Spectral biclustering of microarray cancer data: co-clustering genes and conditions

Yuval Kluger[1,2], Ronen Basri[3], Joseph T. Chang[4], Mark Gerstein[2]

[1]Department of Genetics, Yale University, New Haven, CT

[2]Department of Molecular Biophysics and Biochemistry, Yale University, New Haven, CT

[3]Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot, Israel

[4]Department of Statistics, Yale University, New Haven, CT

ABSTRACT

Global analyses of RNA expression levels are useful for classifying genes and overall phenotypes. Often these classification problems are linked, and one wants to simultaneously find "marker genes" that are differentially expressed in particular “conditions”. We have developed a method that simultaneously clusters genes and conditions, finding distinctive "checkerboard" patterns in matrices of gene expression data, if they exist. In a cancer context, these checkerboards correspond to genes that are markedly up or down regulated in patients with particular types of tumors. Our method, spectral biclustering, is based on the observation that checkerboard structures in matrices of expression data can be found in eigenvectors corresponding to characteristic expression patterns across genes or conditions. Furthermore, these eigenvectors can be readily identified by commonly used linear-algebra approaches, in particular the singular value decomposition (SVD), coupled with closely integrated normalization steps. We present a number of variants of the approach, depending on whether the normalization over genes and conditions is done independently or in a coupled fashion. We then apply spectral biclustering to a selection of publicly available cancer expression data sets, and examine the degree to which it is able to identify checkerboard structures. Furthermore, we compare the performance of our biclustering methods against a number of reasonable benchmarks (e.g. direct application of SVD or normalized cuts to raw data).

INTRODUCTION

Microarray Analysis to Classify Genes and Phenotypes

Microarray experiments for simultaneously measuring RNA expression levels of thousands of genes are becoming widely used in genomic research. They have enormous promise in such areas as revealing function of genes in various cell populations, tumor classification, drug target identification, understanding cellular pathways and prediction of outcome to therapy (Brown and Botstein 1999; Lockhart and Winzeler 2000). A major application of microarray technology is gene expression profiling to predict outcome in multiple tumor types (Golub et al. 1999). In a bioinformatics context, we can apply various data-mining methods to cancer datasets in order to identify class distinction genes and to classify tumors. A partial list of methods includes: (i) data preprocessing (background elimination, identification of differentially expressed genes, and normalization); (ii) unsupervised clustering and visualization methods (hierarchical, SOM, k-means, and SVD); (iii) supervised machine learning methods for classification based on prior knowledge (discriminant analysis, support-vector machines, decision trees, neural networks, and k-nearest neighbors); and (iv) more

ambitious genetic network models (requiring large amounts of data) that are designed to discover biological pathways using such approaches as pairwise interactions, continuous or Boolean networks (based on a system of coupled differential equations) and probabilistic graph modeling based on Bayesian networks (Brown et al. 2000; Friedman et al. 2000; Tamayo et al. 1999).

Our focus here is on unsupervised clustering methods. Unsupervised techniques are useful when labels are unavailable. Examples include attempts to identify (yet unknown) sub-classes of tumors, or work on identifying clusters of genes that are co-regulated or share the same function (Brown et al. 2000; Mateos et al. 2002). Use of unsupervised methods is successful in separating certain types of tumors associated with different types of Leukemia and Lymphoma (Alizadeh et al. 2000; Golub et al. 1999; Klein et al. 2001). However, unsupervised (and even supervised methods) have had less success in partitioning the samples according to tumor type or outcome in diseases with multiple sub-classifications (Pomeroy et al. 2002; van 't Veer et al. 2002). In addition, the methods we propose here are related to one by Dhillon (Dhillon 2001) for co-clustering of words and document.

Checkboard Structures of Genes and Conditions in Microarray Datasets

As a starting point in analyzing microarray cancer datasets, it is worthwhile to appreciate the assumed structure of this data (e.g. whether it can be organized in a checkerboard pattern), and design a clustering algorithm that is suitable for this structure. In particular, in analyzing microarray cancer data sets we may wish to identify both clusters of genes that participate in common regulatory networks and clusters of experimental conditions associated with the effects of these genes, e.g., clusters of cancer subtypes. In both cases we may want to use similarities between expression level patterns to determine clusters. Clearly, advance knowledge of clusters of genes can help in clustering experimental conditions and vice versa. In the absence of knowledge of gene and condition classes, it would be attractive to develop partitioning algorithms that find latent classes by exploiting relations between genes and conditions. Exploiting the underlying two-sided data structure could help the simultaneous clustering, leading to meaningful gene and experimental condition clusters.

The raw data in many cancer gene-expression datasets can be arranged in a matrix form as schematized in figure 1. In this matrix, which we denote by A, the genes index rows i and the different conditions (e.g. different patients) index the columns j. Depending on the type of chip technology used, a value in this matrix A_{ij} could either represent absolute expression levels (such as from Affymetrix GeneChips) or relative expression ratios (such as from cDNA microarrays). The methodology we will construct will apply equally well in both contexts. However, for clarity in what follows, we will assume that the values A_{ij} in the matrix represent absolute levels and that all entries are non-negative (in our numerical analyses we removed genes that did not satisfy this criterion).

A specific assumption in tumor classification is that samples drawn from a population containing several tumor types have similar expression profiles, if they belong to the same type. This structure is also common to datasets from non-biological domains. Observing several experiments, each of which has multiple tumor types, suggests a somewhat stronger assumption: for tumors of the same type there exist subsets of over-expressed (or under-expressed) genes that are not similarly over-expressed (or under-expressed) in another tumor type. Under this assumption the matrix A could be organized in a checkerboard-like structure with blocks of high expression levels and low expression levels, as shown in figure 1. A block of high

expression levels corresponds to a subset of genes (subset of rows) that are highly expressed in all samples of a given tumor type (subset of columns). One of the numerous examples supporting this picture is the CNS embryonal tumors dataset (Pomeroy et al. 2002). However, this simple checkerboard-like structure can be confounded by a number of effects. In particular, different overall expression levels of genes across all experimental conditions or of samples across all genes in multiple tumor datasets can obscure the block structure. Consequently, rescaling and normalizing both the gene and sample dimensions could improve the clustering and reveal existing latent variables in both the gene and tumor dimensions.

Uncovering Checkerboard Structures through Solving an Eigenproblem

In this work, we attempt to simultaneously cluster genes and experimental conditions with similar expression profiles (i.e. to "bicluster" them), examining the extent to which we are able to automatically identify "checkerboard" structures in cancer datasets. Furthermore, we integrate biclustering with careful normalization of the data matrix in a spectral framework model. This framework allows us to use standard linear algebra manipulations, and the resulting partitions are generated using the whole dataset in a global fashion. The normalization step, which eliminates effects such as differences in experimental conditions and basal expression levels of genes, is designed to accentuate biclusters if they exist.

Figure 1 illustrates the overall idea of our approach. It shows how applying a checkerboard-structured matrix A to a step-like classification vector for genes x results in a step-like classification vector on conditions y. Reapplying the transpose of the matrix AT to this condition classification vectors results in a step-like gene classification vector with the same step pattern as input vector x. This suggests that one might be able to ascertain the checkerboard like structure of A through solving an eigenproblem involving AAT. More precisely, it shows how the checkerboard pattern in a data matrix A is reflected in the piecewise constant structures of some pair of eigenvectors x and y that solve the coupled eigenvalue problems $A^T Ax = \lambda^2 x$ and $AA^T y = \lambda^2 y$ (where x and y have a common eigenvalue). This, in turn, is equivalent to finding the singular value decomposition of A. Thus, the simple operation of identifying whether there exists a pair of piecewise constant eigenvectors allows us to determine whether the data has a checkerboard pattern. Simple reshuffling of rows and columns (according to the sorted order of these eigenvectors) then can make the pattern evident. However, different average amounts of expression associated with particular genes or conditions can obscure the checkerboard pattern. This can be corrected by initially normalizing the data matrix A. We propose a number of different schemes, all built around the idea of putting each gene on the same scale, so that it has the same average level of expression across conditions and, likewise for each condition. A graphical overview of our method (in application to real data) is shown in figure 8, where one can see how the data in matrix A is progressively transformed by normalization and shuffling to bring out a checkerboard-like signal.

Two properties of our method are that it implicitly exploits the effect of clustering of experimental conditions on clustering of the genes and vice versa and it allows us to simultaneously identify and organize subsets of genes whose expression levels are correlated and subsets of conditions whose expression level profiles are correlated.

TECHNICAL BACKGROUND

Data normalization

Preprocessing of microarray data often has a critical impact on the analysis. Several preprocessing schemes have been proposed. For instance, Eisen et al.(1998) prescribes the following series of operations: take the log of the expression data, perform 5 to 10 cycles of subtracting either the mean or the median of the rows (genes) and columns (conditions) and then do 5 to10 cycles of row-column normalization. In a similar fashion, Getz et al.(2000) first rescale the columns by their means and then standardize the rows of the rescaled matrix. The motivation is to remove systematic biases in expression ratios or absolute values that are the result of differences in RNA quantities, labeling efficiency and image acquisition parameters, as well as adjusting gene levels relative to their average behavior. Different normalization prescriptions could lead to different partitions of the data. Choice of a normalization scheme that is designed to emphasize underlying data structures or is rigorously guided by statistical principles is desirable for establishing standards and for improving reproducibility of results from microarray experiments.

Singular value decomposition (SVD)

Principal component analysis (PCA) (Pearson 1901) is widely used to project multidimensional data to a lower dimension. PCA determines if we can comprehensively present multidimensional data in *d* dimensions by inspecting whether *d* linear combinations of the variables capture most of the data variability. The principal components can be derived by using singular value decomposition, or "SVD," (Golub and Van Loan 1983), a standard linear algebra technique that expresses a real $n \times m$ matrix A as a product $A = U \Lambda V^T$, where Λ is a diagonal matrix with decreasing nonnegative entries, and U and V are $n \times \min(n,m)$ and $m \times \min(n,m)$ orthonormal column matrices. The columns of the matrices U and V are eigenvectors of the matrices AA^T and $A^T A$, respectively, and the nonvanishing entries $\lambda_1 \geq \lambda_2 \geq \cdots > 0$ in the matrix Λ are square roots of the non-zero eigenvalues of AA^T (and also of $A^T A$). Below we will denote the *i*th columns of the matrices U and V by u_i and v_i, respectively. The vectors u_i and v_i are called the *singular vectors* of A, and the values λ_i are called the *singular values*. The SVD has been applied to microarray experiment analysis in order to find underlying temporal and tumor patterns (Alter et al. 2000; Holter et al. 2000; Lian et al. 2001; Raychaudhuri et al. 2000).

Normalized Cuts Method

In addition, spectral methods have been used in graph theory to design clustering algorithms. These algorithms were used in various fields (Shi and Malik 1997), including for microarray data partitioning (Xing and Karp 2001). A commonly used variant is called the *normalized cuts* algorithm. In this approach the items (nodes) to be clustered are represented as the vertex set V**.** The degree of similarity (affinity) between each two nodes is represented by a weight matrix w_{ij}. For example, the affinity between two genes may be defined

based on the correlation between their expression profiles over all experiments. The vertex set V together with the edges $e_{ij} \in E$ and their corresponding weights w_{ij} define a complete graph $G(V,E)$ that we want to segment. Clustering is achieved by solving an eigen-system that involves the affinity matrix. These methods were applied in the field of image processing, and have demonstrated good performance in problems such as image segmentation. Nevertheless, spectral methods in the context of clustering are not well understood (Weiss 1999). We note that the singular values of the original dataset represented in the matrix A are related to the eigenvalues or generalized eigenvalues of the affinity matrices $A^T A$ and AA^T. These matrices represent similarities between genes and similarities between conditions respectively.

Previous work on biclustering

The idea of simultaneous clustering of rows and columns of a matrix goes back to (Hartigan 1972). Recently, methods for simultaneous clustering of genes and conditions have been proposed (Cheng and Church 2000; Getz et al. 2000; Lazzeroni and Owen 2002). The goal was to find homogeneous submatrices or stable clusters that are relevant for biological processes. These methods apply greedy iterative search to find interesting patterns in the matrices, an approach that is common also in one-sided clustering (Hastie et al. 2000; Stolovitzky et al. 2000). In contrast, our approach is more "global", finding biclusters using all columns and rows.

Another statistically motivated biclustering approach has been tested for collaborative filtering of non-biological data (Hofmann and Puzicha 1999; Ungar and Foster 1998). In this approach probabilistic models were proposed in which matrix rows (genes in our case) and columns (experimental conditions) are each divided into clusters, and there are link probabilities between these clusters. These link probabilities can describe the association between a gene cluster and an experimental condition cluster, and can be found by using iterative Gibbs sampling and approximated Expectation Maximization algorithms (Hofmann and Puzicha 1999; Ungar and Foster 1998).

A spectral approach to biclustering

Our aim is to have co-clustering of genes and experimental conditions in which genes are clustered together if they exhibit similar expression patterns across conditions and, likewise, experimental conditions are clustered together if they include genes that are expressed similarly. Interestingly, our model can be reduced to the analysis of the same eigensystem derived in Dhillon's formulation for the problem of co-clustering of words and documents (Dhillon 2001). To apply Dhillon's method to microarray data one can construct a bi-partite graph, where one set of nodes in this graph represents the genes, and the other represents experimental conditions. An arc between a gene and condition represents the level of over-expression (or under-expression) of this gene under this condition. The bi-partite approach is limited in that it can only divide the genes and conditions into the *same* number of clusters. This is often impractical. As described below, our formulation allows the number of gene clusters to be different from the number of condition clusters.

In addition, Dhillon's optimal partitioning eigenvector has a hybrid structure containing both gene and condition entries, whereas in our approach we search for separate piecewise constant structure of the gene and corresponding sample eigenvectors. Examining Dhillon's and our partitioning approaches using data generated by the generating model discussed below shows advantage of the latter.

SPECTRAL BICLUSTERING

We developed a method that simultaneously clusters genes and conditions. The method is based on the following two assumptions: (1) Two genes that are co-regulated are expected to have correlated expression levels, which might be difficult to observe due to noise. We can obtain better estimates of the correlations between gene expression profiles by averaging over different conditions of the same type. (2) Likewise, the expression profiles for every two conditions of the same type are expected to be correlated, and this correlation can be better observed when averaged over sets of genes of similar expression profiles.

These assumptions are supported by simple analyses of a variety of typical microarray sets. For example, Pomeroy et al. (2002) presented a dataset on five types of brain tumors, and then used a supervised learning procedure to select genes that were highly correlated with class distinction. They based this work on the absolute expression levels of genes in 42 samples taken from these five types of tumors. Using this data, we measured the correlation between the expression levels of genes that are highly expressed in only one type of tumor, and found only moderate levels of correlation. However, if we instead average the expression levels of each gene over all samples of the same tumor type (obtaining vectors with five entries representing the averages of the five types of tumors), the partition of the genes based on correlation between the five-dimensional vectors is more apparent.

This data set well fits the specifications of our approach, which is geared to finding a "checkerboard-like structure", indicating that for each type of tumor there may be few characteristic subsets of genes that are either up-regulated or down regulated. To understand our method (figure 1), consider a situation in which an underlying class structure of genes and of experimental conditions exists. We model the data as a composition of blocks, each of which represents a gene-type–condition-type pairing, but the block structure is not immediately evident. Mathematically, the expression level of a specific gene i under a certain experimental condition j can be expressed as a product of three independent factors. The first factor, which we called the *hidden base expression level,* is denoted by E_{ij}. We assume that the entries of E within each block are constant. The second factor, denoted r_i, represents the tendency of gene i to be expressed under all experimental conditions. The last factor, denoted $c_{j,}$ represents the overall tendency of genes to be expressed under the respective condition. We assume the microarray expression data to be a noisy version of the product of these three factors.

Independent rescaling of genes and conditions

We assume that the data matrix A represents an approximation of the product of these three factors, $E_{ij,}$ $\rho_{i,}$ and χ_j. Our objective in the simultaneous clustering of genes and conditions is, given A, to find the underlying block structure of E. Consider two genes, i and k, which belong to a subset of similar genes. On average, according to this model, their expression levels under each condition should be related by a factor of $\rho_i/\rho_{k.}$ Therefore, if we normalize the two rows, i and k, in A, then on average they should be identical. The similarity between the expression levels of the two genes should be more noticeable if we take the mean of expression levels with respect to all conditions of the same type. This will lead to an eigenvalue problem, as is

shown next. Let R denote a diagonal matrix whose elements r_i (where $i=1,\ldots,n$) represent the row sums of A ($R = diag(A1_n)$, 1_n denotes the n-vector $(1,\ldots,1)$). Let $u = (u^1, u^2, \ldots, u^m)$ denote a "classification vector" of experimental conditions, so that u is constant over all conditions of the same type, For instance, if there are two types of conditions then $u^j = \alpha$ for each condition j of the first type and $u^j = \beta$ for each condition j of the second type. In other words, if we reorder the conditions such that all conditions of the first type appear first then $u=(\alpha,\ldots,\alpha,\beta,\ldots\beta)$. Then, $v=R^{-1}Au$ is an estimate of a "gene classification vector," that is a vector whose entries are constant for all genes of the same type (e.g., if there are two types of genes then $v_i=\gamma$ for each gene i of the first type and $v_i=\delta$ for each gene i of the second type). By multiplying by R^{-1} from the left we normalize the rows of A, and by applying this normalized matrix to u we obtain a weighted sum of estimates of the mean expression level of every gene i under every type of experimental condition. When a hidden block structure exists for every pair of genes of the same type, these linear combinations are estimates of the same value.

The same reasoning applies to the columns. If we now apply $C^{-1}A^{\mathrm{T}}v$, where C is the diagonal matrix whose components are the column sums of A ($C = diag(1_m^T A)$), C^{-1} normalizes the columns of A, and by applying $C^{-1}A^{\mathrm{T}}$ to v, we obtain for each experimental condition j a weighted sum of estimates of the mean expression level of genes of the same type. Consequently, the result of applying the matrix $C^{-1}A^{\mathrm{T}}\,R^{-1}A$ to a condition classification vector, v, should also be a condition classification vector. We will denote this matrix by M_1. M_1 has a number of characteristics: it is positive semi-definite; it has only real non-negative eigenvalues; and its dominant eigenvector is $\frac{1}{\sqrt{m}}1_m$ with eigenvalue 1. Moreover, assuming E has linearly independent blocks, its rank is at least $\min(n_r, n_c)$, where n_r denotes the number of gene classes and n_c denotes the number of experimental condition classes. (In general the rank would be higher due to noise.) Note that for data with n_c classes of experimental conditions, the set of all classification vectors spans a linear subspace of dimension n_c. (This is because a classification vector may have a different constant value for each of the n_c types of experimental conditions.) Therefore, there exists at least one vector that satisfies $M_1u=\lambda u$. (In fact, there are exactly $\min(n_r, n_c)$ such vectors). One of these eigenvectors is the trivial vector $\frac{1}{\sqrt{m}}1_m$. Similarly, there exists at least one gene classification vector that satisfies $M_2v=\lambda v$, with $M_2=R^{-1}AC^{-1}A^{\mathrm{T}}$. (Note that M_1 and M_2 have the same sets of eigenvalues such that if $M_1u=\lambda u$ then $M_2v=\lambda v$ with $v=R^{-1}Au$.) These classification vectors can be estimated by solving the two eigen-systems above. A roughly piecewise constant structure in the eigenvectors indicates the clusters of both genes and conditions in the data.

These two eigenvalue problems can be solved through a standard SVD of the rescaled matrix $\hat{A} \equiv R^{-1/2}AC^{-1/2}$, realizing that the equation $\hat{A}^T\hat{A}w \equiv C^{-1/2}A^TR^{-1}AC^{-1/2}w = \lambda w$ that is used to find the singular values of $\hat{A}$ is equivalent to the above eigenvalue problem $C^{-1}A^{\mathrm{T}}\,R^{-1}Au=\lambda u$ with $u \equiv C^{-1/2}w$ (and similarly $\hat{A}\hat{A}^Tz \equiv R^{-1/2}AC^{-1}A^TR^{-1/2}z = \lambda z$ implies $v \equiv R^{-1/2}z$). The outer product $1_n1_m^T$, which is a matrix containing only entries of one, is the contribution of the first singular value to the rescaled matrix $\hat{A}$. Thus, the first eigenvalue contributes a *constant* background to both the gene and the experimental condition dimensions, and therefore its effect should be eliminated. Note that although our method is defined through a product of A and A^{T} it does not imply that we multiply the noise, as is evident from the SVD interpretation.

Simultaneous normalization of genes and conditions

Because our spectral biclustering approach includes the normalization of rows and columns as an integral part of the algorithm, it is natural to attempt to simultaneously normalize both genes and conditions. As described below, this can be achieved by repeating the procedure described above for independently scaling of rows and columns iteratively until convergence.

This process, which we call *bi-stochastization*, results in a rectangular matrix B that has a doubly stochastic-like structure – all rows sum to a constant and all columns sum to a different constant. According to Sinkhorn's theorem, B can then be written as a product $B=D_1AD_2$ where D_1 and D_2 are diagonal matrices (Bapat and Raghavan 1997). Such a matrix B exists under quite general conditions on A; for example, it is sufficient for all of the entries in A to be positive. In general B can be computed by repeated normalization of rows and columns (with the normalizing matrices as R^{-1} and C^{-1} or $R^{-1/2}$ and $C^{-1/2}$). D_1 and D_2 then will represent the product of all these normalizations. Fast methods to find D_1 and D_2 include the deviation reduction and balancing algorithms (Bapat and Raghavan 1997). Once D_1 and D_2 are found, we apply SVD to B with no further normalization to reveal a block structure.

We have also investigated an alternative to bi-stochastization that we call the *log-interactions* normalization. A common and useful practice in microarray analysis is transforming the data by taking logarithms. The resulting transformed data typically has better distributional properties than the data on the original scale – distributions are closer to Normal, scatterplots are more informative, and so forth. The log-interactions normalization method begins by calculating the logarithm $L_{ij} = \log(A_{ij})$ of the given expression data and then extracting the *interactions* between the genes and the conditions, where the term "interaction" is used as in the analysis of variance (ANOVA).

As above, the log-interactions normalization is motivated by the idea that two genes whose expression profiles differ only by a multiplicative constant of proportionality are really behaving in the same way, and we would like these genes to cluster together. In other words, after taking logs, we would like to consider two genes whose expression profiles differ by an additive constant to be equivalent. This suggests subtracting a constant from each row so that the row means each become 0, in which case the expression profiles of two genes that we would like to consider equivalent actually become the same. Likewise, the same idea holds for the conditions (columns of the matrix). Constant differences in the log expression profiles between two conditions are considered unimportant, and we subtract a constant from each column so that the column means become 0. It turns out that these adjustments to the rows and columns of the matrix to achieve row and column means of zero can all be done simultaneously by a simple formula. Defining $\bar{L}_{i\cdot} = \frac{1}{m}\sum_{j=1}^{m} L_{ij}$ to be the average of the ith row, $\bar{L}_{\cdot j} = \frac{1}{n}\sum_{i=1}^{n} L_{ij}$ to be the average of the jth column, and $\bar{L}_{\cdot\cdot} = \frac{1}{mn}\sum_{i=1}^{n}\sum_{j=1}^{m} L_{ij}$ to be the average of the whole matrix, the result of these adjustments is a matrix of *interactions* $K = (K_{ij})$, calculated by the formula $K_{ij} = L_{ij} - \bar{L}_{i\cdot} - \bar{L}_{\cdot j} + \bar{L}_{\cdot\cdot}$. This formula is familiar from the study of two-way ANOVA, from which the terminology of "interactions" is adopted. The interaction K_{ij} between gene i and condition j captures the extra (log) expression of gene i in condition j that is not explained simply by an overall difference between gene i and other genes or between condition j and other conditions, but rather is special to the

combination of gene i with condition j. Again, as described before, we apply the SVD to the matrix K to reveal block structure in the interactions.

The calculations to obtain the interactions are simpler than the bistochastization, as they are done by a simple formula with no iteration. In addition, in this normalization the first singular eigenvectors u_1 and v_1 may carry important partitioning information. Therefore we do not automatically discard them as was done in the previously discussed normalizations. Finally, we note another connection between matrices of interactions and matrices resulting from bistochastization. Starting with a matrix of interactions K, we can produce a bistochastic matrix simply by adding a constant to K.

Post processing the eigenvectors to find partitions

Each of the above normalization approaches (independent scaling , bistochastization, or log interactions) gives rise, after the SVD, to a set of gene and condition eigenvectors (that in the context of microarray analysis are sometimes termed eigengenes and eigenarrays(Alter et al. 2000; Hastie et al. 1999)). Now in this section, we deal with the issues of how to interpret these vectors. First recall that in the case of the first two normalizations we discussed (the independent and bistochastic rescaling), we discard the largest eigenvalue, which is trivial in the sense that its eigenvectors make a trivial constant contribution to the matrix, and therefore carry no partitioning information. In the case of the log-interactions normalization, there is no eigenvalue that is trivial in this sense. We will use the terminology "largest eigenvalue" to mean the largest nontrivial eigenvalue, which, for example, is the second largest eigenvalue for the independent and bi-stochastic normalizations, whereas it is the largest eigenvalue for the log-interactions normalization. If the dataset has an underlying "checkerboard" structure, there is at least one pair of piecewise constant eigenvectors u and v that correspond to the same eigenvalue. One would expect that the eigenvectors corresponding to the largest nontrivial eigenvalue would provide the optimal partition in analogy with related spectral approaches to clustering (e.g. Shi & Malik, (1997)). In principle, the classification eigenvectors may not belong to the largest nontrivial eigenvalue, and we closely inspect a few eigenvectors that correspond to the first largest eigenvalues. We observed that for various synthetic data with near to perfect checkerboard-like block structure, the partitioning eigenvectors are commonly associated with one of the largest eigenvalues, but in a few cases an eigenvector with a small eigenvalue could be the partitioning one. (This occurs typically when the separation between blocks in E is smaller than the standard deviation within a block.) In order to extract partitioning information from these eigen-systems, we examine all the eigenvectors by fitting them to piecewise constant vectors. This is done by sorting the entries of each eigenvector, testing all possible thresholds, and choosing the eigenvector with a partition that is well approximated by a piecewise constant vector. (Selecting one threshold partitions the entries in the sorted eigenvector into two subsets, two thresholds into three subsets, and so forth.) Note that to partition the eigenvector into two, one needs to consider n-1 different thresholds, to partition it into three, it requires inspection of (n-1)(n-2)/2 different thresholds and so on. This procedure is similar to application of the k-means algorithm to the one-dimensional eigenvectors. (In particular, in the experiments below we performed this procedure automatically to the six most dominant eigenvectors.) A common practice in spectral clustering is to perform a final clustering step to the data projected to a small number of eigenvectors, instead of clustering each eigenvector individually (Shi and Malik 1997). In our experiments we too perform a final clustering step by applying both the k-means and the normalized cuts algorithms to the data projected to the best two or three eigenvectors.

Our clustering method provides not only a division into clusters, but also ranks the degree of membership of genes (and conditions) to the respective cluster according to the actual values in the partitioning sorted eigenvectors. Each partitioning sorted eigenvector could be approximated by a step-like (piecewise constant) structure, but the values of the sorted eigenvector within each step are monotonically

decreasing. These values can be used to rank or represent gradual transitions within clusters. Such rankings may also be useful, e.g., for revealing genes related to pre-malignant conditions, and for studying ranking of patients within a disease cluster in relation to prognosis.

In addition to the uses of biclustering as a tool for data visualization and interpretation, it is natural to ask how to assess the quality of biclusters, in terms of statistical significance, or stability. In general, this type of problem is far from settled; in fact, even in the simpler setting of ordinary clustering new efforts to address these questions regularly continue to appear. One type of approach attempts to quantify the "stability" of suspected structure observed in the given data. This is done by mimicking the operation of collecting repeated independent data samples from the same data-generating distribution, repeating the analysis on those artificial samples, and seeing how frequently the suspected structure is observed in the artificial data. If the observed data contains sufficient replication, then the bootstrap approach of (Kerr and Churchill 2001) may be applied to generate the artificial replicated data sets. However, most experiments still lack the sort of replication required to carry this out. For such experiments, one could generate artificial data sets by adding random noise (Bittner et al. 2000) or subsampling (Ben-Hur et al. 2002) the given data.

We took an alternative approach to assess the quality of a biclustering by testing a null hypothesis of no structure in the data matrix. We first normalized the data and used the best partitioning pair of eigenvectors (among the six leading eigenvectors) to determine an approximate 2x2 block solution. We then calculated the sum of squared errors (SSE) for the least-squares fit of these blocks to the normalized data matrix. Finally, to assess the quality of this fit we randomly shuffled the data matrix and applied the same process to the shuffled matrix. For example, in the breast cell oncogene data set described below, fitting the normalized dataset to a 2x2 matrix obtained by division according to the second largest pair of eigenvectors of the original matrix is compared to fitting of 10000 shuffled matrices (after bi-stochastisation) to their corresponding best 2x2 block approximations. The SSE for this dataset is more than 100 standard deviations smaller than the mean of the SSE scores obtained from the shuffled matrices, leading to a correspondingly tiny *P* value for the hypothesis test of randomness in the data matrix.

Probabilistic Interpretation

In the biclustering approach, the normalization procedure, obtained by constraining the row sums to be equal to one constant and the column sums to be equal to another constant, is an integral part of the modeling that allows us to discern bi-directional structures. This normalization can be cast in probabilistic terms by imagining first choosing a random RNA transcript from all RNA in all samples (conditions), and then choosing one more RNA transcript randomly from the same sample. Here, when we speak of choosing "randomly" we mean that each possible RNA is equally likely to be chosen. Having chosen these two RNA's, we take note of which sample they come from and which genes they express. The matrix entry $(R^{-1}A)_{ij}$ may be interpreted as the conditional probability $p_{s|g}(j \mid i)$ that the sample is *j*, given that the first RNA chosen was transcribed from gene *i*. Similarly, $(C^{-1}A^T)_{jk}$ may be interpreted as the conditional probability $p_{g|s}(k \mid j)$ that the gene corresponding to the first transcript is *k*, given that the sample is *j*. Moreover, the product of the row-normalized matrix and the column-normalized matrix approximates the conditional

probability $p_{g|g}(i\,|\,k)$ of choosing a transcript from gene *i*, given that we also chose one from gene *k*. (Under the assumption that *k* and *i* are approximately conditionally independent given *j,* which amounts to saying that the probability of drawing a transcript from gene *k*, conditional on having chosen sample *j*, does not depend on whether or not the other RNA that we drew happened to be from gene *i*,

$$p_{g|g}(k\,|\,i)=\sum_j p_{s|g}(j\,|\,i)\,p_{g|sg}(k\,|\,j,i)\approx\sum_j p_{s|g}(j\,|\,i)\,p_{g|s}(k\,|\,j)=\left((R^{-1}A)(C^{-1}A^T)\right)_{ik}$$

This expression reflects the tendency of genes *i* and *k* to co-express, averaged over the different samples. Similarly, the product of the column and row-normalized matrices approximates the conditional probability $p_{s|s}(j|l)$ that reflects the similarity between the expression profiles of samples *j* and *l* . Note that the probabilities $p_{g|g}(i|k)$ and $p_{s|s}(j|l)$ define asymmetrical affinity measures between any pair (*i*,*k*) of genes and any pair (*j*,*l*) of samples respectively. Note, this is very different from the usual symmetrical affinity measures, e.g. correlation, used to describe the relationship between genes. However, for bistochastizaton, the matrices B^TB and BB^T represent *s*ymmetrical affinities, $p_{g|g}(i|k)= p_{g|g}(k|i)$ and $p_{s|s}(j|l)= p_{s|s}(l|j)$ respectively.

RESULTS

Overall Format of the Results

We have performed a study in which we have applied the above spectral biclustering methods to five groups of cancer microarray data sets -- lymphoma (microarray and Affymetrix), leukemia, breast cancer and central nervous system embryonal tumors. As explained above, we utilize SVD to find pairs of piecewise constant eigenvectors of genes and conditions, that reflect the degree to which the data can be rearranged in a checkerboard structure. Our methods employ specific normalization schemes that highlight the similarity of both gene and condition eigenvectors to piecewise constant vectors, and this similarity, in turn, directly reflects the degree of biclustering. To assess our procedure, it is useful to see how well it compares to several benchmarks, with respect to achieving the goal of piecewise constant eigenvectors.

Our main results are presented in Figures 3 to 7. These show consistently formatted graphs of the projection of each dataset onto the best two eigenvectors. Each figure is laid out in 6 panels with the first two subpanels associated with our biclustering methods and the following four subpanels showing the benchmarks. In particular:

Subpanel a ("bistochastization") shows biclustering using the biostochastic normalization.

Subpanel b ("biclustering") shows standard biclustering with independent rescaling of rows and columns.

Subpanel c ("SVD") shows SVD applied to the raw data matrix *A*.

Subpanel d ("bi-normalization") shows SVD applied to a transformed matrix obtained by first rescaling its columns by their means and then standardizing the rows of the rescaled matrix as proposed in Getz et al. (2000).

Subpanel e ("normalized cuts") shows a normalized cuts benchmark. Here we apply the normalized cuts algorithm using an affinity matrix obtained from a distance matrix, which, in turn, was derived by calculating the norms of the differences between the standardized columns of *A* as proposed in Xing & Karp (2001). (See caption of figure 3 for more details.) Moreover, we applied the normalized cuts algorithm to an affinity matrix constructed from the column-rescaled row-standardized matrix (Getz et al. 2000), as in subpanel d. We then examined whether a partition is visible in the eigenvectors that correspond to the second

largest eigenvalue (which in the normalized cuts case are supposed to provide approximation of the optimal partition) and in the subspace spanned by two or three eigenvectors with the best proximity to piecewise constant vectors.

Subpanel f ("log-interaction") shows SVD applied to a matrix where the raw expression data is substituted by the matrix K described above.

Overall, by comparing the 6 subpanels in each of the 5 different figures, we see that in the bi-stochastization method (subpanel a) the distributions of the different samples have no or minimal overlap between clusters as well as more tendency to result in more compact clusters. The biclustering method (subpanel b) results in slightly less separable clusters, but it tends to separate the clusters along a single eigenvector. Straight SVD of the different raw data (subpanel c) under performs in comparison to our spectral methods, as can be seen from the intermingled distributions of tumors of different types or less distinct clusters. Performing instead SVD on the log-interaction matrix of the raw expression data tends to produce results that are similar to those obtained with bi-stochastization. (subpanel f). SVD of the column-rescaled row-standardized matrix (Getz et al. 2000) and the normalized cut method result in better partitioning than SVD of the raw data (subpanels d and e). However, in general our spectral methods consistently perform well.

In the following sections we discuss each of the five datasets in detail.

Lymphoma microarray dataset

We first applied the methods to publically available lymphoma microarray data (CLL, FL, DLCL)[1]. The clustering results are shown in Figures 2 and 3. In both cases when we used the doubly stochastic-like matrix B or the biclustering method ($C^{-1}A^{T}R^{-1}A$) of the lymphoma dataset we obtained the desired partitioning of patients in the second largest eigenvectors. The sorted eigenvectors give not only a partition of patients, but also an internal ranking of patients within a given disease. In addition, the outer product of the gene and tumor (sorted) eigenvectors allows us to observe which genes induce a partition of patients and vice versa. This can be seen in Fig. 2. Dividing the eigenvector that corresponds to the second largest eigenvalue (in both methods) using the k-means algorithm (which is equivalent to fitting a piecewise constant vector to each of the eigenvectors) led to a clean partition between the DLCL patients and the patients with other diseases. This is highlighted in the header of Fig. 2 and the x-axis of Fig. 3(a) and (b). The published analysis did not cluster two of the DLCL cases correctly (Alizadeh et al. 2000). Further partitioning of the CLL and the FL patients is obtained by using both the second- and third-largest eigenvectors. To divide the data we applied a recursive, two-way clustering using the normalized cuts algorithm to a two-column matrix composed of the 2nd and 3rd eigenvectors of both matrices. (Performing a final clustering step to the data projected to a small number of eigenvectors is a common practice in spectral clustering.) Using the biclustering matrix with independent row and column normalizations, the patients were correctly divided, with the exception of two of the CLL patients who were clustered together with the FL patients. The best partition was obtained using our doubly stochastic matrix that divided the patients perfectly according to the three types of diseases.

[1] Chronic lymphocytic leukemia (CLL), diffuse large B-cell lymphoma (DLCL), follicular lymphoma (FL)

Lymphoma Affymetrix dataset

The above lymphoma data was generated by microarray technology that provides relative measurements of expression data. We repeated the lymphoma analysis using data from a study relating B-CLL to memory B cells (Klein et al. 2001). This data was generated using Affymetrix U95A gene chips, which presumably allow measurements proportional to absolute mRNA levels. We selected samples taken from CLL, FL and DLCL patients, but in addition we also included samples from DLCL cell lines. As can be seen in Figs. 4(a) and 4(b) the bi-stochastization method cleanly separates the four different sample types and the biclustering separates these samples except one DLCL sample that slightly overlaps with the FL distribution. We note that the DLCL patient expression patterns are closer to those of the FL patients than to the expression profiles of the DLCL cell lines (and $p_{g|g}$(DLCL|FL)> $p_{g|g}$(DLCL|DLCL-cell lines).

Leukemia dataset

We applied our methods to public microarray data of acute leukemia (B and T cell ALL and AML)[2]. The patient distributions of the different diseases of the leukemia dataset become separated in the two dimensional graphs generated by projecting the patient expression profiles onto the 2nd and 3rd gene class partition vectors of the biclustering method (Fig. 5(b)). The bistochastic method also partitions the patients well, with only one ambiguous case that is close to the boundary between ALL and AML (Fig. 5(a)). Application of k-means to a matrix composed of the 2nd and 3rd biclustering eigenvectors results in three misclassifications, which is a slight improvement over the four misclassifications reported in Golub et al. (1999). Further partitioning of the ALL cases is obtained by applying a normalized cuts clustering method to the biclustering eigenvectors, and produces a clear separation between T and B cell ALL. This is a slight improvement over published results (two misclassifications) (Getz et al. 2000; Golub et al. 1999). Another advantage over their methods is that biclustering does not require specification of the number of desired clusters or lengthy searches for subsets of genes.

Dataset from Breast cell lines transfected with the CSF1R oncogene

In another microarray experiment study (Kluger et al. 2001), an oncogene encoding a transmembrane tyrosine kinase receptor was mutated at two different phosphorylation sites. Benign breast cells were transfected with the wild type oncogene, creating a phenotype that invades and metastasizes. The benign cell line was then transfected with the two mutated oncogenes, creating one phenotype that invades and another one that metastasizes. RNA expression levels were measured eight times for each phenotype. Transfection with a single oncogene is expected to generate similar expression profiles, presumably because only a few genes are biologically influenced. Therefore, it was desirable to see if profiles of the different phenotypes can be partitioned.

Figure 8 allows us to examine the extent to which the data can be arranged in a checkerboard pattern. This is done by taking the outer product of the cell type sorted eigenvector that has the most stepwise-like structure (and is associated with the first largest singular value) with the corresponding gene sorted eigenvector. Due to noise in the data and similarity between the different samples, common clustering techniques such as hierarchical, k-means and medoids did not succeed in cleanly partitioning the data, but the

[2] acute lymphocytic leukemia (ALL), acute myelogenous leukemia (AML)

relevant eigen-array obtained following bi-stochastization or log-interaction normalization partitioned the samples perfectly. Expression levels of the four cell lines were measured in two separate sets of four measurements. We chose to measure the ratio of three of the cell lines; benign (*a)*, invasive (*c)* and metastatic (*d)* with respect to the cell line that invades and metastasizes *(b)* in the first batch, and the corresponding ratios were similarly derived for the second batch. In Figs. 6 and 8 the ratios from the first and second batches are denoted by (*a, c, d*) and (*A, C, D*) respectively. As can be seen, the simultaneous normalization methods partition the data such that all the phenotypes are separated into clusters -- i.e. *a* were clustered with *A* in one group, *c* with *C* in another group and *d* with *D* in yet another group, as expected. Further exploration is required in order to relate those gene clusters to biological pathways that are relevant to these conditions.

Central nervous system embryonal tumor dataset

Finally, we analyzed the recently published CNS embryonal tumor dataset (Pomeroy et al. 2002). Pomeroy et al (Pomeroy et al. 2002) partitioned these five tumor types using standard principal component analysis, but after employing a pre-selection of genes exhibiting variation across the data set (see Fig 1(b) in (Pomeroy et al. 2002)). *Using all genes* we find that the bi-stochastization method, and to a lesser degree the biclustering method partitioned the medulloblastoma, malignant glioma, and normal cerebella tumors. As can be seen in Figure 7, the remaining rhabdoid tumors are more widely scattered in the subspace obtained by projecting the tumors onto the 2^{nd}-4^{th} gene partitioning eigenvectors of the biclustering and bi-stochastization methods. Nonetheless, the rhabdoid tumor distribution does not overlap with the other tumor distributions if we use the bi-stochastization method. The primitive neuro-ectodermal tumors (PNETs) did not cluster and were even hard to classify using supervised methods.

CONCLUSION

Unsupervised clustering of genes and experimental conditions in microarray data can potentially reveal genes that participate in cellular mechanisms that are involved in various diseases. In this paper we present a spectral bi-clustering method that utilizes the information gained by clustering the conditions to facilitate the clustering of genes and vice versa. The method incorporates a closely integrated normalization. It also naturally discards the irrelevant *constant* background, such that no additional arguments are needed to ignore the contribution associated with the largest eigenvalue as advocated in Alter et al. (2000). In particular, our method is designed to cluster populations of different tumors assuming that each tumor type has a subset of marker genes that exhibit over-expression and that typically are not over-expressed in other tumors. The main underlying assumption is that we can simultaneously obtain better tumor clusters and gene clusters by correlating genes averaged over different samples of the same tumors. Likewise, the correlation of two tumors is more apparent when averaged over sets of genes of similar expression profiles. In situations where the number of tumor types (the number of clusters of experimental conditions) is equal to the number of typical gene profiles (the number of gene clusters) the biclustering algorithm is related to the modified normalized cuts objective function introduced by Dhillon (Dhillon 2001). In addition, in a situation where the data has approximately a checkerboard structure with more than two clusters on each side, there may be several eigenvectors indicating a partitioning. In this case we may be able to determine the number of clusters by

identifying all these eigenvectors, e.g., using a pairwise measure such as mutual entropy between all pairs of eigenvectors.

The methods presented in this paper, particularly those incorporating simultaneous normalization of rows and columns show consistent advantage over SVD spectral analysis of the raw data, the logarithm of the raw data, other forms of rescaling transformations of the raw data and the normalized cuts partitioning of the raw or rescaled data. Nevertheless, our partitioning results are not perfect. Better results may be obtained by employing a generative model that better suits the data. It has been shown that removal of irrelevant genes that introduce noise can further improve clustering (as in (Xing and Karp 2001)). Furthermore, if partitioning in the gene dimension is sharper than partitioning in the condition dimension or vice versa, we can organize the conditions or genes of the blurrier dimension contiguously. Such arrangements perhaps give one a sense of the progression of disease states or relevance of a gene to a particular disease.

Acknowledgments
Y.K. is supported by the Cancer Bioinformatics Fellowship from the Anna Fuller Fund and M.G. acknowledges support from Human Genome array: Technology for Functional Analysis (an NIH grant number P50 HG02357-01)

References

Alizadeh, A.A., M.B. Eisen, R.E. Davis, C. Ma, I.S. Lossos, A. Rosenwald, J.C. Boldrick, H. Sabet, T. Tran, X. Yu, J.I. Powell, L. Yang, G.E. Marti, T. Moore, J. Hudson, Jr., L. Lu, D.B. Lewis, R. Tibshirani, G. Sherlock, W.C. Chan, T.C. Greiner, D.D. Weisenburger, J.O. Armitage, R. Warnke, L.M. Staudt, and et al. 2000. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature* **403:** 503-511.

Alter, O., P.O. Brown, and D. Botstein. 2000. Singular value decomposition for genome-wide expression data processing and modeling. *PNAS* **97:** 10101-10106.

Bapat, R.B. and T.E.S. Raghavan. 1997. *Nonnegative Matrices and Applications*. Cambridge University Press.

Ben-Hur, A., A. Elisseeff, and I. Guyon. 2002. A stability based method for discovering structure in clustered data. *Pac Symp Biocomput***:** 6-17.

Bittner, M., P. Meltzer, Y. Chen, Y. Jiang, E. Seftor, M. Hendrix, M. Radmacher, R. Simon, Z. Yakhini, A. Ben-Dor, N. Sampas, E. Dougherty, E. Wang, F. Marincola, C. Gooden, J. Lueders, A. Glatfelter, P. Pollock, J. Carpten, E. Gillanders, D. Leja, K. Dietrich, C. Beaudry, M. Berens, D. Alberts, and V. Sondak. 2000. Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature* **406:** 536-540.

Brown, M.P.S., W.N. Grundy, D. Lin, C. Sugnet, J.M. Ares, and D. Haussler. 2000. Knowledge-based analysis of microarray gene expression data by using support vector machines. *PNAS* **97:** 262-267.

Brown, P.O. and D. Botstein. 1999. Exploring the new world of the genome with DNA microarrays. *Nat Genet* **21:** 33-37.

Cheng, Y. and G.M. Church. 2000. Biclustering of expression data. In *ISMB'00*.

Dhillon, I.S. 2001. Co-Clustering Documents and Words Using Bipartite Spectral Graph Partitioning. In *Proceedings of the Seventh ACM SIGKDD Conference*, San Francisco.

Eisen, M., P.T. Spellman, P.O. Brown, and D. Botstein. 1998. Cluster analysis and display of genome-wide expression patterns. *PNAS* **95:** 14863-14868.

Friedman, N., M. Linial, I. Nachman, and D. Pe'er. 2000. Using Bayesian networks to analyze expression data. *Journal of Computational Biology* **7:** 601-620.

Getz, G., E. Levine, and E. Domany. 2000. Coupled two-way clustering analysis of gene microarray data. *Proceedings of the National Academy of Sciences* **97:** 12079-12084.

Golub, G.H. and C.F. Van Loan. 1983. *Matrix Computations*. Johns Hopkins University Press, Baltimore.

Golub, T.R., D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M. Loh, J.R. Downing, M. Caligiuri, C.D. Bloomfield, and E.S. Lander. 1999. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science* **286:** 531-537.

Hartigan, J.A. 1972. Direct clustering of a data matrix. *J. Amer. Statist. Assoc.* **67:** 123-129.

Hastie, T., R. Tibshirani, M.B. Eisen, A. Alizadeh, R. Levy, L. Staudt, W.C. Chan, D. Botstein, and P.O. Brown. 2000. 'Gene shaving' as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biology* **1:** research0003.0001-0003.0021.

Hastie, T., R. Tibshirani, G. Sherlock, M. Eisen, P.O. Brown, and D. Botstein. 1999. Imputing Missing Data for Gene Expression Arrays. Stanford Statistics Department.

Hofmann, T. and J. Puzicha. 1999. Latent Class Models for Collaborative Filtering. In *Proceedings of the International Joint Conference in Artificial Intelligence*.

Holter, N.S., M. Mitra, A. Maritan, M. Cieplak, J.R. Banavar, and N.V. Fedoroff. 2000. Fundamental patterns underlying gene expression profiles: simplicity from complexity. *PNAS* **97:** 8409-8414.

Kerr, M.K. and G.A. Churchill. 2001. Bootstrapping cluster analysis: assessing the reliability of conclusions from microarray experiments. *Proc Natl Acad Sci U S A* **98:** 8961-8965.

Klein, U., Y. Tu, G.A. Stolovitzky, M. Mattioli, G. Cattoretti, H. Husson, A. Freedman, G. Inghirami, L. Cro, L. Baldini, A. Neri, A. Califano, and R. Dalla-Favera. 2001. Gene expression profiling of B cell chronic lymphocytic leukemia reveals a homogeneous phenotype related to memory B cells. *J Exp Med* **194:** 1625-1638.

Kluger, H., B. Kacinski, Y. Kluger, O. Mironenko, M. Gilmore-Hebert, J. Chang, A.S. Perkins, and E. Sapi. 2001. Microarry analysis of invasive and metastatic

phenotypes in a breast cancer model. In *poster presented at the Gordon Conference on Cancer*, Newport, RI.

Lazzeroni, L. and A. Owen. 2002. Plaid models for gene expression data. *Statistica Sinica* **12:** 61-86.

Lian, Z., L. Wang, S. Yamaga, W. Bonds, Y. Beazer-Barclay, Y. Kluger, M. Gerstein, P.E. Newburger, N. Berliner, and S.M. Weissman. 2001. Genomic and proteomic analysis of the myeloid differentiation program. *Blood* **98:** 513-524.

Lockhart, D.J. and E.A. Winzeler. 2000. Genomics, gene expression and DNA arrays. *Nature* **405:** 827-836.

Mateos, A., J. Dopazo, R. Jansen, Y. Tu, M. Gerstein, and G. Stolovitzky. 2002. Systematic learning of gene functional classes from DNA array expression data by using multilayer perceptrons. *Genome Res* **12:** 1703-1715.

Pearson, K. 1901. On Lines and Planes of Closest Fit to Systems of Points in Space. *Phil. Mag.* **2:** 559-572.

Pomeroy, S.L., P. Tamayo, M. Gaasenbeek, L.M. Sturla, M. Angelo, M.E. McLaughlin, J.Y. Kim, L.C. Goumnerova, P.M. Black, C. Lau, J.C. Allen, D. Zagzag, J.M. Olson, T. Curran, C. Wetmore, J.A. Biegel, T. Poggio, S. Mukherjee, R. Rifkin, A. Califano, G. Stolovitzky, D.N. Louis, J.P. Mesirov, E.S. Lander, and T.R. Golub. 2002. Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature* **415:** 436-442.

Raychaudhuri, S., J.M. Stuart, and R.B. Altman. 2000. Principal components analysis to summarize microarray experiments: application to sporulation time series. In *PSB00*, pp. 452-463.

Shi, J. and J. Malik. 1997. Normalized cuts and image segmentation. In *IEEE Conf. Computer Vision and Pattern Recognition*, pp. 731-737.

Stolovitzky, G., A. Califano, and Y. Tu. 2000. Analysis of gene expression microarrays for phenotype classification. In *ISMB'00*.

Tamayo, P., D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lander, and T.R. Golub. 1999. Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. *Proc Natl Acad Sci U S A* **96:** 2907-2912.

Ungar, L. and A. Foster. 1998. A formal statistical approach to collaborative filtering. In *Conference on Automated Learning and Discovery CONALD'98*, CMU.

van 't Veer, L.J., H. Dai, M.J. van de Vijver, Y.D. He, A.A. Hart, M. Mao, H.L. Peterse, K. van der Kooy, M.J. Marton, A.T. Witteveen, G.J. Schreiber, R.M. Kerkhoven, C. Roberts, P.S. Linsley, R. Bernards, and S.H. Friend. 2002. Gene expression profiling predicts clinical outcome of breast cancer. *Nature* **415:** 530-536.

Weiss, Y. 1999. Segmentation using eigenvectors: a unifying view. In *Proceedings IEEE International Conference on Computer Vision*, pp. 975-982.

Xing, E.P. and R.M. Karp. 2001. CLIFF: clustering of high-dimensional microarray data via iterative feature filtering using normalized cuts. In *ISMB'01*.

Figure Captions

Figure 1 This figure gives an overview of important parts of the biclustering process. Part A shows the problem: shuffling a gene expression matrix to reveal a checkerboard pattern associating genes with conditions. Part B shows how this problem can be approached through solving an "eigenproblem." If a gene expression matrix A has a checkerboard structure, applying it to a step-like condition classification vector x will result in a step-like gene classification vector y. Moreover, if one then applies A^T to y, one will regenerate a step-like condition classification vector with the same partitioning structure as x. This suggests one can determine if A has a checkerboard structure through solving an eigenvalue problem. In other words, if A has a (hidden) checkerboard structure there exist some piecewise constant partition vectors $x = v_*$ and $y = u_*$ such that $A^T A v_* = \lambda^2 v_*$ and $AA^T u_* = \lambda^2 u_*$ (bottom quadrant of part B). Note that most eigenvectors v of the eigenvalue problem $A^T A v = \lambda^2 v$ (symbolized by a zigzag structure) are not embedded in the subspace of classification (step-like) vectors x possessing the same partitioning structure, as indicated by a gray arrow protruding from this subspace (parallelogram). On the other hand piecewise constant (step-like) partition eigenvectors v_* are embedded in this subspace and indicated by a green arrow. To reveal whether the data has checkerboard structure one can inspect if some of the pairs of monotonically sorted gene and tumor eigenvectors v_i and u_i have an approximate stepwise (piecewise) constant structure. The outer product $u_* v_*^T$ of the sorted partitioning eigenvectors gives a checkerboard structure. Part C shows how rescaling of matrix A can lead to improved co-partitioning of genes and conditions.

Figure 2 (a) The outer product of the sorted eigenvectors u and v of the 2nd eigenvalue of the equal row- and column-sum bistochastic-like matrix B applied to dataset with three types of Lymphoma CLL(C), FL(F) and DLCL(D). Sorting of v orders the patients according to the different diseases. (b) as in (a) the 2nd singular value contribution to the biclustering method ($C^{-1}A^{\mathrm{T}}\ R^{-1}A$) of Lymphoma CLL(C), FL(F), DLCL(D) partitioned the patients according to their disease with one exception. We pre-selected all genes that had complete data along all experimental conditions (samples).

Figure 3 Lymphoma: Scatter plot of experimental conditions of the two best class partitioning eigenvectors v_i, v_j. The subscripts (i,j) of these eigenvectors indicate their corresponding singular values . CLL samples are denoted by red dots, DLCL by blue dots, and FL by green dots. (a) Bistochastization: the 2nd and 3rd eigenvectors of BB^T (b) Biclustering: the 2nd and 3rd eigenvectors of $R^{-1}AC^{-1}A^T$ (c) SVD: the 2nd and 3rd eigenvectors of AA^T (d) normalization and SVD: the 1st and 2nd eigenvectors of $\bar{A}\bar{A}^T$ where $\bar{A}$ is obtained by first dividing each column of A by its mean and then standardizing each row of the column normalized matrix. (e) Normalized cut algorithm: 2nd and 3rd eigenvectors of the row-stochastic matrix P. P is obtained by first creating a distance matrix S using Euclidean distance between the standardized columns of A, transforming it to an affinity matrix with zero diagonal elements and off diagonal elements defined as $W_{ij} = \exp(-\alpha S_{ij}) / \max(S_{ij})$ and finally normalizing each row

sum of the affinity matrix to one. (f) as in (c) but a with SVD analysis of the log interaction matrix K instead of A.

Figure 4 Scatter plots as in Fig. 3 with another Lymphoma dataset generated using Affymetrix chips [9] instead of microarrays. DLCL samples are denoted by green dots, CLL by blue dots, FL by yellow dots and DLCL cell lines by magenta dots.

Figure 5 Leukemia data is presented in the same format as in Fig. 3. B cell ALL samples are denoted by red dots, T cell ALL by blue dots, and AML by green dots. In this analysis we pre-selected all genes that had positive Affymetrix average difference expression levels.

Figure 6 Breast cell lines transfected with the CSF1R oncogene: Scatter plots as in Fig. 3 for mRNA ratios of benign breast cells and wild type cells transfected with the CSF1R oncogene causing them to invade and metastasize (A,a), ratios of cells transfected with a mutated oncogene causing an invasive phenotype and cells transfected with the wild type oncogene (C,c) and ratios of cells transfected with a mutated oncogene causing a metastatic phenotype and cells transfected with the wild type oncogene (D,d). In this case we pre-selected differentially expressed genes such that for at least one pair of samples the genes had a two fold ratio.

Figure 7 central nervous system embryonal tumor data generated using Affymetrix chips[10] of medulloblastoma (blue), malignant glioma (pink), normal cerebella (cyan), rhabdoid (green) and primitive neuro-ectodermal (red) tumors. Scatter plots of experimental conditions projected onto the three best class partitioning eigenvectors using the same format as in Fig. 3.

Figure 8 Optimal array partitioning obtained by the 1st singular vectors of the log-interaction matrix. The data consists of eight measurements of mRNA ratios for three pair of cell types: (A,a) benign breast cells and the wild-type cells transfected with the CSF1R oncogene causing them to invade and metastatize; (C,c) cells transfected with a mutated oncogene causing an invasive phenotype and cells transfected with the wild type oncogene; and (D,d) cells transfected with a mutated oncogene causing a metastatic phenotype and cells transfected with the wild type oncogene. In this case we pre-selected differentially expressed genes such that for at least one pair of samples the genes had a three fold ratio. The sorted eigen-gene v_1 and eigen-array u_1 have gaps indicating partitioning of patients and genes respectively. As a result, the outer product matrix sort(u_1) sort(v_1)^T has a "soft" block structure. The block structure is hardly seen when the raw data is sorted but not normalized. However it is more noticeable when the data is both sorted and normalized. Also, shown is the conditions projected onto the first two partitioning eigenvectors u_1 and u_2. Obviously, using the extra dimension gives a clearer separation.

Figure 1

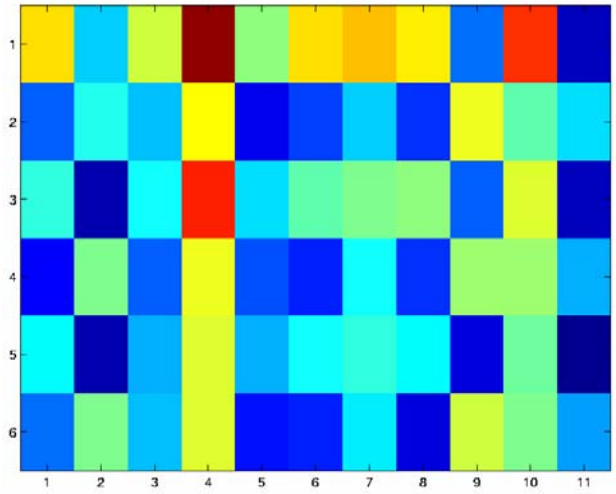
This figure gives an overview of important parts of the biclustering process. Part A shows the problem: shuffling a gene expression matrix to reveal a checkerboard pattern associating genes with conditions. Part B shows how this problem can be approached through solving an "eigenproblem." If a gene expression matrix *A* has a checkerboard structure, applying it to a step-like condition classification vector *x* will result in a step-like gene classification vector *y*. Moreover, if one then applies A^T to y, one will regenerate a step-like condition classification vector with the same partitioning structure as *x*. This suggests one can determine if *A* has a checkerboard structure through solving an eigenvalue problem. In other words, if *A* has a (hidden) checkerboard structure there exist some piecewise constant partition vectors $x = v_*$ and $y = u_*$ such that $A^T A v_* = \lambda^2 v_*$ and $AA^T u_* = \lambda^2 u_*$ (bottom quadrant of part B). To reveal whether the data has checkerboard structure one can inspect if some of the pairs of monotonically sorted gene and tumor eigenvectors v_i and u_i have an approximate stepwise (piecewise) constant structure. The outer product $u_* v_*^T$ of the sorted partitioning eigenvectors gives a checkerboard structure. Part C shows how rescaling of matrix *A* can lead to improved co-partitioning of genes and conditions.

(On Next Page...)

(A) The Problem: Identifying Marker Genes Associated with Certain Conditions

Matrix of raw data

Genes



Conditions

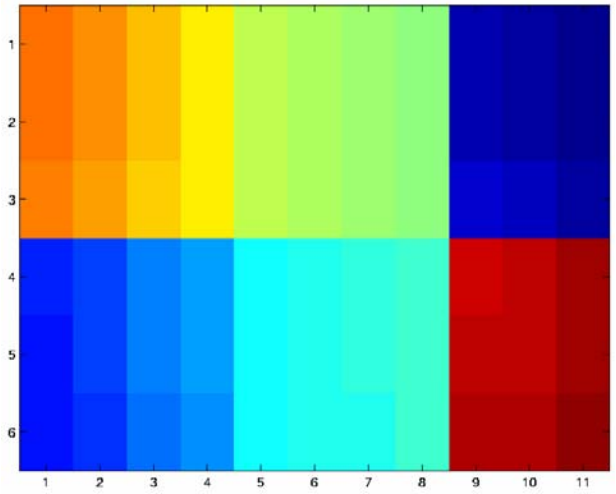
?

Shuffled Matrix

(containing checkerboard "biclusters" of conditions with marker genes)

Reordered Genes

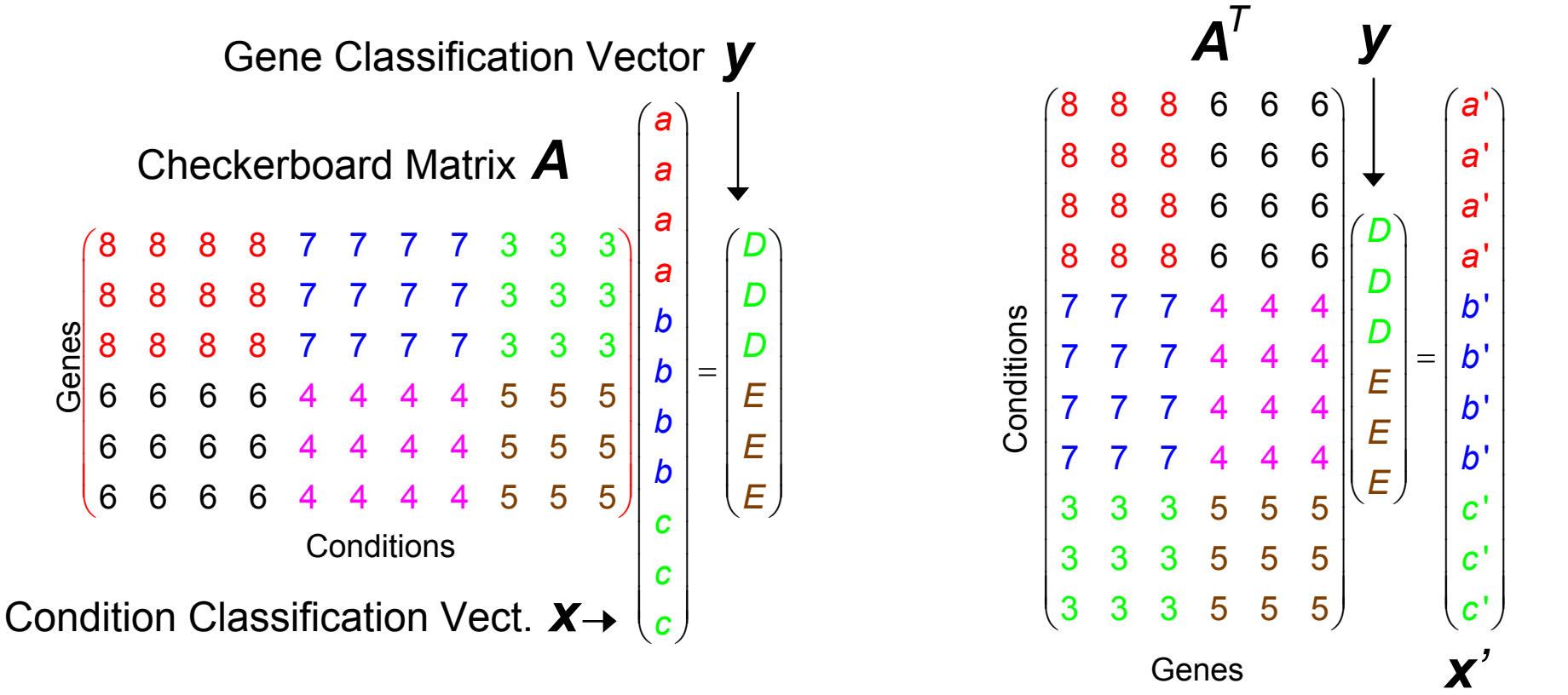
(Sorted according to a classification vector)



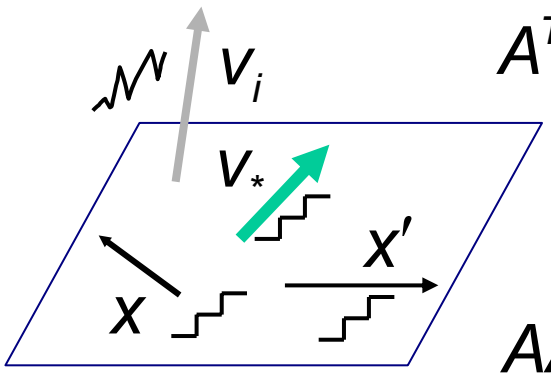
Reordered Conditions

(Sorted according to a classification vector)

(B) Identifying checkerboard matrices by their action on classification vectors: Formulation as "eigenproblem"



$$A^T A x = x' \qquad A^T A v = \lambda^2 v$$



$$AA^T y = y' \qquad AA^T u = \lambda^2 u$$

$$v_* \begin{pmatrix} a \\ a \\ a \\ a \\ b \\ b \\ b \\ b \\ c \\ c \\ c \end{pmatrix} u_*^T \begin{pmatrix} D & D & D & E & E & E \end{pmatrix} = \begin{pmatrix} aD & aD & aD & aE & aE & aE \\ aD & aD & aD & aE & aE & aE \\ aD & aD & aD & aE & aE & aE \\ aD & aD & aD & aE & aE & aE \\ bD & bD & bD & bE & bE & bE \\ bD & bD & bD & bE & bE & bE \\ bD & bD & bD & bE & bE & bE \\ bD & bD & bD & bE & bE & bE \\ cD & cD & cD & cE & cE & cE \\ cD & cD & cD & cE & cE & cE \\ cD & cD & cD & cE & cE & cE \end{pmatrix}$$

(C) A First Step of Matrix Normalization: Rescaling Rows to Same Mean

$$\begin{pmatrix} 8 & 8 & 8 & 8 & 7 & 7 & 7 & 7 & 3 & 3 & 3 \\ 16 & 16 & 16 & 16 & 14 & 14 & 14 & 14 & 6 & 6 & 6 \\ 8 & 8 & 8 & 8 & 7 & 7 & 7 & 7 & 3 & 3 & 3 \\ 12 & 12 & 12 & 12 & 8 & 8 & 8 & 8 & 10 & 10 & 10 \\ 6 & 6 & 6 & 6 & 4 & 4 & 4 & 4 & 5 & 5 & 5 \\ 6 & 6 & 6 & 6 & 4 & 4 & 4 & 4 & 5 & 5 & 5 \end{pmatrix} \begin{pmatrix} a \\ a \\ a \\ a \\ b \\ b \\ b \\ b \\ c \\ c \\ c \end{pmatrix} = \begin{pmatrix} D \\ 2D \\ D \\ 2E \\ E \\ E \end{pmatrix}$$

$$A_{raw}\, x = y$$

step-like zigzag

$$\begin{pmatrix} .12 & .12 & .12 & .12 & .10 & .10 & .10 & .10 & .04 & .04 & .04 \\ .12 & .12 & .12 & .12 & .10 & .10 & .10 & .10 & .04 & .04 & .04 \\ .12 & .12 & .12 & .12 & .10 & .10 & .10 & .10 & .04 & .04 & .04 \\ .11 & .11 & .11 & .11 & .07 & .07 & .07 & .07 & .09 & .09 & .09 \\ .11 & .11 & .11 & .11 & .07 & .07 & .07 & .07 & .09 & .09 & .09 \\ .11 & .11 & .11 & .11 & .07 & .07 & .07 & .07 & .09 & .09 & .09 \end{pmatrix} \begin{pmatrix} a \\ a \\ a \\ a \\ b \\ b \\ b \\ b \\ c \\ c \\ c \end{pmatrix} = \begin{pmatrix} D \\ D \\ D \\ E \\ E \\ E \end{pmatrix}$$

$$R^{-1} A_{raw}\, x = y$$

step-like step-like

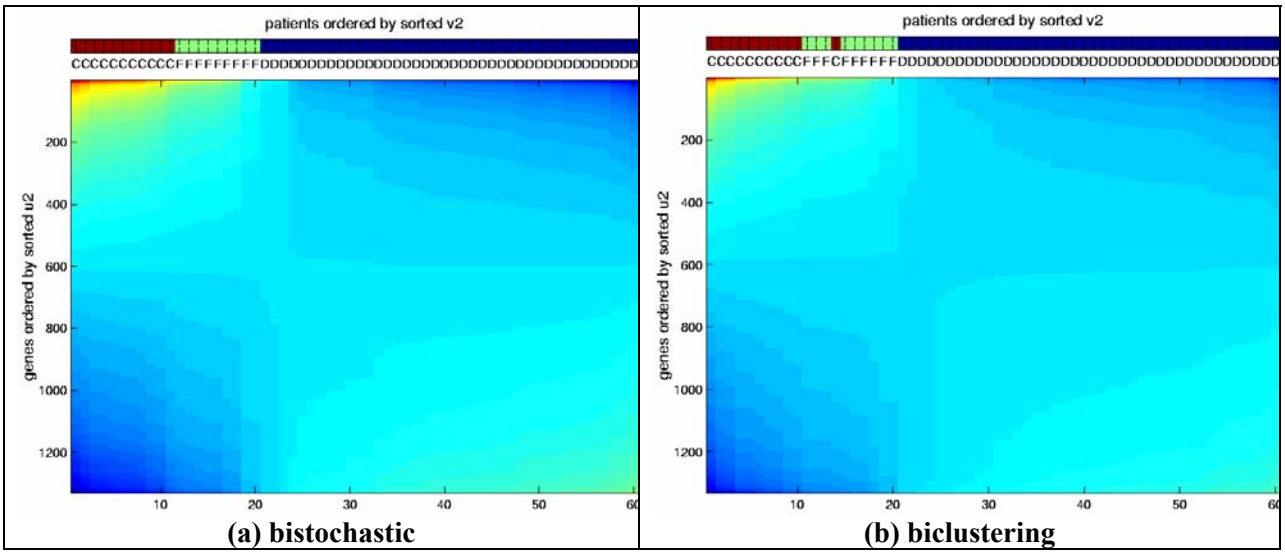


Figure 2 (a) The outer product of the sorted eigenvectors u and v of the 2nd eigenvalue of the equal row- and column-sum bistochastic-like matrix B applied to dataset with three types of Lymphoma CLL(C), FL(F) and DLCL(D). Sorting of v orders the patients according to the different diseases. (b) as in (a) the 2nd singular value contribution to the biclustering method ($C^{-1}A^{T}\ R^{-1}A$) of Lymphoma CLL(C), FL(F), DLCL(D) partitioned the patients according to their disease with one exception. We pre-selected all genes that had complete data along all experimental conditions (samples).

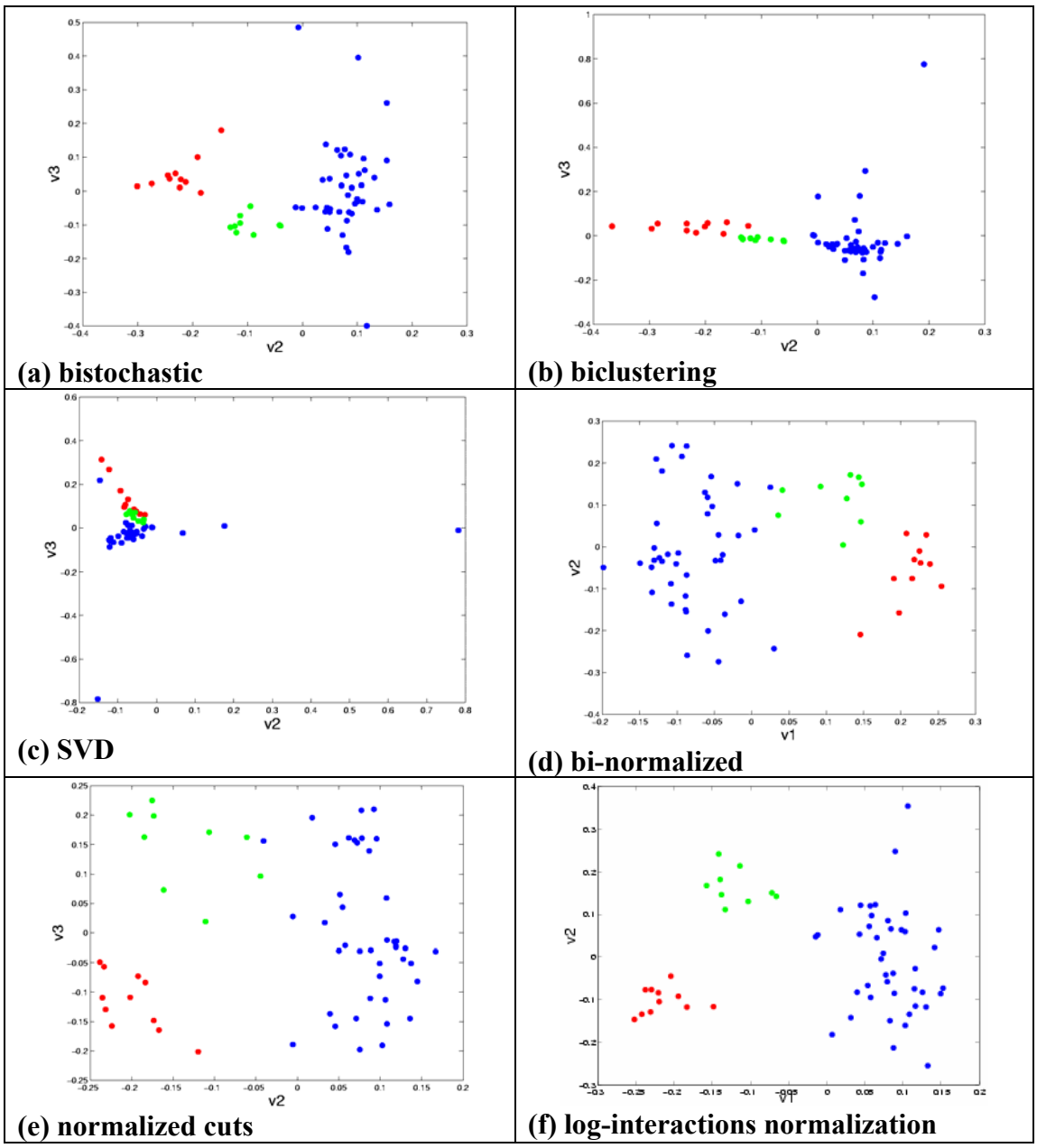


Figure 3 Lymphoma: Scatter plot of experimental conditions of the two best class partitioning eigenvectors v_i,v_j. The subscripts (i,j) of these eigenvectors indicate their corresponding singular values . CLL samples are denoted by red dots, DLCL by blue dots, and FL by green dots. (a) Bistochastization: the 2nd and 3rd eigenvectors of BB^T (b) Biclustering: the 2nd and 3rd eigenvectors of $R^{-1}AC^{-1}A^T$ (c) SVD: the 2nd and 3rd eigenvectors of AA^T (d) normalization and SVD: the 1st and 2nd eigenvectors of $\overline{A}\overline{A}^T$ where $\overline{A}$ is obtained by first dividing each column of A by its mean and then standardizing each row of the column normalized matrix. (e) Normalized cut algorithm: 2nd and 3rd eigenvectors of the row-stochastic matrix P. P is obtained by first creating a distance matrix S using Euclidean distance between the standardized columns of A, transforming it to an affinity matrix with zero diagonal elements and off diagonal elements defined as $W_{ij} = \exp(-\alpha S_{ij}) / \max(S_{ij})$ and finally normalizing each row sum of the affinity matrix to one. (f) as in (c) but a with SVD analysis of the log interaction matrix K instead of A.

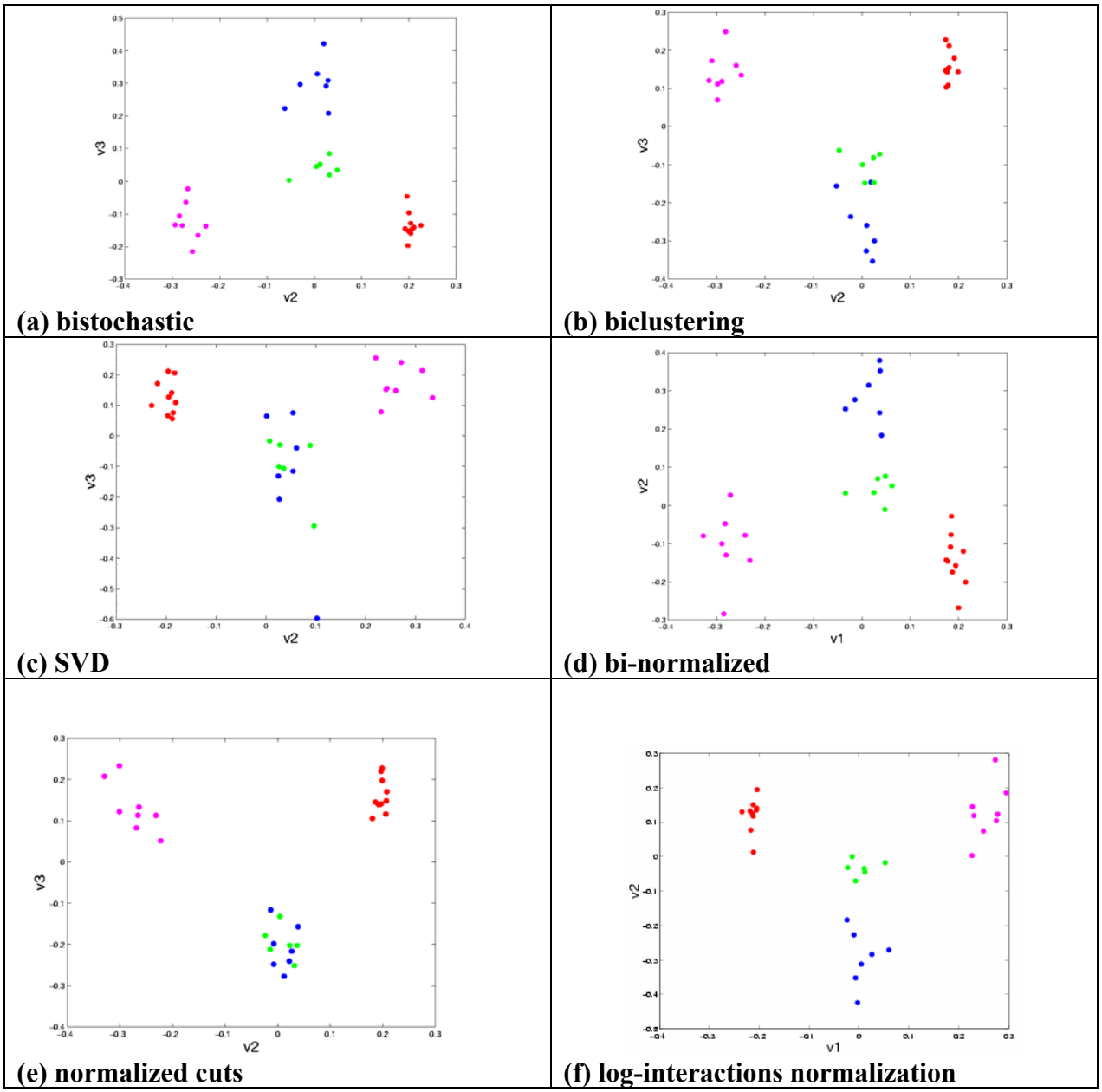


Figure 4 Scatter plots as in Fig. 3 with another Lymphoma dataset generated using Affymetrix chips [9] instead of microarrays. DLCL samples are denoted by green dots, CLL by blue dots, FL by yellow dots and DLCL cell lines by magenta dots.

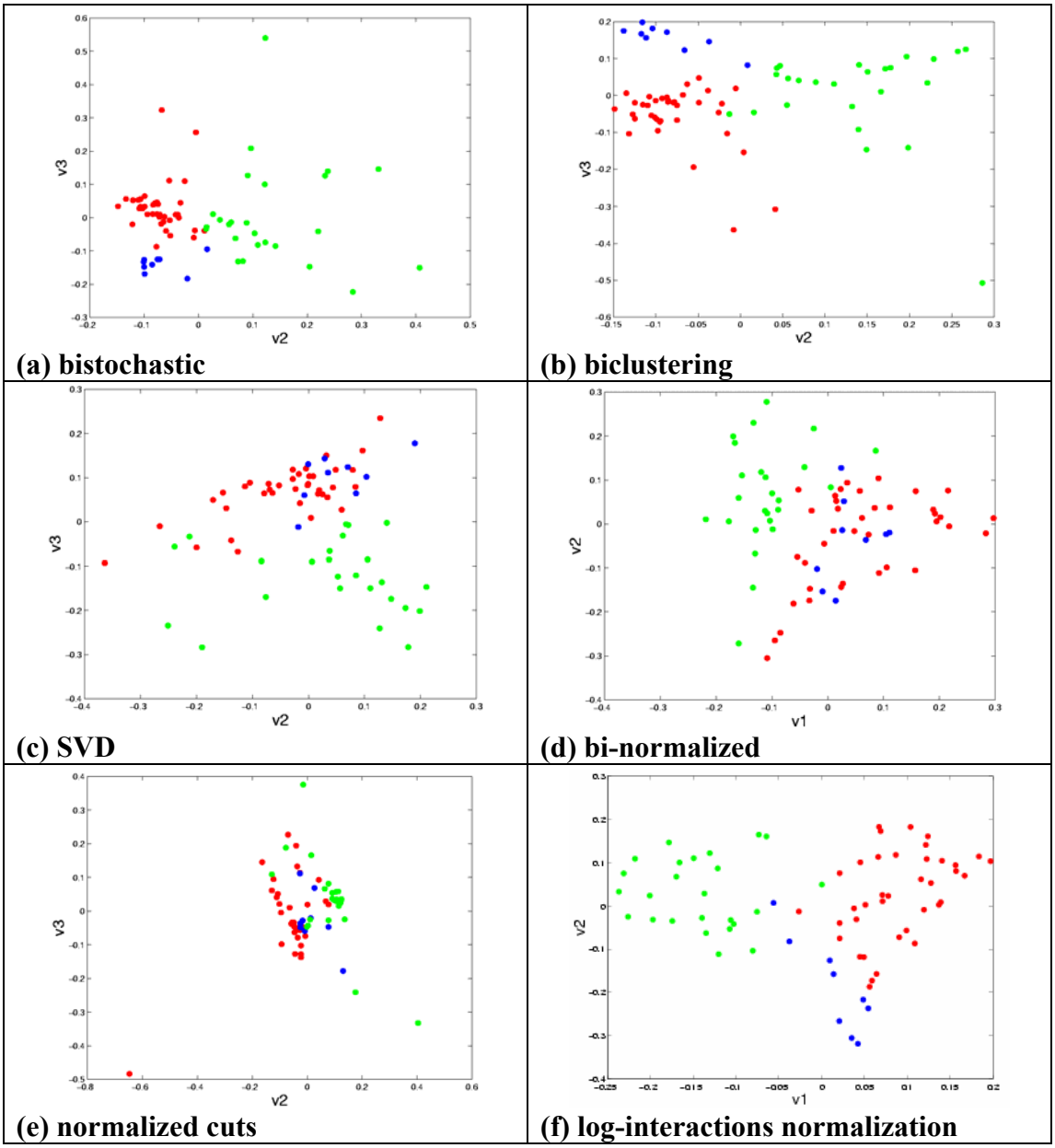


Figure 5 Leukemia data is presented in the same format as in Fig. 3. B cell ALL samples are denoted by red dots, T cell ALL by blue dots, and AML by green dots. In this analysis we pre-selected all genes that had positive Affymetrix average difference expression levels.

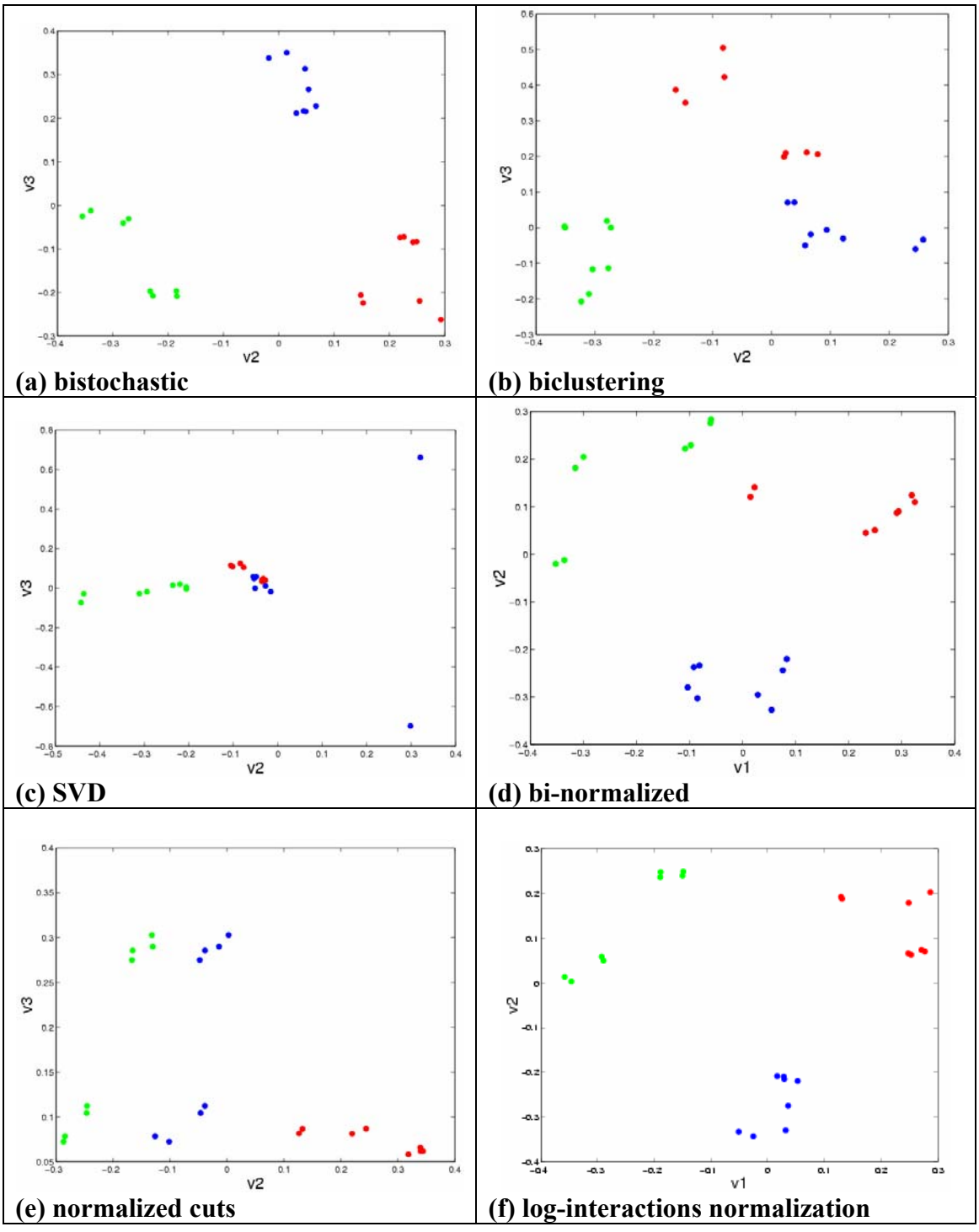


Figure 6 Breast cell lines transfected with the CSF1R oncogene: Scatter plots as in Fig. 3 for mRNA ratios of benign breast cells and wild type cells transfected with the CSF1R oncogene causing them to invade and metastasize (A,a), ratios of cells transfected with a mutated oncogene causing an invasive phenotype and cells transfected with the wild type oncogene (C,c) and ratios of cells transfected with a mutated oncogene causing a metastatic phenotype and cells transfected with the wild type oncogene (D,d).

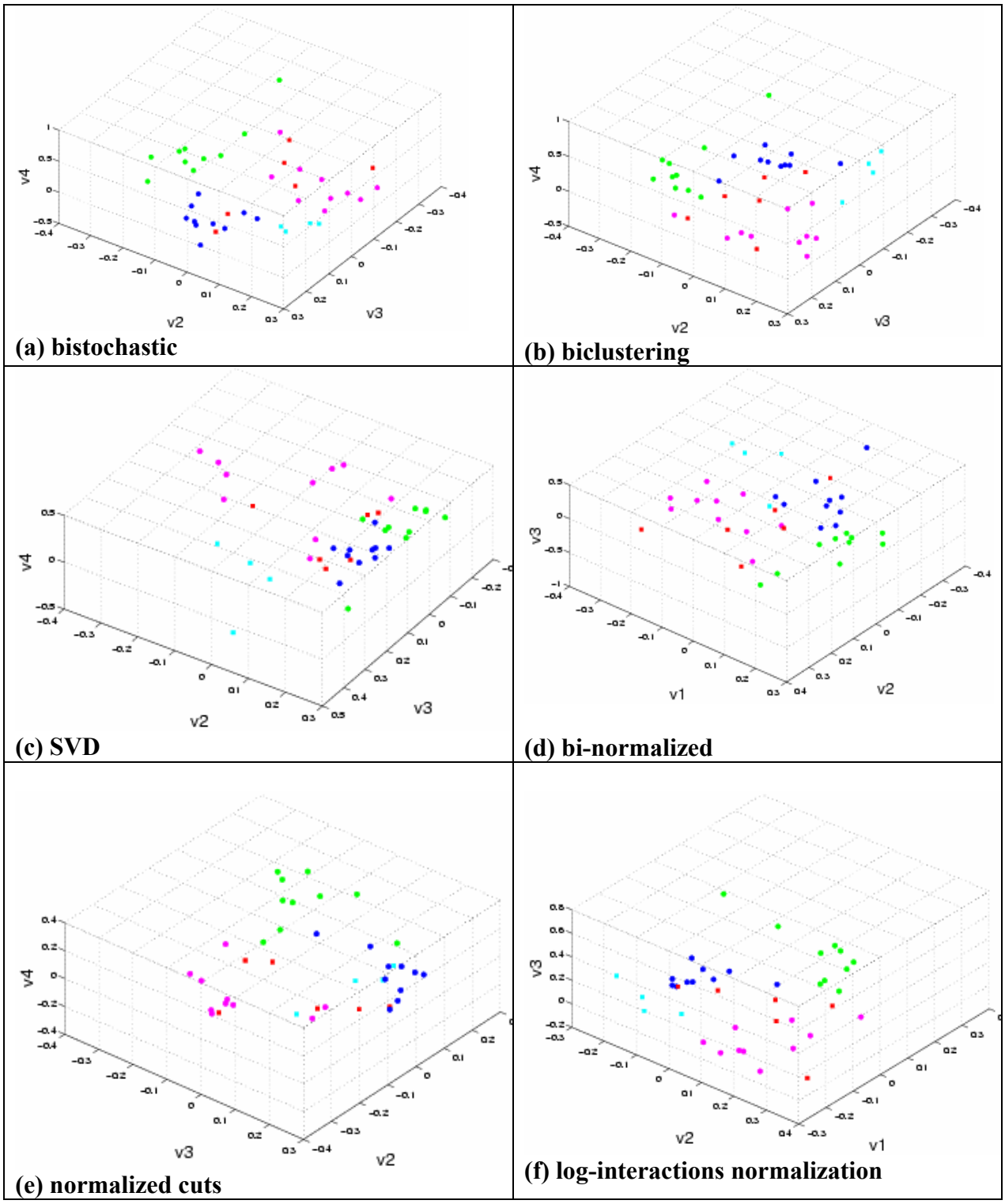
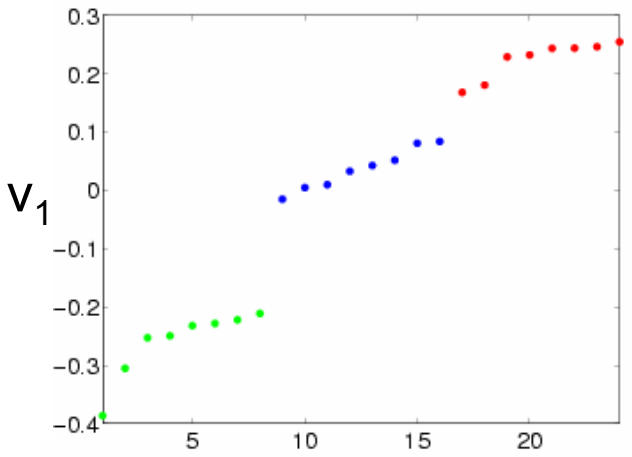


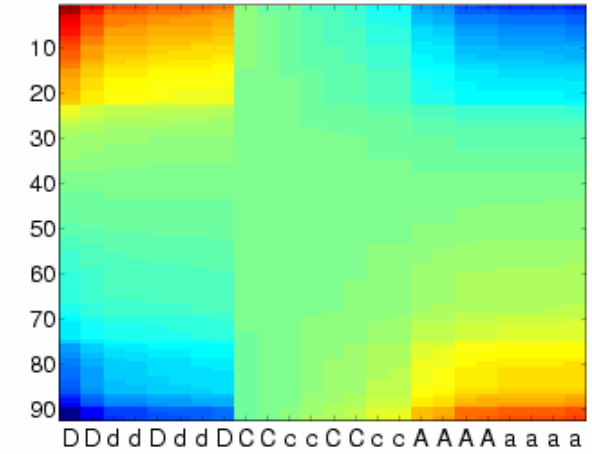
Figure 7 central nervous system embryonal tumor data generated using Affymetrix chips[10] of medulloblastoma (blue), malignant glioma (pink), normal cerebella (cyan), rhabdoid (green) and primitive neuro-ectodermal (red) tumors. Scatter plots of experimental conditions projected onto the three best class partitioning eigenvectors using the same format as in Fig. 3.

Figure 8 Optimal array partitioning obtained by the 1st singular vectors of the log-interaction matrix. The data consists of eight measurements of mRNA ratios for three pair of cell types: (A,a) benign breast cells and the wild-type cells transfected with the CSF1R oncogene causing them to invade and metastatize; (C,c) cells transfected with a mutated oncogene causing an invasive phenotype and cells transfected with the wild type oncogene; and (D,d) cells transfected with a mutated oncogene causing a metastatic phenotype and cells transfected with the wild type oncogene. In this case we pre-selected differentially expressed genes such that for at least one pair of samples the genes had a three fold ratio. The sorted eigen-gene v_1 and eigen-array u_1 have gaps indicating partitioning of patients and genes respectively. As a result, the outer product matrix sort(u_1) sort(v_1)^T has a "soft" block structure. The block structure is hardly seen when the raw data is sorted but not normalized. However it is more noticeable when the data is both sorted and normalized. Also, shown is the conditions projected onto the first two partitioning eigenvectors u_1 and u_2. Obviously, using the extra dimension gives a clearer separation.

(On next page...)

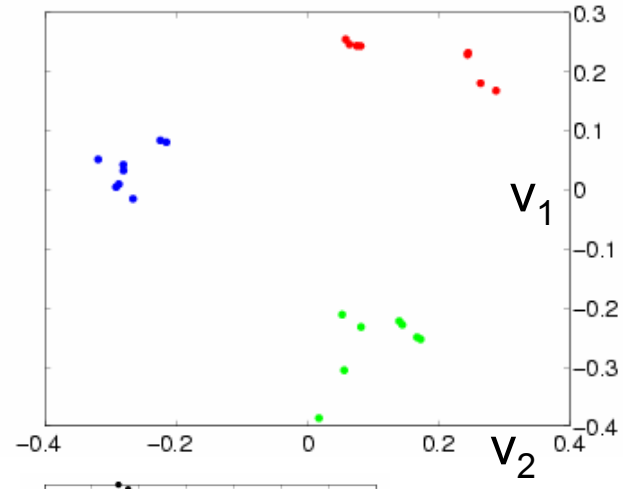
samples projected onto u_1

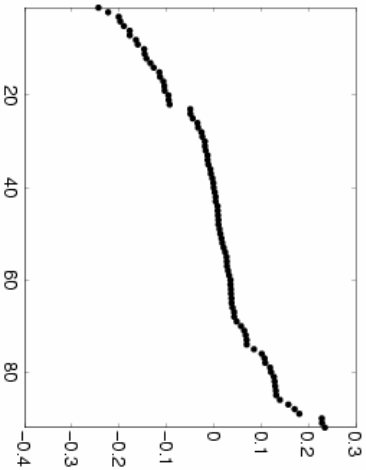




$u_1 v^T_1$

samples projected onto $u_{1,2}$

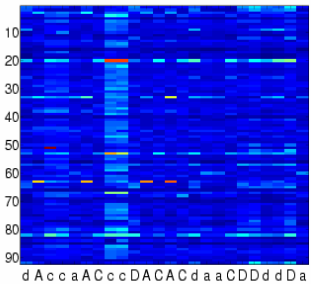




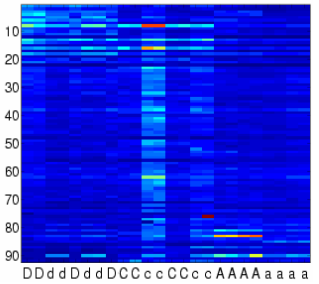
Genes projected onto v_1

u_1

Raw data



Sorted raw data



Sorted & normalized