

Shakespeare: A Combinatoric Approach to Gene Network Modularization

Nicholas Lewin-Koh and Christopher Taylor *

June 22, 2004

1 Introduction

Since the use of mRNA microarrays has become widespread, much effort has been put into novel algorithms for analyzing the resulting expression data. Two areas in which there have been particular emphasis are the clustering of gene expression profiles and the attempt to reverse engineer genetic networks.

Clustering uses statistical or pattern recognition techniques to group genes by the pattern of their responses over time or conditions. Various methods have been applied to this end, including self-organizing maps (Tamayo et al., 1999), superparamagnetic clustering (Domany, 1999; Getz et al., 2000) and percolation (Sasik et al., 2001).

The output of any clustering technique is heavily dependent on the choice of metric. Most of the metrics used in published examples are based on the covariance of the set of expression values of from individual genes, for example using a function of correlation coefficient, or cosine distance. The effect of different choices of metrics with different clustering algorithms has not been fully explored.

Another challenge is the interpretation and validation of the clusters produced. It is usual to interpret these clusters either in biological terms e.g. by appealing to Gene Ontology (Tavazoie et al., 1999) or by appealing to notions of gene regulation - for example, to claim that genes in the same cluster are co-regulated. However, according to Qian et al. (2001), the relationships between co-regulated genes' expression profiles are rarely trivial, often involving time-delay, inversion and other transformations. Correlation will pick out some, but not all, of the underlying structure.

Another area which has attracted much effort is the reverse engineering of genetic networks from microarray time course data. This appeals to the notion of an underlying network of interactions between genes: each gene is directly regulated by a set of regulator genes, each of which in turn has its own set of

*Lilly Systems Biology Pte. Ltd., 1 Science Park Road, Singapore. The authors contributed equally to this work.

regulators. The network as a whole defines a complex function which governs the behavior of the cell and its response to perturbations or stimulus. Reverse engineering attempts to start from expression profiles of individual genes, and to infer the structure (and sometimes the parameters) of the network via some algorithm.

Many algorithms have been used for the purpose of inferring network topology, including Boolean networks (Liang et al., 1998; Lahdesmaki et al., 2003), Bayesian networks (Husmeier, 2003), decision tree methods (Soinov et al., 2003) and dynamical linear models with regression (D’haeseleer et al., 1999). There has been some success, however there are several challenges to be overcome. In many cases, biological validation of the reverse-engineered network is very difficult, as reliable data on the true underlying network for real systems is sparse. Testing on simulated data has been less successful than hoped for (Wessels et al., 2001). A reverse engineered network can fit simulated expression data quite closely even if many of the underlying connections are false (Taylor, 2003, unpublished manuscript). A further problem is that in practice, experimental samples are taken not from single cells, but from cell cultures or organisms. This can confound network identification (Chu et al., 2003).

An alternative approach has been to combine the clustering and reverse engineering ideas, and to aim for a more coarse-grained description of the regulatory structure. The module networks approach of Segal et al. (2003) attempts to simultaneously cluster the data in to modules, and to infer, using a Bayesian approach, a common set of regulator genes and regulation functions across many experimental conditions. The modular approach does not claim to describe the exact underlying network, but still allows hypothesis generation and gives a mechanistic interpretation of the modules. In Segal et al. (2003), their results were cross-validated against predictions based on transcription factor binding motifs and against Gene Ontology-based annotation.

In this work, we also follow a modular network approach. Our method, which we have named Shakespeare, takes time course data from a gene expression experiment and produces a set of control modules. The modules lie in a natural network structure which allows some inference of the regulation function and regulators governing each module. Our underlying metric is not covariance-based. Instead, we utilize the experimental design, and develop a metric on mRNA expression profiles over several treatment conditions. Broadly speaking, two genes will be close together if their expression profiles preserve similar information from the experimental space.

2 The rhAPC data set

The data set is from one of the earlier microarray experiments conducted at Lilly Research laboratories. The purpose of the experiment was to look at the mechanism of action of recombinant human Activated Protein C (rhAPC), which is indicated for severe sepsis. Often the targets which the drug affects within a cell are not entirely known. A large part of identifying the mechanism

of action of a drug is isolating these targets.

This experiment was done in the early period of the microarray lab and was not well replicated. The experiment, which is illustrated in Figure 1, was conducted as follows.

HUVEC, human umbilical vein endothelial cells, were cultured in 30 flasks. Flasks were allocated to a factorial design, involving TNF and three doses of APC, control, low and high at $0\mu\text{g/ml}$, $0.5\mu\text{g/ml}$ and $5.0\mu\text{g/ml}$ respectively. Normal endothelial cells in the body have a basal level of APC. The cultured HUVEC cells lack this basal level. TNF was used to simulate sepsis in the cells and should mimic the cellular conditions during the associated cascade of inflammation. For the four treatments involving rhAPC, the cell cultures were pre-treated with rhAPC at -18hrs from the administration of the insult TNF. In each of the six treatments, a flask was sampled at 0,1,6,12, and 24 hrs. The RNA was extracted from the culture and amplified and the amplified RNA from each sample was applied to two Affymetrix HU95A arrays, so that there were two technical replicates at each time point for each treatment.

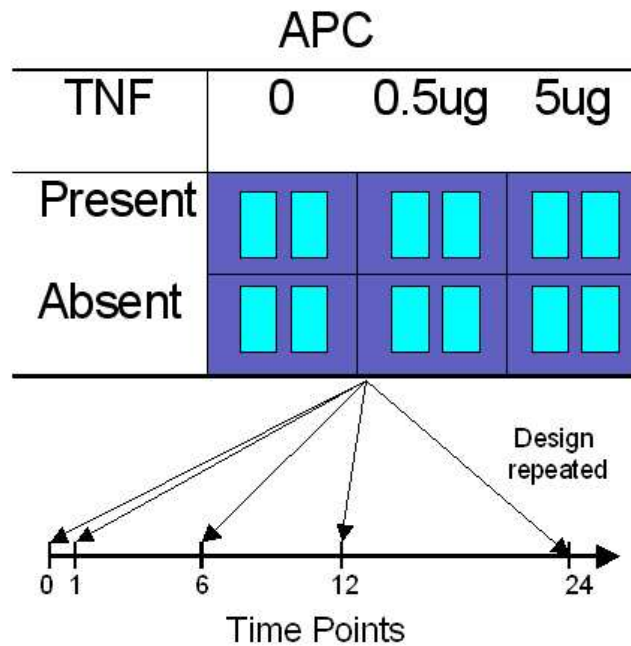


Figure 1: Experimental design

The data required preprocessing prior to analysis. The chips were normalized and background corrected using the MAS5 software from Affymetrix. Differ-

ential expression was assessed using a mixed model approach using a pooled estimate of the controls and testing contrasts for differential expression on the Time \times Gene interaction term. The p -values from these contrasts were corrected for multiplicity using the false discovery rate (FDR), as derived by Benjamini and Hochberg (1995). We selected a liberal cutoff and discarded all genes with $\text{FDR} > 0.1$. We used a second filter to account for the lack of replication by calculating for each gene g_i ,

$$R_i = \text{Control Mean} \pm 3 * \text{Control Range}$$

and used the rule,

$$\text{accept } g_i \iff \text{Signal}(g_i) \notin R_i.$$

Lastly genes were eliminated if all signals were ≤ 30 . The filtering process left us with a list of 1211 differentially expressed genes. We developed our methodology based on observations on the remaining set which we describe in the next section.

3 Background and Motivation

Our proposed techniques were motivated partly by observations in the rhAPC data set described in the previous section. We observed that for a relatively large numbers of genes, the expression profiles over time were grouped similarly when considered across treatments, even though the profiles were not all correlated. So, for example for many genes, the treatments with TNF alone and TNF with the high dose of APC had similar response profiles, the non-TNF treatments had similar profiles, and the the TNF with low dose APC had a different response profile from either. This was the case across many genes, many of which had different response profiles from each other. Examples of these patterns are illustrated in Figure 2, which shows sample expression profiles from two of the observed patterns. The first pattern is shown in the top row of Figure 2. In these figures the control is differentiated from the the two rhAPC treatments, and all the treatments with TNF form a third group. In them bottom row of Figure 2 the control groups with the the two rhAPC treatments. A second group is formed by the TNF and TNF with high rhAPC, and a final group is formed by TNF with low dose rhAPC. In this pattern, interestingly, the effect of TNF is moderated by the low dose of rhAPC and not the high dose.

A possible explanation for these patterns is the following. If a group of genes is co-regulated, the response profiles of the genes in the group may not necessarily be highly correlated. However, each gene in the group should preserve the same information from the treatment space. For example, genes g_1 and g_2 may respond in an uncorrelated fashion to each of the treatments A and B ; however, if they are co-regulated, and the responses of g_1 to A and B over time are the same, then the responses of g_2 to A and B over time should also be the same. With these observations, it is reasonable to construct a metric and modularization procedure on the gene expression profiles which takes into account the pattern of differences between treatment responses for each gene.

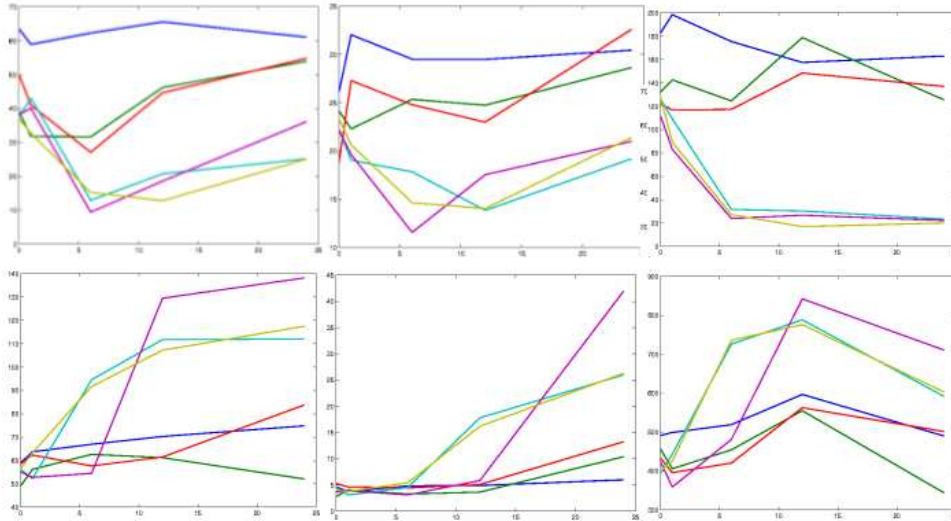


Figure 2: Patterns observed in the data. Expression *vs* time, 6 genes plotted separately. Throughout: dark blue = control, green = APC 0.5 μg , red = APC 5.0 μg , light blue = TNF, purple = TNF + APC 0.5 μg , yellow = TNF + APC 5.0 μg . Upper row, pattern 1. Lower row, pattern 2.

4 Modularization

For an experiment with M treatments, we have a finite experimental space of order M . Each $m \in \{1, \dots, M\}$ is a treatment condition. For each gene g , we assign a partition $Q(g)$ of the treatment space. The modules are then the groups of genes G_k such that for all genes $g_i, g_j \in G_k$, $Q(g_i) = Q(g_j)$.

We make no general recommendation for how to partition the treatment space, though one could use ANOVA, K-nearest neighbors or another classifier with a suitable test procedure to differentiate two treatments. For the rhAPC experiment, we only have one observation (series) per treatment, making any inference procedure difficult. We used the following heuristic to choose partitions.

To assign a partition to a gene, we used the dendrogram D from hierarchical clustering of the expression profiles. The dendrograms were generated using the Euclidean metric and average linkage distance. Each dendrogram is then defined by a sequence of five merges together with the linkage distance l_j at each merge. For each merge, we consider the subtree T below and including that merge, and calculate the distance d_j , where

$$d_j = \frac{(l_{j+1} - l_j)}{\text{diam}(\{X_{n_1}, \dots, X_{n_m}\})}$$

where for each j , X_{n_1}, \dots, X_{n_m} are the leaves in the tree below the j th merge in

Number of Genes	Clustering	Number of Genes	Clustering
166	1 1 1 2 2 2	20	1 2 2 2 2 2
140	3 5 4 1 2 1	19	5 4 1 2 3 1
66	1 2 2 3 3 3	17	3 3 3 1 1 2
61	4 3 3 1 2 1	17	1 2 1 5 3 4
57	1 3 2 2 4 2	14	1 3 2 2 2 2
54	1 2 2 2 3 2	13	4 3 3 3 1 2
45	1 3 2 4 4 4	13	1 5 2 2 3 4
42	4 1 2 1 3 1	11	3 4 3 1 2 2
36	1 2 1 3 4 3	11	1 2 1 1 1 1
34	1 2 3 2 2 2	10	4 1 2 1 3 2
31	3 3 3 1 2 1	10	1 2 2 3 5 4
30	4 4 4 1 2 3	10	1 1 1 1 2 1
22	1 3 1 2 2 2		
22	1 2 1 1 3 1		

Table 1: Modules identified in the rhAPC data set

D. A random simulation of normally-distributed data was run for m vectors, where m is the number of leaves of T ; p-values are generated based on the empirical distribution of the d value for the topmost merge of the random trees. For each gene, we then have $M - 2$ p-values. We cut the dendrogram just below the linkage distance with the lowest p-value, generating a partition for g .

The results of this procedure are summarized in Table 1. Modules of fewer than 10 genes are omitted as not significant. If Q is the partition of $\{1, \dots, M\}$ into disjoint subsets S_1, \dots, S_k , we denote a partition by a vector (q_1, \dots, q_6) where $j \in S_{q_j}$ for each j . For example, 112434 denotes the partition of $\{1, 2, 3, 4, 5, 6\}$ into the subsets $\{1, 2\}$, $\{3\}$, $\{4, 6\}$, and $\{5\}$.

The modules, based on genes with the same partition Q , are not yet very informative. We describe in the next section how we link the modules to form a module network.

5 Module Network and Interpretation

Once modules have been identified using the procedure described above, we generate a network connecting the modules as follows. On the set of partitions of a finite set, there is a natural partial ordering, given by the relation 'is a subpartition of'. We apply this to the partitions associated with the modules, representing the relation as an edge in a directed acyclic graph. The graph for the 13 largest modules identified in the APC data in Table 1 is shown in Figure 4.

We interpret the network heuristically by appealing to an equivalent decision tree formulation for each partition. Each partition of the treatment space can

be generated by at least one decision tree, where each node is a decision on one of the factors in the experiment, and the leaves are mapped to partition elements. A decision tree generated in this way represents the logical function governing the gene's regulation.

We can then express the function as a composition of simpler functions, by identifying subtrees of the decision tree. This equates to looking for subpartitions in the partition network. We can use this information to find candidates for the regulators of a module, by looking at the potential regulators (signaling pathway genes and transcription factor genes, for example) which appear upstream in the module network.

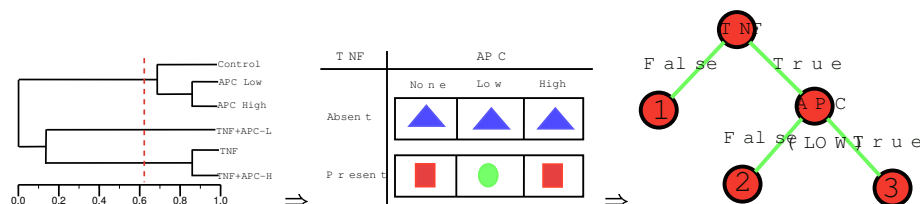


Figure 3: Partition and decision tree for a single gene

The next exercise would be to look at the annotation for the probesets for those genes included in our modular network. Of special interest would be transcription factors and kinases, which play major roles in signaling and control.

6 Cophenetic Metric and Embedding

The second part of this work concentrates on the search for a metric on the expression data which captures the features of the gene response that we are interested in, i.e. the pattern of differences with respect to the experimental design. This will allow us to embed the data into Euclidean space, to visualize the data, and to see the clustering structure. It should also be possible to map the modules identified via the modularization procedure to regions of the embedding space. The approach we propose uses the cophenetic matrix (Everitt et al., 2001).

For a treatment space of size M , the cophenetic matrix associated to a gene g , and its dendrogram from hierarchical clustering, is the $M \times M$ lower triangular matrix $A = (c_{ij})$. The entries c_{ij} are the heights in the dendrogram where two treatments become members of the same partition. We calculate the cophenetic matrix for each gene, and stack the lower triangular part to form a column vector. Pairwise correlation coefficients are then taken to generate a cophenetic correlation coefficient r_{mn} between each pair of genes g_m, g_n . This

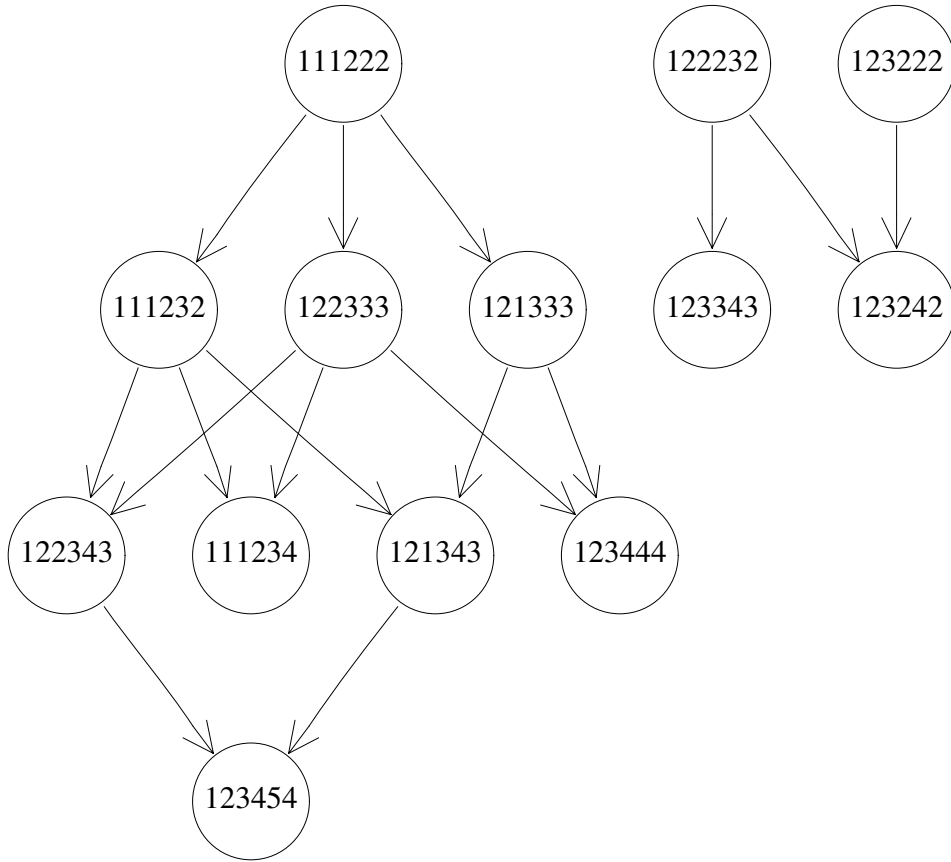


Figure 4: Module Network from rhAPC data set

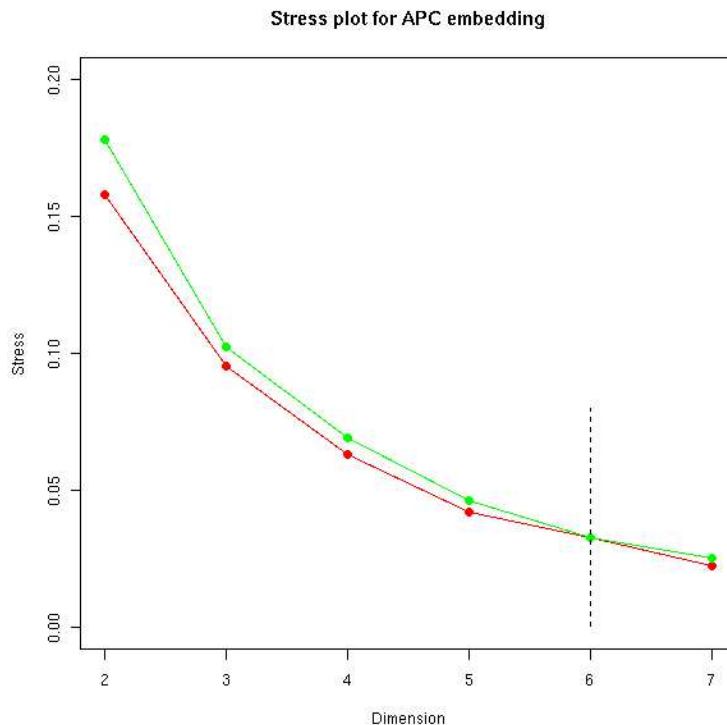


Figure 5: Distortion *vs* dimension for Multi-Dimensional Scaling

is converted to a dissimilarity measure by the transformation

$$d_{mn} = \sqrt{1 - \left(\frac{r_{mn} + 1}{2}\right)^\alpha}.$$

In this example, α was set equal to 2. Finally, we use multi-dimensional scaling to embed the genes into Euclidean space. A cost function was used to find a low dimensionality for the embedding such that the distortion was low. Figure 5 shows that a good choice for this dimensionality is 6. The first 3 dimensions of the embedded data are shown plotted pairwise in figure 6.

Some biological validation of the cophenetic metric can be obtained by looking at networks of known interactions from literature. We used the curated database of interactions from the Pathway Assist software by Ariadne Genomics, merging several curated pathways known to be involved in the cellular response to $\text{TNF}\alpha$. These interactions can be represented by a reference network, with nodes representing proteins and edges known interactions. Nineteen of the differentially expressed genes were found in the reference network, and we calculated the network distance between each pair i.e. the minimum number of edges

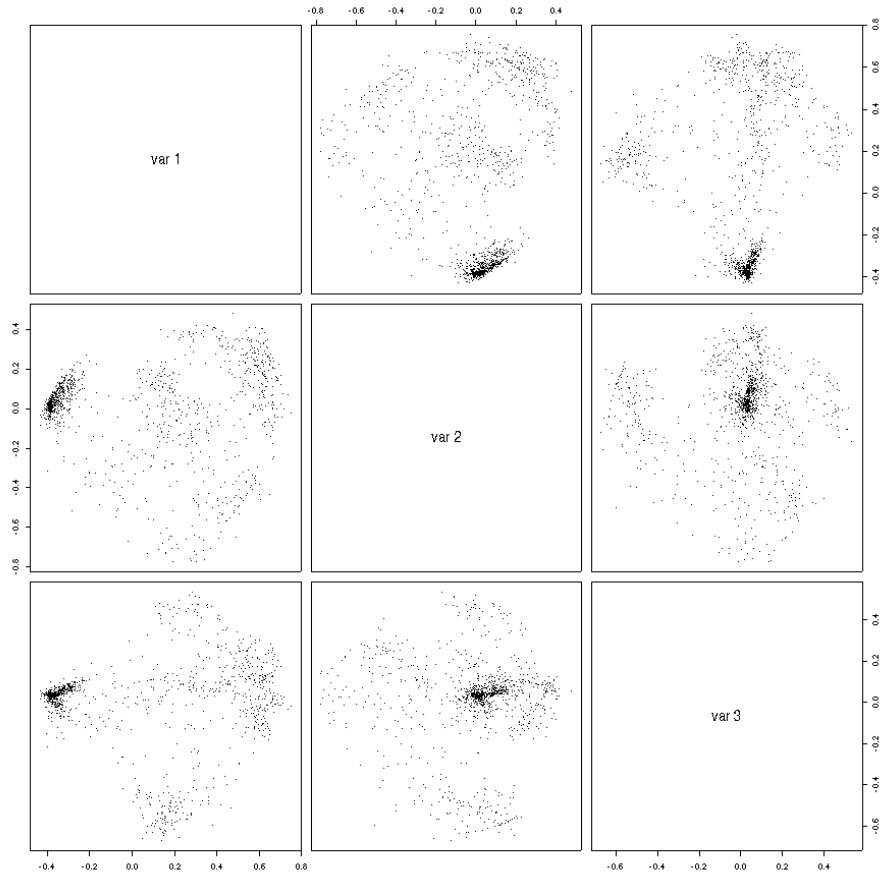


Figure 6: rhAPC data after Cophenetic Embedding: first 3 dimensions, plotted pairwise

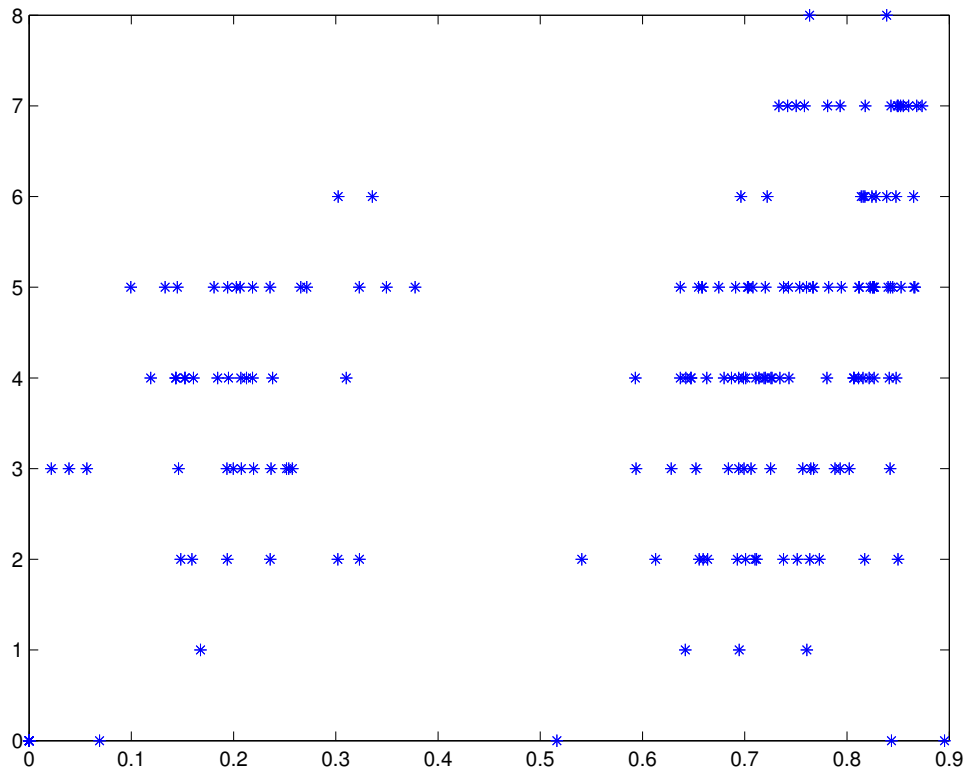


Figure 7: Network Distance *vs* Cophenetic distance

one must traverse in the graph to go from one node to the other. We plotted the cophenetic dissimilarity against the network distance for these genes (see figure 7), and calculated the correlation coefficient over all pairs. The overall correlation coefficient was 0.43. However, the plot indicates a bimodal distribution of cophenetic distances. When we restrict to pairs of genes which are cophenetically close, the correlation coefficient is 0.69, indicating a relationship between our proposed cophenetic metric on expression profiles and known biology. This suggests that our technique could be used in drug discovery, to identify possible 'neighbors' of a given target in the underlying network, enabling prediction of off-target effects or alternative points of intervention for greater efficacy.

7 Discussion and Conclusions

This work has been primarily based on the patterns of differences between the response profiles of each gene under different treatments in a factorial experimental design. We have looked at these data from two angles. Firstly, a modularization algorithm was developed based on hierarchical clustering applied to

each gene independently. This yielded control modules of genes together with a logical description of the behavior of each module, using a decision tree or partition. It also gave a module network structure based on a subtree relation. We have interpreted this network structure heuristically as giving information on the possible regulators of each module.

Our second approach has been to develop a metric, based on the cophenetic matrix, which is intended to measure the similarity between genes' patterns of responses. This is a new metric on gene expression, and has enabled us to visualize the data in a novel way. The metric also correlates with network distance from a curated database of network interactions based on literature. We can therefore infer that our metric has some biological relevance.

So far, the metric does not adequately separate the modules identified in the modularization procedure. Further work will be aimed at unifying the two approaches. In order to fully develop the techniques, they must be applied to replicated data. This will also be the focus of future papers. Another important avenue for research is to formalize the interpretation of the module network. Currently this network structure is suggestive of regulatory interactions. This needs to be put on a more firm footing, and the resulting predictions need to be tested.

The task of inferring regulatory networks from data is not an easy one, and there are some fundamental limits to what can be achieved. However, by exploiting the modular structure of the networks and by carefully designing metrics on the data, it is possible to make some inferences on regulatory structure. In this work we have taken the first steps toward a novel technique based on this approach.

References

- Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57:289–300.
- Chu, T., Glymour, C., Schienese, R., and Spirtes, P. (2003). A statistical problem for inference to regulatory structure from associations of gene expression measurements with microarrays. *Bioinformatics*, 19:1147–1152.
- D'haeseleer, P., Wen, X., Fuhrman, S., and Somogyi, R. (1999). Linear modeling of mRNA expression levels during CNS development and injury. In *Pacific Symposium on Biocomputing*, pages 41–52.
- Domany, E. (1999). Superparamagnetic clustering: The definitive solution of an ill-posed problem. *Physica A*, 263:158–169.
- Everitt, B. S., Landau, S., and Leese, M. (2001). *Cluster Analysis*. Arnold.
- Getz, G., Levine, E., Domany, E., and Zhang, M. Q. (2000). Super-paramagnetic clustering of yeast gene expression profiles. *Physica A*, 279:457–464.

- Husmeier, D. (2003). Reverse engineering of genetic networks with bayesian networks. *Biochem Soc Trans*, 31(Pt 6):1516–18.
- Lahdesmaki, H., Shmulevich, I., and Yli-Harja, O. (2003). On learning gene regulatory networks under the boolean network model. *Machine Learning*, 52:147–167.
- Liang, S., Fuhrman, S., and Somogyi, R. (1998). Reveal, a general reverse engineering algorithm for inference of genetic network architectures. In *Pacific Symposium on Biocomputing*, pages 18–29.
- 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.
- Sasik, R., Hwa, T., Iranfar, N., and Loomis, W. F. (2001). Percolation clustering: A novel approach to the clustering of gene expression patterns in *dictostelium* development. In *Pacific Symposium on Biocomputing*, pages 335–347.
- Segal, E., Shapira, M., Regev, A., Pe’er, D., Botstein, D., Koller, D., and Friedman, N. (2003). Module networks: Identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genetics*, 34(2):166–176.
- Soinov, L. A., Krestyaninova, M. A., and Brazma, A. (2003). Towards reconstruction of gene networks from expression data by supervised learning. *Genome Biology*, 4:R6.1 – R6.10.
- Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E. S., and Golub, T. R. (1999). Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. *Proceedings of the National Academy of Sciences*, 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.
- Taylor, C. N. (2003). Structural reverse engineering via linear modeling. Technical report, Lilly Systems Biology Pte. Ltd. Unpublished.
- Wessels, L. F. A., Someren, E. P. V., and Reinders, M. J. T. (2001). A comparison of genetic network models. In *Pacific Symposium on Biocomputing*, pages 508–519.