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 define cellular pathways and 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: inderjit@cs.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 (Lashkari et al., 1997); current generation microarrays typically measure expression of every gene encoded by a genome. From sets of DNA microarray experiments, an expression vector for each gene can be constructed, where the vector describes the expression of a given gene under a range of cellular conditions, cell types, genetic backgrounds, etc. Analysis of such data can greatly help in understanding and predicting functions of genes, many of which have been sequenced but are as yet of unknown function.

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), *k*-means (Tavazoie et al., 1999; Herwig et al., 1999), self-organizing maps (Tamayo et al., 1999), algorithms based on principal components analysis (Hastie et al., 2000) and graph-based algorithms (Sharan and 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). 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 SER3 gene. So, genes involved in the synthesis of serine show anti-correlated expression with genes involved in the breakdown of serine. A second category of genes we might expect to show anti-correlated expression patterns are genes which act to repress the expression of other genes. Again, we expect that these genes will be generally involved in the same biological pathway, but will show anti-correlated expression patterns.

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 function similarity between such opposed clusters.

In order to achieve this goal, we propose a new clustering algorithm which puts strongly correlated *and* anti-correlated

[*] 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

genes into the same “diametric” cluster. A simple post-processing step can then separate the positively correlated genes from the ones that are negatively correlated. Our clustering algorithm bears some resemblance to the k-means procedure (Duda et al., 2000), 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 breakdown for our goal since each cluster contains positively and negatively-correlated genes. In our diametrical clustering algorithm, each cluster’s prototype turns out to be the dominant singular vector of the matrix whose rows comprise the cluster members. This strategy proves to be successful in identifying diametric clusters. More details are given in the Algorithm section.

We now give a brief outline of the paper. First we discuss some similarity measures used in clustering after which we introduce our algorithm to detect anti-correlated clusters. In the experimental part of the paper, we apply the method to three sets of mRNA expression data and present results from the analyses. Finally, we present conclusions and future work.

A word about notation: small letters such as g, h, x and v will denote vectors, capital letters such as A, G denote matrices. Also, $||g||$ denotes the L^2 norm of vector g while $g^T h$ denotes the usual inner product between vectors.

2 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 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, $i = 1,\dots,n$. When μ_g is set to 0, then $S(g,h)$ equals the cosine of the angle between the vectors g and h.

By shifting each gene vector by its mean and then normalizing it to have unit norm, the Pearson correlation coefficient is seen to simply equal the inner product between the (transformed) gene vectors. More precisely, by making the transformations

$$\tilde{g}_i \;=\; \frac{g_i-\mu_g}{\sum_{j=1}^{n}(g_j-\mu_g)^2}, \quad 1 \le i \le n,$$

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 and Shamir, 2000) and (Brown et al., 2000). Note that each transformed gene vector g resides on the unit (hyper)sphere in n-dimensional space.

3 Algorithm

Our goal is to find clusters containing genes that are either highly positively correlated or highly negatively correlated. Hence, an obvious similarity measure to use is the square of the correlation coefficient, i.e.,

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

where g and h are gene vectors with mean 0 and norm 1. Clearly this measure is high (close to 1) if the genes have high positive or negative correlation.

Having defined a similarity measure, we need an appropriate clustering algorithm. Two choices are to either use a hierarchical clustering algorithm or a graph partitioning approach. However, we reject these choices since the complexity of these algorithms is at least quadratic in the number of genes. We want to be able to process all yeast genes ($\approx 6,000$) and all human genes ($\approx 35,000$) and so it would be desirable for our clustering algorithm to scale linearly with the number of genes.

The popular k-means algorithm is efficient; however it is not suitable for our measure of similarity. Given a cluster which contains genes that have high positive as well as negative correlation, it would be incorrect to compute the cluster centroid (or mean) as the “cluster prototype” as is done in the traditional k-means algorithm. Thus we need a different definition of “cluster prototype” that is suitable for 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). The mathematical formulation is to find a unit vector x_j such that the sum

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

```
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 vectors $v_1, \ldots, v_k$ of each cluster sub-
     matrix $G_1, \ldots, G_k$ respectively.

  2. Re-compute all clusters: for each gene $g$ find its new clus-
     ter index as
         $j^*(g) = \text{argmax}_i (g^T v_i)^2$,
     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 $G_1, \ldots, G_k$ re-
     spectively.

  4. If "converged" go to Phase II, else go to step 2 above.

Phase II:

  1. For each diametric cluster $C_i$ output the 2 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 clus-
     ter "fingerprints".
```

Figure 1: Algorithm for diametrical clustering

is maximized. Using linear algebra, it is well-known that 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 and Loan, 1996). For the sake of completeness, we give a proof in the Appendix. Thus, given a clustering $C_1, C_2, \ldots, C_k$ we can measure its quality by

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

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.

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 and Loan, 1996). Our diametrical algorithm has the pleasing property that 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. For more details, see (Dhillon and Modha, 2001) and (Selim and Ismail, 1984).

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 our algorithm *does not force* a diametric or anti-correlated structure on the data. Indeed, if the data set does not have anti-correlated clusters then one of the clusters found in Phase II will be empty.

The time taken by the algorithm is $O(mnk\tau)$ where τ is the number of iterations required — experimental results show that 15-20 iterations are typical. Detailed analysis and timing results are given in Section 4.

An 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.

4 Experimental Results

4.1 Datasets

Human Fibroblasts: First, we analyzed the human fibroblast data set of (Iyer et al., 1999), which reports the response of human fibroblasts following the addition of serum to the growth media. This data set (available from genome-www.stanford.edu/serum) contains the expression levels of 8,613 human genes which were obtained by depriving human fibroblasts of serum for 48 hours and then stimulating them by the addition of serum. Expression levels were measured at 12 time points after the stimulation, and an additional data-point was obtained from a separate unsynchronized sample. We analyzed the subset of 517 genes reported in (Iyer et al., 1999) whose expression levels changed substantially across the samples. The data was preprocessed by dividing each entry by the expression level at time zero, taking the log of the result, and then normalizing each 12-element expression vector to have unit L^2 norm.

Yeast Cell Cycle: Next, we analyzed the set of gene expression data measured from synchronized yeast cultures through several phases of the cell cycle (http://cellcyclewww.stanford.edu; (Spellman et al., 1998)). This data set contains data from yeast cultures synchronized by four independent methods: α factor based (samples taken every 7 minutes over 119 minutes), arrest of a cdc15 temperature sensitive mutant (samples taken every 10 minutes over 290 minutes), arrest of a cdc28 temperature sensitive mutant taken from (Cho et al., 1998, Section 3.1), and elutriation data (30 minute samples taken over 6.5 hours). In addition it contains experiments in which G1 cyclin Cln3p and B-type cyclin Clb2 were induced. Spellman et al. (1998) identified 800 genes that are cell cycle regulated, out of which we used a subset of 696 genes which have at most four missing values. The data was normalized to have mean 0 and norm 1.

Rosetta yeast: Lastly, we analyzed the Rosetta Inpharmatics yeast data set of (Hughes et al., 2000). This data consists 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. 276 deletion mutants, 11 tetracycline-regulatable alleles of essential genes, and 13 well-characterized compounds were profiled. 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.

4.2 Validation of diametrical clusters

We first present the diametrical clusters obtained for the yeast cell cycle dataset (Spellman et al., 1998), where we see how our algorithm separates genes with opposing expression profiles, and which also tend to peak in diametrically opposite phases of the cell cycle. Secondly we provide evidence on the Rosetta (Hughes et al., 2000) dataset that anti-correlated genes are functionally related.

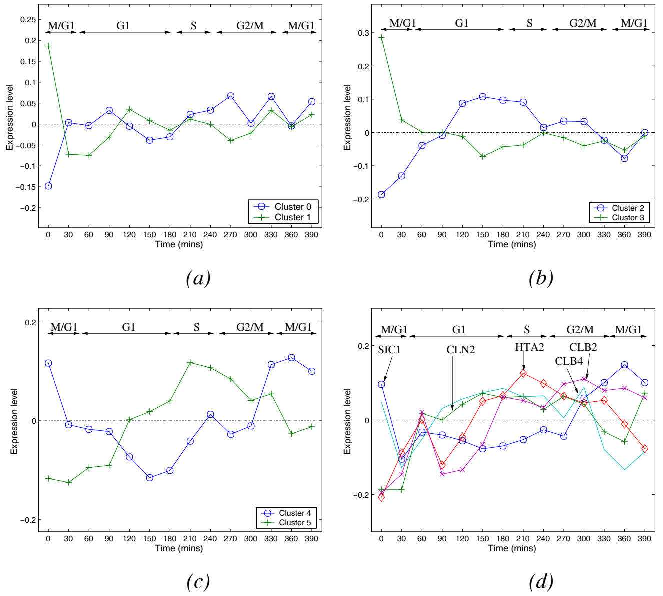
4.2.1 Yeast Cell Cycle

We applied our clustering algorithm on this dataset to produce 12 clusters. Our clustering algorithm clusters the genes based on all the experiments performed in this dataset. However, for ease of analysis and better visual representation we present the results on just the elutriation (30 minute samples taken over 6.5 hours) experiments.

We observe that the diametric clusters represent genes with opposed expression patterns. As genes in this data set all show cyclic expression changes as the cell cycle progresses, the diametric clusters also tend to contain genes whose expressionn levels peak in opposed times in the cell cycle, as plotted in Figure 2. The rest of the clusters (not plotted) from this dataset show similar behaviour.

4.2.2 Relationship between correlation coefficients and functional annotation

To evaluate if anti-correlated genes shared some degree of functional relatedness, we took all yeast genes with functional annotation in the KEGG database (Kanehisa and Goto, 2000), then measured the correlation coefficients between the expression vectors of all pairs of the annotated yeast genes. For each gene pair, we then represented each gene's function with a set containing KEGG keywords, which allowed us to compute the Jaccard coefficients between the gene's KEGG categories (Marcotte and Marcotte, 2002). The Jaccard coefficients of two sets A and B is defined as $\frac{|A \cap B|}{|A \cup B|}$.



(a) *(b)* *(c)* *(d)*

Figure 2: Expression profiles of the mean of the clusters obtained on the yeast cell cycle dataset (only elutriation experiments shown). (a), (b), and (c) each display the mean vectors of the two opposed clusters obtained from Phase II of the algorithm. The phases of the cell cycle are plotted in (d), indicated with the expression profiles of 5 genes with phase-specific expression.

In Figure 3, we have plotted the functional similarity (mean Jaccard coefficient of the KEGG categories) versus the correlation coefficient of the expression vectors. As expected, genes with co-expression (high positive correlation coefficients) show strong functional similarity. However, genes with anti-correlated expression (high negative correlation coefficients) also show functional similarity, validating the search for anti-correlated gene expression clusters.

4.3 Analysis of diametrical clusters

4.3.1 Human Fibroblasts

We applied our algorithm to obtain 5 diametric clusters in Phase I which were separated into 10 clusters in Phase II. We chose 10 clusters so that we could compare our results to previously published results on this dataset. An examination of the expression profiles of the centroid of each cluster, plotted in Figure 4, shows that the diametrical clustering algorithm

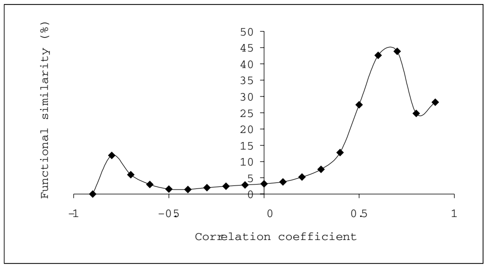
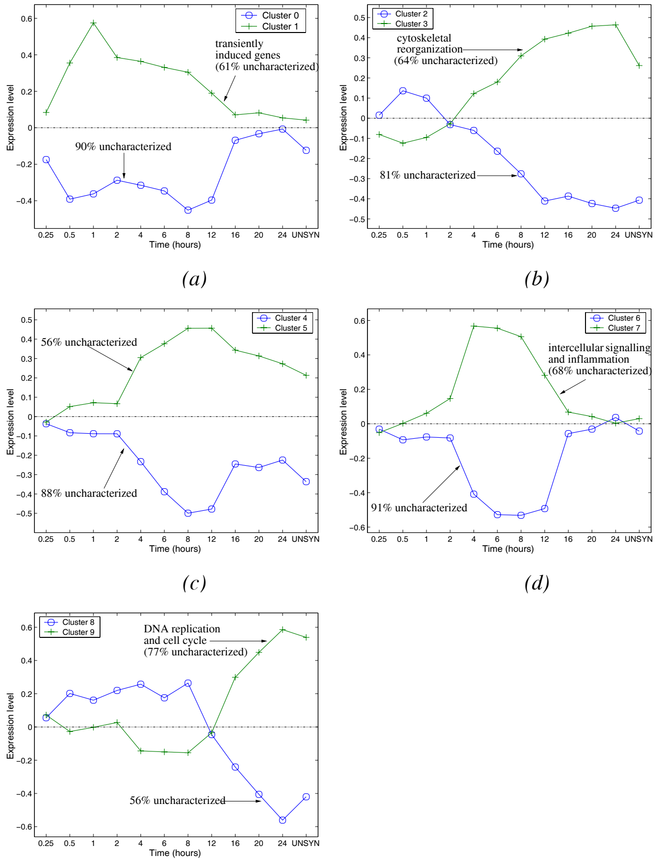


Figure 3: Genes with both highly correlated and highly anticorrelated mRNA expression patterns tend to operate in similar cellular pathways.

nicely identifies genes with opposed expression patterns.



(a)

(b)

(c)

(d)

(e)

Figure 4: Expression profiles of the mean of the clusters obtained on the human fibroblast dataset. Each figure contains the means of the two clusters obtained from Phase II of the algorithm, and identifies opposing systems. Systems which turn off in reponse to serum stimulation can be seen to be systematically understudied.

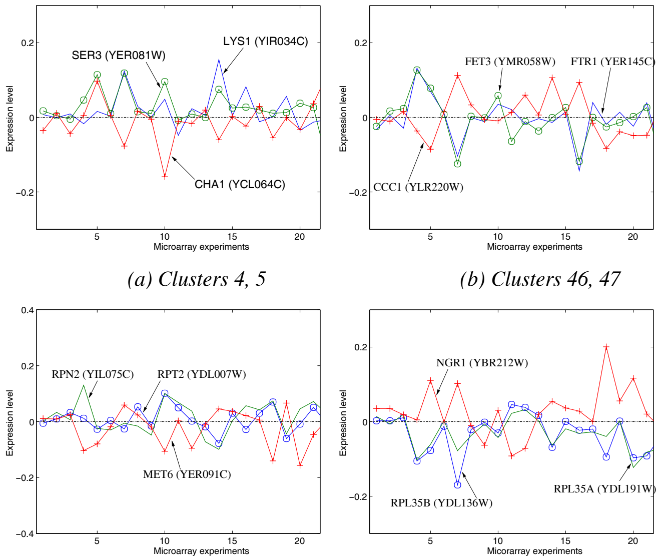
Known relationships: In general, we find the systems induced by serum addition are partly characterized, but the systems turned off in a synchronized manner are considerably under-studied. The asymmetry in knowledge of the cellular systems is especially obvious for the diametric clusters 6 and 7 (Figure 4d). 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, ICAM1, monocyte chemotactic protein 1, and heparin growth factor 2. These genes are induced shortly after the addition of serum, only to be turned off again after a few hours.

The diametric 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, the genes in this cluster are remarkably poorly studied, and of the 80 genes in this cluster, 73 are of entirely unknown function.

Cluster 3 (Figure 4b) 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 metallothionein, 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 course of the experiment. The diametric 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: Analyzing this data for diametrical clusters reveals two clusters whose means are different from those in (Iyer et al., 1999). First, cluster 9 (Figure 4e) 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 genes such 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 diametric 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. In this cluster are 9 genes, only 4 of known function: apolipoprotein D, complement C1S, lipoprotein lipase, and connective tissue growth factor. Thus, it would appear that 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 4a: 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 diametric cluster



(a) Clusters 4, 5

(b) Clusters 46, 47

(c) Clusters 28, 29

(d) Clusters 8, 9

Figure 5: Expression profiles of specific genes from some of the diametric clusters on the Rosetta yeast dataset. The clusters show genes known (a-c) or proposed (d) to work in functionally related, but opposing, cellular systems, whose expression profiles show inverse relationships.

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.

4.3.2 Rosetta yeast

Known relationships: We applied our clustering algorithm to this dataset to produce 40 diametric clusters, thus giving a total of 80 clusters. 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 5. For example, the amino acid synthesis genes mentioned in the introduction cluster together, with the opposed serine catabolism gene CHA1 occurring in the diametric cluster (see Figure 5a).

New relationships: In cluster 46 (Figure 5b) 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 diametric cluster 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 5c: 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 diametric 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 5d), cluster 8 contains more than 50 ribosomal genes. The diametric 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).

4.4 Comparison to other methods

In this section we compare the diametrical clustering to other clustering methods. We evaluate the quality of the clustering using H_{Ave} and S_{Ave} measures (Sharan and Shamir, 2000). Let c_i be the normalized centroid (mean) vector of cluster C_i. Then

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

Intuitively, H_{Ave} measures the average cohesiveness of clusters while S_{Ave} measures the average separation between clusters. In general, we desire higher values of H_{Ave} and lower values of S_{Ave}. We first present results comparing the H_{Ave} and S_{Ave} to other methods followed by a comparison of running times.

4.4.1 Implementation and Platform

We implemented the diametrical clustering algorithm in C++ using the LEDA library, and used the Expander v1.0 software (obtained from Roded Sharan) to conduct the experiments on CLICK (Sharan and Shamir, 2000). We used a 600 MHz Pentium machine running Debian Linux to run our experiments.

4.4.2 Comparison of H_{Ave} and S_{Ave}

All datasets were preprocessed in the same manner as in the studies we compared against. We first applied the diametrical clustering algorithm on the yeast cell cycle data (Spellman et al., 1998) to obtain 6 clusters and compared our results to ones published in (Sharan et al., 2002). We then applied our algorithm on the human fibroblast dataset (Iyer et al., 1999) and compared our results to those published in (Sharan and Shamir, 2000). Finally we ran CLICK on the Rosetta yeast dataset using the Expander v1.0 software and compared the results to our algorithm.

Program	#Clusters	H_{Ave}	S_{Ave}
Yeast cell cycle			
Diametrical	6	0.6	-0.13
CLICK	6	0.66	-0.1
K-Means	49	0.63	0.09
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
Hierarchical	10	0.87	-0.13
Rosetta yeast			
Diametrical	60	0.57	-0.02
CLICK (Expander)	59	0.55	-0.03

Table 1: Comparison of H_{Ave} and S_{Ave} of various methods on all the datasets

The results in Table 1 show that the H_{Ave} and S_{Ave} values produced by our algorithm are quite good and compare favorably with other methods. Note that our algorithm does not explicitly try to optimize these values; instead its focus is on finding opposed gene clusters.

4.4.3 Comparison of running time

We provide a running time comparison of our method to CLICK. Since our algorithm only produces an even number of clusters, we try to produce the closest number of clusters produced by CLICK. Even though we have a naive implementation of our algorithm in C++ the running time is still acceptable for large datasets. In future work, we will optimize the speed of our implementation.

Dataset	CLICK	Diametrical
Human fibroblast	88.28 (5)	1.58 (6)
Yeast cell cycle	60.75 (12)	6.55 (12)
Rosetta	401.67 (59)	663.02 (60)

Table 2: Comparison of running times (in seconds) of our algorithm against CLICK on all the datasets. Next to the time, we also show the number of clusters created by each method.

5 Conclusions and Future Work

In conclusion, we have explicitly searched for genes with opposite patterns of gene expression. To do this efficiently, we have introduced a diametrical clustering algorithm, which identifies pairs of gene clusters, each cluster with an expression profile opposite that of the other. We show that genes with anti-correlated expression patterns are often functionally related, and often encode systems with related, but opposite, functions in the cell. Using this algorithm we discover systems opposing the yeast ribosome and proteasome, we demonstrate the opposition of expression of amino acid "synthesis and degradation" system and of iron acquisition and storage systems, and we show that genes turning off following serum stimulation of fibroblasts are systematically under-studied.

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. Second, in the algorithm we describe, we have detected diametrical clusters by looking at closeness to one-dimensional objects, i.e. lines. In general, we can 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. 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.

6 Acknowledgments

We would like to thank Vishwanath Iyer for helpful discussion. 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), and an NSF Career Award Grant No. ACI-0093404 (ISD).

7 Appendix

Lemma 1 *(Golub and 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 $\sigma_1 > \sigma_2$.

Proof. Let x be an arbitrary unit vector and express it as $x = \sum_i \alpha_i v_i$, where $\sum_i \alpha_i^2 = 1$ and v_i's are the (orthonormal) right singular vectors of G. Since $G^T G v_i = \sigma_i^2 v_i$,

$$f(x) = x^T G^T G x = \sum_i \alpha_i^2 \sigma_i^2.$$

The above quantity is maximized when $\alpha_1 = 1$ and all other α_i's are 0, Hence, the optimal x equals v_1 and the maximum value attained equals $v_1 G^T G v_1 = \sigma_1^2$.

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

Brown, M. P. S., Grundy, W. N., Lin, D., Cristianini, N., Sugnet, C., Furey, T. S., M. Ares, J., and Haussler, D. (2000). Knowledge-based analysis of microarray gene expression data using support vector machines. *Proc. Natl. Acad. Sci.*, 97(1):262–267.

Cho, R., Campbell, M., Winzeler, E., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T., Gabrielian, A., Landsman, D., Lockhart, D., and Davis, R. (1998). A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular Cell*, 2:65–73.

Dhillon, I. S. and Modha, D. S. (2001). Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42(1):143–175.

Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification*. John Wiley & Sons, 2nd edition.

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

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

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

Herwig, R., Poutska, A. J., Mueler, C., Lehrach, H., and Brien, J. O. (1999). Large scale clustering of cDNA-fingerprinting data. *Genome Research*, 9(11):1093–1105.

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., and 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., and Brown, P. (1999). The transcriptional program in the response of human fibroblasts to serum. *Science*, 283(1):83–87.

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

Lashkari, D. A., Risi, J. L. D., McCusker, J. H., Namath, A. F., Gentile, C., Hwang, S. Y., Brown, P. O., and Davis, R. W. (1997). Yeast microarrays for genome wide parallel genetic and gene expression analysis. *Proc. Natl. Acad. Sci.*, 94:13057–13062.

Li, L., Chen, O., Ward, D. M., and Kaplan, J. (2001). Ccc1 is a transporter that mediates vacuolar iron storage in yeast. *Journal of Biological Chemistry*, 276(31):29515–29519.

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

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

Qian, J., Dolled-Filhart, M., Lin, J., Yu, H., and 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.

Selim, S. Z. and Ismail, M. A. (1984). K-means-type algorithms: a generalized convergence theorem and characterization of local optimality. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 6:81–87.

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

Sharan, R. and 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)*, pages 307–316. AAAI Press.

Shatkay, H., Edwards, S., Wilbur, W. J., and 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)*, pages 317–328. AAAI Press.

Spellman, P. T., Sherlock, G., Zhang, M., Iyer, V. R., Anders, K., Eisen, M., Brown, P. O., Botstein, D., and Futcher, B. (1998). Comprehensive identification of cell cycle regulated gene of the yeast Saccharomyces Cerevisia by microarray hybridization. *Mol. Bio. Cell*, 9:3273–3297.

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

Tavazoie, S., Hughes, J., Campbell, M., Cho, R., and Church, G. (1999). Systematic determination of genetic network architecture. *Nature Genetics*, 22:281–285.