8-2008

Addressing Challenges in a Graph-Based Analysis of High-Throughput Biological Data

Andy D. Perkins
University of Tennessee - Knoxville

To the Graduate Council:

I am submitting herewith a dissertation written by Andy D. Perkins entitled "Addressing Challenges in a Graph-Based Analysis of High-Throughput Biological Data." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Computer Science.

Michael A. Langston, Major Professor

We have read this dissertation and recommend its acceptance:

Jian Huang, Robert C. Ward, Arnold M. Saxton

Accepted for the Council:
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a dissertation written by Andy Dale Perkins entitled "Addressing Challenges in a Graph-Based Analysis of High-Throughput Biological Data." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Computer Science.

Michael A. Langston
Major Professor

We have read this dissertation
and recommend its acceptance:

Jian Huang

Robert C. Ward

Arnold M. Saxton

Accepted for the Council:

Carolyn R. Hodges
Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Addressing Challenges in a Graph-Based Analysis of High-Throughput Biological Data

A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Andy Dale Perkins
August 2008

Dedication

To Carrie

Acknowledgments

I would like to thank my major advisor, Dr. Michael A. Langston, for introducing me to the field of computational systems biology and for his guidance and instruction.

Thanks also go to Dr. Jian Huang and Dr. Robert C. Ward, for reviewing this manuscript and offering helpful advice, and to Dr. Brynn H. Voy and Dr. Arnold M. Saxton for their invaluable guidance and unwavering willingness to answer my biology- and statistics-related questions, respectively.

My gratitude goes to my fellow students, who have participated in discussions on the topics contained here, particularly Bhavesh Borate, Suman Duvvuru, John Eblen, Gary Rogers, Jon Scharff, and Yun Zhang.

Thanks to Suzanne Baktash for help with reading, revision, and attempting to hold me accountable to deadlines.

Lastly, I wish to acknowledge Dr. Mikael Benson of the Unit for Clinical Systems Biology at the Sahlgrenska University Hospital and Göteborg University for the extraordinary opportunity to work with his group during the summer of 2007. Thanks also to unit members Dr. Reza Mobini and Fredrik Barrenäs for helping me discover many key biological concepts during my time there, and also for helping me navigate day-to-day life in Sweden.

Abstract

Graph-based methods used in the analysis of DNA microarray technology can be powerful tools in the elucidation of biological relationships. As these methods are developed and applied to various types of data, challenges arise that test the limits of current algorithms. These challenges arise in all phases of data analysis: data normalization, modeling biological networks, and interpreting results.

Spectral graph theory methods are investigated as means of threshold selection, a key step in constructing graphical models of biological data. Also important in constructing graphs is the selection of an appropriate gene-gene similarity metric, and an overview of similarity profiles for some biological data sets is present, along with a similarity thresholding method based upon structural properties of random graphs.

The identification of altered relationships between two or more conditions is a goal of many microarray gene expression studies. Clique-based methods can identify sets of coexpressed genes within each group, but additional computational methods are required to uncover the differential relationships and sets of genes changing together between groups. Differential filters are reviewed to highlight those changing interactions and sets of changing genes. The effect of various normalization methods on these differential results is also studied.

Finally, how methods commonly used in the analysis of gene expression data can be used to investigate relationships in noisy and incomplete historical ecosystem data is explored.

Contents

List of Tables

List of Figures

Chapter 1

Introduction and Background

1.1 Introduction

With the advent of high throughput platforms for gene expression profiling, many novel computational methods have been introduced to extract biological knowledge from the mass of data available. These range from speedy heuristics such as hierarchical clustering to computationally-intensive exact algorithms for extracting dense sets of highly-related genes. Other than the principle challenge of developing tools that produce biologically-relevant results, many challenges arise in the preprocessing and modeling of, and computation on, various data sets.

In addition to the challenges that are currently faced, designing and applying novel computational methods to new, larger, and different types of data sets leads to even more issues. There is a constant effort to develop new technologies that allow researchers to examine the genome at higher resolution and finer granularity. New techniques and platforms are also needed to keep up with current biological knowledge, an example of which is the recent technology of microRNA expression profiling.

1.2 Motivation

This dissertation focuses on the challenges that arise in a graph-based analysis of high-throughput data, specifically the clique-based methods described in [1, 2]. These methods have been applied to a variety of genomic and proteomic data from an array of species [1, 3, 4], as well as historical ecosystem data [5]. Challenges arise in each step of the analysis process. Normalizing data, deciding on parameters for building models, processing results, and dealing with noisy or incomplete data all present unique problems that must be faced to turn raw data into useful information.

Many of these challenges crop up in most graph-based analyses of gene expression data, such as the selection of an appropriate similarity metric for computing gene-gene correlations. These challenges have been met in a variety of ways. Unfortunately, there is often no clear answer or best approach, so it is wise to be equipped with a “toolbox” of methods. What follows is a description of the steps involved in this graph-based approach and an analysis of some issues and questions that arise as one progresses through the entire process.

1.3 Microarray technology background

The first DNA microarray experiment was reported in 1995 by Mark Schena et al., then at Stanford University. The authors analyzed the expression profile of the plant *Arabidopsis thaliana* and performed a differential expression analysis between the wild type and a transgenic line overexpressing the gene HAT4 [6]. Since then, the use of microarray technology has exploded. There are presently more than 8,000 publicly-available gene expression data sets comprised of over 220,000 samples available online in the Gene Expression Omnibus (GEO) repository [7, 8].

DNA microarrays allow the large-scale monitoring of gene expression levels, as measured by cellular mRNA concentration, for thousands of gene transcripts simultaneously. The level of gene expression is measured by the detection of dye-labeled mRNA hybridized to complementary DNA transcripts printed on the microarray. Performing microarray analysis over multiple conditions facilitates the detection of relative changes in expression, which is usually performed with the help of traditional differential expression methods [9, 10].

Although only expression levels from one microarray per sample group are necessary for most differential expression methods, many graph-based approaches require data from multiple arrays from each group. These methods can then uncover changes in consistent relationships between genes rather than comparing expression levels from two different genomic "snapshots." The samples to be processed on these microarrays can be collected from different strains, individuals, or taken at various time points.

1.3.1 Normalization and preprocessing

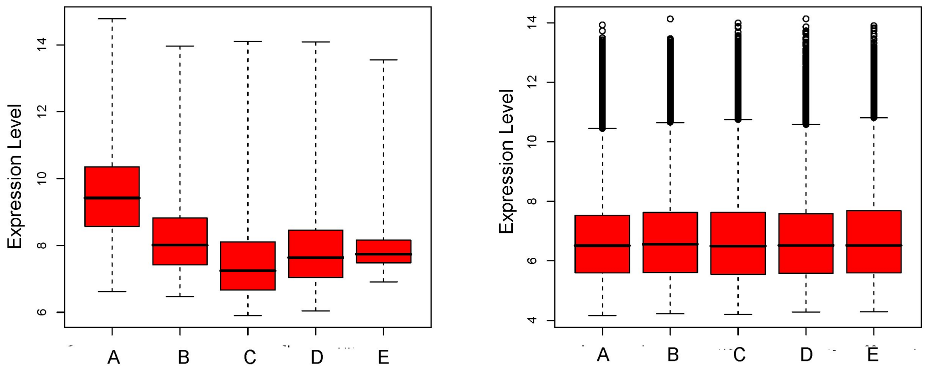
Raw data should be preprocessed before computational or statistical tools can be applied. After background signal is subtracted from each expression level, a normalization method is usually used to bring values on each array into comparable ranges, illustrated in Figure 1.1. This is necessary due to differences in microarray chips, dye intensity, and other sources of variation within the technology itself [11].

There are several methods to preprocess and normalize raw microarray data, including MAS5, RMA, gcRMA, PDNN, VSN, and d-Chip, among others. Each of these is explored in more detail in Chapter 5. It has been observed that varying the normalization method often leads to inconsistent results [13]. The question therefore arises of which preprocessing method is "best," or which is most appropriate under differing circumstances. Chapter 5 investigates the effect of normalization method choice on differential comparisons other than basic differential expression.

1.4 Graph-based analysis toolchain background

Most graph theoretical methods model biological relationships, in this case derived from gene expression data, using the mathematical structure of a graph. Gene transcripts are represented by vertices in the graph while relationships between genes are represented by edges. Edge weights are determined by similarity of the respective transcripts in the expression profile. Modeling biological networks using graphs offers the benefit of having at hand the rich collection of knowledge and results from the field of graph theory.

Along with preprocessing and normalization, the procedure of building the coexpression network and subsequent computation can be viewed as distinct steps and implemented in a graph-based analysis toolchain. Figure 1.2 shows the general progression of the clique-based



(a)

(b)

Figure 1.1: Effect of normalization on gene expression values on five microarrays from the data set described in [12]. (a) Raw .CEL data before normalization. (b) After RMA preprocessing and normalization.

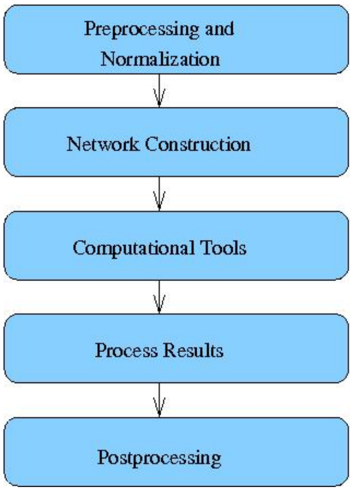


Figure 1.2: A toolchain for the analysis of high throughput microarray data. Computational tools are used to extract sets of coexpressed genes.

analysis toolchain. Specifically, the computation of gene-gene similarities, thresholding, and deriving biological knowledge from long lists of computational results are steps relevant to this work.

1.4.1 Building gene coexpression networks

There are two key steps involved in building gene coexpression networks: computing similarities between gene expression profiles, and applying a threshold to these similarities to retain only putatively biologically-significant relationships. Choices made at these steps can significantly affect results at later stages. Thresholds must be selected so that similarities surpassing a particular level represent only true biological relationships, as supported by the data. Of course, the choice of a similarity metric can cause edges of a particular weight to surpass or fall below that threshold, especially if the value is marginal.

As previously mentioned, the process begins with a weighted graph with edge weights determined by some similarity value computed for each pair of vertices, based upon their associated gene's expression profile. These similarities can be computed using a variety of techniques, with the Pearson product-moment correlation coefficient being the most popular choice. The Pearson correlation coefficient is given in Equation 1.1, with $\overline{x}$ being the sample mean and S_x the sample standard deviation. Other possibilities include the Spearman rank correlation coefficient, the shrinkage estimate, and mutual information. Chapter 3 seeks to identify commonalities in the structure of biological graphs, based upon their clique profiles, constructed using the various similarity metrics above.

$$r_{xy} = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{(n-1)S_x S_y} \tag{1.1}$$

The thresholding step takes as input the weighted graph and a threshold th, and removes all edges whose absolute value fall below that threshold. Absolute edge weights are used since negative correlations are often as informative as the positive ones, in the case of suppressive relationships. The edge weights are removed from the remaining edges, leaving an unweighted graph. Isolated vertices, vertices with degree zero, can be also be removed from the analysis. A network of only putatively biologically-significant relationships remains.

There are a variety of methods to select this threshold value th. A popular choice is to compute the significance of each correlation coefficient. It is well-known that if the true correlation between two random variables is 0, then

$$t = \frac{r}{\sqrt{(1-r^2)(N-2)}} \tag{1.2}$$

is distributed approximately as t with $N-2$ degrees of freedom, where N is the sample size and r is the calculated Pearson correlation coefficient [14]. A simple Student's t-test produces a p-value, which serves as a measure of significance for the associated correlation value. This value indicates the probability of obtaining a result at least as extreme as the observed value. After adjustment for multiple tests, the usual significance threshold of $\alpha < 0.05$ can then be applied. Of course, this method is only applicable to similarities computed using Pearson correlation coefficients, but there is also a method for testing significance of the Spearman rank correlation [15].

The selection of the threshold value th presents one of the most significant challenges in a graph-based analysis. Many papers describing a gene expression microarray study also include a discussion of threshold selection, and there is no clear concensus as to which method should be used. In fact, Elo et al. point out that the performance of traditional methods for threshold selection can vary and that there has been no systematic approach to addressing the thresholding problem [16].

Chapter 2 seeks to address the thresholding problem, while Chapter 3 examines some of the structural properties of various biological graphs using different similarity metrics. Chapter 6 addresses the construction of networks from corrupt and incomplete data.

1.4.2 Computational methods

Two well-known problems in graph theory are used to extract key sets of related genes: maximum clique and maximal clique enumeration. Paraclique [17] is also often employed to adjust for noisy data. Although they are not discussed here, many other problems in graph theory have found utility in biological applications, including biclique, cluster edit, feedback vertex set, and dominating set among others [18, 19, 20, 21].

Definition 1. *Given a graph $G = (V, E)$ a* clique *$C \subseteq V$ is a set of vertices such that there exists an edge between every vertex in C. In other words,*

$$\forall u, v \in C, (u, v) \in E$$

Also known as a complete subgraph, a clique of size five is illustrated in Figure 1.3. The decision version of the clique problem takes as input a graph $G = (V, E)$ and constant k and answers the question "Does G contain a clique of size at least k?" An algorithm to solve the clique problem, along with a binary search, can be used to find the maximum clique size in the graph. The search version of the clique problem also returns the members of the clique. Determining whether a graph has a clique on at least k vertices is $\mathcal{NP}$-complete [22].

Finding all maximal cliques in a graph is also useful for finding sets of genes with similar expression profiles. While most biological graphs might only have a few cliques of maximum size, the number of maximal cliques can number in the thousands or millions.

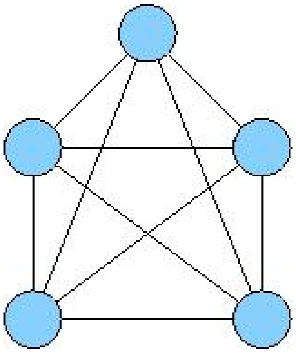


Figure 1.3: A clique on five vertices.

Definition 2. *A* maximal clique *is a complete subgraph to which no additional vertices can be added.*

The four highlighted vertices in Figure 1.4 do indeed form a clique of size four, but it is not maximal–vertex A can be added, resulting in a clique on five vertices. Since a clique of maximum size in the graph G must also be maximal, maximal clique enumeration must also be computationally difficult; in fact, it is $\mathcal{NP}$-hard.

In the context of gene-gene relationships, a maximal clique is a maximal set of completely connected genes–the densest set of relationships that can be extracted from the data. Edges linking vertices in a clique represent the "most trusted associations" between genes [23]. It is important to note that maximal cliques can be overlapping. That is, a single gene can appear in multiple maximal cliques, offering greater fidelity with actual biology when compared with traditional clustering methods.

The significance of these maximal cliques is based upon the principle of guilt by association, the idea that genes having a similar expression profile are more likely to share a biological pathway, and possibly have the same function [24]. By definition, all genes in a clique have similar expression profiles, with the definition of "similar" being determined by the chosen threshold. Therefore, one would expect genes appearing in cliques together to generally have similar function and participate in the same pathways. When the function of a gene is unknown, the function of its fellow clique members can be used to extrapolate its purpose.

A benefit of the clique-based approach is the natural resistance to false positives, since all edges must be present to form a clique. This protection comes at a price–microarray data is inherently noisy, and even a single missing connection can cause a vertex to be lost from the clique. The paraclique algorithm was invented by M. A. Langston and first reported in [17] to overcome this limitation of the technology. Paraclique is based upon the observation that if a vertex is connected to say, all but one of the genes in a clique, then that vertex should likely belong to the clique as well. The single missing edge is likely due to noise bringing the weight of that edge just under the threshold. The algorithm "gloms" these vertices onto the original clique, resulting in a "paraclique," as illustrated in Figure 1.5. Various implementations of the algorithm allow for the specification of the number of allowable missing edges (referred to as the "glom factor"), as well as a lower threshold for edges that can be incorporated into the paraclique.

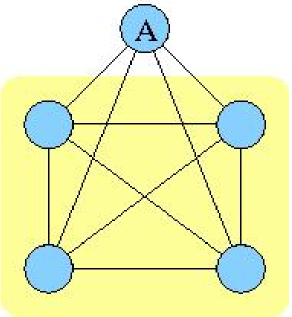


Figure 1.4: The four highlighted vertices do not form a maximal clique.

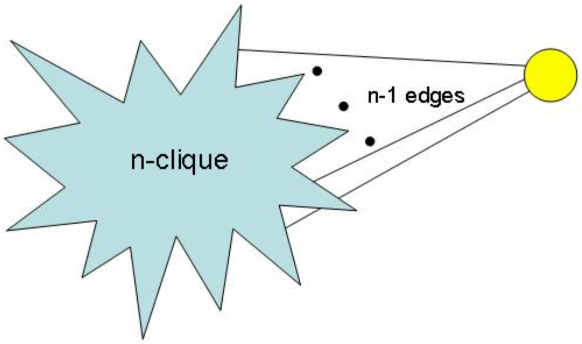


Figure 1.5: A paraclique with glom factor of $n - 1$.

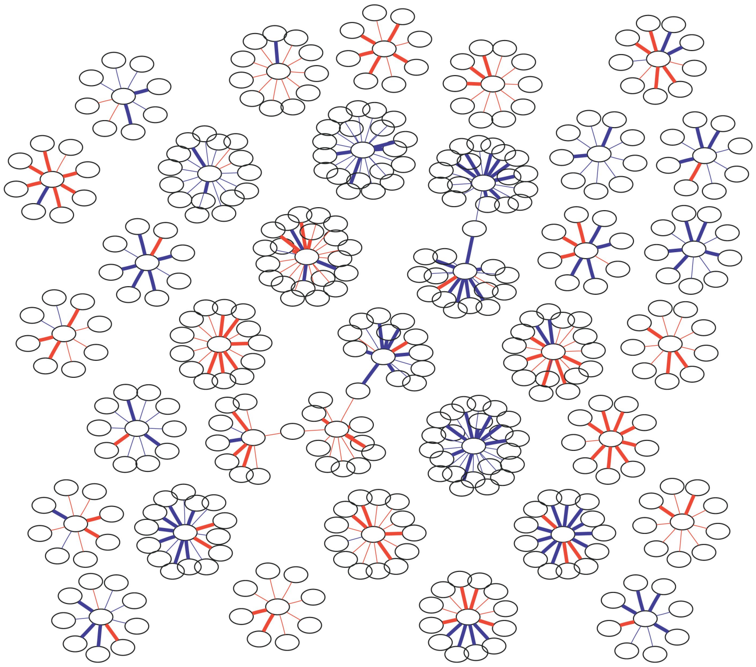


Figure 1.6: Differential correlation graph. Source: [1].

1.4.3 Differential filters

Maximal clique results on biological graphs often take the form of thousands or millions of sets of completely connected vertices. With such a large set of cliques and a high degree of overlap, computational tools are often required just to make sense of the output. To address this abundance of results when performing comparative analyses, differential filters have been developed to identify genes, associations, and groups of interacting genes for further study [1].

In a differential analysis, which involves the comparison of expression profiles over two or more groups, the central problem is to identify which genes and sets of relationships are changing. These changes could result from exposure to some environmental factor or stress or be due to some fundamental difference such as gender. If a graph is created and maximal cliques are enumerated separately for each group, it is possible to identify correlations that "appear" or "disappear" from one group to the next. Examining differences at the subgraph level, which involves identifying sets of genes and associations changing together from one group to another, is also informative. Chapter 4 formalizes these concepts, termed differential correlation and differential topology, respectively. Figure 1.6 from [1] shows a differential correlation graph in which red and blue edges are those that surpass a preselected threshold in one group but fall below a lower threshold in the other.

1.4.4 Postprocessing

Once dense sets of related genes or those genes changing associations between groups have been identified, it is often desirable to compare these results to current biological knowledge. The purpose here is twofold: to ensure that what was found is true and to infer a function or regulatory mechanism for certain genes based upon the genes with which they are associated.

For instance, if some gene A of unknown function is present in a clique or paraclique with many immune response-related genes, then gene A is more likely to have an immune-related function (recalling the idea of guilt-by-association). Likewise, genes that have been identified as participating in many changing relationships at the edge or subgraph level upon exposure to some environmental factor might share a regulatory relationship with, say a transcription factor, whose associated gene transcript might have also been identified. There are many tools available to examine the function of genes and sets of genes, such as Gene Ontology [25], WebGestalt [26], and Ingenuity Pathway Analysis [27].

It is important to view such functional and regulatory annotations using results as a starting point for biological validation. Given the noise inherent in microarray data, results that are highly dependent on threshold selection and the small but ever-present threat of false positive, only biological experiments can truly validate computational findings. The value of the differential screens is that a list of $55,000$ gene transcripts can be distilled into a set of perhaps the 100 possibly most important in the response to a certain disease, stress, or environmental factor. Similarly, methods like paraclique produce putative functional annotations to test, where none previously existed.

1.5 Theory and computational requirements

Although we have seen that both the maximum clique and maximal clique enumeration problems are $\mathcal{NP}$-complete, it is still desirable to solve these problems exactly. A collection of heuristics are explored in [28], but with the expense and effort involved in collecting gene expression data, an exact approach seems more sensible. This is especially true when considering that exact results are necessary to build networks that are maximally reliable and consistent with actual biology. Luckily, solving these problems exactly for large data sets is not as intractable as it might first seem.

Given an input of size n and parameter k, a problem is fixed parameter tractable (FPT) if there exists an algorithm to solve it that runs in $O(f(k)n^c)$ time, where c is constant. This gives hope that there might be polynomial time algorithms for certain problems. The benefit here is that the running time of the algorithm is no longer based upon the input size, but instead upon some fixed parameter k. With a favorable function $f(k)$ and constant c, solving instances of FPT problems becomes feasible. The most comprehensive work on fixed parameter tractability is the 1999 monograph of Downey and Fellows [29].

Maximum clique is not FPT, but its complementary dual, vertex cover, is FPT.

Definition 3. *Given a graph $G = (V, E)$, a* vertex cover *is a set $VC \subset V$ such that each edge in E has an end point in VC. In other words, $\forall(u, v) \in E, (u \in VC) \vee (v \in VC)$.*

Note that a clique C in $G = (V, E)$ is an independent set in G^c. Now, $V - C$ is a vertex cover in G^c. The maximum clique problem can therefore be approached by finding a minimum vertex cover in graph complement. The input to the decision version of the minimum vertex cover problem is a graph G and parameter k. The question to be answered is "Does the graph G contain a vertex cover of size at most k?" The current best bound on the complexity of the vertex cover problem is $O(1.2852k + kn)$ [30].

Although the complexity of the vertex cover problem is much better than one might expect for a $\mathcal{NP}$-complete problem, identifying a maximum clique for some graphs can result in a large value for k. Add to this the fact that the problem must be solved multiple times using a binary search, and that the hardest case–the one in which there is no clique of size k, for the largest value of k–must be solved once to verify there are no larger cliques. In practice, larger biological graphs also tend to have a larger maximum clique size. Hence, computational requirements can still be rather restrictive. For this reason, parallel implementations of the vertex cover algorithm are often employed [31].

Chapter 2

Spectral Thresholding

2.1 Introduction

The two main operations in building gene co-expression networks are computing gene-gene similarities, and setting a threshold past which these similarities are considered biologically significant. This thresholding step has been addressed in the literature in various fashions, including choosing a percentage of the highest correlations as a cutoff value [32, 33], retaining only statistically-significant similarities [34, 35], permutation testing [23], examining network structure and properties [16, 36], and control spot verification [1], among many others.

Here, the focus is on a linear algebra-based method to select a threshold that can quantify "real" biological relationships. It is assumed that some pairwise gene-gene similarity has been computed. As previously mentioned, vertices representing genes are joined by an edge weighted with their similarity. Now an attempt to zero in on an appropriate edge weight threshold with the aid of spectral graph theory techniques can be made.

Spectral graph theory has a long history of use in the natural sciences as well as decades of work developing its mathematical foundations [37, 38, 39, 40]. Spectral methods have been used in computer graphics [41, 42], communications networks [43, 44], and theoretical chemistry [45, 46, 47]. From the biological domain, spectral clustering and partitioning approaches have been applied to high-throughput biological data. Eigenvalues and eigenvectors of graphs produced from microarray data have been studied and used for classification of samples in the classic acute lukemia data of Golub [48] and also to cluster genes and conditions simultaneously, so-called co-clustering, or biclustering [49].

Spectral graph theory involves the decomposition of a graph into its representative eigenvalues and eigenvectors and the study of these pieces. Examination of these spectral components can uncover information about the structure of the graph, in this case representative of a biological network. For example, the number of connected components in a graph can be determined, as well as whether the graph is complete or bipartite, and how well connected it is.

Although they have found wide utility, spectral methods face some limitations when applied to large data sets, such as those generated by high-throughput technologies such as microarrays. Computing exact eigenvalues on large data sets with conventional methods is expensive in both processing and memory requirements. In this paper, a simple spectral clustering algorithm is used. This algorithm is based upon a single selected eigenvalue and eigenvector, which is relatively easy to estimate for sparse matrices. This particular algorithm operates on unweighted, undirected graphs, which makes it possible to apply an edge weight threshold at a variety of points and examine the spectral properties of the resulting graph. This approach was also chosen for the property of being unsupervised, which removes any bias that might be introduced by the selection of parameters such as cluster size or number of iterations.

2.1.1 Definitions

Definition 4. *Given an undirected, unweighted graph $G = (V, E)$, define the* adjacency matrix $\mathbf{A}$ *of G to be*

$$\mathbf{A}_{u,v} = \begin{cases} 1 & \textit{if } (u,v) \in E, \\ 0 & \textit{otherwise.} \end{cases}$$

Definition 5. *Let the diagonal $n \times n$* degree matrix *D be defined to be*

$$\mathbf{D}_{u,v} = \begin{cases} deg(u) & \textit{if } u = v, \\ 0 & \textit{otherwise.} \end{cases}$$

Definition 6. *Define the* Laplacian matrix *of the simple graph G, $\mathbf{L}(G)$, to be*

$$\mathbf{L}(G) = \mathbf{D} - \mathbf{A}$$

or alternatively,

$$\mathbf{L}(G)_{u,v} = \begin{cases} deg(u) & \textit{if } u = v, \\ -1 & \textit{if } u \neq v \textit{ and } (u,v) \in E, \\ 0 & \textit{otherwise.} \end{cases}$$

Definition 7. *The* Normalized Laplacian matrix *of the simple graph G, denoted $\mathcal{L}(G)$ is defined by*

$$\mathcal{L}(G)_{u,v} = \begin{cases} 1 & \textit{if } u = v, \\ \frac{-1}{\sqrt{deg(u) \cdot deg(v)}} & \textit{if } u \neq v \textit{ and } (u,v) \in E, \\ 0 & \textit{otherwise.} \end{cases}$$

2.1.2 Background

From this point forward, the Laplacian and normalized Laplacian will be referred to by L and $\mathcal{L}$, respectively, when the graph being operating on is clear or irrelevant. In addition, unless otherwise noted, operations will be performed on the Laplacian matrix L rather than the normalized Laplacian matrix.

Definitions 6 and 7 illustrate that the Laplacian matrix of a graph contains not only adjacency information, but also the degree structure of the graph. For this reason, it is preferred to analyze the Laplacian matrix rather than the adjacency matrix alone.

The characteristic equation of G, based upon the Laplacian, is given by equation 2.1.

$$det(\mathbf{L} - \lambda\mathbf{I}) = 0 \tag{2.1}$$

The symmetric eigenvalue program will be solved on the Laplacian matrix L by finding solutions to the linear system

$$\mathbf{Lx} = \lambda\mathbf{Ix} \tag{2.2}$$

giving a set of eigenvalues and eigenvectors. Note that the adjacency matrix A, and hence the Laplacian matrix, is symmetric for all undirected graphs and is square of order n, where $n = |V|$. This implies that there are n eigenvalues and eigenvectors. Since L is symmetric and positive-definite, all eigenvalues are real and the eigenvectors are orthogonal.

To facilitate referencing certain eigenvalues by their relative magnitude, the eigenvalues of L are ordered such that they are monotonically increasing as in equation 2.3.

$$\lambda_0 \leq \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_{n-1} \tag{2.3}$$

These eigenvalues, along with their associated eigenvectors $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_{n-1}$ are referred to as the “spectrum” of G.

Since the main focus is on the spectral analysis of biological data, one can make use of the structure of biological networks to solve the eigenvalue problem. As previously mentioned, gene co-expression graphs are undirected, having an edge between vertices only if there is some significant biological relationship between the two genes represented by those vertices. It is also known that, in general, there are relatively few of these biologically significant relationships, resulting in sparse graphs.

Although biological data sets usually do give rise to sparse graphs (and hence sparse Laplacian matrices,) there are often a large number of gene transcripts involved. With such large matrices, solving the eigenvalue problem is often prohibitive in terms of processing time and memory requirements. The Arnoldi methods [50] for large scale eigenvalue problems of ARPACK [51] overcome these limitations when one is interested in only select eigenvalues and eigenvectors. These symmetric, sparse eigensolvers are employed as implemented in ARPACK and incorporated within MATLAB. The sparse matrix support in MATLAB also eases the memory requirements during computation and storage of the Laplacian.

2.1.3 Connectivity

The smallest non-zero eigenvalue is referred to as the Fiedler value, or the algebraic connectivity of the graph, due to [52]. In a connected graph, the algebraic connectivity will be equal to λ_1. This value gives a measure of how well connected the graph is, or a relative measure of the proportion of edges that must be removed before the graph becomes disconnected. This tells how well the graph can be cut. As the graph becomes increasingly sparse, λ_1 tends toward zero.

Since the multiplicity of the zero eigenvalue in the spectrum of the Laplacian is determined by the number of connected components in the graph, $\lambda_1 = 0$ for any disconnected graph. In fact, a graph with k connected components has the property that

$$\lambda_0 = \lambda_1 = \ldots = \lambda_{k-1} = 0 \tag{2.4}$$

$$0 < \lambda_k \leq \lambda_{k+1} \leq \ldots \lambda_{n-1} \tag{2.5}$$

The primary interest here is the smallest non-zero eigenvalue and its associated eigenvector. In an ordered list of eigenvalues for a disconnected graph, one cannot be sure to which component the smallest non-zero eigenvalue belongs. To prevent the analysis from being based upon the spectrum of a smaller, insignificant pieces of the graph, Observation 1 allows the graph to be decomposed into its connected components and their spectra computed individually. This analysis can then be continued on the spectrum of only the largest connected component.

Observation 1. *"The spectrum of a graph is equal to the union of the spectra of its connected components" [37].*

Proof. This follows from the definition of the spectrum of a graph [37]. Let $G = (V, E)$ be a graph on n vertices with k connected components, then the rows and columns of the adjacency matrix of G (and hence its Laplacian matrix L) can be arranged in block diagonal form so that

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & 0 & 0 & 0 \\ 0 & \mathbf{A}_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{A}_k \end{bmatrix}$$

where $\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_k$ are the adjacency matrices of the connected components of G. The goal is to to solve the system of equations $\mathbf{Ax} - \lambda\mathbf{Ix} = 0$:

$$\begin{bmatrix} \mathbf{A}_1 & 0 & 0 & 0 \\ 0 & \mathbf{A}_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{A}_k \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \cdots \\ \mathbf{x}_k \end{bmatrix} = \begin{bmatrix} \lambda\mathbf{x}_1 & 0 & 0 & 0 \\ 0 & \lambda\mathbf{x}_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \mathbf{x}_k \end{bmatrix}$$

Let m_i be equal to the number of vertices of the ith connected component, and hence the order of the matrix $\mathbf{A}_i$. Because of the block nature of the matrix $\mathbf{A}$, the elements of $\mathbf{A}_i$ only operate on the first m_i elements of the vector $\mathbf{x}$ in the matrix-vector multiplication, and contribute only to the first m_i rows and columns of the product matrix. So the equation can be solved as separate systems of linear equations:

$$\mathbf{A}_i\mathbf{x}_i - \lambda\mathbf{x}_i = 0$$

for each connected component i from 1 to k, producing the spectrum of each individual connected component.

Similarly, the spectrum of each connected component, produced from solving k linear systems as above, can be combined into a single system of equations, with $\mathbf{A}$ having the aforementioned block diagonal form. □

2.1.4 Spectral clustering

The goal of a clustering algorithm is to partition data points into subsets of highly-related data items, usually based upon some distance measure or similarity metric. When applied to vertices in a graph, the methods are often referred to as graph partitioning algorithms. For example, there are three natural clusters in Figure 2.1.

There have been many spectral-based clustering algorithms proposed for approximating an optimal partitioning. Among these are the algorithms described in [53, 54]. A good review of spectral clustering algorithms can be found in [55].

The simplest spectral clustering method is a recursive bisectioning of the graph [56, 57]. One simple implementation of a spectral bisection algorithm examines the eigenvector associated with the smallest non-zero eigenvalue, which is often referred to as the Fiedler

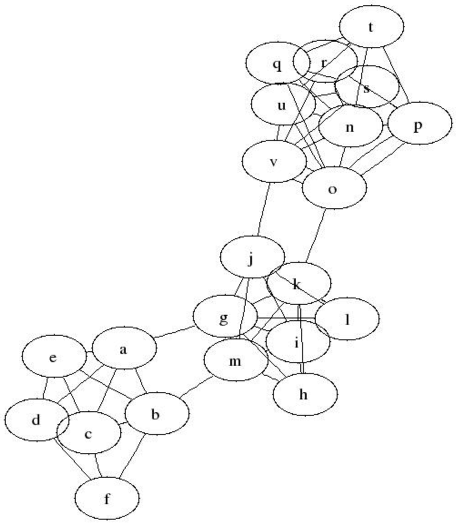


Figure 2.1: A graph comprised of three natural clusters.

vector. Elements of the Fiedler vector are sorted, and separated into two groups: elements that are less than zero, and elements that are greater than zero. Since each eigenvector has length n, and each element in an eigenvector is associated with a vertex in the graph, this gives a bipartitioning of the data. This procedure is then repeated until the desired level of granularity is reached. Other methods, so-called multispectral methods, make use of multiple eigenvectors to partition the data.

The method employed here is based upon the observation that the lower-order eigenvectors, after being sorted in a monotonically increasing fashion, exhibit a step-function-like property, as seen in Figure 2.2 [58]. Large, acute "gaps" in the eigenvector element values are used to delineate cluster boundaries while vertices whose associated eigenvector values reside on "plateaus"–consecutive elements showing little or no change–belong in clusters together. These gaps and plateaus will be used in the search for an ideal edge weight threshold. As previously mentioned, this method has the benefits of being unsupervised and operating on unweighted graphs, which allows a threshold to be applied at various correlations and an analysis of the resulting graph to be performed.

2.2 Application to thresholding

If one is to extract biological knowledge from large microarray data sets, it must be reasonably assured that the model accurately reflects true biological relationships. One way to do this is to determine the cutoff at which the natural separations in the network begin to appear. A threshold of zero results in a completely connected graph. As the threshold increases, the graph becomes less dense–some areas of the graph more than others. This is the nature of scale-free networks, which are frequently claimed to represent many biological graphs [59].

The methods implemented here are especially suited to this task because of the unsupervised nature of the algorithms. In contrast to supervised methods, it is not necessary to specify the number of clusters desired in advance, thereby imposing artificial limitations on the results. The results produced by the spectral methods described are completely

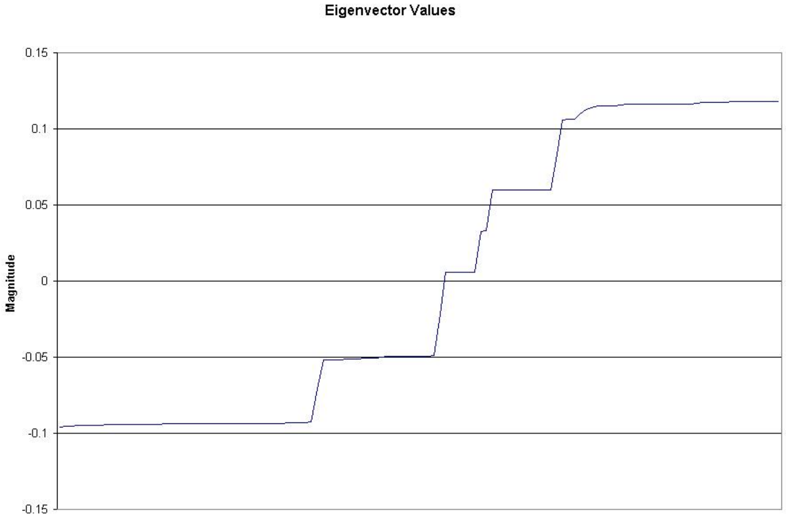


Figure 2.2: Eigenvector values associated with Lower-order eigenvalues show a step-like property when sorted.

data-based. Because only selected eigenvalues and eigenvectors are extracted, using sparse matrix operations, the methods are also not computationally- or memory-intensive.

Two possible thresholding methods are introduced and applied to the data sets described in Section 2.3.1. For both methods, the first step is to build a gene co-expression network, represented as a weighted graph with vertices representing genes and edge weights determined by the Pearson correlation coefficient between two genes. Iterating over thresholds from $0 \ldots 1$ in 0.01 increments, each graph is decomposed into its connected components. A subset of the spectrum is computed for the largest connected component at each threshold.

2.2.1 An eigenvalue-based approach

The first method is a purely eigenvalue-based approach, in which the smallest non-zero eigenvalue of the largest component is computed. A line chart of these eigenvalues at multiple thresholds is examined and inflection points identified. As the threshold increases, biological graphs tend to decrease in connectivity until reaching and holding steady at a low connectivity level before sharply rising. The increase in the magnitude of eigenvalues is likely due to the decrease in size of the largest connected component in the graph. It was observed that it is “easier” for smaller components to be well-connected than larger components.

The point at which the graph transitions from almost disconnected to rather well connected, as indicated by low and subsequently very high algebraic connectivity values, is likely to be a good point at which to threshold edge weights. One would expect this inflection point to be a suitable threshold because it allows for the best separation of subsets of vertices in the graph, without the graph becoming too small. This ensures that the network model represents only the strongest relationships and that so many edges have not been excluded that important vertices are likely to have become isolated.

The difference between each pair of smallest non-zero eigenvalues computed at consecutive thresholds is examined. Changes between these eigenvalues at consecutive thresholds usually remain “flat.” Relatively large changes at high or low thresholds indicate significant fluctuations in connectivity. Looking at this difference between thresholds allows one to identify stable periods during which connectivity remains relatively unchanged. This is an alternate approach to viewing the eigenvalue plot mentioned above, since this plot also identifies those changes from low to high connectivity values.

2.2.2 An eigenvector-based approach

A natural approach is to base the threshold on the results of an unsupervised spectral clustering. Both the smallest non-zero eigenvalue of the largest graph component, as well as its Fiedler vector, are computed. This vector is sorted in monotonically increasing order and transitions of the step-like function are identified.

To identify these gaps and plateaus, a sliding window is employed, which compares eigenvector members `windowsize` units apart. Iterating over each element of the eigenvector, the first point at which the difference between the element at position `windowsize` and the element at position `i` falls below some predetermined level serves as the starting position of a cluster. The end of the cluster is determined by the next point at which this difference exceeds the predetermined level. This iteration continues, identifying all possible clusters. Figure 2.3 demonstrates one iteration of the sliding window operation.

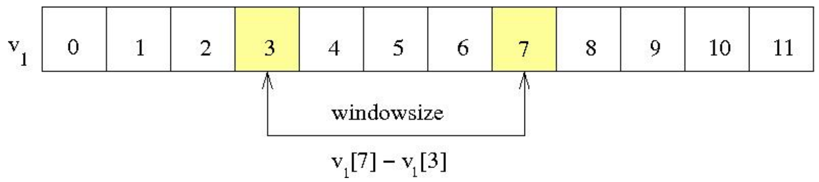


Figure 2.3: A sliding window is used to identify gaps in the sorted Fiedler vector.

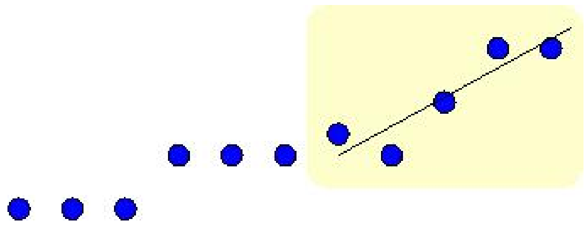


Figure 2.4: Identifying clusters using linear regression slope. In this example, eleven sample points are used to illustrate the positioning of the regression line and sliding window. The yellow box shows the current location of the sliding window and identifies the data points used in the regression line computation.

The sliding window is necessary because it is not sufficient to base cluster boundaries on changes between consecutive eigenvector elements. Transitions between plateaus may occur gradually. When this occurs, the difference between any two consecutive eigenvector members might fall below the threshold for a significant change, while elements `windowsize` distance apart surpass that threshold. In this case, elements at position `i` and `i+windowsize` should belong to two different clusters, but the gradual transition will cause the transition to be missed when considering only consecutive values.

Using a method based upon the eigenvector members themselves helps to avoid arbitrary choices when determining what is a significant difference between the elements at the start and end of the sliding window. A change is considered significant if it is greater than the median plus $\frac{1}{2}$ of a standard deviation of the differences between the smallest nonzero eigenvector values occurring `windowsize` distance apart. Also tested is another method that uses a similar sliding window technique and computes the regression line slope using points lying within the sliding window (see Figure 2.4). It has been suggested that this slope could be used to identify inflection points at which the curve shows a significant change [60]. The slope of the regression line is computed used formula 2.6. If this slope is greater than ϵ, then the difference is considered significant.

$$m = \frac{n\sum(xy) - \sum x \sum y}{n\sum(x^2) - (\sum x)^2} \tag{2.6}$$

The threshold value producing the largest number of clusters is selected. Using the value that produces the largest number of clusters allows for partitioning of the data points at the finest granularity, with the sliding window or regression line technique ensuring that the individual clusters are still well-defined. If two or more threshold values produce the maximum number of clusters, the lowest threshold at which this first occurs should be chosen, giving the largest cluster sizes. Choosing the higher thresholds would only serve to decrease cluster sizes, possibly excluding genes from the analysis that would have otherwise been in a cluster.

Notice that this method does not require the specification of parameters affecting clustering membership, nor does it force all genes into clusters. Genes whose associated eigenvector member fall within a transition area (i.e. do not occur within cluster boundaries) are not placed into clusters. Likewise, genes whose associated eigenvector elements occur in plateaus less than some designated size are excluded from the results.

2.2.3 Implementation

Datagen version 1.4a by Jon Scharff was used to compute pairwise Pearson correlation coefficients for each pair of genes sharing at least 10 observations. To avoid recomputing correlation values at each threshold, a weighted edge list was output with a threshold applied at the lowest correlation for which spectral properties were to be examined. This weighted edge list was then filtered as the threshold was stepped in 0.01 increments to the maximum, resulting in an unweighted graph.

At each step, the graph was decomposed into its connected components and a Perl script was employed to compute the Laplacian matrix of the largest connected component. For compatibility with MATLAB's sparse matrix functions, the Laplacian matrix **L** was output in a sparse format such that the coordinates for each nonzero matrix value are followed by the value at that location. For example,

```
i   j   -1
```

indicates a value of -1 in row i and column j.

This Laplacian matrix was then passed to MATLAB and converted to MATLAB's internal sparse format using the `spconvert` function. The two smallest eigenvalues and eigenvectors were then computed using the function `eigs`. Since the smallest eigenvalue is known to be 0, only the second eigenvalue and associated eigenvector were extracted for analysis. The script then employed both the previously described sliding window and regression slope techniques to identify gaps in this eigenvector.

2.3 Results

2.3.1 Data sets

Two well-studied biological data sets are examined and an appropriate threshold determined:

Yeast cell cycle
The Spellman, et al. *Saccharomyces cerevisiae* data set [61], in which the authors used microarray data to identify and study cell-cycle regulated genes.

GNF human
Homo sapiens microarray data collected by the Genomics Institute of the Novartis Research Foundation [12], to form a panel of baseline expression values over seventy-nine tissue types.

2.3.2 Graph structure

Examining properties of the graph at various thresholds can often provide some intuition into changes occurring at various correlation thresholds. In Figure 2.5, the density of the entire yeast cell cycle and GNF graphs at various thresholds is computed using the formula $d = \frac{e}{n(n-1)/2}$, where e is the number of edges in the graph and n is the number of vertices after isolated ones have been removed.

Both Figures 2.5a and 2.5b show a gradual decline and subsequent increase in graph density, although the densities are much lower in the yeast cell cycle data. It is clear that something is happening at the higher thresholds–above 0.92 in yeast and 0.94 in GNF. The data show that the number of isolated vertices increases rapidly as the thresholds become high, and that there are a core set of edges with high correlations that are present, resulting in higher densities. In choosing an appropriate threshold, one goal is to avoid analyzing graphs that are too dense, but avoid removing too many significant interactions. Lower densities also allow for a better partitioning of the data.

Since it is known that the multiplicity of the zero eigenvalue always represents the number of connected components in a graph, the number of connected components in each of the data sets at the same thresholds can be examined. Figure 2.6 shows that in both cases, the maximum number of connected components occurs at a threshold value of 0.84. This identifies which thresholds produce the largest natural separation in the data, though there is usually one large component (the one used in the spectral analysis) while the others are very small.

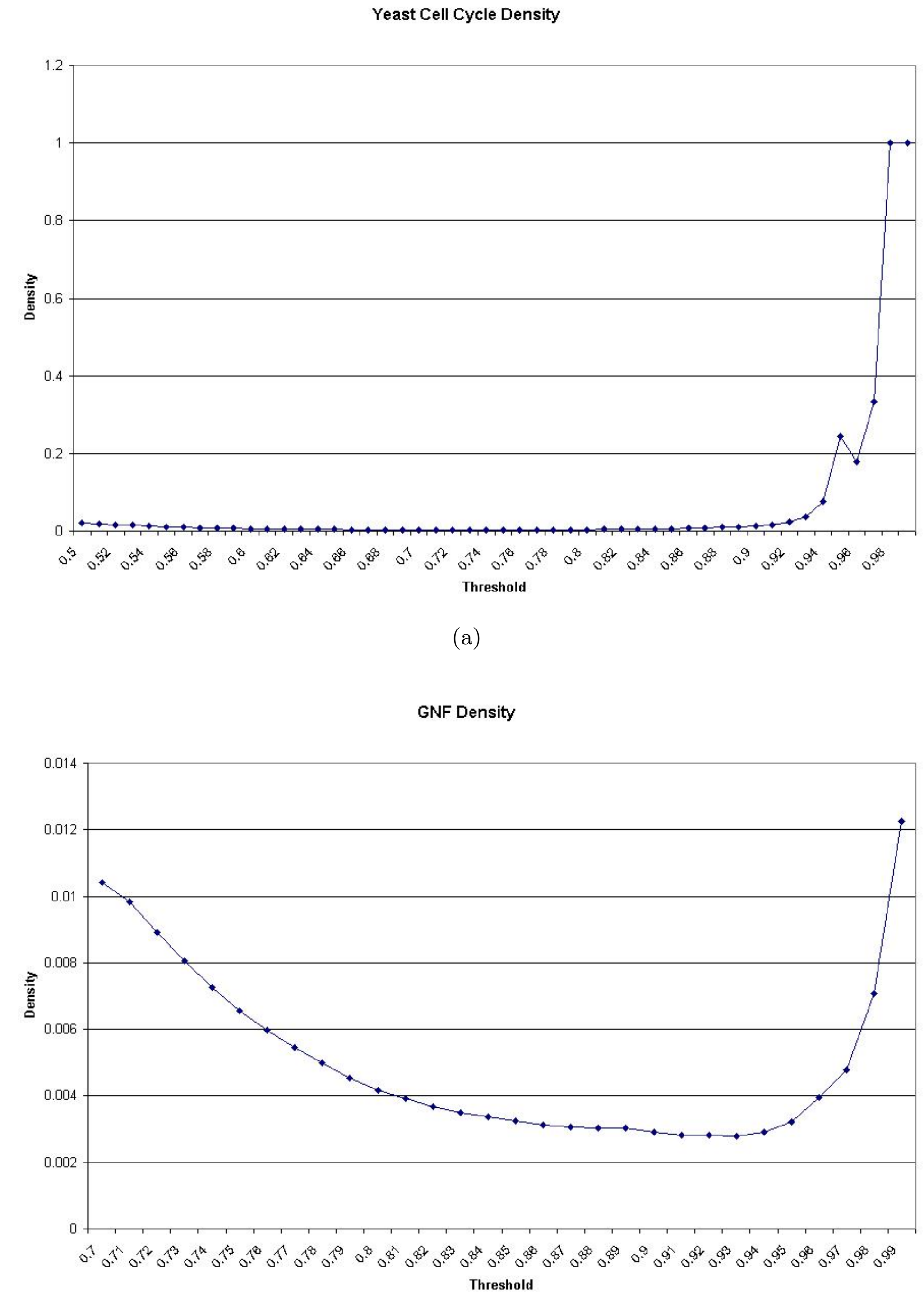
2.3.3 Spectral results

Figure 2.7a shows the smallest nonzero eigenvalue for both the Laplacian and normalized Laplacian of the yeast cell cycle data. This eigenvalue was calculated at thresholds from 0.50 to 0.99 of the Pearson correlation coefficient. By visual inspection, the smallest nonzero eigenvalue of the Laplacian matrix reaches the beginning of an inflection point at a threshold of about 0.77, although the eigenvalue is at a low, relatively stable level between about 0.77 and 0.87. By choosing the first point at which this minimal period is attained, a threshold of 0.77 results. Figure 2.7b shows the same information for the normalized Laplacian matrix. Here, the eigenvalue declines at a much slower and steadier pace, decreasing monotonically until the 0.83 threshold. Knowing that a low, nonzero Fiedler value indicates the presence of nearly disconnected components [58], and hence a good separation between clusters, an appropriate threshold choice here might be 0.82.

As previously mentioned, another way of visualizing the change in eigenvalues is to examine the difference between the smallest nonzero eigenvalue at consecutive thresholds. As the magnitude of the observed eigenvalues decreases, changes between subsequent thresholds will also decrease. Figure 2.8 shows that after a stable period, significant variation between consecutive cutoff values appears at thresholds above 0.87. These spikes are due to increases in the connectivity of the largest component. As one might expect, this coincides with the end of the stable period at 0.87 observed in Figure 2.7. This increase in the smallest eigenvalue, along with much smaller graph sizes and significantly higher densities, suggests that a good separation of clusters might not be possible at these higher thresholds. Therefore, a threshold of at most 0.86 would be most appropriate.

The number of clusters identified by both the basic sliding window difference method as well as the regression line slope method is indicated in Figure 2.9. In both cases, the number of clusters reaches a maximum at threshold 0.78.

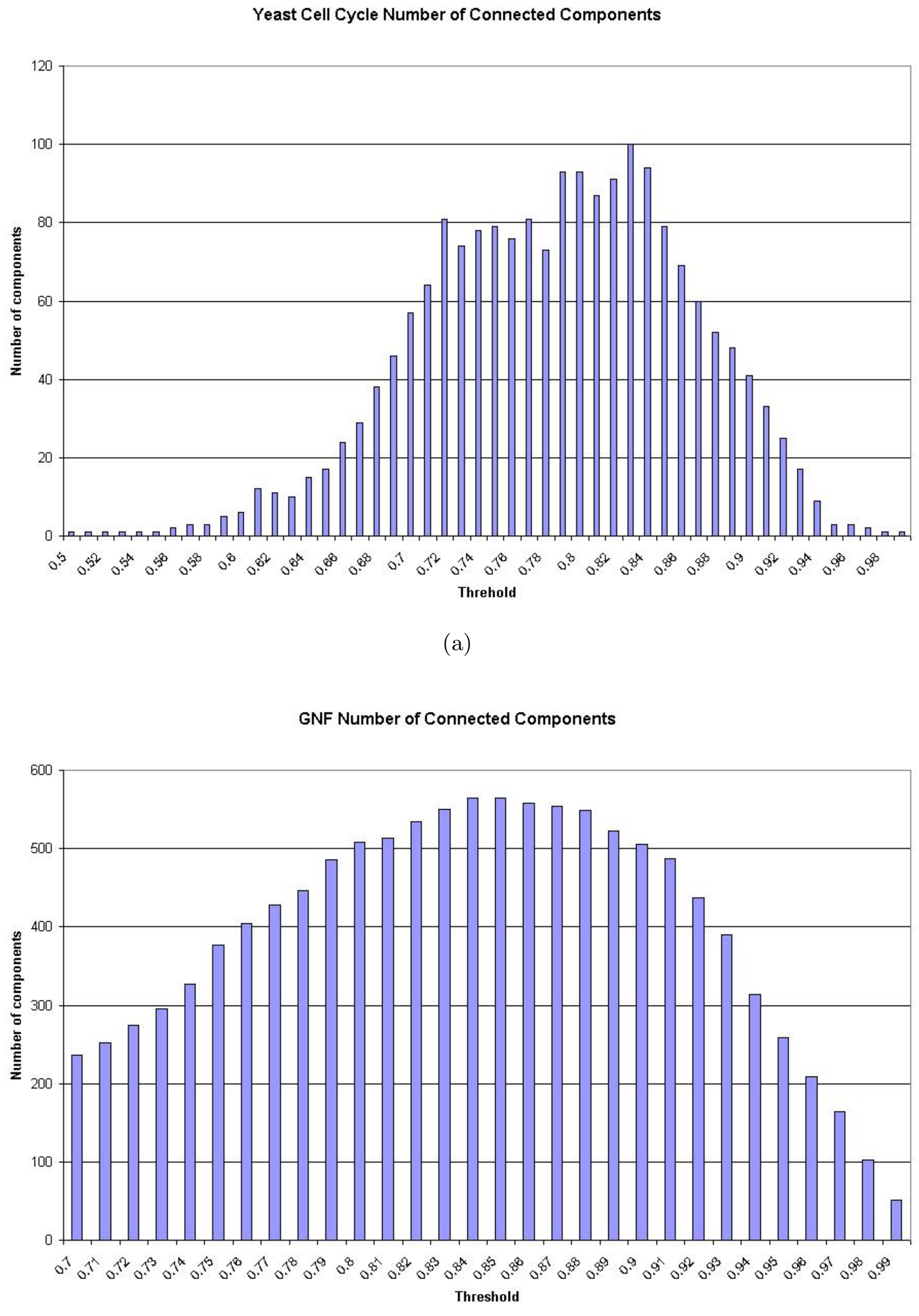
The smallest nonzero eigenvalues of both the Laplacian and normalized Laplacian of the GNF graph were calculated at thresholds between 0.70 and 0.99 and shown in Figure 2.10. In both cases, the eigenvalues show a steady period of low connectivity values up to threshold



(a)

(b)

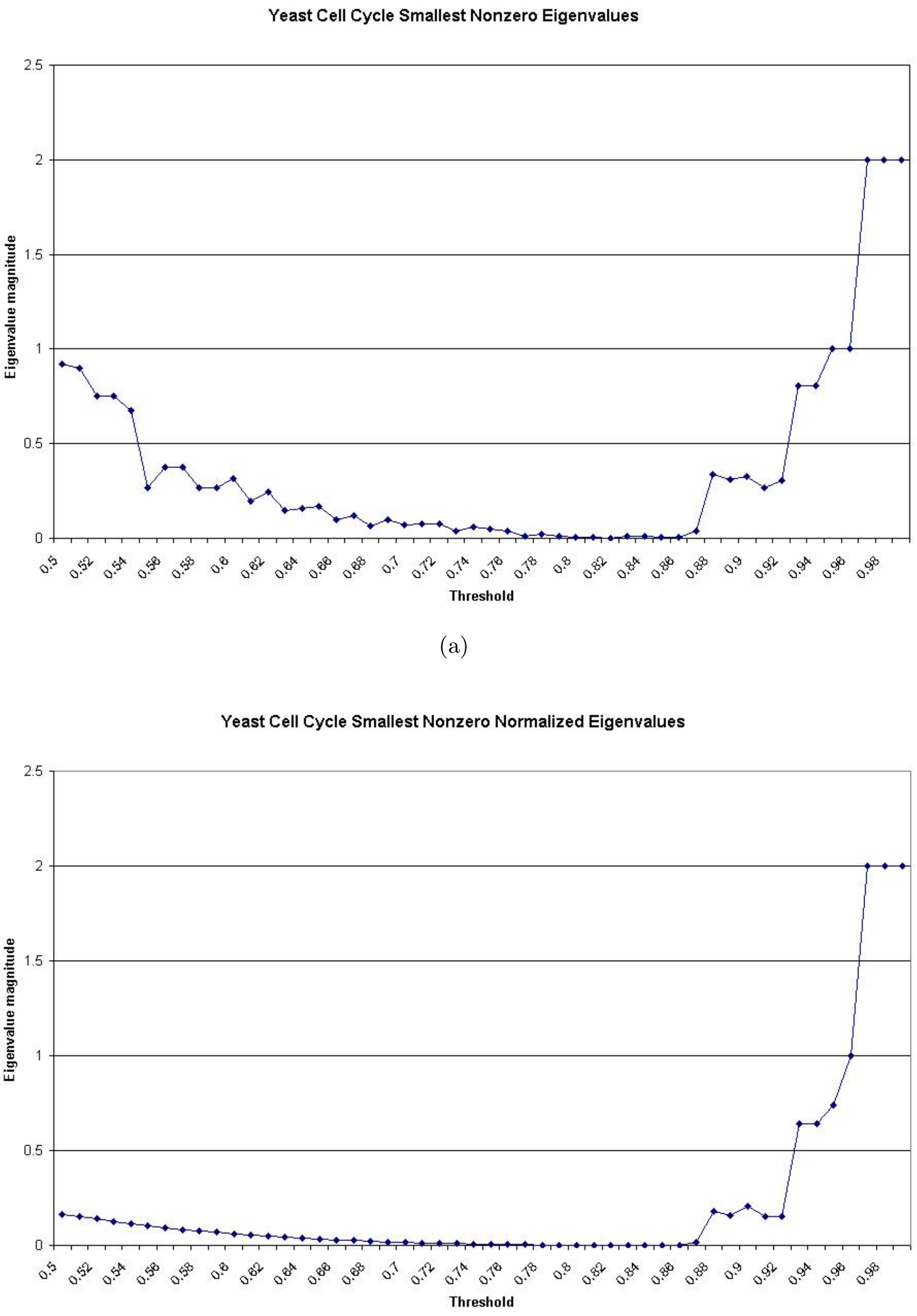
Figure 2.5: Density of the (a) yeast cell cycle and (b) GNF graphs at various thresholds. Isolated vertices have been removed.



(a)

(b)

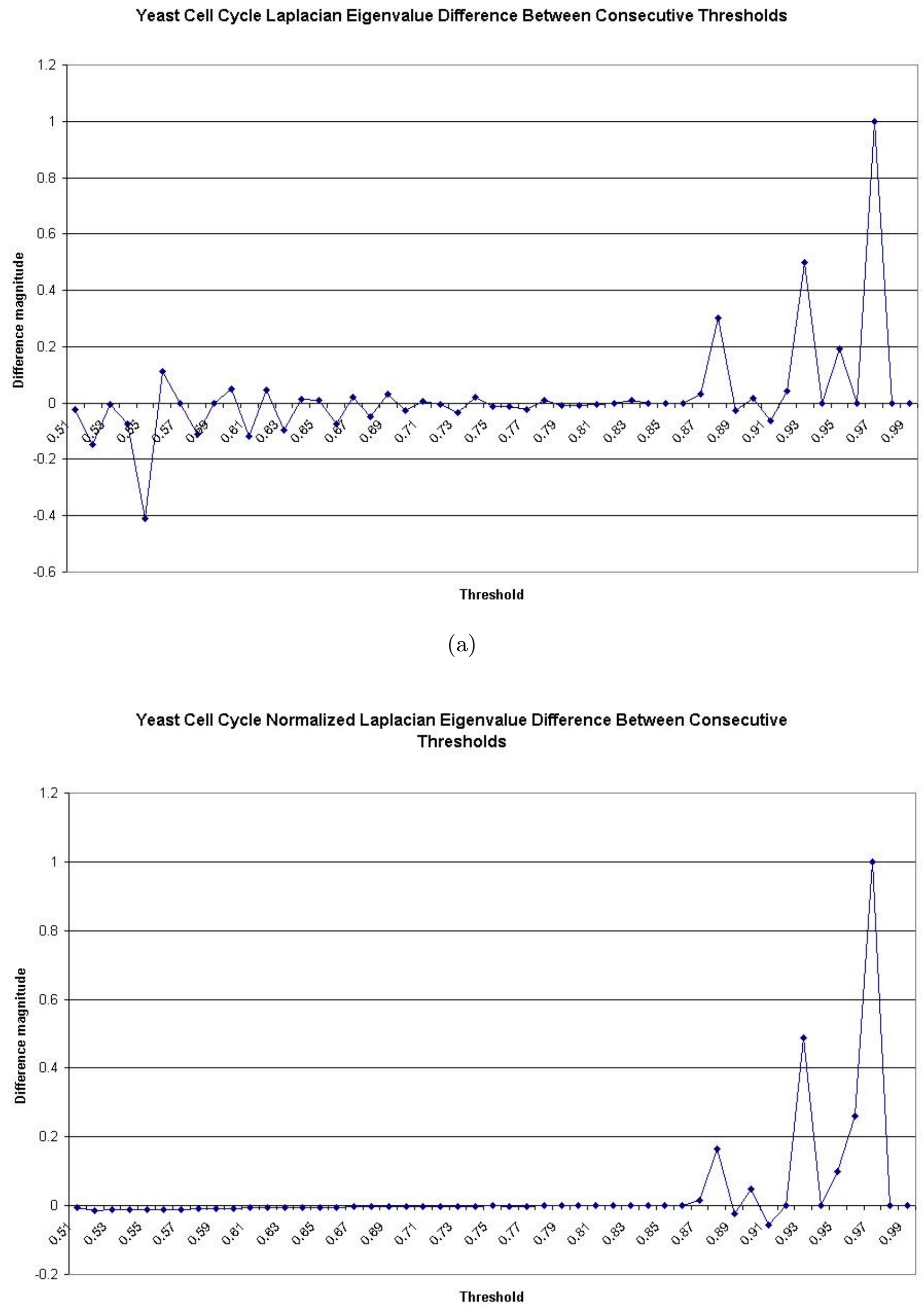
Figure 2.6: Number of connected components in the (a) yeast cell cycle and (b) GNF graphs at various thresholds.



(a)

(b)

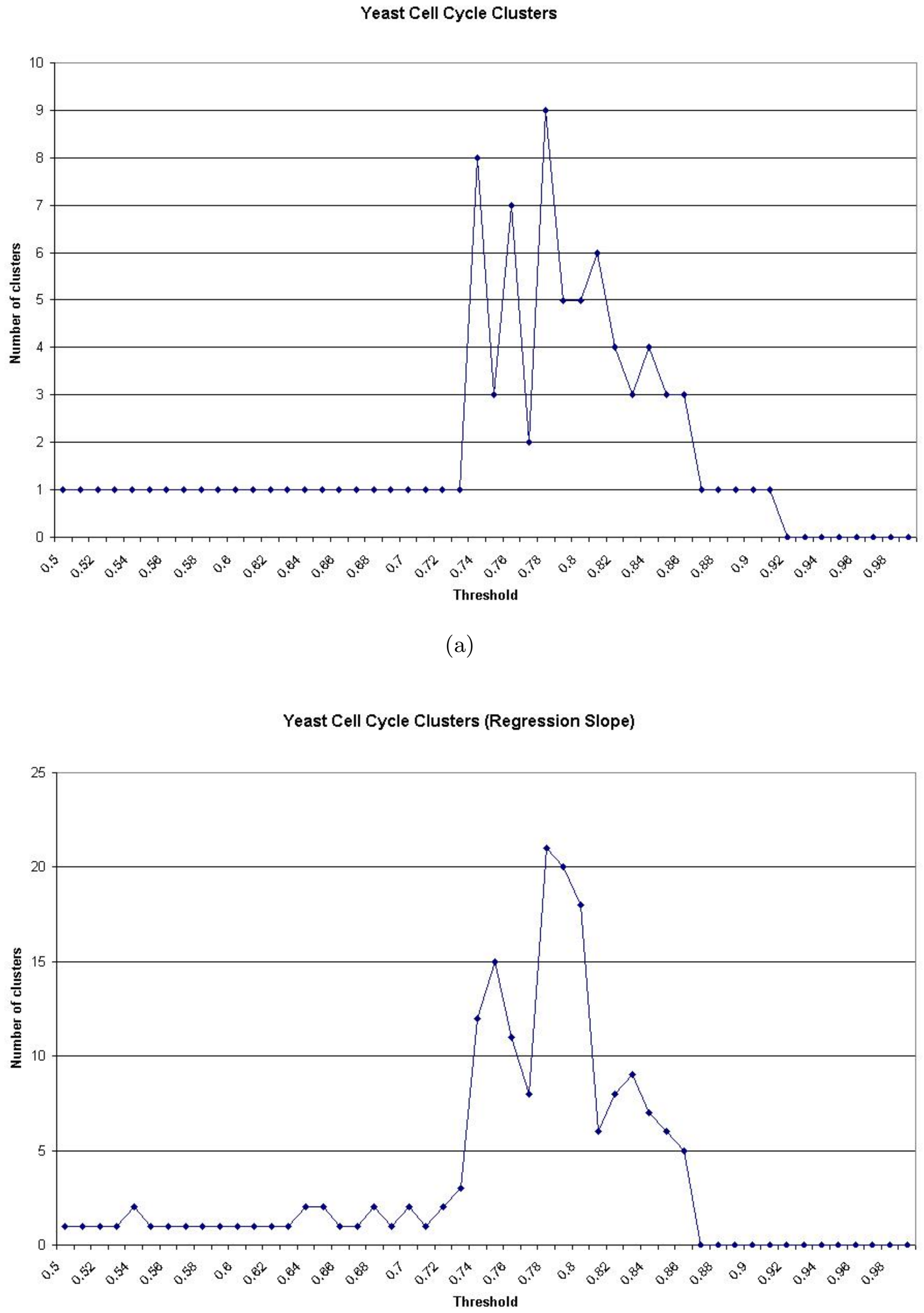
Figure 2.7: Yeast cell cycle smallest nonzero eigenvalues. (a) Plot of smallest nonzero eigenvalues of the Laplacian at various thresholds. (b) Smallest nonzero eigenvalues of the normalized Laplacian at various thresholds.



(a)

(b)

Figure 2.8: Difference between eigenvalues at consecutive thresholds for yeast cell cycle data. (a) Difference between smallest nonzero Laplacian eigenvalues at consecutive thresholds. (b) The same differences between eigenvalues computed on the normalized Laplacian.



(a)

(b)

Figure 2.9: Number of clusters identified in yeast cell cycle data. (a) Using the basic sliding window technique with a median- and standard deviation-based cluster identification method. (b) Using the linear regression slope method.

Table 2.1: Thresholding results produced by several different methods for both the yeast cell cycle and GNF baseline expression panel data sets.

Data set	λ_1	λ_1 norm	Sliding window	Regression slope	$p < 0.05$	1%	0.1%
Yeast	0.77	0.82	0.78	0.78	0.22	0.55	0.72
GNF	< 0.92	< 0.92	0.80	0.90	0.22	0.65	0.77

0.92, indicating a good separation of the data should be possible at the lower thresholds up to 0.92. The chart of the difference between eigenvalues at consecutive thresholds in Figure 2.11 shows the same variation seen in the yeast cell cycle data, beginning here at the 0.92 point.

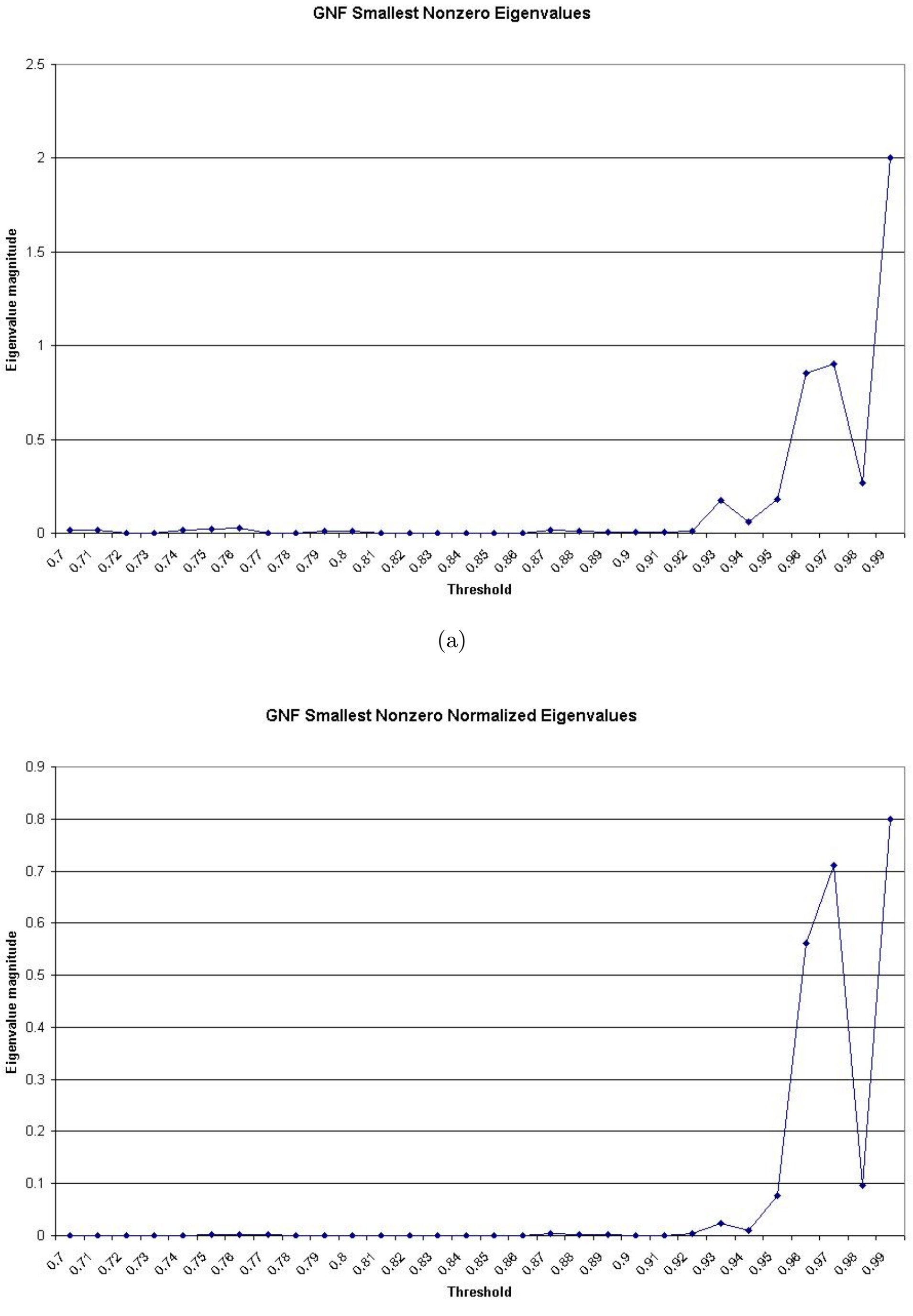
The number of clusters obtained in the GNF network using the basic sliding window technique (Figure 2.12a) and regression line slope method (Figure 2.12b) show maximums at 0.80 and 0.90, respectively. The sliding window method generally detects clusters at a finer granularity, with the regression line slope method recognizing only larger gaps, resulting in a higher threshold.

2.3.4 Comparison with other methods

Table 2.1 show thresholds selected by the spectral methods as well as the conventional p-value threshold, top one percent, and top one-tenth percent of correlations. In every case, the spectral methods selected a higher threshold than either the p-value method at the $\alpha < 0.05$ significance level or choosing the top one or one-tenth percent of correlations. While this results in edges that would otherwise be considered statistically-significant being excluded, it is often desirable to reduce the size of the network. For example, the GNF graph at a threshold of 0.22 has 106629395 edges on 22283 vertices. When analyzing the resulting gene expression network, the size of the graph becomes prohibitive in terms of memory and processing requirements.

2.4 Conclusions

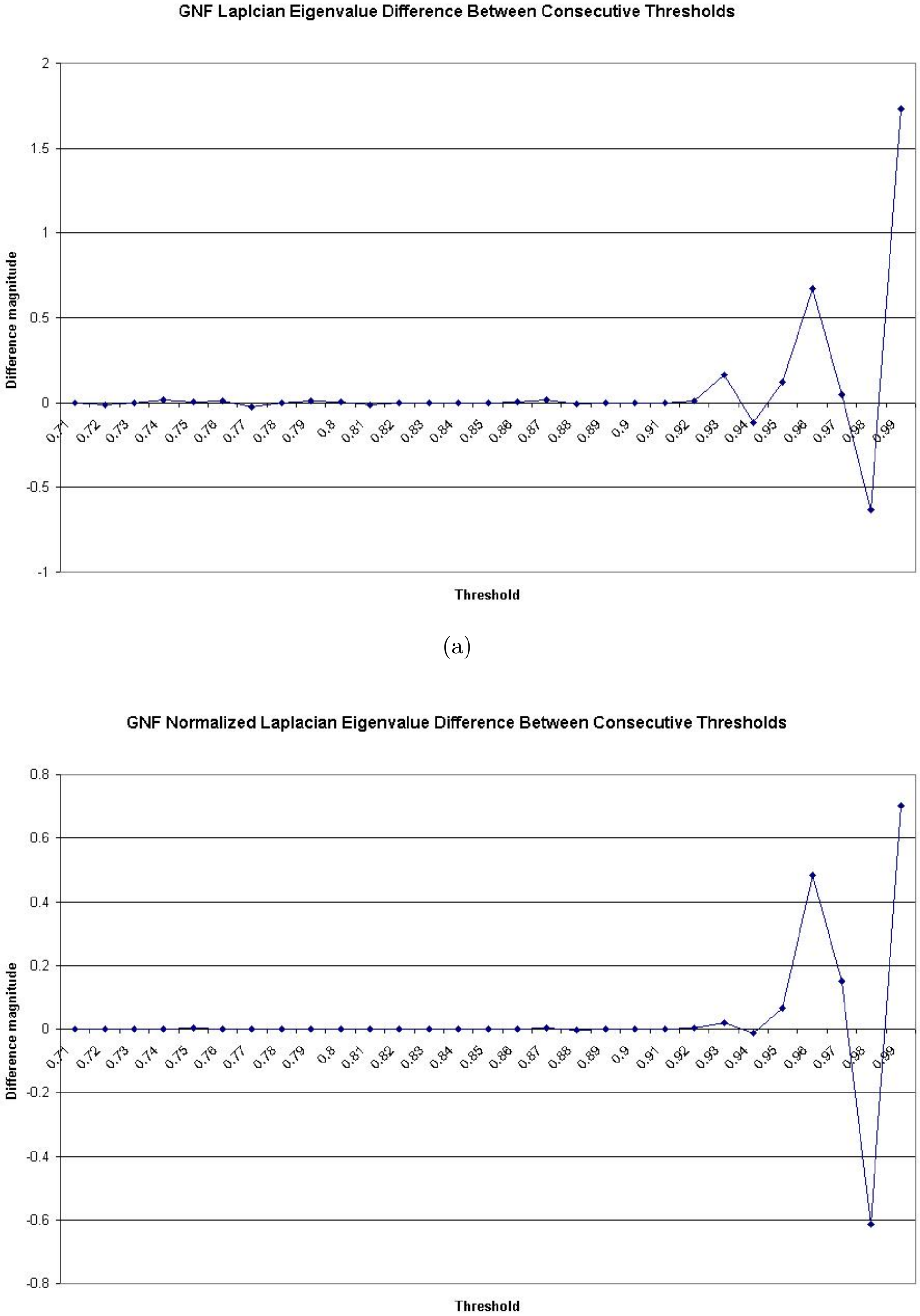
Both an eigenvalue- and eigenvector-based method to select thresholds for building gene coexpression networks from microarray data have been introduced. Both of these methods have been tested in various flavors, including using the normalized Laplacian for analysis, as well as both the basic sliding window and regression line slope approaches to identify clusters from the Fiedler vector. For two well-studied data sets, the spectral methods suggest a much more conservative threshold than either retaining all statistically-significant correlations ($p < 0.05$) or considering only the top one percent of all correlations. These spectral methods are more dependent upon the underlying data and graph structure than the p-value or correlation-based methods, which rely exclusively upon sample size and correlation distribution to formulate a threshold, respectively.



(a)

(b)

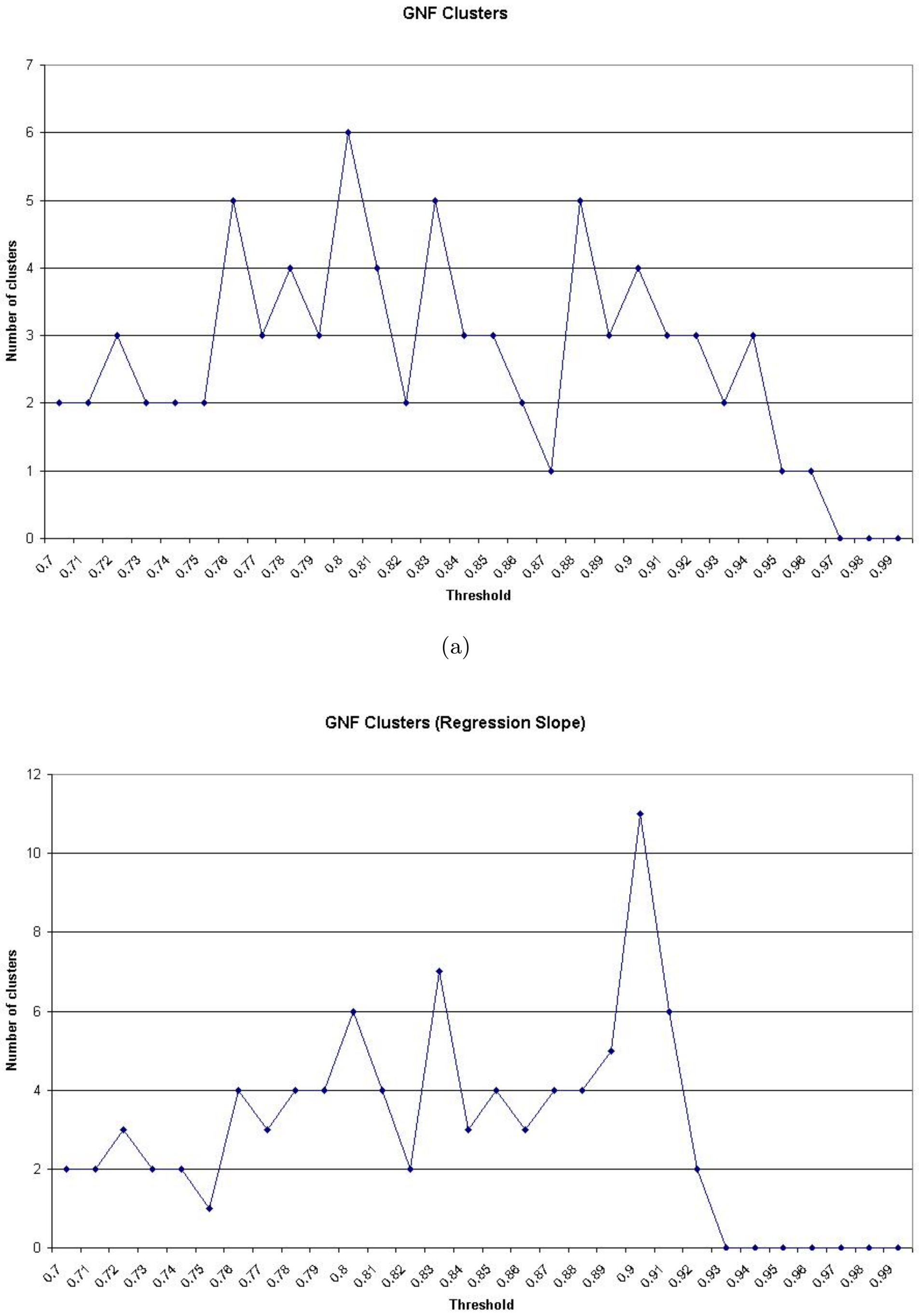
Figure 2.10: GNF smallest nonzero eigenvalues. (a) Plot of smallest nonzero eigenvalues of the Laplacian at various thresholds. (b) Smallest nonzero eigenvalues of the normalized Laplacian at various thresholds.



(a)

(b)

Figure 2.11: Difference between eigenvalues at consecutive thresholds for GNF data. (a) Difference between smallest nonzero Laplacian eigenvalues at consecutive thresholds. (b) The same differences between eigenvalues computed on the normalized Laplacian.



(a)

(b)

Figure 2.12: Number of clusters identified in GNF data. (a) Using the basic sliding window technique with a median- and standard deviation-based cluster identification method. (b) Using the linear regression slope method.

2.4.1 Future work

Since there has been much previous work on spectral partitioning and clustering, an interesting extension to this work would be to compare thresholding results based upon other spectral clustering methods. The method implemented here was simple, being based upon a single eigenvalue or eigenvector. An advanced method should be able to perform a more accurate clustering, increasing the confidence in the threshold chosen to give the best separation of the data points.

Chapter 3

Clique Profiles and Similarity Metrics

3.1 Introduction

One of the key steps in constructing gene coexpression networks is the computation of some similarity of the expression profile between each pair of genes. With edge weights based solely upon this metric, changing the method used can obviously affect which relationships are determined to be biologically significant. Graph structure, including maximum and maximal clique properties, might also then be affected. The degree, correlation, and clique profiles for several biological data sets are examined to identify differences between similarity computation methods.

The question then arises of whether there are "normal" or "usual" degree or clique profiles that one would expect to result from analysis on biological data. Having this information available can help identify data sets that deviate from the norm, indicating that the reason for this deviation must be identified.

This chapter examines some of the common properties in similarity and structural profiles for four different methods. Also presented is a characterization of these properties for a collection of biological data sets and scale-free synthetic networks generated using the Barabasi-Albert method [62].

3.2 Computation and comparison of similarities and clique-based properties

Using several biological data sets from two separate gene expression studies, pairwise similarities were computed for each pair of gene transcriptions. For each data set, similarity results for the methods discussed below were compared on four different criteria: similarity distribution, degree structure, maximum clique size, and clique profile.

3.2.1 Similarity computation

Four different methods were used to compute gene-gene similarities. Correlations computed on a small number of data values are inherently less reliable and values based upon a smaller number of observations often tend toward extremes. Two genes need only move in the same direction, either up or down, over two conditions to produce a perfect correlation.

Pearson and Spearman correlation values computed on fewer than ten pairwise-complete observations were not included in the analysis. While variables with fewer than ten observations cannot result in a reliable correlation value, note that additional correlations might be removed based upon the number of pairwise-complete data items. To help equalize the effect of excluding these correlations in the comparisons, any transcript with fewer than ten present values was removed from all analyses.

Pearson

One of the most well-known and widely used measures of similarity in gene coexpression studies is the Pearson product-moment correlation coefficient, which was discussed in Chapter 1 and whose formula was given in Equation 1.1. Pearson's r represents a measure of linear relationship between two normally distributed variables in the range $-1 \ldots 1$ with a correlation of zero indicating no relationship. Positive and negative correlations result from variables moving in the same and opposite directions over some set of samples, respectively.

Spearman

Spearman's rank-based correlation coefficient ρ presents an option for similarity computation that is useful for data likely to contain larger numbers of outliers. From the surface, Spearman correlation values exhibit similar properties as Pearson correlations, with values lying in the same range. The non-parametric nature of this method, however, makes this correlation coefficient desirable for use with variables not drawn from a normal distribution or for which the theoretical distribution is not known. The formula for computing Spearman correlation values, shown in Equation 3.1 is given in [14].

$$\rho = 1 - \frac{6 \sum D^2}{n(n^2 - 1)} \tag{3.1}$$

where D is the rank difference between corresponding data items. For example, suppose that for two random variables X and Y the rank of value x_i and y_i are 4 and 1, respectively. In this case, $D = 3$.

Shrinkage

The shrinkage-based estimate of the correlation matrix described in [63] presents an approach that is more suitable for the small sample sizes and large number of variables usually found in high throughput biological data. Schäfer and Strimmer argue that the use of the standard covariance in such "small n, large p" situations is inappropriate and introduce a new estimator with better mean squared error than traditional approaches. The shrinkage-based correlation estimator given in [63] is seen in Equation 3.2.

$$r^*_{ij} = \begin{cases} 1 & \text{if } i = j \\ r_{ij} \min(1, \max(0, 1 - \hat{\lambda}^*)) & \text{if } i \neq j \end{cases} \tag{3.2}$$

where r_{ij} is the empirical correlation and

$$\hat{\lambda}^* = \frac{\sum_{i \neq j} \widehat{Var}(r_{ij})}{\sum_{i \neq j} r^2_{ij}}$$

Computation of the unbiased variance estimate $\widehat{Var}$ is described in [63].

Mutual information

The use of the mutual information measure has recently become popular in biological studies [64, 65]. Mutual information (MI) can be considered a measure of dependence between two variables. As such, MI can indicate the presence of nonlinear relationships between genes. Unlike previously mentioned approaches, all MI values are nonnegative and do not fall within a predefined range.

Mutual information is described in [64] in terms of relative entropy. The authors note that entropy is a measure of the information contained within a gene's expression profile, for example. The entropy $H(A)$ of a variable A with some number of observations is given in Equation 3.3. Note that the range of values for the variable A has been divided into n subintervals $x_1 \ldots x_n$ into which the values have been binned.

$$H(A) = -\sum_{i=1}^{n} p(x_i) \log_2 p(x_i) \tag{3.3}$$

where $p(x_i)$ is the proportion of values lying in the interval x_i.

Having defined entropy, mutual information can be computed as in Equation 3.4.

$$MI(A, B) = H(A) + H(B) - H(A, B) \tag{3.4}$$

where the conditional entropy $H(A, B)$ can be calculated as in Equation 3.3 while using the joint entropy $p(x, y)$ and summing over both x and y. A lower mutual information value translates into a higher degree of independence between two variables while larger MI values indicate a non-random relationship [64].

A comprehensive comparison of mutual information with Euclidean distance and Pearson correlation is presented in [66]. While the authors mention that other studies have found a high degree of concordance between these approaches, their results showed that MI produced better clustering results.

3.2.2 Implementations

The stand-alone program "Datagen" version 1.4a by Jon Scharff was employed to compute both Pearson and Spearman correlation coefficients. Shrinkage and mutual information values were computed using the R statistical package version 2.6.1 with various additional packages. The shrinkage correlation matrix was computed using the Corpcor package version 1.4.7 [67]. Missing values are not a problem per se when using the shrinkage approach, since the `cor.shrink` function requires all observations to be present. For this reason, the Bioconductor EMV package version 1.3.1 [68] was used to impute missing values using the k nearest neighbors (KNN) method. The `mutualInfo` function within the Bioconductor package bioDist version 1.10.0 [69] was used to compute pairwise MI values.

3.2.3 Clique-based methods

After the computation of gene-gene similarities using each of the four methods, graphs were constructed using the methods described in Chapter 1. By applying a high pass threshold filter, an unweighted graph was created from only those edges surpassing the selected threshold. The maximum clique size was found and maximal cliques were enumerated from graphs constructed at various thresholds. Due to variations in density and computational resources required for individual graphs, the thresholds used in the analyses varied depending upon both data set and choice of similarity metric.

3.2.4 Scale-free comparisons

Although biological networks are often poorly modeled by graphs on synthetic data, it is often observed that graphs on biological data exhibit the property of scale-freeness. Scale-free networks have the property that their vertex degree distribution follows the power law $P(K) \sim k^{-\gamma}$. Scale-free networks were discussed extensively and properties such as component size and average path length were examined in [62].

Various properties of scale-free networks are compared with the gene expression data sets described in Section 3.3.1. Scale-free networks were constructed using the Barabasi-Albert (BA) method described in [62]. This method uses a growth and preferential attachment approach to construct a graph with the desired degree distribution. As implemented here, the algorithm begins with a small number of nodes m_0 that are connected in a star. A new vertex is added at each time step and is connected to $m = m_0$ of the nodes already in the graph with probability

$$\Pi(k_i) = \frac{k_i}{\sum_j k_j} \tag{3.5}$$

where k_i is the degree of vertex i. Vertices are selected at random for possible connection to the new node until m connections have been made. Constructing graphs using this preferential attachment approach allows graphs with higher degree to accumulate edges faster, resulting in the hub structures seen in scale-free networks.

It is also noted in [62] that the number of vertices and edges in the resulting graph will be $t + m_0$ and mt, respectively, where t is the number of time steps. Therefore, with $m = m_0$, the mechanism for controlling density is the variation of m_0. Scale free graphs were generated here to have similar density and size as the biological graphs under study.

3.3 Results

3.3.1 Data sets

Correlation and degree distributions of two biological data sets were examined: seasonal allergic rhinitis and low dose ionizing radiation data. Clique profiles and maximum clique sizes of these two data sets were also computed at various thresholds, as described in Section 3.3.3.

Seasonal allergic rhinitis
In this allergy data set, studied further in Chapter 4, gene expression was measured in twenty allergic patients outside of allergy season before and again after allergen challenge on Illumina Human WG-6 BeadChips [70]. Twelve arrays passed quality control in the control group and nineteen in the allergen-stimulated group.

Low dose ionizing radiation
This data was collected for a study of the effects of low dose ionizing radiation in spleen

tissue, and was reported in [1]. Twenty samples from both control and radiation-exposed groups were analyzed on custom cDNA microarrays from six standard inbred *Mus musculus* strains.

3.3.2 Similarity profiles

Over all data sets, Pearson correlations fell within the range $-0.99 \ldots 1$ while Spearman coefficients consumed the entire range of $-1 \ldots 1$. Shrinkage values ranged from -0.62 to 0.62. Mutual information values, which are not constrained by the usual $1 \ldots 1$ range, spanned 0.1 to 2.5. This range of MI levels is similar to the limits of 0.1 and 2.5 observed in [64].

Figure 3.1 shows the distribution of similarity values for each method and data set. It is important to note that the amplitude of the curve for each data set will vary depending upon the number of the correlate pairs examined. It is the shape of the distribution that is interesting to compare for different methods. Because the distribution of mutual information values did not vary with each increment of one-hundredth as did the other methods, MI scores were binned at the tenths level.

All distributions exhibited a bell-shaped curve with the exception of the shrinkage correlations computed on the stimulated allergy data set, which had no obvious tails. Shrinkage values for all data sets fell within a smaller range than other methods, with many more correlations centered around zero.

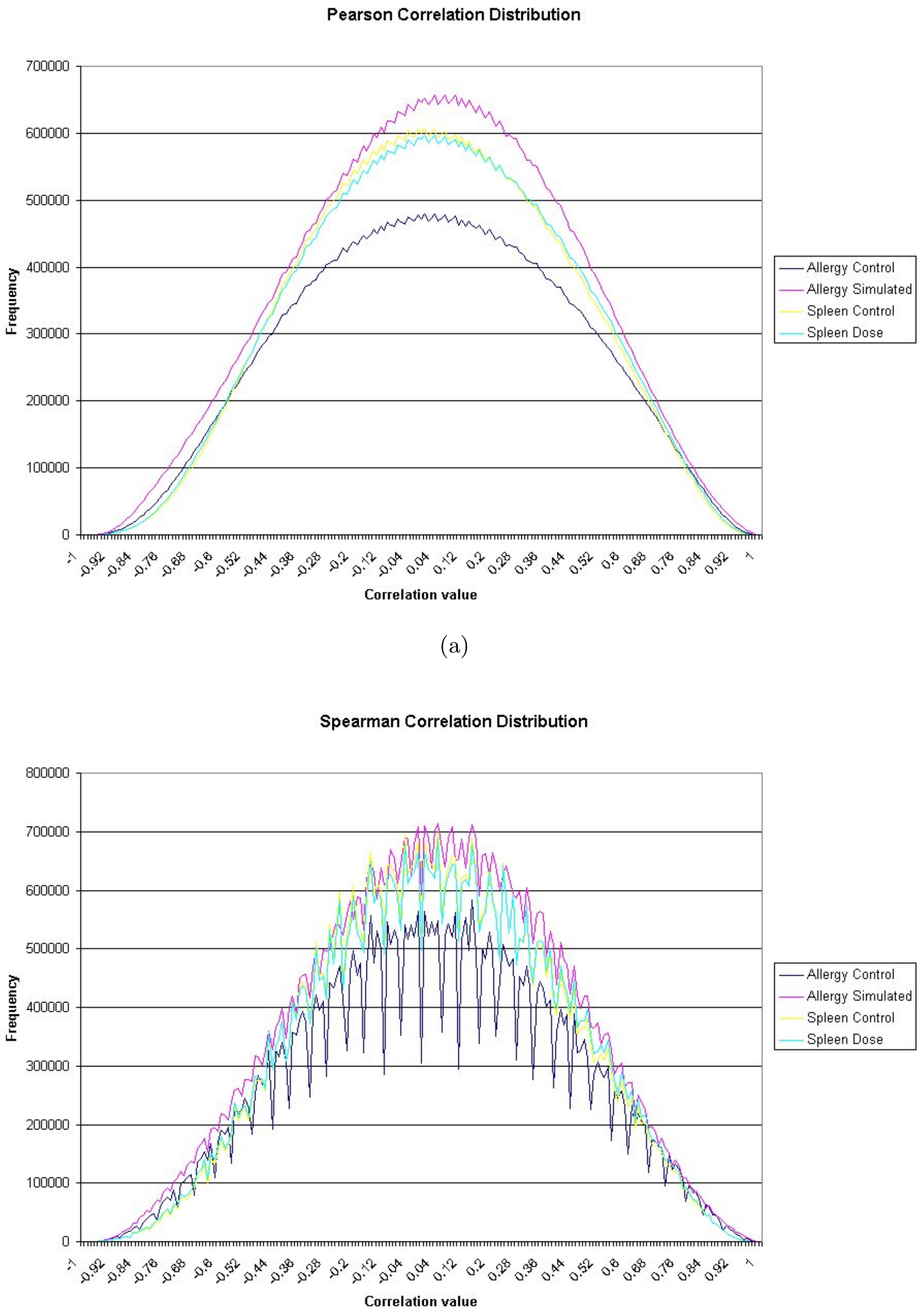
Degree profiles were extracted from graphs produced from each similarity method at thresholds typically used in clique-based analyses. A threshold of 0.85 was applied to both Pearson and Spearman correlation. Shrinkage values varied widely depending upon the data set examined, so a threshold was chosen for each data set to attempt to equalize the maximum degree within each graph. Graphs from mutual information similarities were constructed at the 1.95 MI threshold. Figure 3.2 shows the degree profile produced by similarities from each method.

Again, the degree distribution of graphs produced from each similarity method exhibited a similar shape over all biological data sets. When comparing degree profiles produced by the various methods, mutual information produced several genes of high degree with the degree falling sharply when examining genes from the most to the least connected.

3.3.3 Clique profiles

Maximum clique sizes were computed for graphs from each biological data set using the four previously described similarity methods. A lower bound on the threshold for each method and data set was applied to allow for maximum clique sizes to be found quickly. Prior experience shows that these thresholds also equate to graph of size such that maximal cliques can also be enumerated. The lower bound selected for Pearson and Spearman methods was 0.85. Shrinkage values varied widely, so 0.37, 0.56, 0.36, and 0.40 were chosen as the smallest thresholds for allergy control, allergy stimulated, spleen control, and spleen dose data respectively. The lowest threshold examined for mutual information values was 1.95.

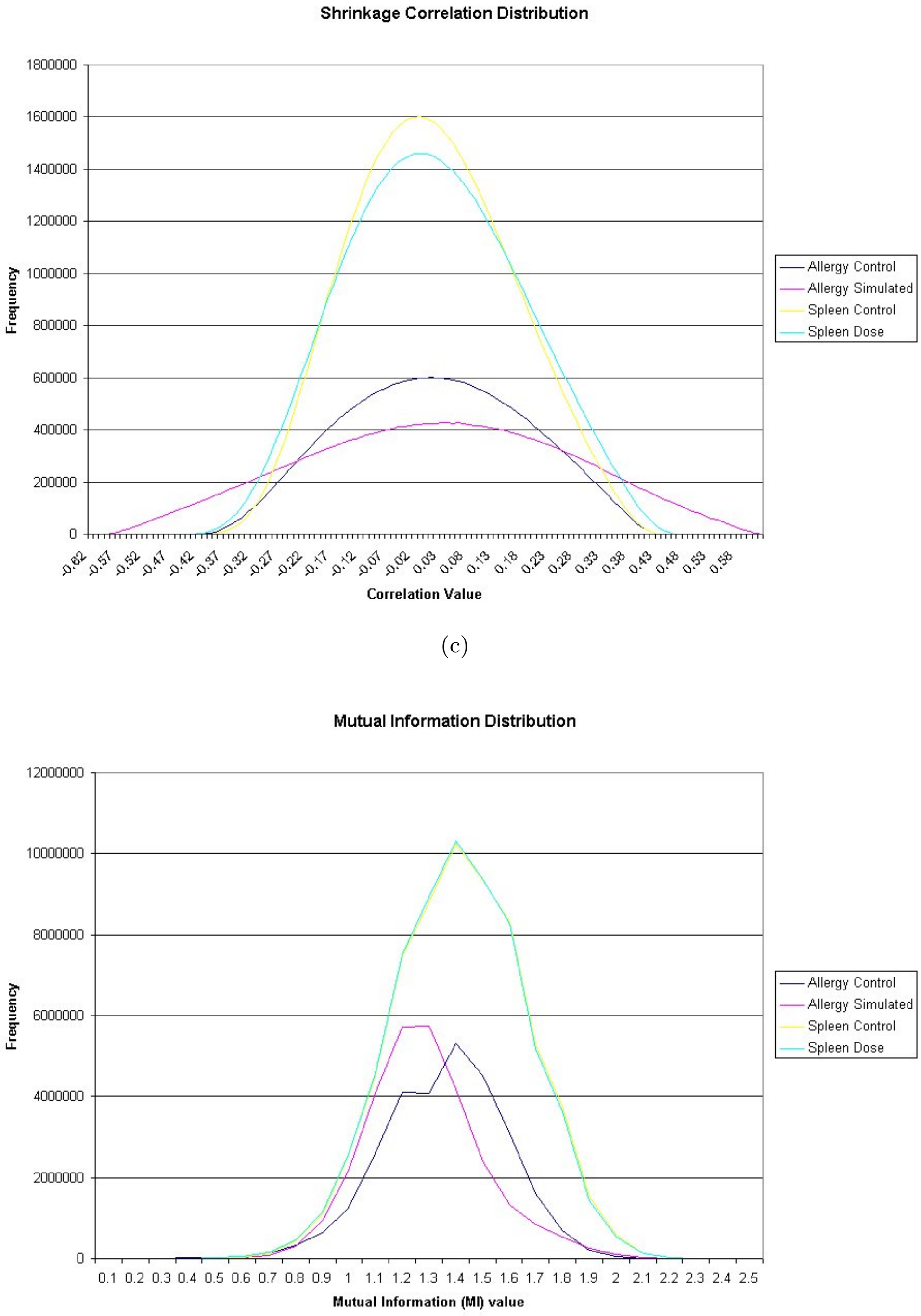
Figure 3.3 illustrates that maximum clique sizes for each data set and similarity method were found to be approximately linearly decreasing as the threshold increased. There was a high degree of similarity observed in the maximum clique sizes on graphs from Pearson



(a)

(b)

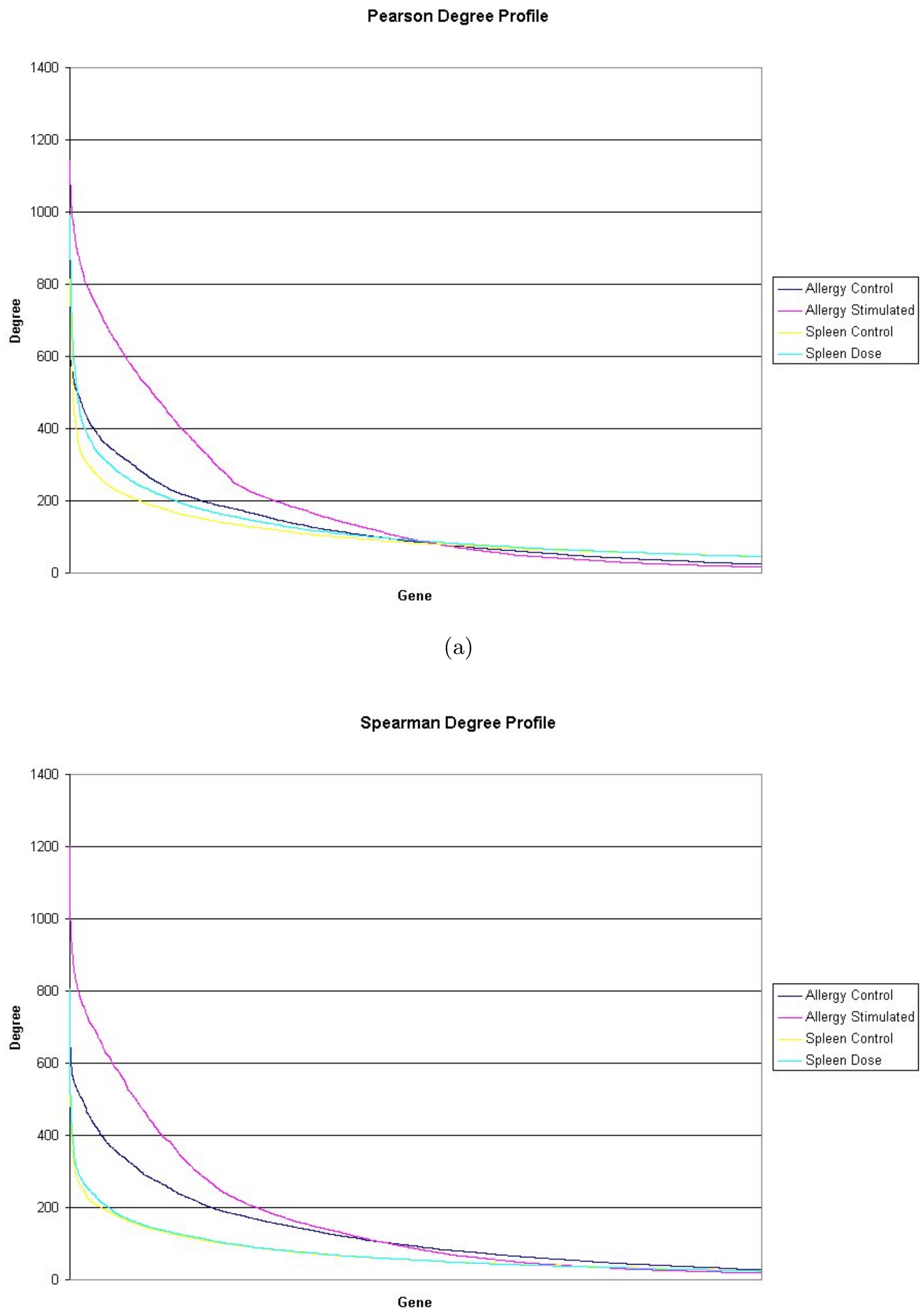
Figure 3.1: Similarity distributions for graphs on allergy and spleen data using various similarity methods. (a) Pearson, (b) Spearman, (c) Shrinkage, (d) Mutual Information.



(c)

(d)

Figure 3.1: Continued.

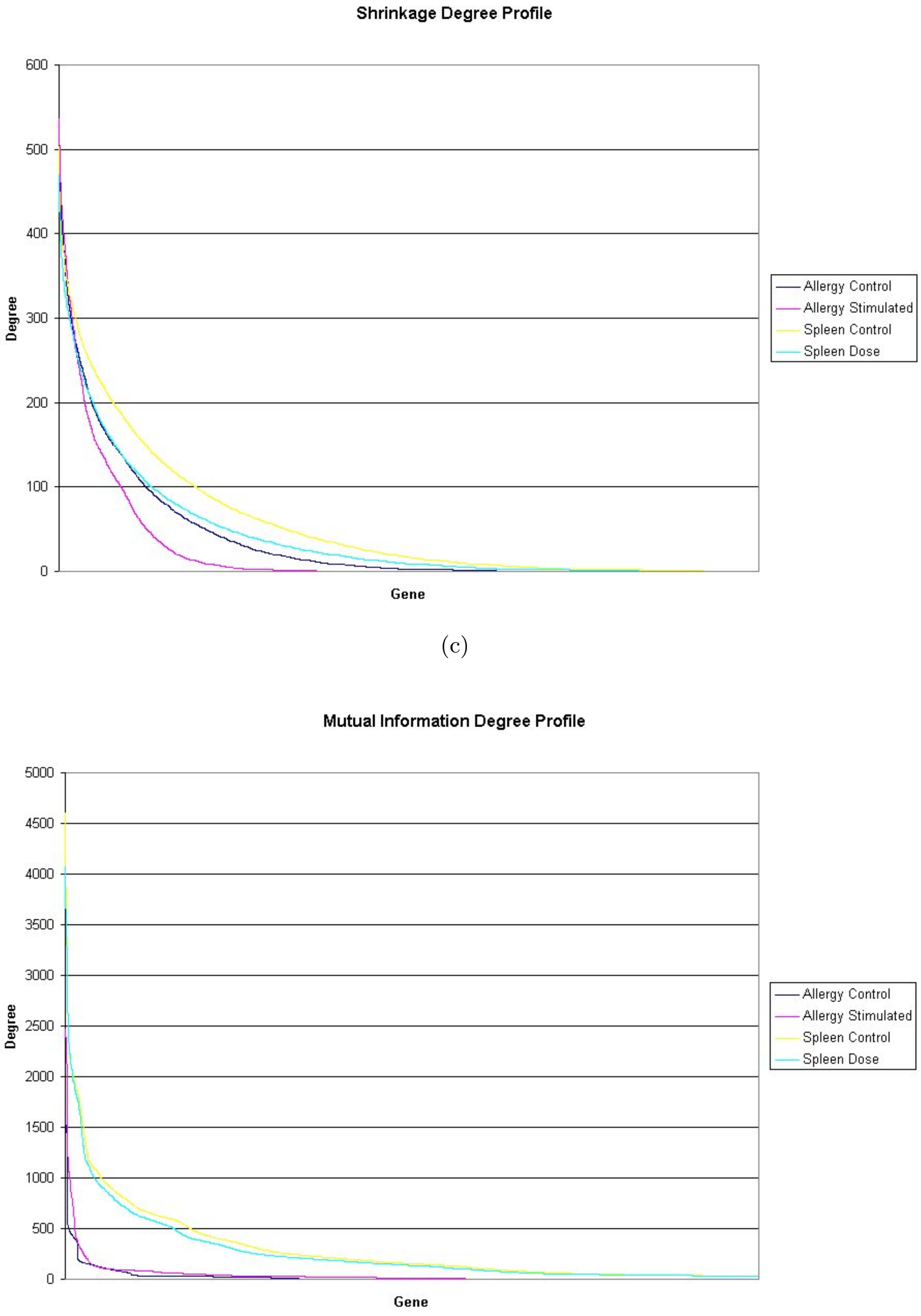


(a)

(b)

Figure 3.2: Degree profiles for graphs on allergy and spleen data using various similarity methods. (a) Pearson, (b) Spearman, (c) Shrinkage, (d) Mutual Information

.



(c)

(d)

Figure 3.2: Continued.

Table 3.1: Summary of scale-free graphs generating using the Barabasi-Albert method.

Graph	Vertices	Edges	Density
BA-8000-12	8000	95856	0.30%
BA-8000-20	8000	159600	0.50%
BA-10000-10	10000	99900	0.20%
BA-10000-15	10000	149775	0.30%
BA-15000-15	15000	224775	0.20%
BA-15000-20	15000	299600	0.27%
BA-20000-20	20000	399600	0.20%
BA-20000-25	20000	499375	0.25%

and Spearman correlations. Sizes on graphs constructed from shrinkage-based correlations in Figure 3.3c showed a considerable difference between the stimulated allergy data and the rest of the data sets. Mutual information maximum clique sizes in Figure 3.3d exhibited a step-like property, which was found to be due to the fact that some thresholds did not exclude any additional edges.

Maximal cliques were enumerated at various thresholds again using the lower threshold bounds defined above. Figure 3.4 shows the number of maximal cliques of each size for all of the similarity methods. Because of the scale of the values involved, the y axis of all graphs is log scale. It is obvious that maximal clique distributions on these biological graphs have a distinct shape, which is approximately bell-shaped in a non-log scale. Many of the distributions are well-skewed in the positive or negative direction, indicating that more small or large cliques are present, respectively.

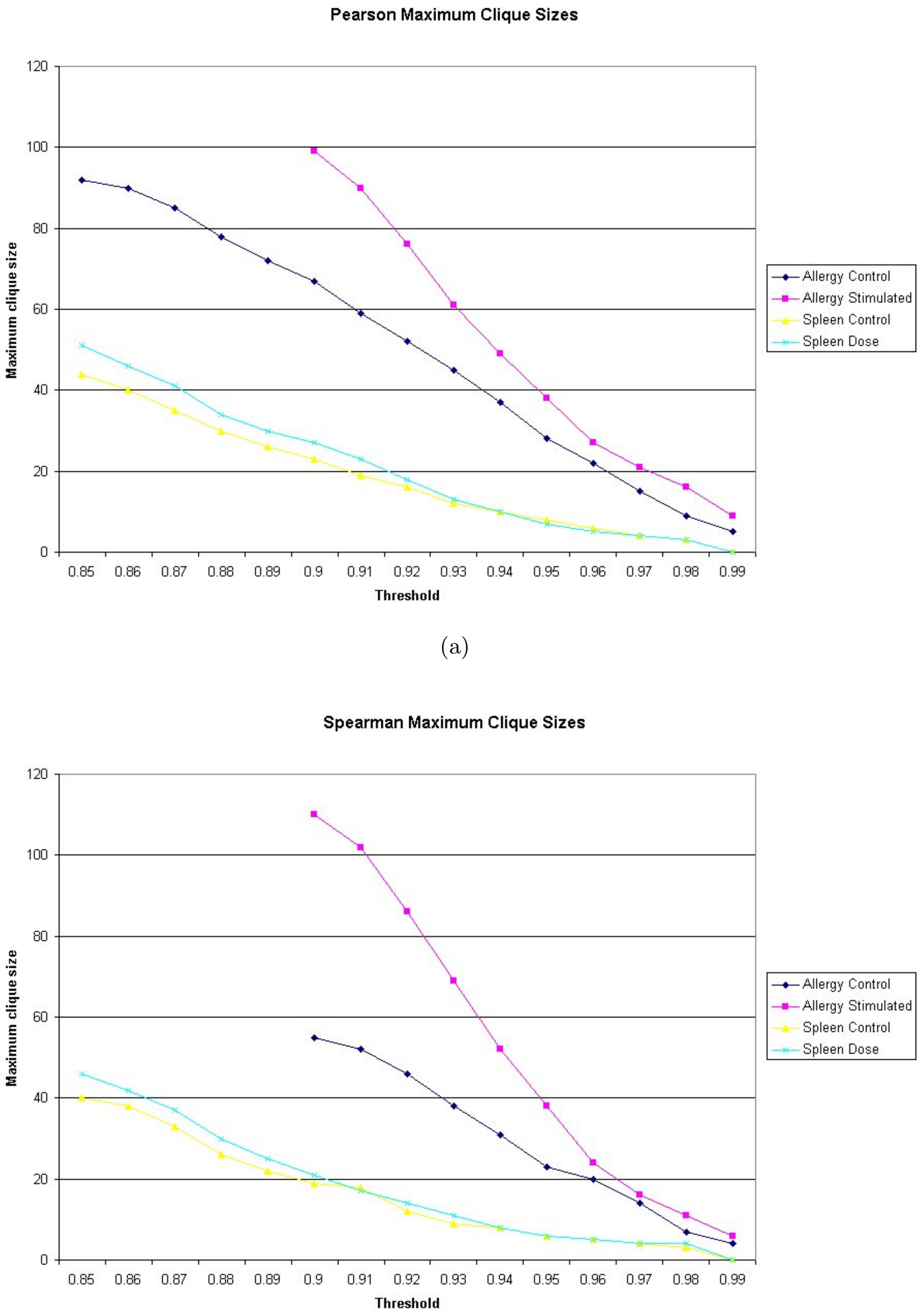
3.3.4 Scale-free comparisons

Scale-free networks of similar size and density as biological graphs on Pearson and Spearman correlations at the lowest thresholds studied were constructed. The Barabasi-Albert method described in Section 3.2.4 was employed to ensure the scale-free property of the graphs' degree distribution. Table 3.1 shows a summary of the graphs constructed. Graphs were named with the number of vertices along with the selected size for the parameter m_0.

The BA-8000-12 and BA-8000-20 graphs most resemble the allergy control and stimulated graphs with a Pearson correlation threshold applied at 0.90. The allergy graphs have size 8487 and 7512 with density 0.30% and 0.50%, for control and stimulated groups respectively. BA-20000-20 and BA-20000-25 are most similar in size and density to spleen control and spleen low dose radiation graphs at a threshold of 0.85. Those biological data sets have size and density 19389, 0.21% and 19371, 0.22%, respectively. A selection of graphs of size 10000 and 15000 with varying densities was also generated.

Sample degree distributions for the randomly-generated scale-free networks are given in Figure 3.5. Only the degree for the first 5000 vertices is pictured, by which point the scale-free nature of the degree distribution is evident.

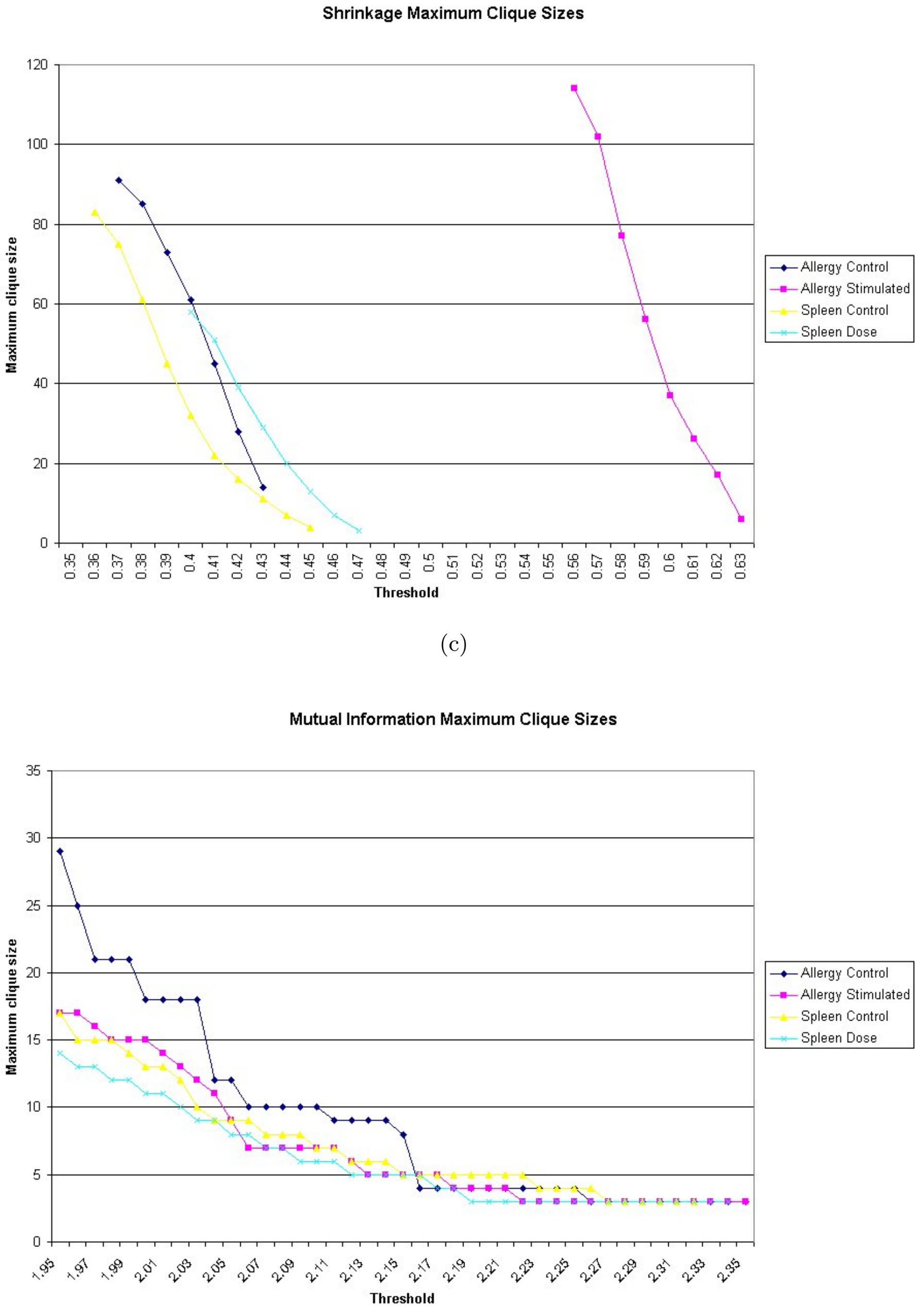
Maximum clique size was measured and averaged over 50 instances of each of the graphs listed in Table 3.1. Figure 3.6 shows that maximum clique size increases steadily with size and density. Note that while BA-8000-12 and BA-8000-20 are smaller than the 10000 vertex graphs, they exhibit larger maximum clique sizes due to their increased density. Compared to graphs constructed from Pearson and Spearman correlations, the synthetic scale-free graphs exhibited a significantly smaller maximum clique sizes than the biological graphs of



(a)

(b)

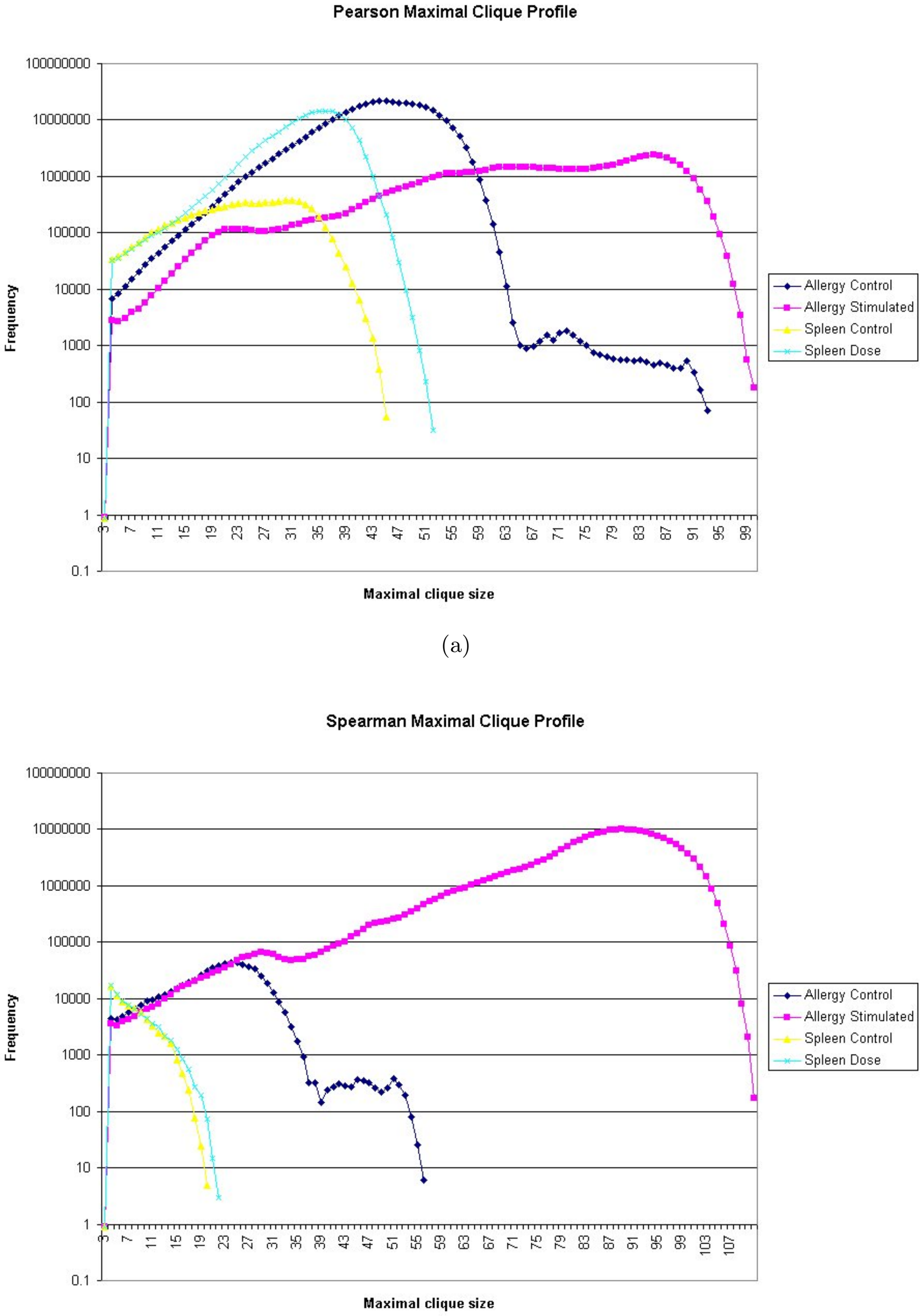
Figure 3.3: Maximum clique sizes for graphs on allergy and spleen data using various similarity methods. (a) Pearson, (b) Spearman, (c) Shrinkage, (d) Mutual Information.



(c)

(d)

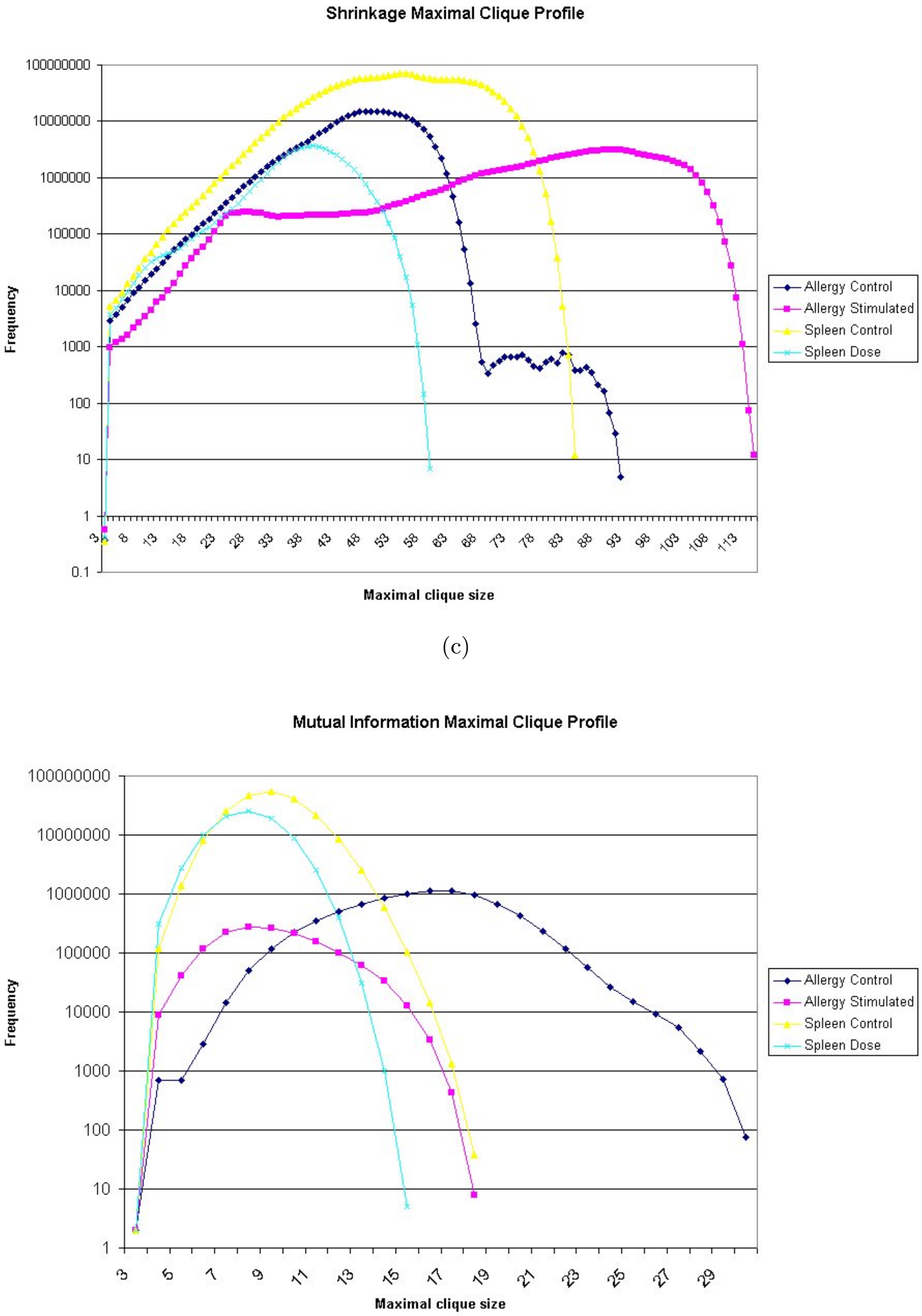
Figure 3.3: Continued.



(a)

(b)

Figure 3.4: Maximal clique profiles for graphs on allergy and spleen data using various similarity methods. (a) Pearson, (b) Spearman, (c) Shrinkage, (d) Mutual Information.



(c)

(d)

Figure 3.4: Continued.

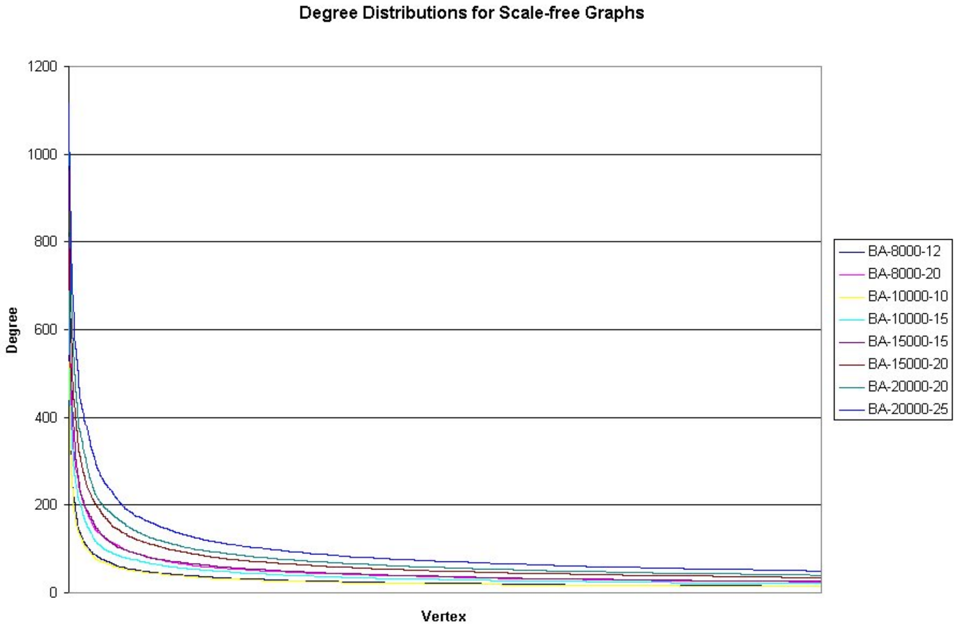


Figure 3.5: Degree distribution for the first 5000 vertices in scale-free graphs generated with the BA method.

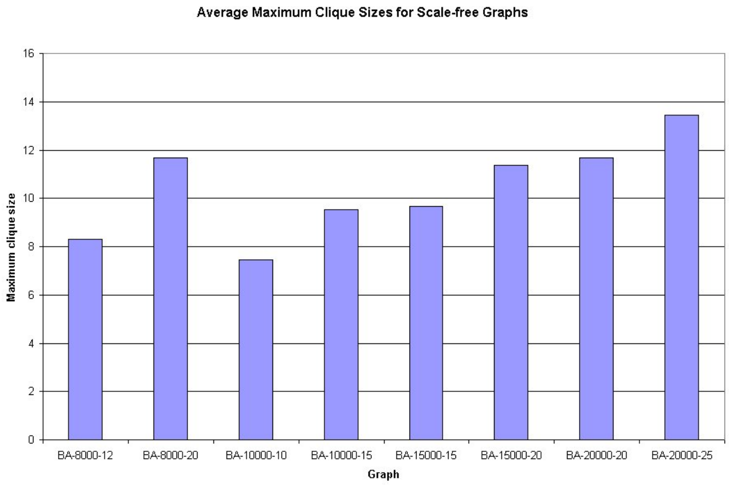


Figure 3.6: Maximum clique sizes for scale-free graphs generated with the BA method.

similar size and density. Allergy control and stimulated graphs at a Pearson threshold of 0.90, for example, achieved maximum clique sizes of 55 and 110, respectively. BA-8000-12 and BA-8000-20, on the other hand, had average sizes of 8 and 12.

Maximal cliques were enumerated from a scale-free graph selected from those generated for each size and density combination. Figure 3.7 shows the maximal clique profile for each of the larger graphs, as well as BA-20000-40, another 20000 vertex graph with 0.40% density. All of the synthetic graphs show a large number of small cliques with the number dropping precipitously as the clique sizes increase. Comparatively, most of the biological profiles at similar densities showed a bell-shaped distribution.

3.4 Conclusions

The analysis of correlation and degree distributions, along with maximum and maximal clique profiles, of various biological data sets can help identify properties that are likely to be found in many biological data sets. Genome-scale computations and visualization of each of these properties indicate that many of them remain essentially unchanged regardless of the similarity computation method used.

Some of the properties that characterize the data sets examined here are bell shaped similarity distributions and scale-free degree distributions. Linear decreasing maximum clique sizes were observed as the threshold was increased. Maximal clique profiles were found to be mostly unimodal and possibly skewed, depending upon the data set studied.

Results on random scale-free networks constructed using the Barabasi-Albert method showed relatively smaller maximum clique sizes than biological graphs from Pearson and Spearman graphs of similar size and density. Figure 3.7 also indicates that significantly fewer maximal cliques are to be expected in the BA scale-free networks.

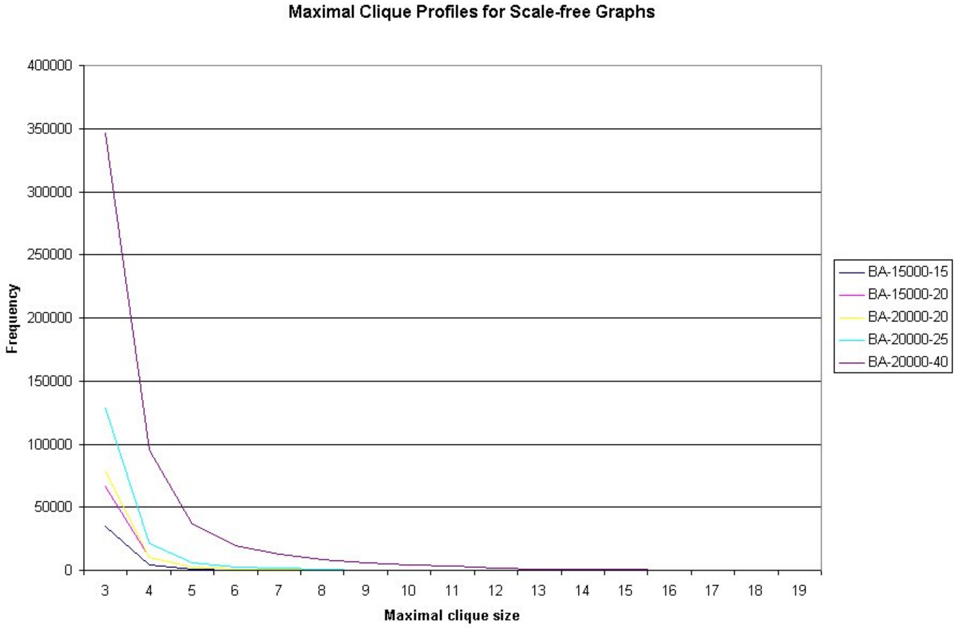


Figure 3.7: Maximal clique profiles for scale-free synthetic graphs.

An ideal extension to this work is a large-scale study of the previously discussed correlation and structural properties of biological graphs. With thousands of gene expression data sets available from the Gene Expression Omnibus website [8], an automated process to compute selected graph metrics could build a compendium of properties for biological data. This work focused explicitly on gene expression data, but the same approach can be applied to other types of biological data such as proteomic and SNP data.

Chapter 4

Identifying Stress- and Disease-Associated Genes From Graph-Based Models

4.1 Introduction

A central goal in many gene expression microarray studies is to identify those genes that are involved in some biological response to stress. This stress can take the form of changes in environmental factors such as temperature, atmosphere, or the introduction of some other external factor like radiation. Microarrays are also often used to quantify genetic change due to disease. The aim in this type of study is to uncover those genes related to complex disease by identifying altered expression levels and gene interactions between healthy and disease states in various organisms.

This chapter focuses on the identification of genes involved in complex diseases, which are those that involve multiple genes. Simultaneous disease-related changes in the expression of multiple genes can be identified in an analysis of microarray data from healthy and disease samples. The classical approach to identifying these genes is a basic differential expression analysis, which is explored further in Section 4.2.1.

Uncovering disease-related genes can assist in the diagnosis and treatment of many diseases. However, traditional differential expression can detect only changes in expression level between two or more groups, not changes in interactions. The maximal clique approach and associated tools introduced in Chapter 1 are ideal for identifying both localized and network-wide changes. The benefit of microarray technology and the clique-based approach is that it looks at relationships between all genes. The challenge to overcome in this case is the difficulty in interpretation of the possibly millions of correlations. Researchers require a systematic approach to extracting biologically meaningful differences based upon these correlation results.

4.2 Triple screen

The use of three separate gene filters, each applied either to correlation or clique results, is introduced to help identify those genes that are most "different" between two or more groups of samples. These screens were developed by M. A. Langston and first introduced in [1] to identify genes involved in a response to low doses of ionizing radiation, but are used here to find genes that are most likely to be involved in some biological response to a disease. Genes and relationships identified by these methods can also be used to distinguish between each type of sample. The filters can be applied in series or simultaneously in parallel. The individual resulting gene sets are then intersected to obtain a single list of genes passing all three screens. The use of these methods helps to overcome the limitations of traditional differential expression methods and the challenges presented by correlation and clique analysis.

4.2.1 Differential expression

A differential expression analysis identifies those genes that exhibit a significant change in expression level between different groups of samples. Genes found to differ in expression between healthy and disease samples, for example, are likely to have been affected in some way by the disease. Multiple samples for each group can be included in the analysis, increasing the fidelity with which differentially expressed genes are identified.

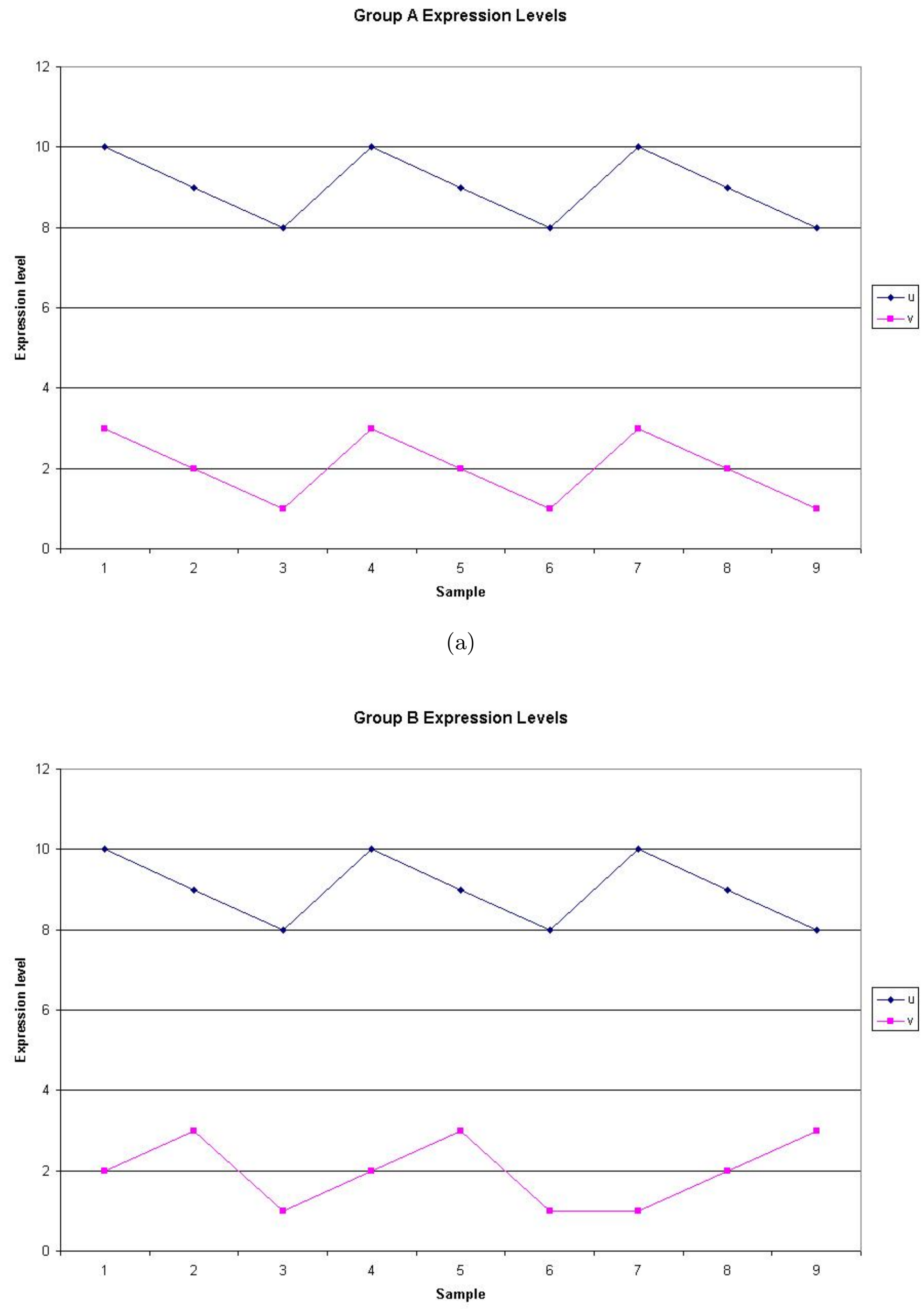
Examining differences in expression for a single gene between two groups is the most commonly used and most basic way to find genes possibly associated with some disease, and has been in use since the first microarray study [6]. Many methods exist for differential expression analysis, including [9] and [10]. The standard differential expression process is performed for each gene under study (usually all genes on the particular microarray chip model). For each gene, the most basic methods generally compute a fold change, which is a measure of the degree of increase or decrease between two groups. Many methods also perform some statistical test as a measure of significance of the differential expression results. Those genes with the largest absolute fold change and most significant p-value are normally retained for further study.

In a graph-theoretical context, differential expression can be viewed as a vertex-level comparison. The goal is to identify those vertices that differ between two groups, and no other graph structures such as edges or subgraphs are examined.

Traditional differential expression methods have some benefits, mainly that they provide a quick, well-accepted approach to identifying changing genes. These methods, however, are based upon changes in expression in individual genes and are unable to detect changes in relationships between two or more genes. For example, suppose we have two groups of samples A and B and that genes u and v are highly and lowly expressed over all samples, respectively, as illustrated in Figure 4.1a. Genes u and v are then not differentially expressed. Notice the pattern of expression for genes u and v are exactly the same, resulting in a correlation value of 1.0. Figure 4.1b shows the same two genes; still neither is differentially expressed, but the expression profile of gene v has changed. This results in a correlation value of 0.0. Although in neither figure are u and v differentially expressed, there is an altered interaction between the two genes from group A to group B.

4.2.2 Differential correlation

In the same vein as traditional differential expression methods, differential correlation identifies pairs of genes that exhibit a significant change in their correlation value between two groups of samples. However, differential correlation addresses the shortcomings of standard differential expression by examining changes in gene-gene correlations between two or more sample groups rather than just changes in expression value. This allows one to uncover



(a)

(b)

Figure 4.1: Expression levels for two hyothetical genes illustrating changes in gene-gene relationships. (a) shows expression levels for sample group A while (b) shows the expression levels for group B.

altered interactions associated with stress or disease, and can identify important genes even if they do not differ significantly in expression level between two or more groups.

Two genes are differentially correlated if their correlation value is less than some lower threshold t_l in one group and greater than an upper threshold in the other group. This is stated more formally in Definition 8. Without loss of generality, the following definitions assume that the lower or less significant correlation value occurs in sample group A.

Definition 8. *Two genes u and v are said to be* differentially correlated *between two groups A and B if $|r_A(u,v)| \leq t_l$ and $|r_B(u,v)| \geq t_h$, for some predetermined thresholds t_l and t_h, called the "lower" and "upper" thresholds, respectively.*

To be more statistically rigorous, specifically when dealing with Pearson correlation coefficients, Definition 9 can be employed. This approach identifies differentially correlated genes based upon a measure of significance of the correlation value. Again, the lower or more significant correlation value can occur in either sample group A or B.

Definition 9. *Two genes u and v are* differentially correlated *if $|p_A(u,v)| < \alpha$ and $|p_B(u,v)| \geq \alpha$ at a particular significance level α.*

Note that this approach does not take into account the size of the difference in the magnitude of correlation between gene u and gene v, but only if the relationship is considered statistically significant. This method can be modified, however to require a difference of say 0.05 between the upper and lower correlations. This would require that $|p_A(u,v)| < \alpha$ and $|p_B(u,v)| \geq \alpha + 0.05$. Since there is a one-to-one correspondence between each correlation value together with the corresponding sample size and the significance p-value, this method approach is somewhat equivalent to Definition 8.

Just as differential expression operates at the vertex level, differential correlation performs edge-level comparisons, and can be viewed in a graph-theoretical framework. Suppose network graphs have been constructed from two different types of samples, resulting in two graphs G_A and G_B.

Definition 10. *Given graphs $G_A = (V, E_A)$ and $G_B = (V, E_B)$, two vertices $u \in V$ and $v \in V$ are* differentially correlated *if $|w_A(u,v)| \leq t_l$ and $|w_B(u,v)| \geq t_h$.*

Any of these three flavors of differential correlation are applicable to graph-based microarray analysis and will find those altered gene-gene relationships between two groups. However, Definition 8 will be used exclusively for the analyses in the remainder of this chapter.

4.2.3 Differential topology

Having already explored methods to identify vertex-level and edge-level differences in two different graphs, the remaining changes to uncover occur at the subgraph level. There are many different ways to compare particular subgraphs within two different graphs, including searching for maximum common subgraphs, and analyzing clique intersection graphs [71].

Two approaches are examined here to identify genes participating in differential topological relationships. The first is to quickly determine those transcripts showing a differential clique abundance and the second performs an exhaustive comparison of the densest subgraphs from each group. Each of these approaches assumes that two graphs, G_A and G_B, representing biological networks have been constructed from two different groups of microarray samples. Maximal cliques have been enumerated from each of these graphs, resulting in two different, but possibly overlapping collections of cliques $\mathcal{A}$ and $\mathcal{B}$.

Differential clique abundance

The most basic approach to differential topology is to compute a measure of differential clique abundance for each gene. The idea is to find those genes that differ most in abundance between maximal cliques enumerated from graphs on two different groups of samples. The hypothesis is that a gene appearing in many cliques in one group but not the other plays an important role in a stress or disease response. To identify these key genes, a scaled difference score is calculated.

Definition 11. *Let $\mathcal{A}$ and $\mathcal{B}$ be two collections of cliques. Define the* scaled difference score (SDS) *for each vertex in either $\mathcal{A}$ or $\mathcal{B}$ to be the absolute percent difference in clique membership between the two collections of cliques, scaled between* 0 *and* 1*.*

Let pct_A and pct_B equal the percent of all cliques $C_A \in \mathcal{A}$ and $C_B \in \mathcal{B}$ containing vertex u, respectively. The unscaled difference score is calculated as

$$DS(u) = 100 \cdot \frac{|pct_A - pct_B|}{\frac{pct_A + pct_B}{2}} \tag{4.1}$$

This scaled difference score is designed to mimic the standard percent change formula

$$\text{Percent change} = 100 \cdot \frac{\text{New} - \text{Original}}{\text{Original}}$$

observing that this formula requires some modification to be applicable in this case. Specifically, it must be ensured that both increases and decreases of equal magnitude bear the same weight. This is achieved in Equation 4.1 by dividing by the average of the two percent changes instead of the initial value.

This difference score $DS(u)$ is computed for each vertex, maintaining the global minimum (MIN) and maximum (MAX), which are used to compute the scaled difference score

$$SDS(u) = \frac{DS(u) - MIN}{MAX - MIN} \tag{4.2}$$

bringing the difference score for each vertex into the range $0 \ldots 1$.

Note that the vertex showing the largest percent difference between the two groups will have a scaled difference score of exactly 1.0 while the vertex with the least difference will have a scaled difference score of 0.0. The scaled difference score says nothing of the absolute magnitude of the difference in abundance, but its magnitude is only relative to all of the other differences. One must look to the absolute difference in abundance between the two collections of cliques for this information.

Clique similarity

It is often desirable to know exactly which cliques from one collection appear in another. A simple clique containment similarity metric, based upon the well-known information theoretic measure known as precision, can be computed for each clique within each group. In practice, one begins with two collections of maximal cliques that were enumerated from two different graphs on the same vertex set. A similarity score is computed between each clique in the first collection and each clique in the second collection. Those cliques with

Table 4.1: Jaccard index example.

(a)

Clique	Members		
A	1	2	3
B	3	4	5

(b)

Clique	Members				
A	1	2	3		
C	3	4	5	6	7

the lowest similarity with each of the cliques in the second collection are the most unique between the two collections.

The Jaccard index is often used to measure the similarity between sets of objects. Here, when comparing clique A to all cliques in collection $\mathcal{B}$, the goal is to find the maximum proportion of clique A that is found in some clique in $\mathcal{B}$. This allows the identification of the most unique cliques in each group. The example in Table 4.1 illustrates why the Jaccard index, defined in Equation 4.3, fails in this case.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{4.3}$$

The two cliques in Table 4.1a have a Jaccard index of $\frac{1}{5}$, since they share one element of five total. However, the Jaccard index of the cliques in Table 4.1b is $\frac{1}{7}$. Obviously, the same amount of clique A is contained within both cliques B and C, but the similarities are different because of the increase number of elements in the union.

This difficulty can be corrected using a modified similarity metric, defined in Definition 12.

Definition 12. *Let $\mathcal{A}$ and $\mathcal{B}$ be two collections of cliques. Define the* **containment similarity** *between set A and set B to be $S(A,B) : \mathcal{A} \times \mathcal{B} \to \mathbb{R}$ such that*

$$S(A,B) = \frac{|A \cap B|}{|A|}$$

Lemma 1. *For every A and B in $\mathcal{C}_1$ and $\mathcal{C}_2$, respectively*

$$0 \leq S(A,B) \leq 1$$

Proof. Clearly both $|A \cap B| \geq 0$ and $|A| \geq 0$, so $\frac{|A \cap B|}{|A|} > 0$. The largest that $|A \cap B|$ can be is $|A|$, since each element of $A \cap B$ must be contained in A, hence $\frac{|A \cap B|}{|A|} \leq 1$. Therefore $0 \leq S(A,B) \leq 1$. □

$S(A,B) = 0$ when the sets A and B have no elements in common while $S(A,B) = 1$ if and only if A is contained entirely in the set B. We can think of $S(A,B)$ as a measure of the number of elements of A that are also contained in the set B. It is worthwhile noting that the two parameters of S are not commutative.

Lemma 2. *$S(A,B) = S(B,A)$ if and only if $|A| = |B|$.*

Proof. (⇒) Suppose $S(A, B) = S(B, A)$, then

$$\begin{aligned}
\frac{|A \cap B|}{|A|} &= \frac{|B \cap A|}{|B|} \\
\Rightarrow |B||A \cap B| &= |A||B \cap A| \\
\Rightarrow |B||A \cap B| &= |A||A \cap B| \\
\Rightarrow |B| &= |A|
\end{aligned}$$

(⇐) Suppose $|A| = |B|$, then

$$\begin{aligned}
S(A, B) = \frac{|A \cap B|}{|A|}&, \text{ but since } |A| = |B|, \\
\frac{|A \cap B|}{|A|} = \frac{|A \cap B|}{|B|} &= \frac{|B \cap A|}{|B|} \\
&= S(B, A)
\end{aligned}$$

□

Observe that this measure of set similarity is inversely related to a measure of set dissimilarity.

$$\overline{S}(A, B) = 1 - S(A, B)$$

Having defined a metric to compute clique-clique similarities, it can now be applied to the collections of maximal cliques enumerated from the graphs G_A and G_B constructed on two different sample sets. The first step is to identify the largest part of each clique in $\mathcal{A}$ that can also be found in some clique in $\mathcal{B}$.

For each $A \in \mathcal{A}$ compute

$$M_i = \max_{1 \leq j \leq |\mathcal{B}|} S(A, \mathcal{B}_j)$$

where $\mathcal{B}_j$ is the j^{th} clique in $\mathcal{B}$. These M_i values are then sorted in ascending order, leaving the cliques most unique to collection $\mathcal{A}$ at the beginning of the order. This process must then be repeated for each clique in $\mathcal{B}$ due to difference in cliques in each collection, and the noncommutativity of the scaled difference score.

4.2.4 Implementation

This conceptually simple comparison requires considerable computational resources, especially when the size of the collections of cliques involved is considered. The graph from each group may have millions of maximal cliques within it, resulting in trillions of comparisons. For this reason, a parallel MPI implementation of the clique similarity has been employed in these analyses.

Each round of comparisons begins with a master process assigning `BLOCKSIZE` cliques from the first set of maximal cliques to `np` worker processes. Workers then compute the similarity from Definition 12 for each of the cliques specified by the master. These scores are returned to the master node, and the most unique cliques are identified.

4.3 Results

Each of the three differential screens were performed on data from a study of seasonal allergic rhinitis (SAR). SAR is thought to be a complex disease, involving altered expression and interactions between many genes. The three differential gene filters described above were used to identify disease- and treatment-associated changes at the vertex, edge, and subgraph levels.

4.3.1 Data set

Samples from twenty patients were collected and underwent microarray analysis, collected from allergic patients under three conditions: control samples collected outside of allergy season (control), samples collected after exposure to an allergen (stimulated), and samples after the patient was treated with cortisone (treated). As is common with microarray data, several of the microarray results were bad, resulting in data containing 13 control, 19 stimulated, and 19 treated samples. The hypothesis proposes that key genes related to SAR will show an increase in expression (or decrease) with allergen challenge, some of which will be reversed by the introduction of the steroid. Moreover, differential correlation and topology can uncover interactions and sets of relationships changing due to the allergen stimulated and subsequent treatment.

Individual data values with an associated detection score less than $0.95 (p < 0.05)$ were removed, as were entire transcripts of poor or uncertain annotation. Correlations based upon fewer than ten pairwise-complete observations were also excluded from the analysis. $\log_2$ expression values were computed and used in the analysis.

4.3.2 Differential screen results

Differential expression

Using the method described in [9], genes differentially expressed between the control and stimulated as well as the stimulated and treated groups were identified. 1493 transcripts were found to show a significant ($\alpha < 0.05$) change in expression level between control and stimulated. This number was 1268 for the stimulated vs. treated comparison, and 997 of these were differentially expressed in both comparisons. All but six of these differentially expressed in control vs. stimulated and also stimulated vs. treated show opposite changes between the two comparisons, as would be expected from the effect of the cortisone treatment.

Differential correlation

Differential correlation was also performed using the criterion that differentially correlated edges must have a correlation value greater than or equal to 0.90 in one graph and less than or equal to 0.15 in the other. In contrast to differential expression, the differential correlation filter identified many genes participating in differential relationships: 6976 and 2904, respectively, from a total of 35713 and 2835 differentially correlated edges.

Figures 4.2 and 4.3 illustrate a portion of the differential correlation graphs for control vs. stimulated and stimulated vs. treated, respectively. In Figure 4.2, red edges are those interactions appearing with allergen stimulation while blue edges show nonsignificant

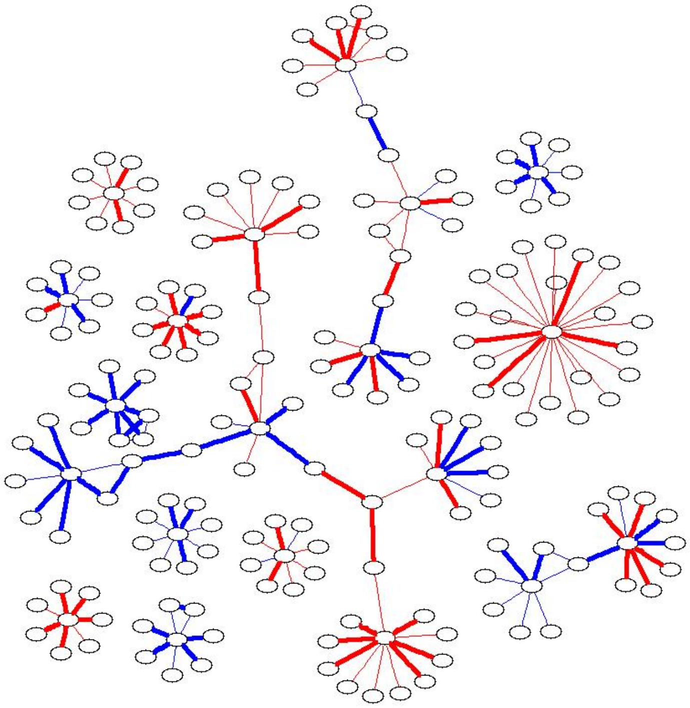


Figure 4.2: Control vs. stimulated differential correlation graph.

correlation due to the allergen stimulation. Similarly, red and blue edges in Figure 4.3 denote edges appearing and disappearing with steroid treatment respectively.

These graphs were constructed to emphasize the star-like structures found within the differential correlation graphs, where only vertices participating in six or more differential relationships were depicted. Also, vertices at the center of stars with neighbors attached to too many other vertices were removed for clarity. These differential correlation graphs have been created on graphs constructed from normally distributed random values in the range $-1 \ldots 1$ and a threshold applied at 0.90, but similar "star" shapes have not been identified. Figure 4.4a, which shows the number of differentially correlated edges incident to each vertex in a sample random graph, illustrates that those pendant vertices are not present to make the star-like structures. In contrast, Figure 4.4b shows a large number of the low-degree vertices available in the control vs. stimulated graph to appear within the stars.

Differential topology

Graphs were constructed from microarray samples of all three sets of samples at the threshold of 0.93 in control and 0.94 in the stimulated and treated groups. These thresholds were chosen to keep the number of edges in each graph relatively equal. Since the total number of vertices (including isolated vertices) in each graph is the same, this also attempts to equalize the graphs' density. Maximal cliques were then enumerated from each graph and the percent of cliques containing each gene was calculated. Table A.1 shows a summary of the structures of the graphs at various thresholds, including the number of vertices, edges, maximal cliques, and maximum clique size.

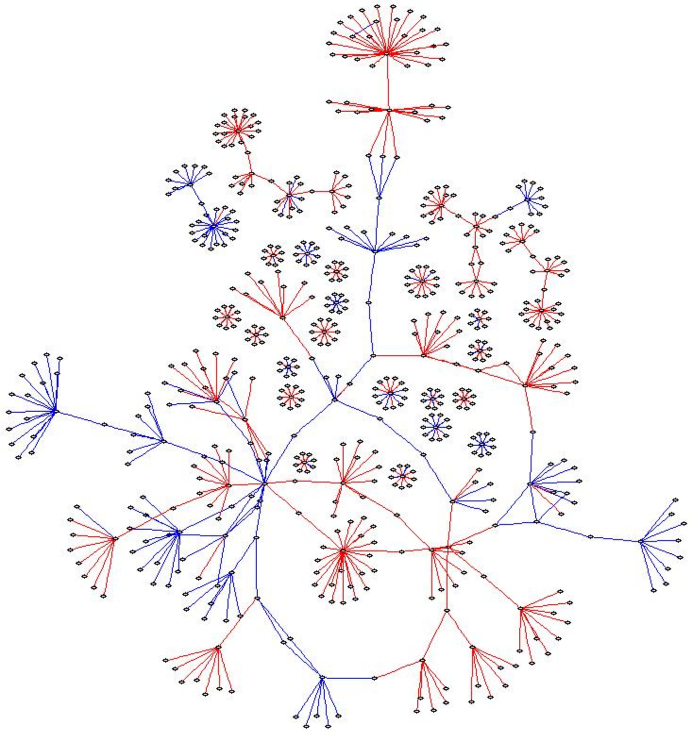
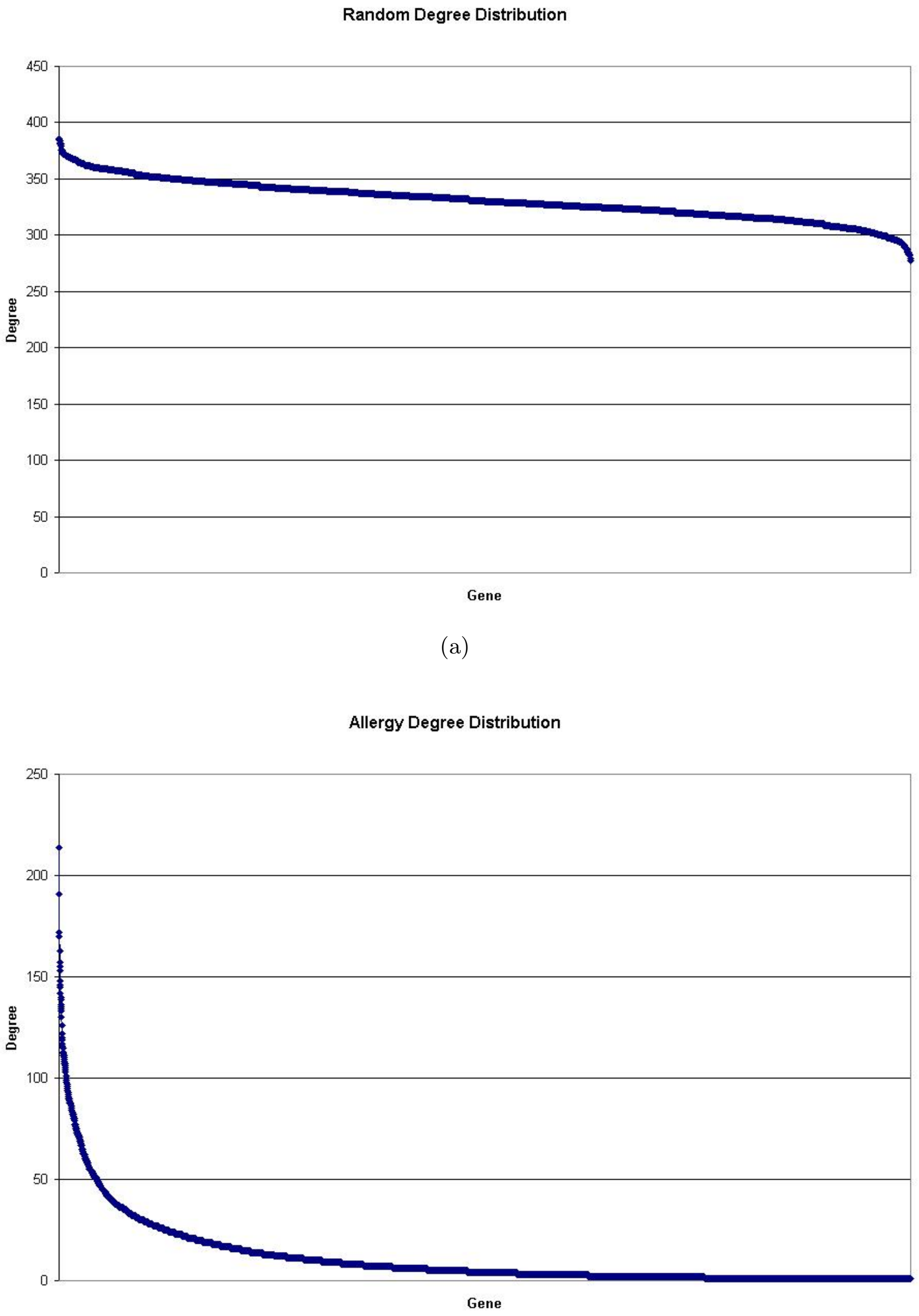


Figure 4.3: Stimulated vs. treated differential correlation graph.



(a)

(b)

Figure 4.4: Degree of vertices within random and allergy differential correlation graphs. (a) random and (b) allergy differential correlation graphs.

Table 4.2: Number of genes identified by each differential filter.

	Exp	Cor	Top	Exp ∩ Cor	Exp ∩ Top	Cor ∩ Top	Exp ∩ Cor ∩ Top
C vs. S	1493	6576	94	981	7	74	4
S vs. T	1268	2904	210	315	30	123	23
C vs. S ∩ S vs. T	999	1664	90	184	7	2	0

In the differential abundance analysis, the control vs. stimulated comparison showed 94 genes with more than a ten percent absolute difference, while this number was 210 in the stimulated and treated comparison. Ninety of these genes were found to have significant topological differences in both comparisons. Again, most genes showed a significant increase in abundance in the stimulated group followed by a much lower clique membership in the cortisone treated cliques. Incidentally, all of the top ten genes showing the largest difference between control and stimulated are also the ten most different genes in stimulated and treated.

Intersecting gene filters

Table 4.2 shows the number of genes passing each of the filters, and all possible combination of filters applied in series. It is evident that the differential topology filter is the most stringent of the three screens. This outcome is to be expected due to the strict requirement that all edges be present between vertices within a clique. Also obvious is that the differential expression filter is often at odds with the differential correlation and topology screens. Intersecting the differentially expressed genes with those identified by the other two filters drastically reduces the number of genes continuing through the combined analysis.

Clique similarity results were examined by considering only those cliques with a maximum similarity of zero between two groups. This allowed the identification of maximal cliques that were enumerated from control individuals, for example, but were completely absent from cliques on allergen-stimulated graphs. These cliques appearing only in one group or the other can be examined for genes known to be key to a particular disease. Other genes appearing in the same cliques might therefore also be determined to be associated with the disease.

Control vs. stimulated comparisons identified 2801 maximal cliques appearing only in the control group. These cliques ranged in size from 3 to 11. The stimulated vs. control comparison showed the largest differences with 6139 cliques present in the stimulated results but not in control. Over 2000 of these maximal cliques had size at least 30 with the largest containing 41 transcripts.

The difference between the stimulated and treated collections of maximal cliques was much less pronounced. One hundred six maximal cliques appearing in the stimulated group were not present in the treated group while 382 were present in treated and not stimulated. The maximum sizes of these cliques were 4 and 6, respectively.

4.4 Conclusions

Filters were reviewed to expand and improve upon the traditional differential expression methods for cases where there is enough data available to compute reliable correlation coefficients. Traditional differential expression methods, in this case, identified almost 1500 genes showing differences from the control to stimulated group. With the goal of laboratory verification, it is often desirable to reduce this number to a smaller set of genes that are likely to be involved in many altered relationships between two groups. The application of differential correlation and differential topology filters brought this number to a manageable 74 genes for further study and validation.

Parallel code to compute pairwise clique similarities was implemented and cliques found to be entirely present in only one group or the other (control and stimulated, for example) were identified. These cliques provide another means to reduce large lists of maximal cliques for each group into only those that best differentiate between two groups of samples. Cliques identified as being unique to one group or the other can then be examined for possible disease-associated genes.

Chapter 5

Effects of Choice of Preprocessing Method on Correlation, Graph Structure, and Differential Comparisons

5.1 Introduction

In Chapter 4, two new measures of differential comparisons were reviewed that augment the standard differential expression analysis: differential correlation and differential topology. These comparison methods are applicable to a wide range of data, specifically any data upon which two or more sets of reliable correlations can be computed. When analyzing biological data, however, the natural question of the effect of data preprocessing on differential screen results arises when working with gene expression data. It is known that choice of preprocessing method affects correlations between gene-gene correlations [72]. Graph structure and differential comparisons are closely linked to pairwise transcript correlations and are likely to also be affected by choice of data preprocessing method. The consistency of differential correlation and differential topology results over several normalization and preprocessing methods is examined.

As already mentioned, preprocessing and normalization of microarray data is a key step to bring all data values into comparable levels and to remove some sources of technical variation. This preprocessing of gene expression microarray data prior to analysis can be separated into three steps [73], the first of which may or may not be applied:

1. *Background subtraction* is the measurement and removal of background signal. This noise is mostly the result of the natural background level of the microarray chip and also nonspecific binding [74, 75]. To facilitate the measurement of background levels, Affymetrix chips include mismatch probes, as described in Section 5.1.1.

2. *Normalization*, which is required to make values comparable among different arrays. This is necessary in any type of differential comparison based upon values from more than one array. This step can incorporate information from an assortment of chips to perform the normalization, or perform a predefined scaling on each chip.

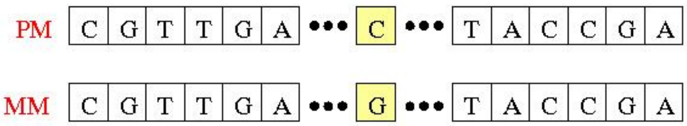

PM
C G T T G A ••• C ••• T A C C G A
MM
C G T T G A ••• G ••• T A C C G A

Figure 5.1: Affymetrix probe pairs consists of both a Perfect Match (PM) and Mismatch (MM) probe.

3. *Expression summarization* includes both the computation of final expression values for each probe set and the possible application of a logarithmic or trigonometric transform.

5.1.1 Affymetrix probe background

This comparison focuses on analysis of data generated with the Affymetrix GeneChip platform because of the prevalence of the Affymetrix system and the available of publicly-accessible raw GeneChip data sets in the NCBI Gene Expression Ombibus (GEO) [7, 8]. Data from other platforms such as Illumina and Agilent were not considered, as many of the software packages and implementations usually used for preprocessing and normalization are not available for data from these platforms.

For each of the data sets examined here, gene transcripts of 25 bases are measured by eleven probe pairs, with each pair comprised of a Perfect Math (PM) and a Mismatch (MM) probe. These PM and MM probes have the exact same sequence, except for an inversion of the middle (thirteenth) base in the Mismatch probe, illustrated in Figure 5.1. The Mismatch probe helps in estimating the intensity of the background signal, since RNA should not hybridize to this sequence. As a consequence, it also serves as a measure of probe-dependent nonspecific hybridization, or stray signal [76].

Although Affymetrix probes contain both Perfect Match and Mismatch probes, many preprocessing and normalization methods (RMA, described later, for instance) ignore the value of the Mismatch probe. There has been disagreement about the usefulness of background subtraction in general. Irizarry et al. show that background correction improves accuracy but decreases the precision of differential expression results [73]. Accuracy and precision represent specificity (the proportion of true negatives) and sensitivity (the proportion of true positives), respectively. Wang et al. also show that background correction using the Mismatch probe value can lead to fewer candidate genes with possible biological relevance in [77]. One cause of the difficulty with the Mismatch probe is that the MM intensity increases with an increase in true signal as measured by the PM probe. Hence, the Mismatch probe is measuring some actual signal and subtracting it from the PM probe would remove that detected signal [74].

5.1.2 Preprocessing methods

Many methods exist to preprocess data and bring disparate values from several arrays into comparable ranges. Six of these approaches were chosen to be examined here because of their pervasiveness in the genomics literature: MAS5, RMA, gcRMA, PDNN, dChip, and VSN. There has been no clear consensus on a "best" or even most favored method, and each of these approach the task of preprocessing and normalization from a slightly different standpoint.

5.1.3 MAS5 (Affymetrix Microarray Suite 5.0)

The method implemented in the Affymetric Microarray Suite version 5.0 is usually generically referred to as MAS5 or MAS5.0. With details described in [76], this approach performs a zone-based local background correction followed by a global scaling of each individual array to the desired mean value. This method makes use of the PerfectMatch−Mismatch difference (PM−MM) to attempt to correct for nonspecific binding. Final expression summaries are generally not log transformed by the software, although $\log_2$ values are used in some intermediate steps.

5.1.4 RMA (Robust Multichip Average)

The robust multichip average (RMA) was introduced in [74, 78]. RMA background correction uses a convolution approach that models the observed signal $S = X + Y$, where X and Y are signal and background, respectively, with $X \sim exp(\alpha)$ and $Y \sim N(\mu, \sigma^2)$. Corrected expression levels are represented by $E(X|S = s)$ [79]. This method uses only PM probe values and ignores the MM probes. Quantile normalization is performed to give values from each array the same distribution.

5.1.5 GC-RMA (GC Robust Multichip Average)

GC-RMA [80, 81] uses a different statistical model to calculate background noise based upon probe sequence information. Rather than subtracting the entire Mismatch probe intensity, this model subtracts a fraction of the MM level, which has been adjusted for the probe affinity. Normalization and summarization remain the same as in standard RMA.

5.1.6 PDNN (Positional-Dependent Nearest Neighbor)

The Positional-Dependent Nearest Neighbor method provides another method for incorporating probe sequence information into the background correction process [82]. In contrast with GC-RMA, this approach makes use of a free energy model and is based upon only the PM probe intensities. Quantile normalization is once again employed to equalize the distribution of expression values among arrays.

5.1.7 dChip

dChip, introduced in [83] uses an invariant set normalization described in [84]. Similar to the MAS5.0 approach, dChip divides each microarray into zones to compute a local background noise value. This is performed post-normalization. A multiplicative model developed around the model-based expression index (MBEI) is then applied to the background-adjusted expression values. This method uses both PM and MM probe intensities, but can also be used with PM-only technologies [85]. As with MAS5.0, final dChip expression values are also not usually log transformed by the software.

5.1.8 VSN (Variance Stabilizing Normalization)

The variance stabilizing normalization is concerned only with the normalization and expression summarization steps of preprocessing. The authors note that the variance of expression values increases with their mean and introduce a variance stabilizing method to help address this problem [86, 87]. In the tests here, the VSN method was used with no background

Table 5.1: Comparison of preprocessing methods.

Method	Background Correction	Seq. Info.	Normalization	PM/MM
MAS5.0	Yes	No	Linear scaling	Both
RMA	Yes	No	Quantile	PM-only
gcRMA	Yes	Yes	Quantile	Both
PDNN	Yes	Yes	Quantile	PM-only
dChip	Yes	No	Invariant set	Both
VSN	N/A	No	Variance stabilizing	N/A

correction and used only PM intensities in the analysis. Rather than using the standard log transform, a trigonometric transformation is employed when computing final expression values. VSN also calibrates arrays with the aim of bringing values on different chips into comparable scales.

5.1.9 Method comparisons

Table 5.1 presents a comparison of all of these methods based upon background correction and normalization type, and whether the approach makes use of probe sequence information or Mismatch probe intensity.

5.1.10 Implementations

While possibly several implementations are available for each of the methods above, those developed for the R statistical language and Bioconductor package were used to standardize the data input and output process across various methods. The MAS5.0 and RMA methods used were implemented in the Bioconductor affy package version 1.12.2[88]. The GC-RMA package used was gcrma 2.6.0 [89], and variance stabilization was performed with the vsn package version 3.2.1. [86]. The PDNN (version 2.3.0.0) [90] and dChip (version 2007) [91] software used were stand-alone Windows-based programs developed by the authors of the respective methods.

5.2 Contrasting differential results between preprocessing methods

The effect of preprocessing method on differential correlation and differential topology results is examined. The first of the three differential screens, differential expression, is not examined here. There have been many studies investigating how the set of differentially expressed genes identified is affected by varying the normalization and background correction method used. Some of these, such as [13, 92] have shown that differential expression is highly dependent upon choice of preprocessing method. Others have observed only little impact, although a different type of microarray and a different subset of methods were studied [93].

5.2.1 Assumptions

It was seen in Chapter 4 that the differential correlation and topology screens can be used to identify putative disease-associated genes. Each of the data sets on which comparisons are performed in Section 5.3 is comprised of two groups of samples. These groups are a set of healthy or "normal" samples and a set of samples from an individual with a particular disease, which will be referred to as the "control" and "disease" groups, respectively.

To assist in designing fair comparisons, certain assumptions must be made. In order for comparisons to reflect likely everyday use by researchers, defaults were used in each software package implementing the various preprocessing methods. Detection scores, if produced by the platform or software, were ignored. While this might affect final data set quality, results from each method should be affected equally. Not removing values with low detection scores also ensures that the data set contains no missing values and that each equal pair of correlation coefficients computed in each data group is assigned the same significance level.

Because Pearson correlation coefficients are most often used in genetic coexpression studies, they were also used here in both differential filters. If a preprocessing method did not output expression values sufficiently transformed (logarithmic or otherwise), expression levels were $\log_2$ transformed prior to correlation computation. Since correlations are the values under study in both the differential correlation and differential topology filters, no adjustments to the data were required to bring values in different groups to comparable levels.

5.2.2 Expression-level-based filtering

To avoid spurious results caused by correlation between unexpressed genes, transcripts that are consistently lowly expressed must be removed from the comparisons. To this end, a very conservative filter that excludes transcripts that are consistently lowly expressed in both groups was applied. [94] gives several options for intensity-based filtering. The approach used here is based upon background levels as measured by control probes.

The Affymetrix arrays studied here contain several poly-adenylated control probes to detect specific sequences from the organism *B. subtilis* [95]. These probes were used to calculate a global measure of low expression based upon the average expression of each of these probes over all arrays. A threshold was set at two standard deviations less than this average, and any transcript that is never greater than this cutoff in either group were removed. This allowed the retention of those genes that were highly expressed in one group, or even one array, but not in the others. On the data sets described in Section 5.3.1, this identified from 0 to 3225 transcripts as lowly expressed for data sets on the HG-U133a and MG-U74v2 arrays, depending upon data set and normalization method used. For data from the HG-U133 Plus 2.0 array, which contains many more transcripts, up to 35371 were removed due to low expression levels.

5.2.3 Variance-based filtering

Probesets showing little variation were also ignored in the comparison, since it is those genes whose expression level is varying–both within group and between groups–that are of the most interest. Several options are given in [96] for dealing with genes exhibiting low variance. Here, low variance genes were defined to be those in the lowest half of all genes in both the control and disease group. This is the most conservative option, since the variance is required to be significant in only one group.

It is often desirable to retain genes that are "flat" in both groups (control and disease), specifically if the gene is consistently low in one group and high in the other. Combining

Algorithm 1: Expression-based filtering

```
Input: Affymetrix microarray data set
Output: List of probes of low expression to be removed
foreach Control do
    $\overline{x}_i$ = Average expression for control $i$ over all arrays
    $\sigma_i$ = Standard deviation for control $i$ over all arrays
$\overline{x}$ = Average expression of all controls
$\sigma$ = Standard deviation of expression for all controls
foreach Probeset do
    foreach Array do
        if $Expression < \overline{x} - 2 \cdot \sigma$ then
            Add Probeset to remove list
return remove
```

Algorithm 2: Variance-based filtering

```
Input: Affymetrix microarray data set
Output: List of probes of low variance to be removed
foreach Probeset do
    $\sigma_{Control}(i)$ = Expression variance in control group for probeset i
    $\sigma_{Disease}(i)$ = Expression variance in disease group for probeset i
$\sigma_{Control} = 50^{\text{th}}$ percentile of variance in control group
$\sigma_{Disease} = 50^{\text{th}}$ percentile of variance in disease group
foreach Probeset do
    if $(\sigma_{Control}(i) < \sigma_{Control})$ AND $(\sigma_{Disease}(i) < \sigma_{Disease})$ then
        Add Probeset to remove list
return remove
```

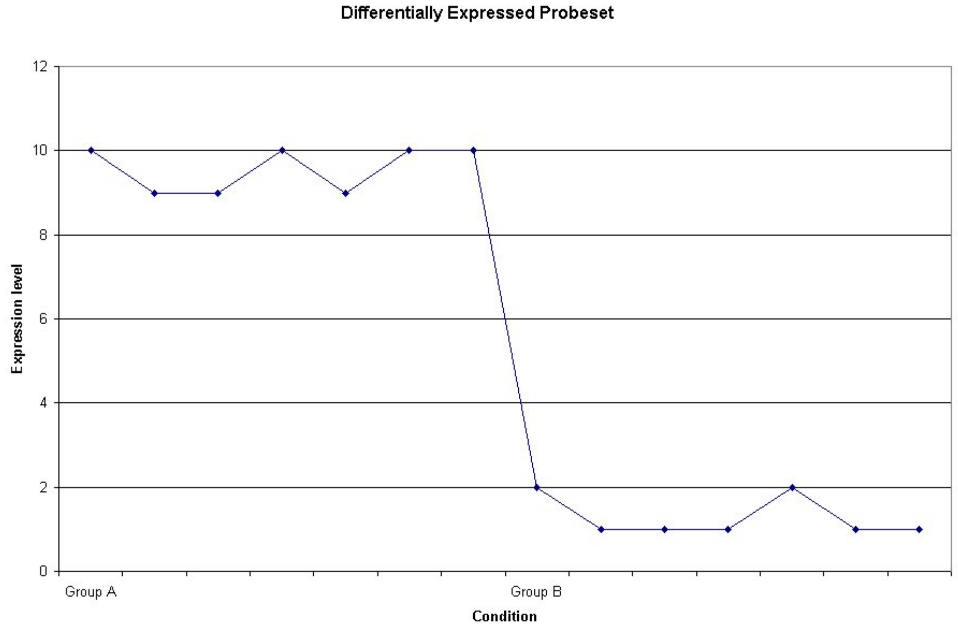


Figure 5.2: Differentially expressed probesets show increase variance over all arrays. Here, expression values are observed for two sample groups A and B. Values are consistently high in group A and low in group B, resulting in a large variance.

data from both groups and choosing those genes with variance in the top fifty percent would keep these key genes in the analysis. Overall, these genes exhibit variance, but there is no within-group variation. However, calculating the variance and choosing the top fifty percent for each group separately consistently produced lower variance limits, resulting in a more conservative approach. Hence this was the approach used here, to allow the greatest number of genes to be retained for further analysis. These higher limits, when considering the groups together, could be caused by differential expression. If a gene is consistently high in one group and low in the other with the mean lying near the median, a large variance would result, as illustrated in Figure 5.2. While variance within each group is relatively small, the variance across the entire data set is great. A more practical reason for considering the groups separately is that there is often a different sample size in each of the two groups.

Genes with variance in the top fifty percent were retained to ensure that the same number were removed from each set for a more fair comparison. Another option would be to consider only genes determined to be differentially expressed via some statistical method or simple fold change. As previously mentioned, differential expression is often at odds with with other two filters of the triple screen. This method is not used, since it has been shown (see citations in Section 5.1) that differential expression results can be influenced significantly by choice of normalization. Yet another method would be to consider an absolute variance cutoff, but data-dependent methods are usually preferable. This could also affect fair comparisons between various methods as different methods can cause individual genes to exhibit more or less variance.

It is important that this filtering process happen post-background correction and post-normalization. It is difficult to know which probesets truly have low intensity or variance across multiple arrays using unnormalized values. Also, these low expression values are often necessary to perform reliable normalization and therefore must be retained.

5.2.4 Correlation computation and differential comparisons

To compare differential results among various preprocessing methods, pair-wise correlation coefficients were computed between each pair of genes for the data sets described in Section 5.3.1 for control and disease groups separately. Before differential correlation and differential topology filters were applied, correlation histograms were plotted. All within-group correlations were assigned the same significance, since there were no missing values. This allows for a direct comparison of the correlation coefficients to be made. Visual inspection of the correlation distributions in Figure 5.3 shows that the same correlation threshold cannot be chosen for all of the normalization methods, so a percentage of the top correlations were chosen from which to create graphs.

Graphs were constructed for control and disease groups separately at the chosen thresholds and the degree of each vertex was computed for a comparison of the most highly connected genes from each method using Jaccard similarities. The number of maximal cliques and maximal clique sizes were also recorded for selected threshold values.

The differential correlation filter described in Chapter 4 was applied using a slight modification of the differential correlation definition. Since the thresholds for significant correlations were systematically computed for both the control and disease group, differentially correlated genes were determined using both of these thresholds, as in Definition 13.

Definition 13. *Two genes u and v are* differentially correlated *if*

$$|r_{Control}(u,v)| > t_{h_{Control}} \quad \textit{and} \quad |r_{Disease}(u,v)| < t_l$$

or

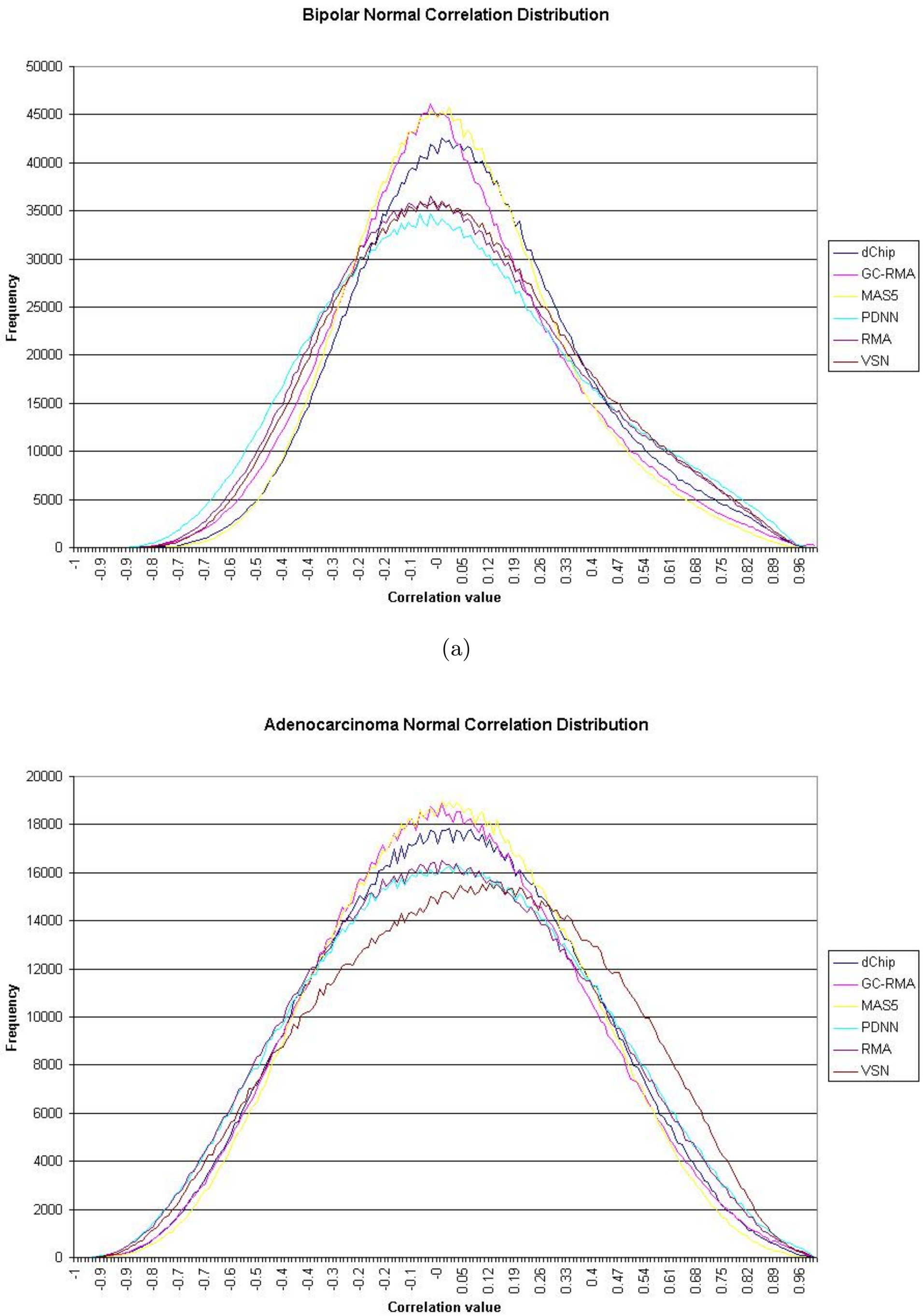
$$|r_{Disease}(u,v)| > t_{h_{Disease}} \quad \textit{and} \quad |r_{Control}(u,v)| < t_l$$

where $t_{h_{Control}}$ and $t_{h_{Disease}}$ are the selected upper thresholds for the control and disease group, respectively. The lower threshold t_l remains the same for both groups.

Differential topology results using the differential abundance method require no such modification, so the topology filter was applied on maximal cliques enumerated from control and disease graphs at the chosen threshold. Genes showing greater than a ten percent difference in clique membership between control and disease groups were identified as exhibiting differential topology.

Pair-wise comparisons of the lists of genes passing the differential correlation and topology filters were performed for each pair of methods. The overlap of the 100 genes showing the greatest correlation and topological difference between methods was computed, as well as the number of vertices and edges identified as participating in significantly different relationships by both preprocessing methods. For differential topology comparisons, the number of genes identified by both methods was also calculated, as was the total number of genes passing both of the differential filters in both methods. Similarities between the methods are computed with the Jaccard similarity index introduced in Chapter 4.

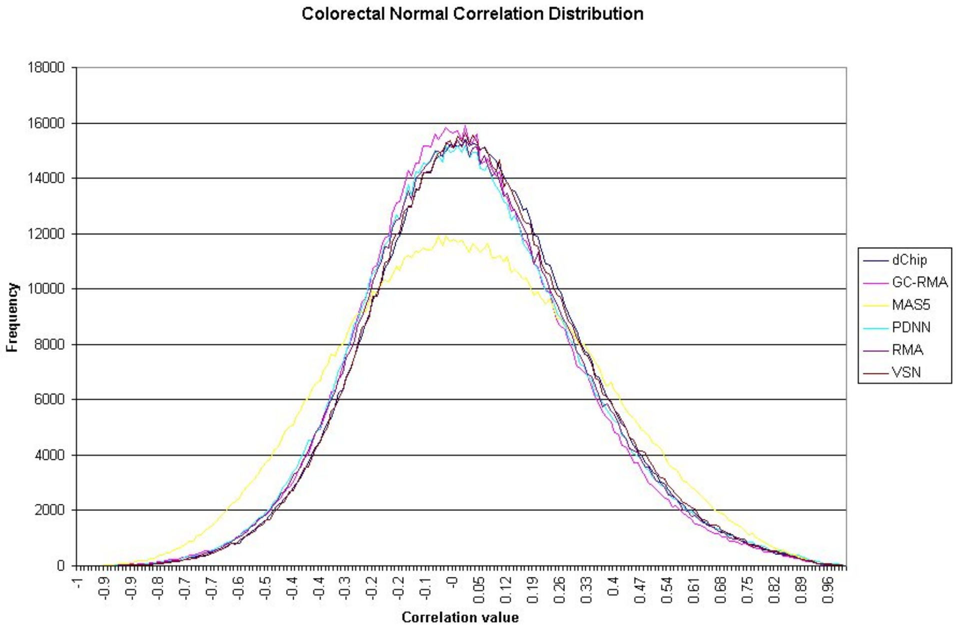
In an attempt to identify outlier preprocessing methods, in respect to differential results, five-way intersections of each of these measures were performed, as well as intersections of all six methods. By removing one preprocessing method from the intersection of the results, it is possible to see whether that one method caused an unproportionate decrease in the



(a)

(b)

Figure 5.3: Control group correlation distributions for each of the three data sets under study. (a) Bipolar disorder, (b) Pulmonary adenocarcinomas, (c) Colorectal adenomas.



(c)

Figure 5.3: Continued.

number of genes identified as differentially correlated or having participating in differential topology.

Lastly, the biological significance of the difference in results between all the methods is tested using web-based ontological tools such as WebGestalt [26].

5.3 Results

5.3.1 Data

Three publicly-available data sets whose experimental design are amenable to differential comparisons were examined. These data sets were chosen because they were produced on Affymetrix GeneChip arrays and should contain enough samples in each group to compute reliable pairwise correlations.

Pulmonary adenocarcinomas
Data was collected to study lung tumors resulting from exposure to urethane in *Mus musculus*. MG-U74v2 microarrays were used to measure gene expression in twenty-nine tumor samples and fifteen samples from adjacent normal tissue [97].

Bipolar disorder
This *Homo sapiens* data set contains measurements of expression levels for sixty-one samples collected post-mortem from thirty bipolar individuals and thirty-one controls, processed on HG-U133A GeneChips [98].

Colorectal adenomas
Samples of a colorectal adenoma were collected from thirty-two patients, along with samples from the patients' normal mucosa of the colon [99]. mRNA expression levels were collected using HG-U133 Plus 2.0 GeneChips.

5.3.2 Effect on correlation and graph structure

Figures 5.3 and 5.4 show the distribution of correlation values for each of the three data sets described above, for the control and disease groups, respectively. Only correlations on genes common to all preprocessing methods after the removal of low intensity and low variance transcripts are shown here. This ensures that the number of correlate pairs computed on data from the various methods is consistent. In both figures, the MAS5.0, GC-RMA, and dChip methods show a larger peak at the zero correlation value in the bipolar and adenocarcinoma groups, caused by a greater number of nonsignificant correlations. The other three methods exhibit heavier tails, and hence, on average, more dense graphs at the higher thresholds.

This distinction is not evident in the correlation on genes in the colorectal adenoma samples, and in fact the MAS5.0 method has a smaller peak at zero and much heavier tails. Different preprocessing methods remove a different set of genes exhibiting low intensity and variance. By computing only correlations on the genes lying at the intersection of these sets, it is possible that other preprocessing methods removed genes that would have exhibited lower correlations for MAS5.0. As previously mentioned, independently visually examining correlations on data produced by different methods presents difficulty. Specifically, distribution sizes will differ due to the varying sample sizes.

When examining graph structure, it is possible to compare the methods in a manner closer to their likely use by retaining gene-gene correlations produced by one method even if those genes are not present in the second. Figures 5.5a and 5.5c show that the GC-RMA method stood out with respect to the large maximum clique size in the bipolar and adenocarcinoma studies. It is also interesting to note that the number of maximal cliques within the GC-RMA graphs, illustrated in Figures 5.5b and 5.5d, was similar to the other methods except at the higher end of the correlation spectrum. On the other hand, Figures 5.5e and 5.5f show that the PDNN and VSN methods produced the largest and most numerous maximal cliques in the graph on colorectal data. However, the VSN difference was more pronounced in the normal-state graph (not shown) than in the disease.

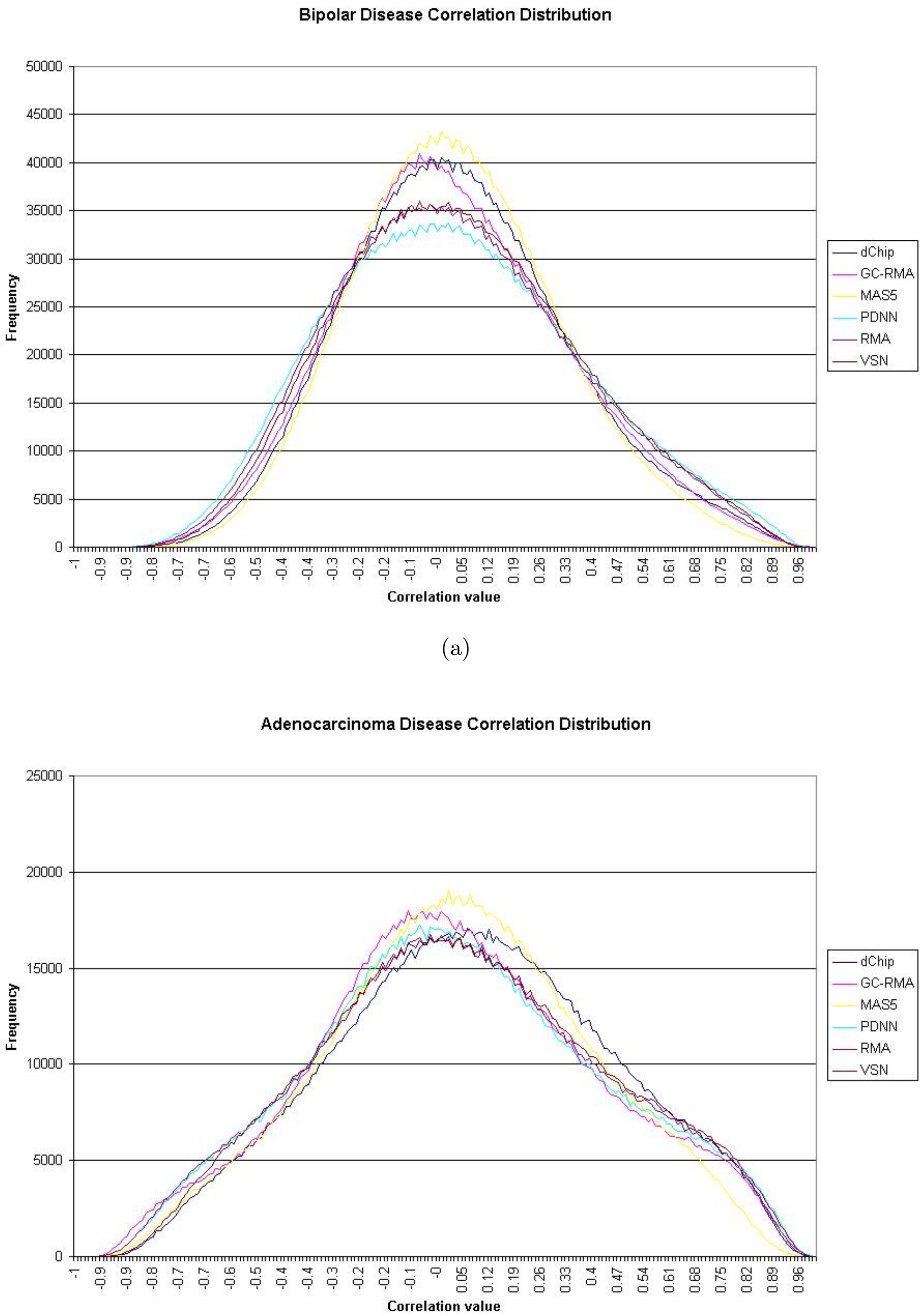
Results from comparisons of the one hundred most highly connected genes between each pair of methods were somewhat dependent upon the data set under study. In the bipolar disorder normal state data, those comparison showing a Jaccard similarity greater than 0.90 were PDNN vs. RMA, PDNN vs. VSN, RMA vs. VSN, and GC-RMA vs. MAS5.0 at various thresholds ranging from 0.85 to 0.96. Differences were also evident between the normal and disease groups. The bipolar disease data demonstrated that top similarities exist between the previously mentioned methods as well as GC-RMA vs. VSN, GC-RMA vs. RMA, and dChip vs. MAS5.0. Similarities above 0.90 in the colorectal data (both normal and disease states) were comprised exclusively of the PDNN vs. RMA and GC-RMA vs. RMA comparisons at various thresholds. For pulmonary adenocarcinoma data, the similarities above 0.90 were PDNN vs. VSN in the normal state, and PDNN vs. VSN and RMA vs. VSN in disease. Full results are available in Tables B.1, B.2, and B.3.

5.3.3 Effect on differential comparisons

As it has been defined, differential correlation is an edge-based comparison imposed upon a vertex by considering vertices participating in a differential relationship as differentially correlated. Considering vertices is often desirable since examining individual differential edges can be unwieldy. Comparisons between normalization methods here consider both sets of differentially correlated edges as well as lists of differentially correlated vertices. Approaching the comparisons from the vertex level is likely to be more robust with respect to normalization method, since there are many opportunities for a vertex to be labeled as differentially correlated.

For the colorectal adenoma data, the difference in comparing differential correlation edges and vertices is clear. In computing pairwise intersection size for the edge comparisons, Figure 5.6a shows that only one method exhibited a Jaccard index greater than 0.2. The similarities between the methods become much greater when considering the vertex comparison in Figure 5.6b, where several indices surpassed 0.6 and one was higher than 0.7.

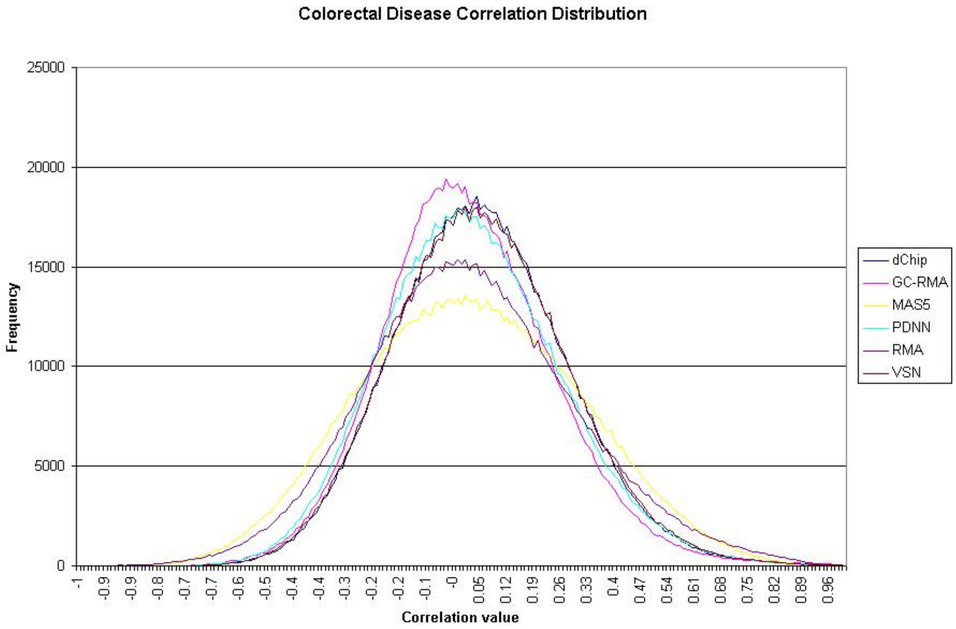
With these first comparisons, shown in Figure 5.6, it is evident that differential correlation results are highly data-dependent. Figure 5.7 illustrates the wide disparity in the pair-wise method similarities. It is interesting to note, however, that the last three comparisons in Figures 5.6b, 5.7b, and 5.7d–PDNN vs. RMA, PDNN vs. VSN, and RMA vs. VSN–all showed relatively high differentially correlated vertex similarities in each of the three data sets. The charts in Figures 5.6a, 5.7a, and 5.7c show that the edge comparisons were in stark contrast with one another depending upon the data set.



(a)

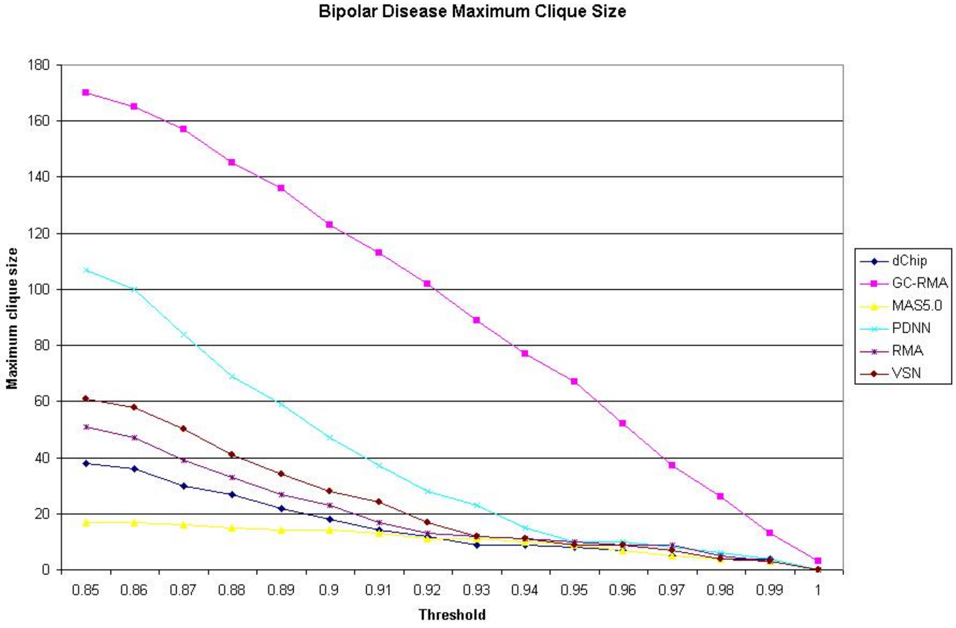
(b)

Figure 5.4: Disease group correlation distributions for each of the three data sets under study. (a) Bipolar disorder, (b) Pulmonary adenocarcinomas, (c) Colorectal adenomas.

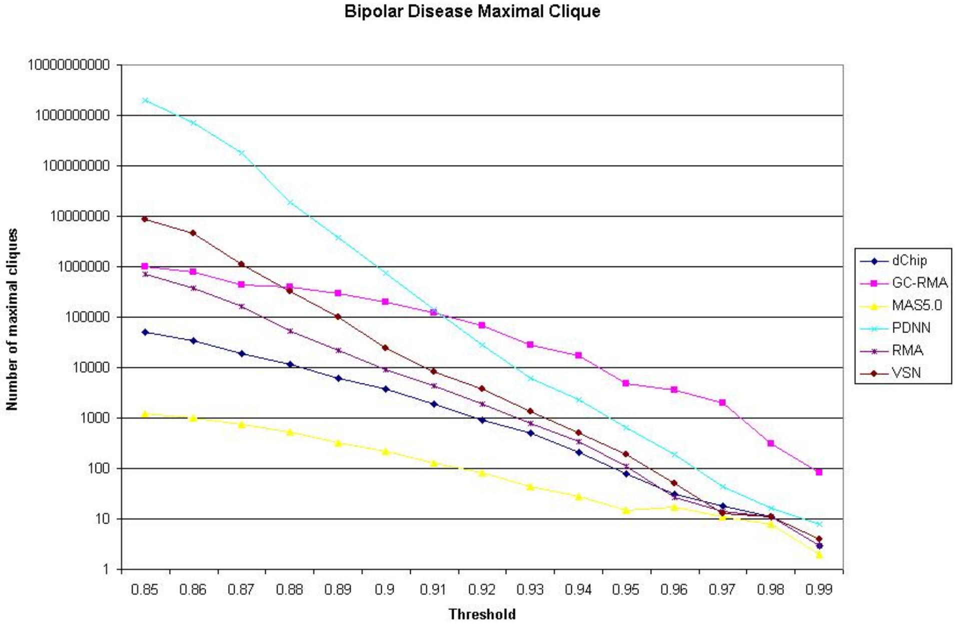


(c)

Figure 5.4: Continued.

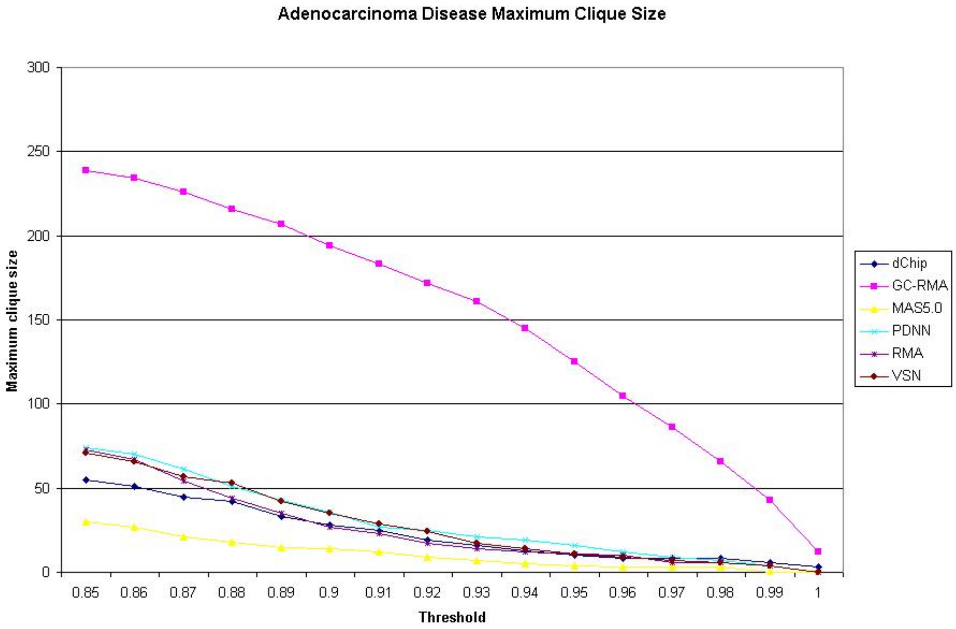


(a)

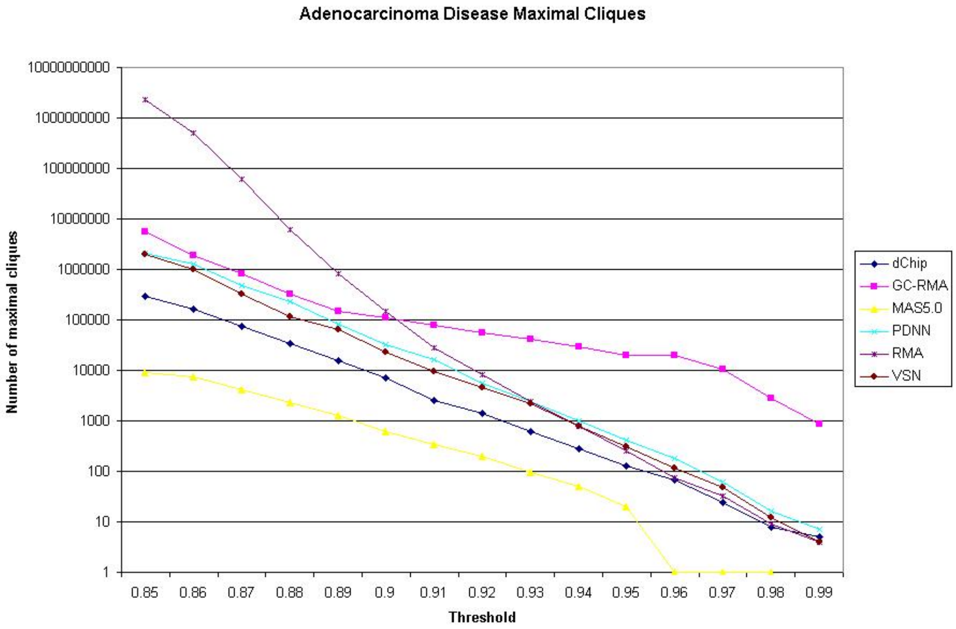


(b)

Figure 5.5: Maximum clique size and number of maximal cliques for each data set. (a) Bipolar disorder maximum, (b) Bipolar disorder maximal, (c) Pumonary adenocarcinoma maximum, (d) Pulmonary adenocarcinoma maximal, (e) Colorectal adenoma maximum, (f) Colorectal adenoma maximal.

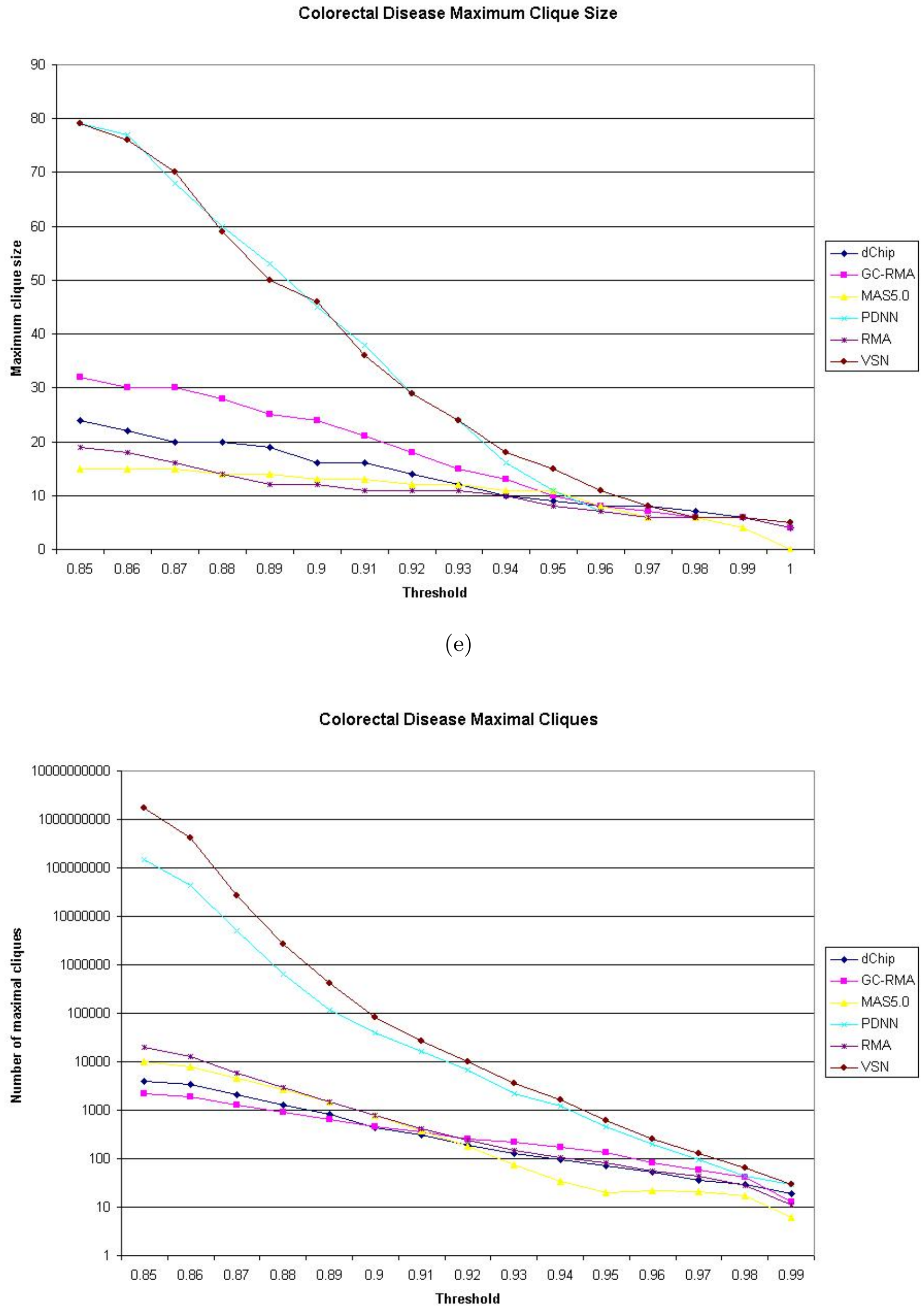


(c)



(d)

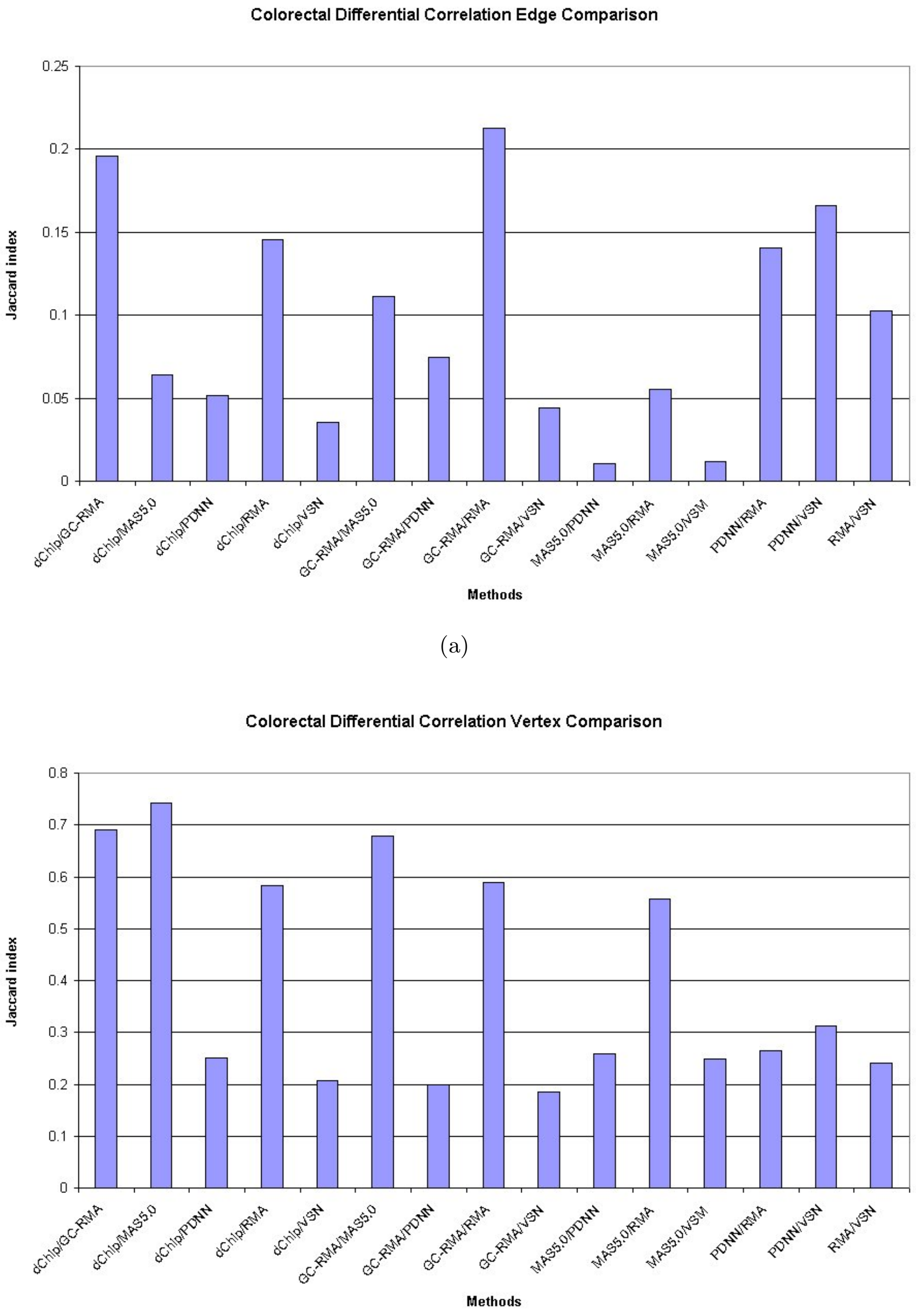
Figure 5.5: Continued.



(e)

(f)

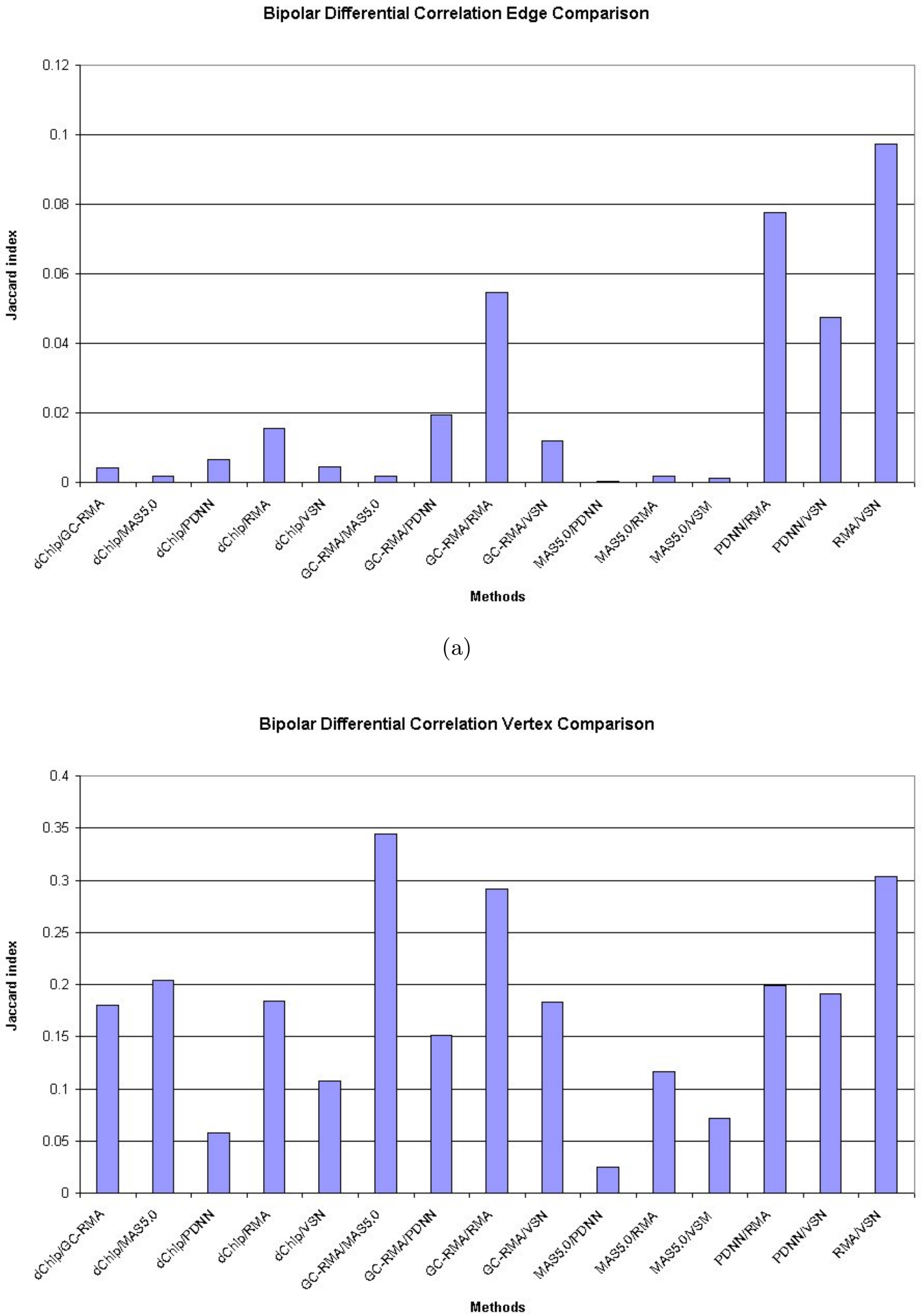
Figure 5.5: Continued.



(a)

(b)

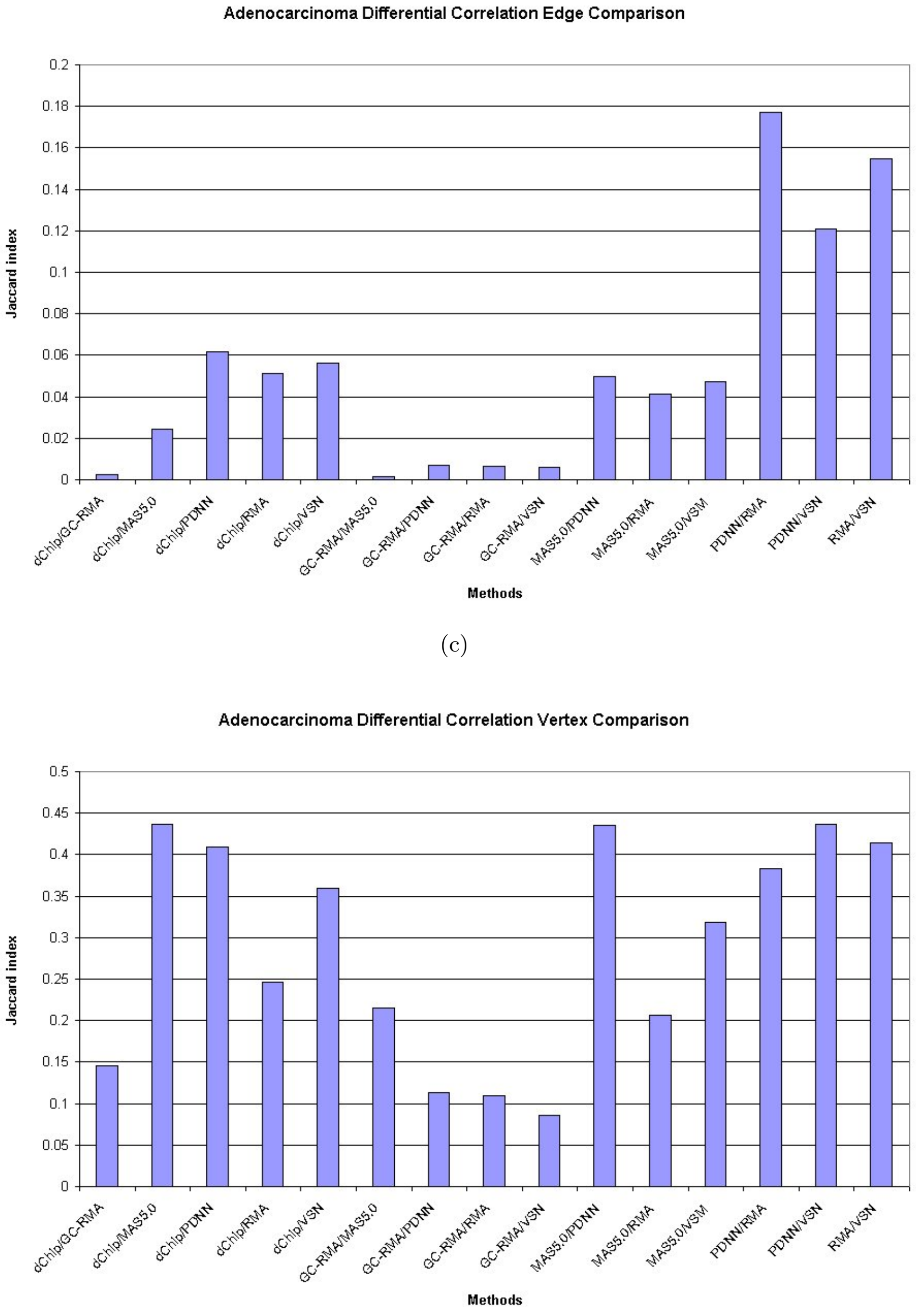
Figure 5.6: Differential correlation similarities for colorectal adenoma data. (a) Differentially correlated edge comparison, (b) Differentially correlated vertex comparison.



(a)

(b)

Figure 5.7: Differential correlation similarities for bipolar disorder and pulmonary adenocarcinoma data. (a) Bipolar disorder vertex comparison, (b) Bipolar disorder edge comparison, (c) Pulmonary adenocarcinoma vertex comparison, (d) Adenocarcinoma edge comparison.



(c)

(d)

Figure 5.7: Continued.

The pair-wise differential topology filter indicated similar results, given in the Appendix, in Figure B.1. Alongside the differential topology results are pair-wise similarities between the top one hundred genes showing the greatest difference in correlation and topology, shown in Figures B.2 and B.3. Where one hundred genes did not pass the differential filter for one or more data sets, the Jaccard similarity was computed on those genes that did pass the filter. Once again, the final three comparisons (PDNN vs. RMA, PDNN vs. VSN, and RMA vs. VSN) showed a relatively high degree of concordance in the one hundred most differentially correlated genes between the methods while the overlap in differential topology results varied greatly between methods. This smaller overlap in differential topology results is likely due to the conservative nature of the clusters identified by the maximal clique approach: the absence of a single edge will remove a vertex from the clique. In contrast, it is much "easier" for a vertex to participate in at least one differential relationship and appear in the list of differentially correlated genes.

Figure 5.8 shows the number of genes identified by both the differential correlation and differential topology filters for each pair of preprocessing methods. It is important to note that many methods show a significant number of genes in common between the two methods, considering that both the correlation and the much more conservative topology filter were applied. The comparisons of dChip vs. MAS5.0 and RMA vs. VSN show similarities across all three data sets, while many methods identify a relatively large number of genes across two of the data sets.

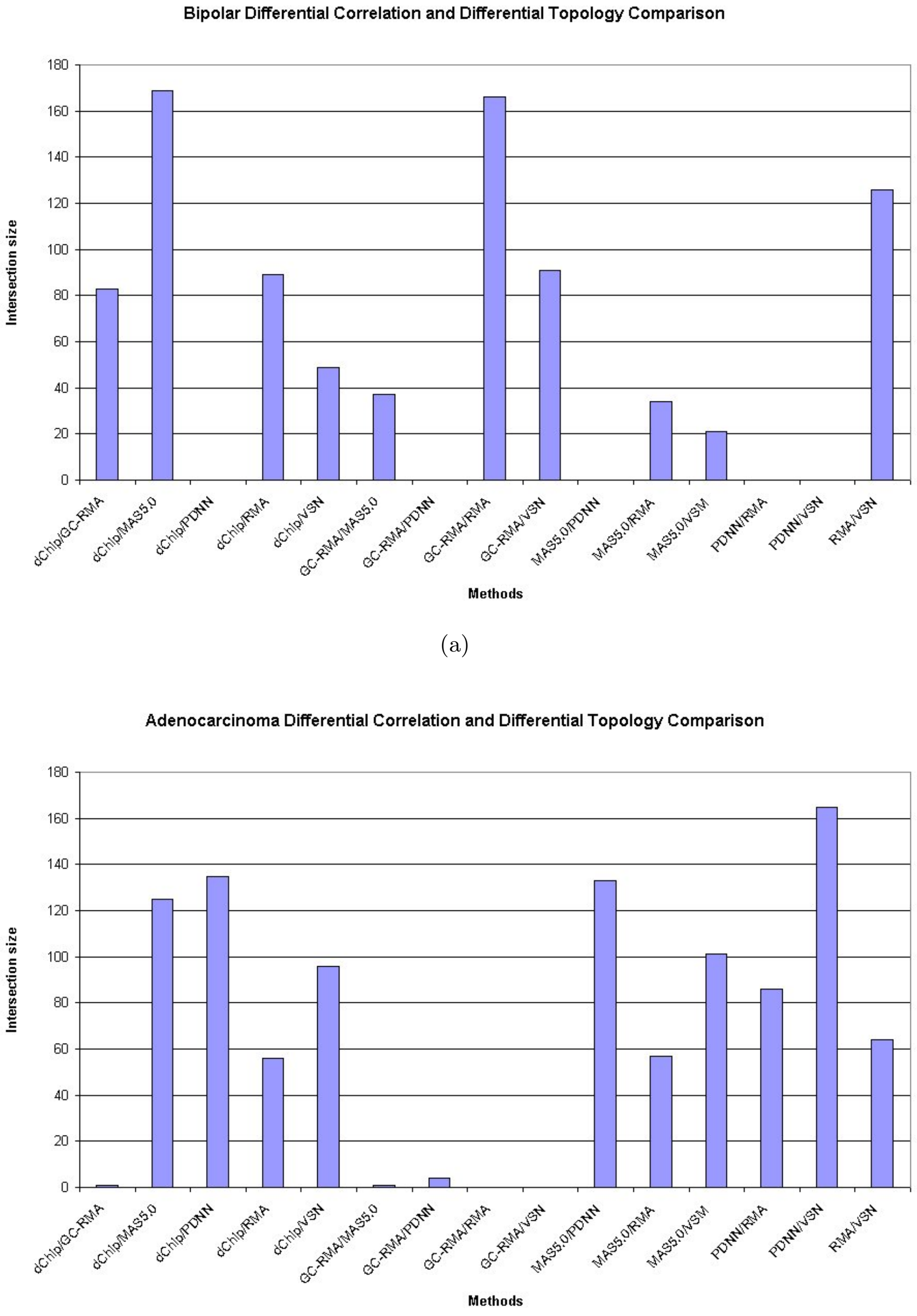
Figure 5.9 condenses the pair-wise comparison into a single metric spanning each of the data sets and differential filters. Each pair of methods was assigned a similarity value that is the average of the similarities for each differential screen (correlation edges, correlation vertices, and topology) over all data sets. It is evident that dChip and MAS5.0 showed significant similarities, as did RMA and VSN. Another five or so methods fell near the 0.2 Jaccard index.

Five-way comparisons to identify outlier methods are shown in Figure 5.10. Methods were removed one at a time and the sets of genes passing the differential correlation vertex filter were intersected and unioned, respectively. Figure 5.10a shows that only in the adenocarcinoma data did any of the methods identify a significantly different set of genes as differentially correlated than all of the others. Removal of the GC-RMA results from the intersection resulted in far more genes passing the filter. Figure 5.10b shows that removal of MAS5.0 from the union caused relatively fewer genes to be identified as differentially correlated in all data sets. This indicates that MAS5.0 data contained significantly more differentially correlated genes than did any of the other methods.

5.3.4 Biological comparisons

It was shown in Figure 5.8 that there is some degree of concordance in differential screen results between many pairs of preprocessing methods. The question then arises of whether the genes identified as exhibiting differential correlation and topology match similar biological pathways when using different data preprocessing methods. Would a researcher examining biological enrichment of differential screen results observe different pathways depending upon the normalization method used?

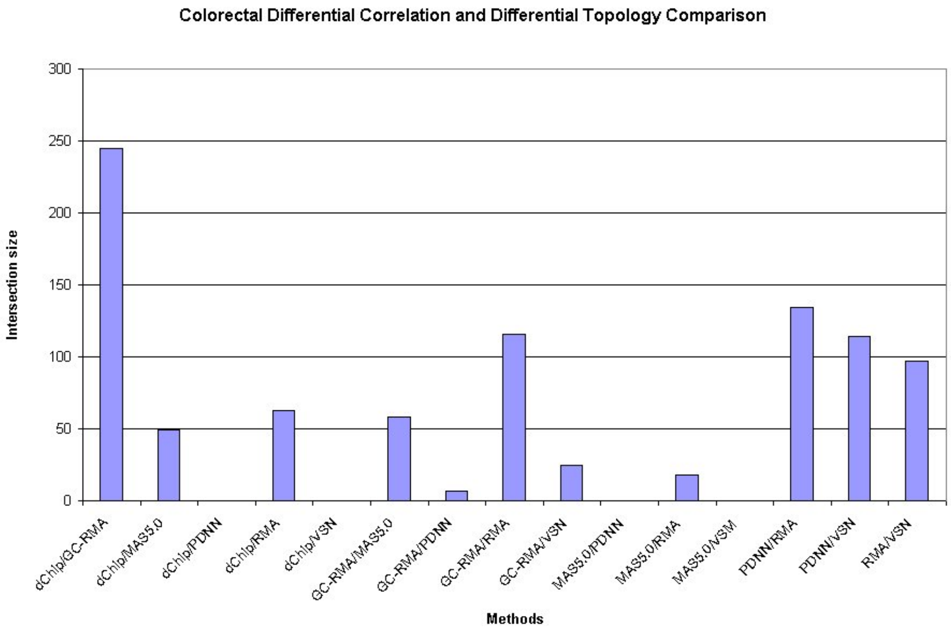
To answer this question, data preprocessed using each method were filtered separately to remove genes with low expression and variance. Genes passing both the differential correlation and differential topology filters were examined for KEGG pathway enrichment at the $\alpha < 0.05$ level for each data set using WebGestalt [26]. The number of pathways



(a)

(b)

Figure 5.8: Number of genes passing both differential correlation and topology filters for each pair of preprocessing methods. (a) Bipolar disorder, (b) Pulmonary adenocarcinoma, (c) Colorectal adenomas.



(c)

Figure 5.8: Continued.

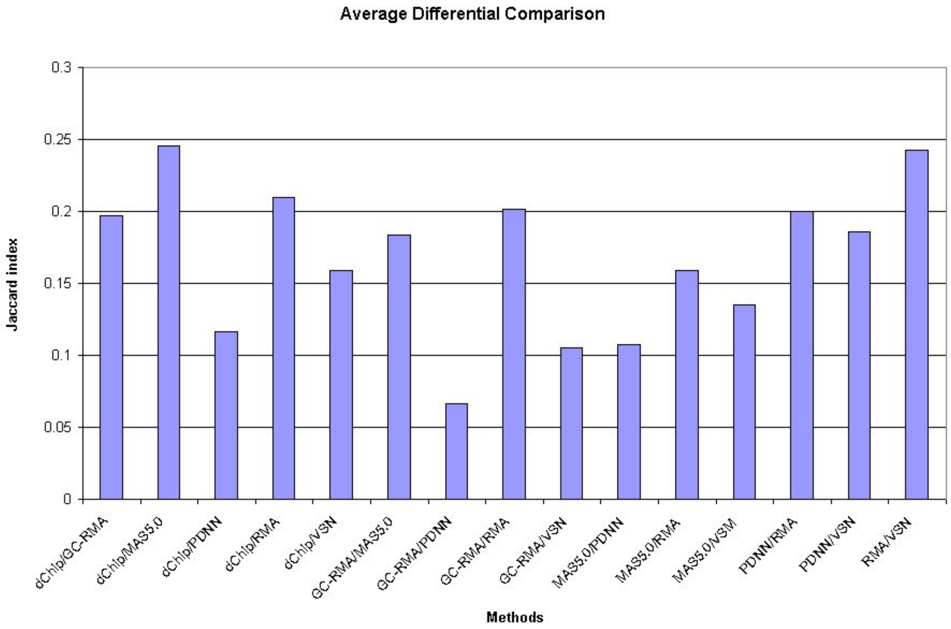
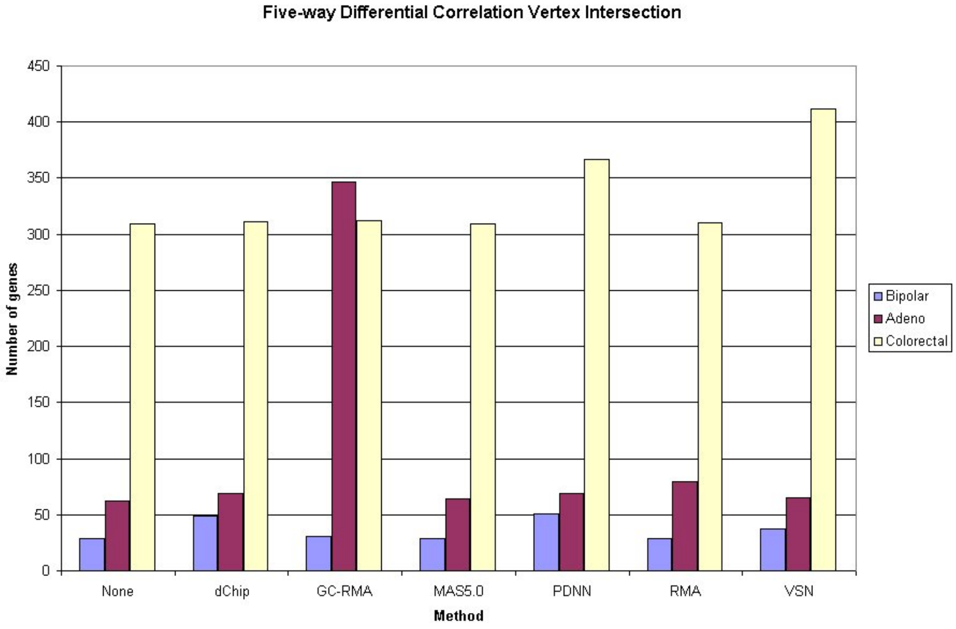
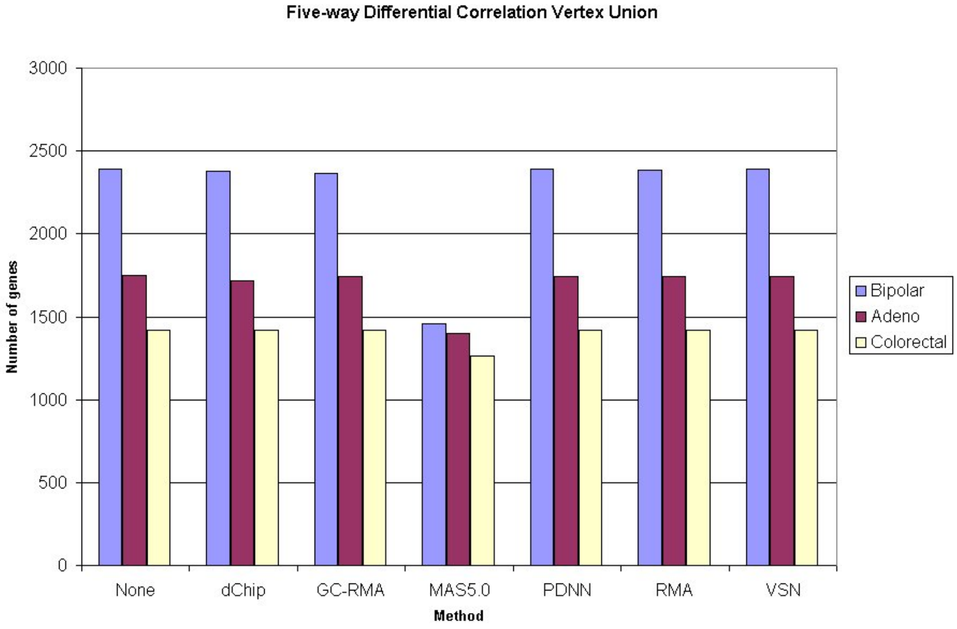


Figure 5.9: Average similarity for each preprocessing methods based upon all differential metrics.



(a)



(b)

Figure 5.10: Five-way intersection and union comparison to identify outlier methods. (a) The intersection, (b) The union.

enriched in results for each pair of methods was computed. Pulmonary adenocarcinoma data showed little overlap between methods, with dChip, PDNN, and VSN sharing only the “Neuroactive ligand-receptor interaction” pathway and the “Cell adhesion molecules (CAMs) pathway showing up in MAS5.0, PDNN, and RMA. Bipolar disorder data showed a much higher degree of similarity between methods in the pathways that were enriched, with between 5 and 18 pathways matching between each pair of methods (PDNN was removed from the comparison since differential results on PDNN data matched no pathways). The average number of pathways matching between two methods was about 11. The three greatest intersections were between GC-RMA and VSN (size 18), RMA (17), and MAS5.0 (16). The similarities in pathway matches between methods in colorectal adenoma data were mainly between GC-RMA and RMA (13), dChip (8), as well as dChip and RMA (7).

In two of the data sets there existed significant overlap in the pathways matched using different preprocessing methods. For all methods, pulmonary adenocarcinoma data hit fewer biological pathways than the other data sets, resulting in fewer chances for overlap between methods. In both the bipolar disorder and colorectal adenoma data, GC-RMA and RMA showed a high degree of concordance with other methods, and most methods hit at least one of the same pathways. Otherwise, it appears that here the choice of preprocessing method did affect the pathways in which differential results were enriched.

5.4 Conclusion

Overall, the choice of data preprocessing method significantly affects the results of the differential correlation and differential topology filters described in Chapter 4, as has already been seen in differential expression results. Although different sets of altered interactions were identified by the screens depending upon the method used, a core of differentially correlated vertices common to all methods was observed for each of the three data sets. These are the genes passing the differential correlation filter regardless of the preprocessing method used. It is possible that these genes are likely to be participating in altered biological relationships and that the differences in correlation are not due solely to noise or the normalization method used. It has been suggested that genes lying at the intersection of those identified by several normalization methods be used to narrow the list of candidates for validation [13].

The data itself seems to play an important role in determining the similarity between various methods. At the lowest level, graph structure results as measured by the degree distribution of the most connected genes, indicate some similarity between several methods but also a significant degree of data dependence. Similarly, it has been noted that clustering results are likely to vary depending upon experimental factors such as microarray platform and noise level [100].

The condensed analysis of similarities between methods, presented in Figure 5.9, showed the highest degree of concordance between the RMA and VSN methods, as well as dChip and MAS5.0. This latter similarity may be due to the similar nature of the two methods, including the zone-based background calculation and the use of mismatch probes in preprocessing. The reason for similarity between RMA and VSN, however, is not so clear. Unlike RMA, the VSN preprocessing did not perform background subtraction, and uses a different normalization method and transform of the subsequent expression values.

Comparisons of biological enrichment showed that genes lying at the intersection of differential correlation and differential topology are often enriched in a similar set of KEGG pathways, although the degree of similarity varies depending upon the data set. Since there is likely to be variation in the number of pathways involved in each experimental design, it is difficult to asses how much variation should be expected from one data set to another, and hence the number of pathways identified in results from each method. For example, in contrast to bipolar disorder data, results on pulmonary adenocarcinoma data were enriched in relatively few pathways over all methods. As with other comparisons, the methods identified as similar varied somewhat depending upon the data set analyzed.

5.4.1 Future work

This analysis did not consider several components that could play a key role in future comparisons. The effect of similarity metric choice on differential screens might contribute as much as preprocessing method to differences in results. Thresholding was minimally investigated in the graph structure results, but Chapter 2 shows that it is not always so simple an issue. It is also unclear whether background correction, normalization, and summarization methods contribute equally to differential results. Future analyses might consider these three normalization steps independent of the others, as suggested in [72].

Chapter 6

A Graph Theoretical Approach to Integrated Ecosystem Analysis*

6.1 Introduction

Up to this point, this discussion has centered on challenges arising in the use of clique tools to analyze networks modeled from DNA microarray data. However, the graph-based approach described in Chapter 1 requires only that reliable pairwise correlations can be computed between some set of variables. Many types of data are amenable to such an analysis. This chapter focuses on the application of clique-based tools to historical ecosystem data, specifically that of the North Sea. It will be seen that the approach is very similar to the approach used in gene coexpression studies, from the preprocessing of the data and building models of ecosystem interactions to the extraction of key relationships.

An analysis of the North Sea ecosystem allows one to answer several key questions important to marine biologists, fisheries researchers, and those setting fisheries and environmental policy in the North Sea area. How do abiotic environmental factors affect marine species in the North Sea? What are the effects of fishing on these species, and how do the biotic and abiotic factors affect one another? More generally, how are points in time and space related to one another based upon changes in the ecosystem? These are the types of factors that may admit an answer in a temporal and spatial analysis of the data.

Previously, these questions have been addressed using dimensionality reduction techniques such as hierarchical clustering and principal component analysis (PCA), especially as implemented in [101]. As mentioned in Chapter 1, a clique-based analysis is often preferable to these methods due to the exact nature of the clique algorithm and the density of the resulting sets of interacting variables.

Ecological data offer some interesting and unique challenges for the graph-based analysis toolchain. As a result of the method of collection, such data are often full of missing or corrupt values. As observations are extracted from the database at increasingly finer levels of temporal and spatial granularity, the number of incomplete entries also increases. There may be no observations for certain oceanic locations for a particular month, for example, but that location will possess a yearly average provided that an observation is made for at least one month. This introduces the related problem of unequal effort, which is described in Section 6.2.

*Portions of the results and figures contained here were initially presented in [5]

Table 6.1: Summary of North Sea ecosystem data available. Source: portions extracted from [102] and [103].

Data type	Source	Spatial	Temporal
Abiotic	ICES	X	X
NAO	Univ. East Anglia		X
Fish CPUE	ICES (IBTS)	X	X
Flux	NORWECOM (PGNSP)		X
Seabirds	WGSE/ESAS	X	X
Plankton	CPR (SAHFOS)	X	X
Fisheries landings	Scotland, England and Wales authorities	X	X
Benthos	North Sea benthos survey	X	
Mammals	WGSE/ESAS	X	

6.2 North Sea historical data

The data used in this analysis were collected from a variety of sources, and are described in more detail in [102]. Temporal and spatial data sets are comprised of a subset of the described data, containing measurements of the following: abiotic observations, fish catch per unit effort (CPUE), seabird abundance, plankton, mammals, benthos, North Atlantic Oscillation index (NAO), Norwegian ecological model (NORWECOM) flux, and fisheries pressures. Table 6.1 summarizes the data available at each level of temporal and spatial granularity.

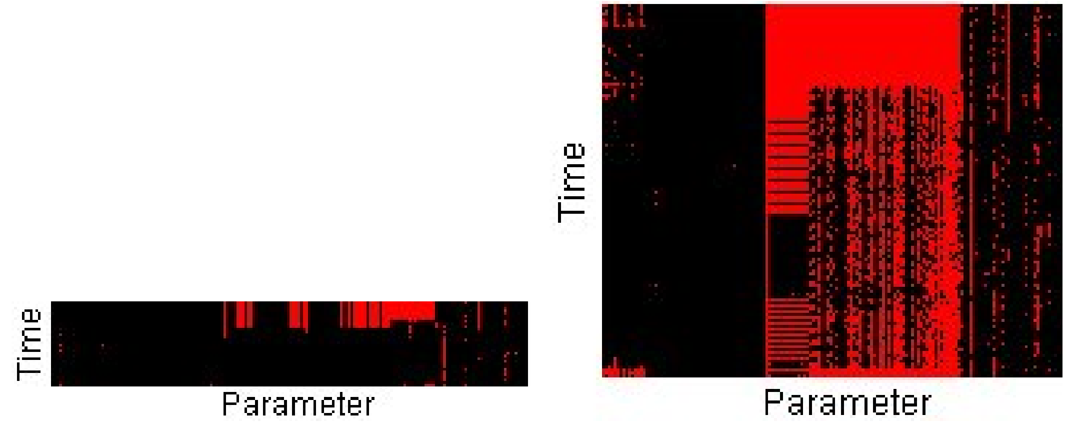
In working with this data, it is useful to define the notion of a “parameter.” “Parameter” is used as a general term for an observation of some biotic or abiotic quantity. *Surface salinity*, which is a measurement of the water’s salt content at the surface, is an example of an abiotic parameter. Biotic parameters include a measure of the abundance of the seabird Auk (*Hydrocoloeus minutus*), or the plankton *Calanus helgolandicus*, for example.

6.2.1 Data collection and processing issues

The problem of unequal effort in data collection was previously mentioned. An illustration of the problem is the effect of an outlier on a sample mean; a mean based upon fewer observations is more heavily influenced by the outlier. If a particular parameter measured temporally has only one or two monthly observations, it is not as reliable as those with twelve monthly measurements. Similarly, the number of measurements over which spatial data is averaged, also varies. A solution to this problem remains elusive, at least until data cataloging the effort spent producing each measurement becomes available in the REGNS database.

6.2.2 Temporal data

Temporal data was extracted from the database at multiple levels of granularity, complicating a temporal analysis. Observations are present for all parameters on a yearly basis. NAO, flux, seabirds, and plankton also have monthly observations, while the finest granularity for which fish CPUE data are available is quarterly. Hence, it is not possible to consider catch per unit effort data in a monthly analysis.



(a) (b)

Figure 6.1: Missing values for (a) yearly and (b) quarterly data. Red indicates missing values and black indicates present

The data is most complete for the years 1973–2004, so the analysis is limited to those years. Some parameters contain observations from the pre-1973 era, but those data are excluded due to the lack of concurrent data from other parameter types. On the other hand, regular observations for some parameters begin post-1973, resulting in missing data values in the tables to be analyzed. Figure 6.1a illustrates the number of missing values present in yearly data, and Figure 6.1b shows the increase in missing values when looking at the data at a finer granularity, in this case monthly. Table 6.2 shows the temporal data available and the time span for which data was used in the analysis.

The North Sea has been divided into five sub-regions and ecological data was extracted from the database separately for each region. Somewhat unintuitive is that this division is only a factor in a temporal analysis. The spatial analysis is performed on data collected over all squares in the North Sea, so all regions are implicitly included. Again, to simply the analysis, the methods and results here are concerned only with data averaged over the entire North Sea.

6.2.3 Spatial data

Not only has the North Sea been divided into regions, but a grid dividing the North Sea into 20km square "ICES statistical squares" has also been defined. These squares span the entire North Sea and the collection of squares for which data has been collected varies among the different parameter types. Purely marine-based species or abiotic parameters (fish CPUE or surface salinity, for example) cannot be observed in squares containing exclusively coastal land. Aligning the entire set of statistical squares with each of the parameter types necessarily involves the inclusion of additional missing values, which were already implicitly present. Figure 6.2 shows a comparison of the missing values present by parameter type.

It is obvious that certain North Sea statistical squares would be home to more fishing activity than others. This affects the observation of plankton, seabirds, and most other biotic species recorded from fishing vessels. It is likely that abiotic measurements are also taken unevenly over the whole of the North Sea. This problem of unequal measurement

Table 6.2: Time period and granularity for which temporal ecosystem data available.

Data type	Time span	Granularity
Abiotic	1973–2004	Yearly
NAO	1973–2004	Monthly
Fish CPUE	1983–2004	Quarterly
Fisheries	1974–2004	Yearly
NORWECOM Flux	1973–2004	Monthly
Seabirds	1980–2004	Monthly
Plankton	1973–2004	Monthly
Landings	1973–2004	Yearly

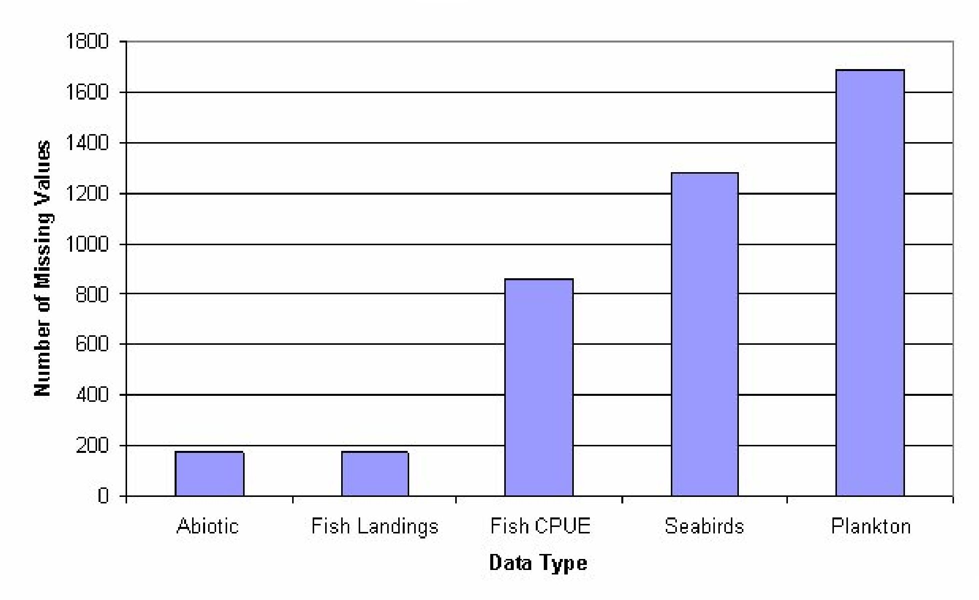


Figure 6.2: Missing values in spatial data by parameter type.

effort is a particular problem with spatial data, as one would expect data collection activities to be more unevenly distributed over space than time.

6.2.4 Parameter-based vs. temporal-spatial analysis

When implementing the similarity computation and network construction described in Chapter 1, it should be noted that there are multiple ways to build a model of ecosystem relationships depending upon the question one wishes to answer. The analysis can be performed from a parameter-based or temporal-spatial standpoint.

In a parameter-based analysis, the goal is to uncover relationships between parameters based upon changes in those parameters in time or space. A temporal-spatial analysis uncovers relationships between discrete time points or locations in the North Sea, respectively. In this sense, the analysis can be approached from three different directions based upon the units between which correlations are computed; biotic and abiotic parameters, time points, or spatial locations. The parameter-based correlations can be derived from temporal, spatial, or a combination of temporal and spatial data.

6.3 Graph-based analysis

Both the parameter-based and temporal-spatial graph analyses of the North Sea ecosystem data proceed as the generalized description in Chapter 1. A graph was constructed with either parameters, time points, or spatial locations represented by vertices. Edge weights were determined by some similarity between the vertices. The large number of missing values present in the data indicates that the number of observations in common between each pair of variables should play a role in computing edge weights. The natural choice is to compute p-values as measures of statistical significance.

Before computing any measure of similarity, it is necessary to recognize that different types of variables are measured in various units and on different scales, so some transformation must be performed to bring the values in comparability with one another. The approach employed here was a simple standardization, which centers the mean of the observations for each parameter at zero with a standard deviation of one.

6.3.1 Parameter-based clique analysis

To elucidate relationships between biotic and abiotic parameters, the ecosystem model was constructed with parameters as vertices and edges assigned weights based upon the statistical significance of the pairwise Pearson's correlation coefficient, measured by the associated p-value. Any correlation based upon fewer than 12 pairwise-complete observations was removed from further analysis. The standard thresholds of $\alpha < 0.05$ and $\alpha < 0.01$ were employed to transform the remainder of the weighted graph into an unweighted graph. Since the significance of the Pearson correlation is dependent upon the sample size, the magnitude of correlation that is considered significant varies for each pair of parameters. If all observations were present, then the correlation threshold was determined to be as low as about $r = 0.349$ and $r = 0.173$ for yearly and quarterly data, respectively. When correlations are computed over time points, and $r = 0.135$ when computed over ICES statistical squares.

Maximal cliques were then enumerated from the unweighted graph. These cliques consist of factors that are changing together in time or space. Since this is an integrated analysis, each clique can be made up of different types of parameters. For example, the clique

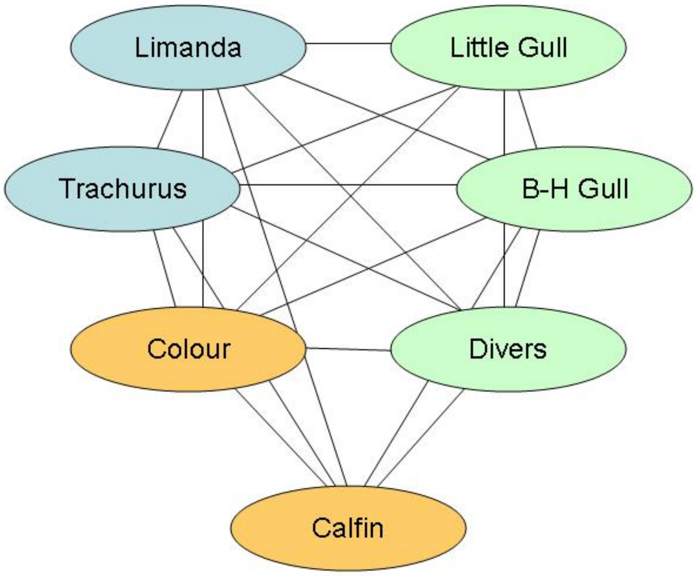


Figure 6.3: An example of a heterogeneous clique enumerated at the $\alpha < 0.05$ threshold. Blue: fish CPUE, green: seabirds, orange: plankton.

shown in Figure 6.3 contains plankton, seabirds, and fish CPUE variables. A legend for the parameters used in this and the remainder of the clique figures in this chapter can be found in Table 6.3.

6.3.2 Temporal-spatial paraclique analysis

As before, a network model was constructed; points in time or space were the variables between which Pearson correlations were calculated. With the large number of parameters available over which to compute correlations in time and space, a paraclique analysis was used to extract dense subsets of highly related time points or spatial locations with the paraclique threshold based upon a raw correlation value cutoff.

Since results on temporal and spatial variables have such a natural visualization, a web-based paraclique tool was developed to allow researchers to investigate relationships in space and time. Computing dense sets of ecosystem relationships in real-time provides a significant computational challenge, met in this case with paraclique codes based upon a very efficient vertex cover-based maximum clique implementation (vc 0.3 by Yun Zhang) on a limited input size.

An interactive tool producing real-time results makes it possible to uncover the different parameter types driving temporal and spatial correlations. By varying the biotic and abiotic conditions used in the analysis, one can quickly uncover latent relationships that might have been obscured in an investigation of the entire data set.

6.3.3 Implementing a web-based temporal-spatial tool

The web tool consists of an HTML and Macromedia Flash interface[†] connecting to custom script-driven back-end maximum clique and paraclique codes. The graphical front end passes user-defined parameters via the HTTP POST protocol to a PHP driver. This script

[†] Flash front end was developed in cooperation with fellow EECS student Gary L. Rogers.

Table 6.3: A legend for biotic and abiotic parameters used in these analyses.

Type	Short name	Description
Fish	Limanda	*Limanda limanda*
	Trachurus	*Trachurus trachurus*
	Ammodytidae	*Ammodytidae*
	Merlangius	*Merlangius merlangus*
	Scomber	*Scomber scombrus*
	Sprattus	*Sprattus sprattus*
	Kitt	*Microstomus kitt*
Seabirds	Little Gull	*Hydrocoloeus minutus*
	Black Headed Gull	*Larus ridibundus*
	Divers	*Gavia*
	Skua	*Stercorarius parasiticus*
	Puffin	*Fratercula arctica*
	Shearwater	*Puffinus puffinus*
Plankton	Calfin	*Calanus finmarchicus*
	Calhel	*Calanus helgolandicus*
	Decap	*Decapoda* total
	Colour	"Greenness"
	Echinol	*Echinoderm* larva
	Dino	*Dinophysis spp.*
	Pp	*Para-Pseudocalanus spp.*
	Proro	*Prorocentrum spp.*
	Dem	
	Totcop	
	Caltot	*Calanus* total traverse
	Cmacro	*Ceratium macroceros*
Abiotic	Btemp	Bottom temperature
	Stemp	Surface temperature

then uses the parameters to choose the data file to be analyzed as well as the subset of data to extract. The selected data items are then loaded into normalization software to bring variables into a comparable range and perform a $\log_2$ transform. Paraclique codes are executed on the normalized data with the parameters such as threshold and "glom factor" chosen by the end user. Points in time or space are converted to graphical coordinates and sent to the GUI for display. For visualization purposes, each paraclique identified in the analysis is displayed in a different color. This allows quick and easy identification of related temporal or spatial points.

The model described here can be modified slightly to accommodate various data types and to perform many types of graph-based analyses. With such a high degree of similarity between historical ecosystem data and other types of biological data, only the method of visualization need be changed to perform a paraclique analysis on gene coexpression data, for example.

6.4 Results

6.4.1 Parameter-based results

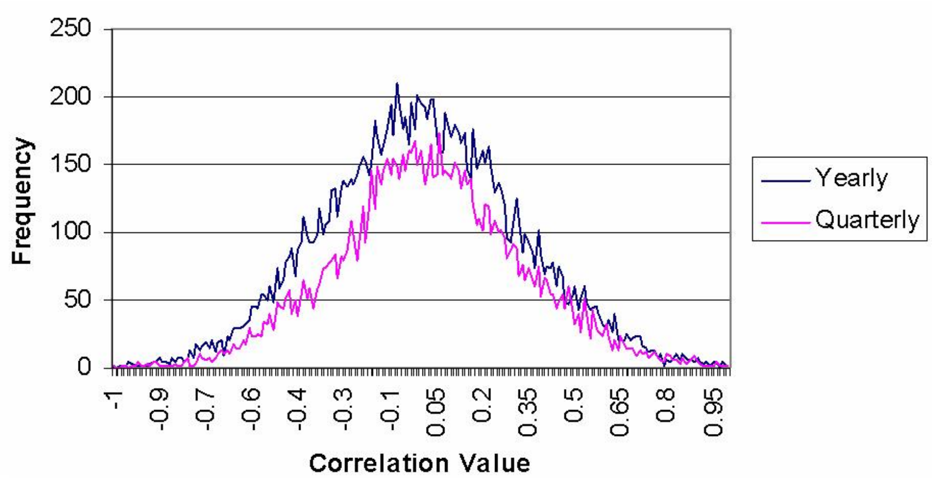
Network models were created for both temporal and spatial data using pairwise Pearson correlation coefficients between only abiotic variables, fish CPUE, seabirds, and plankton data. The distribution of correlation values is illustrated in Figure 6.4. It is clear that correlation distributions of the temporal data, both at yearly and quarterly levels, have well-defined tails, indicating that a good threshold choice likely exists. In contrast, the spatial distribution shows a more square distribution, where it is difficult to define a cutoff between the lower and the highest correlation values.

Unlike clusters produced by traditional clustering algorithms, the maximal cliques enumerated here are overlapping. It is often interesting to examine the parameters lying at the intersection of two or more cliques. Figure 6.5 illustrates two cliques derived from yearly data at the $\alpha < 0.05$ significance level. The one on the left is comprised mostly of fish while the one on the right is dominated by abiotic factors. It is important to note that variables lying exclusively in the left cluster or the right cluster are not well-correlated with those only in the other cluster. Some clustering methods would have forced the fish in the intersection of Figure 6.5 to reside only in the left or right cluster while they are actually highly correlated with parameters in both clusters. In a way, these factors present in both cliques act as a connector between two cliques of various parameter types and as an indicator of changes in their respective observation levels.

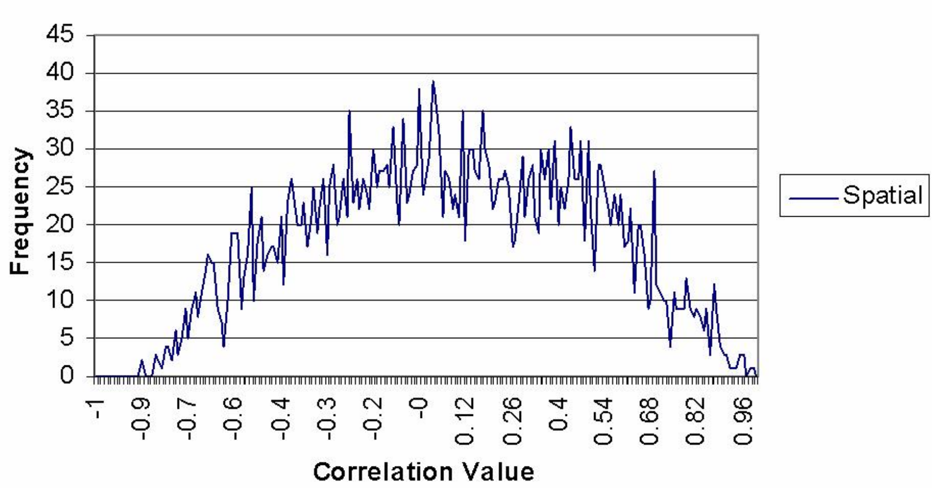
Quarterly data was also analyzed, which produced some larger and more heterogeneous clusters. Maximal cliques were enumerated at the more conservative threshold of $\alpha < 0.01$, producing a large number of overlapping cliques, three of which are depicted in Figure 6.6. It can be seen that plankton dominates the intersection between each pair of cliques, and also the overlapping area of all three cliques. Also present are the bottom and surface temperature variables, indicating that both plankton and temperature link these three particular cliques.

6.4.2 Spatial results

Spatial relationships were extracted from the data using the web-based analysis tool. Abiotic factors, fish landings, fish CPUE, seabirds, and plankton data were used to build the network



(a)



(b)

Figure 6.4: Distribution of correlation coefficients for (a) temporal and (b) spatial data.

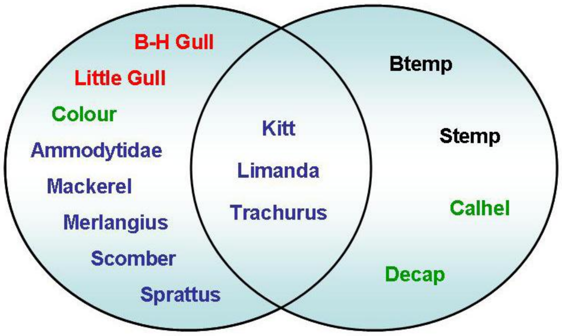


Figure 6.5: Two cliques extracted from yearly relationships illustrating clique overlap. The clique on the left is made up of fish CPUE variables while the one on the right contains mostly abiotic parameters.

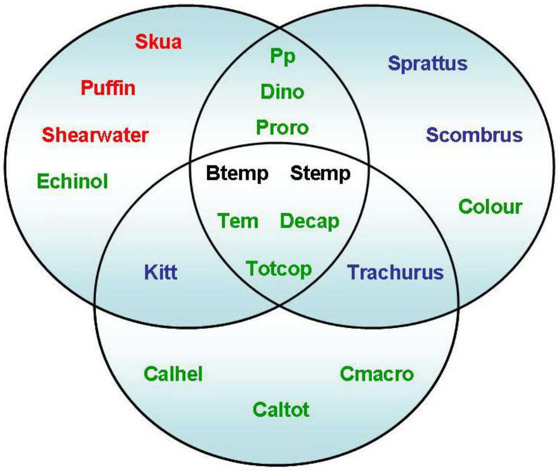
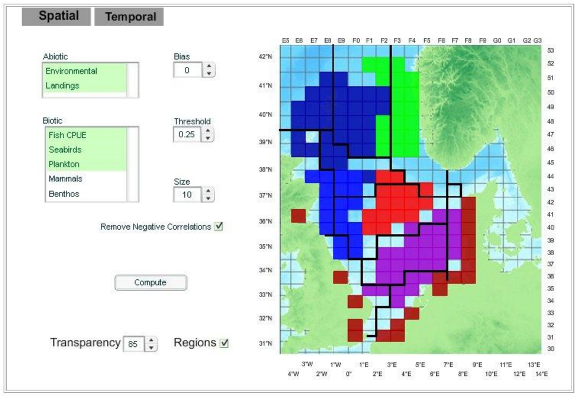


Figure 6.6: Three cliques extracted from the graph on quarterly data. The three cliques were enumerated from a graph constructed at the lower threshold of $\alpha < 0.01$, resulting in more heterogenous cliques.



Abiotic
Environmental
Landings

Bias 0

Biotic
Fish CPUE
Seabirds
Plankton
Mammals
Benthos

Threshold 0.25

Size 10

Remove Negative Correlations ☑

Transparency 85
Regions ☑

Figure 6.7: Ecosystem analysis results on spatial data.

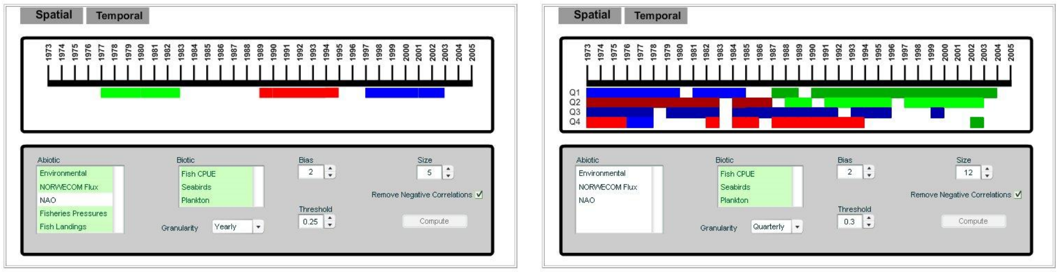
model from which paracliques were computed. Mammal and benthos data were excluded, due to the very incomplete nature of the data. A threshold of 0.25 and a glom factor of 0 were used, which is equivalent to extracting all maximal cliques at the 0.25 threshold. This is useful for finding non-overlapping interactions, which are required for a clean visualization, without decreasing the density of the clusters. For further clarity, any clusters of size less than ten were also removed from the display. Figure 6.7 shows the result of the paraclique computation, where the clusters found roughly coincide with the divisions of the North Sea drawn up by the Oslo and Paris commission based upon the ocean flushing rates [104].

6.4.3 Temporal results

The web-based ecosystem analysis tool was also used to perform a yearly temporal analysis using abiotic, flux, fisheries pressures, fish landings, fish CPUE, seabird, and plankton data. A glom factor of 2 and a paraclique threshold of 0.25 were used, and the resulting clusters were filtered to remove those smaller than five elements. This resulted in three clusters: 1977–1982, 1989–1994, and 1997–2003, illustrated in Figure 6.8a. These clusters are consistent with prior results.

Previously, Weijerman, et al. indicated the possibility of regime shifts in the North Sea based upon the results of a chronological clustering [105]. Similar results were reported in [102]. Kenny, et al. found that there were stable periods in the North Sea during the pre-1983 era and post-1997, but found that the intervening years were dominated by instability. This corresponds with the findings here, but these present results indicate that there might also be another stable period from 1989–1994. This is supported by [105], which found a possible regime change in 1988.

A quarterly analysis was also performed, as seen in Figure 6.8b, but the increased granularity created difficulty in interpretation. Again, a glom factor of 2 was used, this time with a threshold of 0.3 and a minimum clique size of 12. It is interesting to note, however, that quarters three and four are almost always in the same cluster. Quarters one and two are consistently in the same cluster as well, but there seems to be a change around 1987–1988.



(a)

Granularity: Yearly
Bias: 2
Threshold: 0.25
Size: 5
Remove Negative Correlations ☑

(b)

Granularity: Quarterly
Bias: 2
Threshold: 0.3
Size: 12
Remove Negative Correlations ☑

Figure 6.8: Ecosystem analysis results on (a) yearly and (b) quarterly temporal data.

6.5 Conclusions

Challenges arising in the analysis of historical ecosystem data were addressed by considering significance of correlation coefficients in building a network model of ecosystem interactions. Paraclique was also employed in a temporal-spatial analysis to deal with noise and permit the clear visualization of highly overlapping sets of cliques.

In the parameter-based analysis, completely connected sets of interacting variables were extracted, and key sets of biotic and abiotic parameters linking these cliques, plankton and temperature, were identified.

A real-time web-based tool for extracting dense subsets of highly related ecosystem variables was developed and used to perform a temporal and spatial analysis of the North Sea ecosystem. Results were presented that coincide and validate previous findings on post-1973 regime shifts in the North Sea as well as North Sea divisions based upon flushing rates.

6.5.1 Future work

The work here has uncovered many avenues for future investigation. One of the most obvious is expanding the capabilities of the web-based ecosystem analysis tool introduced in Section 6.3.2. Incorporating the parameter-based analysis into this tool is an ideal extension. The present work has also not taken full advantage of the richness of the North Sea ecosystem data that is available. With the incorporation of regional ecosystem data, relationships between spatial locations can be discovered at a finer spatial granularity.

By using the real-time temporal-spatial analysis web tool, it might also be possible to uncover the causes of a regime change in the North Sea. [105] indicated that biological factors contribute more to identifying a putative regime shift in the North Sea, although there may be some contribution by environmental and human-influenced factors. The ecosystem analysis tool described here can be used to vary the parameters used in the analysis and and uncover the variables which may contribute to changes in the North Sea.

Chapter 7

Conclusions

The preceding chapters presented a discussion of some of the challenges that often arise in a graph theoretical analysis of various types of data. Selecting an appropriate threshold, working with data containing many missing values, and uncovering differential relationships from maximal clique results all present unique questions. Although there is often no single best way to deal with the issues that arise, approaches for handling each of these situations were presented and examined.

A key step in constructing gene coexpression networks is determining an appropriate threshold used to identify biologically-significant relationships. A novel threshold selection approach based upon spectral clustering was introduced and shown to produce results that were more conservative and more dependent on the underlying biological data than retaining only the highest or only statistically significant correlations. These methods were examined along with other approaches in a bootstrap analysis [106] of three yeast data sets. It was found that the spectral threshold selection produced thresholds of 0.93, 0.97, and 0.89 for yeast anoxia and reoxygenation [107], and yeast Alpha-factor arrest [61] data sets. These thresholds correspond to maximum clique sizes of 73, 17, and 15, respectively.

Before a threshold selection can be made, pairwise gene similarities must be computed. Chapter 3 examined similarities computed with four methods commonly used in genetic coexpression studies. Correlation and structural properties for graphs on allergy and low dose ionizing radiation microarray data were presented to uncover those properties that are common to biological data sets. Having available a set of correlation and clique profiles expected in biological data can allow researchers to identify data sets deviating from the norm, including those of low quality or erroneous preprocessing. The same information was extracted from scale-free synthetic networks to further identify those properties that are due to biology and not just the scale-free nature of the graphs.

Identifying disease-associated genes was discussed and the differential correlation and differential topology filters were examined to find those genes participating in altered relationships between two groups of samples. These filters were applied to data from a seasonal allergic rhinitis study to identify genes possibly associated with the disease. The effect of various data preprocessing methods on these differential filters had not been examined, so Chapter 5 sought to quantify the similarity in differential results between each of the methods. Significant differences in differential correlation and topology on data from each of the preprocessing methods were identified, although there was a core set of genes passing differential filters for each pair of methods. Biological pathways in which these genes were enriched were examined and it was found that results from two methods are often enriched in some of the same pathways. Particularly, GC-RMA and RMA showed similarity with other methods and with one another on two data sets. Many similarities between methods were found to be largely dependent upon the data set analyzed.

Finally, clique tools were applied to noisy and incomplete historical ecosystem data. It was shown that this data is amenable to such graphical analysis and from it can be produced sets of highly related points in space and time that correspond to previous studies of the North Sea ecosystem. Analysis of various biotic and abiotic ecosystem parameters also identified sets of possibly interacting species, environmental, and man-made factors. A new web-based temporal and spatial analysis tool was developed and discussed, which will make graphical analysis of the North Sea ecosystem readily accessible to researchers in the field.

Directions for future research have been proposed with each chapter. Particularly interesting is the use of the wealth of data found on the Gene Expression Omnibus website [8] to compile a compendium of biological graph properties. Also, considering each step of data preprocessing (background correction, normalization, and expression summarization) separately in a comparison of preprocessing methods would allow for a finer-grained comparison of the methods.

Bibliography

Bibliography

[1] B. H. Voy, J. A. Scharff, A. D. Perkins, A. M. Saxton, B. Borate, E. J. Chesler, L. K. Branstetter, and M. A. Langston. Extracting gene networks for low-dose radiation using graph theoretical algorithms. *PLoS Computational Biology*, 2(7), 2006.

[2] N. E. Baldwin, E. J. Chesler, S. Kirov, and M. A. Langston. Computational, integrative, and comparive methods for the elucidation of genetic coexpression networks. *Journal of Biomedicine and Biotechnology*, 2:172–180, 2005.

[3] J. D. Eblen, I. C. Gerling, A. M. Saxton, J. Wu, J. R. Snoddy, and M. A. Langston. *Graph algorithms for integrated biological analysis, with applications to type 1 diabetes data*, chapter 1. World Scientific, 2008.

[4] A. Fadiel, M. A. Langston, X. Peng, A. D. Perkins, H. S. Taylor, O. Tuncalp, D. Vitello, P. Pevsner, and F. Naftolin. Computational analysis of mass spectrometry data using novel combinatorial methods. In *Proceedigs of the ACS/IEEE International Conference on Computer Systems and Applications*, March 2006.

[5] M. A. Langston, A. D. Perkins, D. J. Beare, R. W. Gauldie, P. J. Kershaw, J. B. Reid, K. Winpenny, and A. J. Kenny. Combinatorial algorithms and high performance implementations for elucidating complex ecosystem relationships from North Sea historical data. In *International Council for the Exploration of the Seas Annual Science Conference*, September 2006.

[6] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*, 270(5235):467–470, 1995.

[7] R. Edgar, M. Domrachev, and A. E. Lash. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, 30(1):207–210, 2002.

[8] http://www.ncbi.nlm.nih.gov/geo/.

[9] G. K. Smyth. Linear models and empirical bayes methods for assessing differential expression in microarray experiments. *Statistical Applications in Genetics and Molecular Biology*, 3(1), 2004.

[10] R. D. Wolfinger, G. Gibson, E. D. Wolfinger, L. Bennett, H. Hamadeh, P. Bushel, C. Afshari, and R. S. Paules. Assessing gene significance from cDNA microarray expression data via mixed models. *Journal of Computational Biology*, 8(6), 2001.

[11] G. K. Smyth and T. Speed. Normalization of cDNA microarray data. *Methods*, 31(4), 2003.

[12] A. I. Su, T. Wiltshire, S. Batalov, H. Lapp, et al. A gene atlas of the mouse and human protein-encoding transcriptomes. *Proceedings of the National Academy of Sciences*, 101(16):6062–6067, 2004.

[13] F. F. Millenaar, J. Okyere, S. T. May, M. van Zanten, L. A. C. J. Voesenek, and A. J. M. Peeters. How to decide? different methods of calculating gene expression from short oligonucleotide array data will give different results. *BMC Bioinformatics*, 7(137), 2006.

[14] R. Lowry. *Concepts and Applications of Inferential Statistics*. Online Book, 2008.

[15] J. H. Zar. Significance testing of the Spearman rank correlation coefficient. *Journal of the American Statistical Association*, 67(339):1972, 578–580.

[16] L. L. Elo, H. Järvenpää, M. Orešič, R. Lahesmaa, and T. Aittokallio. Systematic construction of gene coexpression networks with applications to human T helper cell differentiation process. *Bioinformatics*, 23(16):2096–2103, 2007.

[17] E. J. Chesler and M. A. Langston. Combinatorial genetic regulatory network analysis tools for high throughput transcriptomic data. In *RECOMB Satellite Workshop on Systems Biology and Regulatory Genomics*, December 2005.

[18] Y. Zhang, E. J. Chesler, and M. A. Langston. On finding bicliques in bipartite graphs: a novel algorithm with application to the integration of diverse biological data types. In *Proceedings of the Hawaii International Conference on System Sciences*, January 2008.

[19] F. Dehne, M. A. Langston, X. Luo, S. Pitre, P. Shaw, and Y. Zhang. The cluster editing problem: implementation and experiments. In *Proceedings of the International Workshop on Parameterized and Exact Computation*, September 2006.

[20] M. Fellows, M. Hallett, C. Korostensky, and U. Stege. Analogs & duals of the MAST problem for sequences & trees. In *Proceedings of the Sixth Annual European Symposium on Algorithms*, August 1998.

[21] M. A. Langston, L. Lin, X. Peng, N. E. Baldwin, C. T. Symons, and B. Zhang. A combinatorial approach to the analysis of differential gene expression data: the use of graph algorithms for disease prediction and screening. In *Proceedings of the International Conference for the Critical Assemeessment of Microarray Data Analysis*, November 2003.

[22] M. R. Garey and D. S. Johnson. *Computers and Intractability: a Guide to the Theory of NP-Completeness*. Series of Books in the Mathematical Sciences. W. H. Freeman, 1979.

[23] A. J. Butte, P. Tamayo, D. Slonim, T. R. Golub, and I. S. Kohane. Discovering functional relationships between RNA expression and chemotherapeutic suseptibility using relevance networks. *Proceedings of the National Academy of Sciences*, 97(22):12182–12186, 2000.

[24] C. J. Wolfe, I. S. Kohane, and A. J. Butte. Systematic survey reveals general applicability of "guilt-by-association" within gene coexpression networks. *BMC Bioinformatics*, 6(227), 2005.

[25] The Gene Ontology Consortium. Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25:25–29, 2000.

[26] B. Zhang, S. Kirov, and J. Snoddy. WebGestalt: an integrated system for exploring gene sets in various biological contexts. *Nucleic Acids Research*, 33:W741–W748, 2005.

[27] http://www.ingenuity.com/.

[28] I. M. Bomze, M. Budinich, P. M. Pardalos, and M. Pelillo. *The maximum clique problem*, pages 1–74. Kluwer Academic Publishers, Dordrect, The Netherlands, 1999.

[29] R. G. Downey and M. R. Fellows. *Parameterized Complexity.* Monographs in Computer Science. Springer, 1998.

[30] J. Chen, I. Kanj, and W. Jia. Vertex cover: further observations and further improvements. *Journal of Algorithms*, 41:280–301, 2001.

[31] F. N. Abu-Khzam, M. A. Langston, P. Shanbhag, and C. T. Symons. Scalable parallel algorithms for FPT problems. *Algorithmica*, 45(3):269–284, 2006.

[32] U. Ala, R. M. Piro, E. Grassi, C. Damasco, L. Silengo, M. Oti, P. Provero, and F. Di Cunto. Prediction of human disease genes by human-mouse conserved coexpression analysis. *PLOS Computational Biology*, 4(3):e1000043, 2008.

[33] J. Ruan and W. Zhang. Identification and evaluation of functional modules in gene co-expression networks. In *Proceedings of RECOMB Satellite Conferences on Systems Biology and Computational Proteomics.* Springer, December 2006.

[34] H. K. Lee, A. K. Hsu, J. Sajdak, J. Qin, and P. Pavlidis. Coexpression analysis of human genes across many microarray data sets. *Genome Research*, 14:1085–1094, 2004.

[35] M. Moriyama, Y. Hoshida, M. Otsuka, S. Nishimura, N. Kato, T. Goto, H. Taniguchi, Y. Shiratori, N. Seki, and M. Omata. Relevance network between chemosensitivity and transcriptome in human hepatoma cells. *Molecular Cancer Therapeutics*, 2:199–205, 2003.

[36] T. C. Freeman, L. Goldovsky, M. Brosch, S. van Dongen, P. Mazière, R. J. Grocock, S. Freilich, J. Thornton, and A. J. Enright. Construction, visualization, and clustering of transcription networks from microarray expression data. *PLOS Computational Biology*, 3(10):e206, 2007.

[37] F. R. K. Chung. *Spectral Graph Theory*, volume 92 of *Regional Conference Series in Mathematics.* American Mathematical Society, 1994.

[38] L. W. Beineke and R. J. Wilson, editors. *Topics in Algebraic Graph Theory.* Cambridge University Press, 2004.

[39] D. Cvetković, P. Rowlinson, and S. Simić. *Eigenspaces of Graphs.* Encyclopedia of Mathematics. Cambridge University Press, 1997.

[40] D. Cvetković, M. Doob, and H. Sachs. *Spectra of Graphs: Theory and Application*, volume 87 of *Pure and Applied Mathematics.* Academic Press, 1979.

[41] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[42] D. A. Tolivar and G. L. Miller. Graph partitioning by spectral rounding: applications in image segmentation and clustering. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1053–1060, 2006.

[43] F. Comellas and S. Gago. A star-based model for the eigenvalue power law of internet graphs. *Physica A*, 351:680–686, 2005.

[44] C. Gkantsidis, M. Mihail, and E. Zegura. Spectral analysis of internet topologies. In *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies*, 2003.

[45] D. Cvetković. *Characterizing properties of some graph invariants related to the electron charges in the Hückel molecular orbital theory*, volume 51 of *DIMACS Series on Discrete Math and Theoretical Computer Science*, pages 79–84. American Mathematical Society, 2002.

[46] R. B. King. *Applications of Graph Theory and Topology in Inorganic Cluster and Coordination Chemistry.* CRC Press, 1993.

[47] I. V. Stankevich, M. I. Skvortsova, V. A. Kolmykov, V. F. Subbotin, and V. B. Mnukhin. *Spectral graph theory in chemistry*, chapter 2. Mathematics methods in contemporary chemistry. CRC Press, 1996.

[48] D. Tritchler, S. Fallah, and J. Beyenea. A spectral clustering method for microarray data. *Computational Statistics and Data Analysis*, 49:63–76, 2005.

[49] Y. Kluger, R. Basri, J. T. Chang, and M. Gerstein. Spectral biclustering of microarray data: coclustering genes and conditions. *Genome Research*, 13:703–716, 2003.

[50] R. B. Lehoucq and D. C. Sorensen. Deflation techniques for an implicityly re-started Arnoldi iteration. *SIAM Journal of Matrix Analysis and Applications*, 17:789–821, 1996.

[51] R. B. Lehoucq, D. C. Sorensen, and C. Yang. *ARPACK Users' Guide: Solution of large-scale eigenvalue problems with implicityly restarted Arnoldi methods.* SIAM Publications, Philadelphia, PA, 1998.

[52] M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(98):298–305, 1973.

[53] S. White and P. Smyth. A spectral clustering approach to finding communities in graphs. In *Proceedings of the SIAM International Conference on Data Mining*, 2005.

[54] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: analysis and an algorithm. In *Proceedings of the 14th Advances in Neural Information Processing Systems*, 2001.

[55] D. Verma and M. Meilă. A comparison of spectral clustering algorithms. Technical Report 03-0501, University of Washington Department of Computer Science, Seattle, WA, 2003.

[56] R. Boppana. Eigenvalues and graph bisection: an average-case analysis. In *Proceedings of the 28th IEEE Annual Symposium on Foundations of Computer Science*, pages 280–285, 1987.

[57] A. Pothen, D. H. Simon, and K. P. Liou. Paritioning sparse matrices with eigenvectors of graphs. *SIAM Journal on Matrix Analysis and Applications*, 11(3):430–452, 1990.

[58] C. H. Q. Ding, X. He, and H. Zha. A spectral method to separate disconnected and nearly disconnected Web graph components. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, August 2001.

[59] A. Barabási and Z. N. Oltvai. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.

[60] A. M. Saxton. Personal communication, 2008.

[61] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher. Comprehensive identification of cell cycle-regulated gene of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular Biology of the Cell*, 9:3273–3297, December 1998.

[62] R. Albert and A. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97, 2002.

[63] J. Schäfer and K. Strimmer. A shrinkage approach to large-scale covariance estimation and implications for function genomics. *Statistical Applications in Genetics and Molecular Biology*, 4(32), 2005.

[64] A. Butte and I. Kohane. Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. In *Pacific Symposium on Biocomputing*, volume 5, pages 415–426, January 2000.

[65] R. Steuer, J. Jurths, C. O. Daub, J. Weise, and J. Selbig. The mutual information: detecting and evaluating dependencies between variables. *Bioinformatics*, 18(Suppl. 2):S231–S240, 2002.

[66] I. Priness, O. Maimon, and I. Ben-Gal. Evaluation of gene-expression clustering via mutual information distance measure. *BMC Bioinformatics*, 8(111), 2007.

[67] J. Schäfer, R. Opgen-Rhein, and K. Strimmer. *corpcor: efficient estimation of covariance and (partial) correlation*, 2007. R package version 1.4.7.

[68] R. Gottardo. *EMV: estimation of missing values for a data matrix.* R package version 1.3.1.

[69] B. Ding, R. Gentleman, and V. Carey. *bioDist: different distance measures*. R package version 1.10.0.

[70] R. Mobini, B. Andersson, L. O. Cardell, B. S. Egan, M. Hahn-Zoric, M. Langston, I. Oancea, A. Perkins, J. Soini, S. Rak, and M. Benson. Combined transcriptional profiling and promoter tiling arrays show key role of IRF4 in Th2 cell proliferation in seasonal allergic rhinitis. *Cytokine*, 2007. Submitted for publication.

[71] E. Harley, A. Bonner, and N. Goodman. Uniform integration of genome mapping data using intersection graphs. *Bioinformatics*, 17(6):487–494, 2001.

[72] B. Harr and C. Schlötterer. Comparison of algorithms for the analysis of Affymetrix microarray data as evaluated by co-expression of genes in known operons. *Nucleic Acids Research*, 34(2):e8, 2006.

[73] R. A. Irizarry, Z. Wu, and H. A. Jaffee. Comparison of Affymetrix GeneChip expression measures. *Bioinformatics*, 22(7):789–794, 2006.

[74] R. A. Irizarry, B. Hobbs, F. Collin, Y. D. Beazer-Barclay, K. J. Antoellis, U. Scherf, and T. P. Speed. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, 4(2):249–264, 2003.

[75] H. Binder and S. Preibisch. Specific and nonspecific hybridization of oligonucleotide probes on microarrays. *Biophysical Journal*, 89(1):337–352, 2005.

[76] Affymetrix, Inc. *Statistical Algorithms Description Document*, 2003.

[77] Y. Wang, Z. Miao, Y. Pommier, E. S. Kawasaki, and A. Player. Characterization of mismatch and high-signal intensity probes associated with Affymetrix genechips. *Bioinformatics*, 23(16):2088–2095, 2007.

[78] R. A. Irizarry, B. M. Bolstad, F. Collin, and L. M. Cope. Summaries of Affymetrix GeneChip probe level data. *Nucleic Acids Research*, 31(4):e15, 2003.

[79] B. M. Bolstad. *Low-level analysis of high-density oligonucleotide array data: background, normalization, and summarization.* PhD thesis, University of California, Berkeley, 2004.

[80] Z. Wu, R. A. Irizarry, R. Gentleman, F. Martinez-Murillo, and F. Spencer. A model-based background adjustment for oligonucleotide expression arrays. *Journal of the American Statistical Association*, 99, 2004.

[81] Z. Wu and R. A. Irizarry. Preprocessing of oligonucleotide array data. *Nature Biotechnology*, 22:656–658, 2004.

[82] L. Zhang, M. F. Miles, and K. D. Aldape. A model of molecular interactions on short oligonucleotide microarrays. *Nature Biotechnology*, 21(7):818–821, 2003.

[83] C. Li and W. H. Wong. Model-based analysis of oligonucleotide arrays: expression index computation and outlier detection. *Proceedings of the National Academy of Sciences*, 98(1):31–36, 2001.

[84] C. Li and W. H. Wong. Model-based analysis of oligonucleotide arrays: model validation, design issues and standard error application. *Genome Biology*, 2(8), 2001.

[85] *dChip User's Manual.* http://biosun1.harvard.edu/complab/dchip/manual.htm.

[86] W. Huber, A. von Heydebreck, H. Sültmann, A. Poustka, and M. Vingron. Variance stablization applied to microarray data calibration and to the quantification of differential expression. *Bioinformatics*, 18(S1):S96–S104, 2002.

[87] W. Huber, A. von Heydebreck, H. Sültmann, A. Poustka, and M. Vington. Parameter estimation for the calibration and variance stabilization of microarray data. *Statistical Applications in Genetics and Molecular Biology*, 2(1), 2003.

[88] R. A. Irizarry, L. Gautier, B. M. Bolstad, C. Miller with contributions from M. Astrand, L. M. Cope, R. Gentleman, J. Gentry, C. Halling, W. Huber, J. MacDonald, B. I. P. Rubinstein, C. Workman, and J. Zhang. *affy: methods for affymetrix oligonucleotide arrays*, 2006. R package version 1.12.2.

[89] J. Wu, R. Irizarry with contributions from J. MacDonald, and J. Gentry. *gcrma: background adjustment using sequence information.* R package version 2.6.0.

[90] Li Zhang, Haitao Zhao, James Mitchell, and Clift Norris. *PerfectMatch Manual.* The University of Texas MD Anderson Cancer Center, Houston, TX, October 2004.

[91] C. Li and W. H. Wong. *DNA-Chip Analyzer (dChip)*, pages 120–141. The analysis of gene expression data: methods and software. Springer, New York, 2003.

[92] K. Shedden, W. Chen, R. Kuick, D. Ghosh, J. Macdonald, K. R. Cho, T. J. Giordano, S. B. Gruber, E. R. Fearon, J. M. G. Taylor, and S. Hanash. Comparison of seven methods for producing Affymetrix expression scores based on false discovery rates in disease profiling data. *BMC Bioinformatics*, 6(26), 2005.

[93] C. C. Barbacioru, Y. Wang, R. D. Canales, Y. A. Sun, D. N. Keys, F. Chan, K. A. Poulter, and R. R. Samaha. Effect of various normalization methods on Applied Biosystems expression array system data. *BMC Bioinformatics*, 7(533), 2006.

[94] J. Quackenbush. Microarray data normalization and transformation. *Nature Genetics Supplement*, 32:496–501, 2002.

[95] Affymetrix, Inc. *GeneChip Expression Analysis Technical Manual*, 2004.

[96] R. M. Simon, E. L. Korn, L. M. McShane, M. D. Radmacher, G. W. Wright, and Y. Zhao. *Design and Analysis of DNA Microarray Investigations.* Statistics for Biology and Health. Springer, 2003.

[97] R. S. Stearman, L. Dwyer-Nield, L. Zerbe, S. A. Blaine, et al. Analysis of orthologous gene expression between human pulmonary adenocarcinoma and a carcinogen-induced murine model. *American Journal of Pathology*, 167(6):1763–1775, 2005.

[98] M. M. Ryan, H. E. Lockstone, S. J. Huffaker, M. T. Wayland, et al. Gene expression analysis of bipolar disorder reveals downregulation of the ubiquitin cycle and alterations in synaptic genes. *Molecular Psychiatry*, 11(10):965–978, 2006.

[99] J. Sabates-Bellver, L. G. Van der Flier, M. de Palo, E. Cattaneo, C. Maake, H. Rehrauer, E. Laczko, M. A. Jurowski, J. M. Bujnicki, M. Menigatti, J. Luz, T. V. Ranalli, V. Gomes, A. Pastorelli, R. Faggiani, M. Anti, J. Juricny, H. Clevers, and

G. Marra. Transcriptome profile of human colorectal adenomas. *Molecular Cancer Research*, 5(12):1263–1275, 2007.

[100] J. Seo, M. Bakay, Y. Chen, S. Hilmer, B. Shneiderman, and E. P. Hoffman. Interactively optimizing signtal-to-noise ratios in expression profiling: Project-specific algorithm selection and detection p-value weighting in Affymetrix microarrays. *Bioinformatics*, 20(16):2534–2544, 2004.

[101] K. R. Clarke and R. N. Gorley. *PRIMER v6: User manual/tutorial.* Plymouth, UK. 192pp.

[102] A. J. Kenny, P. Kershaw, D. Beare, M. Devlin, J. B. Reid, P. Licandro, A. Gallego, K. Winpenny, C. Houghton, M. Langston, H. R. Skjoldal, and A. Perkins. Integrated assessment of the North Sea to identify the relationships between human pressures and ecosystem state changes implications for marine management. In *International Council for the Exploration of the Seas Annual Science Conference*, September 2006.

[103] Regional Ecosystem Study Group of the North Sea. *Report of the Regional Ecosystem Study Group of the North Sea (REGNS).* ICES Headquarters, Copenhagen, Denmark, May 2006.

[104] North Sea Task Force. North Sea quality status report. Oslo and Paris Commissions, 1993.

[105] M. Wijerman, H. Lindeboom, and A. F. Zuur. Regime shifts in marine ecosystems of the North Sea and Wadden Sea. *Marine Ecology Progress Series*, 298:21–39, 2005.

[106] B. Borate. Comparative analysis of thresholding algorithms for microarray-derived gene correlation matrices. Master's thesis, The University of Tennessee, 2008.

[107] L. Lai, A. L. Kosorukoff, P. V. Burke, and K. E. Kwast. Metabolic-state-dependent remodeling of the transcriptome in response to anoxia and subsequent reoxygenation in saccharomyces cerevisiae. *Eukaryotic Cell*, 5(9):1468–1489, 2006.

Appendix

Appendix A

Identifying Stress- and Disease-Associated Genes From Graph-Based Models

Table A.1: Basic graph structure metrics for allergy graphs.

	Control				Stimulated				Treated			
Threshold	Vertices	Edges	Maximal	Maximum	Vertices	Edges	Maximal	Maximum	Vertices	Edges	Maximal	Maximum
0.90	8487	104371	543751	67	7512	143307	70739849	99	8577	129866	4041857	135
0.91	7785	74534	181619	59	7029	123416	28974733	90	8164	112385	2022935	128
0.92	6892	51240	63465	52	6031	88875	7246993	76	7190	82124	487737	117
0.93	5907	33526	24514	45	4949	61569	1683893	61	6097	58532	141982	107
0.94	4802	20701	11404	37	3911	40664	206289	49	4894	40513	57091	92
0.95	3631	11815	6736	28	2925	25135	49352	38	3593	27093	29084	80
0.96	2456	6124	2505	22	2067	14168	11096	27	2322	17630	11840	67
0.97	1401	2731	818	15	1304	6917	2957	21	1382	10848	7155	53
0.98	580	908	225	9	714	2547	719	16	695	5615	3458	33
0.99	153	156	22	5	293	542	129	9	308	1752	730	14

Appendix B

Effects of Preprocessing Method on Correlation, Graph Structure, and Differential Comparisons

For Tables B.1, B.2, and B.3, J indicates the Jaccard similarity between each pair of methods.

Table B.1: Similarity in the top one hundred most connected genes for bipolar disorder data.

Normal				Disease			
Threshold	Method 1	Method 2	J	Threshold	Method 1	Method 2	J
0.85	PDNN	RMA	1.00	0.85	GC-RMA	VSN	1.00
0.85	PDNN	VSN	1.00	0.85	PDNN	RMA	1.00
0.85	RMA	VSN	1.00	0.85	RMA	VSN	1.00
0.86	PDNN	RMA	1.00	0.86	GC-RMA	MAS5.0	1.00
0.86	PDNN	VSN	1.00	0.86	GC-RMA	VSN	1.00
0.86	RMA	VSN	1.00	0.86	PDNN	RMA	1.00
0.87	PDNN	RMA	1.00	0.86	PDNN	VSN	1.00
0.87	PDNN	VSN	1.00	0.86	RMA	VSN	1.00
0.87	RMA	VSN	1.00	0.87	GC-RMA	MAS5.0	1.00
0.88	PDNN	RMA	1.00	0.87	GC-RMA	VSN	1.00
0.88	PDNN	VSN	1.00	0.87	PDNN	RMA	1.00
0.88	RMA	VSN	1.00	0.87	PDNN	VSN	1.00
0.89	PDNN	RMA	1.00	0.87	RMA	VSN	1.00
0.89	PDNN	VSN	1.00	0.88	GC-RMA	VSN	1.00
0.89	RMA	VSN	1.00	0.88	PDNN	RMA	1.00
0.90	PDNN	RMA	1.00	0.88	PDNN	VSN	1.00
0.90	PDNN	VSN	1.00	0.88	RMA	VSN	1.00
0.90	RMA	VSN	1.00	0.89	GC-RMA	VSN	1.00
0.91	PDNN	VSN	1.00	0.89	PDNN	RMA	1.00

continued on next page

continued from previous page

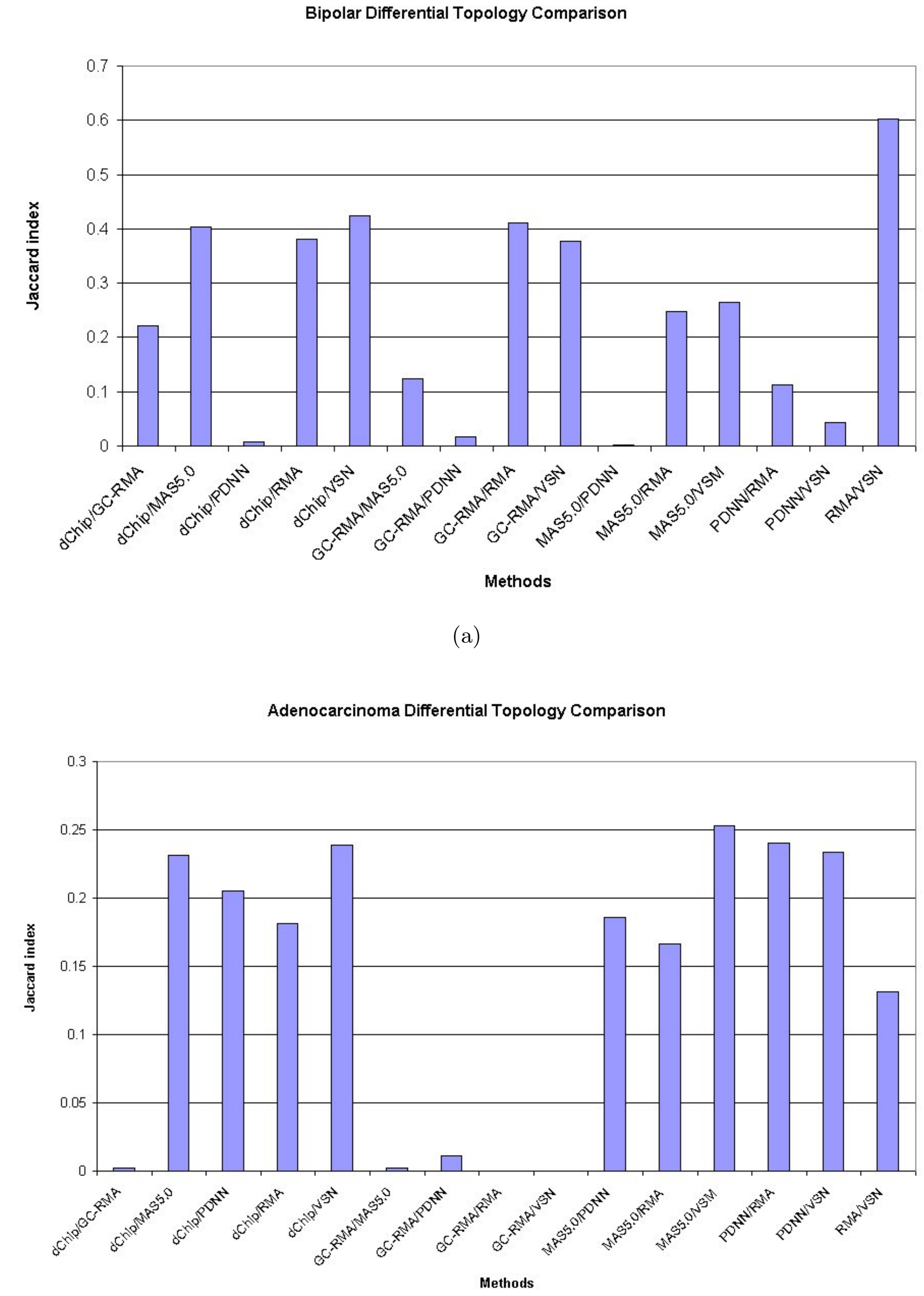
Normal				Disease			
Threshold	Method 1	Method 2	*J*	Threshold	Method 1	Method 2	*J*
0.91	RMA	VSN	1.00	0.89	PDNN	VSN	1.00
0.92	PDNN	VSN	1.00	0.89	RMA	VSN	1.00
0.92	RMA	VSN	1.00	0.90	GC-RMA	MAS5.0	1.00
0.93	PDNN	VSN	1.00	0.90	PDNN	RMA	1.00
0.94	PDNN	VSN	1.00	0.90	PDNN	VSN	1.00
0.91	PDNN	RMA	0.98	0.90	RMA	VSN	1.00
0.92	PDNN	RMA	0.98	0.91	PDNN	RMA	1.00
0.93	PDNN	RMA	0.98	0.91	PDNN	VSN	1.00
0.93	RMA	VSN	0.98	0.91	RMA	VSN	1.00
0.94	PDNN	RMA	0.98	0.92	PDNN	RMA	1.00
0.95	PDNN	RMA	0.98	0.92	PDNN	VSN	1.00
0.85	GC-RMA	MAS5	0.96	0.92	RMA	VSN	1.00
0.86	GC-RMA	MAS5	0.96	0.93	PDNN	RMA	1.00
0.87	GC-RMA	MAS5	0.96	0.93	PDNN	VSN	1.00
0.94	RMA	VSN	0.96	0.93	RMA	VSN	1.00
0.95	PDNN	VSN	0.96	0.94	RMA	VSN	1.00
0.88	GC-RMA	MAS5	0.94	0.85	GC-RMA	MAS5.0	0.98
0.89	GC-RMA	MAS5	0.92	0.85	PDNN	VSN	0.98
0.90	GC-RMA	MAS5	0.90	0.88	GC-RMA	MAS5.0	0.98
0.96	PDNN	RMA	0.90	0.89	GC-RMA	MAS5.0	0.98
				0.90	GC-RMA	VSN	0.96
				0.91	GC-RMA	MAS5.0	0.96
				0.91	GC-RMA	VSN	0.96
				0.93	GC-RMA	MAS5.0	0.96
				0.94	PDNN	RMA	0.96
				0.94	PDNN	VSN	0.96
				0.85	dChip	MAS5.0	0.94
				0.86	dChip	MAS5.0	0.94
				0.92	GC-RMA	MAS5.0	0.94
				0.95	PDNN	RMA	0.94
				0.96	PDNN	RMA	0.94
				0.85	GC-RMA	RMA	0.92
				0.86	GC-RMA	RMA	0.92
				0.95	PDNN	VSN	0.92
				0.87	dChip	MAS5.0	0.90
				0.87	GC-RMA	RMA	0.90
				0.89	GC-RMA	RMA	0.90
				0.95	RMA	VSN	0.90

Table B.2: Similarity in the top one hundred most connected genes for pulmonary adenocarcinoma data.

Normal				Disease			
Threshold	Method 1	Method 2	J	Threshold	Method 1	Method 2	J
0.85	PDNN	VSN	0.92	0.90	PDNN	VSN	0.98
0.86	PDNN	VSN	0.92	0.91	PDNN	VSN	0.98
0.87	PDNN	VSN	0.92	0.91	RMA	VSN	0.98
0.88	PDNN	VSN	0.92	0.92	PDNN	VSN	0.98
0.89	PDNN	VSN	0.92	0.92	RMA	VSN	0.98
0.94	PDNN	VSN	0.92	0.93	PDNN	VSN	0.98
0.95	PDNN	VSN	0.92	0.93	RMA	VSN	0.98
0.90	PDNN	VSN	0.90	0.94	PDNN	VSN	0.98
0.92	PDNN	VSN	0.90	0.88	PDNN	VSN	0.96
				0.89	PDNN	VSN	0.96
				0.90	RMA	VSN	0.96
				0.89	RMA	VSN	0.94
				0.94	RMA	VSN	0.94
				0.95	PDNN	VSN	0.94
				0.85	PDNN	VSN	0.92
				0.87	PDNN	VSN	0.92
				0.86	PDNN	VSN	0.90
				0.95	RMA	VSN	0.90
				0.96	PDNN	VSN	0.90
				0.96	RMA	VSN	0.90

Table B.3: Similarity in the top one hundred most connected genes for colorectal adenoma data.

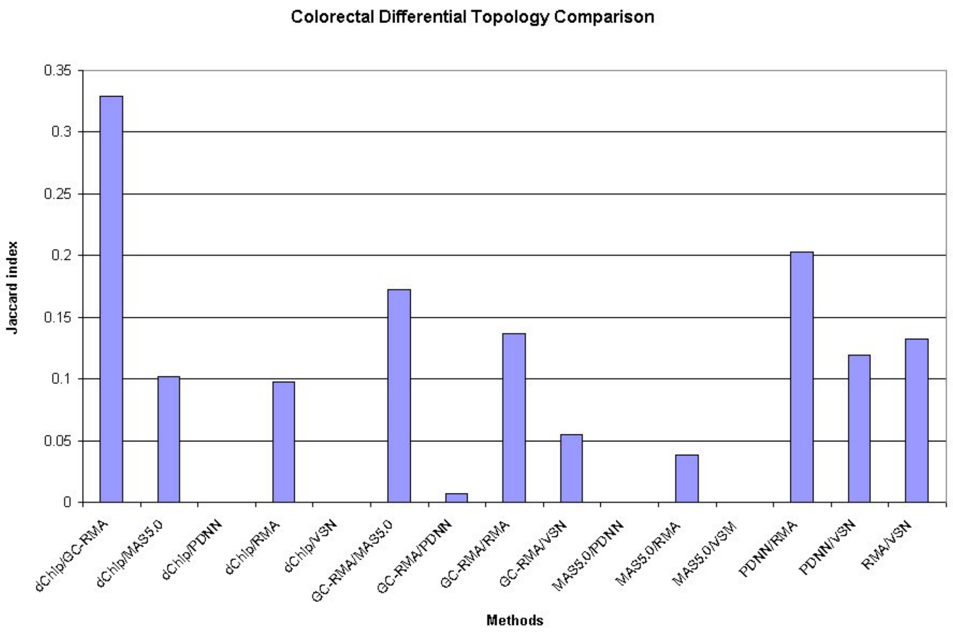
Normal				Disease			
Threshold	Method 1	Method 2	J	Threshold	Method 1	Method 2	J
0.85	PDNN	RMA	1.00	0.93	GC-RMA	RMA	1.00
0.86	PDNN	RMA	1.00	0.93	PDNN	RMA	1.00
0.87	PDNN	RMA	1.00	0.94	PDNN	RMA	1.00
0.88	PDNN	RMA	1.00	0.98	PDNN	RMA	1.00
0.89	PDNN	RMA	1.00	0.89	PDNN	RMA	0.98
0.90	PDNN	RMA	1.00	0.90	PDNN	RMA	0.98
0.93	PDNN	RMA	1.00	0.91	PDNN	RMA	0.98
0.96	PDNN	RMA	1.00	0.92	PDNN	RMA	0.98
0.91	PDNN	RMA	0.98	0.95	GC-RMA	RMA	0.98
0.92	PDNN	RMA	0.98	0.95	PDNN	RMA	0.98
0.94	PDNN	RMA	0.98	0.96	PDNN	RMA	0.98
0.90	GC-RMA	RMA	0.96	0.97	PDNN	RMA	0.98
0.93	GC-RMA	RMA	0.96	0.86	PDNN	RMA	0.96
0.95	GC-RMA	RMA	0.96	0.87	PDNN	RMA	0.96
0.95	PDNN	RMA	0.96	0.88	PDNN	RMA	0.96
0.97	PDNN	RMA	0.96	0.94	GC-RMA	RMA	0.96
0.92	GC-RMA	RMA	0.94	0.85	PDNN	RMA	0.94
0.88	GC-RMA	RMA	0.92	0.92	GC-RMA	RMA	0.94
0.91	GC-RMA	RMA	0.92	0.96	GC-RMA	RMA	0.94
0.94	GC-RMA	RMA	0.92	0.98	GC-RMA	RMA	0.92
0.96	GC-RMA	RMA	0.92	0.97	GC-RMA	RMA	0.90
0.89	GC-RMA	RMA	0.90	0.99	PDNN	RMA	0.90
0.97	GC-RMA	RMA	0.90				



(a)

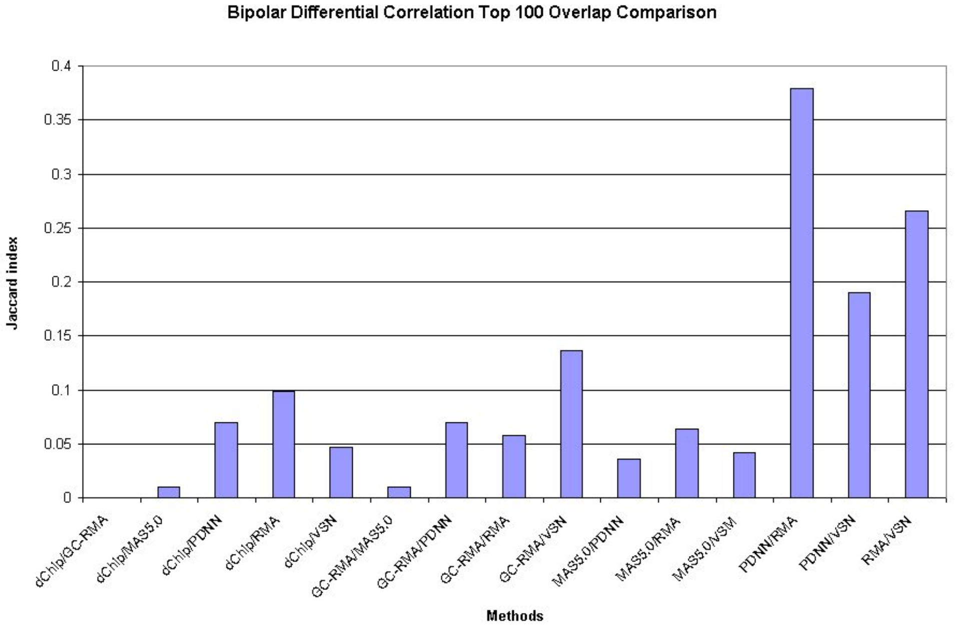
(b)

Figure B.1: Differential correlation similarities for bipolar disorder, pulmonary adenocarcinoma, and colorectal adenoma data. (a) Bipolar disorder comparison, (b) Pulmonary adenocarcinoma comparison, (c) Colorectal adenoma comparison.

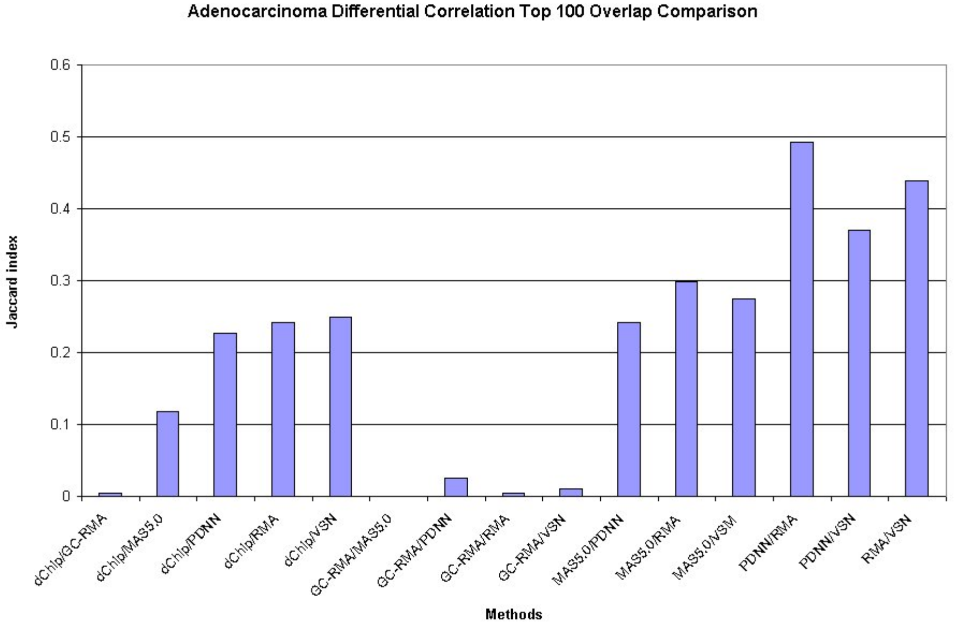


(c)

Figure B.1: Continued.

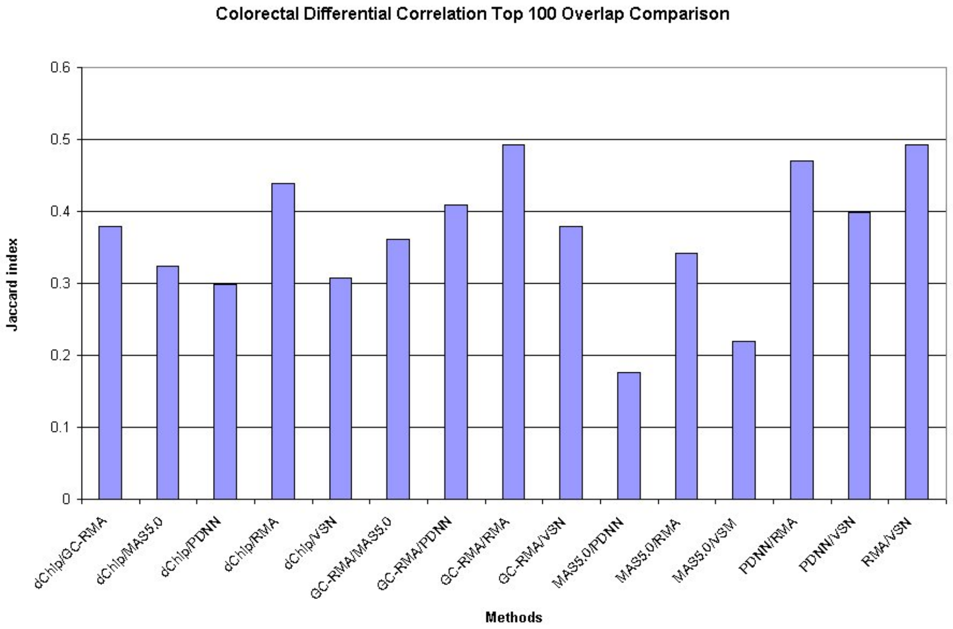


(a)



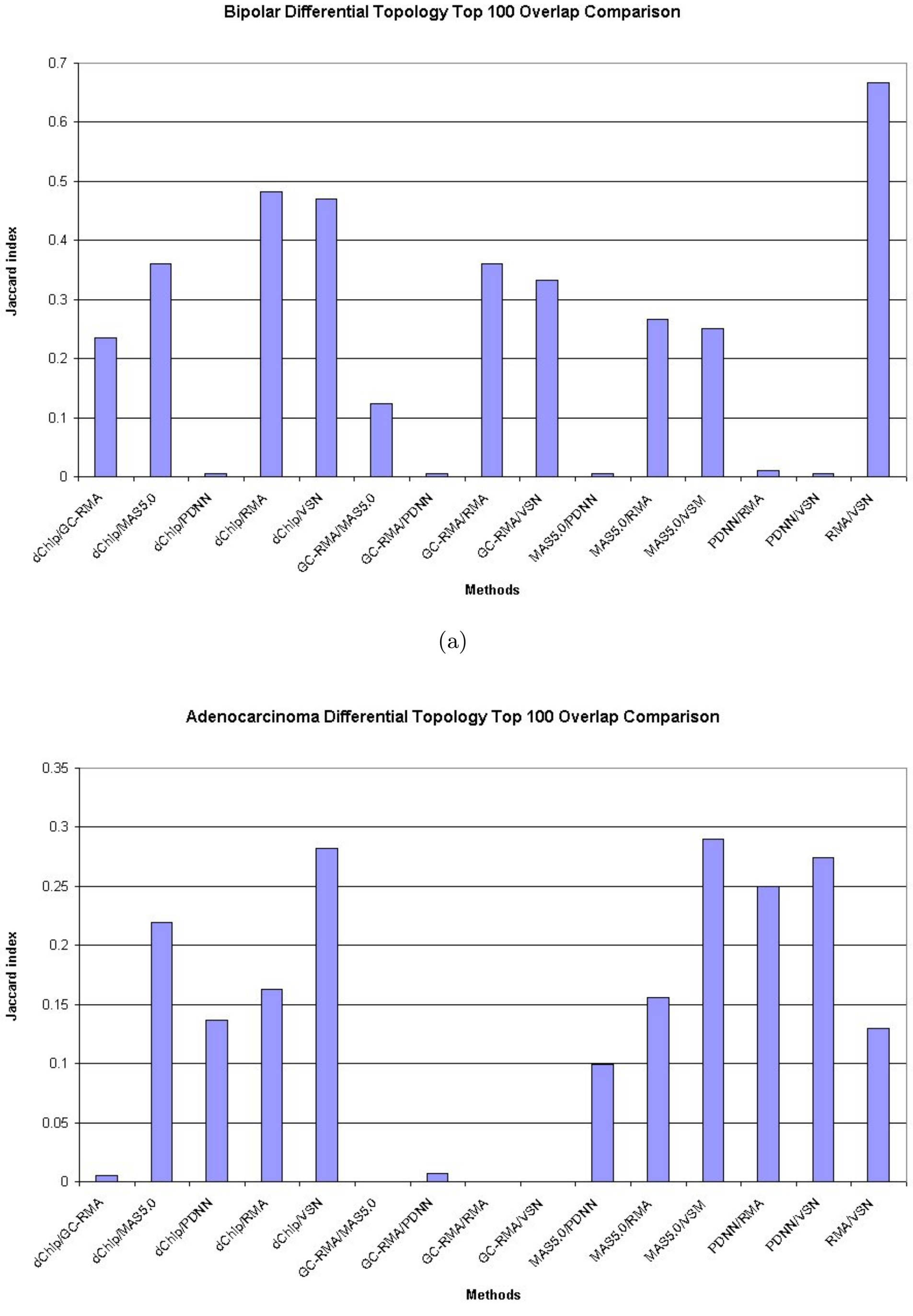
(b)

Figure B.2: Differential topology similarities for the top 100 genes showing the largest difference in topology for bipolar disorder, pulmonary adenocarcinoma, and colorectal adenoma data. (a) Bipolar disorder comparison, (b) Pulmonary adenocarcinoma comparison, (c) Colorectal adenoma comparison.



(c)

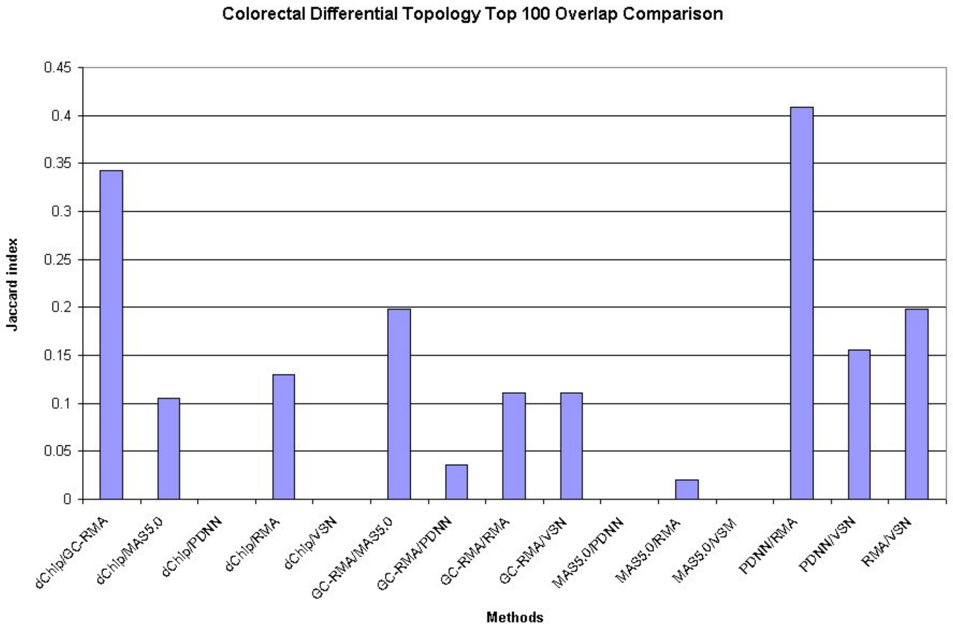
Figure B.2: Continued.



(a)

(b)

Figure B.3: Differential topology similarities for the top 100 genes showing the largest difference in topology for bipolar disorder, pulmonary adenocarcinoma, and colorectal adenoma data. (a) Bipolar disorder comparison, (b) Pulmonary adenocarcinoma comparison, (c) Colorectal adenoma comparison.



(c)

Figure B.3: Continued.

Vita

Andy D. Perkins was born in Murray, KY in 1979. He graduated from Graves County High School in 1997 and went on to study at Murray State University in Murray, KY. He received his BS in Computer Science and Mathematics in 2001 and MS in Mathematics in 2003. After this, he moved to Knoxville, TN to pursue the PhD in Computer Science. He earned his PhD in 2008 from The University of Tennessee under the direction of Dr. Michael A. Langston.