Diametrical Clustering for identifying anti-correlated gene clusters

Inderjit S. Dhillon * Edward M. Marcotte † Usman Roshan*

Abstract: Motivation: Clustering genes based upon their expression patterns allows us to predict gene function. Most existing clustering algorithms cluster genes together when their expression patterns show high positive correlation. However, it has been observed that genes whose expression patterns are strongly anti-correlated can also be functionally similar. Biologically, this is not unintuitive — genes responding to the same stimuli, regardless of the nature of the response, are more likely to operate in the same pathways.
Results: We present a new *diametrical clustering* algorithm that explicitly identifies anti-correlated clusters of genes. Our algorithm proceeds by iteratively (i) re-partitioning the genes and (ii) computing the dominant singular vector of each gene cluster; each singular vector serving as the prototype of a "diametric" cluster. We empirically show the effectiveness of the algorithm in identifying diametrical or anti-correlated clusters. Testing the algorithm on yeast cell cycle data, fibroblast gene expression data, and DNA microarray data from yeast mutants reveals that opposed cellular pathways can be discovered with this method. We present systems whose mRNA expression patterns, and likely their functions, oppose the yeast ribosome and proteosome, along with evidence for the inverse transcriptional regulation of a number of cellular systems.
Availability: See http://www.cs.utexas.edu/users/usman/diametrical for the experimental results. Software is available on request.
Contact: usman@cs.utexas.edu, inderjit@cs.utexas.edu, marcotte@icmb.utexas.edu
Keywords: DNA microarrays, gene expression, clustering, anti-correlated clusters.

1 Introduction & Motivation

DNA microarrays simultaneously measure the mRNA expression of thousands of genes in a single experiment (DeRisi *et al.*, 1997), typically measuring expression of every gene encoded by a genome. From sets of DNA microarray experiments, an expression vector for each gene can be constructed, describing the expression of the gene under a range of cellular conditions, cell types, genetic backgrounds, etc.

A key step in the analysis of gene expression data is the *clustering* of genes into groups that show similar expression values over a wide range of experiments. Given enough independent experiments, genes clustered in this fashion tend to be functionally related (Eisen *et al.*, 1998; Marcotte *et al.*, 1999).

There is already a wealth of work in cluster analysis of genes, ranging from hierarchical clustering (Eisen *et al.*, 1998), self-organizing maps (Tamayo *et al.*, 1999), neural networks (Herrero *et al.*, 2001), simulated annealing (Lukashin & Fuchs, 2001), algorithms based on principal components analysis (Hastie *et al.*, 2000) and graph-based algorithms (Sharan & Shamir, 2000). Most of these algorithms use some measure of correlation between expression vectors, such as correlation coefficient, and tend to put those genes in one cluster that show strong positive correlation between their expression vectors. However, as observed by (Shatkay *et al.*, 2000):

> "Genes that are functionally related may demonstrate strong anti-correlation in their expression levels, a gene may be strongly suppressed to allow another to be expressed, thus clustered into separate groups, blurring the (functional) relationship between them."

In general, we often expect the genes in a given cellular pathway to be co-expressed (positively correlated) to some extent. Genes whose expression is anti-correlated with these might include members of a pathway whose action is opposed to that of the first pathway (Qian *et al.*, 2001). We expect anti-correlated expression patterns from genes which repress the expression of other genes, often genes involved in the same biological pathway.

In this paper, we pose the goal of detecting anti-correlated gene clusters. This provides us a way to *explicitly* look for opposed systems of genes, and also to investigate functional similarity between such opposed clusters.

To achieve this goal, we propose a new clustering algorithm which puts strongly correlated *and* anti-correlated genes into the same "diametric" cluster. A simple post-processing step separates the positively correlated genes from the negatively correlated genes. Our clustering algorithm resembles the k-means procedure (Jain & Dubes, 1988), in that it iteratively alternates between (i) reallocation of cluster members and (ii) computation of "prototypes" of the new clusters. In k-means, each cluster's "prototype" is the centroid (or mean) of its constituent members. However, this simple strategy would break down for our goal since each cluster contains positively and negatively-correlated genes. In our algorithm, each cluster's prototype turns out to be the dominant singular vector of the matrix whose rows comprise the cluster members. This strategy effectively identifies diametric clusters.

In this paper, we first discuss some similarity measures used in clustering, then introduce the algorithm to detect anti-correlated clusters. The algorithm is applied to three sets of mRNA expression data, providing evidence for the inverse transcriptional regulation of several cellular systems. A word

*Department of Computer Sciences, University of Texas, Austin, TX 78712; (512) 471-9725; Fax: (512) 471-8885; {inderjit,usman}@cs.utexas.edu
†Department of Chemistry and Biochemistry, University of Texas, Austin, TX 78712; marcotte@icmb.utexas.edu

about notation: small letters such as g, h, x and v will denote vectors, capital letters such as A, G denote matrices. Also, $g^T h$ denotes the usual inner product between vectors.

2 Similarity Measures and Algorithm

Similarity Measures: Gene expression data from a set of microarray experiments is typically presented as an $m \times n$ matrix G in which the rows correspond to genes, the columns to experiments, and the (i, j) entry in the matrix corresponds to the expression level of gene i in the j-th experiment. Note that m is the total number of genes, while n is the number of experiments.

Most clustering algorithms require a similarity (or distance) measure. A popular gene expression similarity measure is the correlation coefficient (Eisen *et al.*, 1998). For n-dimensional gene vectors g and h, the correlation coefficient is defined as:

$$S(g,h) \;=\; \frac{1}{n}\sum_{i=1}^{n}\left(\frac{g_i-\mu_g}{\sigma_g}\right)\left(\frac{h_i-\mu_h}{\sigma_h}\right) \qquad (1)$$

where g_i is the expression level of gene g in the i-th experiment, μ_g is a number usually taken to be the mean of all expression levels of g, and $\sigma_g = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(g_i-\mu_g)^2}$. When μ_g and μ_h are taken as the means of values in g and h respectively, then $S(g,h)$ is exactly equal to the Pearson correlation coefficient, which is a measure that captures the linear relationship between the observations g_i and h_i, $i = 1, \ldots, n$. When μ_g is set to 0, $S(g,h)$ equals the cosine of the angle between g and h.

By shifting each gene vector by its mean and then normalizing it to have unit norm, the Pearson correlation coefficient equals the inner product between the (transformed) gene vectors. Specifically, by making the transformations $\tilde{g}_i = \frac{g_i-\mu_g}{\sum_{j=1}^{n}(g_j-\mu_g)^2}$, $1 \le i \le n$, to every gene vector, the correlation coefficient in (1) may be written as the inner product between two unit vectors, i.e., $S(g,h) = \tilde{g}^T\tilde{h}$. In this paper, we perform such data transformations before clustering. The inner product has been used previously as a measure of similarity, for example see (Sharan & Shamir, 2000) and (Brown *et al.*, 2000). Note that each transformed gene vector g resides on the unit (hyper)sphere in n-dimensional space.

Algorithm: Our goal is to find clusters containing genes that are either highly positively correlated or highly negatively correlated. Hence, an obvious similarity measure is the square of the correlation coefficient (Graybill & Iyer, 1994), i.e.,

$$S(g,h) \;=\; (g^T h)^2, \qquad (2)$$

where g and h are gene vectors with mean 0 and norm 1. This measure is high (close to 1) if the genes have high positive or negative correlation. Having chosen a similarity measure, we need an appropriate clustering algorithm.

The popular k-means algorithm is efficient but unsuitable with this similarity measure. Given a cluster which contains genes that have high positive as well as negative correlation, it would be incorrect to use the cluster centroid (or mean) as the "cluster prototype" as is done in the traditional k-means algorithm. Thus we need a definition of "cluster prototype" compatible with the squared correlation coefficient.

Given a cluster C_j of genes, the natural question to ask is: what cluster prototype (or representative) vector x_j is closest, on average, to all the gene vectors in the cluster using the similarity measure in (2). Mathematically, we find a unit vector x_j such that the sum

$$\sum_{g\in C_j}(g^T x_j)^2 = \sum_{g\in C_j} x_j^T(gg^T)x_j = x_j^T\left(\sum_{g\in C_j} gg^T\right)x_j$$

is maximized. The optimal solution is achieved when x_j equals the dominant right singular vector of the matrix G_j whose rows comprise all the gene vectors in the cluster (Golub & Loan, 1996). Thus, given a clustering $C_1, \ldots, C_k$ we can measure its quality by the total squared correlation coefficient

$$Q(C_1,C_2,\ldots,C_k) \;=\; \sum_{j=1}^{k}\sum_{g\in C_j}(g^T v_j)^2, \qquad (3)$$

where v_j is the dominant singular vector of cluster C_j. Our goal of finding k *diametric* clusters can be posed as the search for clusters that maximize this quality.

Algorithm Diametrical_Clustering(G,k)

Input: G is the $m \times n$ gene-expression matrix where m is the no. of genes and n is the no. of experiments,
k is the number of desired diametric clusters.

Phase I:

1. Initialize the k clusters, and compute the dominant right singular vector v_i of each cluster sub-matrix G_i for $i = 1, \ldots, k$.
2. Re-compute all clusters: for each gene g find its new cluster index as
$$j^*(g) = \mathrm{argmax}_i (g^T v_i)^2,$$
resolving ties arbitrarily. Thus compute the new gene clusters C_j, $1 \le j \le k$, as
$$C_j = \{g : j^*(g) = j\}.$$
3. Re-compute $v_1, \ldots, v_k$ to be the dominant right singular vectors of the new cluster sub-matrices $G_1, \ldots, G_k$ respectively.
4. If the difference in $Q(C_1, \ldots, C_k)$ (see (3)) between two consecutive iterations is less than δ go to Phase II, else go to step 2 above.

Phase II:

1. For each diametric cluster C_i output the 2 anti-correlated clusters: $C_{i,0} = \{g \in C_i \;\&\; g^T v_i \ge 0\}$, $C_{i,1} = \{g \in C_i \;\&\; g^T v_i < 0\}$, and their normalized centroid (mean) vectors as the cluster "fingerprints".

Figure 1: Algorithm for diametrical clustering

Figure 1 gives an algorithm that searches for such a clustering. Phase I of the algorithm alternates between two steps: (a) obtain a new clustering based on the closeness of genes to the current set of singular vectors, (b) re-compute the set of singular vectors for this new clustering. The dominant singular vector of each of the clusters can be efficiently computed by using power iteration or the faster converging Lanczos algorithm (Golub & Loan, 1996). Each iteration always increases the quality measure given in (3) (a proof is given in the appendix). Thus the quality measure will converge to a limiting value, and the iteration is guaranteed to terminate with an appropriate convergence criterion (see Dhillon & Modha (2001)).

Phase II of the algorithm separates each diametric cluster into a pair of anti-correlated clusters. As shown in Figure 1 this is done by simply separating the genes in each diametric cluster C_i according to whether they have positive or negative inner product with the cluster's singular vector, i.e., $g^T v_i$ is positive or negative. Note that the algorithm *does not force* a diametric or anti-correlated structure on the data. If the data set does not have anti-correlated clusters then one of the clusters found in Phase II will be empty.

The algorithm is computationally efficient: the time taken is $O(mnk\tau)$ where τ is the number of iterations required — experimental results show that 15-20 iterations are typical. In the rest of the paper we will refer to the Phase I clusters as diametric clusters, and the Phase II clusters as anti-correlated clusters.

3 Experimental Results

We analyzed three large sets of mRNA expression data. First, we analyzed human fibroblast gene expression data (Iyer et al., 1999) reporting the response of human fibroblasts after addition of serum to the growth media. This data set contains expression levels for the 517 human genes whose expression changed substantially following serum stimulation. The data (12 time points and an unsynchronized sample) was preprocessed by dividing each entry by the unsynchronized sample expression level, taking the log of the result, then normalizing each 12-element expression vector to have unit L^2 norm.

Two yeast data sets were analyzed. The first consists of gene expression data from synchronized yeast cultures growing through several phases of the cell cycle (Spellman et al., 1998). The data represents 82 time points from yeast cultures synchronized by four independent methods for a subset of 696 genes which have at most four missing values. Each gene vector was normalized to have mean 0 and norm 1. The second yeast dataset is that of Rosetta Inpharmatics (Hughes *et al.*, 2000), consisting of 300 experiments measuring expression of 6,048 yeast genes, in which transcript levels of a mutant or compound-treated culture were compared to those of a wild-type or mock-treated culture. We examined the subset of 5,246 genes which had no missing expression measurements, and normalized each 300-element expression vector to have unit L^2 norm.

Validating anticorrelated mRNA expression: To test the extent of anticorrelated gene expression, we measured functional relatedness of anti-correlated genes. We took the 1,174 yeast genes with functional annotation in the KEGG pathway database (Kanehisa & Goto, 2000), then measured the correlation coefficients between the Rosetta expression vectors of all pairs of the annotated yeast genes. We represented each gene's function as a set containing KEGG categories, which allowed us to compute the Jaccard coefficient between the KEGG categories (Verjovsky Marcotte & Marcotte, 2002) of every gene pair. The Jaccard coefficient of two sets A and B is defined as $\frac{|A\cap B|}{|A\cup B|}$ where $|A|$ denotes the size of set A. In Figure 2, we have plotted the mean Jaccard coefficient versus the correlation coefficient of the expression vectors. As expected, genes with co-expression (high positive correlation coefficients) show strong functional relatedness (i.e. large Jaccard coefficients). However, genes with anti-correlated expression (high negative correlation coefficients) also show functional similarity, validating the search for anti-correlated gene expression clusters. We observed similar results when function keywords were obtained from the SWISS-PROT database (Bairoch & Apweiler, 2000).

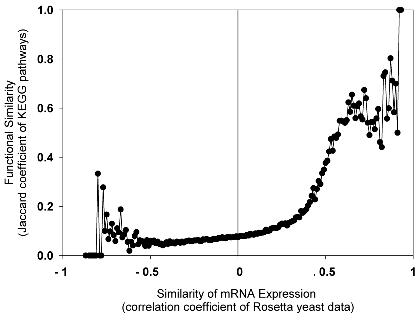


Figure 2: Yeast genes with both highly correlated and highly anti-correlated mRNA expression patterns tend to operate in similar cellular pathways.

Analysis of diametrical clusters: We applied diametrical clustering to the human fibroblast and Rosetta yeast expression sets. In general, analysis of the human data revealed that systems downregulated upon serum stimulation are systematically understudied. The yeast data revealed a number of presumably coordinately regulated, opposed cellular systems which have not been previously observed.

Human fibroblasts: We applied our algorithm to obtain 5 diametric clusters in Phase I which were separated into 10 anti-correlated clusters in Phase II. Again, the diametrical clustering algorithm nicely identifies genes with opposed expression patterns (see Figure 3).

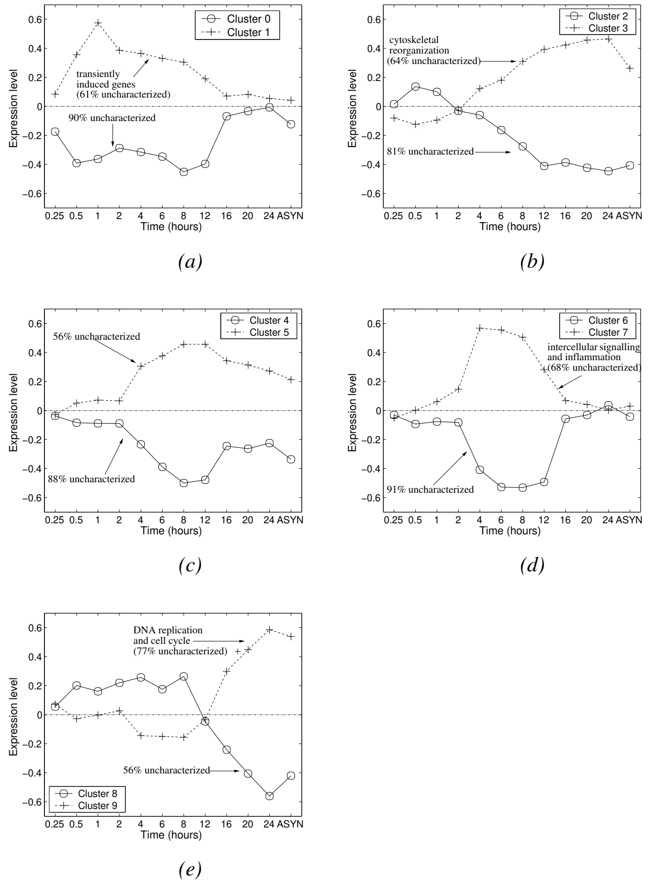


Figure 3: Human genes responding to serum stimulation exhibit diametric mRNA expression patterns. In each figure are plotted the mean expression profiles of two opposed clusters obtained on the fibroblast data set. Systems which are downregulated in response to serum stimulation are seen to be systematically understudied.

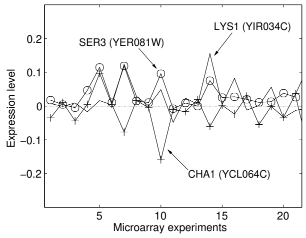
Known relationships: In general, we find the systems induced by serum addition are partly characterized, but the systems turned off in a synchronized manner are considerably under-studied. The asymmetry in knowledge of the cellular systems is especially obvious for the diametric clusters 6 and 7 (Figure 3d). Cluster 7 includes a number of genes involved in inter-cellular signaling, as well as inflammation, angiogenesis and re-epithelialization, including IL1beta, thrombomodulin, IL8, heparin binding growth factor and ICAM1. These genes are induced shortly after the addition of serum, only to be turned off again after a few hours. The anti-correlated cluster 6 contains 80 genes, which are expressed in the G0 resting state, down-regulated following a short interval after serum addition, only to be expressed again shortly after. These genes include stress response genes, such as heat shock factor 2, and genes inhibitory of cell growth, such as the cdk6 inhibitor. However, of the 80 genes in this cluster, 73 are of entirely unknown function.

Cluster 3 (Figure 3b) includes a number of genes involved in cytoskeletal reorganization, such as the G-protein coupled receptor EDG-1 and desmoplakin, as well as genes such as the GTP-binding protein RAN and the RAN-specific GTPase activating protein. These genes show quite low expression initially, gradually rising in expression levels through the experiment. The anti-correlated cluster 2 shows exactly the opposite pattern: genes expressed high at the beginning of the experiment whose expression levels fall gradually over time. The 57 genes in this cluster include fibrillin, farnesyl diphosphate farnesyltransferase, carnitine palmitoyltransferase, and 46 genes of unknown function.

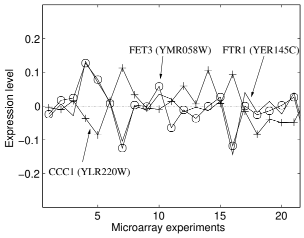
New relationships: A comparison with the clustering of this data by (Iyer *et al.*, 1999) reveals two novel clusters by diametrical clustering. First, cluster 9 (Figure 3e) contains a number of genes related to DNA replication and cell cycle progression, including the G2/M-specific cyclin A and the cyclin dependent kinases regulatory subunit, as well as importin 1, proliferating cell nuclear antigen, centromeric protein E, and ribonucleotide reductase. These genes all show minimal expression in the G0 resting state, but are induced following a considerable time lag after serum addition. The anti-correlated cluster 8 shows a set of genes with the opposite expression pattern, initially expressed in G0, but then turning off with a timing well synchronized to the genes of cluster 9. Of the 9 genes in this cluster, only 4 of known function: apolipoprotein D, complement C1S, lipoprotein lipase, and connective tissue growth factor. Thus, in a fashion coordinated with the reentry into the cell cycle, genes are downregulated for serum lipid transport, fibrogenesis, and complement activation.

A second novel diametric cluster is shown in Figure 3a: Cluster 1 represents those genes showing a transient induction immediately following the addition of serum, such as endothelin 1, interleukin 6, tropomyosin alpha, and the early growth response protein 1. Genes in the anti-correlated cluster 0 show a transient decrease in expression, recovering about 16-20 hours following serum addition. However, unlike the transiently activated genes, of which just less than half are characterized, 26 of the 29 genes in this diametric cluster are of unknown function.

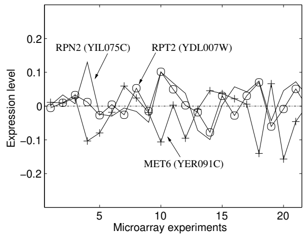
Rosetta yeast: *Known relationships:* We applied diametrical clustering to the Rosetta dataset to produce 40 clusters in Phase I, thus giving a total of 80 anti-correlated clusters in Phase II. Our analysis reveals a number of opposed cellular systems, listed in full at http://www.cs.utexas.edu/users/usman/diametrical. Four pairs of diametric clusters are shown in Figure 4. As an example, the yeast amino acid bio-synthesis genes (CPA2, HIS4, HIS5, LYS1, ARG4, HOM3, etc.) are strongly co-expressed (correlation coefficients > 0.7 over 300 microarray experiments (Hughes *et al.*, 2000) with the SER3 gene, which catalyzes the first committed step in serine synthesis. The CHA1 gene, encoding the serine/threonine deaminase which breaks down serine in the opposed catabolic pathway, shows strongly anti-correlated expression (correlation coefficient = -0.7) with the



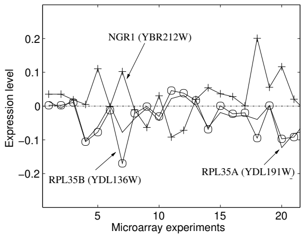
(a) Clusters 4, 5



(b) Clusters 46, 47



(c) Clusters 28, 29



(d) Clusters 8, 9

Figure 4: Clustering the Rosetta data reveals opposing systems of yeast genes. The plots show expression profiles of genes known (a-c) or proposed (d) to work in functionally related, but diametrically expressed, cellular systems.

SER3 gene. So, genes involved in the synthesis of serine show anti-correlated expression with genes involved in the break down of serine (see Figure 4a).

New relationships: In cluster 46 (Figure 4b) we observe that a large number of iron and copper uptake and acquisition genes are co-expressed, including FIT1, FIT2, FIT3, the ferric reductase FRE2, FRE6, the iron permease FTR1, the ferroxidase FET3, the copper transporter CTR2, and the enterobactin transporter ENB1. The anti-correlated cluster 47 contains the CCC1 gene, which is known to transport excess iron from the cytosol to store it in the vacuole (Li *et al.*, 2001). Thus, the systems of iron acquisition and handling of excess iron are in opposition and show diametric expression.

A third example of opposed systems is shown in Figure 4c: a number of proteasomal and vesicular transport genes are co-expressed, including proteasomal proteins alpha 5 and 7, beta 1,3,4,6, and 7, SNX4, RPN 1, 2, 7, 11, and 12, RPT 2, 4, and 6, and the proteasome maturation factor UMP1. The anti-correlated cluster contains genes involved in carbohydrate and amino acid synthesis, including acetate coA ligase, ILV5, MET6, dihydrofolate reductase DFR1. We speculate that the amino acids produced by proteosomal degradation relieve the cell from having to synthesize the amino acids. Therefore, the protein degradation and amino acid synthesis genes can be inversely regulated, as we observe.

As a fourth example (Figure 4d), cluster 8 contains more than 50 ribosomal genes. The anti-correlated cluster contains a set of genes of unknown function, including YJL149W, YNL116W, YNR005C, YMR184W, ECM37, MLF3, YBR016W, YJR120W, YDL172C, YDL053C, YMR140W, YNL140C, YMR141C, YBR273C, as well as BMH2, a homolog of the mammalian 14-3-3 protein which interacts with the proteasome, NGR1, a gene possibly involved in growth regulation, and AAP, a gene which represses translation of the arginine bio-synthetic gene CPA1 in the presence of excess arginine. It is possible that these uncharacterized genes, whose expression patterns oppose that of the ribosome, may represent systems which regulate translation (such as AAP) or protein degradation (such as BMH2).

Performance Comparisons: We now provide some statistics on the stability of diametrical clustering and compare it to other clustering methods.

Implementation and Platform: We implemented our clustering algorithm and *k*-means in C++ using the LEDA library. The convergence parameter δ (see Figure 1) was set to 0.001, and singular vectors were computed by the power iteration using 20 iterations. CLICK (Sharan & Shamir, 2000) was used from the Expander v1.0 version (available from (Sharan & Shamir, 2000)) and GeneShaving (Hastie *et al.*, 2000) from the GeneClust software package (Parmigiani *et al.*, Feb 2003 , In press). All experiments were performed on a 500MHz Pentium PC with 128MB RAM running Debian Linux.

Stability: To measure the stability of the algorithm we computed the standard deviation of the total squared correlation coefficient (see (3)) over 20 runs. Table 1 shows that the standard deviation of the squared correlation coefficient is small compared to its mean value and hence our algorithm is quite stable. The standard deviations of H_{Ave} and S_{Ave} (defined later) values (not shown here) are also small on all the datasets.

DataSet	# Diametric Clusters	Mean	Std. Dev.
Human fibroblasts	5	310.75	0.41
Yeast cell cycle	6	155.84	1.42
Rosetta	25	637.48	1.03

Table 1: The stability of the diametrical clustering algorithm is indicated by the low standard deviation of the total squared correlation coefficient for clusters produced by Phase I of the algorithm.

Comparison with clusters from other algorithms: We compared the actual clusters of our algorithm to those of *k*-means, CLICK, Hierarchical clustering which was obtained from (Iyer *et al.*, 1999), Random clustering, and GeneShaving, as applied to the human fibroblasts dataset. To compare two different clusterings, C and C', we used the Hubert statistic (Jain & Dubes, 1988). Let V be the proximity matrix for C, where $V[i,j] = 1$ iff genes i and j are in the same cluster, and 0 otherwise. Similarly we define W for clustering C'. Then the Hubert statistic is defined as

$$\frac{1}{M}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}(V(i,j)-\mu_v)(W(i,j)-\mu_w)/\sigma_v\sigma_w$$

where m is the total number of genes, $M = \binom{m}{2}$, $\mu_v = \frac{1}{M}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} V(i,j)$, $\sigma_v^2 = \mu_v - \mu_v^2$, and μ_w and σ_w are defined similarly. Intuitively, the Hubert statistic measures how well two sets of clusterings are correlated and ranges from -1 to 1. A value near 1 indicates high correlation, while low values indicate poor correlation between the clusterings.

Table 2 shows the Hubert statistic values for clusterings obtained using various algorithms on the human fibroblast dataset. All the methods were made to produce 10 clusters (10 anti-correlated for diametrical) except for CLICK which produces 5.7 clusters on the average. This table shows that in addition to uncovering anti-correlated structure in the data, the diametrical clustering algorithm reveals clusters similar to those revealed by other algorithms.

One other algorithm, GeneShaving, is theoretically capable of identifying anti-correlated genes. GeneShaving finds a cluster by repeatedly computing the largest principal component of the relevant part of the expression matrix and then shaving off genes with the smallest absolute inner product with this component (Hastie *et al.*, 2000). The next cluster is then found by using a similar strategy after orthogonalizing the expression matrix against the average gene of the previous cluster. As a result, GeneShaving produces overlapping clusters and not every gene is assigned to some cluster which is in contrast to our algorithm. For example with 10 clusters on the human fibroblasts dataset and 6 clusters on the yeast cell cycle data 9% and 73.5% of the genes were left out respectively. This explains the low correlation of all methods with GeneShaving clusterings. Thus, even though both our algorithm and GeneShaving can return anti-correlated clusters, the actual clusterings are not correlated.

	Random	Kmeans	CLICK	Hierarchical	GeneShaving
Diametrical	**-0.001**	**0.531**	**0.446**	**0.439**	**0.105**
Random		-0.001	-.0007	0.025	-0.0005
Kmeans			0.482	0.453	0.083
CLICK				0.4	-0.087
Hierarchical					0.1235

Table 2: A comparison of cluster contents obtained from different algorithms, as measured by the Hubert statistic and averaged over 20 trials, reveals that all methods (except for GeneShaving) yield equally correlated clusterings.

Comparison of cluster quality: We next compare the average coherence and separation of our clusters to those of other algorithms. We evaluate these using H_{Ave} and S_{Ave} measures (Sharan & Shamir, 2000). Let c_i be the normalized centroid (mean) vector of cluster C_i. Then

$$H_{Ave} = \frac{1}{m}\sum_{i=1}^{k}\sum_{g\in C_i} g^T c_i \quad , \quad S_{Ave} = \frac{1}{\sum_{i\neq j}|C_i||C_j|}\sum_{i\neq j}|C_i||C_j|c_i^T c_j .$$

where $|C_i|$ denotes the size of cluster C_i. Intuitively, H_{Ave} measures the average cohesiveness of clusters, while S_{Ave} measures the average separation between clusters. High values of H_{Ave} imply that the clusters have high cohesiveness, while low values of S_{Ave} mean that the clusters are well separated. In general, we desire higher values of H_{Ave} and lower values of S_{Ave}.

On the yeast cell cycle and human fibroblast data, we compared our results to those published in (Sharan *et al.*, 2002) and (Sharan & Shamir, 2000) respectively but used our implementation of *k*-means. On the Rosetta dataset we compared our cluster quality to CLICK using the Expander v1.0 software. All datasets were preprocessed in the same manner in the studies we compared against, and for diametrical, *k*-means and CLICK we averaged the results over 20 runs.

Table 3 shows that diametrical clustering produces clusters of comparable quality to the other algorithms. Note that the numbers in Table 3 for diametrical clustering are for the Phase II (anti-correlated) clusters.

Dataset	Algorithm	#Clusters	H_{Ave}	S_{Ave}
Yeast cell	**Diametrical**	**6**	**0.6**	**-0.1**
	CLICK	6	0.66	-0.1
	Kmeans	6	0.6	-0.06
	GeneCluster (SOM)	6	0.62	-0.07
	CAST	5	0.6	-0.15
Human fibroblast	**Diametrical**	**10**	**0.88**	**-0.09**
	CLICK	10	0.88	-0.34
	Kmeans	10	0.88	-0.12
	Hierarchical	10	0.87	-0.13
Rosetta yeast	**Diametrical**	**50**	**0.56**	**-0.02**
	CLICK (Expander)	50	0.52	-0.027
	Kmeans	52	0.57	-0.018

Table 3: Clustering quality indicates that diametrical clustering compares favorably with other algorithms. Note that our algorithm does not explicitly try to optimize these values, instead focusing on finding diametric gene clusters.

Comparison of running time: We finally provide a comparison of running times in Table 4 averaged over 20 trials. The GeneShaving implementation was only available on S-plus software, so we did not include its running time numbers.

We give results for the closest number of clusters produced by CLICK. Even though we have a simple implementation of our algorithm in C++, Table 4 shows that the running time is still acceptable for large datasets. In future work, we intend to optimize the speed of our implementation.

Dataset	CLICK	**Diametrical**	Kmeans
Human fibroblast	128.72 (5.7)	**1.5 (6)**	0.42 (6)
Yeast cell cycle	72.5 (10.5)	**10.38 (12)**	5.4 (12)
Rosetta	402.27 (50)	**858.15 (50)**	836.74(50)

Table 4: A comparison of running times (in seconds) averaged over 20 trials reveals that diametrical clustering is computationally efficient. The average number of clusters created by each algorithm is indicated in parentheses.

4 Conclusions and Future Work

Using our diametrical clustering algorithm, we discover systems opposing the yeast ribosome and proteasome, we demonstrate the opposing mRNA expression profiles of amino acid synthetic and degredative systems, as well as of iron acquisition and excess iron storage systems. Finally, we demonstrate that human fibroblast genes downregulated following serum stimulation are systematically understudied, suggesting that diametrical clustering should be widely applicable for bringing to light similarly non-obvious relationships between cellular systems.

A number of improvements to our analysis are apparent. Foremost, there are problems with k-means like strategies — for example, empty clusters, initialization strategies, the need to specify the number of clusters, etc., which could be improved. Another interesting point to note is that our diametric clustering algorithm proceeds by clustering together gene vectors according to their closeness to the lines described by the singular vectors. These lines are 1-dimensional objects — on the other hand, traditional clustering algorithms like k-means cluster vectors based on their proximity to points, which are 0-dimensional objects. Our algorithm could be modified to look for closeness to higher dimensional objects, which might suggest linear dependencies between clusters and may give even more insight into the organization and regulation of genes. Also, we could use some filtration techniques to separate and identify outliers in the dataset.

The correlation coefficient is a popular measure for measuring similarity between genes and its use is experimentally validated by Figure 2. The squared correlation coefficient restricts us to examining quadratic relationships in the data. It is likely that other non-linear relationships will exist, especially in time series and cell cycle data. Gene expression relationships may be heteroscedastic in nature. An interesting extension of our methodology would be to explore non-linear relationships, possibly using kernel methods (Schölkopf *et al.*, 1998).

Finally, it would be very interesting to look for conserved regulatory motifs upstream of the genes in diametrical clusters. It is not immediately apparent if the genes would be expected to share common motifs, but as they seem to be responding to common stimuli, albeit in opposite directions, it is not unreasonable to expect to find common control elements, possibly even those responsible for the general response, while elements responsible for the specific direction of response might be found in the separated clusters.

5 Acknowledgments

We would like to thank Vishwanath Iyer for helpful discussion and Usman Shakil for helping with the web page. This work was supported by a grant from the Welch Foundation (EMM), a Dreyfus New Faculty Award (EMM), the Texas Advanced Research Program (EMM and ISD), the NSF (EMM), and NSF Career Award Grant No. ACI-0093404 (ISD).

6 Appendix

Lemma 1 *(Golub & Loan, 1996) Suppose $g_1, g_2, \ldots, g_m$ are n-dimensional real vectors that form the rows of the $m \times n$ matrix G. Then the unit vector x that maximizes $f(x) = x^T\left(\sum_i g_i g_i^T\right)x = x^T G^T G x$ is the dominant right singular vector v_1 of G (or equivalently, the dominant eigenvector of $G^T G$). The optimal value equals $f(v_1) = v_1^T\left(\sum_i g_i g_i^T\right)v_1 = \sigma_1^2$, where σ_1 is the largest singular value of G and $\sigma_1 > \sigma_2$.*

Theorem 1 *Phase 1 of Algorithm* Diametrical_Clustering *given in Figure 1 never decreases the quality measure $Q(C_1, \ldots, C_k) = \sum_{j=1}^k \sum_{g \in C_j} (g^T v_j)^2$ from one iteration to the next.*

Proof. Let $C_1^{(t)}, \ldots, C_k^{(t)}$ be the clusters at iteration t, and let $v_1^{(t)}, \ldots, v_k^{(t)}$ be the corresponding singular vectors. Then

$$\begin{aligned} Q(C_1^{(t)}, \ldots, C_k^{(t)}) &= \sum_{j=1}^{k} \sum_{g \in C_j^{(t)}} (g^T v_j^{(t)})^2 \\ &\leq \sum_{j=1}^{k} \sum_{g \in C_j^{(t)}} (g^T v_{j^*(g)}^{(t)})^2 \\ &\leq \sum_{j=1}^{k} \sum_{g \in C_j^{(t+1)}} (g^T v_j^{(t+1)})^2 \\ &= Q(C_1^{(t+1)}, \ldots, C_k^{(t+1)}) \end{aligned}$$

where the first inequality is due to step 2 of the algorithm (see Figure 1), and the second inequality follows from Lemma 1.

References

Bairoch, A. & Apweiler, R. (2000). The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. *Nucleic Acids Research*, **1;28(1)**, 45–48.

Brown, M. P. S., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C., Furey, T. S., M. Ares, J. & Haussler, D. (2000). Knowledge-based analysis of microarray gene expression data using support vector machines. *Proc. Natl. Acad. Sci.*, **97**, 262–267.

DeRisi, J. L., Iyer, V. R. & Brown, P. O. (1997). Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, **24;278(5338)**, 680–686.

Dhillon, I. S. & Modha, D. S. (2001). Concept decompositions for large sparse text data using clustering. *Machine Learning*, **42**, 143–175.

Eisen, M. B., Spellman, P. T., Brown, P. O. & Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci.*, **95**, 14863–14868.

Golub, G. H. & Loan, C. F. V. (1996). Matrix computations. Johns Hopkins Studies in the Mathematical Sciences. The Johns Hopkins University Press, Baltimore, MD, USA, third edition.

Graybill, F. A. & Iyer, H. K. (1994). Regression analysis: concepts and applications. Duxbury Press.

Hastie, T., Tibshirani, R., Eisen, M. B., Alizadeh, A., Levy, R., Staudt, I., Chan, W. C., Botstein, D. & Brown, P. (2000). 'Gene shaving' as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biology*, **1**, 1–21.

Herrero, J., Valencia, A. & Dopazo, J. (2001). A hierarchical unsupervised growing neural network for clustering gene expression patterns. *Bioinformatics*, **17**, 126–136.

Hughes, T., Marton, M., Jones, A., Roberts, C., Stoughton, R., Armour, C., Bennett, H., Coffey, E., Dai, H., He, Y., Kidd, M., King, A., Meyer, M., Slade, D., Lum, P., Stepaniants, S., Shoemaker, D., Gachotte, D., Chakraburtty, K., Simon, J., Bard, M. & Friend, S. (2000). Functional discovery via a compendium of expression profiles. *Cell*, **102**, 109–126.

Iyer, V., Eisen, M., Ross, D., Schuler, G., Moore, T., Lee, J., Trent, J., Staudt, L., Hudson, J., Boguski, M., Lashkari, D., Shalon, D., Botstein, D. & Brown, P. (1999). The transcriptional program in the response of human fibroblasts to serum. *Science*, **283**, 83–87.

Jain, A. K. & Dubes, R. C. (1988). Algorithms for Clustering Data. Prentice Hall, Englewood Cliffs.

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

Li, L., Chen, O., Ward, D. M. & Kaplan, J. (2001). CCC1 is a transporter that mediates vacuolar iron storage in yeast. *Journal of Biological Chemistry*, **276(31)**, 29515–29519.

Lukashin, A. V. & Fuchs, R. (2001). Analysis of temporal gene expression profiles: clustering by simulated annealing and determining the optimal number of clusters. *Bioinformatics*, **17**, 405–414.

Marcotte, E. M., Pellegrini, M., Thompson, M. J., Yeates, T. & Eisenberg, D. (1999). A combined algorithm for genome-wide prediction of protein function. *Nature*, **402**, 83–86.

Parmigiani, G., Garett, E. S., Irizarry, R. A. & Zeger, S. L., (eds.) (Feb 2003 (In press)). The analysis of gene expression data: Methods and software. Springer.

Qian, J., Dolled-Filhart, M., Lin, J., Yu, H. & Gerstein, M. (2001). Beyond synexpression relationships: local clustering of time-shifted and inverted gene expression profiles identifies new, biologically relevant interactions. *Journal of Molecular Biology*, **314**, 1053–1066.

Schölkopf, B., Smola, A. & Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, **10**, 1299–1319.

Sharan, R., Elkon, R. & Shamir, R. (2002). Cluster analysis and its applications to gene expression data. In *Ernst Schering Workshop on Bioinformatics and Genome Analysis*. Springer Verlag, pp. 83–108.

Sharan, R. & Shamir, R. (2000). CLICK: A clustering algorithm with applications to gene expression analysis. In *Proceedings of 8th International Conference on Intelligent Systems for Molecular Biology (ISMB)*. AAAI Press, pp. 307–316.

Shatkay, H., Edwards, S., Wilbur, W. J. & Boguski, M. (2000). Genes, themes, and microarray: using information retrieval for large-scale gene analysis. In *Proceedings of 8th International Conference on Intelligent Systems for Molecular Biology (ISMB)*. AAAI Press, pp. 317–328.

Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E. S. & Golub, T. (1999). Interpreting patterns of gene expression with self organizing maps. *Proc. Natl. Acad. Sci.*, **96**, 2907–2912.

Verjovsky Marcotte, C. J. & Marcotte, E. M. (2002). Predicting functional linkages from gene fusions with confidence. *Applied Bioinformatics*, **2(1)**, 93–100.