Finding Large Average Submatrices in High Dimensional Data

Shabalin, A.[1], Weigman V.J.[2], Perou C.M.[3,4,5], Nobel A.B.[1,3]

September 17, 2008

[1] Department of Statistics and Operations Research, University of North Carolina at Chapel Hill

[2] Department of Biology, University of North Carolina at Chapel Hill

[3] Lineberger Comprehensive Cancer Center, University of North Carolina at Chapel Hill

[4] Department of Genetics, University of North Carolina at Chapel Hill

[5] Department of Pathology, University of North Carolina at Chapel Hill

Abstract

The search for sample-variable associations is an important problem in the exploratory analysis of high dimensional data. Biclustering methods search for sample-variable associations in the form of distinguished submatrices of the data matrix. (The rows and columns of a submatrix need not be contiguous.) In this paper we propose and evaluate a statistically motivated biclustering procedure (LAS) that finds large average submatrices within a given real-valued data matrix. The procedure operates in an iterative-residual fashion, and is driven by a Bonferroni-based significance score that effectively trades off between submatrix size and average value. We examine the performance and potential utility of LAS, and compare it with a number of existing methods, through an extensive three-part validation study using two gene expression datasets. The validation study examines quantitative properties of biclusters, biological and clinical assessments using auxiliary information, and classification of disease subtypes using bicluster membership. In addition, we carry out a simulation study to assess the effectiveness and noise sensitivity of the LAS search procedure. In closing, we propose that LAS is a simple and effective exploratory tool for discovery of biologically relevant structures in high dimensional data.

Software and Supplementary Materials are available at: https://genome.unc.edu/las/.

1 Introduction

Unsupervised exploratory analysis plays an important role in the study of large, high-dimensional datasets that arise in a variety of applications, including gene expression microarrays. Broadly speaking, the goal of such analysis is to find patterns or regularities in the data, without *ab initio* reference to external information about the available samples and variables. One important source of regularity in experimental data are associations between sets of samples and sets of variables. These associations correspond to distinguished submatrices of the data matrix, and are generally referred to as biclusters, or subspace clusters. In microarray analysis, biclusters, in conjunction with auxiliary clinical and biological information, can provide a first step in the process of identifying disease subtypes and gene regulatory networks.

In this paper we propose and evaluate a statistically motivated biclustering procedure that finds large average submatrices within a given real-valued data matrix. The procedure, which is called LAS (Large Average Submatrix), operates in an iterative fashion, and is based on a simple significance score that trades off between the size of a submatrix and its average value. A connection is established between maximization of the significance score and the minimum description length principle.

Beyond examining LAS's performance we compare it with a number of existing methods, through an extensive validation study using two independent gene expression datasets. The validation study has three parts. The first concerns quantitative properties of the biclustering methods such as bicluster size, overlap and coordinate-wise statistics. The second is focused biological and clinical assessments using auxiliary information about the samples and genes under study. In the the third part of the study, the biclusters are used to perform classification of disease subtypes based in their sample membership. In addition, we carry out a simulation study to assess the effectiveness and noise sensitivity of the LAS search procedure.

1.1 Biclustering

Sample-variable associations can be defined in a variety of ways, and can take a variety of forms. The simplest, and most common, way of identifying associations in gene expression data is to independently cluster the rows and columns of the data matrix using a multivariate clustering procedure [Weinstein et al. (1997); Eisen et al. (1998); Tamayo et al. (1999); Hastie et al. (2000)]. When the rows and columns of the data matrix are re-ordered so that each cluster forms a contiguous group, the result is a partition of the data matrix into non-overlapping rectangular cells. The search for sample variable associations then consists of identifying cells whose entries are, on average, bright red (large and positive) or bright green (large and negative) [Weigelt et al. (2005a)]. In some cases, one can improve the results of independent row-column clustering by simultaneously clustering samples and variables, a procedure known as co-clustering [Kluger et al. (2003); Dhillon (2001); Getz et al. (2000)].

Independent row-column clustering (IRCC) has become a standard tool for the visualization and exploratory analysis of microarray data, but it is an indirect approach to the problem of finding sample-variable associations. By contrast, biclustering methods search directly for sample-variable associations, or more precisely, for submatrices U of the data matrix X whose entries meet a predefined criterion. Submatrices meeting the criterion are typically referred to as biclusters. It is important to note that the rows and columns of a bicluster (and more generally a submatrix) need not be contiguous. A number of criteria for defining biclusters U have been considered in the literature, for example: the rows of U are approximately equal to each other [in Aggarwal et al. (1999)]; the columns of U are approximately equal [in Friedman and Meulman (2004)]; the elements of U are well-fit by a 2-way ANOVA model [Cheng and Church (2000); Lazzeroni and Owen (2002); Wang et al. (2002)]; the rows of U have equal [Ben-Dor et al. (2003)] or approximately equal [Liu et al. (2004)] rank statistics; all elements of U are above a given threshold [Prelic et al. (2006)].

The focus of this paper is the simple criterion that the average of the entries of the submatrix U is large and positive, or large and negative. Submatrices of this sort will appear red or green in the standard heat map representation of the data matrix, and are similar to those targeted by independent row-column clustering.

1.2 Features of Biclustering

While its direct focus on finding sample-variable associations makes biclustering an attractive alternative to row-column clustering, biclustering has a number of other features that we briefly discuss below.

Row-column clustering assigns each sample, and each variable, to a unique cluster. By contrast, the submatrices produced by biclustering methods may overlap, and need not cover the entire data matrix which better reflects the complexity of many scientific problems. For example, the same gene can play a role in multiple pathways, and a single sample may belong to multiple phenotypic or genotypic subtypes. Multiple bicluster membership for rows and columns can directly capture this aspect of experimental data.

In row-column clustering, the group to which a sample is assigned depends on all the available variables, and the group to which a variable is assigned depends on the all the available samples. By contrast, biclusters are locally defined: the inclusion of samples and variables in a bicluster depends only on their expression values inside the associated submatrix. In particular, locality allows biclusters to be more robust to the values of irrelevant genes than row-column clustering. This feature of biclustering gives it greater exploratory power and flexibility than row-column clustering, though there is a computational price to pay for these benefits. For more on the potential advantages of biclustering, see Madeira and Oliveira (2004); Jiang et al. (2004); Parsons et al. (2004).

Figure 1 illustrates the differences between the blocks arising from independent row-column clustering and those arising from biclustering. Note that while one may display an individual bicluster as a contiguous block of variables and samples by suitably reordering the rows and columns of the data matrix, when considering more than two biclusters, it is not always possible to display them simultaneously as contiguous blocks.

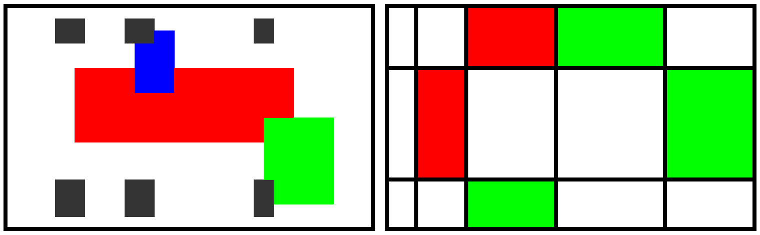

Figure 1: Illustration of bicluster overlap (left) and row-column clustering (right).

The flexibility and exploratory power of biclustering methods comes at the cost of increased computational complexity. Most biclustering problems are NP complete, and even the most efficient exact algorithms (those that search for every maximal submatrix satisfying a given criterion) can be prohibitively slow, and produce a large number of biclusters, when they are applied to large datasets. The LAS algorithm relies on a heuristic (non-exact), randomized search to find biclusters; as do many of the other methods we tested.

2 LAS

In this paper we present and assess a significance-based approach to biclustering of real-valued data. Using a simple Gaussian null model for the observed data, we assign a significance score to each submatrix U using a Bonferroni-corrected p-value that is based on the size and average value of the entries of U. The Bonferroni correction accounts for multiple comparisons that arise when searching among many submatrices for one having a large average value. In addition, the correction acts as a penalty that controls the size of discovered submatrices. (Connections between LAS and the Minimum Description Length principle are discussed in Section 2.3 below.)

2.1 Basic Model and Score Function

Let $X = \{x_{i,j} : i \in [m], j \in [n]\}$ be the observed data matrix. (Here and in what follows, $[k]$ denotes the set of integers from 1 to k.) A submatrix of X is an indexed set of entries $U = \{x_{i,j} : i \in A, j \in B\}$ associated with a specified set of rows $A \subseteq [m]$ and columns $B \subseteq [n]$. In general, the rows in A and the columns in B need not be contiguous.

The LAS algorithm is motivated by an additive block model under which X is expressed as the sum of K constant, and potentially overlapping, submatrices plus noise. More precisely, the model states that

$$x_{i,j} \;=\; \sum_{k=1}^{K} \alpha_k \, I(i \in A_k, j \in B_k) \;+\; \varepsilon_{ij}, \qquad i \in [m],\; j \in [n] \tag{1}$$

where $A_k \subseteq [m]$, $B_k \subseteq [n]$, $\alpha_k \in \mathbb{R}$, $\{\varepsilon_{ij}\}$ are independent $N(0,1)$ random variables, and $I(\cdot)$ is an indicator function equal to one when the condition in parentheses holds. When $K = 0$, the model (1) reduces to the simple null model

$$\{x_{i,j} : i \in [m], j \in [n]\} \;\text{ are i.i.d with } x_{i,j} \sim \mathcal{N}(0,1) \tag{2}$$

under which X is an $m \times n$ Gaussian random matrix.

The null model (2) leads naturally to a score function for submatrices. The score assigned to a $k \times l$ submatrix U of X with average $\text{Avg}(U) = \tau > 0$ is

$$S(U) \;=\; -\log\left[\binom{m}{k}\binom{n}{l}\Phi\left(-\tau\sqrt{kl}\right)\right]. \tag{3}$$

The term in square brackets is a Bonferroni corrected upper bound on the probability that there exists a $k \times l$ submatrix with average greater than or equal to τ in an $m \times n$ Gaussian random matrix. (This probability can be thought of as a p-value associated with the null model (2) and the test function $\mathrm{Avg}(U)$.) The term $\binom{m}{k}\binom{n}{l}$ is the number of $k \times l$ submatrices of an $m \times n$ matrix, and $\Phi(-\tau\sqrt{kl})$ is the probability that the average of kl independent standard normals exceeds $\tau > 0$.

The score function $S(\cdot)$ measures departures from the null (2) on the basis of submatrix dimensions and average values. It provides a simple, one-dimensional yardstick with which one can compare and rank submatrices of different sizes and intensities. Among submatrices of the same dimensions, it favors those with higher average.

2.2 Description of Algorithm

The LAS score function is based on the normal CDF, and is sensitive to departures from normality that arise from heavy tails in the empirical distribution of the expression values. Outliers can give rise to submatrices that, while highly significant, have very few samples or variables. To address this issue, we considered the standard Q-Q plot of the empirical distribution of the data matrix entries against the standard normal CDF. Both the breast cancer and lung cancer datasets considered in Section 4 exhibited heavy tails. In such cases our implementation of the LAS algorithm suggests transformation $f(x) = \mathrm{sign}(x)\log(1 + |x|)$ to each entry of the data matrix. After transformation the Q-Q plot indicated excellent agreement with the normal distribution.

The LAS algorithm initially searches for the submatrix of X maximizing the significance score $S(\cdot)$. Once a candidate submatrix is found, it is removed from X (via mean subtraction) and the search procedure is repeated on the residual matrix. The core of the algorithm is a randomly initialized, iterative search procedure for finding a maximally significant submatrix of a given matrix. The pseudo code for the algorithm is as follows:

Input: Data matrix X

Search: Find a submatrix U^* of X that approximately maximizes the score function $S(\cdot)$.

Residual: Subtract the average of U^* from each of its elements in X.

Repeat: Return to Search.

Stop: When $S(U^*)$ falls below a threshold, or a user-defined number of submatrices are produced.

The output of the algorithm is a collection of submatrices having significant positive averages. Repeating the algorithm with X replaced by $-X$ yields submatrices with significant negative averages.

It is not feasible in the search procedure to check the score of each of the 2^{n+m} possible submatrices of X. Instead, the procedure iteratively updates the row and column sets of a candidate submatrix in a greedy fashion until a local maximum of the score function is achieved. For fixed k, l, the basic search procedure operates as follows.

```
Initialize: Select $l$ columns of $B$ at random.
Loop: Iterate until convergence of $A$, $B$
    Let $A := k$ rows with the largest sum over the columns in $B$.
    Let $B := l$ columns with the largest sum over the rows in $A$.
Output: Submatrix associated with final $A$, $B$.
```

As currently implemented, the initial values of k and l are selected at random from sets $\{1, \ldots, \lceil m/2 \rceil\}$ and $\{1, \ldots, \lceil n/2 \rceil\}$ respectively, and are held fixed until the algorithm finds a local maximum of the score function. On subsequent iterations, the algorithm adaptively selects the number of rows and columns in order to maximize the significance score. Each run of the basic search procedure yields a submatrix that is a local maximum of the score function (a submatrix that cannot be improved by changing only its column set or its row set). The basic search procedure is repeated 1000 times, and the most significant submatrix found is returned in the main loop of the algorithm.

The only operational parameters of the LAS algorithm are the number of times the basic search procedure is run in each main loop of the algorithm, and the stopping criterion. This is an important feature of LAS, one that makes the method well suited for ready application to scientific problems. Many biclustering methods involve several parameters that may require tuning for optimal performance, and whose minor alteration can result in substantial changes in the method's outputs. On the contrary LAS has the minimum (basically zero) number of parameters. In our experiments on real data (see Section 5.2), we found that 1000 iterations of the main loop of the algorithm is enough for stable performance of the algorithm.

2.3 Penalization and MDL

The score function employed by LAS can be written as a sum of two terms. The first, $-\log \Phi(-\sqrt{kl}\tau)$, is positive and can be viewed as a "reward" for finding a $k \times l$ submatrix with average τ. The second, $-\log[\binom{m}{k}\binom{n}{l}]$, is negative and is a multiple comparisons penalty based on the number of $k \times l$ submatrices in X. The penalty depends separately on k and l, and it's combinatorial form suggests a connection with the Minimum Description Length Principle (MDL), following Rissanen;Grunwald (2004), and Barron and Yu (1998). The MDL principle is a formalization of Occam's Razor, in which the best model for a given set of data is the one that leads to the shortest overall description of the data.

In the Appendix we describe a code for submatrices based on natural family of block-additive models, and show that the description length of a submatrix is approximately equal to a linear function of its LAS score. The penalty term in the LAS score function corresponds to the length of the code required to describe the location of a $k \times l$ submatrix, while the "reward" is related to the reduction in code length achieved by describing the residual matrix instead of the original matrix. The connection with MDL provides support for the significance based approach to biclustering adopted here.

3 Description of Competing Methods

In this section we describe the methods that we will compare to the LAS algorithm in the validation sections below. We considered biclustering methods that search directly for sample variable associations, as well as biclusters derived from independent row-column clustering.

3.1 Biclustering Methods

Initially, we compared LAS with six existing biclustering methods: PLAID, CC, SAMBA, ISA, OPSM, and BiMax. These methods employ a variety of objective functions and search algorithms. We limited our comparisons to methods that (i) have publicly available implementations with straightforward user interfaces, (ii) can efficiently handle large datasets arising from gene expression and metabolomic data, and (iii) are well suited to use by biologists. The methods are described in more detail below.

The PLAID algorithm of Lazzeroni and Owen (2002) employs an iterative procedure to approximately express the data matrix X as a sum of submatrices whose entries follow a two-way ANOVA model. At each stage, PLAID searches for a submatrix maximizing explained variation, as measured by reduction in the overall sum of squares. We set the parameters of PLAID so that at each stage it fits a constant submatrix (with no row or column effects). With these settings, the PLAID method is most closely related to LAS, and also derives from a block-additive model like (1). The discussion in Section 2 discusses differences between the methods in the context of validation.

The CC biclustering algorithm of Cheng and Church (2000) searches for maximal-sized submatrices such that the sum of squared residuals from an two-way ANOVA fit falls below a given threshold. Whereas PLAID searches for a submatrix maximizing explained variation, CC searches for large submatrices with small unexplained variation. The LAS, PLAID and CC algorithms discover biclusters sequentially. Once a candidate target is identified, LAS and PLAID form the associated residual matrix before proceeding to the next stage. By contrast, CC replaces the values of the target submatrix by Gaussian noise.

The SAMBA algorithm of Tanay et al. (2002) adopts a graph theoretic approach, in which the data matrix is organized into a bipartite graph, with one set of nodes corresponding to genes, and the other corresponding to samples. Weights are then assigned to edges that connect genes and samples based on the data matrix, and the subgraphs with the largest overall weights are returned.

Ihmels et al. (2002) developed a biclustering algorithm (ISA) that searches for maximal submatrices whose row and column averages exceed preset thresholds. Both LAS and ISA rely on iterative search procedures that are variants of EM and Gibbs type algorithms. In both methods, the search procedure alternately updates the columnset (given the current rowset) and then the rowset (given the current columnset) until converging to a local optimum.

The OPSM algorithm of Ben-Dor et al. (2003) searches for maximal submatrices whose rows have the same order statistics. Like LAS, the OPSM algorithm makes use of a multiple comparison corrected p-value in assessing and comparing biclusters of different sizes.

Each of the algorithms above employs heuristic strategies to search for distinguished submatrices. By contrast, the Bimax algorithm introduced by Prelic *et al*, uses a divide-and-conquer approach to find *all* inclusion-maximal biclusters whose values are above a user-defined threshold. Bimax is the only exact algorithm among those considered here.

We ran all methods except Plaid and CC with their default parameter settings. LAS, CC and PLAID allow the user to choose the number of biclusters produced; we selected 60 biclusters for each method. The settings of Plaid were chosen so that the submatrix fit at each stage is a constant, without row and column effects. Once the CC method identifies a bicluster, it removes it from the data matrix by replacing its elements by noise.

When the CC method was run with the default parameter $\delta = 0.5$, it initially produced a single bicluster that contained most of the available genes and samples, leaving very little information from which additional biclusters could be identified. To solve this problem, we reduced the δ parameter in CC from 0.5 to 0.1. A description of the computer used to run the biclustering algorithms, and the set of parameters used for each method is provided in the Appendix.

3.2 Independent Row-Column Clustering (IRCC)

In addition to the methods described above, we also produced biclusters from k-means and hierarchical clustering. We applied k-means clustering independently to the rows and columns the data matrix, with values of k ranging from 3 to 15. In each case, we produced 30 clusterings and selected the one with the lowest sum of within-cluster sum of squares. The set of $85 \times 117 = 9,945$ submatrices (not all column clusters were unique) obtained from the Cartesian product of the row and column clusters is denoted by KM.

We applied hierarchical clustering independently to the rows and columns of the data matrix, using a Pearson correlation based distance and average linkage. All clusters associated with subtrees of the dendrogram were considered, but row clusters with less that 10 rows, and column clusters with less than 8 columns, were discarded. The resulting set of $34 \times 2806 = 95,404$ submatrices obtained from the Cartesian product of the row and column clusters is denoted by HC.

4 Comparison and Validation

We applied LAS and the biclustering methods described in the previous section to two existing gene expression datasets: a breast cancer study from Hu et al. (2006), and a lung cancer study from Bhattacharjee et al. (2001) The datasets can be downloaded from University of North Carolina Microarray Database (UMD, http://genome.unc.edu) and http://www.broad.mit.edu/mpr/lung/ respectively. In this section we describe and implement a number validation measures for assessing and comparing the performance of the biclustering methods under study. The validation results for the breast cancer study are detailed below; the results for the lung cancer data are contained in the Supplementary Materials. The validation measures are applicable to any biclustering method, and most gene expression datasets.

4.1 Description of Hu Data

The breast cancer dataset considered here is from a previously published breast cancer study, described in Hu et al., which was based on 146 Agilent 1Av2 microarrays. Initial filtering and normalization followed the protocol in Hu *et al.*: genes with intensity less than 30 in the red or green channel were removed; for the remaining genes, red and green channels were combined using the $\log_2$ ratio. The initial log-transformed dataset was row median centered, and missing values were imputed using a k-nearest neighbor algorithm with k = 10. Among the 146 samples, there were 29 pairs of biological replicates in which RNA was prepared from different sections of the same tumor. To avoid giving these samples more weight in the analysis, we removed the replicates keeping only the primary tumor profiles. After preprocessing, the dataset contained 117 samples and 13,666 genes. In what follows, the dataset will be referred to as **Hu**.

In the analysis below, the Hu dataset was the starting point for each biclustering method under study.

4.2 Quantitative Comparisons

LAS, Plaid and CC were set to produce 60 biclusters; the output of the other methods was determined by their default parameters, with values ranging from 15 (OPSM) to 1977 (BiMax). KM and HC produced 9,945 and 95,404 biclusters, respectively. Table 1 shows the number of biclusters produced by each method.

All biclustering methods were run on the same computer, having an AMD64 FX2 Dual Core processor with 4GB of RAM (a complete specification is provided in the Appendix). The running time of LAS was 2 hours; ISA and OPSM finished in about 30 minutes; CC, Plaid, and SAMBA finished in less then 10 minutes. The Bimax algorithm took approximately 5 days. Hierarchical clustering took 2 minutes, while k-means clustering (with $k = 3, ..., 15$ and 30 repeats) took 1 hour 40 minutes. All methods except BiMax ran in a reasonable amount of time. Our primary focus in validation was the quality of their output.

4.2.1 Bicluster Sizes

In Figure 2 we plot the row and column dimensions of the biclusters produced by the different methods. The resulting scatter plot shows marked differences between the methods under study, and provides useful insights into their utility and potential biological findings. (A similar figure could

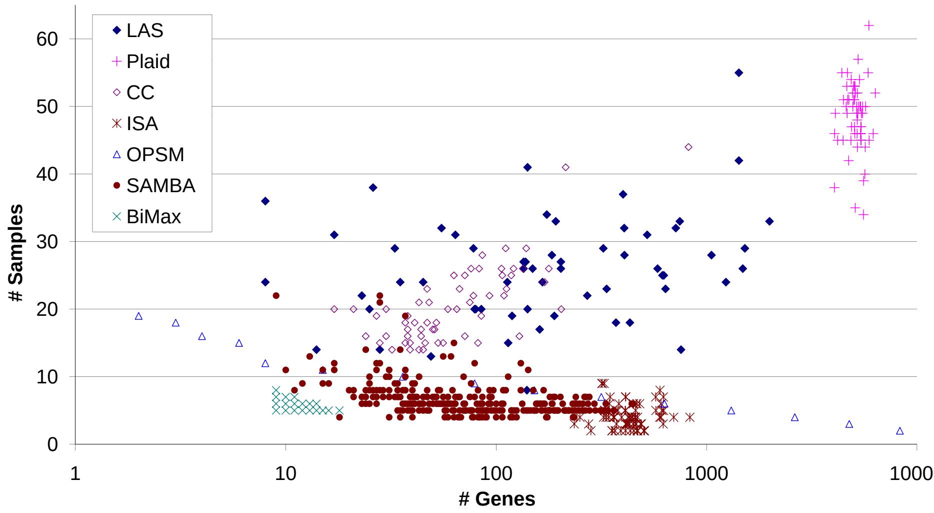

Figure 2: Bicluster sizes for different methods.

be used, e.g., to assess the effects of different parameter settings for a single method of interest.) Both LAS and CC produce a relatively wide range of bicluster sizes, with those of LAS ranging from 8 × 8 (genes × samples) to 1991 × 55. The other methods tested produced biclusters with a more limited range of sizes. Biclusters produced by ISA, OPSM, and SAMBA have a relatively small number of samples, less than 10 samples per bicluster on average in each case. (Some of the points denoting OPSM clusters have been obscured in the figure.) The biclusters produced by Bimax had at most 8 samples, and at most 18 genes. By contrast, Plaid produced large biclusters, having an average of 49 samples and 5130 genes per bicluster.

The differences between LAS and Plaid bear further discussion. We ran Plaid with settings (constant fit, no row and column effects) that made it most similar to LAS. With these settings, both methods rely on similar models, and proceed in stages via residuals, but differ in their objective functions. Plaid seeks to maximize the explained variation $kl\tau^2$, or equivalently $-\log\Phi(-\sqrt{kl}\tau)$. By contrast, the score function maximized by LAS includes a combinatorial penalty term involving k and l that acts to control the size of the discovered submatrices. In this, and other, experiments, the penalty excludes very large submatrices, and produces a relatively wide range of bicluster sizes. (While the combinatorial penalty is small for values of k close to m and l close to n, submatrices of this size tend to have a small average value.)

4.2.2 Effective Number of Biclusters

Distinct biclusters produced by the same method may exhibit overlap. On the one hand, the flexibility of overlapping gene and sample sets has the potential to better capture underlying biology. On the other hand, extreme overlap of biclusters can reduce a method's effective output: two moderate sized biclusters that differ only in a few rows or columns do not provide much more information than either bicluster alone. Whatever the source of overlap, it is helpful to keep it in mind when evaluating other features of a method, such as the number of biclusters it produces that are deemed to be statistically significant. To this end, we measure of the effective number of biclusters in a family $U_1, \ldots, U_K$ by

$$F(U_1, \ldots, U_K) \;=\; \sum_{k=1}^{K} \frac{1}{|U_k|} \sum_{x \in U_k} \frac{1}{N(x)},$$

where $N(x) = \sum_{k=1}^{K} I\{x \in U_k\}$ is the number of biclusters containing matrix entry x. The measure $F(\cdot)$ has the property that if, for any $1 \leq r \leq K$, the biclusters $U_1, \ldots, U_K$ can be divided into r non-overlapping groups of identical biclusters, then $F(U_1, \ldots, U_K) = r$.

Method	# of Clusters	Eff. # of Clusters	Ratio
LAS	60	48.6	0.810
Plaid	60	6.4	0.106
CC	60	60.0	1.000
ISA	72	42.3	0.588
OPSM	15	9.1	0.605
SAMBA	289	171.7	0.594
BiMax	1,977	42.9	0.022
KM	9,945	78.7	0.008
HC	95,404	800.4	0.008

Table 1: Output summary for different biclustering methods. From left to right: total number of biclusters produced; effective number of biclusters; the ratio of the effective number to the total number of biclusters.

Table 1 shows the effective number of biclusters produced by each method. The low overlap of the CC algorithm is due to the fact that it replaces the values in discovered submatrices by Gaussian noise, so that a matrix element is unlikely to belong to more than one bicluster. Bimax is an exhaustive method with no pre-filtering of its output; it produced a large number of small, highly overlapping biclusters. Biclusters produced by LAS had modest levels of overlap, less than those of all other methods, except CC. The high overlap of Plaid biclusters is explained in part by their large size.

4.2.3 Score-Based Comparison of LAS and Standard Clustering

Ideally, a direct search for large average submatrices should improve on the results of independent row-column clustering. To test this, we computed the significance score $S(C)$ for every cluster produced by KM and HC, and compared these to the scores obtained with LAS. The highest score achieved by a KM and HC biclusters were 6316 and 5228, respectively. The first LAS biclusters had scores 12883 (positive average) and 10070 (negative average); the scores of the first 6 LAS biclusters were higher than scores of all the biclusters produced by KM or HC. The highest score achieved by a Plaid bicluster was 12542, which also dominated the scores achieved by KM and HC. These results show that LAS is capable, *in practice*, of finding submatrices that cannot be identified by standard clustering methods. We also note that LAS produces only 60 biclusters, while KM and HC produce 9,945 and 95,404 biclusters, respectively.

4.2.4 Summary Properties of Row and Column Sets

One potential benefit of biclustering methods over independent row-column clustering is that the sample-variable associations they identify are defined locally: they can, in principle, identify patterns of association that are not readily apparent from the summary statistics across rows and columns that drive k-means and hierarchical clustering. Nevertheless, local associations can sometimes be revealed by summary measures of variance and correlation, and it is worthwhile to consider the value of these quantities as a way of seeing (a) what drives different biclustering methods, and (b) the extent to which the local discoveries of these methods are reflected in more global summaries.

For each method under study, Table 2 shows the average, across the biclusters, of (i) the average pairwise correlation of their constituent genes, (ii) the average pairwise correlation of their constituent samples, (iii) the

	Correlation		Std. Deviation	
	Gene	Sample	Gene	Sample
Matrix	0.01	0.01	0.51	0.56
KM	0.21	0.22	0.52	0.54
HC	0.46	0.24	0.53	0.56
LAS	0.35	0.10	0.79	0.56
Plaid	0.03	0.03	0.56	0.56
CC	0.10	0.05	0.58	0.54
ISA	0.25	0.31	0.56	0.60
OPSM	0.52	0.06	0.55	0.58
SAMBA	0.26	0.02	0.93	0.56
BiMax	0.09	0.26	1.94	0.61
Subtypes		0.14		0.54

Table 2: Average standard deviation and average pairwise correlation of genes and samples, for biclusters, KM and HC clusters, and the whole data matrix. As a reference point, the last row shows the summary statistics for samples belonging to the same disease subtype.

average standard deviation of their constituent genes, and (iv) the average standard deviation of their constituent samples. Average values for the entire matrix are shown in the first row of the table. In each case, the summary statistics associated with the biclustering methods are higher than the average of these statistics over the entire matrix. As HC is based entirely on gene and sample correlations, we expect its correlation values to be large compared with other methods, and this is the case. The low values of gene correlation for KM result from the fact that we are using a relatively small numbers of gene clusters, which tend to have a large number of genes and therefore low average pairwise correlations. Similar remarks apply to the low gene (and sample) correlation values associated with Plaid.

BiMax seem to be driven by all four measures with gene correlation playing a relatively minor role. ISA is influenced by a mix of three measured; it is not affected by gene standard deviation. LAS appears to be driven by a mix of gene correlation and standard deviation. In each column, the average for LAS is less than and greater than those of two other methods. The average summary statistics of LAS do not appear to be extreme, or to reflect overtly global behavior. The remaining biclustering methods appear to be driven by two, or in some cases only one, of the measured summary statistics. We note that the average pairwise correlation of the samples in LAS biclusters best matches the average pairwise correlation of samples in the cancer subtypes (described in Subsection 4.3.1 below).

4.3 Biological Comparisons

The previous section compares LAS with other biclustering and IRCC methods on the basis of quantitative measures that are not directly related to biological or clinical features of the data. In this section we consider several biologically motivated comparisons. In particular, we carry out a number of tests that assess the gene and sample sets of each bicluster using auxiliary clinical information and external annotation. The next subsection considers sample-based measures of subtype capture.

4.3.1 Subtype capture

Breast cancer encompasses several distinct diseases, or subtypes, which are characterized by unique and substantially different expression signatures. Each disease subtype has associated biological mechanisms that are connected with its pathologic phenotype, and the survival profiles of patients [see Golub et al. (1999); Sorlie et al. (2001); Weigelt et al. (2005b); Hayes et al. (2006)]. Breast cancer subtypes were initially identified using hierarchical clustering of gene expression data, and have subsequently been validated in several datasets [see Fan et al. (2006)] and across platforms [see Sorlie et al. (2003)]. They are one focal point for our biological validation.

Hu et al. assigned each sample in the dataset to one of 5 disease subtypes (Basal-like, HER2-enriched, Luminal A, Luminal B, and Normal-like) using a nearest shrunken centroid predictor method of Tibshirani et al. (2002) and a pre-defined set of 1300 intrinsic genes. The centroids for the predictor were derived from the hierarchical clustering of of 300 samples chosen both for data quality, and the representative features of their expression profiles. In addition, each sample in the Hu dataset was assigned via a clinical assay to one of two estrogen receptor groups, denoted ER+ and ER-, which constitute the ER status of the tumor. The ER status of tumors is closely related to their subtypes: in the Hu dataset, HER2-enriched and Basal-like samples are primarily (74% and 94%) ER-negative, while Normal-like and Luminal A and B are primarily (83%, 86% and 91%) ER-positive.

Here we compare the ability of biclustering methods to capture the disease subtype and ER status of the samples. In order to assess how well the set of samples associated with a bicluster captures a particular subtype, we measured the overlap between the two sample groups using the p-value from a standard hypergeometric test (equivalent to a one-sided Fisher's exact test). For each biclustering method, we identified the bicluster that best matched each subtype, and recorded its associated p-value. As a point of comparison, we include the subtype match of column clusters produced by k-means and hierarchical clustering. The results are shown in in Figure 3.

The figure indicates that LAS captures ER status and disease subtypes better than the other biclustering methods, with the single exception of the Luminal A subtype, which was better captured by CC. In addition, LAS is competitive with KM and HC, performing better or as well as these methods on the Luminal A, Luminal B, Basal-like and HER2-enriched subtypes.

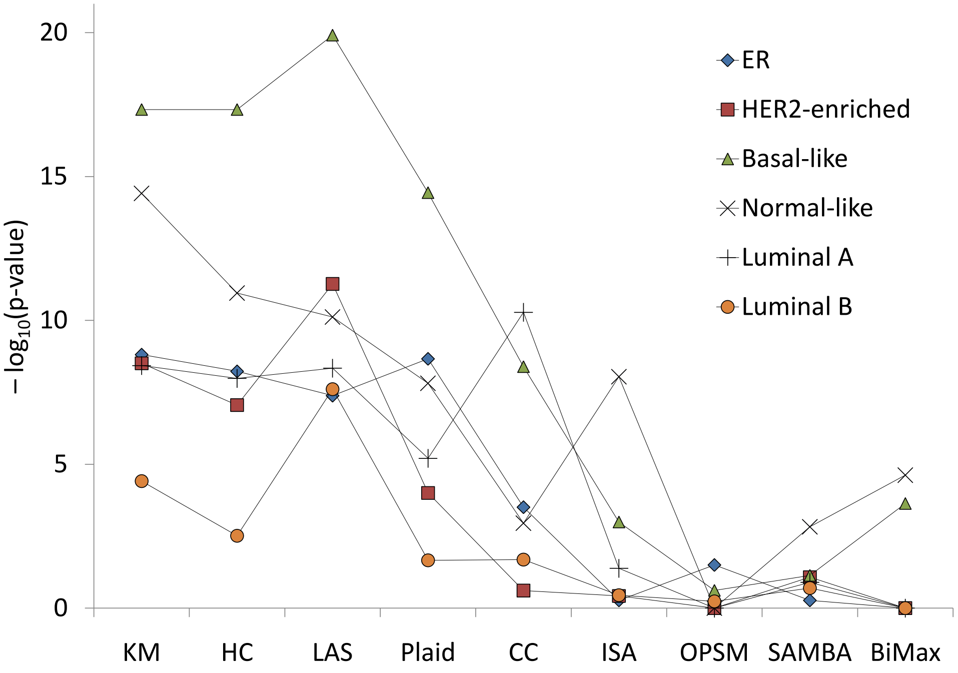

Figure 3: The minus $\log_{10}$ p-values of best subtype capture for different biclustering and sample clustering methods.

Another view of subtype capture is presented in the bar-plot of Figure 4. For the Basal-like disease subtype, the figure shows the number of true, missed and false discoveries associated with the the best sample cluster (as judged by hypergeometric p-value) that was produced by each method. The

Basal-like subtype contains 32 samples. The best LAS bicluster captured 27 of the 32 Basal-like samples with no false positives. Plaid had fewer missed samples, but a larger number of false positives, due to the large size of its sample clusters. As the disease subtypes were identified in part through the use of hierarchical clustering the good performance of KM and HC is unsurprising. Other biclustering methods were not successful in capturing Basal-like or other subtypes, due in part to the small number of samples in their biclusters. Barplots like Figure 4 for other subtypes are provided in the Supplementary Materials.

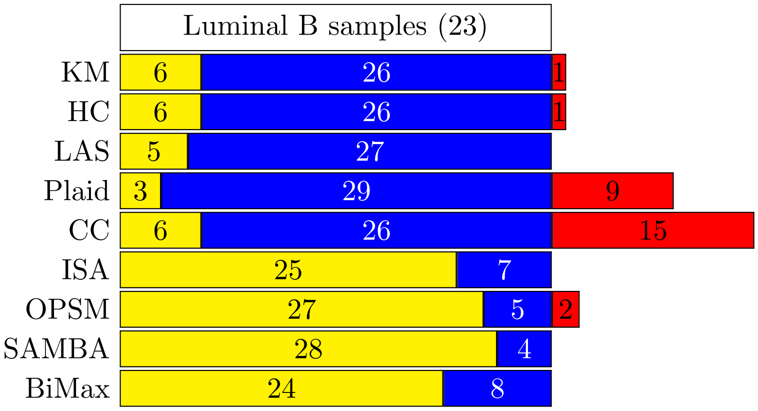

Figure 4: Bar-plot of missed, true and false discoveries for different biclustering methods and the Basal-like subtype. Bars represent: light - missed discoveries, dark - true discoveries, gray - falsely discoveries.

4.4 Biclusters of Potential Biological Interest

In order to assess the potential biological and clinical relevance of the biclustering methods under consideration, we applied three different tests to the gene and sample sets of each bicluster. The first test makes use of clinical information concerning patient survival. The second tests for over-representation of functional categories and genomic neighborhoods (cytobands) among the gene sets of different biclusters, and the third tests for the differential expression of these same gene categories between the sample set of a bicluster and its complement. The tests are described in more detail below.

We chose not to include KM and HC in this analysis for several reasons. The tests conducted here are intended to provide a rough biological assessment of the gene and sample sets of biclusters that are produced with the primary goal of capturing gene-sample associations. In this sense, the tests here are assessing secondary features of these methods. By contrast, gene and sample based tests are separately assessing the primary features of KM and HC, for which biclusters are a byproduct of their independent gene and sample grouping.

For 105 samples out of 117 in the dataset, we have information regarding overall survival (OS) and relapse free survival (RFS). We applied the standard logrank test [see Bewick et al. (2004)] to determine if there are significant differences between the survival times associated with samples in a bicluster, and the survival times associated with samples in its complement. Biclusters whose patients have significantly lower (or higher) survival rates are of potential clinical interest, as their gene sets may point to biological processes that play a deleterious (or beneficial) role in survival. A bicluster was called significant if its samples passed the log-rank test for overall or relapse free survival at the 5% level. The number of biclusters meeting the criterion is presented in the **Survival** column of Table 3.

We next tested the gene set of each bicluster for over-representation of biologically derived functional categories and genomic neighborhoods. For the former, we considered KEGG categories (Kyoto Encyclopedia of Genes and Genomes, Kanehisa and Goto (2000), `http://www.genome.jp/kegg/`). For the latter we considered cytobands, which consist of disjoint groups of genes such that the genes in a group have contiguous genomic locations. Definitions of KEGG and cytoband categories were taken from R metadata packages on Bioconductor (Bioconductor v 1.9, packages hgug4110b and hgu95av2).

For each bicluster gene set we computed a Bonferroni corrected hypergeometric p-value to assess its overlap with each KEGG category, and computed a similar p-value for each cytoband. We considered 153 KEGG categories and 348 cytobands that contained at least 10 genes (post filtering) on our sample arrays. A gene set was deemed to have significant overlap if any of the p-values computed in this way was less than 10^{-10}. This threshold was selected to adjust for the anti-conservative behavior of the hypergeometric test in the presence of even moderate levels of gene correlation (see Barry et al. (2005) for more details). The column **Gene** of Table 3 shows the number of biclusters having signficant overlap with at least one KEGG category or cytoband.

The third test concerns the differential expression of KEGG and cytoband categories across the sample set of a bicluster and its complement. From each bicluster we formed a treatment group consisting of the samples in the bicluster, and a control group consisting of the complementary samples which are not in the bicluster. We tested for KEGG categories showing differential expression across the defined treatment and control groups using the SAFE procedure of Barry et al., and counted the number of categories passing the test at the 5% level. The permutation based approach in SAFE accounts for multiple comparisons and the (unknown) correlation among genes. A similar testing procedure was carried out for cytobands.

If no KEGG category were differentially expressed across the treatment and control groups corresponding to a particular bicluster, roughly 5% of the categories would exhibit significant differential expression by chance. We considered a bicluster sample set to yield significant differential expression of KEGG categories if the number of significant categories identified by SAFE exceeds the 5th percentile of the $\mathrm{Bin}(153, .05)$ distribution. An analogous determination was made for cytobands. The number of biclusters whose sample sets yields significant differential expression for KEGG categories or cytobands is presented in the **Sample** column of the Table 3.

	# of	Survival	KEGG/Cytoband		2 out	All 3
	BC's	5% level	Gene	Sample	of 3	
LAS	60	10	15	24	11	1
Plaid	60	10	3	17	2	0
CC	60	8	0	12	2	0
ISA	72	2	18	23	5	0
OPSM	15	0	0	3	0	0
SAMBA	289	15	20	72	10	1
BiMax	1977	329	0	0	0	0

Table 3: The number of biclusters passing tests for survival, and gene-set enrichment and sample-set differential expression of KEGG categories and cytobands. A detailed description of the tests is given in the text.

The rightmost columns of Table 3 show the number of biclusters passing two or three tests. From an exploratory point of view, these biclusters are of potential interest, and represent a natural starting point for further experimental analysis. Accounting for the number (or effective number) of biclusters produced by each method, specifically the large output of SAMBA and the small output of OPSM, LAS outperformed the other methods under study, particularly in regards to biclusters satisfying two out of the three tests.

4.5 Classification

Biclustering algorithms identify distinguished sample-variable associations, and in doing so, can hope to capture and extract useful information about the data under study. To assess how much information about disease subtypes and ER status is captured by the *set* of biclusters produced by different methods, we examined the classification of disease subtypes using patterns of bicluster membership in place of the original expression measurements. Similar applications of biclustering for the purpose of classification were previously investigated in Tagkopoulos et al. (2005) and unpublished works of Grothaus (2005); Asgarian and Greiner (2006).

To be more specific, once biclusters have been produced from the data matrix, we replaced each sample by a binary vector whose jth entry is 1 if the sample belongs to the jth bicluster, and 0 otherwise. A simple k-nearest neighbor classification scheme using weighted Hamming distance was applied to the resulting binary matrix, using the subtype assignments of training samples, to classify unlabeled test samples. The number of rows in the derived binary matrix is equal to the number of biclusters; in every case this is far fewer than the number of genes in the original data.

To be more precise, let $X = [x_1, \ldots, x_n]$ be an $m \times n$ data matrix, and let $C_1, \ldots, C_K$ be the index sets of the biclusters produced from X by a given biclustering method. We map each sample (column) x_i into a binary vector $\pi(x_i) = (\pi_1(x_i), ..., \pi_K(x_i))^t$ that encodes its bicluster membership:

$$\pi_k(x_i) \;=\; \begin{cases} 1 & \text{if } x_i \text{ belongs to the sample set of bicluster } C_k \\ 0 & \text{otherwise.} \end{cases}$$

The original data matrix X is then replaced by the $K \times n$ "pattern" matrix $\Pi = \{\pi(x_1), \ldots, \pi(x_n)\}$. In the Hu data, for example, the 13,666 real variables in X are replaced by $K < 300$ binary variables in Π. Subtype and ER designations for the initial data matrix X carry over to the columns of Π.

For each of the breast cancer subtypes in the Hu data, we used 10-fold cross validation to assess the performance of a 5-nearest neighbor classification scheme applied to the columns of the binary pattern matrix Π. The nearest neighbor scheme used a weighted Hamming distance measure, in which the weight of each row is equal to the square of the t-statistic for the correlation r between the row and the response, $t^2 = (n-2)r^2/(1-r^2)$. In each case, the weights were calculated using only the set of training samples. For each subtype, the average number of cross-validated errors was divided by the total number of samples, in order to obtain an overall error rate. The results are displayed in Figure 5. For comparison, we include 10-fold cross

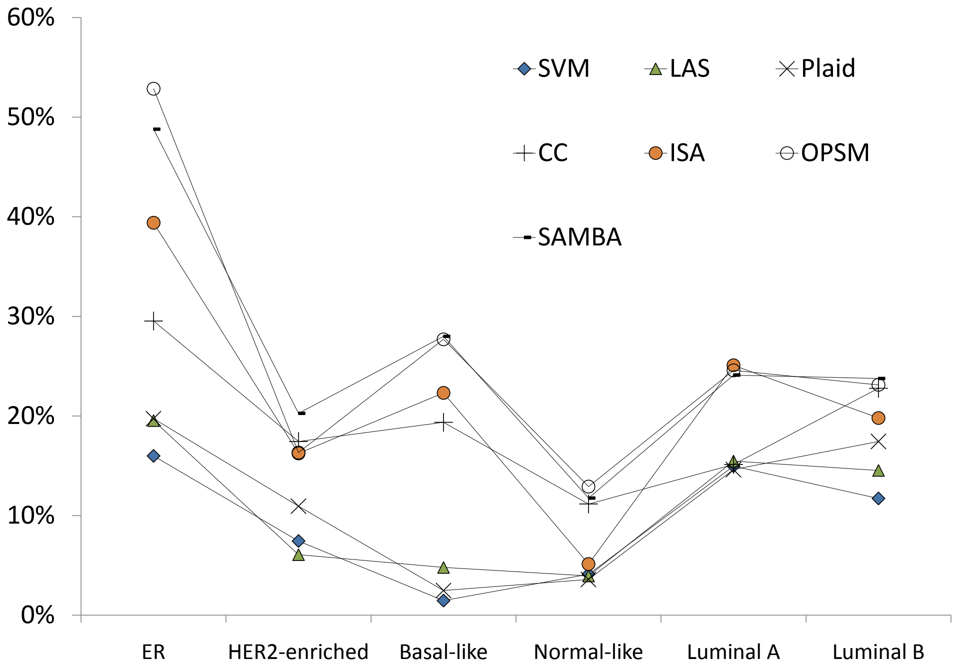

Figure 5: Classification error rates for SVM on the original data and the 5-nearest neighbor with weighted Euclidean distance applied to the “pattern” matrix.

validation error rates of a support vector machine (SVM) classifier applied to the original, expression-based data. As the figure shows, the error rates of LAS and Plaid are similar to those of SVM across the phenotypes under consideration. Using the pattern information from 60 biclusters, LAS and Plaid were able distinguish individual subtypes with the same degree of accuracy as SVM applied to the original data with 13,666 variables.

4.6 Lung Data

Validation results for the lung cancer data are contained in the Supplementary Materials. They are similar to the results for the breast cancer data considered above. The principal difference was the improved performance of ISA in tests of subtype capture. While ISA biclusters continued to have small sample sets, the disease subtypes for the lung data contained fewer samples than those in the breast data.

5 Simulations

In addition to real data, we also investigated the behavior of the LAS algorithm on a variety of artificially created datasets. Our primary goals were to assess (i) the ability of the algorithm to discover significant submatrices in a simple one-term model, (ii) the stability of the algorithm with respect to the initial random number seed, and (iii) the sensitivity of the algorithm to noise. The results of these simulations are described below. All relevant figures and tables appear in the Supplementary Materials.

5.1 Null Model with One Embedded Submatrix

The essential step in the LAS algorithm is to identify a submatrix of a given matrix that maximizes the score function. The approach taken by LAS is heuristic. As there are no efficient algorithms for finding optimal matrices, outside of small examples, we cannot check directly if the submatrix identified by LAS is optimal. In order to evaluate the LAS search procedure we generated a number of data matrices of the same size as the Hu dataset, with i.i.d. N(0,1) entries. For $k = 4, 8, 16, \ldots, 4096$ and $l = 4, 8, 16, 32$ we added a constant $\alpha = 0.1, 0.2, ..., 1$ to a $k \times l$ submatrix of the initial Gaussian matrix. The basic LAS search was carried out on each of the $11 \times 4 \times 10 = 440$ resulting matrices, and was considered a success if the search returned a bicluster whose score was at least as high as that of the embedded submatrix. The LAS search failed in only three cases; in each the embedded submatrix had relatively low scores (less than 100, while scores of other submatrices ranged up to 61,415.5). The search procedure was successful in all cases when the number of iterations used in the procedure was increased from 1,000 to 10,000.

5.2 Stability

In order to check the stability of LAS with respect to the randomization used in the basic search procedure, we ran LAS 10 times on the Hu dataset with the same parameters, but a different random seed. To assess the performance of the algorithm, rather than its raw output, for each of the 10 runs we calculated the validation measures from Section 4: effective number of biclusters, average size, subtype capture p-values as in Section 4.3.1, and number of biclusters that passed different biological tests as in Section 4.4. The results are presented in Supplementary Materials. There is little variation in calculated measures across different runs of the algorithm. The effective number of biclusters ranged from 48.2 to 49.0, and average size ranged from 355×26 to 363×27 The number of biclusters with significant survival ranged from 9 to 13 and number of biclusters having significant overlap with at least one KEGG category or cytoband ranged from 13 to 16. The SAFE analysis is extremely computationally intensive, so we did not perform it for these simulations. Although the output of LAS is not deterministic, its summary statistics for average size and overlap are stable, and it is consistently successful in capturing cancer subtypes.

5.3 Noise Sensitivity

In order to assess the the effect of noise on the LAS output, we added zero mean Gaussian noise with standard deviation $\sigma = 0, 0.1, 0.2, \ldots, 1$ to the normalized Hu dataset (after tail transformation and column standardization). The resulting matrix was then column standardized, and LAS was applied to produce 60 biclusters.

For each level of noise we calculated validation measures for the LAS output; these are presented in the Supplementary Materials. As the level of noise increases, the average number of genes in LAS biclusters decreased, as well as number of biclusters with significant Cytoband or KEGG category. However, within the tested range of noise levels the average number of samples did not change noticeably, and the subtype capture performance did not markedly decrease. The results indicate both high noise resistance of LAS and the remarkable strength of subtype signal.

6 Discussion

Biclustering methods are a potentially useful means of identifying sample-variable associations in high-dimensional data, and offer several advantages over independent row-column clustering. Here we have presented a statistically motivated biclustering algorithm called LAS that searches for large average submatrices. The algorithm is driven by a simple significance-based score function, which is derived from a Bonferroni corrected p-value under a Gaussian random matrix null model. We show that maximizing the LAS score function is closely related to minimizing the overall description length of the data in a block-additive Gaussian model.

The LAS algorithm operates in a sequential-residual fashion; at each stage the search for a submatrix with maximum score is carried out by a randomly initialized iterative search procedure that is reminiscent of EM type algorithms. The only operational parameters of LAS are the number of biclusters it produces before halting, and the number of randomized searches carried out in identifying a bicluster. In our experiments on real data, we found that 1000 randomized searches per bicluster was enough to ensure stable performance of the algorithm.

We evaluated LAS along with a number of competing biclustering methods, using a variety of quantitative and biological validation measures. On the quantitative side, LAS produced biclusters exhibiting a wide range of gene and sample sizes, and low to moderate overlap. The former feature implies that LAS is capable of capturing sample-variable associations across a range of different scales, while the latter indicates that distinct LAS biclusters tend to capture distinct associations. Other methods varied considerably in their sizes and overlap, with a number of methods producing biclusters having a small number of samples and genes.

Many LAS biclusters had significantly higher scores than biclusters obtained by the more traditional KM and HC methods, which are based, respectively, on k-means and hierarchical clustering. This suggests that the constraints associated with independent row-column clustering (considering rows and columns separately, assigning each row or column to a single cluster) substantially limit the ability of these methods to identify significant biclusters, and that more flexible methods may yield substantially better results.

In regards to capturing disease subtypes, LAS was competitive with, and often better than, KM and HC. Other methods did not perform particularly well, though we note that ISA did a good job of capturing and classifying the smaller disease subtypes present in the lung cancer data. In tests for survival, over-representation of functional categories, and differential expression of functional categories, LAS outperformed the other biclustering methods. These tests, unlike the quantitative measures of size and overlap, were based on clinical and biological information.

The classification study in Section 4.5 shows that simple binary profiles of bicluster membership can contain substantive information about sample biology. In particular, nearest neighbor classification of disease subtypes using the membership profiles derived from LAS or PLAID was competitive with a support vector machine classifier applied to the full set of expression data. We note that the biclustering methods applied here are unsupervised, and depend only on the expression matrix: none makes use of auxiliary information about samples or variables.

Our simulation study shows that the LAS search procedure is effective at capturing embedded submatrices (or more significant ones) having moderate scores. Although the search procedure makes use of random starting values, its performance is stable across different random seeds. The ability of the algorithm to capture subtypes does not substantially deteriorate when a moderate amount of noise added to the data matrix, which leads strength to a broader application of LAS across different high dimensional data.

While the validation of biclustering here has focused on gene expression measurements, it is important to note that LAS and other biclustering methods are applicable to a wide variety of high-dimensional data. In preliminary experiments on high density array CGH data produced on the Agilent 244k Human Genome CGH platform, LAS was able to capture known regions of duplications and deletion (data not shown). The dataset contained roughly 250 samples and 240,000 markers. We note that among the seven biclustering methods compared in the paper only LAS and Plaid were able to computationally handle datasets of this size.

LAS biclusters capture features of the data that are of potential clinical and biological relevance. Although some findings, such as disease subtypes, are already known, very often the methods used to establish them involve a good deal of subjective intervention by biologists or disciplinary scientists. LAS provides a statistically principled alternative, in which intervention (to select among biclusters those that may be of interest) can take place after the initial discovery process is complete.

Appendix

Minimum Description Length connection

The code describing the data Let the data matrix X be standardized (have zero mean and unit variance of the elements) and let U be the selected bicluster. The code describing the data matrix has to preserve the information about both bicluster (size, location, average of its elements) and the residual matrix.

Coding the bicluster It is not possible to code the real-valued data precisely with a finite-length code, so we will construct a code describing the data with a given precision of C binary digits after the period.

The size of submatrix is defined by variables $k \in [m]$ and $l \in [n]$. Coding these variables requires $\log_2(mn)$ bits (we ignore rounding issues here and in what follows). There are total $\binom{m}{k}\binom{n}{l}$ different $k \times l$ submatrices in $m \times n$ matrix, so the code describing the location of the submatrix takes $\log_2[\binom{m}{k}\binom{n}{l}]$ bits of code. To code the submatrix average we assume that it lies in the interval $[-8, 8]$ (we did not observe $|\tau| > 1.5$ in our experiments). Then the code describing the average τ of the submatrix U takes $-\log_2(1/16) + C = 4 + C$.

Coding the residuals Finally, we describe the residual matrix. The dataset is standardized, so its total variation (sum of squares) is nm. A $k \times l$ submatrix with average τ explains the variation of $\tau^2 kl$. This means that the variation of the residual matrix is $nm - \tau^2 kl$.

It follows that the remaining variation is $nm - \tau^2 kl = nm \left[1 - \frac{kl\tau^2}{nm}\right]$ So the elements of the residual matrix are approximately distributed as $N(0, 1 - \frac{kl\tau^2}{nm})$.

Coding of a random variable X with density $f(x)$ takes $-\log_2(f(X)) + C$ bits, that is on average $-\int \log_2(f(x))f(x)dx + C$. Let C_N be the average code length for standard normal random variable, $C_N = -E\log_2(\phi(z)) + C$ where $z \sim N(0, 1)$ and $\phi(z)$ is density of standard normal distribution. Then for $x \sim N(0, \sigma^2)$ average code length is $C_N - \log_2(\sigma^2)/2$.

Thus coding of the residual matrix takes on average $nm[C_N - \log_2 \left[1 - \frac{kl\tau^2}{nm}\right]]$ bits of code.

Total code length So the length of the code describing the data using a $k \times l$ bicluster with average τ is:

$$MDL(U) = \log_2(nm) + 4 + C + \log_2[\binom{m}{k}\binom{n}{l}] + nm[C_N - \log_2\left[1 - \frac{kl\tau^2}{nm}\right]/2]$$

In all practical applications the explained variation was a small fraction ($< 1/1000$) of the total variation. Thus we can apply first order approximation: $\log_2[1 + x] = x/\ln(2) + o(x)$. Then

$$MDL(U) \approx const + \log_2[\binom{m}{k}\binom{n}{l}] - kl\tau^2/2\ln(2)$$

We pulled the constant terms and terms depending on n and m out as they do not depend on the selected bicluster.

LAS score function Let's now consider the LAS score function.

$$S(U) = -\log\binom{m}{k} - \log\binom{n}{l} - \log(\Phi(-\sqrt{v^2 kl}))$$

For large x we can approximate $\Phi(-x) = \exp[-x^2/2]/x + o(x)$, getting

$$S(U) \approx \ln(2)\left[-\log_2 \binom{m}{k}\binom{n}{l} + \tau^2 kl/2\ln(2) - \log_2(\tau^2 kl)/2\right]$$

Easy to see that except for the small factor of $\log_2(\tau^2 kl)/2$ the code length and score function approximations are linearly dependent:

$$S(U) \approx const - \ln(2)MDL(U)$$

Running configurations for other methods

All other biclustering methods described in this paper were run on the same machine as LAS: AMD64 FX2 2.8GHz, 4GB RAM, running Windows XP Professional (64 bit). The same imputed dataset as run through LAS was loaded into the other programs. If a the method was written in Java, the 'Xmx1024m' key was added to the command line for proper memory allocation. In all cases, we preferred to use the default running parameters as given by the software used to run the algorithms (*BicAT* for BiMax, CC, ISA, OPSM and *Expander* for SAMBA).

Running parameters. **Plaid**, as it is scripting based, a script was written to iterate over the steps $findm$, accept, shuffle 60 times, to produce 60 biclusters. **Cheng-Church**: *seed* = 13, $\Delta = 0.1$, $\alpha = 1.2$, $NumberOutput$ = 30, **ISA**: *seed* = 13, *t_g* = 2, *t_c* = 2, *StartingNum* = 100, **OPSM**: $PassedModels = 10$, **BiMax**: $Gene_{min} = 10$, $Sample_{min} = 5$, **SAMBA**: try covering all probes, $OptionFiles = valsp\_3ap$, $OverlapPrior = 0.1$, $ProbesToHash = 100$, $Memory_{max} = 500$, $HashKernal_{min} = 4$, $HashKernal_{max} = 7$. The *OverlapPrior* value ensures that for each new cluster generated, its elements were 90% unique to any previously discovered bicluster.

References

Aggarwal, C., Wolf, J., Yu, P., Procopiuc, C., and Park, J. 1999. Fast algorithms for projected clustering. *Proceedings of the 1999 ACM SIGMOD international conference on Management of data*, 61–72.

Asgarian, N. and Greiner, R. 2006. Using rank-1 biclusters to classify microarray data. *Department of Computing Science, and the Alberta Ingenuity Center for Machine Learning, University of Alberta, Edmonton, AB, Canada, T6G2E8*.

Barron, A. and Yu, J. 1998. The minimum description length principle in coding and modeling. *Information Theory, IEEE Transactions on 44,* 6, 2743–2760.

Barry, W., Nobel, A., and Wright, F. 2005. Significance analysis of functional categories in gene expression studies: a structured permutation approach. *Bioinformatics 21,* 9, 1943–1949.

Ben-Dor, A., Chor, B., Karp, R., and Yakhini, Z. 2003. Discovering local structure in gene expression data: The order-preserving submatrix problem. *Journal of Computational Biology 10,* 3-4, 373–384.

Bewick, V., Cheek, L., and Ball, J. 2004. Statistics review 12: Survival analysis. *Critical Care 8,* 5, 389–394.

Bhattacharjee, A., Richards, W. G., Staunton, J., Li, C., Monti, S., Vasa, P., Ladd, C., Beheshti, J., Bueno, R., Gillette, M., Loda, M., Weber, G., Mark, E. J., Lander, E. S., Wong, W., Johnson, B. E., Golub, T. R., Sugarbaker, D. J., and Meyerson, M. 2001. Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. *Proceedings of the National Academy of Sciences 98,* 24, 13790–13795.

Cheng, Y. and Church, G. 2000. Biclustering of expression data. *Proc Int Conf Intell Syst Mol Biol 8*, 93–103.

Dhillon, I. 2001. Co-clustering documents and words using bipartite spectral graph partitioning. *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, 269–274.

Eisen, M., Spellman, P., Brown, P., and Botstein, D. 1998. Cluster analysis and display of genome-wide expression patterns.

Fan, C., Oh, D. S., Wessels, L., Weigelt, B., Nuyten, D. S., Nobel, A. B., van't Veer, L. J., and Perou, C. M. 2006. Concordance among Gene-Expression-Based Predictors for Breast Cancer. *N Engl J Med 355,* 6, 560–569.

Friedman, J. and Meulman, J. 2004. Clustering objects on subsets of attributes. *Journal of the Royal Statistical Society Series B(Statistical Methodology) 66,* 4, 815–849.

Getz, G., Levine, E., and Domany, E. 2000. Coupled two-way clustering analysis of gene microarray data. *Proceedings of the National Academy of Sciences 97,* 22, 12079.

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., and Lander, E. S. 1999. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science 286,* 5439, 531–537.

Grothaus, G. 2005. Biologically-Interpretable Disease Classification Based on Gene Expression Data.

Grunwald, P. 2004. A tutorial introduction to the minimum description length principle. *Arxiv preprint math.ST/0406077*.

Hastie, T., Tibshirani, R., Eisen, M., Alizadeh, A., Levy, R., Staudt, L., Chan, W., Botstein, D., and Brown, P. 2000. Gene shavingas a method for identifying distinct sets of genes with similar expression patterns. *Genome Biol 1,* 2, 1–21.

Hayes, D. N., Monti, S., Parmigiani, G., Gilks, C. B., Naoki, K., Bhattacharjee, A., Socinski, M. A., Perou, C., and Meyerson, M. 2006. Gene Expression Profiling Reveals Reproducible Human Lung Adenocarcinoma Subtypes in Multiple Independent Patient Cohorts. *J Clin Oncol 24,* 31, 5079–5090.

Hu, Z., Fan, C., Oh, D., Marron, J., He, X., Qaqish, B., Livasy, C., Carey, L., Reynolds, E., Dressler, L., Nobel, A., Parker, J., Ewend, M., Sawyer, L., Wu, J., Liu, Y., Nanda, R., Tretiakova, M., Orrico, A., Dreher, D., Palazzo, J., Perreard, L., Nelson, E., Mone, M., Hansen, H., Mullins, M., Quackenbush, J., Ellis, M., Olopade, O., Bernard, P., and Perou, C. 2006. The molecular portraits of breast tumors are conserved across microarray platforms. *BMC Genomics 7,* 1, 96.

Ihmels, J., Friedlander, G., Bergmann, S., Sarig, O., Ziv, Y., and Barkai, N. 2002. Revealing modular organization in the yeast transcriptional network. *Nat Genet 31,* 4, 370–7.

Jiang, D., Tang, C., and Zhang, A. 2004. Cluster analysis for gene expression data: a survey. *Knowledge and Data Engineering, IEEE Transactions on 16,* 11, 1370–1386.

Kanehisa, M. and Goto, S. 2000. KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research 28,* 1, 27–30.

Kluger, Y., Basri, R., Chang, J., and Gerstein, M. 2003. Spectral biclustering of microarray data: Coclustering genes and conditions.

Lazzeroni, L. and Owen, A. 2002. Plaid models for gene expression data. *Statistica Sinica 12,* 1, 61–86.

Liu, J., Yang, J., and Wang, W. 2004. Biclustering in gene expression data by tendency. *Computational Systems Bioinformatics Conference, 2004. CSB 2004. Proceedings. 2004 IEEE*, 182–193.

Madeira, S. and Oliveira, A. 2004. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics 1,* 1, 24–45.

Parsons, L., Haque, E., and Liu, H. 2004. Subspace clustering for high dimensional data: a review. *ACM SIGKDD Explorations Newsletter 6,* 1, 90–105.

Prelic, A., Bleuler, S., Zimmermann, P., Wille, A., Buhlmann, P., Gruissem, W., Hennig, L., Thiele, L., and Zitzler, E. 2006. A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics 22,* 9, 1122.

Rissanen, J. An introduction to the MDL principle.

Sorlie, T., Perou, C. M., Tibshirani, R., Aas, T., Geisler, S., Johnsen, H., Hastie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Thorsen, T., Quist, H., Matese, J. C., Brown, P. O., Botstein, D., Lonning, P. E., and Borresen-Dale, A.-L. 2001. Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences 98,* 19, 10869–10874.

Sorlie, T., Tibshirani, R., Parker, J., Hastie, T., Marron, J. S., Nobel, A., Deng, S., Johnsen, H., Pesich, R., Geisler, S., Demeter, J., Perou, C. M., Lonning, P. E., Brown, P. O., Borresen-Dale, A.-L., and Botstein, D. 2003. Repeated observation of breast tumor subtypes in independent gene expression data sets. *Proceedings of the National Academy of Sciences 100,* 14, 8418–8423.

TAGKOPOULOS, I., SLAVOV, N., AND KUNG, S. 2005. Multi-class biclustering and classification based on modeling of gene regulatory networks. *Bioinformatics and Bioengineering, 2005. BIBE 2005. Fifth IEEE Symposium on*, 89–96.

TAMAYO, P., SLONIM, D., MESIROV, J., ZHU, Q., KITAREEWAN, S., DMITROVSKY, E., LANDER, E., AND GOLUB, T. 1999. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation.

TANAY, A., SHARAN, R., AND SHAMIR, R. 2002. Discovering statistically significant biclusters in gene expression data. *Bioinformatics 18,* Suppl 1, S136–S144.

TIBSHIRANI, R., HASTIE, T., NARASIMHAN, B., AND CHU, G. 2002. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences 99,* 10, 6567.

WANG, H., WANG, W., YANG, J., AND YU, P. 2002. Clustering by pattern similarity in large data sets. *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, 394–405.

WEIGELT, B., HU, Z., HE, X., LIVASY, C., CAREY, L., EWEND, M., GLAS, A., PEROU, C., AND VAN'T VEER, L. 2005a. Molecular portraits and 70-gene prognosis signature are preserved throughout the metastatic process of breast cancer.

WEIGELT, B., HU, Z., HE, X., LIVASY, C., CAREY, L. A., EWEND, M. G., GLAS, A. M., PEROU, C. M., AND VAN'T VEER, L. J. 2005b. Molecular Portraits and 70-Gene Prognosis Signature Are Preserved throughout the Metastatic Process of Breast Cancer. *Cancer Res 65,* 20, 9155–9158.

WEINSTEIN, J. N., MYERS, T. G., O'CONNOR, P. M., FRIEND, S. H., FORNACE, ALBERT J., J., KOHN, K. W., FOJO, T., BATES, S. E., RUBINSTEIN, L. V., ANDERSON, N. L., BUOLAMWINI, J. K., VAN OSDOL, W. W., MONKS, A. P., SCUDIERO, D. A., SAUSVILLE, E. A., ZAHAREVITZ, D. W., BUNOW, B., VISWANADHAN, V. N., JOHNSON, G. S., WITTES, R. E., AND PAULL, K. D. 1997. An information-intensive approach to the molecular pharmacology of cancer. *Science 275,* 5298, 343–349.