Automatic Reconstruction of the Building Blocks of Molecular Interaction Networks

Corban G. Rivera

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Science

Richard Helm
T. M. Murali, Chair
Naren Ramakrishnan
Brett Tyler
Anil Vullikanti

August 11, 2008
Blacksburg, Virginia

Keywords: Network Legos, Cross-condition analysis, Molecular Interaction Networks, Active Networks, Pathway Perturbation

Automatic Reconstruction of the Building Blocks of Molecular Interaction Networks

Corban Rivera

(ABSTRACT)

High-throughput whole-genome biological assays are highly intricate and difficult to interpret. The molecular interaction networks generated from evaluation of those experiments suggest that cellular functions are carried out by modules of interacting molecules. Reverse-engineering the modular structure of cellular interaction networks has the promise of significantly easing their analysis. We hypothesize that:

(i) cellular wiring diagrams can be decomposed into overlapping modules, where each module is a set of coherently-interacting molecules and

(ii) a cell responds to a stress or a stimulus by appropriately modulating the activities of a subset of these modules.

Motivated by these hypotheses, we develop models and algorithms that can reverse-engineer molecular modules from large-scale functional genomic data. We address two major problems:

(I) Given a wiring diagram and genome-wide gene expression data measured after the application of a stress or in a disease state, compute the active network of molecular interactions perturbed by the stress or the disease.

(II) Given the active networks for multiple stresses, stimuli, or diseases, compute a set of network legos, which are molecular modules with the property that each active network can be expressed as an appropriate combination of a subset of modules.

To address the first problem, we propose an approach that computes the most-perturbed subgraph of a curated pathway of molecular interactions in a disease state. Our method is based on a novel score for pathway perturbation that incorporates both differential gene expression and the interaction structure of the pathway. We apply our method to a compendium of cancer types. We show that the significance of the most perturbed sub-pathway is frequently larger than that of the entire pathway. We identify an association that suggests that IL-2 infusion may have a similar therapeutic effect in bladder cancer as it does in melanoma.

We propose two models to address the second problem. First, we formulate a Boolean model for constructing network legos from a set of active networks. We reduce the problem of computing network legos to that of constructing closed biclusters in a binary matrix. Applying this method to a compendium of 13 stresses on human cells, we automatically detect that about four–six hours after treatment with chemicals cause endoplasmic reticulum stress, fibroblasts shut down the cell cycle far more aggressively than fibroblasts or HeLa cells do in response to other treatments.

Our second model represents each active network as an additive combination of network legos. We formulate the problem as one of computing network legos that can be used to recover active networks in an optimal manner. We use existing methods for non-negative matrix approximation to solve this problem. We apply our method to a human cancer dataset including 190 samples from 18 cancers. We identify a network lego that associates integrins and matrix metalloproteinases in ovarian adenoma and other cancers and a network lego including the retinoblastoma pathway associated with multiple leukemias.

Contents

List of Figures

Chapter 1

Introduction

A major challenge for biology in the twenty-first century is to integrate multi-level views of cell physiology generated by high-throughput biological screens into a comprehensive understanding of how cells and organisms function. The dramatic revolution in genomics that we have witnessed since the genome of the bacterium *Haemophilus influenzae* was sequenced in 1995 has provided us with the genomes of more than 500 organisms [17]. Our ability to interrogate the cell in more sophisticated ways is also improving at a dramatic pace. For many model organisms, we now have available large repositories of heterogeneous types of genomic and biological data such as transcriptional profiles [28, 33, 46, 152, 170]; protein-protein interaction (PPI) data [7, 8, 21, 157]; protein-DNA binding data [100]; regulatory, signaling, and biochemical networks [83, 100, 154]; and catalogues of gene and protein functions [4].

This explosion in genomics and the parallel revolutionary scale-up in our ability to probe the state of a cell on a genome-wide scale has brought about the advent of systems biology [65, 74, 75, 82, 91]. Rather than studying individual genes or proteins, biologists investigate the behaviour and relationships among all the molecules of a biological system of interest. Integrating different types of information on molecular interactions such as protein-protein, protein-DNA, and synthetically lethal interactions yields a multi-modal wiring diagram of the cell. The cellular wiring diagram forms the foundation of the work presented in this thesis.

Hartwell et al. [65] in a seminal paper argue that cellular functions are carried out by modules of interacting molecules. Biological processes are rarely carried out by a single gene; rather processes are carried out by a network of interconnected molecules acting in concert. The notion of a module is useful if it incorporates a subset of molecules whose function can be viewed in isolation. Several examples of cellular mechanics lend themselves to this idea. Protein synthesis is a discrete function carried out by the ribosome and its associated proteins. Signaling cascades are achieved by groups of proteins with preferential binding. Modules can overlap and have dynamic activity. For example, the dynamic perturbation of groups of integrins facilitate cellular motility and binding in association with metastasis [117]. The availability of genome-wide biochemical assays under diverse cellular perturbations enable the exploration of modules as a tool for understanding cellular activity.

This Ph.D. thesis rests on two hypotheses that are motivated by the promise of modular cell biology. First, we hypothesize that cellular wiring diagrams can be decomposed into overlapping modules, where each module is a set of coherently-interacting molecules. Second, we suggest that a cell responds to a stress or a stimulus by appropriately modulating the activities of a subset of these modules. The goal of this thesis is to test these hypotheses by developing models and algorithms that can reverse-engineer molecular modules from large-scale functional genomic data. In fact, the second hypotheses inspires a novel approach for computing modules that is the basis for this thesis. Suppose that for a diverse variety of stresses or diseases, we have available the cellular interaction networks perturbed by or responding in each stress/disease. Given this set of perturbed networks, can we reconstruct modules in such a manner that they may be combined to yield each perturbed network with a high degree of accuracy. Accordingly, we address two major problems:

(I) Given a wiring diagram and genome-wide gene expression data measured after the application of a stress (e.g., nutrient starvation) or in a disease state (e.g., leukemia), compute the response network of molecular interactions perturbed by the stress or the disease.

(II) Given the response networks for multiple stresses, stimuli, or diseases, compute a set of network legos, molecular modules with the property that each response network can be expressed as an appropriate combination of a subset of network legos.

This two-stage approach is predicated on our belief that decomposing the wiring diagram into overlapping modules must be done in the context of response networks that the modules compose. By design, network legos are "building blocks" of the response networks from which they are computed.

1.1 Organization of this Thesis

The organization of this thesis is as follows.

1. In Chapter 2 we discuss terms, notation, and related literature that is useful for understanding the concepts and context of this thesis.

2. In Chapter 3, we propose an approach that integrates a curated pathway of molecular interactions with gene expression data from both cellular stress samples such as cancer tissue and normal samples such as healthy tissue to compute whether the pathway is perturbed in response to the stress.

3. The primary contribution of Chapter 4 is a model that defines network legos as Boolean combinations of active networks. We propose an algorithm based on closed itemset mining to discover network legos. As a prelude to this algorithm, we propose an approach that integrates a molecular interaction network with gene expression data to compute the active network, the molecular interactions within a cell perturbed by a single stress or stimulus. We formulate

4. In Chapter 5, we develop an alternate formulation of network legos as linear combinations of active networks. This models takes weighted active networks into account in a meaningful way. We use a method for non-negative matrix approximation to compute network legos so as to directly directly optimize for recoverability, i.e., how well each active network can be represented as an additive combination of computed network legos.

5. In Chapter 6, we conclude with lessons learned as a result of this thesis and the contribution to the field of modular systems biology.

1.2 Introduction to Pathway Perturbation

The normal functioning of a living cell is characterized by complex interaction networks involving many types of molecules. Associations detected between diseases and perturbations in well-defined pathways within such interaction networks have the potential to illuminate the molecular mechanisms underlying disease progression and response to treatment.

As an emerging trend in the field of systems biology, web sites that host curated biological pathways have become commonplace for many model organisms [2, 85, 177]. Curated pathways defined by sets of molecular interactions can be considered canonical biological modules. A perturbation in the quantity of molecules associated with a pathway in response to a stress or treatment gives clues about the molecular basis for an observed phenotype. Biologists would like to know which set of pathways are perturbed by a cellular stress or treatment.

Approach In Chapter 3, we propose a knowledge-driven approach that integrates a curated pathway of molecular interactions with gene expression data from both cellular stress samples such as cancer tissue and normal samples such as healthy tissue to compute the perturbation of the pathway in response to the cellular stress. We develop a score for pathway perturbation based on a rigorous statistical procedure that combines estimates of differential expression of genes with the interaction structure in the pathway. We use a simulated annealing approach to identify the most significantly perturbed set of interactions.

Results We apply our method to a compendium of 18 cancer types and a set of 20 cancer and immune signaling pathways. We show that the significance of the most perturbed subpathway is frequently higher than that of the entire pathway. We compare our method to Gene Set Enrichment Analysis (GSEA) [118, 162, 169] and we find that our method has superior sensitivity. We exploit the compendium of pathway-cancer perturbation scores to construct separate pathway and cancer association diagrams. We find that the TNF-α, TGF-β, EGFR1, and B cell receptor pathways are perturbed in response to many cancer types. Using the pathway association diagram and analyzing the underlying gene expression data, we find evidence that up regulated expression of TNF-α, TGF-β, and Integrins may suggest metastatic potential. We also identify an association between melanoma and bladder cancer

that suggests that IL-2 infusion may have a similar therapeutic effect in bladder cancer as it does in melanoma.

Contributions We present a computational method that compares expression profiles of genes in cancer samples to samples from normal tissues to detect perturbations of pre-defined pathways in the cancer. Our approach explicitly takes into account the interactions between the gene products in a pathway.

Application Scenario A biologist can use the knowledge-based active network algorithm to investigate the response network for a stress condition. To begin, the biologist needs gene expression measurements from samples under the stress condition, unstressed condition. Using a set of pathways from databases such as REACTOME [177], NCI, and Netpath, the biologist runs the knowledge-based active networks algorithm using the microarray measurements. The algorithm produces the stress-response network in the context of each significantly perturbed pathway.

1.3 Introduction to Boolean Network Legos

Integrating the wiring diagram of a cell with the transcriptional profile for a particular experimental condition is a powerful technique for obtaining insights into the interactions that are activated in the cell in that condition. Methods like the one proposed in Chapter 3 compute "response modules" that provide quasi-dynamic models of cellular processes. In Chapter 4, we test the hypothesis that a cell responds to a stress or a stimulus by appropriately modulating the activities of a subset of these modules.

Approach In Chapter 4, we present a top-down computational approach that identifies building blocks of cellular networks from active networks. We compute the active network that responds to a single disease state or experimental condition. We systematically combine active networks computed for different experimental conditions using Boolean (set-theoretic) formulae to reveal *network legos*, which are modules of coherently interacting genes and gene products in the wiring diagram.

(i) each active network as a union of a subset of network legos and

(ii) each network lego as an intersection or difference of a subset of active networks.

These network legos are potential building blocks of the wiring diagram since we can express each active network as a union of network legos. Figure 1.1 demonstrates both the decomposition of active networks into network legos and the composition of network legos back into active networks.

Results We apply the algorithm to two human gene expression datasets. We apply our approach to three leukemias, Acute Lymphoblastic Leukemia (ALL), Acute Myeloid Leukemia (AML), and Mixed Lineage Leukemia (MLL), studied by Armstrong et al. [3]. We find that the Kit receptor pathway is significantly activated in AML but not in ALL or in MLL, a fact that has considerable support in the literature. We also apply our method to a collection of 178 arrays measuring the gene expression responses of HeLa cells and primary human lung fibroblasts to cell cycle arrest, heat shock, endoplasmic reticulum stress, oxidative stress, and crowding [120]. Our analysis automatically reveals that about four–six hours after treatment with DTT or menadione, fibroblasts shut down the cell cycle far more aggressively than fibroblasts or HeLa cells do in response to other treatments.

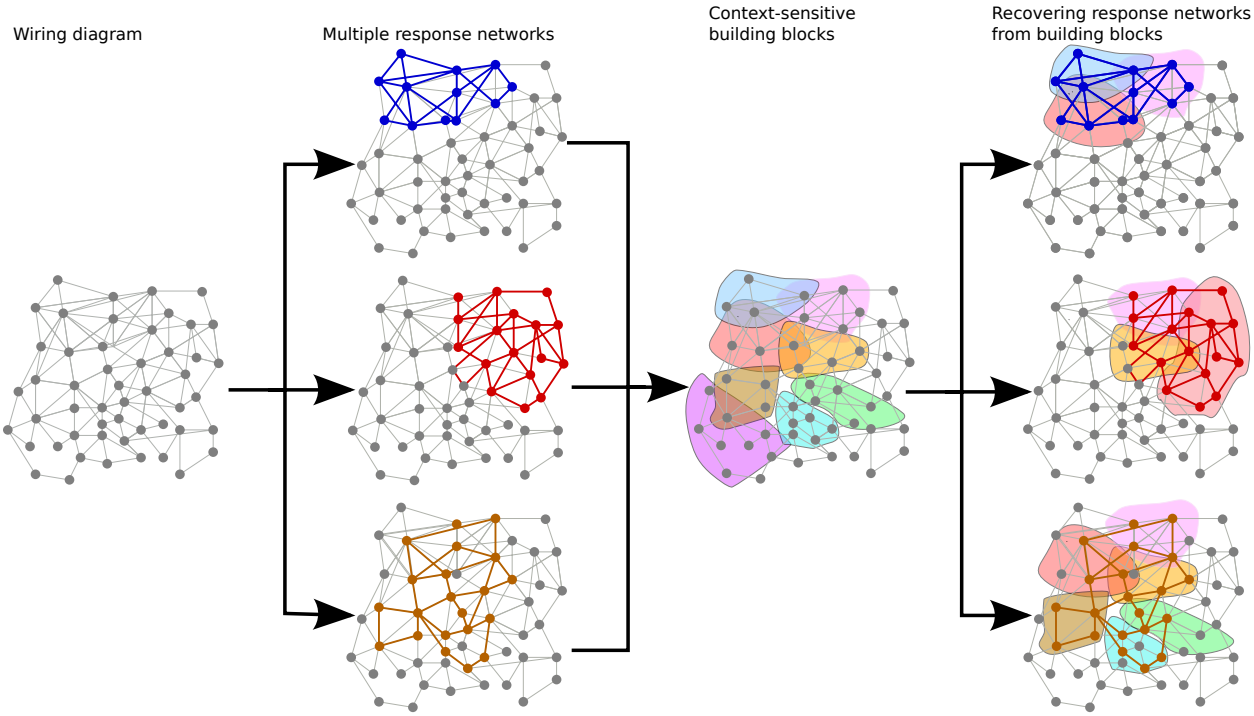


Figure 1.1: A schematic of network lego computation. Column 1: the wiring diagram; column 2: three response networks that are subgraphs of the wiring diagram; column 3: network legos computed from the response networks; column 4: each response network expressed in terms of network legos.

Contributions We present a Boolean method for automatically identifying the building blocks of molecular interaction networks in Chapter 4. We represent similarities and differences between the network of interactions activated in response to different cell states both as a set theoretic formula involving cell states and as a network lego, a functional module of co-expressed molecular interactions. A novel contribution of our work is the directed acyclic graph (DAG) that relates all cell states (and the active networks corresponding to the cell states). This DAG provides a high-level abstract view of the similarities and differences between cell states.

Application Scenario There are two ways in which a biologist can use our system. In the first, our system allows the systematic comparison of responses to a small number of different conditions, diseases, or perturbations tested in the same lab. In the second, a biologist can analyse a specific condition of interest in the context of a larger set of conditions, compute the building blocks of the networks activated in these conditions, and ask how the building blocks compose the active network for the specific condition of interest to the biologist. Network legos suggest similarities and differences between the datasets permitting identification of known as well as unknown cellular stress associations.

1.4 Introduction to Weighted Network Legos

Chapter 4 introduces the motivation, concepts, and an algorithm for finding network legos based on a Boolean model. In Chapter 5, we build upon our work and establish a new model for network legos: as a linear combination of active networks. We seek an efficient algorithm to find network legos that directly optimizes recoverability and incorporates edge weights of active networks in a meaningful way.

Approach We formulate the problem such that we directly optimize for active network recoverability, *i.e.* how well each active network can be represented by a linear combination of network legos. Our approach takes as input a set of active networks; each such network is a weighted subgraph of the molecular interaction network and is associated with a particular cellular stress or treatment. Each network lego we compute is itself a weighted subgraph of the wiring diagram. Furthermore, we can represent:

(i) each active network as an additive combination of a subset of network legos and

(ii) each network lego as a compact linear combination of the active networks.

We show that this problem can be naturally solved using powerful and well-known techniques for non-negative matrix approximation.

Results We apply our method to both synthetic and real data. We generate the synthetic datasets using a generative model that allows us to measure network lego discovery with increased levels of noise. We see that our ability to discover network legos correctly improves with the number of active networks in the dataset. We show that the performance of our method slowly degrades with increased noise. We apply our method to a human cancer dataset including 190 samples from 18 cancers. We compute active networks for each of the 190 samples using the method described in Chapter 3. We use a measure of partition stability to identify an appropriate number of network legos to compute. We identify a network lego that implicates integrins and matrix metalloproteinases in ovarian adenoma. We find a network lego including the retinoblastoma pathway associated with multiple leukemias. We find that the mitogen-activated protein kinase pathway is perturbed in association with multiple tumors of the central nervous system.

Contributions We present a method for finding network legos that directly optimizes for network lego recoverability. We take the interaction weights from active networks into account in a meaningful way. The reformulation of the problem into that of non-negative matrix approximation allows for much greater scalability in the number of cellular stresses and interactions. A contribution of this work is the dual view of a network lego as both a linear combination of active networks and a non-negative interaction weighted network.

Application Scenario The application scenario is the same as in Section 1.3, except that the additive model is more useful when the number of conditions being compared is in the hundreds or thousands.

Chapter 2

Background

Genome scale biological assays measure many facets of cellular state. Genome sequencing, protein-protein interaction assays, protein-DNA binding experiments, and DNA microarrays are some of the sources of high-throughput biological data. With the extent of whole genome data available, biologists have the capability to make inferences and hypothesis on a much broader scale than before. The field of systems biology has emerged to study the mechanisms controlling many intricate biological processes. Systems biologists construct mathematical models of biological systems and computational methods developed to organize, analyze, and reason about the influx of high-throughput biological data. The aim is to understand a cell not just as a collection of individual molecules but as a set of modules that behave coherently and interact with each other. In this chapter, we review experimental methods used to generate such datasets and computational methods to analyze them. Given the hundreds of papers that have been published on these topics, we restrict our attention a representative sample of those techniques that are most relevant to our approach.

2.1 Experimental Data Sources

A wealth of molecular interaction and gene expression data generated through high-throughput whole-genome biological assays is now available. Genome scale biological assays measure many facets of cellular state.

2.1.1 Sources of Protein-Protein Interaction Data

Many biological processes require the collaboration of groups of proteins that act together as complexes. Protein complexes can be formed by covalent protein interactions [122], ionic and hydrogen interactions [108, 187], and electrostatic interactions [123, 187]. To detect these physically interacting proteins, biologists have developed high-throughput assay techniques.

Yeast two-hybrid The two-hybrid technique has been employed to detect PPIs on a genome-wide scale in many organisms such as *S. cerevisiae* [78, 172, 179] and *D. melanogaster* [59]. The two-hybrid protein interaction detection mechanism [48] works by generating a signal if a pair of query proteins interact. In high-throughput and large-scale yeast two-hybrid experiments, every gene in a "bait" library is cloned and augmented with a sequence binding domain. For each gene in a "prey" library, a clone is made containing an activation domain. Clone pairs from the cross product of the activation and binding domain sets are systematically tested for resulting transcription. If the pair of clones bind to form a complete transcription factor, the indicator gene is transcribed. From the pairs of clones that activate transcription of the indicator gene, the set of interacting proteins is derived.

Co-immunoprecipitation Co-immunoprecipitation is another method to discover interacting proteins [134]. The interaction detection mechanism works by isolating a bait protein and any proteins bound to the bait. The bait protein is cloned and augmented with a antibody binding tag. To isolate the bait from whole cell lysate, an antibody which is known to bind to the antibody tag on the bait is added. Next, a G-protein, known to bind to most antibodies, is used to extract the antibody, bait protein, and any proteins bound to the bait protein [134]. Subsequently, the purified protein complex is denatured into its component proteins for identification. A western blot is used to identify the if specific proteins are present in the sample. The experiment yields a complex of two or more proteins containing the bait. Co-immunoprecipitation has been applied on a genomewide scale to detect many protein complexes [58].

Caveats Associated with Molecular Interaction Data. While protein interactions have potential to provide many useful insights into fundamental biological questions, high-throughput biological assays to detect protein-protein interactions may find many interactions that do not take place in the cell [9]. New research [161] has predicted the size of the human PPI network to be on the order of $650,000$ interactions, about six times the size of our current datasets. The prediction suggests that we currently do not have evidence for many interactions that do take place in the cell.

2.1.2 Microarray Gene Expression Data

Microarray technologies that measure gene expression have revolutionized our ability to understand the dynamics of biological systems. Microarray technologies are currently used to quantitatively probe the levels of mRNA, proteins, methylation, and post-transcriptional modifications. In this study, we focus on microarrays that measure gene expression levels because of the wealth of publicly available sources of data. All of these techniques use probe sets that are covalently attached to a chemical matrix. The first reported use of DNA microarrays was by Schena et al. [141] in 1995. As miniaturization progressed, Lashkari et al. [96] was the first to measure all the genes of *S. cerevisiae* simultaneously. Modern microarray technology can measure the expression level of tens of thousands of genes. The number of

probes is easily large enough to measure the activity of all the genes in human including several splice variants of the genes.

Microarray Experiment A microarray contains a collection of probe sets. Each probe set corresponds to a single target mRNA. Each probe in the probe set is composed of the anti-sense nucleotide sequence of the target mRNA. Its expected that only the target mRNA should have good binding affinity to the probe. Each probe is affixed to a stationary phase matrix. The probes in each probe set are located close to each other on the matrix. The microarray experimental process compares a collection of treated or disease cells with untreated cells. The mRNA is separately isolated from both treated and control cells. The mRNA is reverse transcribed into fluorescent labeled cDNA. Both treated and untreated fluorescent labeled samples competitively bind to the probes. The fluorescence can be measured and converted into an absolute or relative estimate of the mRNA level, depending on the microarray technology used. There are multiple online repositories for microarray gene expression data sets with the largest being the Stanford Microarray Database [36] with over 12000 microarray experiment sets.

Detecting Differential Gene Expression A common problem in microarray studies is finding the differentially expressed genes, those genes that have elevated or reduced levels of mRNA expression with respect to their control. Several common strategies exist including fold change, the t-test, and analysis of variance (ANOVA). These methods operate on a per-gene basis and must include multiple hypothesis correction like Bonferroni or false discovery rate adjustment by Benjamini and Hochberg [15]. Other methods have been proposed that search for differential expression in a more holistic approach. These methods utilize covariance between genes to determine differential expression. The covariance based methods include the elastic net [72] and gradient directed regularization [50] to name a few.

2.2 Clustering Algorithms

Clustering is a common strategy to find associations between genes. Clustering is the process by which genes are grouped into gene sets based on similarities in their patterns of gene expression. Clustering is considered an unsupervised learning method in that for a given input set the correct result set is unknown. All clustering methods require a measure δ to determine the distance or similarity between a pair of gene expression patterns across conditions. Some of the measures in use include the Manhattan distance, Euclidean distance, Pearson's correlation, Spearman's correlation, and mutual information [131]. Clustering is a mature field with hundreds of journal articles. Many of the details in the following section come from a review article on clustering by Jain, Murty, and Flynn [79]

Hierarchical clustering Hierarchical clustering begins by placing each point in a separate cluster and computing the distance between all pairs of clusters. Hierarchical clustering

proceeds by merging clusters using either single-link, average-link, and total-linkage clustering until all points reside in a single cluster. A similarity threshold σ partitions the points into clusters.

Single-linkage defines the nearest clusters as the pair that have the nearest pair of points over all cluster pairs. In total-linkage, the nearest clusters are the pair that have the nearest distance between the furthest pair of points between the two clusters. Average-linkage maintains a centroid point for each cluster. The nearest clusters are the pair that have the nearest centroids over all cluster pairs.

k-means clustering The k-means clustering algorithm partitions the points into k clusters, where k is a user-defined parameter. The goal of the algorithm is to minimize the sum over all the points of the distance of a point to the centroid of the cluster the point belongs to. The algorithm begins by randomly partitioning the points into k groups. The algorithm repeatedly computes the centroid of each group and associates each point with the closest centroid. The process converges when an objective energy function in minimized such as within group variation or between group coupling. The procedure converges when no additional cluster reassignments are made. The k-means algorithm is very popular although it is not guaranteed to converge.

Wilkin and Huang [183] compare two k-means clustering algorithms to show the difference between Lloyd's K-means Clustering and the Progressive Greedy K-means Clustering with respect to running times and clustering cohesiveness. They find that Lloyd's K-means Clustering algorithm is more efficient on both synthetic and gene expression data.

Wu [185] provided a recent application of weighted k-means clustering to gene expression data. Wu provides a genetic algorithm for the problem of weighted k-means clustering. As genetic algorithms are known to provide more comprehensive coverage of the search space, the method GWKMA does not suffer from some of the shortcomings of traditional k-means clustering, such as sensitivity to initial center selection, convergence in local minima, and only detecting spherical clusters. The method was applied to gene expression data and protein mass spectrometry data and synthetic data. Wu shows that GWKMA outperforms k-means clustering on multiple datasets.

Self organizing maps Self organizing maps use a set of nodes where each node has an associated weight vector of the same dimension as the input data. The algorithm begins by randomly assigning values to the weight vector in the nodes. For each input vector, the algorithm computes the node n that is closest to the input vector. The weight vector of nodes close to n are adjusted to be closer to n. The process converges when there are no remaining input vectors.

Spectral Clustering Spectral clustering has become a recently favored algorithm due to its simple implementation and improved performance over k-means. Spectral clustering is able to discover cluster with more diverse structure than k-means. The input to the

algorithm is an adjacency matrix A and the number of clusters k. The degree matrix $\mathcal{D}$ is a diagonal matrix that contains the node degrees along the diagonal. The Laplacian matrix is $\mathcal{L} = \mathcal{D} - A$. The first k eigen vectors $u_1, u_2 \ldots u_n$ are chosen from the Laplacian matrix $\mathcal{L}$. The algorithm uses k-means to cluster the set of points defined by $p_i = u_{1,i}, u_{2,i}, \ldots, u_{n,i}$ for all i [29].

Non-Negative Matrix Approximation Non-negative matrix factorization also more appropriately called non-negative matrix approximation decomposes a non-negative matrix (one that contains non-negative entries) $\mathbf{M}$ into two non-negative matrices $\mathbf{W}$ and $\mathbf{L}$ so that $\mathbf{M} \approxeq \mathbf{WL}$. Lee and Seung [98] popularized non-negative matrix factorization by showing that when applied to a set of gray-scale images containing faces, the matrices $\mathbf{W}$ and $\mathbf{L}$ could be interpreted as containing basis vectors representing "parts" of faces and how these parts could be combined to obtain the input images.

Non-negative matrix factorization has been applied in many contexts and applications such as document clustering [149, 188], image classification [63, 97, 98], and gene expression data analysis [22, 90, 107, 165]. The roots of non-negative matrix factorization can be traces back to Paatero and Tapper [124, 125] who were the first to describe a positive decomposition of a matrix into a weighting matrix and a loading matrix. In the now classic formulation, Lee and Seung [97, 98] presented the use of non-negative matrix factorization for image recognition and text mining. The method was quickly accepted for its interpretability, scalability, and ease of implementation. Lee and Seung's method was the first to use multiplicative updates to increase the speed of non-negative matrix factorization conversion.

Lee and Seung [98] argued that compared to other approaches for matrix factorization like PCA, non-negative matrix factorization produces a decomposition of the original matrix into parts. Decomposition into parts has interested many researchers, because the basis vectors are directly interpretable. Since then, non-negative matrix factorization has been used extensively in diverse application such as summarizing gene expression data sets [107], cancer classification [22, 53], document clustering [149, 188], music transcription [156], image classification [63, 104] and many others. Soon authors [40, 104] began to question when non-negative matrix factorization gives a good decomposition into parts. They notice that the parts based factorization is dependent on the training set. The training set should contain examples that correspond to all combinations of parts.

Kim, Sra, and Dhillon [163] propose a projection based method for non-negative matrix approximation based on alternating least-squares (ALS). They state that existing (ALS) methods do not have theoretical proof of their convergence properties. The iterative method partitions variables into free and fixed groups. They prove that their method has a limit point. They compare their method to Lee and Seung [98] and they find that their method converges in fewer iterations.

Brunet et al. were the first to use non-negative matrix factorization (NMF) to find a set of meta-genes [22] in the leukemias ALL and AML. Brunet et al. uses NMF to describe a leukemia gene expression data set in terms of a few significant meta-genes. Each meta-gene is a linear combination of genes from the gene expression data set. The small set of

meta-genes approximates the variation within the large set of gene expression measurements. Tamayo et al. [165] use a similar meta-gene construction and apply their method leukemia and lung cancer samples. Kim and Park [90] apply sparse non-negative matrix factorization to gene expression data analysis. They apply their method to the problem of cancer classification. Liu, Yuan and Ye [107] also use non-negative matrix factorization to solve the problem of microarray dimensionality reduction. They consider 11 datasets, and they find for one leukemia dataset that a sample may have been misclassified.

Ding, Li, and Peng [38] showed the equivalence between a constrained version of non-negative matrix factorization and Laplacian spectral clustering. The constrained decomposition takes the form

$$M \approxeq HH^T.$$

They also prove the equivalence between non-negative matrix factorization and clustering of rows and columns in a bipartite graph. They prove that the constrained version of non-negative matrix approximation produces results which are semi-orthogonal.

Biclustering Biclustering is a two dimensional clustering strategy. In a gene expression dataset, a bicluster could be defined (in generic terms) as a subset of genes and a subset of samples such that the selected genes are co-expressed only in the subset of selected samples; these genes may not be coherently expressed in the other samples. Biclusters are attractive since they capture condition-specific patterns of co-expression. Since Cheng and Church [26] introduced this idea to analyze gene expression data, a plethora of biclustering algorithms have appeared in the literature. We discuss a representative subset of these methods below, referring the reader to more comprehensive surveys for descriptions of other approaches [110, 168].

It is defined with a minimum support threshold t. Element $\mathcal{H}_{ij}$ is one if gene i is perturbed in sample j and zero otherwise. We define a bicluster as a subset of rows and a subset of columns containing only ones. A closed bicluster is a bicluster such that the inclusion of any additional row or column would introduce a zero into the bicluster. A classic approach to this problem is the *Apriori* algorithm. The algorithm begins by generating a set of candidate biclusters that contain only one sample and more than t activated genes. Each candidate is expanded in a breadth first approach to include an additional sample and a subset of activated genes. A candidate bicluster converges when there is no additional sample to add that has a set of at least t active genes in common.

Cheng and Church [26] presented one of the earliest applications of biclustering on gene expression data. They propose a node deletion and node addition based algorithm to find non-overlapping biclusters. They introduce the notion of average mean squared residue. They apply their method to dataset of human and yeast gene expression data.

Tanay et al. [166, 167] integrate diverse sources of data such as gene expression, protein interactions, growth phenotype, and transcription factor binding data. They propose a method called Statistical-Algorithmic Method for Bicluster Analysis (SAMBA) to mine a bipartite graph where edges connect genes to properties. They compute the log likelihood that a subset of genes and a subset of properties would be so densely connected by chance.

They reduce the problem of finding significant biclusters to one of finding maximum bounded bicliques. They construct a global map of yeast cellular activity by connecting biclusters with 1/3 matching genes. If a function annotates more than 40% of the genes in a bicluster, they predict that this function annotates the remaining genes in the bicluster. They predict the function of over 800 uncharacterized genes. They apply their method to *S. cerevisiae* datasets, and they report numerous modules corresponding to biological processes. They show that the modules have a hierarchical organization.

Li et al. [103] propose a general biclustering scoring method based on Kolmogorov complexity. The score measures the amount of non-randomness present in a bicluster. They develop a Markov Chain Monte Carlo algorithm for identifying high scoring biclusters. They measure the performance of their method on both synthetic and real yeast gene expression data. They compare their algorithm to SAMBA [167] and Chang and Church [26].They find that their method is comparatively robust to noise. They also identify larger biclusters than SAMBA, because SAMBA requires that the degree of every gene is bounded. They apply their method to yeast cell cycle gene expression data and evaluate the functional enrichment of the genes in the biclusters.

Gu and Liu [62] propose a Bayesian approach to microarray biclustering called the Bayesian biclustering model (BBC). They propose a Gibbs sampling procedure for inference of the biclusters. They compare their method to SAMBA [167] and others. They find that their method recovers more significant patterns. They show that their method recovers the expected clusters in synthetic data. They apply their method to yeast gene expression data and perform functional enrichment. They find clusters associated with they cell cycle, cell differentiation, and DNA processing.

An additive bicluster is one such that the difference between any two columns is a constant. A constant bicluster has the property that the difference between any two columns in the bicluster is zero. Cheng et al. [25] propose a polynomial time algorithm for identifying additive and constant biclusters. The method can identify many of the additive and constant biclusters present in the data. They provide an extension of their method for multiplicative biclusters. They show results from both artificial data and yeast cell cycle gene expression data. They verify their clusters with functional enrichment from the Gene Ontology [4].

2.3 Gene Set Enrichment

Understanding the biological meaning present in a set of perturbed genes can be a difficult task. Functional enrichment is commonly used to reveal biological meaning in a group of perturbed genes. Given a specific biological function, this procedure answers the question of whether the function annotates a surprisingly large number of genes in the group. Many resources currently exist that provide functional annotations [4, 162] for many genes. In this context, genes are grouped together into a gene set if they share an annotation. Many methods [30, 35, 39, 61, 126, 127, 150] exist to identify gene sets that are significantly over-represented in a group of perturbed genes. All of the gene set enrichment methods take a collection of gene sets $\mathcal{S}$ as input.

Fisher's Exact Test The hypergeometric distribution is an appropriate statistical technique for determining the over-representation of a given gene set in a set of perturbed genes. This technique has been used in a number of papers in bioinformatics. The hypergeometric model assigns a score to the likelihood of randomly selecting a given number of marked elements (those in a given gene set) from a universe U of elements. Let $T \subseteq U$ be the set of marked elements. Suppose we have a computational procedure that selects a set of elements $U' \subseteq U$ and that U' contains a subset T' of T. Let u, u', t, and t' denote the sizes of the sets U, U', T, and T' respectively. We are interested in computing the probability that this event is statistically significant. The null hypothesis H_0 is that if we select u' items from a set of u items uniformly at random without replacement, our random sample will contain at least t' samples from the set T. The alternative hypothesis H_1 is that this event cannot happen at random. We accept H_1 if and only if the probability of H_0 is less than a user specified threshold. The probability of this event is

$$H(u, t, u', t') = \frac{\binom{t}{t'}\binom{u-t}{u'-t'}}{\binom{u}{u'}}.$$

Since H_0 is the probability that the random sample contains t' or more samples from T, the probability that H_0 is true is the sum

$$Pr(H_0 \; is \; true) = \sum_{i=t'}^{min(t,u')} \frac{\binom{t}{i}\binom{u-t}{u'-i}}{\binom{u}{u'}}.$$

GSEA Gene Set Enrichment Analysis (GSEA) [118, 162, 169] is a prominent method for assessing functional enrichment; this method is useful when genes can be ranked by a score. The method takes as input a treatment and control gene expression data set. GSEA computes a t-score or similar statistic for each gene and sorts genes in descending value of this statistic. GSEA uses a modified Kolmogorov-Smirnov test to assess whether the genes in the given gene set have surprisingly high or surprisingly low ranks in the sorted order. The significance of the enrichment score computed by the test is generated by permutation testing.

2.4 Gene Networks that Respond to Cellular Stress

A number of techniques overlay gene expression data for a condition on the wiring diagram to compute the active network for that condition [66, 76, 136, 146]. [173] use the wiring diagram as a constraint network and compute dense subgraphs in the gene co-expression network as long as the genes in the dense subgraphs induce a connected subgraph of the wiring diagram. These methods typically focus on a single condition of interest.

High-throughput interaction data generated using some of the methods listed in Section 2.1.1 are catalogued in a number of publicly available repositories [7, 158, 177, 186]. The availability of this data has inspired many methods [66, 76, 76, 111, 136, 146, 173, 178, 180]

that integrate gene expression data with molecular interaction networks to determine the sub-network that is perturbed. All such methods take as input a gene expression dataset $\mathcal{D}$ and a wiring diagram $\mathcal{W}$.

Simulated Annealing Ideker et al. [76] proposed one of the earliest approaches for integrating the wiring diagram $\mathcal{W}$ and gene expression data. Their method combines simulated annealing with Stouffer's z-score as the objective function to minimize. For each gene in $\mathcal{D}$, they estimate a z-score $z(g)$ corresponding to the perturbation of gene g. Given a specific subgraph Q of $\mathcal{W}$, they compute the overall perturbation of Q using the Stouffer's z-score:

$$z(Q) = \frac{\sum_{g \in \mathcal{G}} z(g)}{\sqrt{|\mathcal{G}|}}.$$

They use simulated annealing to return a connected network that demonstrates high perturbation.

Covariance Based Methods Several more recent methods begin by identifying gene expression correlations between all pairs of genes in the $\mathcal{D}$. Let S be a symmetric similarity matrix where S_{ij} contains the Pearson's correlation between genes i and j. Ulitsky and Shamir [173] develop a method that computes Jointly Active Connected Subnetworks. (JACS). They use $\mathcal{W}$ as a constraint network. Their goal is to find highly dense subgraphs of the co-expression network with the property that the subgraph of $\mathcal{W}$ induced the genes in a dense subgraph is connected.

2.5 Compendium Based Gene Expression Analysis

Lee et al. [99] perform a large scale analysis of 60 large human gene expression datasets consisting of 3924 microarrays. They identify pairs of genes that are consistently co-expressed across at least three datasets. They identify a network with 8805 genes and 220649 interactions. They show that correlation across datasets is correlated with functional relatedness. They perform hierarchical clustering and apply the MCODE algorithm [8] to find dense subgraphs in the co-expression network. They demonstrate that clusters in the network relate to functionally coherent sets of genes.

Stuart et al. [160] build a co-expression network from a set of 3182 gene expression samples across four organisms: *H. sapiens*, *D. melanogaster*, *C. elegans*, and *S. cerevisiae*. Here, each node is a *metagene*, a set of genes across multiple organisms whose protein sequences are one another's best reciprocal BLAST hits. An edge connects two metagenes if they have statistically-significant similar expression profiles in the gene expression compendium. They found 22163 such significant correlations across species. They report that many of the genes correspond to core biological functions such as the cell cycle, protein expression, and secretion. The authors show that metagenes in a number of regions of this FLN are enriched in specific biological functions. Other groups have obtained similar results [16, 176, 184].

An analysis of aging in *C. elegans* and *D. melanogaster* [112] found that many orthologous gene share transcriptional profiles across organisms.

Bergmann, Ihmels, and Barkai [16] compile a compendium of gene expression data sets from six organisms including *S. cerevisiae, C. elegans, E. coli, A. thaliana, D. melanogaster*, and *H. sapiens*. They compute modules as groups of genes that share similar expression profiles across species. They find that the node degrees in the co-expression network for each organism roughly follows a power-law distribution. Highly connected genes tend to be highly conserved with a modular transcriptional program. They also find that functionally related genes tend to have similar transcriptional profiles across species.

2.6 Reverse-engineering Gene Regulatory Networks

A number of approaches have been considered for reverse-engineering gene regulatory networks from gene expression data. The fundamental hypothesis underlying such methods is that if two genes share similar expression profiles, then the two genes may interact physically or functionally in the cell. An important issue these methods tackle is that of distinguishing between direct and indirect interactions.

Information Theoretic Approach Basso et al. [10] propose a method called ARACNe. They start by compute the mutual information between all pairs of genes: the weight of an edge connecting genes x and y is calculated

$$I(x;y) = \sum_{x \in V} \sum_{y \in V} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}.$$

V is the set of all genes, $p(x) = Pr[X = x]$ in the discrete case. Basso et al. [12] formulate the network inference problem as a reduction from a fully connected network of genes. ARACNe removes indirect links by applying the data processing inequality (DPI), which states if (x, y) and (y, z) are directly interacting then $I(x, z) < I(x, y)$ and $I(x, z) < I(y, z)$. Therefore, for each triangle in this network, the edge of lowest weight is removed. The process continues until no other edges can be removed. They prove that asymptotically their method finds the correct structure provided the structure is a tree. They apply ARACNe to 336 transcriptional profiles in B-cells. The reconstruction revealed a scale-free network structure. They compare their method to leading Bayesian network approaches using synthetic data. They find that their method produces more true positives and fewer false positives. ARACNe correctly identified many targets of the transcription factor MYC as well as several previously unknown transcription factors that they validated through biochemical assays.

Bayesian Network Representations Many researchers formulate the network inference problem probabilistically using Bayesian networks [51, 56, 140]. Using this strategy, regulatory networks are modeled as graphs. The nodes represent variables from diverse biological entities such as genes, proteins, transcriptional regulators, and phenotypes, and the edges

represent dependencies between the variables. The levels of the variables in the Bayesian network are determined by the structure of the graph. These methods search the space of graph structures that yield variable levels that best match experimental data.

Segal et al. [145] describe a method for detecting condition-specific regulatory patterns of the type "Regulator X controls the set of genes Y in condition Z." The method takes as input as set of candidate regulator genes and a collection of gene expression data sets $\mathcal{D}$. The method constructs "module networks" made up of a subset of genes (a module) and a regulatory program for the module. A regulatory program is a binary decision tree each of whose nodes represents up- or down-regulation of a single regulator and each of whose leaves represents a subset of the samples. Thus, any root-to-leaf path in the decision tree represents both a specific expression pattern of a set of regulators, samples which display this expression pattern, and the expression profiles of genes in the module in those sample. Segal et al. use a Bayesian framework to assign a likelihood score to different regulatory program structures, and they use a greedy hill climbing algorithm to identify high scoring structures. They apply their method to identify module networks in *S. cerevisiae*.

Optimization using Linear Regression Other researchers use regression [19, 54, 67, 194] to infer regulatory networks from gene expression measurements. These methods assume that the average expression of a gene g can be expressed as a linear combination of the set of regulating genes Y_g. Regression methods learn weights for each element in the regulatory set Y_g. In these methods, positive weights indicate positive regulation, and negative weights indicate inhibitory regulation. Like Bayesian methods, regression methods also seek model improvements that yield gene expression levels that best match experimental measurements over all genes.

Partial Correlations Some methods use partial correlation to infer biochemical networks from gene transcripts. Fuente et al. [52] use partial correlation to identify directed and undirected relationships between genes in a correlation network. The method works by assuming an edge between all gene pairs a and b. If there exists a pair of genes c and d such that a and b conditioned on c and d does not share a partial correlation, the edge between a and b is removed from the network. Fuente et al. apply their method to gene expression measurements from yeast. They find both directed and undirected relationships between genes. Because of a high false negative rate, the authors indicate that the method can be used for suggesting relationships between biochemical products but not a network inference approach.

Knowlege-Based Approaches Gat-Viks and Shamir [57] address the problem of pathway expansion by integrating gene expression measurements with curated biological pathways. These methods model curated biological pathways using a Bayesian network. Gat-Viks and Shamir compute a likelihood score for the model based on the comparison of the predicted variables levels to the levels given by gene expression measurements. The method proposed searches the space of model refinements that increase the likelihood score. Gat-

Viks and Shamir propose a greedy algorithm to find networks with high likelihood scores. The process of model refinement produces networks that include interactions that were previously not known to operate in the original pathway. Gat-Viks and Shamir annotate all interactions in the refined networks with the biological pathway.

2.7 Unlabeled Graph Mining

The problem of graph mining involves finding graph structures with frequent edge-disjoint embeddings in a large graph. In this section, we focus on versions of this problem where nodes in both the subgraphs searched for and the graphs(s) searched in are unlabeled.

Apriori-based approaches Inokuchi, Washio, and Motoda proposed one of the earliest frequent graph mining approaches in a method called Apriori-based Graph Mining (AGM). AGM [77] uses an *Apriori*-based algorithm for mining frequent subgraphs. The graph is represented as an adjacency matrix to facilitate itemset mining. They mine across multiple adjacency matrices to find undirected graph patterns with high support. The method was applied to both synthetic and carcinogenesis datasets. They show that their method can identify frequent atomic interaction structures.

FSG [94] proposed an Apriori-like method using a breadth-first-search procedure to find frequent subgraphs in large graph datasets. They evaluate their method using both real and artificial datasets. They show results from mining all frequent subgraphs in a graph with 200000 edges.

Depth-first-search approaches Yan and Han proposed gSpan [190] for finding frequent subgraphs. Their method assigns a canonical label to graphs based on a unique minimal depth-first-search (DFS) code. They use the DFS code to impose a lexographic ordering on the subgraphs. They propose to find frequent subgraphs by DFS. The method avoids the candidate generation phase of apriori-based approaches. The strategy of applying canonical labels to subgraphs enables them to avoid having to solve the NP-Complete subgraph isomorphism problem. They convert the problem of finding frequent connected subgraphs into one of mining minimum DFS codes. They compare their method to FSG and show the improved performance of their method on both synthetic data and a dataset of chemical compounds.

Kuramochi and Karypis [95] use the horizontal and vertical pattern discovery paradigm to find subgraphs with a frequent number of edge-disjoint embeddings in a single large labeled input graph. They propose two algorithms for frequent subgraph finding called HSIGRAM (breadth-first-search based) and VSIGRAM (depth-first search based). They propose a canonical labeling scheme for graphs based on concatenating together the values in the upper triangle of the adjacency matrix of the subgraph. They reduce the problem of finding edge disjoint embeddings of the graph into one of finding the maximal independent set (MIS). They solve the problem of finding the maximal independent set by applying an

algorithm for finding the maximum size clique in the complement graph. They apply their method to a variety of single large graphs including the web, VLSI, and author citations.

Frequent Tree Mining Ruckert and Kramer [139] propose a generalized version of space trees to facilitate graph mining in a tool called FreeTreeMiner. A space tree stores versions of tree structures in suffix trie structure. They propose their technique for identifying unrooted trees in graphs. They construct the space tree data structure using DFS. The construction involves identification of tree root nodes, and imposing an ordering on the remaining nodes. They derive canonical representations of trees from the data structure. They discuss the possibility of combining their space trees data structure with gSpan [190] to find frequently embedded subgraphs. They apply their method to two dataset from the National Cancer Institute's Developmental Therapeutics Program. They find 834 frequent trees with at least 39 support.

Branch-and-Bound with Pruning Yan et al. [189] vectorize graphs by decomposing the graph into a vector of frequent subgraphs. They propose two optimization techniques for removing infrequent subgraphs. They first technique prunes subgraphs based on structural similarity to infrequent subgraphs. The second prunes subgraphs based on frequency correlation to infrequent subgraphs. They show that their method can perform orders of magnitude faster than branch-and-bound approaches.

2.8 Mining Relational Graphs

We now turn our attention to "relational graphs", i.e., graphs whose nodes are labeled. In this context, we assume that the nodes in any single graph have unique labels. The set of algorithms discussed here solve the problem of finding subgraphs with specific properties given a set of relational graphs. Typically, these methods detect subgraph structures that are common to many graphs.

Path Mining Tohsato, Matsuda, and Hashimoto [171] describe a multiple alignment algorithm used to find similarity between metabolic pathways. Kelley et al. [88] developed the PathBlast algorithm for finding pathways that are conserved between *S. cerevisiae* and *H. pylori*. The QPath algorithm [155] takes a query path and a PPI network and identifies homologous pathways in the network, allowing both insertions and deletions of proteins in the identified paths; the query path has unlabeled nodes.

Alignment Graph Approach Sharan et al. extended the work of Kelley et al. to find complexes conserved between *S. cerevisiae* and *H. pylori* [151] and to find both pathways and complexes conserved between three species, *S. cerevisiae*, *D. melanogaster*, and *C. elegans* [153]. Ideker et al. [164] used this technique to demonstrate that the *P. falciparum* PPI

network diverges from those of other eukaryotes. Koyuturk, Grama, and Szpankowski [93] developed a model for computing CPIMs that incorporated evolutionary forces that result in gene duplication.

The Graemlin [49] algorithm uses explicit models of functional evolution to align multiple PPI networks. Graemlin permits searches for arbitrary module structures by requiring the user to specify an edge scoring matrix to encapsulate the desired module structure; when given a query module, Graemlin automatically computes the matrix from the structure of the query.

DFS and BFS Approaches Yan, Zhou, and Han [192] propose a method for mining patterns across many labeled graphs. They propose two algorithms: CLOSECUT (a pattern growth approach) and SPLAT (a pattern reduction approach). CLOSECUT is based on expanding candidates with sufficient support. Additional candidates are generated by decomposing previously expanded subgraphs by repeatedly applying a minimum cut procedure. SPLAT is based on repeatedly applying the minimum cut algorithm on the full graph and returning those parts that exceed the minimum support threshold. SPLAT identifies exact recurrences of dense subgraphs. The cut procedure returns when a pattern has been reached with sufficient support. They apply their method to yeast gene expression data and find frequent modules that correspond to basic cellular processes such as ribosome biogenesis and helicase activity.

Borgelt and Berthold [20] propose an algorithm to identify discriminatory sections of molecules that distinguish class association. The algorithm generates candidate graph structures in a bottom-up breadth-first manner. They perform support-based, size-based, and structure-based pruning of the search space. Structure-based pruning prevents consideration of isomorphisms. They apply their algorithm to the groups of chemical compounds found in the National Cancer Institute's HIV-screening database. They give results from structures from different classes of atoms including sulfur-based, nitrogen-based, and selenium-based fragments.

Summary Graph Approach Hu et al. [189] propose a method for finding frequent patterns across large networks of biological data. The method proposed is called CODENSE. The method finds approximate recurrence of dense subgraphs but requires that individual edges show high correlation across the input graph set. They combine the input graphs into a summary graph $S(V, E)$ where the edges are labeled with the number of networks where the edge is present. Edges with a label less than a threshold are removed. A second order graph $T(U, F)$ is constructed such that $U = E$ and $(a, b) \in F$ if edges a and b have a correlated presence across the input graphs greater than γ and $a, b \in U$. The apply the dense subgraph finding algorithm MODES to T. MODES is based on repeatedly finding the minimum cut in the input graph. They apply their method to 39 co-expression networks taken from diverse gene expression datasets. They use their method to predict functions for 169 uncharacterized yeast genes.

Huan et al. [70] find frequent components of a protein structural family by mining many rela-

tion graphs. They propose several methods for constructing graphs from proteins including distance thresholding, the Delaunay triangulation, and quasi-Delaunay edges. They perform a DFS for frequently occurring subgraphs. They compare the protein structural elements they find from their method to the elements contained in the Structural Classification of Proteins (SCOP) database. They report that several of the large structural elements were found by their method. They show a superposition of the conserved graph structure for 7 serine proteases.

Yan et al. [191] propose a method for finding frequent graph patterns across many large labeled graphs. They search for sets of vertices that are frequently and densely connected across the graphs. To reduce the search space they propose two methods. They divide the input graphs into non-overlapping graph sets, and they construct a summary graph from the set of graphs. Their method is composed of four parts: partitioning of graphs, summary graph construction, dense subgraph computation, and density refinement. They propose a scheme for computing the weights of edges in the summary graph as the normalized count of triangles containing the edge. They apply their method to 105 human microarray datasets. They represent each microarray dataset by a co-expression graph. They show that both density and frequency of recurrence are correlated with functional relatedness. They report many conserved graph structures that may represent regulatory modules. They perform ChIP-chip experiments and verify that some of the networks correspond to regulatory modules. They identify a correlation between the support of the conserved graph structure and the likelihood of it being a regulatory module. They compare their method to modules found in individual graphs, modules found across all graphs, modules found in pre-defined subsets of graphs, and modules found in nearest-neighbor subsets of graphs. Their results show that predefining graph sets using nearest-neighbor clustering produces the most homogeneous modules. They show that a hybrid approach called NeMO combines the summary graphs for the the nearest-neighbor approach with the partitioning approach has the most frequently homogeneous modules overall.

Chapter 3

Sensitive Detection of Pathway Perturbations in Cancers

3.1 Introduction

Complex diseases such as cancer are associated with the alteration or dis-regulation of multiple pathways and processes in the cell. Discovering and cataloguing which pathways are perturbed in each type of cancer is important for improving our understanding of the mechanisms underlying these diseases. In particular, such studies can pin-point pathways that may be uniquely perturbed in one or a small number of related cancers, thus providing potential targets for therapeutic studies.

Many methods have been developed to study the activation of pre-defined gene sets in human diseases and tissues [11, 30, 39, 44, 61, 64, 71, 89, 102, 126, 127, 162, 169]. In this context, a "gene set" is usually taken to be a collection of genes that share a common attribute, e.g., Gene Ontology annotation or membership in a pathway. For instance, Subramanian et al. [162] developed "Gene Set Enrichment Analysis" to test if a gene set is differentially expressed in two phenotypes by ranking all genes by some measure (e.g., the t statistic) and using a modified Kolmogorov-Smirnov statistic to decide whether the genes in the gene set have surprisingly high or low ranks. Segal et al. [144] used a hierarchical clustering algorithm to combine pre-defined gene sets into modules. They characterized gene-expression profiles in specific (sets of) tumors as a combination of activated and de-activated modules.

These methods ignore physical or functional interactions between the genes (or their products) in a gene set. Analysis of gene expression measurements in the context of the interaction structure inherent in a pathway can take into account both perturbations in gene expression and the topological properties of the network. More recent methods have sought to capture information about the activation of a pathway from the perspective of the interactions in the pathway. Draghici et al. [42] combined the hypergeometric model with an additional weighted term that measures how well the data matches the expected pattern of induc-

tion and repression. Efroni et al. [45] used pathway perturbation measurements to predict prognosis and tumor grade. Both approaches measure the perturbation of a pathway in its entirety. Thus, they may not be sensitive to situations when only a sub-pathway is highly perturbed.

In this chapter, we develop a systematic methodology to detect which pathways are perturbed in a disease. Here, we use the term *pathway* to refer to a network of physical interactions between genes and gene products that together perform a specific biological function. Given (the interactions in) a pathway P (e.g., the B-cell receptor pathway) and genome-wide gene expression measurements for a disease phenotype (e.g., Acute Myelogenous Leukemia) and a control phenotype (e.g., normal blood cells), our method computes the sub-pathway of P that is most perturbed in the disease (when compared to the control). The computation of the most perturbed sub-pathway distinguishes our approach from existing methods. Our approach is sensitive to the possibility that the pathway is not perturbed in its entirety but some portion of it is significantly perturbed.

3.2 Methods

Overview of Our Algorithm Our algorithm takes the interactions in a pathway P and gene expression measurements for a disease phenotype and a control phenotype as input. We first assess the differential expression of each gene in P. We develop a statistic based on the Liptak-Stouffer z-score that measures the combined perturbation of the genes in P. This statistic takes into account both the interactions in P and the differential expression of each gene. We use this statistic to compute which sub-pathway of P is maximally perturbed. Finally, we use a permutation-based test to assess the statistical significance of the maximally-perturbed sub-pathway.

3.2.1 Condition-Specific Pathway Activation

We define a *pathway* $P = (G, I)$ to be a graph composed of a set G of genes and a set I of physical or functional interactions between the genes in G or their gene products. Typically, P may be composed of multiple connected components. Given genome-wide gene expression measurements in multiple patients diagnosed with a disease d in a tissue and from normal samples of that tissue, our goal is to determine whether the pathway $P = (G, I)$ is perturbed in the disease (when compared to normal tissue) and to compute the subgraph of P that is most perturbed in the disease. We note that our pathway perturbation algorithm described here can take as input any gene expression pre-processing method that computes p-values for gene perturbation.

For each gene $g \in G$, let $p(g)$ denote the p-value of its differential expression in d (when compared to normal tissue). We compute $p(g)$ as the p-value of the two-sided t-test under the null hypothesis that the distributions of the expression values of g in the disease samples and in the normal samples have identical means (but may have different variances). We

convert the p-value into a z-score $z(g) = N^{-1}(1 - p(g))$, where N^{-1} is the inverse of the normal cumulative distribution function. At this stage, we do not impose a cut-off on $z(g)$. Instead, we include all genes in subsequent analysis. The rationale for this choice is that while individual genes may not be differentially expressed to a statistically-significant extent, significant perturbations may be noticeable at the level of sets of genes [118].

The method we develop takes the interaction structure of P into account. Let $Q = (G', I')$ be a subgraph of P. We define the *degree* $d_Q(g)$ of a gene $g \in I'$ to be the number of interactions in I' that are incident on g. We define the *perturbation* of a subgraph $Q(G', I')$ of P to be the weighted Liptak-Stouffer z-score [148]

$$z(Q) = \frac{\sum_{g \in G'} d_Q(g) z(g)}{\sqrt{\sum_{g \in G'} d_Q^2(g)}}.$$

The numerator of $z(Q)$ is the weighted sum of the z-scores of all genes that appear in Q, where each gene is weighted by the number of interactions in Q that are incident on it. Dividing by the square root of the sum of squared gene degrees ensures that $z(Q)$ is normally distributed with mean 0 and standard deviation 1. Thus, this formulation of perturbation combines p-values over multiple genes in a statistically-sound way [68]. Each gene in Q contributes both its z-score and its degree in Q to $z(Q)$. Thus, $z(Q)$ incorporates both the differential expression of the genes in Q as well as the network of interactions between them. While other meta-analytic measures may also be suitable [68, 73], we focus on the Liptak-Stouffer score in this paper.

3.2.2 Computing the Most-Perturbed Sub-Pathway

Among all subgraphs of P, let $\widehat{P}$ be the one with maximum value of perturbation. Since $\widehat{P}$ is the most differentially-perturbed subgraph of P, we use its perturbation to assess the overall perturbation of P. Thus, our formulation does not require that every gene in P be differentially expressed in order for us to declare that P itself is perturbed in the disease. To complete our algorithm, it suffices to describe how to compute $\widehat{P}$. Note that we do not require that $\widehat{P}$ be connected, since P itself may not be connected. Ideker et al. demonstrated that a similar problem is $\mathcal{NP}$-complete [76]. Hence, we use a heuristic approach based on simulated annealing (SA), as sketched in Algorithm 1. To initialize $\widehat{P}$, we include each interaction in P with a uniform probability of 0.5. We perform the following series of operations for $100|I|$ iterations. (Recall that I is the set of interactions in P.) We select a node or an edge uniformly at random from P. Let the selected element be a. If a is already in $\widehat{P}$, we delete it from $\widehat{P}$; if a is a node, we also delete all edges incident on a from $\widehat{P}$. If a is not a member of $\widehat{P}$, we add it to $\widehat{P}$; if a is a node, we insert into $\widehat{P}$ all edges incident on a in P. Let $\widehat{P}'$ be the resulting subgraph. We compute $z(\widehat{P}')$ and compare it to $z(\widehat{P})$. If $z(\widehat{P}')$ is larger, we accept the modification and continue, since we have increased the z-score. Otherwise, we accept the modification with a probability of $e^{(z(\widehat{P})' - z(\widehat{P}))/T}$, where T is the temperature in the current iteration. Over all the iterations, we decrease the temperature T geometrically from $T_s = 100$ to $T_e = 10^{-5}$. We output the final value of $\widehat{P}$.

Remarks: For pathways of different sizes, we experimented with performing more than $80 \times |I|$ iterations. We did not find a significant benefit from increasing the number of iterations. Figure 3.1 (a) compares the simulated annealing algorithm with 80 times the pathway size to 10000 times the pathway size. The figure includes an $x = y$ dashed line for reference. We find that even with many more iterations and a slower cooling schedule, the additional iterations do not significantly improve the result. We also found that including the addition and deletion of nodes (along with incident edges) yielded better performance over addition and deletion of edges alone. Figure 3.1 shows the improvement of the simulated annealing algorithm with the inclusion of node additions and subtractions. We find that a number of iterations more than 80 times the number of edges has little to no effect on performance. Figure 3.2 shows a comparison with a simulated annealing algorithm with addition and deletion of both nodes and edges to a simulated annealing algorithm that only allows the addition and deletion of edges. We find that the inclusion of node additions and deletions positively effects the performance of the algorithm.

Algorithm 1 Compute $\widehat{P}$, the subgraph of P with the maximum perturbation.

Initialize $\widehat{P}$ by including each interaction in P with probability 0.5.
$T \leftarrow T_s$
for $i = 1 \dots 100|I|$ **do**
  $\widehat{P}' \leftarrow \widehat{P}$
  Select a node or an edge $a \in P$ uniformly at random.
  if p is in $\widehat{P}$ **then**
    Delete p from $\widehat{P}'$.
  else
    Insert p into $\widehat{P}'$.
  end if
  if $z(\widehat{P}') > z(\widehat{P})$ **then**
    Set $\widehat{P}$ to be $\widehat{P}'$
  else
    Set $\widehat{P}$ to be $\widehat{P}'$ with probability $e^{(z(\widehat{P}')-z(\widehat{P}))/T}$
  end if
  $T \leftarrow T \times e^{\frac{\log(\frac{T_e}{T_s})}{100|I|}}$
end for

3.2.3 Estimating the Statistical Significance of Perturbed Pathways

A potential drawback of our definition of $z(\widehat{P})$ is that it assumes that the z-scores of the individual genes are independent. To ensure that $z(\widehat{P})$ is not an over estimate of the significance of a perturbed pathway as a result of this assumption, we perform a permutation-based test. To build a null distribution for disease d and pathway P, we repeat the following procedure $100,000$ times:

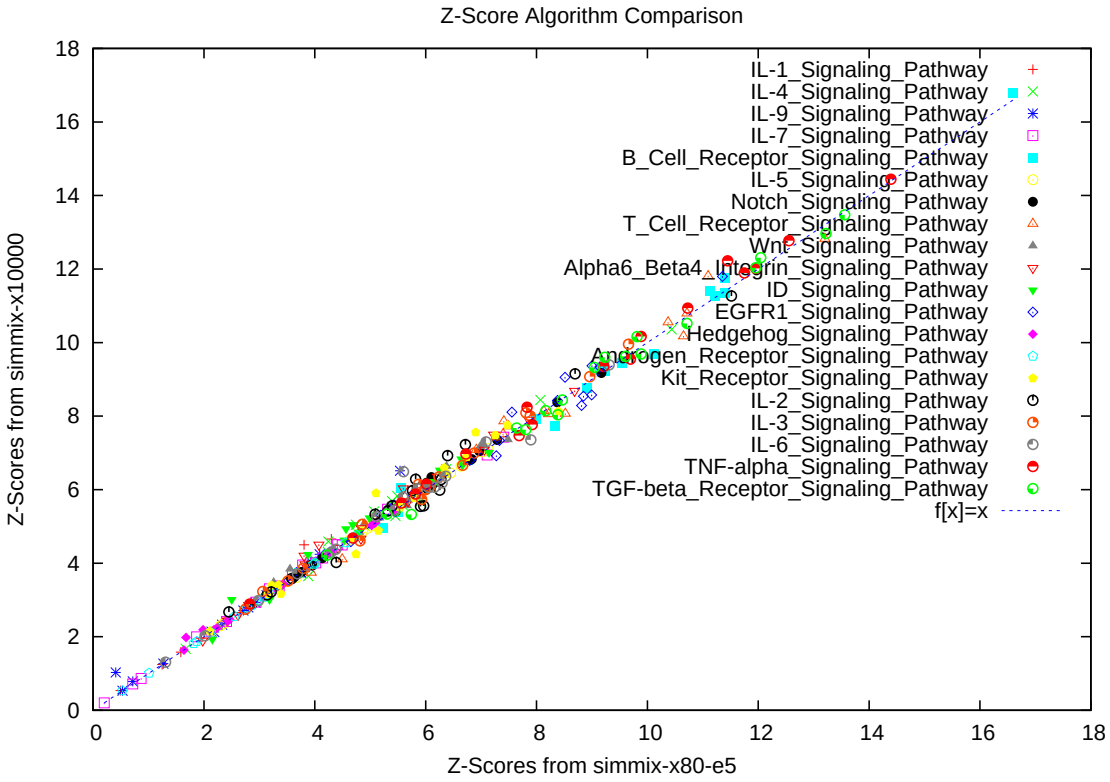


Figure 3.1: Simulated annealing algorithm comparison. For a pair of algorithms we show a scatterplot of results taken from the GCM dataset. A dashed $y = x$ line is included for reference. Each pathway is given a distinct symbol to help identify pathway specific trends. We find that all of the pathways in the study follow common trends. The simmix algorithm includes additions and subtractions of both nodes and edges. The effect of additional iterations of simulated annealing on algorithm performance. We compare the performance of simmix with a number of iterations equal to 80 times pathway size with its performance for a number of iterations equal to 10000 times pathway size.

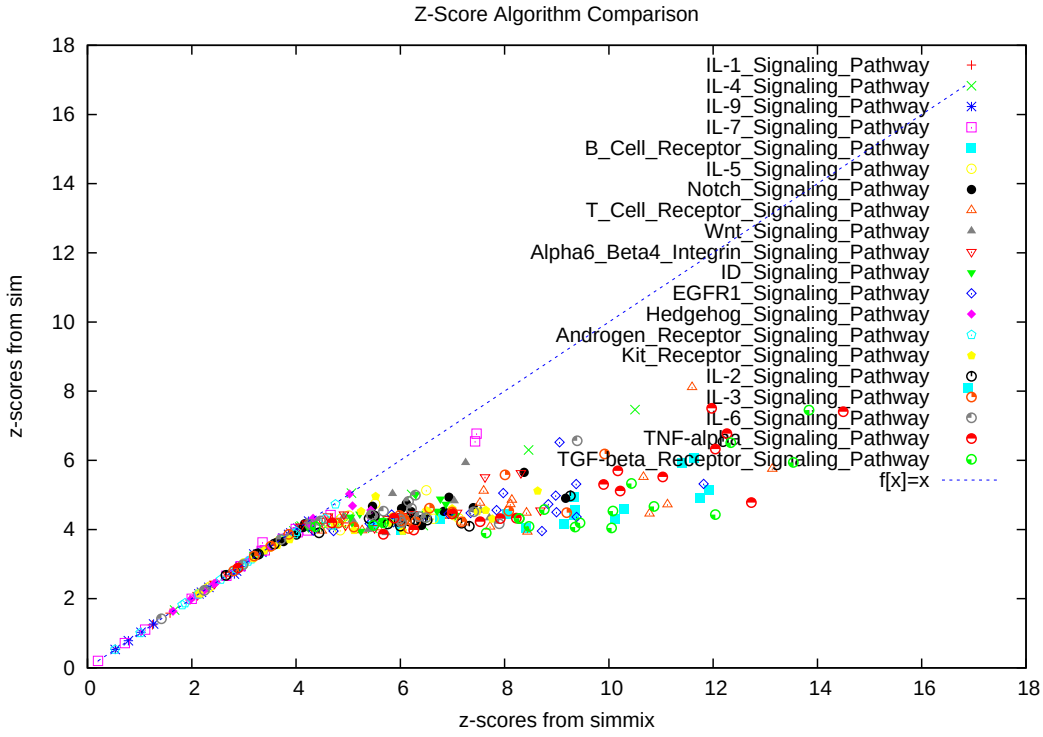


Figure 3.2: Simulated annealing algorithm comparison. For a pair of algorithms we show a scatterplot of results taken from the GCM dataset. A dashed $y = x$ line is included for reference. Each pathway is given a distinct symbol to help identify pathway specific trends. We find that all of the pathways in the study follow common trends. The simmix algorithm includes additions and subtractions of both nodes and edges. The algorithm sim allows only edge additions and subtractions. We compare simmix to sim with a number of iterations equal to 80 times the pathway size.

(i) Permute node labels (and associated gene expression data) in the pathway. Let k be the number of genes in P. Replace these k genes with k other genes, selected uniformly at random from the universe of all genes measured in the gene expression data set for d and present in a protein interaction network containing 9352 genes and 39890 interactions (assembled from multiple sources [101, 132, 138, 159]). Let $\tilde{P}$ be the new pathway. Note that $\tilde{P}$ and P are isomorphic to each other, i.e., they have identical interaction structures.

(ii) Obtain the z-scores of the genes in $\tilde{P}$ from the gene expression data set for d.

(iii) Use the simulated annealing algorithm to compute $z(\tilde{P})$.

We compute the p-value for $z(\widehat{P})$ as the fraction of random trials where $z(\tilde{P}) > z(\widehat{P})$. Since we are testing multiple pathway-disease pairs, we control the false discovery rate using the method of Benjamini and Hochberg [15]. We use the adjusted p-value in all of the subsequent analysis.

3.2.4 Pathway and Disease Association Analysis

Here, we describe our method for pathway association analysis. We compute associations between diseases in an analogous manner. The inputs to pathway association analysis are a set $\mathcal{P}$ of pathways, a set $\mathcal{D}$ of diseases, and the p-values of perturbation of each pathway $P \in \mathcal{P}$ in every disease $d \in \mathcal{D}$. Our method uses two parameters t and c. Given a p-value threshold $t > 0$, let $\mathcal{D}_P(t)$ denote the set of diseases that perturb P with p-value at most t. The parameter c controls when we deem two pathways to be associated. Given two distinct pathways P and Q, we say that $P \preceq Q$ if

(i) $c < \dfrac{|\mathcal{D}_P(t)|}{|(\mathcal{D}_P(t) \cup \mathcal{D}_Q(t)|)} < 1$ and

(ii) if there is no third pathway R such that $P \preceq R$ and $R \preceq Q$.

To compute all pairs of pathways related by $\preceq$, we first compute all pathway pairs that satisfy the first condition in the definition of $\preceq$. These pathway pairs induce a directed graph D. In order to satisfy the second condition, we compute the transitive reduction of D. Note that if c is small enough, D may contain many two-node cycles. On the other hand, if $c = 1$, D is empty. We select the smallest value of c such that there is no cycle with more than two nodes in D. In this chapter, we used $t = 0.05$ and $c = 0.8$.

3.3 Results

We obtained curated pathways from the Netpath database [121]. These 20 pathways include 10 proliferation-associated signalling pathways and 10 immune response signaling pathways.

We used gene expression measurements in the Global Cancer Map (GCM) [133]. The GCM dataset contains 190 samples spanning 18 cancers (adenocarcinomas of the breast, colon, lung, ovary, pancreas, prostate, and uterus; follicular and large B-cell lymphomas; melanoma; bladder; acute lymphoblastic leukaemias of the B cell and T cell; acute myleogenic leukaemia; renal carcinoma; mesothelioma; and glioblastoma and medulloblastoma, which are two cancers of the central nervous system) and 90 samples from 13 normal tissues (bladder, breast, cerebellum, colon, germinal center, lung, kidney, ovary, pancreas, peripheral blood, prostate, uterus, and whole brain). We compared the samples for each cancer in the dataset to the samples from the corresponding normal tissue (e.g., prostate cancer and normal prostate). For each gene and each cancer, we determined the set of expression values from the cancer samples and the set of expression values from the associated normal samples. We applied our algorithm to 360 cancer-pathway pairs (18 cancers times 20 pathways).

3.3.1 Interaction Perturbation within Pathways

The pathways in Netpath are carefully curated and we consider them canonical for the purposes of this study. A natural question that arises is whether the most-perturbed sub-pathway of a pathway P contains a significant fraction of the interactions in P. For each pathway, we counted how many interactions appeared to be perturbed in at least one cancer (considering only p-values less than 0.05). Table 3.3 shows that in as many as twelve pathways, fewer than 75% of the interactions in the pathway are perturbed.

3.3.2 Significance of Partial Pathway Activation

When a signaling pathway is perturbed, not all components of the pathway will undergo transcriptional perturbation, because many perturbations occur that the post-transcriptional or past translational level. Thus when only transcriptional data are available, many pathways may appear to be partially perturbed. An important innovation in our approach is the ability to sensitively detect partial pathway perturbation. In Figure 3.4, for each of the 360 pathway-cancer pairs, we plot the z-score measuring the perturbation of the entire pathway in the cancer (x-axis) against the z-score of the most-perturbed sub-pathway in that cancer (y-axis). Since each point in the plot is above the $x = y$ line, this comparison clearly demonstrates that calculating perturbation at the sub-pathway level is substantially more sensitive than calculating it at the whole pathway level.

3.3.3 Comparison to GSEA

Our approach explicitly uses the interaction structure of pathways to calculate their perturbation. To assess the advantages of this approach, we compared our method to the gene-oriented method "Gene Set Enrichment Analysis" (GSEA) [162]. GSEA compares two phenotypes of interest by sorting all the genes based on the difference in their expression profiles in the two phenotypes, e.g., by using the t statistic. Given a gene set of interest,

GSEA uses a modified Kolmogorov-Smirnov statistic to test whether the genes in the gene set are ranked toward the top or the bottom of the sorted list. GSEA measures the statistical significance of an observed score by repeatedly permuting the phenotype labels of the samples.

We converted each Netpath pathway into the set of genes that are members of the pathway. We performed each of the 360 pathway-cancer comparisons using GSEA, ranking genes by the t statistic and generating 100,000 random permutations to assess the statistical significance of the computed scores. GSEA identified no Netpath gene set as significant, even with an FDR-adjusted p-value less than 0.1, in any of the comparisons. We had observed that perturbed pathways computed by our method may contain both up- and down-regulated genes. We reasoned that GSEA may not detect corresponding gene sets as significantly differentially expressed since these gene sets contain both genes with low ranks (large positive t statistics) and with high ranks (large negative t statistics). Therefore, we repeated the analysis using GSEAs option to rank genes by the absolute value of the t statistic. Even with this option, GSEA identified no pathway-cancer pairs as significant, even at the 0.1 level.

GSEA uses the null hypothesis that the distribution of the perturbation of the genes in

Figure 3.3: Statistics on the number of interactions in perturbed sub-pathways. We say that an interaction is *measured* if both associated genes have been measured in the gene expression experiments. We report statistics only for pathways that are perturbed in at least one cancer with a p-value at most 0.05. For each pathway, we report the number of cancers where the pathway is perturbed, the number of measured interactions in a pathway, the number of interactions significantly perturbed in at least one cancer, and the percentage of measured interactions that appear in at least one most-perturbed sub-pathway.

Pathway	**#cancers**	**#measured interactions**	**#perturbed interactions**	**%perturbed interactions**
IL-5	3	26	7	26.9
IL-9	1	11	4	36.4
IL-3	4	58	29	50.0
IL-2	3	75	39	52.0
Kit Receptor	7	55	31	56.3
EGFR1	10	78	45	57.6
IL-6	7	46	27	58.7
Alpha6 Beta4 Integrin	4	32	19	59.4
Notch	2	25	15	60.0
IL-4	2	36	23	63.9
T Cell Receptor	7	102	75	73.5
TGF-beta Receptor	15	155	114	73.5
ID	6	53	41	77.4
IL-7	3	20	16	80.0
B Cell Receptor	13	105	87	82.9
TNF-alpha	9	109	94	86.2

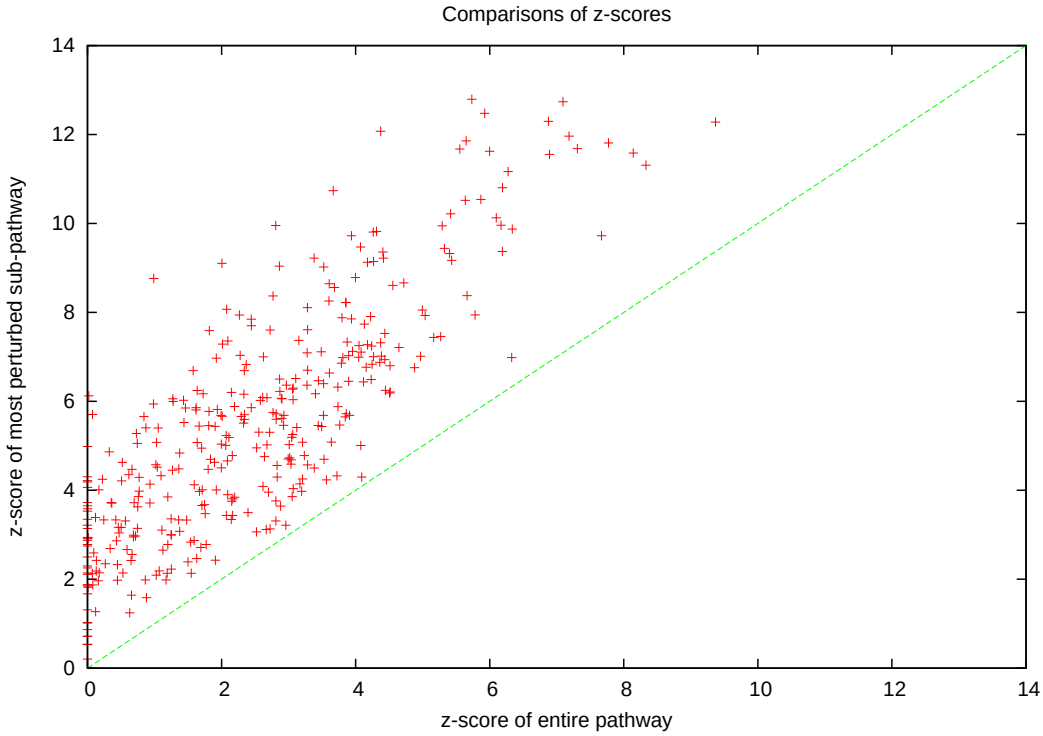


Figure 3.4: Results of computing perturbations of 20 Netpath pathways in 18 human cancers. Each point represents a pathway-cancer pair; the x-axis is the z-score of the entire pathway and the y-axis is the z-score of the most-perturbed sub-pathway, with larger z-scores indicating greater perturbation. All points lie above the $x = y$ line (dashed green) indicating that we can always find a part of each pathway that is more significantly perturbed than the entire pathway.

a particular gene set is the same as the distribution of the rest of the genes measured in the transcriptional data set. Our approach uses the null hypothesis that the distribution of the perturbation of the genes in a particular pathway P is the same as the distribution of an equal number of randomly-selected genes, where the interactions between the randomly-selected genes are isomorphic to the interactions in P. To test the hypothesis that the stricter null hypothesis of GSEA prevents it from finding significant perturbations detected by our method, we used our results to construct a new gene set for each cancer. Each new gene set was composed of only those genes that participate in at least one of the most-perturbed sub-pathways in that cancer as determined by our method. For 13 out of the 18 cancers, GSEA found that the combined gene set constructed based on our results was more significant than the gene set for any individual pathway. Yet, only three of these combined gene sets had an FDR-corrected p-value less than 0.05. From this comparison with GSEA, we conclude that incorporating interaction structure is an important aspect of determining pathway perturbation.

3.3.4 Pathway Perturbation Results

Given gene expression measurements for the cancers in the GCM dataset, we assembled the differential perturbation of each pathway in each cancer into the matrix shown in Figure 3.5. Of the 360 pathway-cancer pairs we analyzed, 40 pairs had FDR-corrected p-values less than 0.01, 96 pairs had p-values less than 0.05 and 169 pairs had p-values less than 0.1. Four pathways are perturbed in at least half the cancers, with p-value less than than 0.05: TGF-beta receptor pathway (15 cancers), B-cell receptor pathway (13), EGFR1 pathway (10), and TNF-alpha pathway (9). These four pathways are commonly perturbed by four cancers: ALL-T, large B-cell lymphoma, Follicular Lymphoma, and Melanoma. Four pathways, including the Androgen receptor, Hedgehog, IL-1, and Wnt signaling pathways were not significantly perturbed by any condition in our dataset. Melanoma and bladder cancer perturb as many as 11 and 13 pathways, respectively. Two cancers of the central nervous system (CNS)—glioblastoma and medulloblastoma—perturb four and five pathways respectively, with two shared pathways. Prostate adenoma did not significantly perturb any of the pathways in our dataset.

3.3.5 Pathway and Cancer Association Results

To identify potential functional associations between groups of pathways and groups of cancers, we computed dependencies between pairs of pathways of the form pathway 1 is almost always perturbed when pathway 2 is perturbed. This association analysis integrates the perturbation of pathways in all the diseases, producing a directed graph of connections between pathways that suggest potential dependencies between pathways and their perturbations (Figure 3.6); note that the directed relationships between the pathways in this figure do not necessarily imply directed regulatory associations. We performed a similar association analysis on the diseases to produce a directed graph in which an edge connects a disease that perturbs many pathways to one that perturbs almost a subset of pathways perturbed by

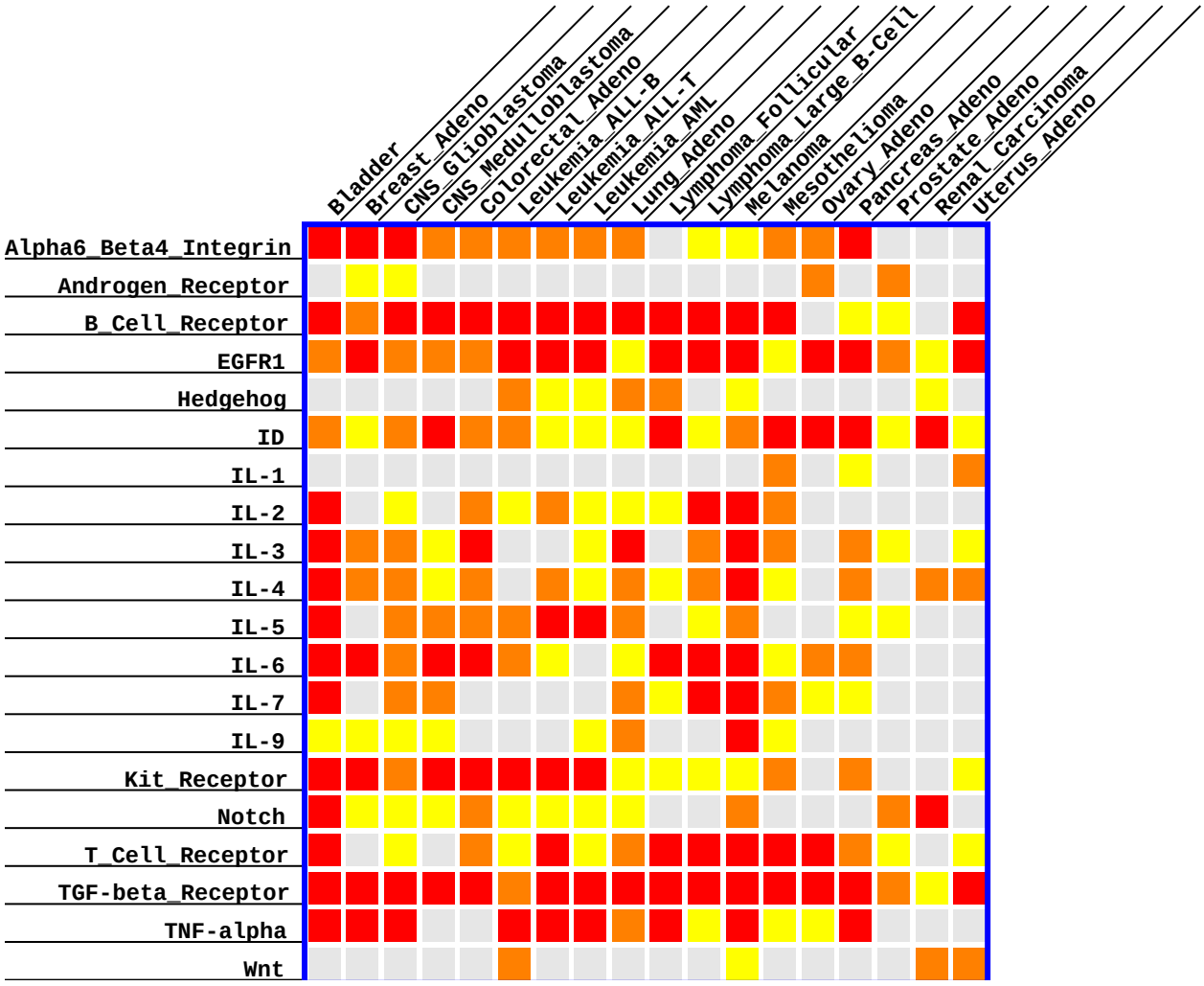


Figure 3.5: An overview of the perturbations of 20 Netpath pathway in 18 cancers in the GCM dataset. Each row is a pathway and each column is a cancer. The color of a cell indicates the FDR-corrected p-value of the perturbation of a pathway in a cancer: red ≤ 0.05, orange ≤ 0.1, yellow ≤ 0.2, and gray > 0.2.

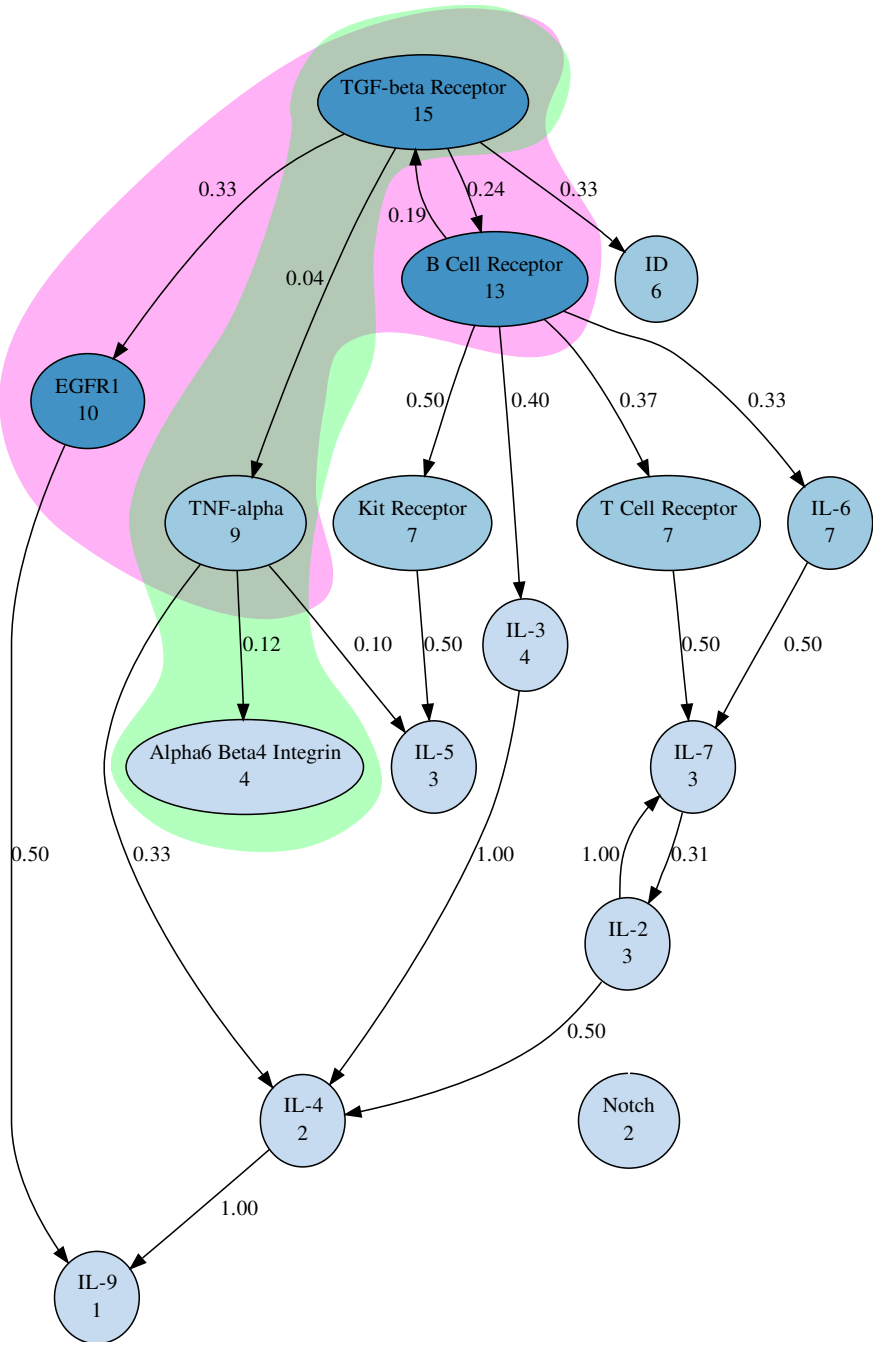


Figure 3.6: The pathway association graph is derived from pathway-cancer pairs with a perturbation p-value at most 0.05. Each node is a pathway. A node's label is the number of cancers in which the pathway is perturbed. An edge connects two pathways if the pathway at the tail of the edge is perturbed in at least 80% of the cancers that either pathway is perturbed in. The edge labels indicate the fraction of gene overlap between associated pathways. The most interesting associations are those where the connected pathways share few overlapping genes. We discuss two sets of pathways (inside the magenta and green areas) in the text.

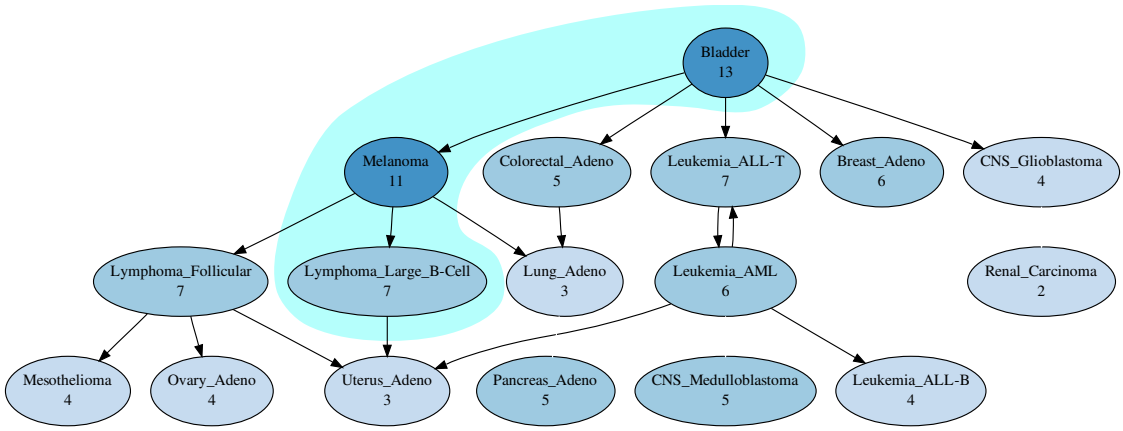


Figure 3.7: The cancer association graph is derived from pathway-cancer pairs with a perturbation p-value at most 0.05. Each node is a cancer. A node's label is the number of pathways that the cancer perturbs. An edge connects two cancers if the cancer at the tail of the edge perturbs at least 80% of the pathways perturbed in either cancer. We discuss the association between the three cancers inside the cyan area in the text.

the first disease (Figure 3.7) Disease phenotypes at the "roots" of this graph, which broadly activate many pathways, may be pleiotropic. In contrast, disease phenotypes at the "leaves" of the graph may be the result of the perturbation of a small number of finely-tuned pathways. Below, we discuss three specific associations, indicated by shaded areas in Figures 3.6 and 3.7. We also highlight these associations in matrix form in Figure 3.8.

3.3.6 Common perturbation of the TGF-beta receptor, B cell receptor, TNF-alpha, and EGFR1 pathways

The pathway association graph in reveals a strong association between these four pathways (highlighted by the magenta area in Figure 3.6 and the magenta rectangle in Figure 3.8), each of which is perturbed in at least half the cancers (as seen in Figure 3.5). TGF-beta regulates cell growth and arrests the cell cycle [41]. Deactivation of the TGF-beta signalling pathway is a hallmark of many cancers [41, 113, 114]. The EGFR1 pathway regulates growth in many cells. The characteristic over-expression of the EGFR pathway in many cancers [135, 147] is well represented in our results. We find consistent deactivation of the B cell receptor pathways across many of the GCM cancers. B-cell receptor pathway perturbation plays an active role in immune signaling and response [47]. TNF-alpha is constitutively expressed at a low level in normal cells; it activates the immune response and also initiates apoptosis. Since the TNF-alpha signaling pathway is involved in apoptosis, it is inhibited across many cancers. This is reflected in the results of our analysis.

3.3.7 Perturbations of the TNF-alpha, TGF-beta, and Alpha6 Beta4 Integrin Pathways May Suggest Metastatic Potential

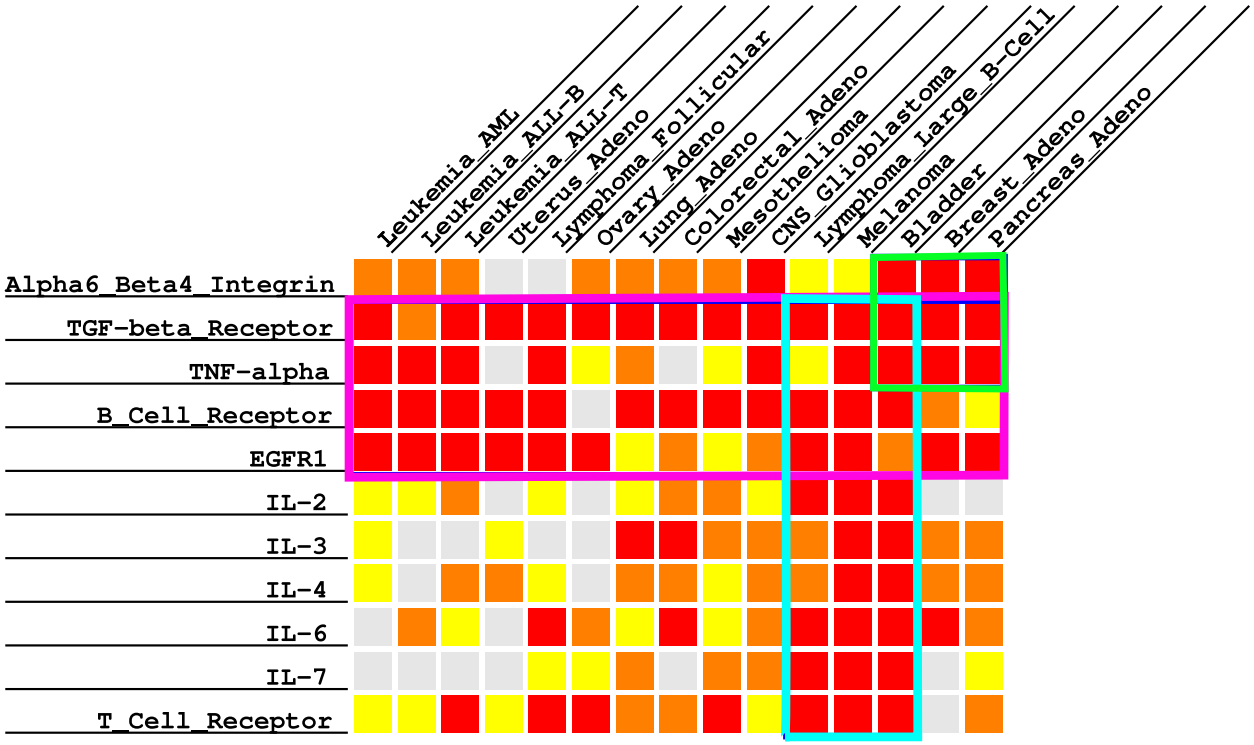


Figure 3.8: Perturbation results for a subset of pathway-cancer pairs. Rows and columns are omitted or reordered in comparison to Figure 3.5. We discuss three sets of pathway-cancer pairs in the text. The pathways in the magenta (respectively, green) rectangle correspond to the pathways in the magenta (respectively, green) area in Figure 3.6. The cancers in the green rectangle correspond to those in the green area in Figure 3.7.

Our association analysis suggests a link between the TNF-alpha, TGF-beta, and Alpha6 Beta4 integrin signaling pathways, which are part of the green area in Figure 3.6 and the green rectangle in Figure 3.8. As noted in this figure, the gene overlap between these pathways is low (0.04 between the TGF-beta receptor and the TNF-alpha receptor pathways and 0.12 between the TNF-alpha and the Alpha6 Beta4 signaling pathways), suggesting that genes common to these pathways are not sufficient to explain this association. Perturbations of the TNF-alpha, TGF-beta, and Alpha6 Beta4 integrin signaling pathways are correlated across multiple cancers including breast, bladder, and pancreas cancer. In particular, we find repression of these pathways in breast and bladder cancers and induction in pancreatic cancer, as displayed in the pathway layouts in Figure 3.9. We believe that the correlated expression of the TGF-beta, TNF-alpha, and Alpha6 Beta4 pathways corresponds to metastatic potential. It is well known that cancer is characterized by uncontrolled proliferation, resistance to cell death, and metastasis. Metastasis is the process by which cancer cells disassociate from the extracellular matrix, travel to both near and distant tissues, and rebind to the extra cellular matrix. Metastasis is a major cause of death by cancer,

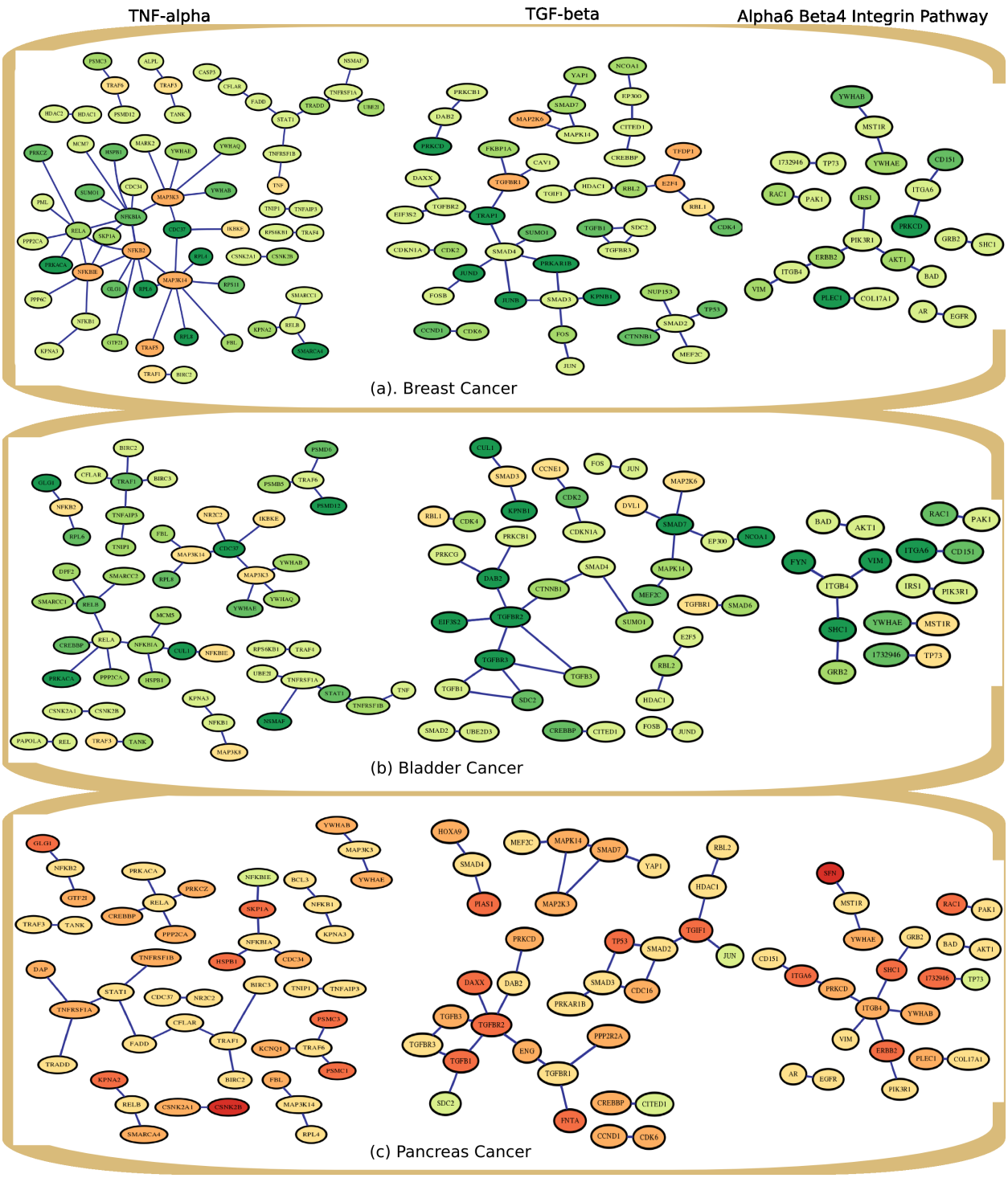


Figure 3.9: The image shows the correlated perturbation between TNF-alpha, TGF-beta, and Alpha6 Beta4 integrin signaling pathways with respect to breast, bladder, and pancreas cancers. The color of a gene corresponds to differential expression of the gene in the disease state. See Figure 3.10 for the color scale. The figure shows that breast and bladder cancers strongly down regulate expression of the TNF-alpha, TGF-beta, and Alpha6 Beta4 integrin signaling pathways, while pancreas cancer induces all three pathways.

as surgical resection of primary tumors can enhance survival.

Characteristics of invasive pancreatic cancer A loss of the ability of TGF-beta to inhibit cell growth is a hallmark of many types of invasive cancer [60]. Once TGF-beta can no longer curtail cell growth, it is common to find increased TGF-beta mRNA expression in invasive cancer [60]. Gold et al. [60] note that several cancers including pancreatic cancer undergo disregulation of cell growth by TGF-beta, a phenomenon reflected in our analysis. The dual role of TGF-beta as tumor suppressor and tumor activator is outlined by Bachman and Park [6]. The increased expression that we observe in pancreatic cancer for signaling pathways TNF-alpha and TGF-beta is supported by other studies [130]. Pancreatic cancer is associated with acutely poor prognosis [175]. The recurrence of pancreatic cancer after surgical resection is particularly high due in part to the stimulation by TNF-alpha of molecules involved in adhesion [87, 175].

Down-regulated expression of Integrin-associated pathways in breast and bladder cancer Our analysis shows that the TNF-alpha, TGF-beta, and Alpha6 Beta4 integrin pathways are correlated and down expressed in breast and bladder cancer, as shown in Figure 3.9. Initially, metastasis requires decreased cellular adhesion to the extracellular matrix to allow cellular mobility. Breast cancer showing repressed integrin perturbation may correspond to low-grade pre-metastatic cancer. Mukhopadhyay et al. [119] found low levels of Alpha6 integrin expression in poorly tumorigenic and non-metastatic breast cancer. We observe the corresponding inhibition of the Alpha6 Beta4 integrin pathway in our results. In contrast, high-grade aggressively-metastatic breast cancer has been associated with high levels [119] of Alpha6 integrin expression.

Jones et al. [80] found that breast cancer down regulates integrins compared to normal tissues. Ziober, Lin, and Kramer [198] find that Alpha2, Alpha3, and Alpha6 Beta4 integrins are all down regulated in expression in bladder and breast cancers. Breast cancer invasiveness was greatly reduced by the presence of TGF-beta [80]. The correlation we find between TGF-beta, and Alpha6 Beta4 integrin pathways in the GCM data supports their findings.

Integrins and Metastasis Integrins are transmembrane proteins responsible for cellular adhesion and signal transduction [117]. Ziober, Lin, and Kramer [198] outline the important role that integrin proteins play in metastasis. These proteins bind to the extracellular matrix and permit invasion into the basement membrane and give increased access to blood vessels. Ziober, Lin, and Kramer [198] also note the dynamic change in regulated expression of integrins during tumor development and suggest that the integrin expression changes can indicate shift towards a highly invasive metastatic cancer. For many integrins, down-regulated expression is associated with release of cells from the extracellular matrix, enhanced cellular mobility, and weak metastatic potential. Up-regulated integrins are associated with increased adhesion to the extracellular matrix and invasive cancer growth. The dynamic shift from down- to up-regulated integrin expression may be a useful therapeutic indicator of the development of a more invasive metastasis. We conclude that the patterns of decreased

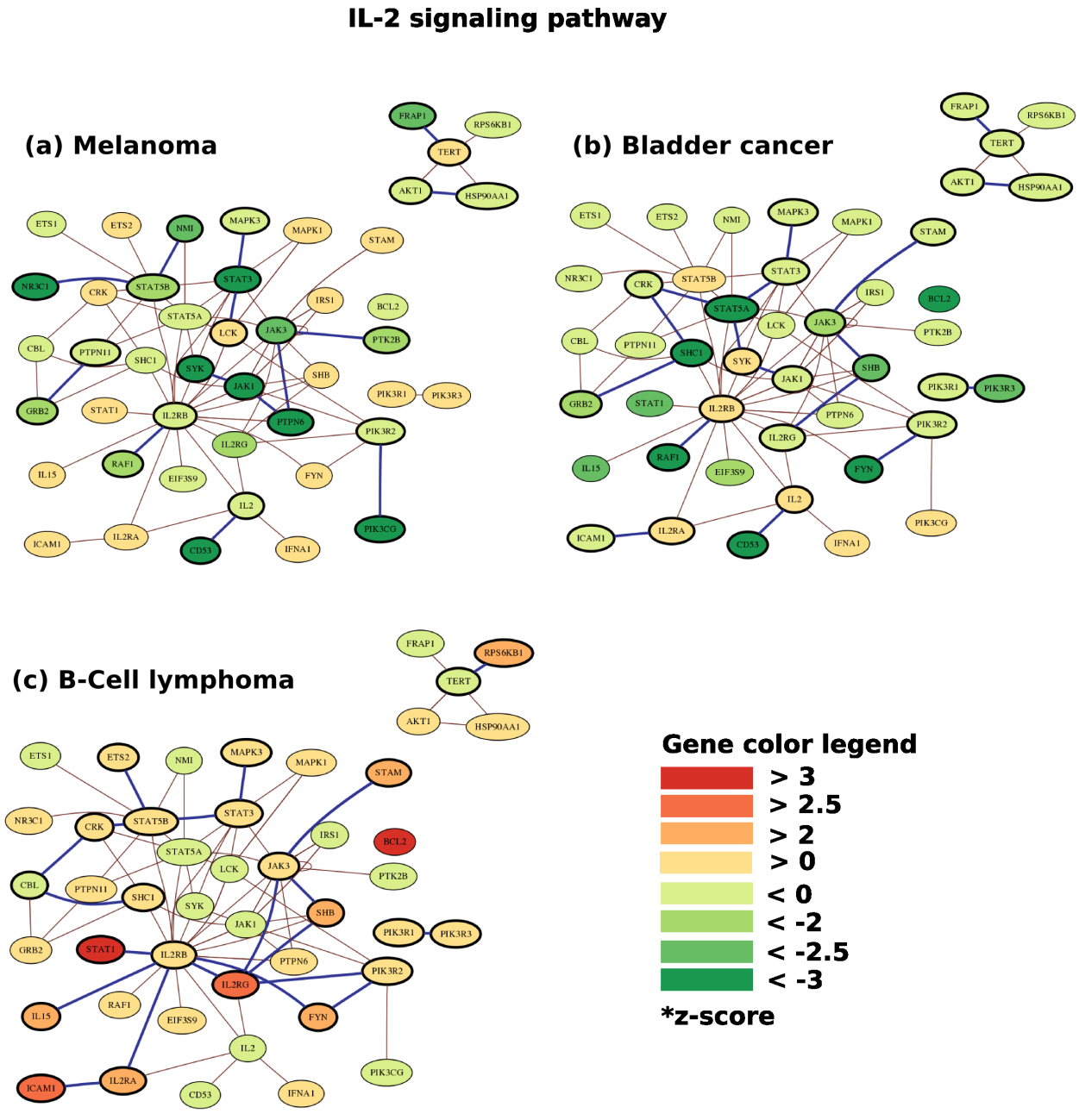


Figure 3.10: The perturbation of the IL-2 signaling pathway in melanoma, bladder cancer and large B-cell lymphoma. Here we show gene products as nodes and curated interactions as edges between those nodes. Red and orange represent levels of mRNA induction, and dark green and light green represent levels of repression. Thick dark blue edges denote interactions that are members of most-perturbed sub-pathways while thin gray edges are other interactions that belong to the pathway.

breast and bladder cancer expression measured in the GCM may indicate early tumor onset with reduced metastatic potential. In contrast, the pancreatic cancer gene expression samples from the GCM are associated with high grade tumorigenesis, poor prognosis, and increased metastatic potential.

3.3.8 Differences between Melanoma, Bladder Cancer, and Large B-cell Lymphoma with Respect to the IL-2 Signaling Pathway and Others

Figure 3.7 displays a strong association between bladder cancer, melanoma, and Large B-cell lymphoma (the cyan area in the figure). Each of the four pathways discussed earlier (TGF-beta receptor, B cell receptor, TNF-alpha, and EGFR1 pathways) is perturbed in at least two of these three cancers. In addition, this association is driven by the shared perturbation of multiple interleukin pathways in these cancers (IL-2, IL-3, IL-4, IL-6 and IL-7 signaling pathways), as demonstrated by the cyan rectangle in Figure 3.8. These pathways share a number of genes such as Stat1, Stat5a, Jak1, Jak3, Shc1, and Fyn. Interleukins such as IL-2 are signalling compounds that control the function and regulation of the immune system. Below, we first discuss the association between melanoma and bladder cancer and then the connection to B-cell lymphoma.

We observe that melanoma and bladder cancer perturb the most pathways (13 and 11, respectively). The genes in these perturbed pathways are overwhelmingly down-regulated in the cancers. For example, we find that the IL-2 signalling pathway is down-regulated in melanoma, as shown in Figure 3.10(a). A recent finding by Critchley-Thome et al. [34] supports this result. They found that of the 25 significantly altered genes in T cells and B cells from melanoma patients (compared to healthy controls), 17 were stimulated by interferon (IFN). T cells in the 66% of patients who did not respond to IFN-treatment exhibited many functional abnormalities including decreased expression of cytokines such as IL-2. The IL-2 signaling pathway is also down-regulated in bladder cancer, as show in Figure 3.10(b). Bacillus Calmette Guerin (BCG) is a commonly used treatment to prevent recurrences of bladder cancer [174]. However, the therapeutic mechanism of BCG is not well understood. Van der Meijden [174] indicated that in superficial bladder cancer, the immune response causes secretion of cytokines like IL-2. Our data suggests a link between melanoma and bladder cancer with respect to IL-2 and other pathways. Since recombinant IL-2 has been used successfully to treat melanoma, positive expression of IL-2 may also be associated with the therapeutic mechanism of BCG in superficial bladder cancer. Interleukins such as IL-2 have been used directly as therapies for bladder cancer [128]. Thus, the association that we find between bladder and melanoma cancers is well represented in the literature.

Recombinant IL-2 is known to be one of the few pharmacological treatment options for melanoma and bladder tumors [5, 31]. Liu et al. also found that IL-2 enhances the therapeutic effects of anti-idiotype antibodies for treating B-Cell Lymphoma [106]. Unlike bladder and melanoma cancers, B-cell lymphoma shows up-regulation of the IL-2, IL-6, and IL-7 pathways (see Figure 3.10(c). Wen et al. [181] indicate that deficiencies in PLCγ2 impedes

synthesis of the B-cell receptor and consequently increases activation of the IL-7 pathway. Our results suggest that the therapeutic action of IL-2 against bladder and melanoma tumors occurs by a common mechanism but that a different mechanism is responsible for the effect of IL-2 on large B-cell lymphomas.

3.4 Summary

Our results indicate that integrating differential gene expression with the interaction structure in a pathway is a powerful approach for detecting links between a cancer and the pathways perturbed in it. The use of Stouffer's z-score to combine multiple p-values provides an important advantage over methods that consider pathway membership alone: in many perturbed pathways, we noticed that the receptor protein at the head of the pathway was very slightly differentially expressed, often not to a statistically significant extent whereas many genes with products downstream of the receptor were differentially expressed. Use of meta analysis to combine p-values enabled detection of the perturbation of the pathway even in such cases. Earlier approaches for pathway analysis [42, 148] have usually studied small numbers of cancers. In contrast, we analyzed 18 cancer types in the GCM data set. Studying a relatively large number of cancers enabled us to carry out pathway and cancer association analyses. By analyzing the activation of a compendium of pathways in multiple cancers, we were able to detect patterns of perturbation specific to cancer types and across cancer types.

Chapter 4

A Boolean Model for Network Legos

4.1 Introduction

In this chapter, we present a top-down computational approach that identifies building blocks of molecular interaction networks by

(i) integrating gene expression measurements for a particular disease state (e.g., leukaemia) or experimental condition (e.g., treatment with growth serum) with molecular interactions to reveal an *active network*, which is the network of interactions active in the cell in that disease state or condition and

(ii) systematically combining active networks computed for different experimental conditions using set-theoretic formulae to reveal *network legos*, which are functional modules of coherently interacting genes and gene products in the wiring diagram. These network legos are potential building blocks of the wiring diagram, since we can express each active network as a composition of network legos.

Given a wiring diagram and the transcriptional measurements for a particular condition, we use the gene expression data to induce edge weights in the wiring diagram. We find dense subgraphs [23] in this weighted graph to compute the active network for that condition.[1] Given the active networks for a number of different conditions, we first represent the active networks in an appropriately-defined binary matrix and compute closed biclusters [1, 195] in the matrix. Each bicluster simultaneously represents a set-theoretic combination of particular active networks and a subgraph of the wiring diagram; we call such a subgraph a "block". We exploit the subset structure between blocks to arrange them in a DAG. When the number of active networks is large, we may compute a very large number of highly-similar blocks. Not all these blocks are likely to be network legos. We assess the statistical significance of each block by simulation and identify those that are maximally significant,

[1]Note that this approach is an alternative formulation of the problem solved in Chapter 3. Since we use the active network algorithm only in the context of computing network legos, we discuss it in this chapter.

i.e., more significant than any descendant or an ancestor in the DAG. We deem these blocks to be network legos.

We develop two measures to assess the quality of the network legos we compute. *Stability* measures to what degree we can recompute the same legos when we remove each active network in turn from the input. *Recoverability* measures to what extent we recoup the original active networks when we combine network legos. These two notions test two different aspects of network lego computation. Considering active networks to be the inputs and network legos to be the outputs, stability measures how much the outputs change when we perturb the inputs by removing one of the inputs at a time. In contrast, recoverability asks whether we can reclaim the inputs by combining the outputs; thus recoverability is a measure of how well the network legos serve as building blocks. To assess the biological content of network legos, we measure the functional enrichment of the genes and interactions that belong to a network lego. For each function, we track its degree of enrichment in the DAG to highlight differences among the network legos. For each network lego, we also ask if any functions are enriched only in that network lego and correlate such functions with the expression patterns of the genes in that network lego.

We demonstrate two ways in which a biologist can use our system. In the first, our system allows the systematic comparison of responses to a small number of different conditions, diseases, or perturbations tested in the same lab. The comparison of three leukaemias (ALL, AML, and MLL) [3] we discuss in Section 4.3.1 is such an application. Using our method, we show that the activation of the Kit receptor pathway is a hallmark of AML but not of ALL and MLL; thus, the activation of this pathway distinguishes AML from the other two leukaemias. In the second, a biologist can analyze a specific condition of interest in the context of a large compendium of other conditions, compute the building blocks of the networks activated in these conditions, and ask how the building blocks compose the active network for the specific condition of interest to the biologist. In Section 4.3.2, we apply our approach to a collection of 178 arrays measuring the gene expression responses of HeLa cells and primary human lung fibroblasts to 13 distinct stresses including cell cycle arrest, heat shock, endoplasmic reticulum stress, oxidative stress, and crowding [120]. Our method computes 143 network legos. We carefully examine the compositions of these network legos to demonstrate that they are true building blocks of the active networks for these 13 stresses. We use leave-one-out validation to prove that our algorithm to construct network legos is stable: when we remove each active network and recompute network legos, we are able to recompute most network legos at least 95% fidelity. We also demonstrate that we can recover active networks with almost perfect accuracy by composing network legos. Further analysis of the network legos reveals that the active networks corresponding to cell cycle arrest contain interactions that are quite distinct from the network of interactions activated by the other stresses. When we remove the two cell cycle arrest data sets, we compute only 15 network legos. Of the 11 remaining active networks, we recover five with complete accuracy and one with 99.9% accuracy. We recover the other five active networks with accuracies ranging from 71% to 92%. Functional enrichment analysis of these network legos shows that the only lego enriched in genes controlling and participating in the cell cycle is one that distinguishes the reaction to endoplasmic reticulum stress of fibroblasts from the other stresses. Taken together, these statistics indicate that the network legos we detect

are indeed building blocks of the networks activated in response to the stresses studied by Murray et al. [120] and that the network legos yield biologically-useful insights into the similarities and differences between the two cell types.

The success of our approach stems from a number of factors. First, unlike other approaches that simultaneously integrate multiple gene expression data sets in the context of the network scaffold, we compute individual active networks for each data set and associate the active network with the corresponding disease or perturbation. This approach allows us to explicitly compare and contrast different conditions. Second, we treat interactions (rather than genes or proteins) as the elementary objects of our analysis. Therefore, different network legos may share genes, allowing for the situation when a gene participates in multiple biological processes and is activated differently in these processes. Finally, we develop a simple but effective method to assess the statistical significance of a network lego and to recursively weed out sub-networks that masquerade as building blocks but contain true network legos. Taken together, network legos and the accompanying set-theoretic formulae provide a dynamic and multi-dimensional view of cell circuitry obtained by integrating molecular interaction networks, gene expression data, and descriptions of experimental conditions.

4.2 Algorithms

We describe the main computational ingredients of our approach in stages. We first define useful terminology. Next, we present our method to integrate a cellular wiring diagram with the gene expression data for a single condition to compute the active network for that condition. Third, we describe how we combine active networks for different conditions to form blocks. Fourth, we discuss how we compute the statistical significance of blocks, arrange them in a DAG, and exploit the DAG to identify network legos, which are the most statistically-significant blocks in the DAG. Finally, we present our methods to measure the stability of network legos and assess how well we can recover active networks from the network legos.

4.2.1 Definitions

We denote the wiring diagram of molecular interactions for an organism by W; each node of W is a gene (or gene product) and each edge represents an interaction. Let G be the set of genes in W. The gene expression data set for a condition c consists of a set of samples S_c, each with an expression value for each gene in G; we denote by g_c the vector of values for gene g in the condition c. Our method takes as input the wiring diagram W for an organism and a compendium of gene expression data sets, each for a different condition.

Active networks. Given a gene expression data set for a condition c, we say that a gene *responds in* c if the expression values of the gene in vary by more than an input threshold. Let g and h be two genes that respond in c and let $e = (g, h)$ be an interaction in W. We say

that e is *active in* c if if g_c and h_c are correlated to a statistically-significant extent. Let the weight w_e of interaction e denote this degree of correlation. We define the *active network* A_c *in* c to be the subgraph of W with maximum density, where we define the density $d(G)$ of a graph $G = (V, E)$ as the total weight of its edges divided by the number of nodes in it, i.e.,

$$d(G) = \frac{\sum_{e \in E} w_e}{|V|}.$$

We describe the details of how we detect responding genes, active interactions, and active networks in Section 4.2.2.

Blocks. Let $\mathcal{A}$ denote a set of active networks, one for each of the conditions in the input compendium. We define a *block* to be a triple $(G, \mathcal{P}, \mathcal{N})$, where G is a subgraph of W, $\mathcal{P}$ and $\mathcal{N}$ are disjoint subsets of $\mathcal{A}$, and $\mathcal{P} \neq \emptyset$ such that

$$G = \Big(\bigcap_{P \in \mathcal{P}} P \Big) \bigcap \Big(\bigcap_{N \in \mathcal{N}} (\mathcal{W} - N) \Big) = \Big(\bigcap_{P \in \mathcal{P}} P \Big) - \Big(\bigcup_{N \in \mathcal{N}} N \Big),$$

where "$\cap$," "$-$," and "$\cup$" respectively denote the intersection, difference, and union of the edge sets of two graphs, and

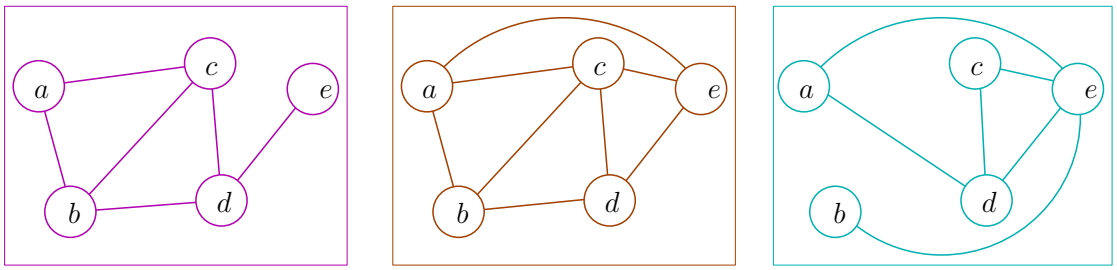
1. $\mathcal{P}$ is maximal, i.e., there is no active network $P \in \mathcal{A} - \mathcal{P}$ such that $G \subseteq P$, and
2. $\mathcal{N}$ is maximal, i.e., there is no active network $N \in \mathcal{A} - \mathcal{N}$ such that $G \cap N = \emptyset$.

In other words, we can form G by taking the intersection of all the active networks in $\mathcal{P}$ and removing any edge that appears in any of the active networks in $\mathcal{N}$. We require that $\mathcal{P}$ contain at least one active network so that G is not formed solely by the intersection of the networks in $\mathcal{N}$; such a block is unlikely to be biologically interesting. We also require that $\mathcal{P}$ and $\mathcal{N}$ be disjoint so that G is not the empty graph. Requiring $\mathcal{P}$ and $\mathcal{N}$ to be maximal ensures that we include all the relevant active networks in the construction of G. These criteria imply that it is enough to specify $\mathcal{P}$ and $\mathcal{N}$ to compute G uniquely; we include G in the notation for a block for convenience and drop $\mathcal{P}$ and $\mathcal{N}$ when they are understood from the context. We refer to $\big(\bigcap_{P \in \mathcal{P}} P\big) \bigcap \big(\bigcap_{N \in \mathcal{N}} (\mathcal{W} - N)\big)$ as the *formula* for the block. Figure 4.1(a) displays three toy active networks and Figure 4.1(b) shows examples of three blocks formed from these active networks.

Network legos. We now describe how we identify network legos in a set $\mathcal{B}$ of blocks. We note that n active networks can compose at most $3^n - 2^n$ blocks. Given two distinct blocks $(G_1, \mathcal{P}_1, \mathcal{N}_1)$ and $(G_2, \mathcal{P}_2, \mathcal{N}_2)$ in $\mathcal{B}$, we say that $G_1 \prec G_2$[2] if

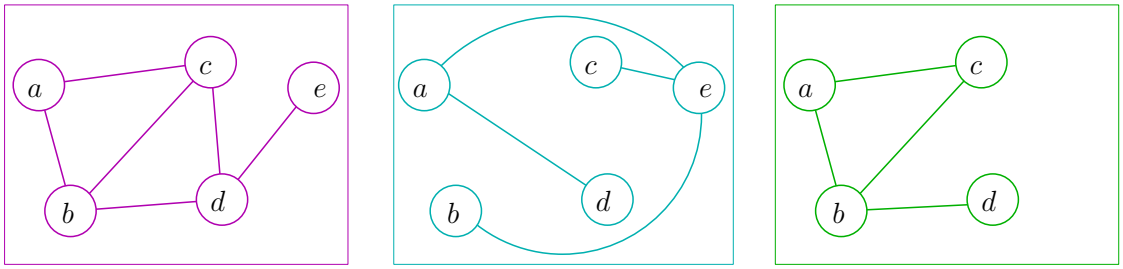
(i) $\mathcal{P}_1 \subseteq \mathcal{P}_2$ and $\mathcal{N}_1 \subseteq \mathcal{N}_2$ or

[2]In the rest of this paragraph, we abuse notation and use G to refer to the block $(G, \mathcal{P}, \mathcal{N})$.



A_1 A_2 A_3

(a) Three active networks A_1, A_2, and A_3.



$(G_1, \mathcal{P}_1, \mathcal{N}_1)$ $(G_2, \mathcal{P}_2, \mathcal{N}_2)$ $(G_3, \mathcal{P}_3, \mathcal{N}_3)$

(b) Examples of three blocks formed by combining the active networks A_1, A_2, and A_3.

Figure 4.1: Examples of active networks and blocks. In the block $(G_1, \mathcal{P}_1, \mathcal{N}_1)$, $\mathcal{P}_1 = \{A_1, A_2\}$, $\mathcal{N}_1 = \emptyset$, and $G_1 = A_1 \cap A_2$; in the block $(G_2, \mathcal{P}_2, \mathcal{N}_2)$, $\mathcal{P}_2 = \{A_3\}$, $\mathcal{N}_2 = \{A_1\}$, and $G_2 = A_3 - A_1$; and in the block $(G_3, \mathcal{P}_3, \mathcal{N}_3)$, $\mathcal{P} = \{A_1, A_2\}$, $\mathcal{N} = \{A_3\}$, and $G_3 = A_1 \cap A_2 - A_3$. Therefore, we have the following relations: $G_1 < G_3$ and $G_2 < G_3$.

(ii) $\mathcal{P}_1 \subseteq \mathcal{N}_2$ and $\mathcal{N}_1 \subseteq \mathcal{P}_2$.

Further, we say that $G_1 < G_2$ if there is no block $G_3 \in \mathcal{B}$ such that $G_1 \prec G_3 \prec G_2$. The operators $<$ and $\prec$ represent partial orders between blocks, with $\prec$ being the transitive closure of $<$. Given a set $\mathcal{B}$ of blocks, let $\mathcal{D}_\mathcal{B}$ denote the directed acyclic graph representing the partial order $<$: each node in $\mathcal{D}_\mathcal{B}$ is a block in $\mathcal{B}$ and an edge connects two blocks related by $<$. For a block G, let $\sigma_G \in [0, 1]$ denote the statistical significance of G. We describe a method to compute this value in Section 4.2.4. We define a *network lego* to be a block $(G, \mathcal{P}, \mathcal{N}) \in \mathcal{B}$ such that $\sigma_G < \sigma_H$, for every $H \in \mathcal{B}$ where $G \prec H$ or $H \prec G$. In other words, $(G, \mathcal{P}, \mathcal{N})$ is a network lego if it is more statistically significant that blocks formed by combining any subset of $\mathcal{P}$ and $\mathcal{N}$ or by combining any superset of $\mathcal{P}$ and $\mathcal{N}$. In this sense, we claim that G is a building block of the active networks in $\mathcal{A}$.

4.2.2 Computing the active network for a single condition

Given a gene expression dataset for a condition c, we compute its active network A_c using the following steps:

1. **Filter genes.** We use a variational filter to remove all genes whose expression profiles

have a small dynamic range from W. Specifically, we log-transform and zero-centre each gene's expression values. We discard a gene and all its interactions in the wiring diagram W if all the transformed expression values of the gene lie between -1 and 1 [144]. We deem the remaining genes to have responded in the condition.

2. **Filter interactions.** To each interaction $e = (g, h)$ remaining in W, we assign a weight equal to the absolute value of the Pearson's correlation coefficient of g_c and h_c, reasoning that this weight indicates how "active" the interaction is in the experimental condition. We discard edges whose weights are not statistically significant by using the following procedure: (i) We construct 50 random versions of the gene expression data set by permuting each gene's expression values independently. (ii) For each random data set, we compute a histogram of the absolute value of the Pearson's correlation coefficient of the expression profiles of all pairs of genes. (iii) We average these 50 histograms and keep only those interactions in W whose edge weights are significant at the 0.01 level. Let $\mathcal{W}_c$ be the resulting weighted interaction network.

3. **Compute A_c.** We compute A_c using a greedy algorithm [23]. We define the *weight* w_v of a vertex $v \in \mathcal{W}_c$ to be the total weight of the edges incident on v. We repeatedly delete the node of smallest weight in $\mathcal{W}_c$. After each deletion, we update the weights of the neighbors (before deletion) of this node and record the density of the remaining network. We set A_c to be most dense of all the networks so generated. Algorithm 2 contains pseudo-code for this step.

Algorithm 2 Computing the active network A_c for a condition c. The input to the algorithm is $\mathcal{W}_c$, the edge-weighted subgraph of $\mathcal{W}$ containing responding genes and perturbed edges.

```
W_max ← W_c
while W_c is not empty do
  x ← arg min_{v∈W_c} w_v
  W_c ← W_c \ {x}
  if d(W_c) > d(W_max) then
    W_max ← W_c
  end if
end while
return W_max
```

Remarks:

1. It is possible to find the subgraph of largest density using linear programming or parametric network flows [23]. The greedy algorithm described above finds a subgraph that is at least half as dense as the most dense subgraph.

2. In practice, we embed the greedy algorithm in the following heuristic: we repeatedly apply this approximation algorithm, remove the edges of the subgraph it computes, and re-invoke the algorithm on the remaining graph until the density of the remaining

graph is less than the density of $\mathcal{W}_c$. We deem the union of the computed dense subgraphs to be the active network A_c.

3. The rationale for computing the most dense subgraph in $\mathcal{W}_c$ is to mitigate the effect of isolated interactions in $\mathcal{W}_c$, since a dense subgraph corresponds to a sub-network of concerted activity in $\mathcal{W}_c$. Many other natural definitions of the weight of a subgraph are computationally intractable to optimize. For instance, if the weight of a subgraph is the total weight of its edges divided by the number of possible edges in the subgraph, it is well-known that the problem of computing the subgraph of highest weight is NP-complete [86].

4.2.3 Computing the set of blocks in a set of active networks

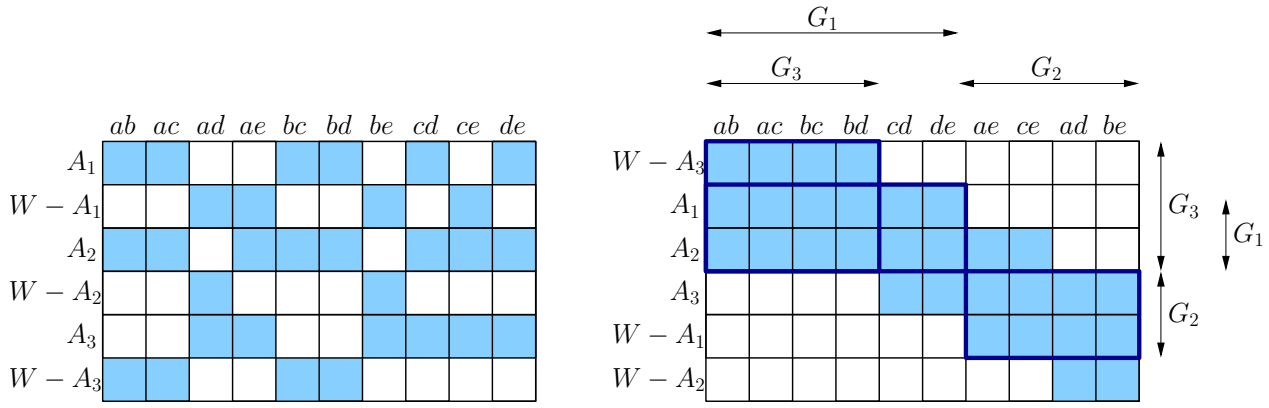
Given a set $\mathcal{A}$ of active networks, we reduce the problem of computing blocks defined by the active networks in $\mathcal{A}$ to the problem of computing closed biclusters in a binary matrix [1, 195]. Consider a binary matrix M where each column corresponds to an interaction in W. The matrix M contains two rows for each active network $A \in \mathcal{A}$: the *positive* row corresponds to the interactions in A and the *negative* row to the interactions in $W - A$. In the positive row corresponding to A, we set a cell to be one if and only if the corresponding interaction belongs to A; this cell is zero in the negative row for A. Thus, M is a qualitative representation of which interactions are present in each active network and which are present in its complement.

In a binary matrix such as M, a *bicluster* (R, C) is a subset R of rows and a subset C of columns such that the sub-matrix spanned by these rows and columns only contains ones. A *closed* bicluster is a bicluster with the property that each row (respectively, column) not in the bicluster contains a zero in at least one column (respectively, row) in the bicluster. Therefore, it is not possible to add a row or a column to a closed bicluster without introducing a zero into the corresponding sub-matrix. We can partition R into two subsets R_P and R_N where R_P (respectively, R_N) consists of all the positive (respectively, negative) rows in R. There is a natural one-to-one mapping from a closed bicluster (R, C) to a block $(G, \mathcal{P}, \mathcal{N})$:

1. G is the subgraph of W induced by the interactions corresponding to the columns in C;
2. $\mathcal{P}$ is the set of active networks corresponding to the rows in R_P; and
3. $\mathcal{N}$ is the set of active networks corresponding to the rows in R_N.

Requiring a bicluster to be closed is equivalent to ensuring that $\mathcal{P}$ and $\mathcal{N}$ are maximal and that C contains all the interactions in G. Figure 4.2(a) displays the matrix corresponding to the three active networks in the example in Figure 4.2(b) displays a layout of all the biclusters in this matrix and highlights the three biclusters corresponding to the three blocks.

Before describing our algorithm, we define one more concept. Given a set R of rows in M, we define the *closure* of R to be the closed bicluster (R^*, C^*), where C^* is the set of columns



(a) The binary matrix M corresponding to active networks. Each shaded square corresponds to a entry whose value is one.

(b) A layout of all closed biclusters in M. We use arrows to delineate the rows and columns that span these three biclusters.

Figure 4.2: The binary matrix and biclusters corresponding to blocks. Here, $\mathcal{W}$ is the complete graph on the five nodes.

that contain ones in all the rows in R and $R^* \supseteq R$ is the set of rows that contain ones in all the columns in C. Given R, we can compute its closure by two scans over M.

We give the pseudo-code for computing network legos in Algorithm 3. To construct closed biclusters and the resulting set $\mathcal{B}$ of blocks, we use a variation of the well-known *Apriori* level-wise algorithm for computing itemsets [1]. In Algorithm 3, we do not distinguish between a closed bicluster and the corresponding block.

Remarks. Since we consider only the positive rows in M in the first step, every closed bicluster we compute contains at least one positive row. In practice, we hash the row sets of the biclusters to avoid reporting a bicluster more than once. The worst-case running time of the algorithm is exponential in the number of rows in M.

4.2.4 Assessing the statistical significance of a block

To measure the statistical significance of a block, we construct an empirical distribution of block sizes. We repeatedly select a subset of rows uniformly at random from the binary matrix M, compute the columns common to these rows, and convert the resulting bicluster into a block. We ensure that the random subset of rows does not contain an active network and its complement, since such a subset will trivially result in an bicluster with zero columns. Given a block $(G, \mathcal{P}, \mathcal{N})$ computed in the real dataset, let m be the number of interactions in G. To estimate the statistical significance σ_G of $(G, \mathcal{P}, \mathcal{N})$, we only consider the distribution formed by random blocks $(H, \mathcal{P}', \mathcal{N}')$ where $|\mathcal{P}| = |\mathcal{P}'|$ and $|\mathcal{N}| = |\mathcal{N}'|$. We set σ_G to be the fraction of such blocks that have at least m interactions. Since the number of interactions in a block will decrease with an increase in $|\mathcal{P}|$ or in $|\mathcal{N}|$, these constraints ensure that we compare G with appropriate random blocks in order to estimate σ_G. We only retain blocks

Algorithm 3 Computing the set of blocks

Compute the closure of each positive row r in M. Let $\mathcal{C}$ be the set of biclusters so computed.
$\mathcal{B} \leftarrow \mathcal{C}$
while $\mathcal{C}$ is non-empty **do**
  $\mathcal{C}' \leftarrow \emptyset$
  for each bicluster (R, C) in $\mathcal{C}$ **do**
    for each row $r \notin R$ **do**
      Compute the closure (R^*, C^*) of $R \cup \{r\}$.
      if $C^* \neq \emptyset$ **then**
        $\mathcal{C}' \leftarrow \mathcal{C}' \cup \{(R^*, C^*)\}$
      end if
    end for
  end for
  $\mathcal{C} \leftarrow \mathcal{C}'$
  $\mathcal{B} \leftarrow \mathcal{B} \cup \mathcal{C}$
end while
Construct the DAG $\mathcal{D}_{\mathcal{B}}$ connecting the blocks in $\mathcal{B}$ as per the partial order $<$.

that are significant at the 0.01 level. We compute the DAG defined by these blocks. We perform two topological traversals of this DAG, one from the roots to the leaves and the other from the leaves to the roots, to identify the maximally-significant blocks. The resulting set of blocks are the network legos we desire to compute. Let $\mathcal{L}$ denote the set of network legos.

4.2.5 Stability and recoverability analysis

Stability. It is clear that the set $\mathcal{L}$ of network legos we compute depend on the active networks in $\mathcal{A}$. To assess this dependence, we modify a method for suggested by [144]. We remove each network $N \in \mathcal{A}$ in turn and recompute network legos from the set $\mathcal{A} - \{N\}$. Let $\mathcal{L}_N$ denote the resulting set of network legos. For each network lego L in $\mathcal{L}$, we compute the most similar network lego L' in $\mathcal{L}_N$ using the set-similarity measure $(|L \cap L'|/|L \cup L'|)$ and store this measure as $s_{L,N}$. Given a similarity threshold t, for each network lego L in $\mathcal{L}$, we compute the fraction of networks in $\mathcal{A}$ such that $s_{L,N} \geq t$. The higher this fraction is, the more resilient L is to perturbations in the input.

Recoverability. If the network legos in $\mathcal{L}$ are true building blocks of the active networks in $\mathcal{A}$ that they spring from, it should be possible to recover each active network in $\mathcal{A}$ from the network legos in $\mathcal{L}$. For each active network A, we define

$$\mathcal{L}_A = \big\{(G, \mathcal{P}, \mathcal{N}) \in \mathcal{L} | A \in \mathcal{P}\big\},$$

to be the set of network legos in $\mathcal{L}$ where A does not appear negated in the network lego. We compute the union of the network legos in $\mathcal{L}_A$ and compute what fraction of A's edge

set appears in the union. The larger this fraction is, the more "recoverable" A is from the computed network legos.

4.2.6 Properties of Blocks

Intuitively, network legos should be subgraphs of the active networks they compose. In this section, we state and prove a few observations about blocks that formalize this notion.

Lemma 4.2.1. *Let $(G, \mathcal{P}, \mathcal{N})$ be a block. Then the following conditions hold:*

(a) For every active network $A \in \mathcal{P}$, G is a subgraph of A.

(b) For every active network $B \in \mathcal{N}$, $G \cap B = \emptyset$.

(c) For every active network $C \notin \mathcal{P} \cup \mathcal{N}$, G contains at least one interaction not in C.

Proof. By the definition of a block, $G = \left(\bigcap_{P \in \mathcal{P}} P\right) \bigcap \left(\bigcap_{N \in \mathcal{N}} (\mathcal{W} - N)\right) = \left(\bigcap_{P \in \mathcal{P}} P\right) - \left(\bigcup_{N \in \mathcal{N}} N\right)$. Here, set operations are defined on the edge sets of active networks.

(a) By the first form of the definition of G, the edges in G are a subset of $\bigcap_{P \in \mathcal{P}} P$. Since $A \in \mathcal{P}$, we see that the edges in G are a subset of the edges in A, implying that G is a subgraph of A.

(b) By the second form of the definition of G, we see that no edge in the set $\bigcup_{N \in \mathcal{N}} N$ can appear in G. Therefore, $G \cap B$ is empty.

(c) By the definition of a block, $\mathcal{P}$ and $\mathcal{N}$ are maximal. Therefore, since $C \notin \mathcal{P}$, we see that $G \not\subseteq C$ and since $C \notin \mathcal{P}$, we see that $G \cap C \neq \emptyset$. Therefore, G must contain at least one interaction that is not in C.

□

A simple corollary of this lemma is that G has fewer interactions than the active network with the smallest number of interactions in $\mathcal{P}$.

The following two lemmas prove there is a one-to-one correspondence between closed biclusters and blocks, thus establishing the the correctness of our algorithm for computing blocks. We first set up some notation to state the lemmas. We assume that W has m edges and an arbitrary but fixed ordering of these edges. Given an unweighted subgraph G of W, let $\mathbf{G}$ denote a binary vector of length m defined as follows: (i) the dimensions of $\mathbf{G}$ are in a one-to-one correspondence with the ordering of the edges of W and (ii) the value of an element of $\mathbf{G}$ is equal to 1 if the corresponding interaction is in G and 0 otherwise. Let $\mathcal{A}$ denote a set of n active networks, each of which is a subgraph of W. Consider the $2n \times m$ matrix M whose rows are the vectors $\mathbf{A}$ and $\mathbf{W} - \mathbf{A}$, for every active network $A \in \mathcal{A}$.

Lemma 4.2.2. *Consider a block $(G, \mathcal{P}, \mathcal{N})$. Then (R, C) is a closed bicluster in M where R and C are defined as follows:*

(a) $R = \left(\bigcup_{A \in \mathcal{P}} \mathbf{A} \right) \bigcup \left(\bigcup_{B \in \mathcal{N}} (\mathcal{W} - \mathbf{B}) \right)$.

(b) A column of M is an element of C iff the corresponding interaction is in G.

Lemma 4.2.3. *Consider a closed bicluster (R, C) in M such that R contains at least one positive row. Then the triple $(G, \mathcal{P}, \mathcal{N})$ is a block where $G, \mathcal{P}$, and $\mathcal{N}$ are defined as follows:*

(a) An edge of W is in G iff the corresponding column is an element of C.

(b) An active network A is an element $\mathcal{P}$ iff $\mathbf{A} \in R$.

(c) An active network B is an element $\mathcal{N}$ iff $\mathcal{W} - \mathbf{B} \in R$.

4.3 Results

We applied the algorithm described in the previous section to human data sets. We obtained a network of 31108 molecular interactions between 9243 human gene products by integrating the interactions in the IDSERVE database [132], the results of large scale yeast two-hybrid experiments [138, 159], and 20 immune and cancer signalling pathways in the Netpath database (http://www.netpath.org). The IDSERVE database includes human curated interactions from BIND [7], HPRD [129], and Reactome [81], interactions predicted based on co-citations in article abstracts, and interactions that transferred from lower eukaryotes based on sequence similarity [101]. We derived functional annotations for the genes in our network from the Gene Ontology (GO) [4] and from MSigDB [162]. In addition, we annotated each Netpath interaction in our network with the name of the pathway it belonged to. We used these annotations to compute the functional enrichment of the nodes and edges in the network legos using the hypergeometric distribution. We controlled the false discovery rate using the method proposed by Benjamini and Hochberg [15].

We present two analyses below. In the first, we compare and contrast three types of leukaemias. In the second, we compute the network legos for a set of environmental stresses imparted to two human cell types.

4.3.1 ALL, AML, and MLL

Armstrong et al. [3] demonstrated that lymphoblastic leukaemias involving translocations in the *MLL* gene constitute a disease different from conventional acute lymphoblastic (ALL) and acute myelogenous leukaemia (AML). The authors based their analysis on the comparison of gene expression profiles from individuals diagnosed with ALL, AML, and MLL. We reasoned that the networks of molecular interactions activated in these diseases may

also show distinct differences. First, we computed active networks for each leukaemia, as described in Section 4.2.2. Next, we computed all 19 ($3^3 - 2^3$) blocks induced by these three active networks, using the method presented in Section 4.2.3. Since the number of blocks is small, we did not compute their statistical significance. Instead, we treated every block as a network lego. We connected the network legos in the directed acyclic graph (DAG) displayed in Figure 4.3. In this DAG, each node represents a single network lego, e.g., the leftmost node on the top row represents the MLL active network while the leftmost node in the middle row represents the interactions activated in AML but not in MLL (the formula AML − MLL). A solid blue edge directed from a child to a parent indicates that the formula for the child (e. g., MLL) appears as a part of the formula for the parent (e.g., MLL − AML), while a dashed green edge indicates that the child's formula (e.g., MLL) appears negated in the parent's formula (e.g., AML − MLL).

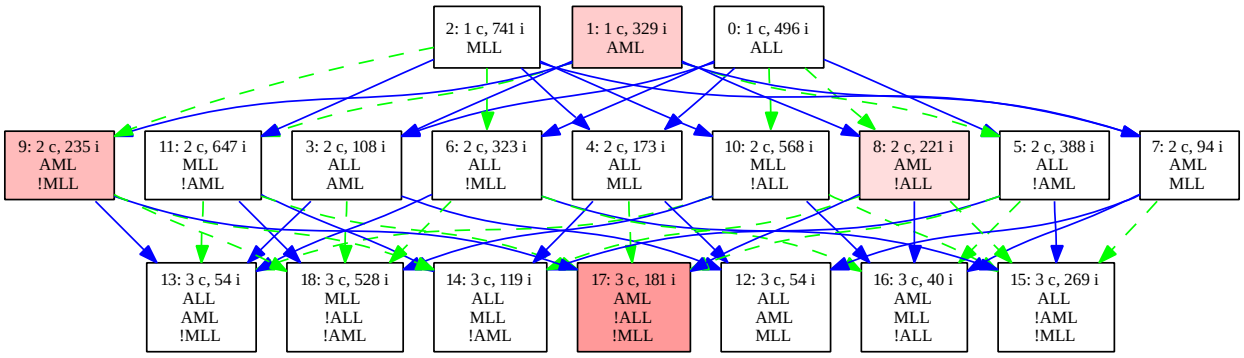


Figure 4.3: The lattice connecting combinations of ALL, AML, and MLL active networks. Each node contains an index, the number of 'c'onditions, the number of 'i'nteractions and the active networks participating in the formula. We use '!' to indicating set difference. Colors indicate the enrichment of the interactions in the KIT pathway in the computed combinations. Darker colors denote more significant enrichment values.

To assess the biological content of the results and to illustrate one type of analysis our approach facilitates, we computed Netpath pathways enriched in the interactions in the networks corresponding to the 19 formulae. Figure 4.3 demonstrates that the interactions in the KIT pathway are differentially enriched in the 19 networks. The darker the color of a node, the more statistically significant is the enrichment of this pathway in the corresponding network. We first note that the only formulae enriched in this pathway are the ones that involve AML (and not the complement of AML). The statistical significance is the lowest (FDR-corrected p-value 3.5×10^{-7}) for the formula AML − ALL − MLL. We interpret these statistics to imply that this pathway is activated only in AML and not in ALL or in MLL. Evidence in the literature supports this conclusion. The c-KIT receptor is activated in almost all subtypes of AML [137, 143]. Similarly, [142] report that "mutations in codon D816 of the KIT gene represent a recurrent genetic alteration in AML." By studying 1,937 patients diagnosed with acute leukaemia, [14] found that c-kit was expressed in 67% of AML cases but only in 4% of ALL cases, and that most of these ALL cases exhibited myeloid markers. We note that gain-of-function mutations in c-Kit have been observed in many human cancers [32]. Our analysis only suggests that in the context of ALL, AML, and MLL,

the KIT pathway may be activated only in AML.

4.3.2 Human Stresses

We computed network legos by applying our methods to the human interaction network and the gene expression responses of HeLa cells and primary human lung fibroblasts to heat shock, endoplasmic reticulum stress, oxidative stress, and crowding [120]. The dataset we analysed includes transcriptional measurements obtained by Whitfield et al. [182] for studying cell cycle arrest by using a double thymidine block or with a thymidine-nocodazole block. Overall, the dataset contains 13 distinct stresses over the two cell types. The authors note that each type of stress resulted in a distinct response and that there was no general stress response unlike in the case of *S. cerevisiae* [55]. Therefore, this dataset poses a challenge to our system. Can we find network legos that combine active networks for multiple stresses?

Structural Analysis of network legos The number of genes in the 13 active networks we computed ranged from 165 (for crowding of WI38 cells) to 1148 (for the thymidine-nocodazole block) with an average of 684 genes per active network. The number of interactions ranged from 257 to 3667 with an average of 1874 interactions per active network. Theoretically, we can compute 1586131 ($3^{13} - 2^{13}$) blocks involving 13 distinct active networks. Our method computed 444201 blocks, indicating that the remaining combinations of active networks are not closed or yield blocks without any interactions. We computed a null distribution of block sizes using a million random samples. Of the 444201 blocks, 12386 blocks were statistically significant at the 0.01 level. We identified 143 network legos in the DAG induced by the relation $<$ on these blocks. We observed that all but one of the 143 network legos involved at least six distinct active networks, indicating that these network legos are not the result of combining a small number of active networks. The following table displays the distribution of the number of legos involving k conditions, where $5 \leq k \leq 12$. Interestingly, no network lego involved all 13 active networks.

#conditions	5	6	7	8	9	10	11	12
#legos	1	6	10	36	34	20	28	8

In light of the statement by Murray et al. [120] that each type of stress resulted in a distinct response, it is important to ask whether most of our network legos primarily involve complemented active networks. Over all network legos $(G, \mathcal{P}, \mathcal{N})$, we counted the total size of the "$\mathcal{P}$ sets" and the "$\mathcal{N}$ sets." The ratio of these numbers was 2:3, indicating that a large fraction of the network legos represented features common to multiple stresses. The active networks that appeared most often in the positive form were the two treatments that resulted in cell cycle arrest. Each participated in as many as 119 network legos. In most of these network legos, almost all the other active networks appeared in complemented form. The complements of the cell cycle arrest active networks did not participate in any network

legos. This observation indicates that the interactions activated by cell cycle arrest are quite distinct from the network of interactions activated by the other stresses.

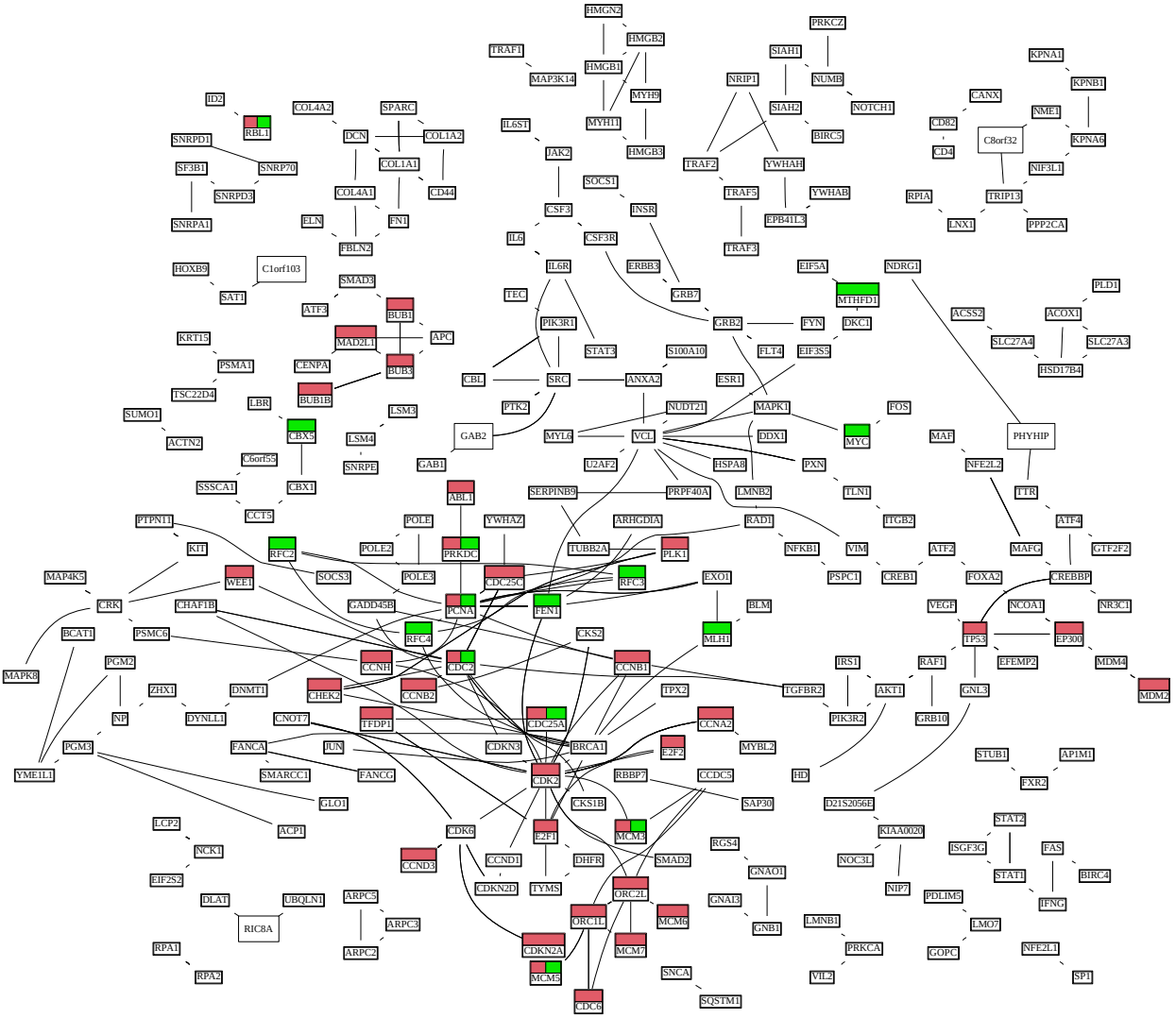
We obtained very good stability and recovery results. Upon the removal of each active network, we were able to recompute each network lego with at least 95% fidelity. We were also able to recover 11 active networks with 100% accuracy by composing network legos. The two active networks we could not recover completely were the double thymidine network (97% recovery) and the thymidine-nocodazole network (86% recovery). When we tested the recoverability of active networks using the blocks at the roots of the DAG connecting statistically-significant blocks, the recovery for these two active networks dropped to 85% and 75% respectively. This result underscores the fact that identifying network legos as those that are maximally statistically-significant in the DAG of blocks is a useful concept.

Since the cell-cycle treatments resulted in active networks that were quite distinct from those for the other stresses, we repeated the analysis after removing the double thymidine and thymidine-nocodazole active networks. The 11 remaining active networks yielded only 77117 blocks (out of the 175099 possible). Of these, 1629 blocks were statistically significant. These blocks yielded 15 network legos. This much smaller set of network legos suggests that a number of the 143 network legos in the complete analysis were needed to capture unique aspects of the cell cycle active networks. Each network lego involved at least seven active networks. No network lego involved all 11 stresses. The ratio of total size of the "$\mathcal{P}$ sets" and the "$\mathcal{N}$ sets." over the 15 network legos was 1:2. Of the 11 active networks, we recovered five with complete accuracy and one with 99.9% accuracy. We recovered the remaining with accuracies ranging from 71% to 92%. Taken together, these statistics indicate that the network legos we detect are indeed building blocks of the networks activated in response to the stresses studied by Murray et al. [120].

Biological analysis of network legos We focus on one of the 15 network legos we computed in our analysis without the cell cycle arrest treatments. This *ER stress* network lego corresponds to the formula

$$\Big(\text{Fibroblast DTT} \cap \text{Fibroblast Menadione}\Big) - \Big(\text{HeLa Crowding} \cup \text{HeLa Heat} \cup \text{HeLa Menadione} \cup \text{Fibroblast Crowding} \cup \text{Fibroblast Heat}\Big)$$

The only two stresses that appear in positive form in this formula are the treatment of fibroblasts with DTT and menadione. These chemicals induce endoplasmic reticulum (ER) stress. This network lego is the only one significantly enriched in functions related to the cell cycle (e.g., p-value 3×10^{-30} for the KEGG [84] "Cell cycle" pathway and 2.3×10^{-24} for the REACTOME [81] pathway describing the transition from G1 to S) and in targets of the E2F1 transcription factor [162] (p-value 8×10^{-13}), which is a known regulator of cell cycle progression. E2F1 arrests cells in the G1 phase by forming a transcriptional repressor complex with the Retinoblastoma protein [196]. Figure 4.4 displays a layout of



Colour	Function Name	p-value
	Cell cycle pathway	3×10^{-30}
	E2F1 targets	8×10^{-13}

Figure 4.4: A layout of the interactions in the *ER stress* network lego.

the *ER stress* network lego, specifically highlighting the genes annotated with the KEGG "Cell cycle" pathway and as targets of E2F1. Figure 4.5 shows a heat map of the expression profiles of the genes annotated with these two functions in the seven conditions in the network lego. Examination of the gene expression patterns in Figure 4.5 reveals that about four–six hours after treatment with DTT or menadione, fibroblasts shut down the cell cycle far more aggressively than fibroblasts or HeLa cells do in response to other treatments. Thus, this network lego automatically identifies a unique characteristic of fibroblast response to ER stress in the context of the other stresses in the compendium.

4.4 Discussion

We have presented a novel approach for combining gene expression data sets with a multi-modal wiring diagram to compute network legos, which are context-sensitive building blocks of the wiring diagram. This combination provides a dynamic view of the interactions that are activated in the wiring diagram under different conditions. We represent similarities and differences between the network of interactions activated in response to different cell states both as a set theoretic formula involving cell states and as a network lego, a functional module of co-expressed molecular interactions. A novel contribution of our work is the DAG that relates all cell states (and the active networks corresponding to the cell states). This DAG provides a high-level abstract view of the similarities and differences between cell states.

The literature on network motifs [115, 116, 154, 193] provides an alternative perspective on finding the building blocks of cellular circuits. [197] constructed an integrated *S. cerevisiae* interaction network, identified three- and four-node network motifs, and organized these motifs into network themes and further into thematic maps. It would be interesting to study whether the top-down approach presented here to construct network legos yields network modules that are similar in structure and organization to those computed by the bottom-up approach used by Zhang et al..

Since we explicitly compute all closed biclusters in $\mathcal{B}$, the worst-case running time of our algorithm may be exponential in the number of active networks. An interesting avenue of future research is to develop a method that avoids this exorbitant running time, perhaps by computing network legos that directly optimize for stability and/or recoverability. Another important open question is that of developing incremental algorithm that can efficiently recompute the network legos upon the addition or deletion of an active network.

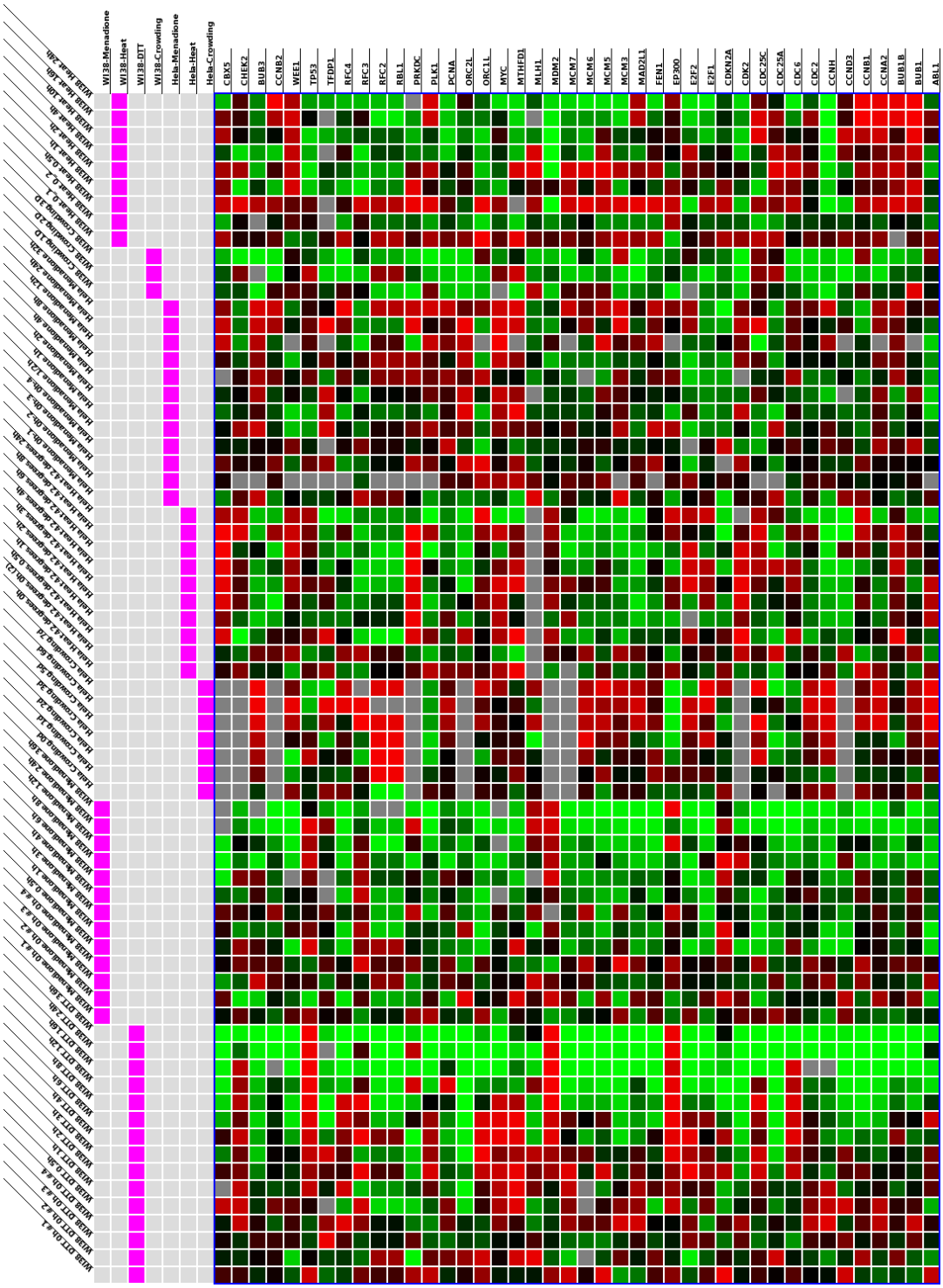


Figure 4.5: A heat map of the gene expression measurements in the seven conditions participating in the *ER stress* network lego. Columns correspond to genes annotated with at least one of the functions mentioned in the text. Rows correspond to samples. Each vertical pink line delineates a set of samples belonging to one of the stresses in the network lego. The two lowermost pink lines correspond to fibroblast response to treatment with menadione and with DTT, which are the two stresses that appear in positive form in the *ER stress* network lego.

Chapter 5

An Additive Weighted Model for Network Legos

5.1 Introduction

In Chapter 4, we proposed a Boolean model for representing network legos. Although the approach yielded interesting biological insights, it has three limitations:

1. There is no mathematical guarantee on how well active networks can be recovered from computed network legos.
2. The Boolean formulation only allows for a qualitative consideration of active networks and cannot take the degree of activation or perturbation of an interaction into account.
3. Exhaustive enumeration of blocks can take time exponential in the number of active networks in the worst case. Thus, this method does not scale up to large numbers of active networks.

In this chapter, we present an approach that addresses these limitations. Our new approach has the following features:

(i) We can interpret a network lego in two ways:
   (a) as a weighted subgraph of coherently interacting genes and gene products in the wiring diagram and
   (b) as a compact linear combination of the active networks.

(ii) We can represent each active network as an additive combination of network legos.

(iii) We can frame the question of reconstructing (weighted) network legos from (weighted) active networks directly in terms of optimizing the recoverability of the active networks/

In this chapter, we model the network of interactions activated by the cell in response to the stimulus as an additive combination of such a subset of modules. The primary contribution of this chapter is the formulation of the linear model for network legos in a cellular wiring diagram. Given an integer $k > 0$, we formulate the problem as one of explicitly computing k network legos in a wiring diagram and additive combinations of the network legos such that the active networks can be represented as well as possible by the combinations. We show that algorithms for non-negative matrix approximation are natural candidates for solving this problem. We evaluate the well-known approach of Lee and Seung [97] in this chapter.

5.2 Algorithms

5.2.1 Definitions

Let W denote the network of known molecular interactions in the cell with m edges. We assume that W is an unweighted, undirected graph in which each node is a gene and each edges connects two genes.

Active networks. Given the gene expression data set for a condition c, we define the *active network A_c in c* to be the network of interactions that are perturbed in the cell in the condition c. We represent A_c as a weighted undirected subgraph of W, where each edge has a positive weight that represents the extent to which it is perturbed. We describe the details of how we compute active networks below.

We can represent the active network A_c by a vector in which each dimension corresponds to a unique edge of W. The coordinate of the vector in each dimension is the weight of the corresponding edge in A_c. Edges that do not belong to A_c have a weight of 0. Consequently, given a set of n response networks, one for each in a compendium of conditions, we can represent all these networks in an $n \times m$ matrix $\mathbf{M}$, in which the ith row contains the ith response network. Each column of $\mathbf{M}$ corresponds to one of the interactions in W so that $\mathbf{M}_{il}$ represents the weight of interaction l in active network i.

Network legos. We now formulate our model for network legos. Each network lego is a weighted subgraph of W. Thus, we can represent a set of k network legos in a $k \times m$ matrix $\mathbf{L}$. We assume that k is a parameter that is input by the user or can be exhaustively searched over. Recall that we hypothesize that any active network can be represented as an additive combination of a subset of network legos. It is now natural to represent the relationship between active networks, network legos, and the weights in these combinations as

$$\mathbf{M}_{n\times m} \approx \mathbf{W}_{n\times k}\mathbf{L}_{k\times m}$$

Here, each column of $\mathbf{L}$ corresponds to an edge of W and each row stores which interactions appear in a specific network lego. Each column of $\mathbf{W}$ corresponds to a network lego and

each row to a active network: the value $\mathbf{W}_{ij}$ represents the contribution of network lego j to active network i. All three matrices contain non-negative entries: $\mathbf{M}$ has this property by construction (since we allow edges in active networks only to have positive weights) while $\mathbf{W}$ and $\mathbf{L}$ obtain this property from our model. We define the *relative error* of a factorization of $\mathbf{M}$ into positive matrices $\mathbf{W}$ and $\mathbf{L}$ as

$$\frac{\sum_{ij}(\mathbf{M}_{ij} - (\mathbf{WL})_{ij})^2}{\sum_{ij}\mathbf{M}_{ij}^2}.$$

Since $\mathbf{WL}$ may only approximate $\mathbf{M}$, our goal is to compute $\mathbf{W}$ and $\mathbf{L}$ so that this approximation is as good as possible. Accordingly, we desire to compute values for these matrices that minimize

$$\sum_{ij}\left(\mathbf{M}_{ij} - (\mathbf{WL})_{ij}\right)^2,$$

where $(\mathbf{WL})_{ij}$ denotes the entry in the ith row and jth column of $\mathbf{WL}$.

5.2.2 Computing Active Networks

We use a variation of the pathway perturbation approach presented in Chapter 3. In this chapter, we construct an active network for each gene expression sample. Given a sample for a particular condition c and a set of d normal samples H from the same tissue type, we compute the perturbation of gene g in sample c compared to the expression of g in the normal samples. We assume that the distribution of the expression of g in the normal samples can be described by a Gaussian distribution with mean μ_g and standard deviation σ_g. Then, the p-value of the perturbation of gene g is given by the statistic

$$t_g = \frac{c_g - \mu_g}{\frac{\sigma_g}{\sqrt{n}}},$$

which follows the t distribution with $n-1$ degrees of freedom. After computing the p-value for each gene, we apply the method of Chapter 3 to the wiring diagram W.

5.2.3 Non-Negative Matrix Approximation

We evaluate two algorithms for non-negative matrix approximation in this chapter. The method developed by Lee and Seung [98] and described in Algorithm 4 begins by randomly initializing $\mathbf{W}$ and $\mathbf{L}$. The following steps are repeated until convergence. First, hold $\mathbf{L}$ fixed and update all values in $\mathbf{W}$. Second, normalize the sum of each column of $\mathbf{W}$ to one. Third, hold $\mathbf{W}$ fixed and update all values in $\mathbf{L}$. Lee and Seung prove that this algorithm converges to a local optimum. In practice we say that the algorithm converges when the relative difference between two consecutive values of the objective function is less than 0.001.

Algorithm 4 Approximately factorize non-negative $\mathbf{M}$ into $\mathbf{WL}$

Require: $\mathbf{M} \geq 0$
  Randomly initialize $\mathbf{W}$ and $\mathbf{L}$
  $q_0, q_1 \longleftarrow 1$
  repeat
    $q_0 \longleftarrow q_1$
    $\mathbf{W}_{hi} \longleftarrow \mathbf{W}_{hi} \sum_{j=1}^{m} \frac{\mathbf{M}_{hj}}{(\mathbf{WL})_{hj}} \mathbf{L}_{ij}$
    $\mathbf{W}_{hi} \longleftarrow \frac{\mathbf{W}_{hi}}{\sum_{j=1}^{m} \mathbf{W}_{ji}}$
    $\mathbf{L}_{ij} \longleftarrow \mathbf{L}_{ij} \sum_{h=1}^{n} \mathbf{W}_{hi} \frac{\mathbf{M}_{hj}}{(\mathbf{WL})_{hj}}$
    $q_1 \longleftarrow \sum_{hj} (\mathbf{M}_{hj} - (\mathbf{WL})_{hj})^2$
  until $q_0 - q_1 \leq 0.001$
  return $\mathbf{W}$ and $\mathbf{L}$

The Boolean model developed in Chapter 4 naturally ensures that in a block $(G, \mathcal{P}, caln)$, G was a subgraph of every active network in $\mathcal{P}$. We briefly discuss an approach to achieve a similar property with the additive model. Suppose we require that every network lego computed has at most a maximum number of edges. We can achieve this property using Hoyer's [69] approach of specifying a lower bound on the sparsity of the rows of $\mathbf{L}$, where the sparsity of a vector x of length m is

$$S(x) = \frac{\sqrt{n} - \frac{\sum_i |x_i|}{\sqrt{\sum_i x_i^2}}}{\sqrt{n} - 1}.$$

If x is very sparse and contains only one non-zero element, then $S(x) = 1$. In contrast, if all elements of x are equal and non-zero then $S(x) = 0$. The sparsity function smoothly interpolates between the two extremes. Then a lower bound on $S(x)$ is equivalent to an upper bound on $\frac{\sum_i |x_i|}{\sqrt{\sum_i x_i^2}}$. If x is a unit vector (*i.e.* $\sum_i x_i = 1$), then $\sum_i \lceil x_i \rceil$ is an upper bound on the number of edges in the corresponding network lego.

The procedure begins with random initialization of $\mathbf{W}$ and $\mathbf{L}$. The algorithm repeats the following steps until convergence. First, hold $\mathbf{L}$ fixed and update each column of $\mathbf{W}$ separately while maintaining the sparsity requirement. Second. normalize the columns of $\mathbf{W}$ such that they sum to one. Third, hold $\mathbf{W}$ fixed and update each row of $\mathbf{L}$ separately while maintaining the sparsity requirement. We say that the algorithm has converged when the relative difference between consecutive values of the objective function are less than .001.

Many methods currently exist to perform non-negative matrix approximation. Lin [105] compares a number of different methods for non-negative matrix approximation on both synthetic and real data. He concludes that projected gradient, and multiplicative update methods converge in a shorter time. Methods such as alternating least-squares [163] converge in fewer iterations but requires a longer overall time to converge due in part to expensive operations during each iteration.

5.2.4 Model Selection

An important aspect of non-negative matrix approximation is selecting the parameter k; in our context, k is the number of network legos. In the literature, this problem is termed "model selection". To select the number of network legos, we adapt a the method proposed by Ben-Hur, Elisseeff, and Guyon [13]. They argue that the model should be chosen such that it represents a stable partition of the data with respect to missing data. They propose a sub-sampling approach to introduce missing data into the procedure. For each choice of model parameter k, we perform the following steps with the aim of estimating the stability of the computed factors for different sub-samples of the input set of active networks:

1. **Construct sub-samples of $\mathcal{A}$.** We construct multiple (in our case, 30) sets of sub-samples from the set $\mathcal{A}$ of active networks. Each sub-sample contains at least $\max(\frac{3}{4}|\mathcal{A}|, k)$ active networks, chosen uniformly at random without replacement. . Note that we must choose greater than k active networks; otherwise, the matrix has a trivial exact factorization. Note also any two sub-samples will have at least $|\mathcal{A}|/2$ active networks in common.

2. **Compute network legos for each sub-sample $\mathcal{S}$.** Let $\mathbf{M}^{(S)}$ denote the matrix formed by retaining only those rows in $\mathbf{M}$ that correspond to active networks in $\mathcal{S}$. For each sub-sample $\mathcal{S}$, we perform the non-negative matrix approximation decomposition into $\mathbf{M}^{(S)} \approx \mathbf{W}^{(S)}\mathbf{L}^{(S)}$.

3. **Assign each active network in a sub-sample $\mathcal{S}$ to a column of $\mathbf{W}^{(S)}$.** For each row $1 \leq a \leq |\mathcal{S}|$, set
$$\mathbf{c}_a = \arg\max_i \mathbf{W}_{ai},$$
i.e., assign the active network corresponding to row a to the column of $\mathbf{W}^{(S)}$ that has the largest value in the ath row of $\mathbf{W}^{(S)}$.

4. **Compute the co-clustered matrix for sub-sample $\mathcal{S}$.** Construct a symmetric matrix $C^{(\mathcal{S})} \in \{0,1\}^{|\mathcal{S}|\times|\mathcal{S}|}$ such that $C_{ij} = C_{ji} = 1$ iff $\mathbf{c}_i = \mathbf{c}_j$, i.e., if active networks corresponding to rows i and j in $\mathbf{M}^{(\mathcal{S})}$ are assigned to the same column in the previous step.

5. **Compute the distance between co-clustered matrices for every pair of sub-samples.** For every pair of sub-samples $\mathcal{S}$ and $\mathcal{T}$, compute the distance $\kappa(\mathcal{S}, \mathcal{T})$ between the co-clustered matrices $C^{(\mathcal{S})}$ and $C^{(\mathcal{T})}$ as follows:
$$\kappa(\mathcal{S}, \mathcal{T}) = \frac{\langle C^{(\mathcal{S})}, C^{(\mathcal{T})}\rangle}{\langle C^{(\mathcal{S})}, C^{(\mathcal{S})}\rangle + \langle C^{(\mathcal{T})}, C^{(\mathcal{T})}\rangle - \langle C^{(\mathcal{S})}, C^{(\mathcal{T})}\rangle},$$
where we define the dot product for a pair of co-clustered matrices $C^{(1)}$ and $C^{(2)}$ by
$$\langle C^{(1)}, C^{(2)}\rangle = \sum_{ij} C^{(1)}_{ij} C^{(2)}_{ij}.$$

6. **Measure the stability of the factorizations of all the sub-samples.** Assign the median value of all the inter-co-clustered matrix distances computed in the previous step as the *partition stability* σ_k.

Finally, we choose that value of k that maximizes σ_k.

5.2.5 Synthetic Dataset Generation

We generate synthetic data to evaluate the effectiveness of non-negative matrix approximation for rediscovering the basis networks that we use to construct the synthetic data. The synthetic data generation is governed by multiple parameters during the following four steps: (i) basis network construction, (ii) basis network overlap, (iii) synthesis of combined networks, and (iv) addition of noise.

Basis Network Construction We first construct a set $\mathcal{S}$ of r node-disjoint basis networks. The number of nodes in each network is chosen uniformly at random between the integers s_0 and s_1. The completeness of each network is chosen uniformly at random between positive numbers $0 \leq d_0 \leq d_1 \leq 1$. The induced network has e edges and n nodes, where we define the *completeness* of a network with

$$\frac{2e}{n(n-1)}$$

We repeatedly add edges at random to a basic network until we obtain the chosen completeness. We assign each edge a weight chosen uniformly at random between 0 and 1.

Basis Network Overlap Since true gene modules may share genes, we modify basis networks so that they may overlap. We choose a network overlap o fraction between $0 \leq o \leq 1$ let p be the total number of nodes in the basis networks. We select a pair of distinct basis networks (N_1, N_2) uniformly at random from $\mathcal{S}$. We repeat the following steps $\lceil op \rceil$ times. We select a pair of nodes (n_1, n_2) at random such that $n_1 \in N_1$ and $n_2 \in N_2$. We assign $n_1 = n_2$ to increase the number of overlapping nodes by one.

Synthesis of Combined Networks We construct m combined networks from the basis networks. For each combined network, we select c basis networks uniformly at random (without replacement) from $\mathcal{S}$, where $c_0 \leq c \leq c_1$ and c is itself chosen uniformly in this range. The combined network is the union of the selected basis networks. If an edge appears in multiple selected basis networks, the weight of that edge in the the combined network is the sum of the weights of that edge in the selected basis networks.

Addition of Noise We add edges at random to each combined network. We construct the union of the edges over all basis networks. To each combined network, we add edges selected uniformly at random from the union;the number of edges added is a fraction γ of

the number of edges in the combined network. This noise generation procedure naturally includes changing the weights of edges already included in the combined networks.

5.3 Results for Synthetic Data

Using synthetic data, we evaluate the algorithm of Lee and Seung and the approach proposed by Hoyer.

5.3.1 Comparison of Lee and Seung's and Hoyer's algorithm on Synthetic Data

The method by Lee and Seung [98] is a well-know method for non-negative matrix approximation. Other algorithms like the projected gradient method by Hoyer [69] have additional benefits such as the enforcement of the sparsity of the resulting factors. We compare Hoyer's method to the method by Lee and Seung using synthetic data to evaluate the performance of the method under our additive model. We find that the method by Lee and Seung outperforms Hoyer's method based on their ability to recover basis networks. We performed the comparison using ideal choices of the parameters including model complexity (*i.e.* 100) and **L** sparsity constraints (*i.e.* 0.8591) to allow each algorithm the best possibility for recovering the basis networks. The 0.8591 **L** sparsity figure is based on a maximum number of 278 edges in a basis network and 12590 distinct edges across all basis networks. In Figure 5.1, we show the basis network recovery performance of the method proposed by Hoyer. We apply Lee and Seung's method to the same dataset in Figure 5.2. For each row of **L**, we identify the basis network that most closely matches the row in a Euclidean sense. We find that with no noise or basis network overlap the Lee and Seung recover over 75% of the expected basis networks, while the method proposed by Hoyer identifies none of the expected basis networks. For the remainder of this chapter, we use the Lee and Seung algorithm to perform non-negative matrix approximation [98].

5.3.2 Performance of Lee and Seung's Algorithm on Synthetic Datasets

We construct two synthetic datasets to examine the effectiveness of basis network recovery using non-negative matrix approximation. We construct the synthetic dataset based on parameters set to mimic the size and density of our molecular interaction networks. Our synthetic data set consists of 100 basis networks of sizes ranging from 50 to 75 nodes with completeness between 0.08 and 0.1. We construct combined networks from between 6 to 10 basis networks. We show results from two synthetic datasets one with 200 and then with 1000 active networks.

We fix $k = 100$, the number of basis networks in the synthetic datasets. We measure the

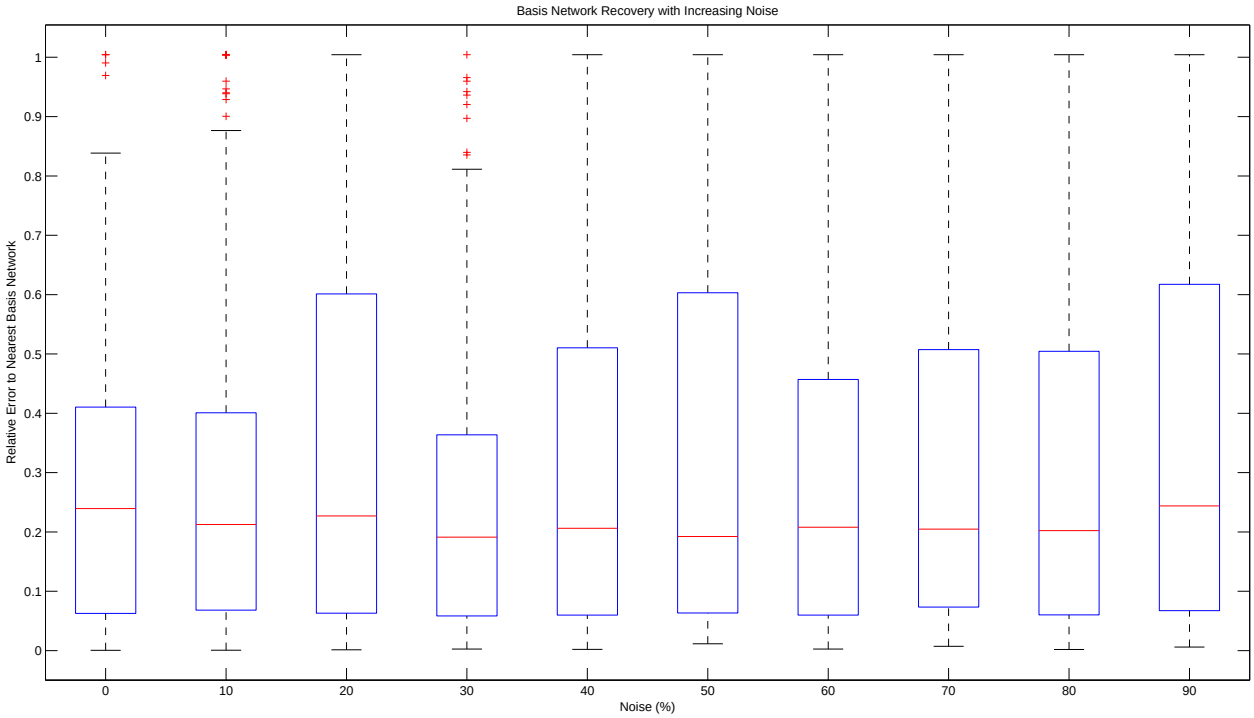


Figure 5.1: Basis network recovery using the method proposed by Hoyer [69]. We examine the performance of the algorithm with increasing amounts of relative noise in the range of 0% to 90%.

ability of the non-negative matrix approximation procedure to correctly recompute the basis networks. For each for of $\mathbf{L}$, we compute the relative error between that row and each basis network (after converting the network to a vector). We noted the smallest of these values, reasoning that the corresponding basis network is best approximated by this row of $\mathbf{L}$. We plot this using box-and-whisker a plot. Figure 5.3 displays basis network recovery with increased overlap (0%, 20%, and 40%) and random edge additions (0%-90%) for increasing levels of noise.

We show that the non-negative matrix approximation procedure recovers about 50% of the basis networks for 200 synthetic active networks and nearly 75% for the dataset of 1000 synthetic active networks. The improved performance of 1000 synthetic active networks is supported by the work of Donoho and Stodden [40]. Donoho and Stodden argue that for a dataset containing graphs with all combinations of basis networks, non-negative matrix approximation will successfully recover the unique set of basis networks. While we have far fewer than 2^{100} active networks sufficient to recover all basis networks, we show that a small fraction of synthetic active networks can resolve many basis networks.

In Figure 5.3 we have measured the effect of adding increasing percentages of noise in the form of random edge additions to each synthetic active network. We show that the non-negative matrix approximation procedure slowly degrades with increased noise up to 90% of size of the original active network. The performance continues to degrade with increased basis network overlap. One explanation is that as nodes from basis networks are joined,

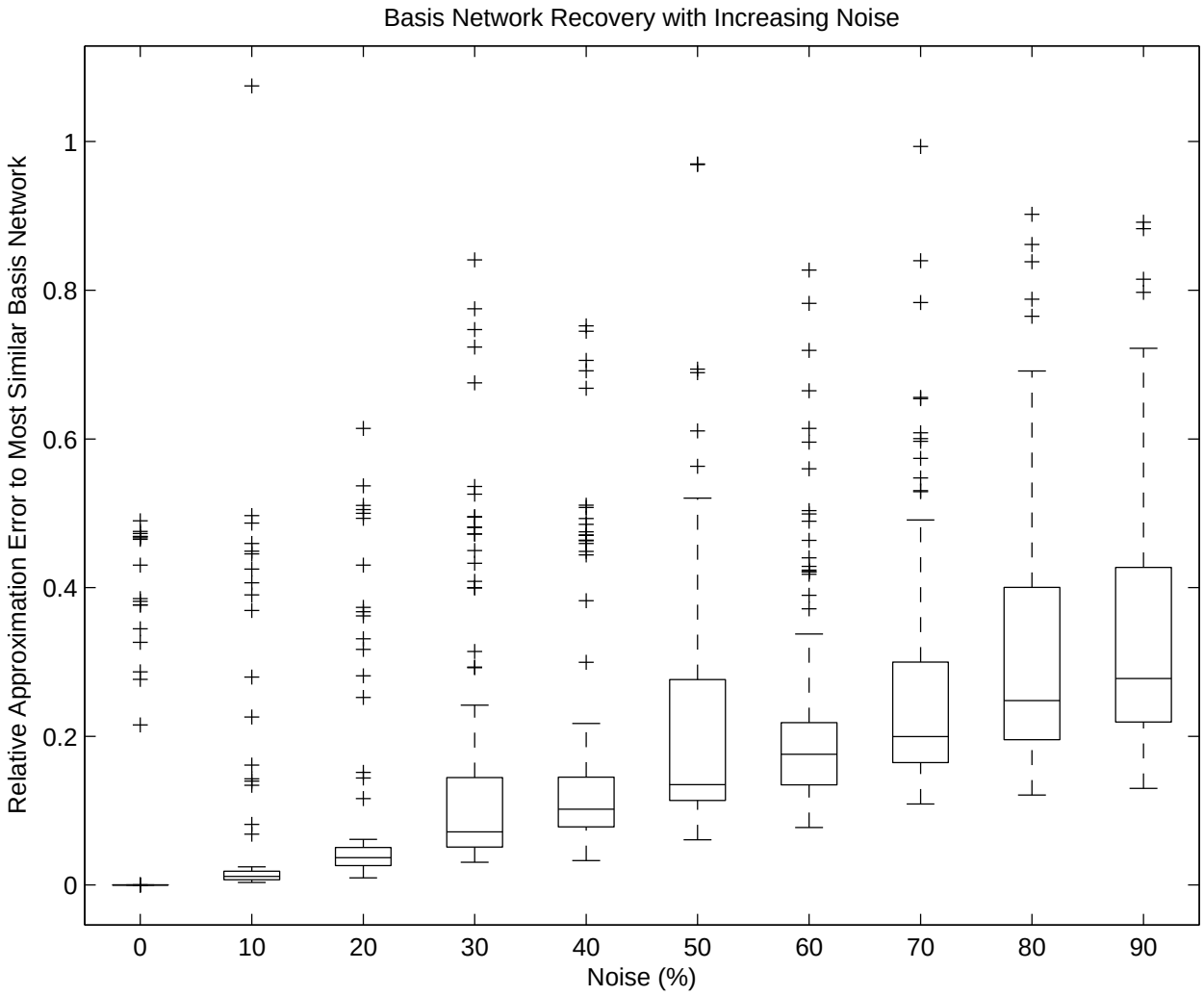


Figure 5.2: We examine the performance of the Lee and Seung algorithm with increasing amounts of relative noise in the range of 0% to 90%.

a fraction of edges are joined as well. The common set of edges between a pair of basis networks effectively induces a new basis network for the non-negative matrix approximation procedure to recover.

5.4 Results for Human Cancer Data

We apply our method to the Global Cancer Map (GCM) gene expression data (also analyzed in the context of specific pathways in Chapter 3) and a human PPI network to identify network legos associated with human cellular responses to cancers.

5.4.1 Human PPI and Cancer Datasets

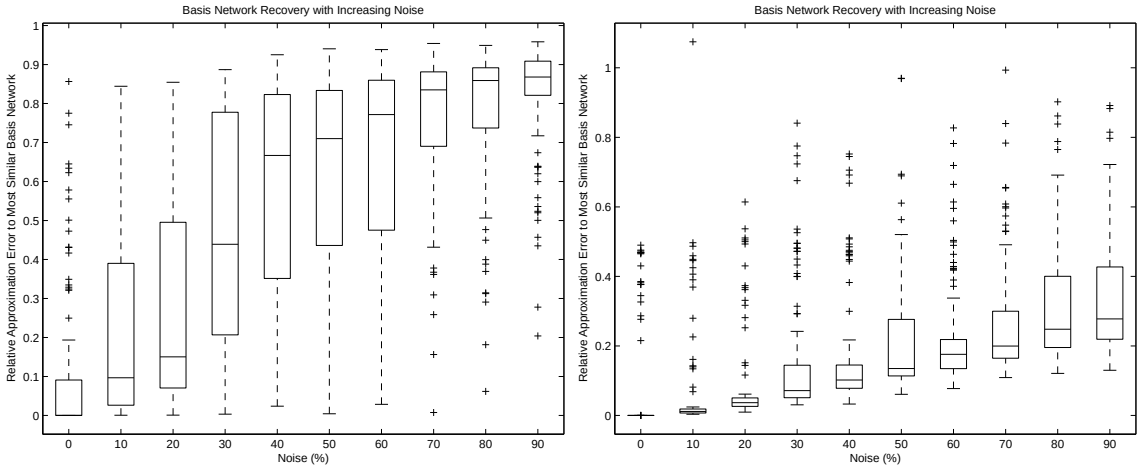
We obtained a network of 31108 molecular interactions between 9243 human gene products by integrating the interactions in the IDSERVE database [132], the results of large scale yeast two-hybrid experiments [138, 159], and 20 immune and cancer signalling pathways in the Netpath database (http://www.netpath.org). The IDSERVE database includes human curated interactions from BIND [7], HPRD [129], and Reactome [81], interactions predicted based on co-citations in article abstracts, and interactions that transferred from lower eukaryotes based on sequence similarity [101]. We derived functional annotations for the genes in our network from the Gene Ontology (GO) [4] and from MSigDB [162].

We used gene expression measurements in the Global Cancer Map (GCM) [133]. The GCM dataset contains 190 samples spanning 18 cancers and 90 samples from 13 normal tissues. Using the method described in Section 5.2.3, we constructed 190 active networks, one for each cancer sample in the dataset.

5.4.2 Effect of Increasing the Number of Network Legos

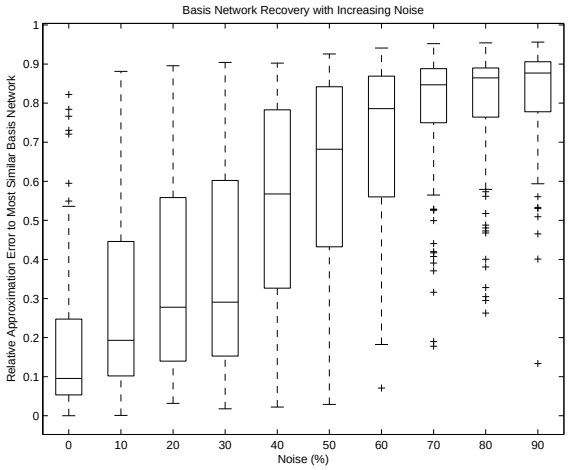
We obtained the results in this section and the next by running the Lee and Seung algorithm 30 times, each with a different random starting value for $\mathbf{W}$ and $\mathbf{L}$. We averaged values over the 30 runs. Note that we did not use the sub-sampling method in these two sections. As we vary k, we track four values: the error and the median sparsity of the rows of $\mathbf{L}$, $\mathbf{W}$, and $\mathbf{W}^{-1}$.

In Figure 5.4, we show trends for various measures of performance as we increase the model complexity, i.e., k, the number of network legos. Figure 5.4(a) shows that the relative error steadily decreases with increasing k. Figures 5.4(b)–(d) display the variation of the median sparsity of the rows of $\mathbf{L}$, the columns of $\mathbf{W}$, and the rows of $\mathbf{W}^{-1}$, respectively, with increasing k. All three values increase with k. Each row of $\mathbf{L}$ corresponds to a network lego; hence, increase in the median sparsity of the rows of $\mathbf{L}$ suggests that as k increases, the median network lego contains fewer interactions. The values in each column of $\mathbf{W}$ are weights that describe how often a lego is used by active networks. The decrease in this

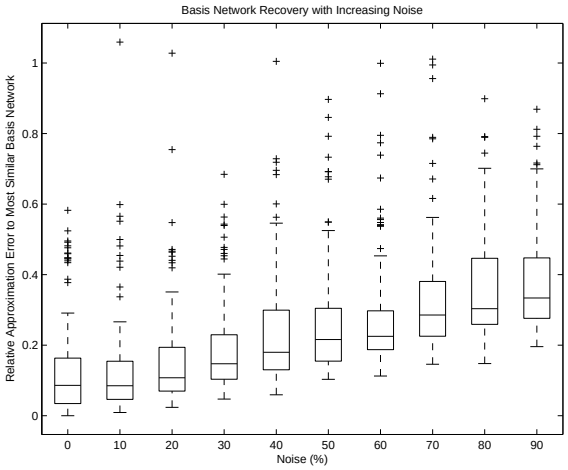


(a) 200, 0%

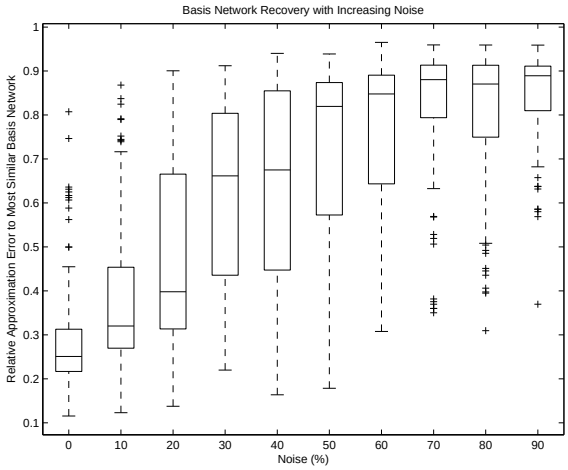
(b) 1000, 0%



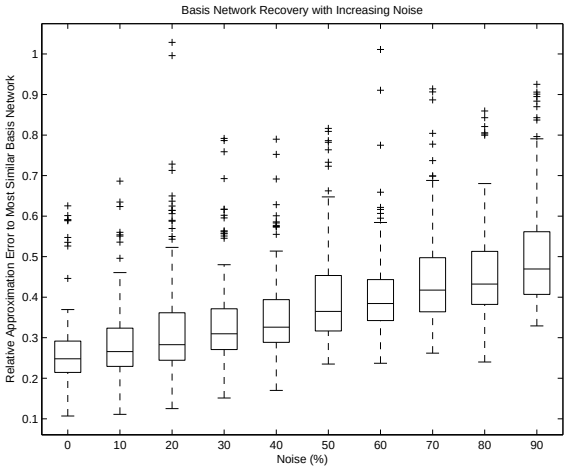
(c) 200, 20%



(d) 1000, 20%



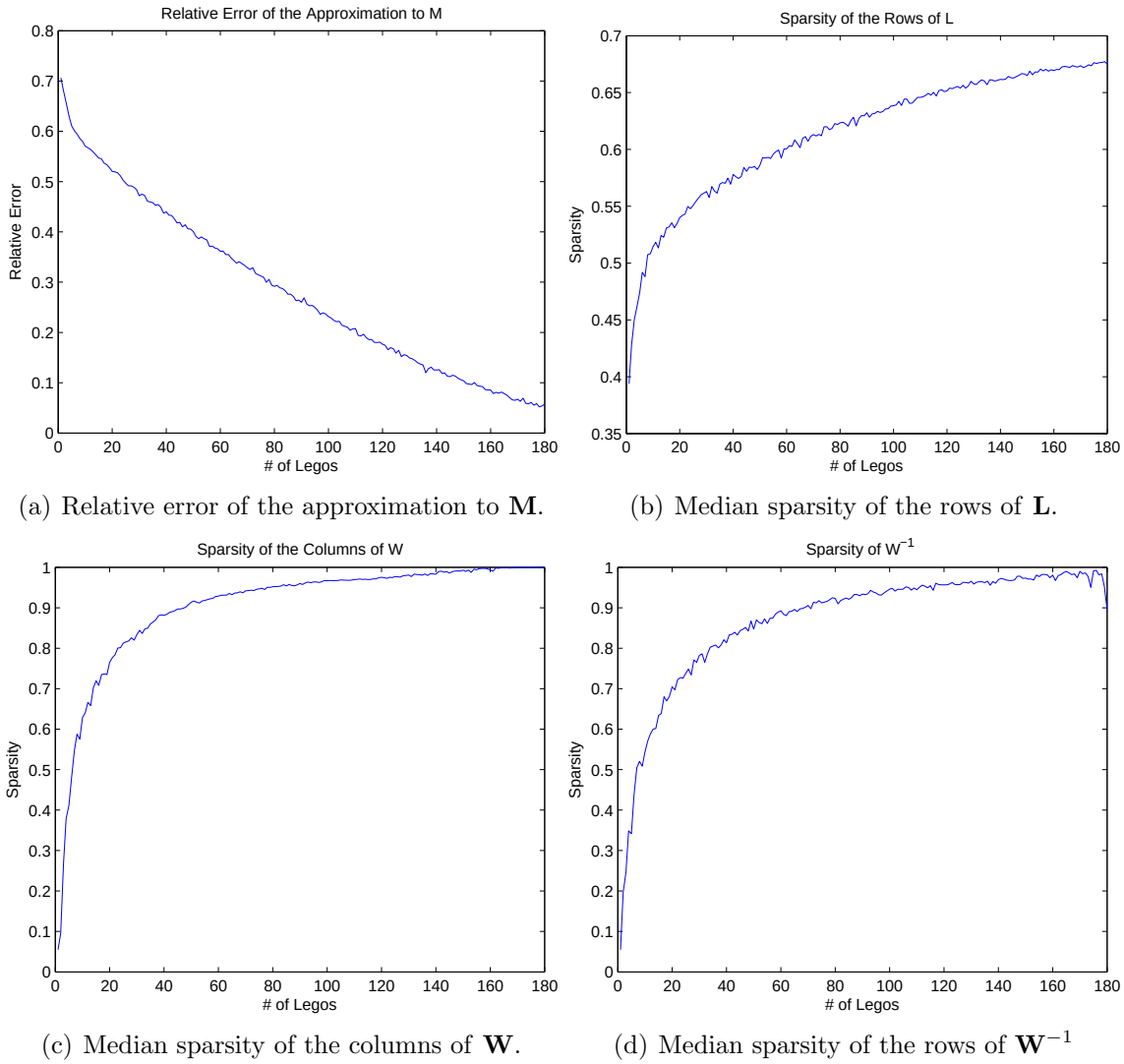
(e) 200, 40%



(f) 1000, 40%

Figure 5.3: An illustration of the effectiveness of non-negative matrix approximation for basis network recovery for different synthetic dataset sizes and percentages of network legos overlap. Each figure contains a box-and-whisker plot showing the distribution of relative error distances to the nearest basis network for each row of $\mathbf{L}$. The caption of each figure lists the number of active networks in the synthetic dataset and the amount of overlap between basis networks.

measure suggests that as we increase k, each network lego is being included in fewer active networks. Finally, the rows of $\mathbf{W}^{-1}$ denote how active networks can be linearly combined to yield network legos; note that $\mathbf{W}^{-1}$ may contain negative values. Once again, increase in the median sparsity of $\mathbf{W}^{-1}$ points to fewer active networks being used to construct the average network lego, as we increase k.



(a) Relative error of the approximation to $\mathbf{M}$.

(b) Median sparsity of the rows of $\mathbf{L}$.

(c) Median sparsity of the columns of $\mathbf{W}$.

(d) Median sparsity of the rows of $\mathbf{W}^{-1}$

Figure 5.4: Variation of four measures of performance with an increasing number of network legos.

5.4.3 Comparison to Known Sample Partition

We translate each cancer sample in the GCM dataset into an active network as described in Section 5.2.1. Each of the 190 samples has a class label (a cancer type) associated with it. The known class labels induce a gold standard partition of the active networks to cancer types. In Figure 5.5, we show the partition distance between non-negative matrix approximation

partitions with increasing model complexity to the known partitions of the cancer samples. The samples are known to come from 18 cancer types. The model with the closest partition distance to the known partition occurs with 20 network legos. The result shows that non-negative matrix approximation produces clusters that are close to the known partition at approximately the expected model complexity.

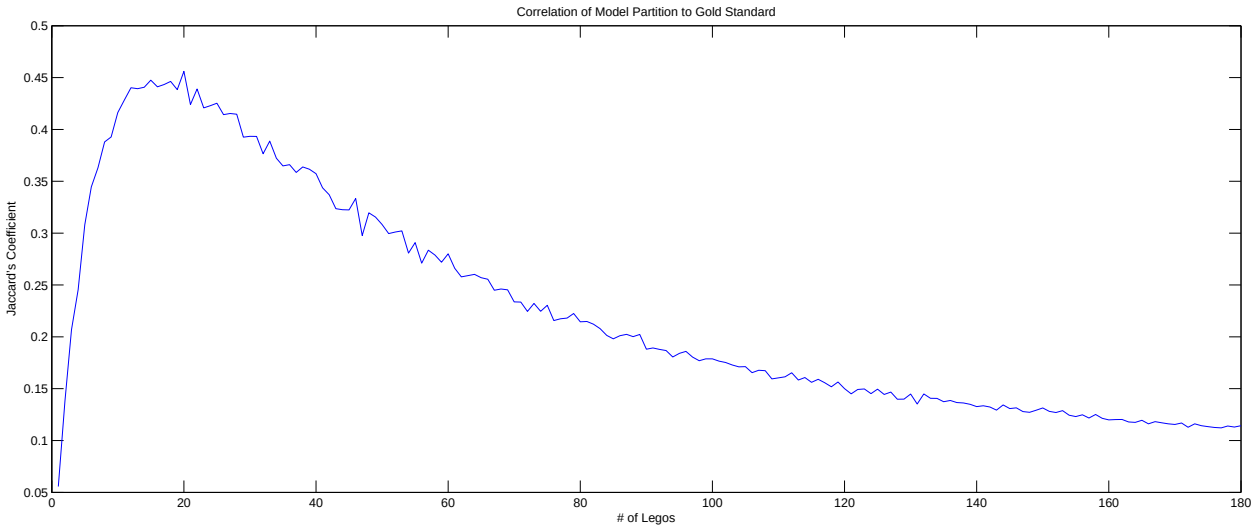


Figure 5.5: Variation with the number of legos of the partition distance between non-negative matrix approximation-based clustering of the sample active networks and the known classes of the cancer samples.

5.4.4 Choosing the Number of Network Legos

Our choice of the number of network legos is influenced by two factors: relative error and partition stability. We apply the method described in Section 5.2.4 to identify values that have high average partition stability. In Figure 5.6, we show a distribution of partition stability values for k in the range of 1–100 and in the range 100–150. We show a subset of the range of k for clarity. The reader should refer to Figure 5.4(a) for corresponding residual error values. We find that k values closer to 100 have better partition stability. Very low values of k show a similar increase in partition stability, but the high residual error at low values makes them poor choices. There is a trend towards increasing median partition stability for $k > 40$. With a median partition stability of just under 0.45, a selection of 137 and 147 network legos may indicate a natural partitioning of the active networks.

5.4.5 Analysis of GCM network legos

With $k = 137$, we find a low relative error of 0.135 and a high median partition stability of 0.6. We use this value of k for further analysis.

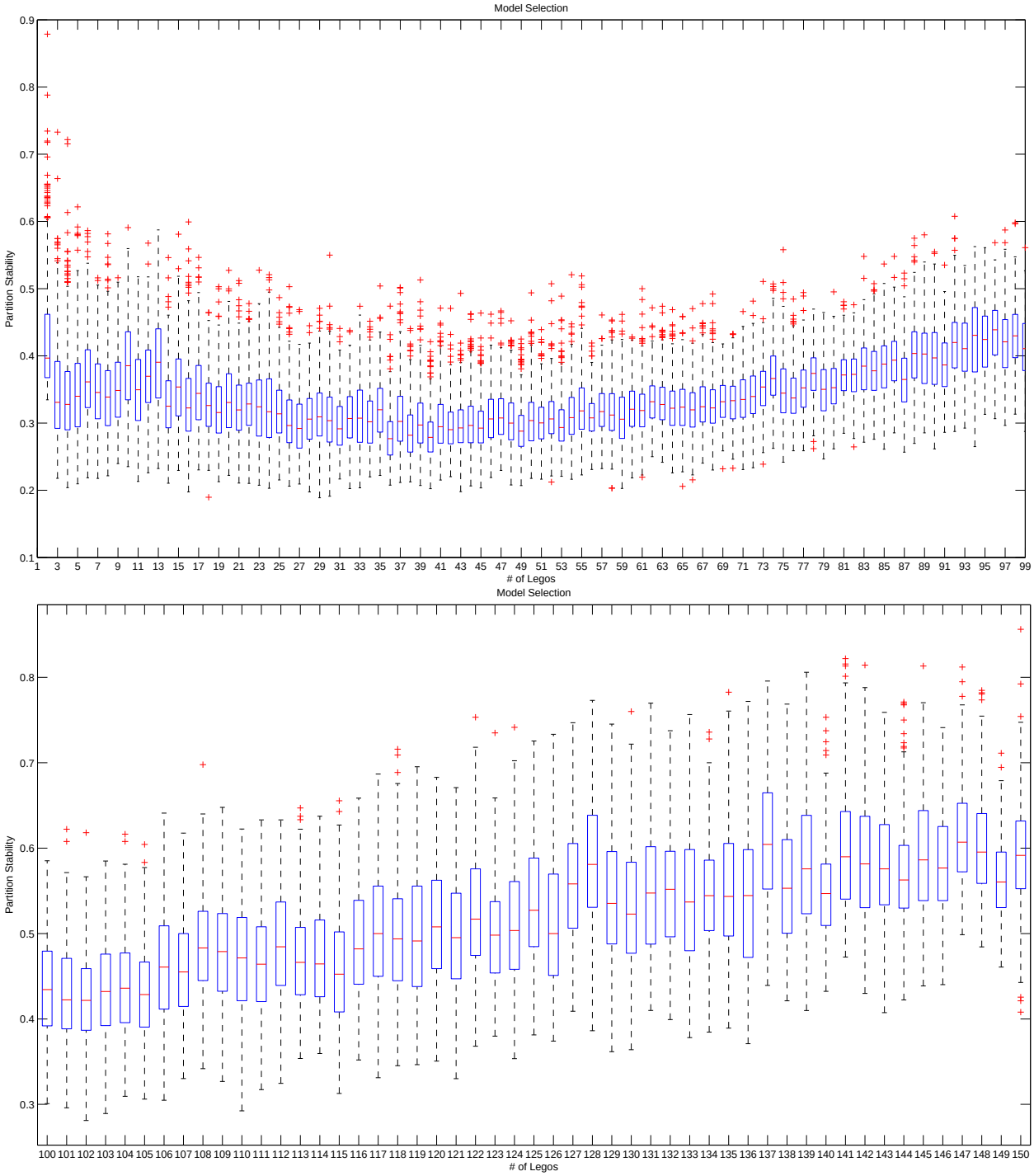
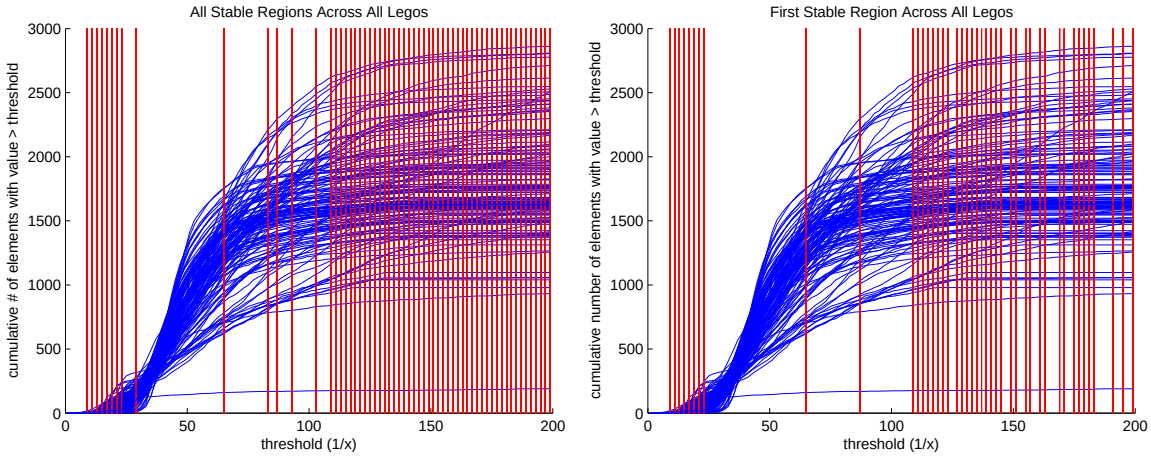


Figure 5.6: Partition stability distributions for model sizes from 1–100 and 100–150. For each model size, we show a box-and-whisker plot representing the distribution of subsample partition similarity values between every pair of sub-samples.

Vacant Regions in the Distribution of Network Lego Edge Weights We find that through the addition and subtraction of active networks there is frequently a subnetwork that is emphasized. The subnetwork can be identified by its relatively enhanced interaction weights. Next, we define a few terms to help identify these particularly high weighted subnetworks contained in network legos.

An observation we make is that the distribution of network lego edge weights contain vacant regions. For each network lego L, we define vacant regions defined by boundaries a and b such that L does not have an edge weight in the range a–b and $b - a > \delta$. We are interested in these regions because they provide natural partitions between the highest edge weights and lowest edge weights.

In Figure 5.7, we highlight vacant regions in the range of network lego edge weights. For each network lego, we show the cumulative number of edges with weight greater than a threshold t that decreases with the terms of a harmonic series. We identify vacant regions as those regions that have no edges with weight in the range of four terms of the harmonic series. We refer to the set of interactions with weights greater than the start of the first vacant region as the *top interactions*. In the following section, we further characterize the significance of the top interactions for a selected set of network legos.



(a) (b)

Figure 5.7: Cumulative distribution of the weights of network lego edges. For each network lego, we plot the total number of edges with weight greater than a threshold (blue lines). We identify vacant regions with red vertical bars. (a) We identify every vacant region in any network lego. (b) We identify the first vacant region in any network lego.

5.4.6 Integrins, Metalloproteinases and Ovarian Adenoma

We identify a network lego created primarily through the addition of five response networks from ovarian adenoma samples. The 18 top interactions consist connect only 21 proteins. Of the 21 proteins, 10 are annotated with Matrix Metalloproteinases (MMPs) (p-value 2.69×10^{-11}) and 8 associated with integrins. Figure 5.8 shows the network lego.

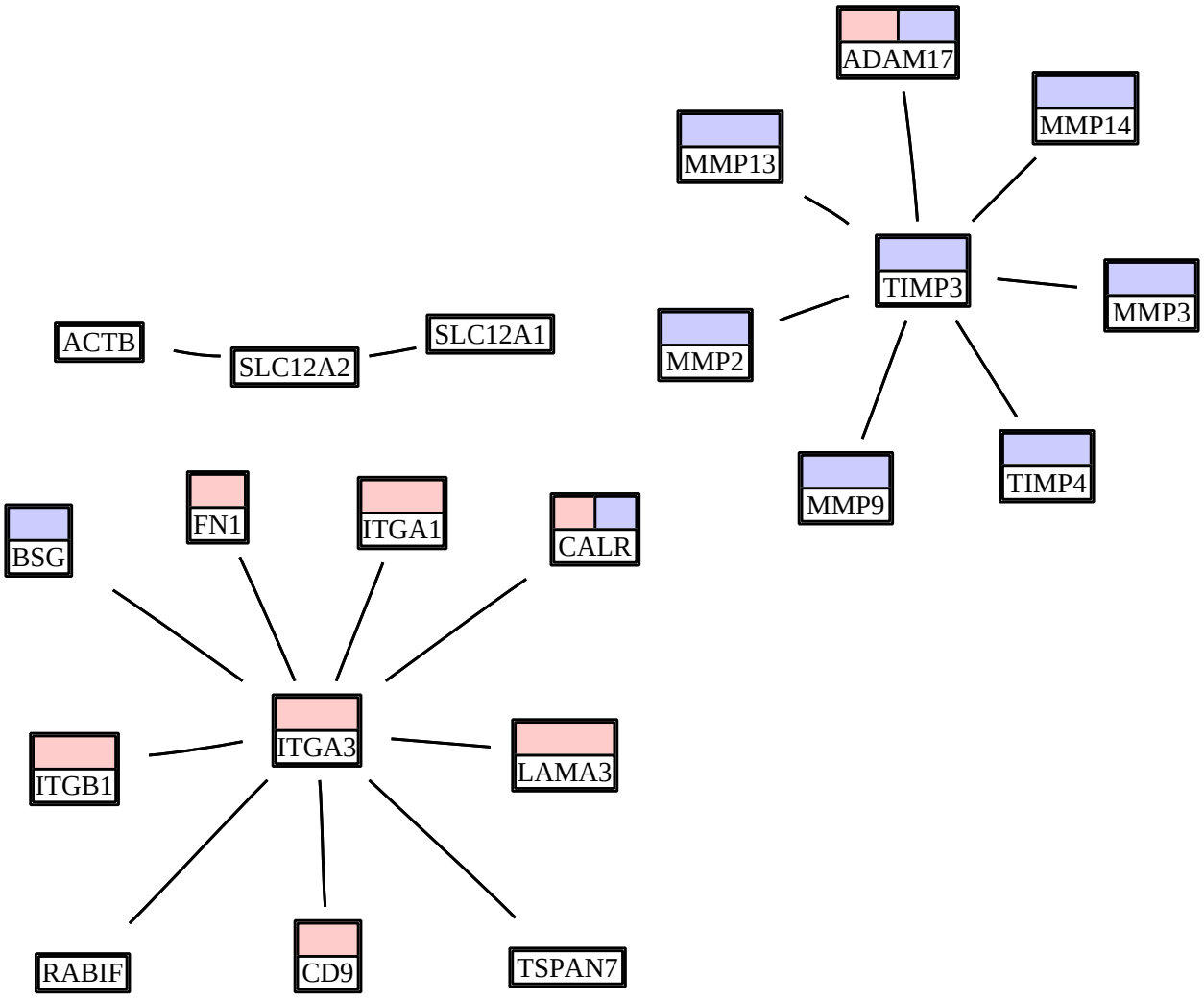


Figure 5.8: The 18 top interactions of a network lego enriched in integrins and matrix metalloproteinases. The lego is constructed from the addition of $five$ ovarian adenomas.

The network lego contains two star networks, one primarily enriched in integrins and integrin associated proteins and a second primarily enriched in MMPs and MMP associated proteins. Examination of the associated column of **W** indicates that this network lego is used to reconstruct the active networks of nearly all cancer types in the GCM dataset. The broad use of this network lego indicates that it may be a common component of cellular response to cancer.

Integrins and Metastasis Integrins are transmembrane proteins responsible for cellular adhesion and signal transduction [117]. Ziober, Lin, and Kramer [198] outline the important role that integrin proteins play in metastasis. These proteins bind to the extracellular matrix and permit invasion into the basement membrane and give increased access to blood vessels. This mechanism for metastasis is known to occur in breast cancer [80], bladder cancer [198] and others.

Matrix Metalloproteinases Birkedal-Hansen et al. [18] describe the activity of matrix metalloproteinases (MMPs). MMPs are Zn-endopeptidases that can cleave most of the components of the extracellular matrix. As MMPs enable cellular motility, their association with metastasis and angiogenesis has been thoroughly reviewed [37, 43]. MMP perturbation is known to be associated with metastasis development across many cancer types.

5.4.7 Retinoblastoma pathway and Leukemias

Chim, Fung, and Liang [27] discuss the role of the retinoblastoma (RB) pathway as one of several cell cycle checkpoints. The presence of the RB protein is important for maintaining the S1 checkpoint. Unphosphorylated RB prevents entry into S1 by binding to E2F, a primary transcription factor for S1 related genes. Disruption of the RB pathway is a common mechanism of pathogenesis in multiple myeloma. We find a network lego primarily constructed from an AML sample and ALL sample. The top interactions in the network lego consists of 57 genes of which 5 have been annotated with the RB (p-value 1.3×10^{-3}) pathway, 7 are associated with the cell cycle (p-value 6.9×10^{-3}). The construction of the network lego suggests that RB inhibition is a feature of multiple leukemias. Kornblau et al. [92] confirm inhibition of RB protein in AML and Cheng et al. [24] confirm the lack of the RB protein in ALL.

5.4.8 MAPK pathway and CNS Tumors

We explore the choice of $k = 74$ next. Based on partition stability, $k = 74$ is more stable than nearby choices of k. We identify a network lego with 431 genes among the top interactions. The network lego is primarily constructed from five medulloblastoma samples and one glioblastoma sample. Of the 431 genes 19 are part of the MAPK pathway (p-value 9.2×10^{-8}) and 12 are part of the P38MAPK pathway (p-value 6.8×10^{-8}). Macdonald et al. [109] have implicated MAPK pathway as being associated with metastasis in medulloblastoma. They

find that platelet-derived growth factor α (PDGFA) increases medulleoblastoma migration. They find that both neutralizing antibodies to PDGFA and U0126 inhibit PDGFA-stimulated migration.

5.5 Summary

In the method we propose, each network lego has a dual representation as a weighted network of interactions and as a linear combination of active networks. The weight of an interaction corresponds to the importance of the interaction to the network lego. These network legos are in a sense the building blocks of their associated set of active networks. We propose that the network legos are network units perturbed as a whole when the cell is individually subjected to multiple stresses. The linear combination of active networks representing a network lego provide the context that defines the network lego: the corresponding stresses or stimuli and the associated weights. The weights indicate the relative importance of a stress response in defining the network lego. In Figure 1.1, we illustrate the decomposition of active networks in to network legos. The decomposition is followed by a process by which we reconstitute each active network by a set of network legos.

Chapter 6

Conclusions

In this thesis, we propose a number of new methods for analyzing molecular interaction and gene expression data in an integrated manner. This thesis not only proposes new methods for analyzing data from large scale biochemical assays but also promotes a new way of thinking about modules in the context of molecular interaction networks.

6.1 Chapter Specific Contributions

In this section, we highlight the contributions and findings associated with the projects discussed in this thesis.

6.1.1 Pathway Perturbation

We propose a knowledge-driven approach that integrates a curated pathway of molecular interactions with gene expression data from both cellular stress samples such as cancer tissue and normal samples such as healthy tissue from the same patient to compute the perturbation of the pathway in response to the cellular stress. We develop a score for pathway perturbation based on a rigorous statistical procedure using a meta-analytic technique coupled with permutation testing. We use a simulated annealing approach to identify the most significantly group of perturbed interactions. We apply our method to a compendium of 18 cancer types and a set of 20 cancer and immune signaling pathways. We show that the significance of the most perturbed subpathway is frequently more significant than the entire pathway. We compare our method to GSEA and we find that our method has superior sensitivity. We exploit the compendium of pathway-cancer perturbation scores to construct separate pathway and cancer association diagrams. We find that the TNF-α, TGF-β, EGFR1, and B cell receptor pathways are perturbed in response to many cancer types. Using the pathway association diagram and analyzing the underlying gene expression data, we find evidence that up regulated expression of TNF-α, TGF-β, and Integrins may suggest metastatic potential. We identify an association between melanoma and bladder cancer that suggests that IL-2

infusion may have a similar therapeutic effect in bladder cancer as it does in melanoma.

6.1.2 Boolean Network Legos

We propose a data-driven approach that integrates a molecular interaction network with gene expression data to compute the active network, the molecular interactions within a cell perturbed by a single stress or stimulus. We formulate a Boolean approach to finding network legos for a set of active networks. We decompose the problem of finding network legos into two parts. First, we enumerate candidate network legos by formulating the problem as one of finding all closed itemsets. Second, we construct a directed acyclic graph (DAG) using the subset associations between the itemsets. We exploit the DAG to identify the most significant candidate network legos. We assess the quality of a network lego by defining measures of stability, and recoverability. We apply Network Lego to two human gene expression datasets. We first apply our method to a dataset of three leukemias ALL, AML, and MLL. The decomposition of the three leukemias into network legos allowed us to accurately identify the c-Kit pathway as being perturbed as an integral part of AML proliferation but not as part of ALL or MLL. We apply our approach to a collection of 178 arrays measuring the gene expression responses of HeLa cells and primary human lung fibroblasts to 13 distinct stresses including cell cycle arrest, heat shock, endoplasmic reticulum stress, oxidative stress, and crowding. Our study reveals that about four–six hours after treatment with DTT or menadione, fibroblasts shut down the cell cycle far more aggressively than fibroblasts or HeLa cells do in response to other treatments.

6.1.3 Weighted Network Legos

We propose a formulation of network legos that takes weighted active network interaction into account in a meaningful way. We formulate the problem such that we can directly optimize for recoverability, *i.e.* how well each active network can be represented by a linear combination of network legos. We solve the problem by finding the non-negative matrix approximation of a matrix that encodes the set of active networks. We apply our method to both synthetic and real data. We generate the synthetic datasets using a generative model that allows us to measure network lego recovery with increased levels of noise. We test the performance of our method on two datasets, one containing 200 and the other containing 1000 large synthetic active networks. We show that with more active networks our method correctly resolves more of the network legos. We show that the performance of our method slowly degrades with increased noise. We apply our method to a human cancer dataset including 190 samples from 18 cancers. We compute active networks for each of the 190 samples using the pathway perturbation method. We show that the recoverability of the active networks increases with the number of network legos. We show that the non-negative matrix approximation procedure can recapture the original partitioning of the data. We use a measure of partition stability to identify a stable number of partitions for the cancer data. We identify a network lego that associates the integrins and matrix metalloproteinases together in ovarian adenoma and other cancers. We find a network lego including the RB

pathway associated with multiple leukemias. We find that the MAPK pathway is perturbed in association with multiple tumors of the central nervous system.

6.2 New Facets of Modular Cell Biology

This thesis contributes to the community not only with novel computational methods and tools, but in the way that we view biological systems. We have advanced the notion of modular cell biology, and we have augmented the concept with new ways to think about context sensitive modules.

Throughout this thesis we explicitly investigate and highlight examples of what we believe to be modular behavior of cells. In Chapter 3 we offer results that show modular activity within pathways under a compendium of cancer phenotypes. We find evidence of modular perturbation in the TNF-α, TGF-β, and Integrin receptor pathways that may suggest metastatic potential.

We find evidence that suggests not only that modules exist but also that cells respond to stress or a stimulus by appropriately modulating the activities of a subset of these modules. We have referred to these modules as network legos. Through Chapters 4 and 5, we propose the concept of network legos which provide a new facet of interpretation to modules. Each network lego has a dual representation: a network of interactions and a mathematical combination of active networks. These network legos are in a sense the building blocks of their associated set of active networks. We suggest that each network lego is a molecular machine perturbed as a unit by the cell and that the cell responds to any given stress or stimulus by appropriately perturbing the network legos.

Our understanding of cell biology and its associated molecular interaction networks is far from comprehensive. We believe that modular cell biology is a valuable tool for understanding complex networks in molecular systems biology. This thesis has conceptualized and developed expressive notions to advance modular cell biology —active networks, network legos, and representations of active networks in terms of network legos— to comprehend and situate experimental conditions in the context of other datasets. Our approach provides a unified framework that empowers biologists to pose sophisticated queries about different cellular states.

Bibliography

[1] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the Twentieth International Conference on Very Large Databases*, pages 487–499, Santiago, Chile, 1994.

[2] C. Alfarano, C. E. Andrade, K. Anthony, N. Bahroos, M. Bajec, K. Bantoft, D. Betel, B. Bobechko, K. Boutilier, E. Burgess, K. Buzadzija, R. Cavero, C. D'Abreo, I. Donaldson, D. Dorairajoo, M. J. Dumontier, M. R. Dumontier, V. Earles, R. Farrall, H. Feldman, E. Garderman, Y. Gong, R. Gonzaga, V. Grytsan, E. Gryz, V. Gu, E. Haldorsen, A. Halupa, R. Haw, A. Hrvojic, L. Hurrell, R. Isserlin, F. Jack, F. Juma, A. Khan, T. Kon, S. Konopinsky, V. Le, E. Lee, S. Ling, M. Magidin, J. Moniakis, J. Montojo, S. Moore, B. Muskat, I. Ng, J. P. Paraiso, B. Parker, G. Pintilie, R. Pirone, J. J. Salama, S. Sgro, T. Shan, Y. Shu, J. Siew, D. Skinner, K. Snyder, R. Stasiuk, D. Strumpf, B. Tuekam, S. Tao, Z. Wang, M. White, R. Willis, C. Wolting, S. Wong, A. Wrong, C. Xin, R. Yao, B. Yates, S. Zhang, K. Zheng, T. Pawson, B. F. Ouellette, and C. W. Hogue. The biomolecular interaction network database and related tools 2005 update. *Nucleic Acids Res*, 33(Database issue), January 2005.

[3] S. Armstrong, J. Staunton, L. Silverman, R. Pieters, M. den Boer, M. Minden, S. Sallan, E. Lander, T. Golub, and S. Korsmeyer. MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia. *Nat Genet*, 30(1):41–7, 2002.

[4] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. the gene ontology consortium. *Nat Genet*, 25(1):25–29, May 2000.

[5] M. B. Atkins, M. T. Lotze, J. P. Dutcher, R. I. Fisher, G. Weiss, K. Margolin, J. Abrams, M. Sznol, D. Parkinson, M. Hawkins, C. Paradise, L. Kunkel, and S. A. Rosenberg. High-dose recombinant interleukin 2 therapy for patients with metastatic melanoma: analysis of 270 patients treated between 1985 and 1993. *J Clin Oncol*, 17(7):2105–2116, July 1999.

[6] K. E. Bachman and B. H. Park. Duel nature of tgf-beta signaling: tumor suppressor vs. tumor promoter. *Current opinion in oncology*, 17(1):49–54, January 2005.

[7] G. D. Bader, D. Betel, and C. W. Hogue. Bind: the biomolecular interaction network database. *Nucleic Acids Res*, 31(1):248–250, January 2003.

[8] G. D. Bader and C. W. Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, 4(1), January 2003.

[9] J. S. Bader, A. Chaudhuri, J. M. Rothberg, and J. Chant. Gaining confidence in high-throughput protein interaction networks. *Nat Biotechnol*, 22(1):78–85, January 2004.

[10] Z. Bar-Joseph, G. K. Gerber, T. I. Lee, N. J. Rinaldi, J. Y. Yoo, F. Robert, D. B. Gordon, E. Fraenkel, T. S. Jaakkola, R. A. Young, and D. K. Gifford. Computational discovery of gene modules and regulatory networks. *Nat Biotechnol*, 21(11):1337–42, 2003.

[11] W. T. Barry, A. B. Nobel, and F. A. Wright. Significance analysis of functional categories in gene expression studies: a structured permutation approach. *Bioinformatics*, 21(9):1943–9, 2005.

[12] K. Basso, A. A. A. Margolin, G. Stolovitzky, U. Klein, R. Dalla-Favera, and A. Califano. Reverse engineering of regulatory networks in human b cells. *Nat Genet*, March 2005.

[13] A. Ben-Hur, A. Elisseeff, and I. Guyon. A stability based method for discovering structure in clustered data. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, pages 6–17, 2002.

[14] M. Bene, M. Bernier, R. Casasnovas, G. Castoldi, W. Knapp, F. Lanza, W. Ludwig, E. Matutes, A. Orfao, C. Sperling, et al. The Reliability and Specificity of c-kit for the Diagnosis of Acute Myeloid Leukemias and Undifferentiated Leukemias. *Blood*, 92(2):596, 1998.

[15] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society*, 57:289–300, 1995.

[16] S. Bergmann, J. Ihmels, and N. Barkai. Similarities and differences in genome-wide expression data of six organisms. *PLoS Biol*, 2(1):E9, 2003.

[17] A. Bernal, U. Ear, and N. Kyrpides. Genomes OnLine Database (GOLD): a monitor of genome projects world-wide. *Nucleic Acids Res*, 29(1):126–7, 2001. http://www.genomesonline.org.

[18] H. Birkedal-Hansen, W. G. Moore, M. K. Bodden, L. J. Windsor, B. Birkedal-Hansen, A. DeCarlo, and J. A. Engler. Matrix metalloproteinases: a review. *Critical reviews in oral biology and medicine : an official publication of the American Association of Oral Biologists*, 4(2):197–250, 1993.

[19] R. Bonneau, D. J. Reiss, P. Shannon, M. Facciotti, L. Hood, N. S. Baliga, and V. Thorsson. The inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. *Genome Biol*, 7(5), 2006.

[20] C. Borgelt and M. R. Berthold. Mining molecular fragments: Finding relevant substructures of molecules. In *ICDM '02: Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM'02)*, Washington, DC, USA, 2002. IEEE Computer Society.

[21] B.-J. Breitkreutz, C. Stark, and M. Tyers. The GRID: the General Repository for Interaction Datasets. *Genome Biol*, 4(3):R23, 2003.

[22] J. P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proc Natl Acad Sci U S A*, 101(12):4164–4169, March 2004.

[23] M. Charikar. Greedy approximation algorithms for finding dense components in a graph. In *Proceedings of 3rd International Workshop on Approximation Algorithms for Combinatorial Optimization*, pages 84–95. Springer, 2000.

[24] J. Cheng, P. Scully, J. Y. Shew, W. H. Lee, V. Vila, and M. Haas. Homozygous deletion of the retinoblastoma gene in an acute lymphoblastic leukemia (t) cell line. *Blood*, 75(3):730–735, February 1990.

[25] K.-O. Cheng, N.-F. Law, W.-C. Siu, and A. W. Liew. Identification of coherent patterns in gene expression data using an efficient biclustering algorithm and parallel coordinate visualization. *BMC Bioinformatics*, 9:210+, April 2008.

[26] Y. Cheng and G. M. Church. Biclustering of expression data. *Proc Int Conf Intell Syst Mol Biol*, 8:93–103, 2000.

[27] C. S. Chim, T. K. Fung, and R. Liang. Disruption of ink4//cdk//rb cell cycle pathway by gene hypermethylation in multiple myeloma and mgus. *Leukemia*, 17(12):2533–2535, 2003.

[28] K. Christie, S. Weng, R. Balakrishnan, M. Costanzo, K. Dolinski, S. Dwight, S. Engel, B. Feierbach, D. Fisk, J. Hirschman, E. Hong, L. Issel-Tarver, R. Nash, A. Sethuraman, B. Starr, C. Theesfeld, R. Andrada, G. Binkley, Q. Dong, C. Lane, M. Schroeder, D. Botstein, and J. Cherry. Saccharomyces Genome Database (SGD) provides tools to identify and analyze sequences from Saccharomyces cerevisiae and related sequences from other organisms. *Nucleic Acids Res*, 32 Database issue:D311–4, 2004.

[29] F. R. K. Chung. *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92) (Cbms Regional Conference Series in Mathematics)*. American Mathematical Society, February 1997.

[30] H. J. Chung, M. Kim, C. H. Park, J. Kim, and J. H. Kim. Arrayxpath: mapping and visualizing microarray gene-expression data with integrated biological pathway

resources using scalable vector graphics. *Nucleic Acids Res*, 32(Web Server issue), July 2004.

[31] J. Connor, R. Bannerji, S. Saito, W. Heston, W. Fair, and E. Gilboa. Regression of bladder tumors in mice treated with interleukin 2 gene-modified tumor cells. *J Exp Med*, 177(4):1127–1134, April 1993.

[32] D. Cozma and A. Thomas-Tikhonenko. Kit-Activating Mutations in AML: Lessons from PU.1-Induced Murine Erythroleukemia. *Cancer Biol Ther*, 5(6):579–81, 2006.

[33] D. Craigon, N. James, J. Okyere, J. Higgins, J. Jotham, and S. May. NASCArrays: a repository for microarray data generated by NASC's transcriptomics service. *Nucleic Acids Res*, 32 Database issue:D575–7, 2004.

[34] R. J. Critchley-Thorne, N. Yan, S. Nacu, J. Weber, S. P. Holmes, and P. P. Lee. Down-regulation of the interferon signaling pathway in T lymphocytes from patients with metastatic melanoma. *PLoS Med*, 4(5):e176, 2007.

[35] K. D. Dahlquist, N. Salomonis, K. Vranizan, S. C. Lawlor, and B. R. Conklin. Genmapp, a new tool for viewing and analyzing microarray data on biological pathways. *Nat Genet*, 31(1):19–20, May 2002.

[36] J. Demeter, C. Beauheim, J. Gollub, T. Hernandez-Boussard, H. Jin, D. Maier, J. C. Matese, M. Nitzberg, F. Wymore, Z. K. Zachariah, P. O. Brown, G. Sherlock, and C. A. Ball. The stanford microarray database: implementation of new analysis tools and open source release of software. *Nucleic Acids Res*, 35(Database issue), January 2007.

[37] E. Deryugina and J. Quigley. Matrix metalloproteinases and tumor metastasis. *Cancer and Metastasis Reviews*, 25(1):9–34, March 2006.

[38] C. Ding, T. Li, and W. Peng. On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *Comput. Stat. Data Anal.*, 52(8):3913–3927, April 2008.

[39] S. W. Doniger, N. Salomonis, K. D. Dahlquist, K. Vranizan, S. C. Lawlor, and B. R. Conklin. Mappfinder: using gene ontology and genmapp to create a global gene-expression profile from microarray data. *Genome Biol*, 4(1), 2003.

[40] D. Donoho and V. Stodden. When does non-negative matrix factorization give a correct decomposition into parts, 2003.

[41] J. Donovan and J. Slingerland. Transforming growth factor-beta and breast cancer: Cell cycle arrest by transforming growth factor-beta and its disruption in cancer. *Breast Cancer Res*, 2(2):116–124, 2000.

[42] S. Draghici, P. Khatri, A. L. Tarca, K. Amin, A. Done, C. Voichita, C. Georgescu, and R. Romero. A systems biology approach for pathway level analysis. *Genome Res.*, pages gr.6202607+, September 2007.

[43] M. J. Duffy, T. M. Maguire, A. Hill, E. Mcdermott, and N. O'Higgins. Metalloproteinases: role in breast carcinogenesis, invasion and metastasis. *Breast cancer research*, 2(4):252–257, 2000.

[44] E. Edelman, A. Porrello, J. Guinney, B. Balakumaran, A. Bild, P. G. Febbo, and S. Mukherjee. Analysis of sample set enrichment scores: assaying the enrichment of sets of genes for individual samples in genome-wide expression profiles. *Bioinformatics*, 22(14):e108–16, 2006.

[45] S. Efroni, C. F. Schaefer, and K. H. Buetow. Identification of key processes underlying cancer phenotypes using biologic pathway analysis. *PLoS ONE*, 2(5), 2007.

[46] M. Eisen, P. Spellman, P. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA*, 95:14863–14868, 1998.

[47] C. Evans, A. G. Dalgleish, and D. Kumar. Review article: immune suppression and colorectal cancer. *Alimentary Pharmacology and Therapeutics*, 24(8):1163–1177, October 2006.

[48] S. Fields and R. Sternglanz. The two-hybrid system: an assay for protein-protein interactions. *Trends Genet*, 10(8):286–292, August 1994.

[49] J. Flannick, A. Novak, B. S. Srinivasan, H. H. McAdams, and S. Batzoglou. Graemlin: general and robust alignment of multiple large interaction networks. *Genome Res*, 16(9):1169–81, 2006.

[50] J. H. Friedman and B. E. Popescu. Gradient directed regularization for linear regression and classification. Technical report, Penn State, 2004.

[51] N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using bayesian networks to analyze expression data. *J Comput Biol*, 7(3-4):601–620, 2000.

[52] A. Fuente, N. Bing, I. Hoeschele, and P. Mendes. Discovery of meaningful associations in genomic data using partial correlation coefficients. *Bioinformatics*, 20(18):3565–3574, December 2004.

[53] Y. Gao and G. Church. Improving molecular cancer class discovery through sparse non-negative matrix factorization. *Bioinformatics*, 21(21):3970–3975, November 2005.

[54] T. S. Gardner, D. di Bernardo, D. Lorenz, and J. J. Collins. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301(5629):102–105, July 2003.

[55] A. P. Gasch, P. T. Spellman, C. M. Kao, M. B. Eisen, G. Storz, D. Botstein, and P. O. Brown. Genomic expression programs in the response of yeast cells to environmental changes. *Mol Biol Cell*, 11(12):4241–57, 2000.

[56] I. Gat-Viks and R. Shamir. Chain functions and scoring functions in genetic networks. *Bioinformatics*, 19 Suppl 1, 2003.

[57] I. Gat-Viks and R. Shamir. Refinement and expansion of signaling pathways: the osmotic response network in yeast. *Genome Res*, 17(3):358–367, March 2007.

[58] A. C. Gavin, M. Bsche, R. Krause, P. Grandi, M. Marzioch, A. Bauer, J. Schultz, J. M. Rick, A. M. Michon, C. M. Cruciat, M. Remor, C. Hfert, M. Schelder, M. Brajenovic, H. Ruffner, A. Merino, K. Klein, M. Hudak, D. Dickson, T. Rudi, V. Gnau, A. Bauch, S. Bastuck, B. Huhse, C. Leutwein, M. A. Heurtier, R. R. Copley, A. Edelmann, E. Querfurth, V. Rybin, G. Drewes, M. Raida, T. Bouwmeester, P. Bork, B. Seraphin, B. Kuster, G. Neubauer, and G. Superti-Furga. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*, 415(6868):141–147, January 2002.

[59] L. Giot, J. S. Bader, C. Brouwer, A. Chaudhuri, B. Kuang, Y. Li, Y. L. Hao, C. E. Ooi, B. Godwin, E. Vitols, G. Vijayadamodar, P. Pochart, H. Machineni, M. Welsh, Y. Kong, B. Zerhusen, R. Malcolm, Z. Varrone, A. Collis, M. Minto, S. Burgess, L. Mcdaniel, E. Stimpson, F. Spriggs, J. Williams, K. Neurath, N. Ioime, M. Agee, E. Voss, K. Furtak, R. Renzulli, N. Aanensen, S. Carrolla, E. Bickelhaupt, Y. Lazovatsky, A. Dasilva, J. Zhong, C. A. Stanyon, R. L. Finley, K. P. White, M. Braverman, T. Jarvie, S. Gold, M. Leach, J. Knight, R. A. Shimkets, M. P. Mckenna, J. Chant, and J. M. Rothberg. A protein interaction map of drosophila melanogaster. *Science*, 302(5651):1727–1736, December 2003.

[60] L. I. Gold. The role for transforming growth factor-beta (tgf-beta) in human cancer. *Critical reviews in oncogenesis*, 10(4):303–360, 1999.

[61] P. Grosu, J. P. Townsend, D. L. Hartl, and D. Cavalieri. Pathway processor: a tool for integrating whole-genome expression results into metabolic networks. *Genome Res*, 12(7):1121–1126, July 2002.

[62] J. Gu and J. Liu. Bayesian biclustering of gene expression data. *BMC Genomics*, 9(Suppl 1), 2008.

[63] D. Guillamet, J. Vitria, and B. Schiele. Introducing a weighted non-negative matrix factorization for image classification. *Pattern Recognition Letters*, 24(14):2447–2454, October 2003.

[64] Z. Guo, T. Zhang, X. Li, Q. Wang, J. Xu, H. Yu, J. Zhu, H. Wang, C. Wang, E. J. Topol, Q. Wang, and S. Rao. Towards precise classification of cancers based on robust gene functional expression profiles. *BMC Bioinformatics*, 6:58, 2005.

[65] L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray. From molecular to modular cell biology. *Nature*, 402(6761 Suppl), December 1999.

[66] A. Haugen, R. Kelley, J. Collins, C. Tucker, C. Deng, C. Afshari, J. Brown, T. Ideker, and B. Van Houten. Integrating phenotypic and expression profiles to map arsenic-response networks. *Genome Biol*, 5(12):R95, 2004.

[67] B. Hayete, T. S. Gardner, and J. J. Collins. Size matters: network inference tackles the genome scale. *Mol Syst Biol*, 3, February 2007.

[68] L. Hedges and I. Olkin. *Statistical Methods for Meta-analysis*. Academic Press, 1985.

[69] P. O. Hoyer. Non-negative matrix factorization with sparseness constraints, Aug 2004.

[70] J. Huan, D. Bandyopadhyay, W. Wang, J. Snoeyink, J. Prins, and A. Tropsha. Comparing graph representations of protein structure for mining family-specific residue-based packing motifs. *J Comput Biol*, 12(6):657–671, 2005.

[71] R. Huang, A. Wallqvist, and D. G. Covell. Targeting changes in cancer: assessing pathway stability by comparing pathway gene expression coherence levels in tumor and normal tissues. *Mol Cancer Ther*, 5(9):2417–27, 2006.

[72] Z. Hui and H. Trevor. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, April 2005.

[73] D. Hwang, A. G. Rust, S. Ramsey, J. J. Smith, D. M. Leslie, A. D. Weston, P. de Atauri, J. D. Aitchison, L. Hood, A. F. Siegel, and H. Bolouri. A data integration methodology for systems biology. *Proc Natl Acad Sci U S A*, 102(48):17296–301, 2005.

[74] T. Ideker, T. Galitski, and L. Hood. A new approach to decoding life: systems biology. *Annu Rev Genomics Hum Genet*, 2:343–72, 2001.

[75] T. Ideker and D. Lauffenburger. Building with a scaffold: emerging strategies for high- to low-level cellular modeling. *Trends Biotechnol*, 21(6):255–62, 2003.

[76] T. Ideker, O. Ozier, B. Schwikowski, and A. F. Siegel. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics*, 18 Suppl 1, 2002.

[77] A. Inokuchi, T. Washio, and H. Motoda. An apriori-based algorithm for mining frequent substructures from graph data. In *Principles of Data Mining and Knowledge Discovery*, pages 13–23, 2000.

[78] T. Ito, K. Tashiro, S. Muta, R. Ozawa, T. Chiba, M. Nishizawa, K. Yamamoto, S. Kuhara, and Y. Sakaki. Toward a protein-protein interaction map of the budding yeast: A comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. *Proc Natl Acad Sci U S A*, 97(3):1143–1147, February 2000.

[79] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264–323, September 1999.

[80] J. L. Jones, J. E. Royall, D. R. Critchley, and R. A. Walker. Modulation of myoepithelial-associated alpha6beta4 integrin in a breast cancer cell line alters invasive potential. *Exp Cell Res*, 235(2):325–333, September 1997.

[81] G. Joshi-Tope, M. Gillespie, I. Vastrik, P. D'Eustachio, E. Schmidt, B. de Bono, B. Jassal, G. Gopinath, G. Wu, L. Matthews, S. Lewis, E. Birney, and L. Stein. Reactome: a knowledgebase of biological pathways. *Nucleic Acids Res*, 33(Database issue):D428–32, 2005.

[82] A. R. Joyce and B. O. Palsson. The model organism as a system: integrating 'omics' data sets. *Nat Rev Mol Cell Biol*, 7(3):198–210, 2006.

[83] M. Kanehisa and S. Goto. Kegg: kyoto encyclopedia of genes and genomes. *Nucleic Acids Res*, 28(1):27–30, 2000.

[84] M. Kanehisa, S. Goto, M. Hattori, K. F. Aoki-Kinoshita, M. Itoh, S. Kawashima, T. Katayama, M. Araki, and M. Hirakawa. From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res*, 34(Database issue):D354–7, 2006.

[85] M. Kanehisa, S. Goto, S. Kawashima, and A. Nakaya. The kegg databases at genomenet. *Nucl. Acids Res.*, 30(1):42–46, January 2002.

[86] R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[87] T. M. Kate, L. J. Hofland, P. M. van Koetsveld, J. Jeekel, and C. H. van Eijck. Pro-inflammatory cytokines affect pancreatic carcinoma cell. endothelial cell interactions. *JOP : Journal of the pancreas*, 7(5):454–464, 2006.

[88] B. P. Kelley, B. Yuan, F. Lewitter, R. Sharan, B. R. Stockwell, and T. Ideker. Pathblast: a tool for alignment of protein interaction networks. *Nucleic Acids Res*, 32(Web Server issue), July 2004.

[89] P. Khatri and S. Drăghici. Ontological analysis of gene expression data: current tools, limitations, and open problems. *Bioinformatics*, 21(18):3587–3595, September 2005.

[90] H. Kim and H. Park. Sparse non-negative matrix factorizations via alternating non-negativity-constrained least squares for microarray data analysis. *Bioinformatics*, May 2007.

[91] H. Kitano. Systems biology: a brief overview. *Science*, 295(5560):1662–4, 2002.

[92] S. M. Kornblau, H. J. Xu, W. Zhang, S. X. Hu, M. Beran, T. L. Smith, J. Hester, E. Estey, W. F. Benedict, and A. B. Deisseroth. Levels of retinoblastoma protein expression in newly diagnosed acute myelogenous leukemia. *Blood*, 84(1):256–261, July 1994.

[93] M. Koyuturk, A. Grama, and W. Szpankowski. Pairwise Local Alignment of Protein Interaction Networks Guided by Models of Evolution. *RECOMB*, 1, 2005.

[94] M. Kuramochi and G. Karypis. An efficient algorithm for discovering frequent subgraphs. *IEEE Trans. on Knowl. and Data Eng.*, 16(9):1038–1051, September 2004.

[95] M. Kuramochi and G. Karypis. Finding frequent patterns in a large sparse graph. *Data Mining and Knowledge Discovery*, 11(3):243–271, November 2005.

[96] D. A. Lashkari, J. L. DeRisi, J. H. McCusker, A. F. Namath, C. Gentile, S. Y. Hwang, P. O. Brown, and R. W. Davis. Yeast microarrays for genome wide parallel genetic and gene expression analysis. *Proceedings of the National Academy of Sciences of the United States of America*, 94(24):13057–13062, November 1997.

[97] D. Lee and H. Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, volume 13, pages 556–562, 2000.

[98] D. D. Lee and S. H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, October 1999.

[99] H. K. Lee, A. K. Hsu, J. Sajdak, J. Qin, and P. Pavlidis. Coexpression analysis of human genes across many microarray data sets. *Genome Res*, 14(6):1085–1094, June 2004.

[100] T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, Z. Bar-Joseph, G. K. Gerber, N. M. Hannett, C. T. Harbison, C. M. Thompson, I. Simon, J. Zeitlinger, E. G. Jennings, H. L. Murray, D. B. Gordon, B. Ren, J. J. Wyrick, J.-B. Tagne, T. L. Volkert, E. Fraenkel, D. K. Gifford, and R. A. Young. Transcriptional Regulatory Networks in Saccharomyces cerevisiae. *Science*, 298(5594):799–804, 2002.

[101] B. Lehner and A. G. Fraser. A first-draft human protein-interaction map. *Genome Biol*, 5(9):R63, 2004.

[102] D. M. Levine, D. R. Haynor, J. C. Castle, S. B. Stepaniants, M. Pellegrini, M. Mao, and J. M. Johnson. Pathway and gene-set activation measurement from mRNA expression data: the tissue distribution of human pathways. *Genome Biol*, 7(10):R93, 2006.

[103] H. Li, X. Chen, K. Zhang, and T. Jiang. A general framework for biclustering gene expression data. *J Bioinform Comput Biol*, 4(4):911–933, August 2006.

[104] S. Z. Li, X. W. Hou, H. J. Zhang, and Q. S. Cheng. Learning spatially localized, parts-based representation. In *Computer Vision and Pattern Recognition*, volume 1, pages I–207–I–212 vol.1, 2001.

[105] C.-J. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural Comput.*, 19(10):2756–2779, October 2007.

[106] S. J. Liu, Y. P. Sher, C. C. Ting, K. W. Liao, C. P. Yu, and M. H. Tao. Treatment of b-cell lymphoma with chimeric igg and single-chain fv antibody-interleukin-2 fusion proteins. *Blood*, 92(6):2103–2112, September 1998.

[107] W. Liu, K. Yuan, and D. Ye. Reducing microarray data via nonnegative matrix factorization for visualization and clustering analysis. *Journal of Biomedical Informatics*, In Press, Accepted Manuscript, 2008.

[108] J. A. Loo. Studying noncovalent protein complexes by electrospray ionization mass spectrometry. *Mass Spectrometry Reviews*, 16(1):1–23, December 1998.

[109] T. J. Macdonald, K. M. Brown, B. Lafleur, K. Peterson, C. Lawlor, Y. Chen, R. J. Packer, P. Cogen, and D. A. Stephan. Expression profiling of medulloblastoma: Pdgfra and the ras/mapk pathway as therapeutic targets for metastatic disease. *Nat Genet*, 29(2):143–152, October 2001.

[110] S. C. Madeira and A. L. Oliveira. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1(1):24–45, 2004.

[111] I. A. Maraziotis, K. Dimitrakopoulou, and A. Bezerianos. Growing functional modules from a seed protein via integration of protein interaction and gene expression data. *BMC Bioinformatics*, 8:408+, October 2007.

[112] S. McCarroll, C. Murphy, S. Zou, S. Pletcher, C. Chin, Y. Jan, C. Kenyon, C. Bargmann, and H. Li. Comparing genomic expression patterns across species identifies shared transcriptional profile in aging. *Nat Genet*, 36(2):197–204, 2004.

[113] T. W. McGarvey, E. Tait, J. E. Tomaszewski, and S. B. Malkowicz. Expression of transforming growth factor-beta receptors and related cell-cycle components in transitional-cell carcinoma of the bladder. *Mol Urol*, 3(4):371–380, 1999.

[114] E. E. Medrano. Repression of tgf-beta signaling by the oncogenic protein ski in human melanomas: consequences for proliferation, survival, and metastasis. *Oncogene*, 22(20):3123–3129, May 2003.

[115] R. Milo, S. Itzkovitz, N. Kashtan, R. Levitt, S. Shen-Orr, I. Ayzenshtat, M. Sheffer, and U. Alon. Superfamilies of evolved and designed networks. *Science*, 303(5663):1538–42, 2004.

[116] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–7, 2002.

[117] G. J. Mizejewski. Role of integrins in cancer: Survey of expression patterns. *Proc Soc Exp Biol Med*, 222(2):124–138, November 1999.

[118] V. K. Mootha, C. M. Lindgren, K. F. Eriksson, A. Subramanian, S. Sihag, J. Lehar, P. Puigserver, E. Carlsson, M. Ridderstråle, E. Laurila, N. Houstis, M. J. Daly, N. Patterson, J. P. Mesirov, T. R. Golub, P. Tamayo, B. Spiegelman, E. S. Lander, J. N. Hirschhorn, D. Altshuler, and L. C. Groop. Pgc-1alpha-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nat Genet*, 34(3):267–273, July 2003.

[119] R. Mukhopadhyay, R. L. Theriault, and J. E. Price. Increased levels of alpha-6 integrins are associated with the metastatic phenotype of human breast cancer cells. *Clinical and Experimental Metastasis*, 17(4):323–330, June 1999.

[120] J. I. Murray, M. L. Whitfield, N. D. Trinklein, R. M. Myers, P. O. Brown, and D. Botstein. Diverse and specific gene expression responses to stresses in cultured human cells. *Mol Biol Cell*, 15(5):2361–74, 2004.

[121] The Netpath Database. http://www.netpath.org.

[122] J. M. Nocek, J. S. Zhou, S. D. De Forest, S. Priyadarshy, D. N. Beratan, J. N. Onuchic, and B. M. Hoffman. Theory and Practice of Electron Transfer within Protein-Protein Complexes: Application to the Multidomain Binding of Cytochrome c by Cytochrome c Peroxidase. *Chem. Rev.*, 96:2459–2489, 1996.

[123] R. Norel, F. Sheinerman, D. Petrey, and B. Honig. Electrostatic contributions to protein protein interactions: Fast energetic filters for docking and their physical basis. *Protein Science*, 10(2147-2161), 2001.

[124] P. Paatero. Least squares formulation of robust non-negative factor analysis. *Chemometrics and Intelligent Laboratory Systems*, 37, May 1997.

[125] P. Paatero and U. Tapper. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126, 1994.

[126] D. Pan, N. Sun, K. H. Cheung, Z. Guan, L. Ma, M. Holford, X. Deng, and H. Zhao. Pathmapa: a tool for displaying gene expression and performing statistical tests on metabolic pathways at multiple levels for arabidopsis. *BMC Bioinformatics*, 4, November 2003.

[127] R. Pandey, R. K. Guru, and D. W. Mount. Pathway miner: extracting gene association networks from molecular pathways for predicting the biological significance of gene expression microarray data. *Bioinformatics*, 20(13):2156–2158, September 2004.

[128] F. G. Perabo and S. C. Müller. Current and new strategies in immunotherapy for superficial bladder cancer. *Urology*, 64(3):409–421, September 2004.

[129] S. Peri, J. Navarro, R. Amanchy, T. Kristiansen, C. Jonnalagadda, V. Surendranath, V. Niranjan, B. Muthusamy, T. Gandhi, M. Gronborg, N. Ibarrola, N. Deshpande, K. Shanker, H. Shivashankar, B. Rashmi, M. Ramya, Z. Zhao, K. Chandrika, N. Padma, H. Harsha, A. Yatish, M. Kavitha, M. Menezes, D. Choudhury, S. Suresh, N. Ghosh, R. Saravana, S. Chandran, S. Krishna, M. Joy, S. Anand, V. Madavan, A. Joseph, G. Wong, W. Schiemann, S. Constantinescu, L. Huang, R. Khosravi-Far, H. Steen, M. Tewari, S. Ghaffari, G. Blobe, C. Dang, J. Garcia, J. Pevsner, O. Jensen, P. Roepstorff, K. Deshpande, A. Chinnaiyan, A. Hamosh, A. Chakravarti, and A. Pandey. Development of human protein reference database as an initial platform for approaching systems biology in humans. *Genome Res*, 13(10):2363–71, 2003.

[130] B. Poch, E. Lotspeich, M. Ramadani, S. Gansauge, H. Beger, and F. Gansauge. Systemic immune dysfunction in pancreatic cancer patients. *Langenbeck's Archives of Surgery*, 392(3):353–358, May 2007.

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

[132] A. K. Ramani, R. C. Bunescu, R. J. Mooney, and E. M. Marcotte. Consolidating the set of known human protein-protein interactions in preparation for large-scale mapping of the human interactome. *Genome Biol*, 6(5):R40, 2005.

[133] S. Ramaswamy, P. Tamayo, R. Rifkin, S. Mukherjee, C. H. Yeang, M. Angelo, C. Ladd, M. Reich, E. Latulippe, J. P. Mesirov, T. Poggio, W. Gerald, M. Loda, E. S. Lander, and T. R. Golub. Multiclass cancer diagnosis using tumor gene expression signatures. *Proc Natl Acad Sci U S A*, 98(26):15149–54, 2001.

[134] L. J. Ransone. Detection of protein-protein interactions by coimmunoprecipitation and dimerization. *Methods Enzymol*, 254:491–497, 1995.

[135] J. S. Reis-Filho, F. Milanezi, S. Carvalho, P. T. Simpson, D. Steele, K. Savage, M. B. Lambros, E. M. Pereira, J. M. Nesland, S. R. Lakhani, and F. C. Schmitt. Metaplastic breast carcinomas exhibit egfr, but not her2, gene amplification and overexpression: immunohistochemical and chromogenic in situ hybridization analysis. *Breast Cancer Res*, 7(6), 2005.

[136] D. Reiss, I. Avila-Campillo, V. Thorsson, B. Schwikowski, and T. Galitski. Tools enabling the elucidation of molecular pathways active in human disease: application to Hepatitis C virus infection. *BMC Bioinformatics*, 6(1):154, 2005.

[137] M. Reuss-Borst, H. Buhring, H. Schmidt, and C. Muller. AML: immunophenotypic heterogeneity and prognostic significance of c-kit expression. *Leukemia*, 8(2):258–63, 1994.

[138] J. Rual, K. Venkatesan, T. Hao, T. Hirozane-Kishikawa, A. Dricot, N. Li, G. Berriz, F. Gibbons, M. Dreze, N. Ayivi-Guedehoussou, N. Klitgord, C. Simon, M. Boxem, S. Milstein, J. Rosenberg, D. Goldberg, L. Zhang, S. Wong, G. Franklin, S. Li, J. Albala, J. Lim, C. Fraughton, E. Llamosas, S. Cevik, C. Bex, P. Lamesch, R. Sikorski, J. Vandenhaute, H. Zoghbi, A. Smolyar, S. Bosak, R. Sequerra, L. Doucette-Stamm, M. Cusick, D. Hill, F. Roth, and M. Vidal. Towards a proteome-scale map of the human protein-protein interaction network. *Nature*, 437(7062):1173–8, 2005.

[139] U. Rückert and S. Kramer. Generalized version space trees. In J. F. Boulicaut and S. Dzeroski, editors, *KDID*, pages 119–129, 2003.

[140] K. Sachs, D. Gifford, T. Jaakkola, P. Sorger, and D. A. Lauffenburger. Bayesian network approach to cell signaling pathway modeling. *Sci STKE*, 2002(148), September 2002.

[141] 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, October 1995.

[142] S. Schnittger, T. M. Kohl, T. Haferlach, W. Kern, W. Hiddemann, K. Spiekermann, and C. Schoch. KIT-D816 mutations in AML1-ETO-positive AML are associated with impaired event-free and overall survival. *Blood*, 107(5):1791–9, 2006.

[143] S. Schwartz, A. Heinecke, M. Zimmermann, U. Creutzig, C. Schoch, J. Harbott, C. Fonatsch, H. Loffler, T. Buchner, W. D. Ludwig, and E. Thiel. Expression of the C-kit receptor (CD117) is a feature of almost all subtypes of de novo acute myeloblastic leukemia (AML), including cytogenetically good-risk AML, and lacks prognostic significance. *Leuk Lymphoma*, 34(1-2):85–94, 1999.

[144] E. Segal, N. Friedman, D. Koller, and A. Regev. A module map showing conditional activity of expression modules in cancer. *Nat Genet*, 36(10):1090–8, 2004.

[145] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nat Genet*, 34(2):166–176, June 2003.

[146] E. Segal, H. Wang, and D. Koller. Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics*, 19 Suppl 1:I264–I272, 2003.

[147] G. Selvaggi, S. Novello, V. Torri, E. Leonardo, P. De Giuli, P. Borasio, C. Mossetti, F. Ardissone, P. Lausi, and G. V. Scagliotti. Epidermal growth factor receptor overexpression correlates with a poor prognosis in completely resected non-small-cell lung cancer. *Ann Oncol*, 15(1):28–32, January 2004.

[148] S. R. Setlur, T. E. Royce, A. Sboner, J. M. Mosquera, F. Demichelis, M. D. Hofer, K. D. Mertz, M. Gerstein, and M. A. Rubin. Integrative microarray analysis of pathways dysregulated in metastatic prostate cancer. *Cancer Res*, 67(21):10296–10303, November 2007.

[149] F. Shahnaz, M. W. Berry, Pauca, and R. J. Plemmons. Document clustering using nonnegative matrix factorization. *Information Processing & Management*, 42(2):373–386, March 2006.

[150] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res*, 13(11):2498–2504, November 2003.

[151] R. Sharan, T. Ideker, B. P. Kelley, R. Shamir, and R. M. Karp. Identification of protein complexes by comparative analysis of yeast and bacterial protein interaction data. In *RECOMB '04: Proceedings of the eighth annual international conference on Computational molecular biology*, pages 282–289, New York, NY, USA, 2004. ACM Press.

[152] R. Sharan and R. Shamir. *Current Topics in Computational Biology*, chapter Algorithmic Approaches to Clustering Gene Expression Data, pages 269–299. MIT Press, 2002.

[153] R. Sharan, S. Suthram, R. M. Kelley, T. Kuhn, S. Mccuine, P. Uetz, T. Sittler, R. M. Karp, and T. Ideker. From the cover: Conserved patterns of protein interaction in multiple species. *Proc Natl Acad Sci U S A*, 102(6):1974–1979, February 2005.

[154] S. S. Shen-Orr, R. Milo, S. Mangan, and U. Alon. Network motifs in the transcriptional regulation network of escherichia coli. *Nat Genet*, 31(1):64–8, 2002.

[155] T. Shlomi, D. Segal, E. Ruppin, and R. Sharan. QPath: a method for querying pathways in a protein-protein interaction network. *BMC Bioinformatics*, 7:199, 2006.

[156] P. Smaragdis and J. C. Brown. Non-negative matrix factorization for polyphonic music transcription. In *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, pages 177–180, 2003.

[157] V. Spirin and L. A. Mirny. Protein complexes and functional modules in molecular networks. *Proc Natl Acad Sci U S A*, 100(21):12123–8, 2003.

[158] C. Stark, B. J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers. Biogrid: a general repository for interaction datasets. *Nucleic Acids Res*, 34(Database issue), January 2006.

[159] U. Stelzl, U. Worm, M. Lalowski, C. Haenig, F. Brembeck, H. Goehler, M. Stroedicke, M. Zenkner, A. Schoenherr, S. Koeppen, J. Timm, S. Mintzlaff, C. Abraham, N. Bock, S. Kietzmann, A. Goedde, E. Toksoz, A. Droege, S. Krobitsch, B. Korn, W. Birchmeier, H. Lehrach, and E. Wanker. A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–68, 2005.

[160] J. M. Stuart, E. Segal, D. Koller, and S. K. Kim. A gene-coexpression network for global discovery of conserved genetic modules. *Science*, 302(5643):249–55, 2003.

[161] M. P. Stumpf, T. Thorne, E. de Silva, R. Stewart, H. J. An, M. Lappe, and C. Wiuf. Estimating the size of the human interactome. *Proceedings of the National Academy of Sciences*, 105(19):6959–6964, May 2008.

[162] A. Subramanian, P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander, and J. P. Mesirov. From the cover: Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *PNAS*, 102(43):15545–15550, October 2005.

[163] J. Sun, Y. Xie, H. Zhang, and C. Faloutsos. Less is more: Sparse graph mining with compact matrix decomposition. *Stat. Anal. Data Min.*, 1(1):6–22, February 2008.

[164] S. Suthram, T. Sittler, and T. Ideker. The plasmodium protein network diverges from those of other eukaryotes. *Nature*, 438(7064):108–112, November 2005.

[165] P. Tamayo, D. Scanfeld, B. L. Ebert, M. A. Gillette, C. W. Roberts, and J. P. Mesirov. Metagene projection for cross-platform, cross-species characterization of global transcriptional states. *Proc Natl Acad Sci U S A*, 104(14):5959–5964, April 2007.

[166] A. Tanay, R. Sharan, M. Kupiec, and R. Shamir. Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data. *Proc Natl Acad Sci U S A*, 101(9):2981–2986, March 2004.

[167] A. Tanay, R. Sharan, and R. Shamir. Discovering statistically significant biclusters in gene expression data. *Bioinformatics*, 18 Suppl 1, 2002.

[168] A. Tanay, R. Sharan, and R. Shamir. *Handbook of Computational Molecular Biology*, chapter Biclustering Algorithms: A Survey, pages 26–1. CRC Press, 2006.

[169] L. Tian, S. A. Greenberg, S. W. Kong, J. Altschuler, I. S. Kohane, and P. J. Park. Discovering statistically significant pathways in expression profiling studies. *Proc Natl Acad Sci U S A*, 102(38):13544–13549, September 2005.

[170] R. Tibshirani, T. Hastie, M. Eisen, D. Ross, D. Botstein, and P. Brown. Clustering methods for the analysis of DNA microarray data. Technical report, Department of Statistics, Stanford University, 1999. Available at http://www-stat.stanford.edu/ tibs/ftp/jcgs.ps.Z.

[171] Y. Tohsato, H. Matsuda, and A. Hashimoto. A multiple alignment algorithm for metabolic pathway analysis using enzyme hierarchy. *Proc Int Conf Intell Syst Mol Biol*, 8:376–383, 2000.

[172] P. Uetz, L. Giot, G. Cagney, T. A. Mansfield, R. S. Judson, J. R. Knight, D. Lockshon, V. Narayan, M. Srinivasan, P. Pochart, A. Qureshi-Emili, Y. Li, B. Godwin, D. Conover, T. Kalbfleisch, G. Vijayadamodar, M. Yang, M. Johnston, S. Fields, and J. M. Rothberg. A comprehensive analysis of protein-protein interactions in saccharomyces cerevisiae. *Nature*, 403(6770):623–627, February 2000.

[173] I. Ulitsky and R. Shamir. Identification of functional modules using network topology and high-throughput data. *BMC Systems Biology*, 1(1), 2007.

[174] A. P. van der Meijden. Non-specific immunotherapy with bacille calmette-guérin (bcg). *Clin Exp Immunol*, 123(2):179–180, February 2001.

[175] W. M. van Grevenstein, L. J. Hofland, J. Jeekel, and C. H. van Eijck. The expression of adhesion molecules and the influence of inflammatory cytokines on the adhesion of human pancreatic carcinoma cells to mesothelial monolayers. *Pancreas*, 32(4):396–402, May 2006.

[176] V. van Noort, B. Snel, and M. Huynen. Predicting gene function by conserved co-expression. *Trends Genet*, 19(5):238–42, 2003.

[177] I. Vastrik, P. D'Eustachio, E. Schmidt, G. Joshi-Tope, G. Gopinath, D. Croft, B. de Bono, M. Gillespie, B. Jassal, S. Lewis, L. Matthews, G. Wu, E. Birney, and L. Stein. Reactome: a knowledgebase of biological pathways and processes. *Genome Biology*, 8:R39+, March 2007.

[178] B. Vogelstein and K. W. Kinzler. Cancer genes and the pathways they control. *Nat Med*, 10(8):789–799, August 2004.

[179] A. J. Walhout and M. Vidal. High-throughput yeast two-hybrid assays for large-scale protein interaction mapping. *Methods*, 24(3):297–306, July 2001.

[180] J. W. Watters and C. J. Roberts. Developing gene expression signatures of pathway deregulation in tumors. *Mol Cancer Ther*, 5(10):2444–2449, October 2006.

[181] R. Wen, Y. Chen, L. Bai, G. Fu, J. Schuman, X. Dai, H. Zeng, C. Yang, R. P. Stephan, J. L. Cleveland, and D. Wang. Essential role of phospholipase c gamma 2 in early b-cell development and myc-mediated lymphomagenesis. *Mol Cell Biol*, 26(24):9364–9376, December 2006.

[182] M. L. Whitfield, G. Sherlock, A. J. Saldanha, J. I. Murray, C. A. Ball, K. E. Alexander, J. C. Matese, C. M. Perou, M. M. Hurt, P. O. Brown, and D. Botstein. Identification of genes periodically expressed in the human cell cycle and their expression in tumors. *Mol Biol Cell*, 13(6):1977–2000, 2002.

[183] G. A. Wilkin and X. Huang. A practical comparison of two k-means clustering algorithms. *BMC bioinformatics*, 9 Suppl 6, 2008.

[184] 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, 2005.

[185] F. X. Wu. Genetic weighted k-means algorithm for clustering large-scale gene expression data. *BMC bioinformatics*, 9 Suppl 6, 2008.

[186] I. Xenarios, D. W. Rice, L. Salwinski, M. K. Baron, E. M. Marcotte, and D. Eisenberg. DIP: the database of interacting proteins. *Nucleic Acids Res*, 28(1):289–291, January 2000.

[187] D. Xu, C. J. Tsai, and R. Nussinov. Hydrogen bonds and salt bridges across protein protein interfaces. *Protein Engineering*, 10(9):999–1012, 1997.

[188] W. Xu, X. Liu, and Y. Gong. Document clustering based on non-negative matrix factorization. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 267–273, New York, NY, USA, 2003. ACM Press.

[189] X. Yan, H. Cheng, J. Han, and P. S. Yu. Mining significant graph patterns by leap search. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 433–444, New York, NY, USA, 2008. ACM.

[190] X. Yan and J. Han. gSpan: Graph-based substructure pattern mining, 2002.

[191] X. Yan, M. R. Mehan, Y. Huang, M. S. Waterman, P. S. Yu, and X. J. Zhou. A graph-based approach to systematically reconstruct human transcriptional regulatory modules. *Bioinformatics*, 23(13), July 2007.

[192] X. Yan, J. X. Zhou, and J. Han. Mining closed relational graphs with connectivity constraints. In *KDD '05: Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 324–333, New York, NY, USA, 2005. ACM Press.

[193] E. Yeger-Lotem, S. Sattath, N. Kashtan, S. Itzkovitz, R. Milo, R. Pinter, U. Alon, and H. Margalit. Network motifs in integrated cellular networks of transcription-regulation and protein-protein interaction. *Proc Natl Acad Sci U S A*, 101(16):5934–9, 2004.

[194] M. K. Yeung, J. Tegnér, and J. J. Collins. Reverse engineering gene networks using singular value decomposition and robust regression. *Proc Natl Acad Sci U S A*, 99(9):6163–6168, April 2002.

[195] M. J. Zaki and C.-J. Hsiao. CHARM: An efficient algorithm for closed itemset mining. In *SIAM International Conference on Data Mining*, pages 457–473, 2002.

[196] H. S. Zhang, A. A. Postigo, and D. C. Dean. Active transcriptional repression by the Rb-E2F complex mediates G1 arrest triggered by p16INK4a, TGFbeta, and contact inhibition. *Cell*, 97(1):53–61, 1999.

[197] L. V. Zhang, O. D. King, S. L. Wong, D. S. Goldberg, A. H. Tong, G. Lesage, B. Andrews, H. Bussey, C. Boone, and F. P. Roth. Motifs, themes and thematic maps of an integrated Saccharomyces cerevisiae interaction network. *J Biol*, 4(2):6, 2005.

[198] B. L. Ziober, C.-S. Lin, and R. H. Kramer. Laminin-binding integrins in tumor progression and metastasis. *Seminars in Cancer Biology*, 7(3):119–128, June 1996.