then a LOOCV evaluation of a classification procedure (e.g., the naive Bayesian classifier) would lead to accurate predictions. In other words, we can score a labeling by the accuracy reported by LOOCV evaluation of classification with respect to this labeling. This suggests that the distinctions made by the labelings are inherent in the data and not an artifact. From a different point of view: the suggested classes can be successfully in-silico diagnosed.

We note that this classification score is related to the overabundance score. We can expect that for partitions with high overabudnace of informative genes, we will also be able to find a good classifier. Indeed, actual biological partition score high on both methods.

6 Class Discovery

In many experimental designs it is useful to find tissue classification in gene expression data. Such classifications might be due to biological phenomena (e.g., disease subtypes), or due to mechanical or protocol “noise”. Identifying classifications can lead to biological discovery or can uncover experimental or data handling errors.

The strategy we propose is simple. As we demonstrated above, biologically meaningful classifications are often characterized by overabundance of informative genes. This overabundance might be due to a small set of genes that are highly informative about the classification, or due to a larger set of genes, each of them not as surprising, but the collection of them is.

This suggests that we should examine partitions of the samples into two groups. We can then evaluate these partitions and measure to what degree they have the overabundance of informative genes. The partitions that display high overabundance are proposed a putative classifications. To carry out this intuition we need to choose a score for overabundance and then to perform *search* for high scoring partitions.

A somewhat more general formulation is to consider partitions of the samples to three groups, ’-’, ’+’ and ’0’, where the latter group corresponds to unlabeled samples that will not participate in the learned classification. This provides an additional degree of freedom in discovering meaningful classifications. Note that our surprise score inherently penalizes partitions with a high number of unlabeled samples. Reducing the number of labeled samples causes the p-values to be typically larger (there are fewer rank vector arrangements). Consequently, the surprise scores for partitions with just a few labels are smaller.

In the previous sections we described two scores for overabundance, max-surprise, and the classification score. The later score has several shortcomings for our purposes. First, it is computationally intensive, since we need to perform LOOCV iterations, and in each of these perform gene selection (which in turn, requires computing scores for all genes). Second, since the number of samples is small, the range of the score is quite limited. This implies that the classification score gives little guidance during search for high scoring labeling vectors. Third, if we use the classification score during search we are going to perform large number of LOOCV evaluations; statistical considerations show that even in random data, enough repetitions of this test will find high scoring artifactual labeling vectors (Klockars & Sax 1986, Ng 1997). To avoid these problems, we mainly use the classification score to evaluate candidate labeling vectors that are found using surprise scores we discussed in the previous section.

Thus, we prefer to use the max-surprise score to guide the search for putative classifications.

Once we choose a score for putative labeling vectors we need to find a labeling that maximizes the score. This is a discrete optimization problem and we use heuristic search techniques to find high-scoring labelings. We can formalize our problem as a search over a graph, where our goal is to find a vertex with maximum score, where one assumes some locality in the scores (i.e., neighboring vertices have similar scores). Presently, the vertices correspond to potential labelings of the samples, and the score we attempt to maximize is the max-surprise score. The edges in the graph are between pairs of labelings that agree on all labels except one sample, the labeling of which changed from "0" to either "-" to "+". Thus, we can move from one labeling to another by modifying the label of exactly one sample from classified to unclassified, or vice versa. Note that over a set of M samples, each vertex has at most $2M$ neighbors.

A common search method is the *first ascend hill climb*. In this procedure we consider all the neighbors of our current labelings in some random order. We evaluate the score of each neighbor, and once we find a neighbor with a better score, we "move" there and continue. If all neighbors have worse scores than the current candidate, we are at a local maximum, which is returned.

This procedure is straight forward and has the intuitive aspect of climbing up-hill toward better solution. However, it can get "stuck" at local maxima. Unfortunately, local maxima are common in the class discovery optimization landscape. A common method to escape local maxima is *simulated annealing* (Kirkpatrick et al. 1983). This method resembles the first ascend procedure. However, now the search procedure maintains a temperature parameter t. This parameter is updated during the search by an exponentially decreasing *cooling schedule*: at the kth step of the process the temperature is $t_0\alpha^{[k/d]}$ where $\alpha < 1$ and d is an integer. Now, if the score of the current labeling is s, and a random neighbor labeling scores s', we move to that neighbor with probability $\min(1, e^{\frac{s'-s}{t}})$. Thus, at very high temperatures the probability of taking a score-decreasing step is close to 1. It gets closer to 0 as the temperature decreases. The procedure terminates after a fixed number of steps (or equivalently, after t reaches a pre-specified temperature) and returns the best scoring labeling it encountered.

Recall that we want to construct *several* different partitions of the data. Toward this end, we employ a simple strategy of *peeling* the data set. First, we perform a heuristic search and find a high scoring putative labeling. Then, we "peel off" the genes that support this labeling from the data set. More precisely, we remove all genes with score smaller than or equal to the score that attains maximum surprise. We then reiterate the search on the remaining genes until either we exhausted all genes, or the score of the best labeling on these genes falls below a pre-specified threshold. By iteratively peeling the data set we discover a set of partitions, each supported by a disjoint set of genes. Once we finalize the search, we reevaluate each of the labeling vectors with respect to the original data set (since some previously removed genes can be relevant to a putative partition and effect its score).

7 Model and Simulations

All attempts to stochastically model gene expression data are intrinsically problematic. It is impossible to make a reasonable set of model assumptions that is universally valid for a complicated system such as the living cell. Modeling approaches are, however, successful in highlighting biological phenomena that do follow the model and thus allow for selective inference of knowledge from data.

The purpose of the stochastic simulation exercise we describe in this section is threefold: to validate our computational class discovery methods on a general stochastic model; to identify the mode of convergence to planted classes; to compare performance across methods and test the effects of various parameter changes.

7.1 The Stochastic Model: Planted Classes

We assume that the gene expression dataset stochastically depends on a hidden, biologically significant, classification $\mathcal{C}$ of the tissues into subclasses. As in real datasets, we further assume that the classification $\mathcal{C}$ effects only a small fraction of the genes, called the *$\mathcal{C}$-genes*, while the other genes, called *random genes*, express independently of $\mathcal{C}$.

For simplicity we describe a binary classification model. Modelling data with more classes in the same manner is straight forward. We assume that there exists a hidden classification $\mathcal{C}$, that partitions the M tissues into a class A tissues and b class B tissues. We denote by m the total number of genes, and by $1 - e$ the fraction of $\mathcal{C}$-gene. That is, there are $(1 - e)m$ $\mathcal{C}$-genes, and em random genes.

For each $\mathcal{C}$-gene, g, we model its expression levels in the different tissues using two distributions - $\mathcal{D}_A$ for class A tissues, and $\mathcal{D}_B$ for class B tissues. We assume that $\mathcal{D}_A$ and $\mathcal{D}_B$ are normal distributions with a constant coefficient of variation[1], s. That is,

$$\mathcal{D}_A = N(\mu_A, \mu_A s), \quad \mathcal{D}_B = N(\mu_B, \mu_B s).$$

The means of the distribution, μ_A, and μ_B are uniformly chosen from the interval $[-1.5d, 1.5d]$. Thus, the expected distance between the two means is d (a parameter of the model).

The expression level of the random genes in all tissues, independent of their class, is assumed to be normally distributed with zero mean and standard deviation of one. Note that any classification of the tissues, $\mathcal{C}'$, might be supported by some random genes. However, the true classification $\mathcal{C}$, will be supported by a statistically significant number of genes because of the $\mathcal{C}$-genes.

In summary, the planted classification model is fully specified by a classification $\mathcal{C}$, of the tissues into two classes of sizes a and b respectively, and by the model parameters:

- m - the total number of genes.
- e - the fraction of random genes in the data.
- d - the expected distance between the two mean expression levels, pertaining to the two planted classes.
- s - the coefficient of variation for expression level distributions.

7.2 Results on Synthetic data

In this section we report a simulation based evaluation of our discovery process. We varied the model parameters (a, b, m, e, d, s), and employed Max-Surprise with the $TNoM$ score in a simulated annealing local search.

[1] This assumption is supported, for example, when the expression levels are logs of red to green signal ratios in a two dye expression profiling measurement (Chen et al. 1997).

Simultaneously varying d, s, and e, we observed that the search results were relatively insensitive to the parameter d (compared to s and e). Hence, we concentrate on s and e in the rest of the simulations.

In order to choose realistic parameter values we examined the leukemia data set (Golub et al. 1999) and best fit the model parameters to it. Omitting the details of the fitting process, the resulting values are: $m = 7129$, $a = 25$, $b = 47$, $e = 0.72$, $d = 555$, $s = 0.75$.

In our stochastic model we implicitly assume that all genes are independently distributed. However, in biological dataset, there are complicated dependencies among genes. Therefore, the effective number of independent genes in the real data set is much smaller than 7129. One way to choose a better model value for m is to choose it such that the Max-Surprise score of the hidden classification $\mathcal{C}$ (fixing the other parameters to the above values) would resemble the Max-Surprise score of the AML/ALL classification in the leukemia data (which is 2603). Using this approach we derive $m = 600$. Therefore, set the *leukemia parameters* to be

$$m = 600, \;\; a = 25, \;\; b = 47, \;\; e = 0.72, \;\; d = 555, \;\; s = 0.75$$

To test the performance of our methods on leukemia parameters, we generated 10 synthetic datasets according to the planted classification model, and compared the returned classification of the tissues to the original, planted classification. In all 10 cases, the original class was recovered perfectly.

To better study the effect of the model parameters on the algorithm performance, and to learn our algorithm limits we have varied each of the parameters (m, d, e, s) in turn, while fixing the others to their leukemia value. In the reported results below, we use $\mathcal{C}$ to denote the planted classification (that has proportions of 25 class A tissues vs. 47 class B tissues), and by $\mathcal{A}$ the classification returned by our algorithm. Recall that our algorithm searches for the tissue classification with the maximal Max-Surprise score. As we vary the model parameters, the Max-Surprise score lead of $\mathcal{C}$ (compared with the score of other classifications) changes, and thus the algorithm performance is accordingly effected:

m - Increasing m, the number of genes, increases Max-Surprise$(\mathcal{C})$, and thus makes it easier for the search heuristic to find it. We have found that $m \approx 250$ is the phase transition point. If m is larger, then the algorithm consistently recovers the hidden classes. However, for smaller m's, $\mathcal{C}$ is *not* the optimal classification (with respect to the Max-Surprise score), and thus a different classification, $\mathcal{A}$, is recovered. The difference between $\mathcal{A}$ and $\mathcal{C}$ depends on m; the smaller m is, the larger is the difference. For example, setting $m = 100$, we get that on average Max-Surprise$(\mathcal{C}) \approx 360$, while Max-Surprise$(\mathcal{A}) \approx 390$. Still, $\mathcal{A}$ and $\mathcal{C}$ are very close (differ on average only on 3 tissues).

d - We have found that d, the expected distance between the two means has very little effect on Max-Surprise$(\mathcal{C})$, and thus has very limited effect on the algorithm performance. In particular, we have varied d in the range 1 through 1000, and in all cases the algorithm recovered $\mathcal{C}$ perfectly.

e - In our model, e represented the fraction of random genes in the data, genes that express independently from the planted classification. Stated differently, we are trying to recover planted classifications that are supported on a $1-e$ fraction of the genes. As the Max-Surprise score of a classification reflects the over-abundance of informative genes, we expect Max-Surprise based methods to perform well even for high values of e. Indeed, in this study by

			Max-Surprise			LOOCV	Jaccard
Data set	Labeling	p/n/c	Score	p-Value	#	acc. (%)	Coeff.
Leukemia	original	47/25/0	2601	0.0154	1173	91.7	1
	1	43/29/0	13784	0.0007	1890	98.6	0.469
	2	32/40/0	7541	0.0126	2182	91.7	0.344
	3	43/29/0	11524	0.0054	2400	93.0	0.469
	4	46/26/0	2690	0.0558	2014	87.5	0.949
	5	22/50/0	3891	0.1292	3483	70.8	0.376
	6	45/27/0	2784	0.0759	2355	86.1	0.902
Lymphoma	Original	50/46/0	8259	0.0010	1188	87.5	1
	1	24/37/35	14514	$1.9 * 10^{-5}$	1148	100	0.780
	2	53/23/20	8342	0.0049	1598	100	0.382
	3	34/36/26	7728	0.0012	1148	91.4	0.485
	4	33/29/34	6674	0.0013	1046	88.7	0.539
	5	35/35/26	1654	0.0319	937	85.7	0.323
	6	53/17/26	5201	0.0545	2157	94.2	0.385
Lymphoma DLBCL	Original	23/22/51	545	0.0139	359	97.8	1
	1	33/12/51	2669	0.0668	1625	88.9	0.362
	2	23/22/51	2005	0.0139	776	95.5	0.324
	3	25/20/51	917	0.0460	815	88.9	0.354
	4	31/14/51	2171	0.1318	1975	82.2	0.350

Table 1: Evaluation of the discovered labelings and the original labelings in three data sets. The table reports the composition of the labeling; the $TNoM$ based max-surprise score, the p-value at the point of max surprise and the number of genes with that p-value; LOOCV accuracy of predictions the labeling (ignoring control samples); and the *Jaccard coefficient* that measures the similarity of the labeling to the original labeling.

simulations we have varied e in the range $[0, .99]$, and observed that the algorithm consistently recovered $\mathcal{C}$, up to $e = 0.95$. For higher values of e, we typically get Max-Surprise$(\mathcal{A}) >$ Max-Surprise$(\mathcal{C})$.

s - The coefficient of variation, s, plays a major role in our model. It represents the inherent random nature of the expression profile of a gene within tissues of the same class. For large values of s we get very spread distributions, contributing to higher $TNoM$ scores, and thus a lower Max-Surprise$(\mathcal{C})$. In this study we varied s in the range $[0.5, 5]$. The transition point was found at around $s = 2$. For smaller s, the planted classification $\mathcal{C}$ is recovered, for larger $\mathcal{C}$, we typically recover classification $\mathcal{A}$ with larger Max-Surprise score.

Our simulation study can be summed up as follows. First, the algorithm is very robust, performing under high levels of noise, either in the form of random genes ($e \approx .95$), or in form of high coefficient of variation ($s \approx 2$). Second, for a wide range of parameters, even much more pessimistic than those that correspond to the leukemia dataset, the algorithm consistently recovers the planted classification. Finally, if either there are too few genes ($m < 250$), or too high noise level ($e > 0.95$, or $s > 2$), than the planted classification is no longer the optimal classification, and we cannot hope to perfectly recover it.

8 Class Discovery in Gene expression Data

To evaluate the usefulness of our approach, we applied it to several gene expression data sets. They all come with a known classification that is either based on pathological considerations, or was discovered using manual analysis of gene expression data. The data sets are:

- **Leukemia:** 72 expression profiles reported by Golub *et al* (1999). These samples are divided to two variants of leukemia: 25 samples of *acute myeloid leukemia* (AML) and 47 samples of *acute lymphoblastic leukemia* (ALL). mRNA was extracted from 63 bone marrow samples and 9 peripheral blood samples. Gene expression levels in these 72 samples were measured using high density oligonucleotide microarrays spanning 7129 genes.

- **Lymphoma:** 96 expression profiles reported by Alizadeh *et al* (Alizadeh et al. 2000). 46 of these are of *diffused large b-cell lymphoma* (DLBCL) samples. The remaining 50 samples are of 8 types of tissues. In our analysis we used gene expression measurements of 4096 genes shown in (Alizadeh et al. 2000, Figure 1).

- **Lymphoma-DLBCL:** This data set is the subset of 46 DLBCL samples from the lymphoma data set. Alizadeh *et al* separated these samples into two classes *Germinal centre B-like DLBCL*, and *Activated B-like DLBCL*.

In each of these data sets we run the peeling procedure using the maximum surprise score of Section 4 with the $TNoM$ score. Table 1 summarizes the scores of the top discovered classifications using the various scoring mechanisms we discussed above and also summarize their difference to the published classification of the data sets. Note that LOOCV evaluation in this table is not an independent statistical validation of the discovered partitions, since information about the expression profile of any sample is effecting the classifier learned from any $m - 1$ set. High LOOCV success rate is, however, a statistical property that is typically associated with biologically meaningful partitions.

On the leukemia data set we run our search procedure with the additional constraint that it should only examine labeling without control tissues. Peeling found six labelings, the first four of which are shown in Table 1. All six labelings score better than the original labeling by the max-surprise score, and by the number of “significant” genes. The first three labelings also have better LOOCV accuracy than the original score. Thus, we believe that each of these captures a significant distinction. Note that the first three labelings are quite different from the original one (the Jaccard coefficient is low). Of the next three labelings, two (4 and 6) are very similar to the original labeling, yet receive slightly lower LOOCV scores.

In the Lymphoma data set, peeling also found 6 labelings. The top 2 labelings score better than the original labelings both in terms of max-surprise score and LOOCV accuracy. The first labeling contains a large group which contains mostly DLBCL samples (34 out of 37), and another group consisting mostly of samples of other types of lymphoma (Fl and CLL) . We note though that additional 12 DLBCL are set as controls. Thus, we suspect that this classification is based on genes whose expression separates DLBCL samples from the types we mentioned above.

When we focused on the DLBCL samples (constraining all others samples to be controls), peeling found 4 labelings. These labelings are all quite different than the one reported by Alizadeh *et al* (Alizadeh et al. 2000). All three score higher in terms of max-surprise and are supported by larger number of genes. The classification of Alizadeh *et al*, however, has higher LOOCV accuracy.

We consider two ways for evaluating the discovered DLBCL partitions. We now briefly describe both.

The Lymphoma data sets contains 51 samples other than DLBCL. These samples describe gene expression profiles in related cell types. Alizadeh et al. use these samples to annotate clusters of genes (e.g., "the genes that are up-regulated in activated B-cells"). Then, they classify DLBCL samples according to their behavior in these gene clusters. The group labeled "Germinal centre like DLBLC" exhibit up-regulation of genes that are also up-regulated in germinal centre samples.

We propose a more direct approach to assigning sample classes. After selecting the genes that are significant for the partition (i.e., these with p-value better than the max-surprise p-value), we train the naive Bayesian classifier on the proposed partition. We then apply this classifier to classify the control samples. For example, in Figure 3 we plot the expression levels of these genes, and sort the control samples by their classification score ($\log\frac{P(+|x)}{P(-|x)}$) to one of the classes discovered by Alizadeh et al. As we can see, the germinal centre samples are most similar to the "germinal centre like" DLBCL samples, and the blood B-cell samples are most similar to the "activated B-cell like" DLBCL samples.

Performing the same procedure on our sample clusters also leads to proposed annotation of the non-DLBCL samples. For example, for the 3rd partition found, we get the plot shown in Figure 3. In this case, some of the B-cell samples receive classification scores that point to "class 1". On the other hand, samples of T-cells receive scores that are more extreme than these found in "Class 2". This suggests that this 3rd partition corresponds to T-cell vs. B-cell distinction.

Another way of evaluating these partition is by examining the clinical data about the patients. For some of the DLBCL samples, Alizadeh *et al* report survival data (Alizadeh et al. 2000). They show that the classification they discover in the data is a good predictor of patient survival chances. They also show, that this distinction is informative even if they focus only on low clinical risk patients. (Clinical risk is evaluated using *international prognostic index*, a standard medical index, evaluated at the time the sample was taken.) In Figure 4 we plot survival rates for patients for the four putative DLBCL classifications described in Table 1. As we can see, some of the classifications, such as the forth one, are not predictive about patient survival. On the other hand, the second and third classifications are predictive about the survival chances of patients with good prognostic evaluation, and the third classification is also predictive for the whole patient population. Although these survival curves are not as distinct as the ones for the classification of Alizadeh et al., this shows that the classifications we discover might be relevant to the development of the disease.

In conclusion, in two of these data sets we manage to recover close approximations to known biologically meaningful classifications. In addition, in all three data sets we uncovered classifications that are as strongly pronounced in the data (large number of genes at significant p-value). These classifications might be biologically meaningful or artifacts of the sample preparation, or hybridization procedures. In either case, it is important that the analysis of the results take into account such strong signals in the expression data.

9 Conclusions

In this paper we put forth the problem of class discovery and distinguish it as a special subclass of the broad category of clustering problems. We describe how to efficiently compute statistical significance to how well individual genes separate tissue classes (for both the $TNoM$ and the $INFO$

methods). Based on these efficient methods, we propose several criteria for evaluating the statistical significance of putative sample classifications. The central idea is to quantify the overabundance of genes that are informative with respect to any such putative classification. We then combine these methods with search heuristics and develop an efficient search procedure for finding multiple significant classifications in data sets.

The main criterion we use in searching for new classifications is the max-surprise score. This score is appealing both because of its clear definition and because it can be efficiently evaluated. Our evaluation on synthetic data shows that searching using the max-surprise score can recover a "true" classification under a wide range of operating parameters including the number of relevant and irrelevant genes, the amount of variance in the expression level, and the difference between the expression of genes in two classes.

When we applied this procedure to real-life cancer related gene expression data sets, we found multiple highly pronounced classifications that were supported by independent evaluation methods.

The work reported here opens several intriguing research questions. First, the max-surprise score exploits a strong independence assumption. This assumption can potentially overstate the surprise of the scores we observe in the data. Thus, although our procedures performed well in practice, we still might be able to improve upon them by relaxing this independence assumption. We would like to estimate the distribution of $Q(l)$ under the null hypothesis without assuming independence. A first cut approach is based on stochastic simulation. Unfortunately, simple stochastic simulation is useful only for estimating the distribution of scores with relatively large p-value. For scores with small p-values, we will need massive repetitions of the simulation to get a single case where such a score is attained. We are currently working on developing more sophisticated methods for estimating the distribution of $Q(l)$ under the null-hypothesis, without assuming gene independence. These estimates will be used to obtain better quantification of the surprise associated with any putative partition.

Another issue is the search procedure. In this work we mainly focused on the criteria for evaluating putative classifications, and used simulated annealing, a fairly generic search method, with parameters that ensure a wide search. In addition we used peeling for finding multiple classifications. In the future, we plan to study the theoretical properties of this optimization problem, aiming at developing principled methods for this task.

Acknowledgements

We thank Ash Alizadeh, Izidore Losses, Naftali Kaminski, Mike Bittner, Yidong Chen, Markus Ringner, Benjamin Yakir, and the anonymous reviewers for comments on earlier versions of this paper and useful discussions relating to this work. We also thank Ash Alizadeh for making the DLBCL survival data available.

References

Alizadeh, A. A., Eisen, M. B., Davis, R. E., Ma, C., Lossos, I. S., Rosenwald, A., Boldrick, J. C., Sabet, H., Tran, T., Yu, X., Powell, J. I., Yang, L., Marti, G. E., Moore, T., Hudson, J., J., Lu, L., Lewis, D. B., Tibshirani, R., Sherlock, G., Chan, W. C., Greiner, T. C., Weisenburger, D. D., Armitage, J. O., Warnke, R., Staudt, L. M. & et al. (2000), 'Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling', *Nature* **403**(6769), 503–11.

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D. & Levine, A. J. (1999), 'Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays', *PNAS* **96**(12), 6745–50.

Ben-Dor, A., Bruhn, L., Friedman, N., Nachman, I., Schummer, M. & Yakhini, Z. (2000), 'Tissue classification with gene expression profiles', *Journal of Computational Biology* **7**, 559–584.

Ben-Dor, A., Friedman, N. & Yakhini, Z. (2000), Scoring genes for relevance, Technical Report 2000-38, School of Computer Science & Engineering, Hebrew University, Jerusalem. http://www.cs.huji.ac.il/∼nir/Abstracts/BFY1.html, and Technical Report AGL-2000-13, Agilent Labs, Agilent Technologies, 2000, http://www.labs.agilent.com/resources/techreports.html.

Ben-Dor, A., Shamir, R. & Yakhini, Z. (1999), 'Clustering gene expression patterns', *J. Comp. Bio.* **6**(3-4), 281–97.

Bittner, M., Meltzer, P., Chen, Y., Jiang, Y., Seftor, E., Hendrix, M., Radmacher, M., Simon, R., Yakhini, Z., Ben-Dor, A., Sampas, N., Dougherty, E., Wang, E., Marincola, F., Gooden, C., Lueders, J., Glatfelter, A., Pollock, P., Carpten, J., Gillanders, E., Leja, D., Dietrich, K., Beaudry, C., Berens, M., Alberts, D. & Sondak, V. (2000), 'Molecular classification of cutaneous malignant melanoma by gene expression profiling', *Nature* **406**(6795), 536–40.

Chen, Y., Dougherty, E. & Bittner, M. (1997), 'Ratio-based decisions and the quantitative analysis of cDNA microarray images', *Journal of Biomedical Optics* **2**(4), 364–374.

Cover, T. M. & Thomas, J. A. (1991), *Elements of Information Theory*, John Wiley & Sons, New York.

DeGroot, M. H. (1989), *Probability and Statistics*, Addison Wesley, Reading, MA.

Duda, R. O. & Hart, P. E. (1973), *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York.

Eisen, M. B., Spellman, P. T., Brown, P. O. & Botstein, D. (1998), 'Cluster analysis and display of genome-wide expression patterns', *PNAS* **95**(25), 14863–8.

Feller, W. (1970), *An introduction to Probability Theory and Its Applications*, Vol. I, third edn, John Wiley & Sons.

Friedman, J. (1997), 'On bias, variance, 0/1 - loss, and the curse-of-dimensionality', *Data Mining and Knowledge Discovery* **1**. in print.

Friedman, N., Geiger, D. & Goldszmidt, M. (1997), 'Bayesian network classifiers', *Machine Learning* **29**, 131–163.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D. & Lander, E. S. (1999), 'Molecular classification of cancer: class discovery and class prediction by gene expression monitoring', *Science* **286**(5439), 531–7.

Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. (1983), ‘Optimization by simulated annealing’, *Science* **220**(4598), 671–680.

Klockars, A. J. & Sax, G. (1986), *Multiple Comparisons*, Sage Publications.

Ng, A. Y. (1997), Preventing “overfitting” of cross-validation data, *in* ‘Proc. 14th Inter. Conf. Machine Learning’, pp. 245–253.

Schummer, M., Ng, W. V., Bumgarner, R. E., Nelson, P. S., Schummer, B., Bednarski, D. W., Hassell, L., Baldwin, R. L., Karlan, B. Y. & Hood, L. (1999), ‘Comparative hybridization of an array of 21,500 ovarian cDNAs for the discovery of genes overexpressed in ovarian carcinomas’, *Gene* **238**(2), 375–85.

Sharan, R. & Shamir, R. (2000), CLICK: A clustering algorithm with applications to gene expression analisys, *in* ‘ISMB’00’.

Slonim, D. K., Tamayo, P., Mesirov, J. P., Golub, T. R. & Lander, E. S. (2000), Class prediction and discovery using gene expression data, *in* ‘Fourth Annual International Conference on Computational Molecular Biology’.

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E. S. & Golub, T. R. (1999), ‘Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation’, *PNAS* **96**(6), 2907–12.

Tavazoie, S., Hughes, J. D., Campbell, M. J., Cho, R. J. & Church, G. M. (1999), ‘Systematic determination of genetic network architecture’, *Nat Genet* **22**(3), 281–5. Comment in: Nat Genet 1999 Jul;22(3):213-5.

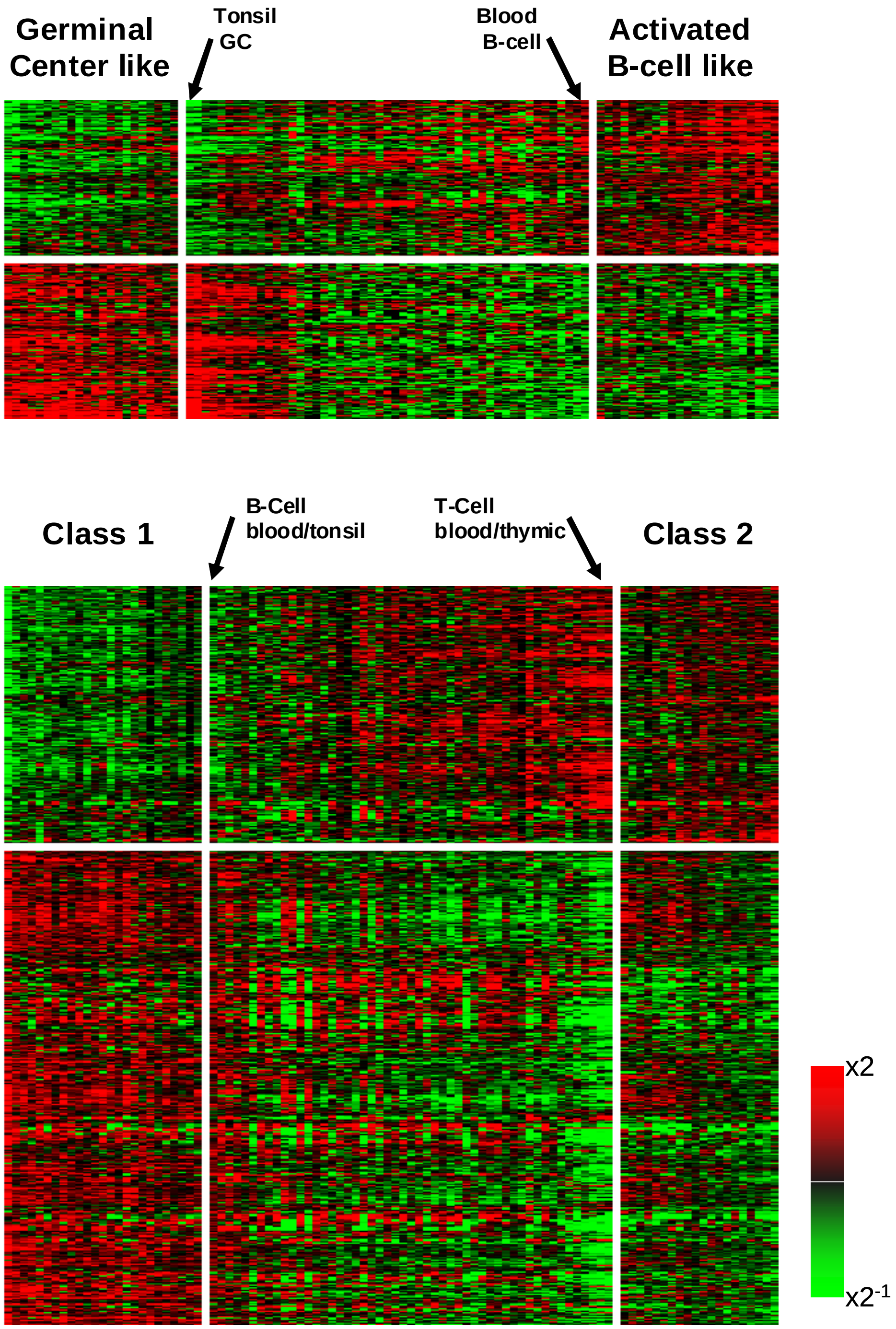


Figure 3: Expression patterns of genes relevant to a partition. Top: Paritition of Alizadeh *et al*, Bottom: 3rd discovered partition. For each partition, we plot the expression level of the genes that have relevance p-value ($TNoM$) smaller or equal to the max-surprise of the partition (350 genes for the original and about 800 for the 3rd discovered partition). Rows correspond to genes, columns correspond to samples. Samples are sorted by the classification score induced by the partition (see text).

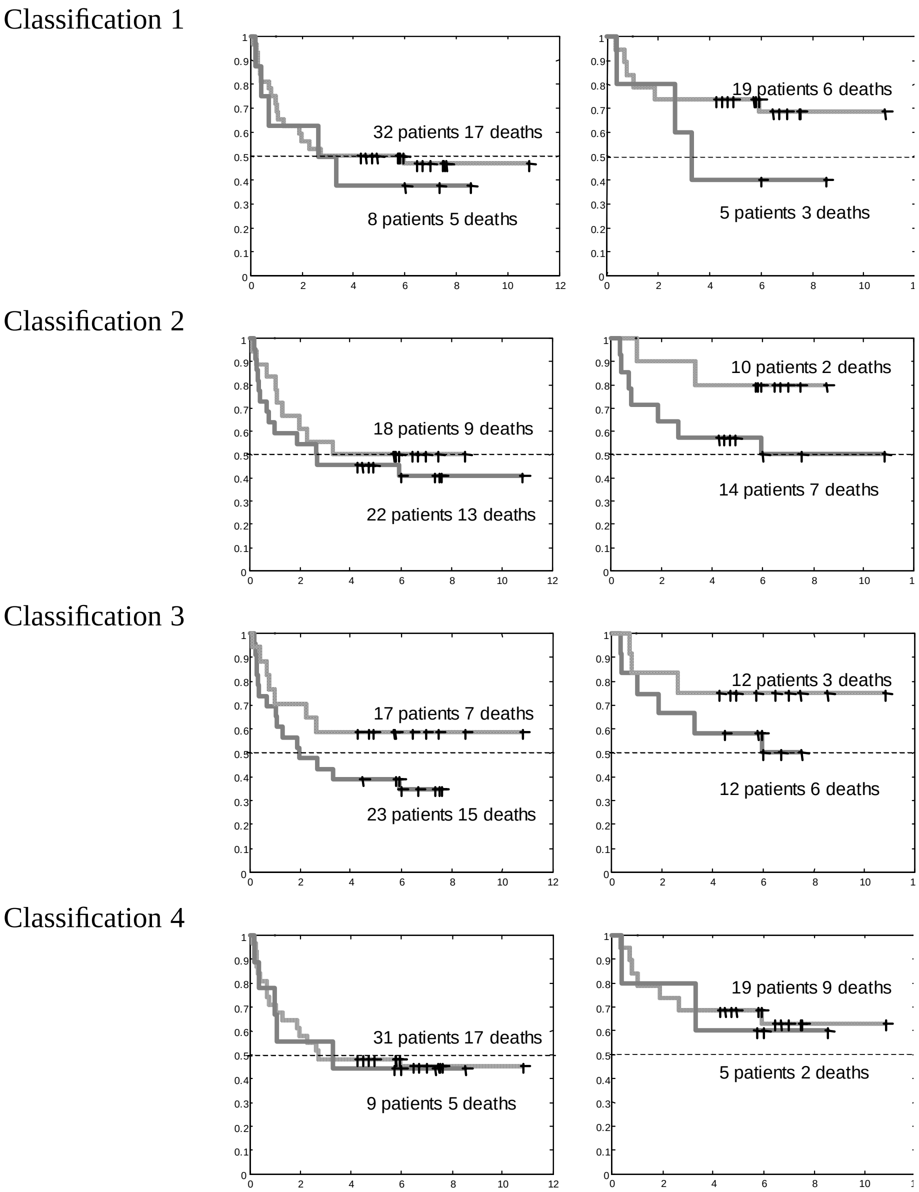


Figure 4: Kaplan-Meier survival plots for the 4 DLBCL classifications described in Table 1. The x-axis is the number of years after the samples were taken, and the y-axis is the fraction of patients survived so far. Each plot shows the survival rate for the two classes defined by a putative classification. The plots on the left column show the survival rate of all 40 patients for whom survival data is available. The plots on the right column, show the survival rate of the 24 patients with low clinical risk (see (Alizadeh et al. 2000) for details).