Andrew Post

CS 2750
Project Report
April 5, 2003

Abstract
Microarray technology makes it possible to monitor simultaneously the expression patterns of thousands of genes during cell differentiation and response. Measurements of gene expression levels over time can provide valuable insight into protein function.In this report is described a two-step technique of clustering followed by learning Bayesian networks to: (1) cluster genes with similar expression patterns and characterize the functional characteristics of those clusters; and (2) prioritize genes within a cluster by importance in carrying out that cluster's function(s).

Background
All cells in an organism contain the same genomic data, but their protein makeup can differ dramatically both temporally and spatially. The process of translating genomic data into proteins is called gene expression. Analysis of gene expression patterns is a valuable tool for making inferences about gene function and biological pathways.

In the past several years, microarray technology has made it possible to monitor simultaneously the expression patterns of thousands of genes within a single organism during cell differentiation and response[1]. The output of a microarray experiment is a matrix in which rows are data for a gene and columns are observations during different conditions. Conditions can be consecutive time steps, thus making each row a time series. Typical experiments have anywhere from a dozen to over one-hundred observations for each gene. Each observation is a ratio of expression level versus control. Machine learning techniques are necessary to make sense of these massive datasets.

Analysis of expression patterns has traditionally been done with clustering algorithms. It is assumed that genes whose proteins work together to carry out a function have related expression patterns. Genes with related expression patterns may have similar (gene i = gene j), proportional (gene i α gene j), or opposite (gene i -α gene j) expression profiles. For example, one gene's protein may serve to increase or decrease expression of another gene. Techniques that are currently in use include: plotting expression time series and finding patterns by visual inspection[2]; hierarchical clustering techniques[3,4]; self-organizing maps (SOMs)[5]; and k-means[6]. If a cluster is enriched in genes with a particular function, it may be hypothesized that uncharacterized genes in that cluster also participate in the same function.

The primary drawbacks of clustering algorithms in analyzing gene expression data are: (1) different distance metrics produce different clusters; and (2) clusters have no internal structure without doing further analysis. A biological relevant distance metric is still the topic of much debate. Choice of metric is complicated by the noisy nature of gene expression data. Complete pairwise correlations have been used in many studies. Their advantage is that they can detect pairs of genes with similar, proportional, or opposite expression profiles. However, correlations are sensitive to the choice of an arbitrary threshold and to noise. Euclidean distance tends to be

less sensitive to noise: two genes that exhibit poor pairwise correlation may still appear close by virtue of their correlation patterns with other genes.

More recently, attempts have been made to use expression data to uncover the structure of cellular processes. Friedman et al. has developed a method of Bayesian clustering[7] that has been used to identify genes whose proteins may be particularly important in carrying out a cellular function. It makes use of a heuristic called the Sparse Candidate algorithm[8]. It reduces the search space by identifying the best candidate parents for each node based on a statistical measure, and then restricting the search to networks in which only candidate variables can be parents. Dependencies are learned between genes that have related time series. Importance of these dependencies is ranked by order relations and Markov relations. Order relations rank genes by how often they appear as ancestors of other genes. Markov relations describe pairs of genes that often have arcs between them.

Unfortunately, Friedman's Bayesian clustering method is not fast enough to use with typical data sets containing thousands of genes. It would also be difficult to interpret the large and complex networks that would result. Therefore, I propose a two-step process that: (1) uses traditional clustering techniques to identify small groups of genes that may participate in a single cellular process; and (2) uses Friedman's Bayesian clustering technique to identify genes that are particularly important in carrying out that function.

Methods

Data Set:

A publicly available dataset was used that is described in Cho et al[2] and is available from many sources, including the GeneSpring software demo package (http://www.sigenetics.com). Briefly, yeast cells were synchronized at the beginning of the cell cycle, and expression levels of 6457 genes were monitored every 10 minutes for 160 minutes (about 2 cell cycles). The 90 minute expression levels were thrown out due to irregularities in expression monitoring. Duplicate observations were averaged.

Expression levels were normalized by dividing each value by the median for that gene. All genes whose expression levels had a variance of less than 1.2 were thrown out, thus reducing the data set to 3029 genes with 16 observations each. Slopes of pairwise consecutive observations were added to account for offset but parallel patterns as described in Wen et al.[3], thus resulting in a final dataset with 31 observations per gene. Code for performing these preprocessing steps can be found in Appendix I.

Clustering:

K-means clustering was carried out with the SOM Matlab toolbox (http://www.cis.hut.fi/projects/somtoolbox/). Clustering under a range of target clusters from 2-50 was performed. Sum squared error and the Davies-Bouldin index were calculated for each target cluster size. The Davies-Bouldin index is a ratio of the sum squared error within clusters to the sum squared error between clusters. The best clusterings are those that minimize the Davies-Bouldin index (i.e. those that minimize the sum square error within cluster and maximize the sum squared error between clusters).

A technique described by Tavazoie et al.[6] was then used to find clusters that are significantly enriched for genes with similar functions. Briefly, the genes in each cluster are mapped to the 259 functional categories in the Munich Information Center for Protein Sequences (MIPS) Comprehensive Yeast Genome Database (CYGD) (http://mips.gsf.de/proj/yeast/CYGD/db/index.html). The hypergeometric distribution is then used obtain the statistical significance of that mapping. For this experiment, the mapping was done using SequenceUpToDate, a software package previously developed by the author for automated querying of online bioinformatics databases via HTTP. A module had to be written to support queries to the MIPS database (see Appendix II). For each cluster, P values were calculated for observing the frequencies of genes in particular functional categories. For calculating P, the hypergeometric distribution was used to obtain the probability of observing at least *k* genes from a functional category within a cluster of size *n*. The formula is given by:

$$P = 1 - \sum_{i=0}^{k} \frac{\binom{n}{i}\binom{g-n}{f-i}}{\binom{g}{f}}$$

where *f* is the number of functional categories in the genome, and *g* is the size of the genome. As 259 MIPS functional categories were tested, *P* values less than 0.0002 were considered not significant, as their total expectation within the cluster would be higher than 0.05. Matlab's statistics toolbox was used to perform this calculation.

Structural Analysis of Clusters:

Clusters that had one or more statistically significant functional enrichments were further analyzed using context-specific Bayesian clustering. Only the slope observations (columns 17-31) were used. First, the slope values were discretized into three categories: -1, 0, and 1, depending on whether the slope value was increasing, flat, or decreasing. Second, Friedman's Sparse Candidate algorithm was used to learn a network from the discretized slope data. It reduces the search space by identifying the best candidate parents $C_i^n = \{Y_1, \ldots, Y_k\}$ for each node X based on a statistical measure such as Euclidean distance, correlation, or mutual information, and then restricting the search to networks $\boldsymbol{B}$ in which only the candidate variables can be parents of X. Stopping criteria is based on a network scoring technique such as Minimum Description Length (MDL) in which the search stops when $\text{Score}(B_n) = \text{Score}(B_{n-1})$, or a candidate-based criterion in which $C_i^n = C_i^{n-1}$. To avoid being locked into prior candidate choices that were suboptimal in hindsight, an iterative algorithm was used to adapt candidate sets during search. To guarantee monotonic improvement in network score, the selected candidates for X_i's parents is restricted to include X_i's current parents (i.e. the selection must satisfy $C_i^n \subseteq C_i^{n+1}$). Iterations are stopped when the best score for the current iteration is no better than that for the previous iteration. An implementation of the Sparse Candidate algorithm called mrbn 0.2.0 was used (http://mrbn.dyndns.org/). It had to be modified to accept the format of this data set (see Appendix III). It uses mutual information for selecting candidate parents. A hill climbing search algorithm is used for searching. Minimum Description Length (MDL) scoring is used to stop the search.

To aid in interpreting the induced networks, non-parametric bootstrapping[7] was applied to compute confidence measures of features of the induced networks. All of the temporal

information on the gene expression data was discarded and each observation was treated as an independent sample. The following steps were then performed:

- For $i = 1, 2, \ldots, m$
  - Resample, with replacement, N instances from D. Denote by Di the resulting dataset.
  - Apply the learning procedure on Di to induce a network structure $\hat{G}_i = \hat{G}(D_i)$
- For each feature of interest, define

$$p_N^{*,n}(f) = \frac{1}{m}\sum_{i=1}^{m} f\left(\hat{G}_i\right)$$

The value p is a measure of confidence in the feature of interest, and m is the number of resamplings. Features of interest were all arcs in the induced networks. From this confidence measure, a *dominance score* for all genes was calculated, which is defined as the sum of confidence measures for all outbound arcs emanating from a gene. Genes with high dominance scores were then interpreted as being highly important for the function of a cluster. Code for calculating the dominance score from mrbn's output network can be found in Appendix IV.

Results

Clustering:

The K-means technique was used to cluster the yeast gene expression data. K-means was performed repeatedly over a range of cluster numbers. The Davies-Bouldin index and sum squared error for each cluster number are plotted in Figure 1. The Davies-Bouldin index becomes asymptotic at about 30 clusters. A cluster number of 30 was chosen rather than a larger cluster number out of concern that the number of biologically relevant features in this data set that are discernable from gene expression data is probably less than 30.

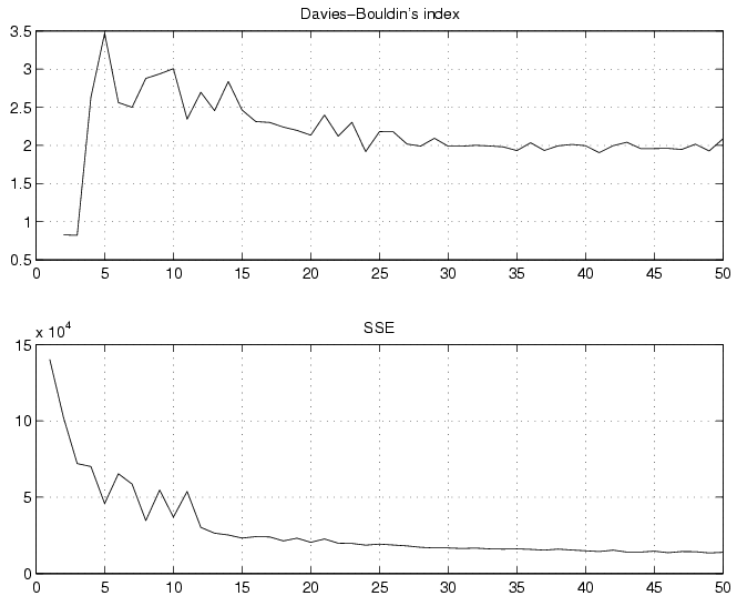

Figure 1: Davies-Bouldin index and Sum Square Error (SSE) for performing K-means clustering with a cluster number from 1 to 50

Out of the 30 clusters found by K-means, 21 were found to be significantly enriched for one or more MIPS database functional categories. Many of these clusters were either very small, thus containing low potential for discovery of new gene functionality, or very big, thus making it likely that the cluster did not have one primary function. Four of these clusters are plotted in Figures 2-5. The functional categories found for these four clusters are shown in Tables 1-4. These four clusters are representative of clusters that are most likely to have a single primary function and that are reasonably sized.

There were 68 genes in cluster 9. Twenty-four genes were not classified or unknown to the database. Six significant functional categories were found. Cluster 9 appears to be classifiable as containing genes whose products are located in the nucleus and that are involved in DNA synthesis and replication, which is part of the mitotic cell cycle.

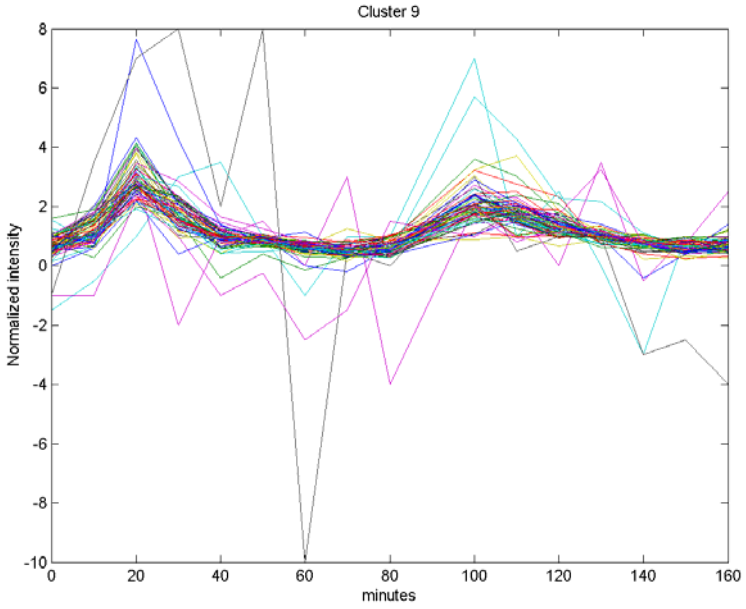

Figure 2: Data from cluster 9

Table 1: Significant functional classifications in cluster 9

Function (# in genome)	Number of genes	P
SUBCELLULAR LOCALISATION: nucleus (284)	21	<0.000001
CELL CYCLE AND DNA PROCESSING: DNA processing: DNA synthesis and replication (49)	15	<0.000001
CELL CYCLE AND DNA PROCESSING: cell cycle: mitotic cell cycle and cell cycle control (144)	11	<0.000001
METABOLISM: nucleotide metabolism: deoxyribonucleotide metabolism (8)	4	<0.000001
CELL CYCLE AND DNA PROCESSING: DNA processing: DNA recombination and DNA repair (45)	6	0.000003
CELL CYCLE AND DNA PROCESSING: cell cycle: mitotic cell cycle and cell cycle control:	2	0.000230

cell cycle checkpoints (12)

There were 49 genes in cluster 10. Twenty-five were not classified or unknown to the database. Seven functional categories were significant. Cluster 10 appears to contain genes involved in pheromone response.

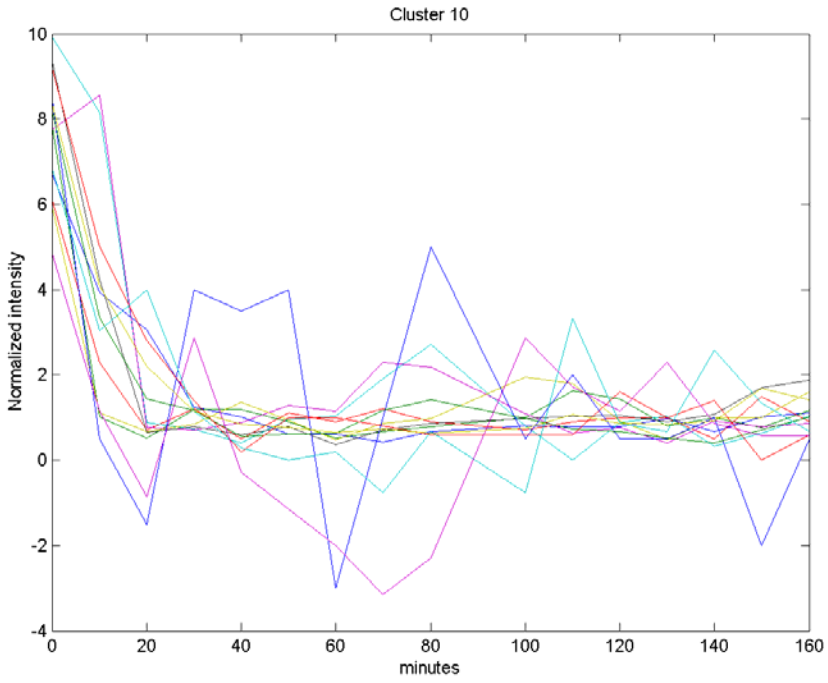

Figure 3: Data from cluster 10

Table 2 : Significant functional classifications in cluster 10

Function (# in genome)	Number of genes	P
CELL FATE: cell differentiation: fungal cell differentiation: pheromone response, mating-type determination, sex-specific proteins (66)	7	<0.000001
REGULATION OF / INTERACTION WITH CELLULAR ENVIRONMENT: cellular sensing and response: chemoperception and response: pheromone response (18)	4	<0.000001
CELL FATE: cell differentiation: fungal cell differentiation: budding, cell polarity and filament formation (84)	6	0.000003
CELL FATE: cell growth / morphogenesis: directional cell growth: other morphogenetic activities (6)	2	0.000008
CELL CYCLE AND DNA PROCESSING: cell cycle: cytokinesis (22)	3	0.000019
CELLULAR COMMUNICATION/SIGNAL TRANSDUCTION MECHANISM: transmembrane signal transduction (2)	1	0.000056
SUBCELLULAR LOCALISATION: extracellular / secretion proteins (13)	2	0.000111

There were 258 genes in cluster 13. Ninety-three genes were not classified or unknown to the database. Ten functional categories were significant. Cluster 13 appears to contain cytoplasmic proteins that are involved in protein synthesis.

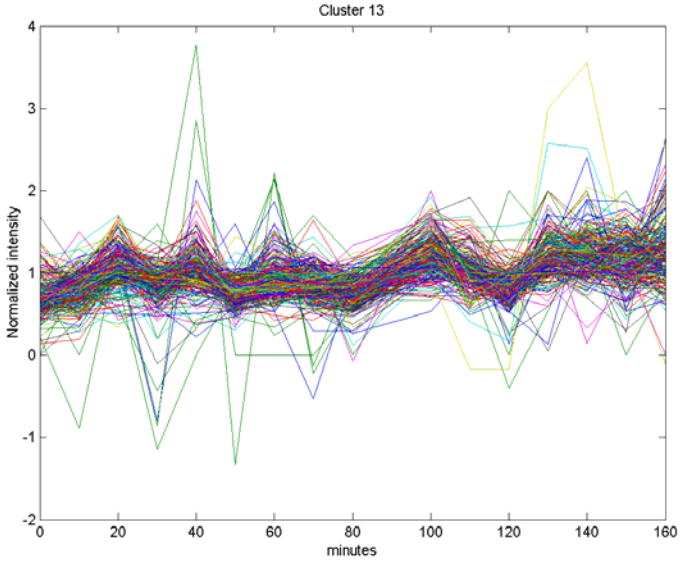

Figure 4: Data from cluster 13

Table 3: Significant functional classifications in cluster 13

Function (#number in genome)	Number of genes	P
SUBCELLULAR LOCALISATION: cytoplasm (239)	75	<0.000001
PROTEIN SYNTHESIS: ribosome biogenesis (83)	56	<0.000001
SUBCELLULAR LOCALISATION: nucleus: chromosome (23)	8	<0.000001
METABOLISM: amino acid metabolism: amino acid degradation: degradation of amino acids of the cysteine-aromatic group: degradation of glycine (1)	1	<0.000001
PROTEIN SYNTHESIS: translation: initiation (1)	1	<0.000001
CONTROL OF CELLULAR ORGANIZATION: Golgi (1)	1	<0.000001
METABOLISM: amino acid metabolism: amino acid biosynthesis: biosynthesis of the cysteine-aromatic group (1)	1	<0.000001
PROTEIN SYNTHESIS: translation (23)	6	0.000021
SUBCELLULAR LOCALISATION: nucleus (284)	25	0.000058
TRANSCRIPTION(rRNA transcription: rRNA processing (19)	5	0.000067

There were 360 genes in cluster 25. One-hundred thirty-six were not classified or unknown to the database. Twenty-three functional categories were significant. This cluster appears to be primarily involved in transcriptional control.

Table 4: Significant functional classifications in cluster 25

Function (#number in genome)	Number of genes	P
SUBCELLULAR LOCALISATION: nucleus (284)	58	<0.000001
TRANSCRIPTION: mRNA transcription: mRNA synthesis: transcriptional control (139)	36	<0.000001
CELL CYCLE AND DNA PROCESSING: cell cycle: mitotic cell cycle and cell cycle control (144)	32	<0.000001
METABOLISM: C-compound and carbohydrate metabolism: regulation of C-compound and carbohydrate utilization (40)	12	<0.000001
REGULATION OF / INTERACTION WITH CELLULAR ENVIRONMENT: ionic homeostasis: homeostasis of anions: homeostasis of sulfates (2)	2	<0.000001
TRANSPORT FACILITATION: transport mechanism: other transport mechanisms (1)	1	<0.000001
CELL RESCUE, DEFENSE AND VIRULENCE: degradation of foreign: degradation of foreign (1)	1	<0.000001
PROTEIN FATE: protein modification: modification with sugar residues (1)	1	<0.000001
TRANSPORT FACILITATION: peptide transporters (1)	1	<0.000001
METABOLISM: amino acid metabolism: amino acid biosynthesis: biosynthesis of the aspartate family: biosynthesis of lysine (1)	1	<0.000001
METABOLISM: amino acid metabolism: amino acid biosynthesis (62)	14	0.000001
SUBCELLULAR LOCALISATION: centrosome (22)	8	0.000001
TRANSPORT FACILITATION: ion transporters: anion transporters (13)	6	0.000002
CELL FATE: cell death (6)	4	0.000003
TRANSCRIPTION: rRNA transcription: rRNA synthesis (11)	5	0.000010
SUBCELLULAR LOCALISATION: endoplasmic reticulum (44)	10	0.000020
TRANSCRIPTION: other transcription activities (23)	7	0.000020
TRANSCRIPTION: rRNA transcription: rRNA processing (19)	6	0.000044
TRANSCRIPTION(tRNA transcription: tRNA synthesis (5)	3	0.000045
CELL FATE: cell aging (3)	2	0.000172
CELL CYCLE AND DNA PROCESSING: other cell division and DNA synthesis activities (3)	2	0.000172
CELL FATE: cell differentiation: fungal cell differentiation: budding, cell polarity and filament formation (84)	13	0.000195
CELL FATE: cell differentiation: fungal cell differentiation: pheromone response, mating-type determination, sex-specific proteins (66)	11	0.000240

Structural Analysis of Clusters:

Clusters 9 and 10 were chosen for structural analysis. Context-specific Bayesian clustering is computationally intensive, and it was discovered that the cluster size had to be less than 100 genes in order for the process to run to completion in a reasonable amount of time on modern desktop hardware. A 50-fold bootstrap was performed for each cluster.

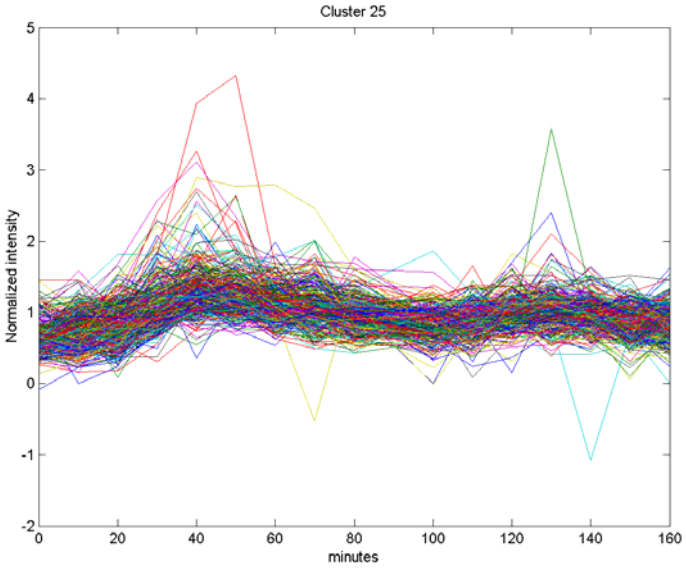

Figure 5: Data from cluster 25

The top ten dominance scores for cluster 9 are shown in Table 5. The highest scoring gene, YOR074C, encodes for thymidylate synthase, a protein that is required for the S-phase of the cell cycle, located in the nucleus, and is involved in DNA synthesis and replication. It is a member of the top two functional categories of cluster 9.

The top ten dominance scores for cluster 10 are shown in Table 6. The highest scoring gene, YLR308W is classified as a sporulation and germination gene. This is consistent with this cluster's primary function as determined by functional classification.

Table 5: Top 10 dominance scores for cluster 9

Gene	Dominance score
YOR074C	2.82
YPR121W	1.89
YPR175W	1.74
YPL256CW	1.24
YNL300W	1.15
YOR214C	0.90
YOL007C	0.89
YPR019W	0.80
YNL289W	0.73
YPL267W	0.71

Table 6: Top 10 dominance scores for cluster 10

Gene	Dominance score
YLR308W	3.17
YML084W	2.99
YER150w	2.93
YHR125W	2.47
YDL038c	2.24
YDR258c	1.78
YDR475C	1.73
YDL037c	1.49
YDL024c	0.98
YDL021Wc	0.80

Conclusion:

K-means clustering was used to find patterns in yeast cell cycle time series data, and examined the structure of these clusters using Bayesian learning to find important genes in those clusters. The functional scoring method appears to be useful in identifying statistically significant and interesting patterns in clusters. Clusters were isolated that appear to contain genes involved in DNA synthesis and replication, pheromone response, protein synthesis, and transcriptional control. In two of those clusters, genes were identified by dominance scores that may play important roles in those functions. A gene of unknown function with a high dominance score can be studied for its involvement in that cluster's primary functionality.

The Sparse Candidate Bayesian learning algorithm seems only appropriate for clusters under 100 genes in size when used on standard desktop hardware. Perhaps the algorithm can be used with larger clusters on more powerful machines.

It would be interesting to test this approach's usefulness to molecular biologists in guiding their research.

References:

1. Claverie J-M. Computational methods for the identification of differential and coordinated gene expression. Hum Mol Genet 1999;8(10):1821-1832.
2. Cho RJ et al. A genome-wide transcriptional analysis of the mitotic cell cycle. Mol Cell 1998;2:65-73.
3. Wen X et al. Large-scale temporal gene expression mapping of central nervous system development. Proc Natl Acad Sci USA 1998;95:334-339.
4. Eisen MB, Spellman PT, O. Brown P, Botstein D. Cluster analysis and display of genome-wide expression patterns. Proc Natl Acad Sci USA 1998;95:14863-14868.
5. Tamayo P et al. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. Proc Natl Acad Sci USA 1999;96:2907-2912.
6. Tavazoie S, Hughes JD, Campbell MJ, Cho RJ, Church GM. Nature Genetics 1999;22:281-285.
7. Friedman N, Linial M, Nachman I, Pe'er D. Using Bayesian networks to analyze expression data. RECOMB '00 127-135.
8. Friedman N, Nachman I, Pei D. Learning Bayesian network structure from massive datasets: the "Sparse Candidate" algorithm. Proc UAI 1999:206-215.