AN INTEGRATED, MODULE-BASED BIOMARKER DISCOVERY FRAMEWORK

by

Grace T. Huang

B.S., Cornell University, 2004

Submitted to the Graduate Faculty of

School of Medicine in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2013

UNIVERSITY OF PITTSBURGH

SCHOOL OF MEDICINE

This dissertation was presented

by

Grace T. Huang

It was defended on

September 30, 2013

and approved by

Takis V. Benos*, Ph.D. Dept. of Computational Biology, University of Pittsburgh

Chakra S. Chennubhotla, Ph.D. Dept. of Computational Biology, University of Pittsburgh

Ziv Bar-Joseph, Ph.D. School of Computer Science, Carnegie Mellon University

Naftali Kaminski, M.D. School of Medicine, Yale University

Ioannis Tsamardinos, Ph.D. Dept. of Computer Science, University of Crete

Thesis Director: Gregory F. Cooper, M.D. Ph.D. Dept. of Biomedical Informatics,

University of Pittsburgh

AN INTEGRATED, MODULE-BASED BIOMARKER DISCOVERY FRAMEWORK

Grace T. Huang

University of Pittsburgh, 2013

Identification of biomarkers that contribute to complex human disorders is a principal and challenging task in computational biology. Prognostic biomarkers are useful for risk assessment of disease progression and patient stratification. Since treatment plans often hinge on patient stratification, better disease subtyping has the potential to significantly improve survival for patients. Additionally, a thorough understanding of the roles of biomarkers in cancer pathways facilitates insights into complex disease formation, and provides potential druggable targets in the pathways.

Many statistical methods have been applied toward biomarker discovery, often combining feature selection with classification methods. Traditional approaches are mainly concerned with statistical significance and fail to consider the clinical relevance of the selected biomarkers. Two additional problems impede meaningful biomarker discovery: gene multiplicity (several maximally predictive solutions exist) and instability (inconsistent gene sets from different experiments or cross validation runs).

Motivated by a need for more biologically informed, stable biomarker discovery method, I introduce an integrated module-based biomarker discovery framework for analyzing high-throughput genomic disease data. The proposed framework addresses the aforementioned challenges in three components. First, a recursive spectral clustering algorithm specifically

tailored toward high-dimensional, heterogeneous data (ReKS) is developed to partition genes into clusters that are treated as single entities for subsequent analysis. Next, the problems of gene multiplicity and instability are addressed through a group variable selection algorithm (T-ReCS) based on local causal discovery methods. Guided by the tree-like partition created from the clustering algorithm, this algorithm selects gene clusters that are predictive of a clinical outcome. We demonstrate that the group feature selection method facilitate the discovery of biologically relevant genes through their association with a statistically predictive driver. Finally, we elucidate the biological relevance of the biomarkers by leveraging available prior information to identify regulatory relationships between genes and between clusters, and deliver the information in the form of a user-friendly web server, mirConnX.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

I am indebted to many whose help made this dissertation possible.

First and foremost, I am sincerely grateful for the guidance and support of my thesis advisor, Dr. Takis Benos. His encouragements and mentorship throughout the years had helped me grow into who I am today. I deeply appreciate Dr. Chakra Chennubhotla, for the numerous stimulating discussions we had that evolved into an integral part of this work, and whose enthusiasm inspires me greatly. I am also hugely grateful for the guidance of Dr. Ioannis Tsamardinos. A conversation with him some years back led to the conception of this work, and the fruitful discussions throughout the years shaped the work into what it is today. In addition, without the critical comments and feedback of the rest of my committee members Dr. Greg Cooper, Dr. Ziv Bar-Joseph, and Dr. Naftali Kaminski, this dissertation would not be complete.

I would like to thank Dr. Abha Bais and Dr. Seyoung Kim for helpful discussions and mentorship, Harry for all his technical support, and all the rest of the past and current Benos lab members: Rachel, Sabah, Claudia, AJ, and Kristina, for discussions and critiques. I would like to also thank all the individuals in the Kaminski lab, the Tawbi lab, the Tseng lab, the TCGA group, and the CMPI group with whom I have had the pleasure to collaborate with along the way. I am thankful for the friendship of Justin, John, Cordelia, Anindita, Virginia, Andrej, Jacob, Luis, Ming, Guy and Marcel and many more, whose love and support helped carry me through. Finally, this work would not be possible without the love and support of my family.

1.0 INTRODUCTION

1.1 PERSONALIZED MEDECINE AND BIOMARKER DISCOVERY

Human diseases such as cancer have been shown to be complex and heterogeneous [1]. They often exhibit diverse morphologies, molecular characteristics, and clinical properties. Traditionally, a uniform drug regimen is administered to patients displaying similar pathology. However, these patients often vary in clinical outcome and responsiveness to drug therapy. This one-size-fits-all approach is suboptimal and often ineffective. As such, the biomedical community has recognized the need for individualized therapy, and considerable research effort have been directed toward the development of *personalized medicine*.

Personalized medicine holds one of the greatest promises of modern clinical medicine. The term is coined for customized medical decisions or drug products tailored for individual patients, often based on emerging technology or diagnostic tools not previously available. In reality, personalized medicine is still at its infancy. Treatment plans are not yet routinely devised at the granularity of individual patients, but instead at the level of patient cohorts. Since proper selection of treatment plans hinges on definition of patient cohorts, accurate diagnosis and patient stratification has the potential to significantly improve patient outcome, survival, and quality of life [2].

Disease diagnosis and subtyping can be achieved by detecting the presence or abundance of *biomarkers*. We define a biomarker to be an entity that can be measured to provide actionable information regarding biological processes, disease progression, or responses to therapeutic intervention. Additionally, biomarkers can be used to measure progress and therapeutic response of disease after a treatment plan [3]. An ideal biomarker is one that is present or absent only in diseased patients versus healthy controls, or one whose relative abundance differ among subtypes of diseases.

Identifying such biomarkers for complex human disorders is a principal and challenging task. Not only are prognostic biomarkers useful in assessing risk of disease progression and stratifying patients, a thorough understanding of the roles of biomarkers in cancer pathways facilitates insights into complex disease formation, and provides potential druggable targets in the pathways. The latter is especially crucial, as progress has yet to be achieved in improving survival for several common cancers such as lung or colon cancers. Thus, novel therapeutic strategies based on a deeper understanding of the cellular and molecular mechanisms of disease formation are urgently needed [1].

An example of a traditional biomarker for complex human disease is immunohistochemical (IHC) panel including estrogen receptor (ER), progesterone receptor (PR), and human epidermal growth factor receptor 2 (HERs). This set of single biomarkers has been used for several years in various aspects of breast cancer management. The presence of absence of these markers identifies three main subtypes [4]: luminal (ER+ and/or PR+), HER2-like (mainly ER- and HER2+), and basal-like (ER-, PR-, HER2, or triple-negative described by Schneider *et al.*[5]. They vary in their prevalence and prognosis, with triple negative patients facing the worst prognosis. These biomarkers have enabled physicians to make informative decision on cancer treatment. However, the prognostic ability of these markers is not ideal and difficulties still exist in stratifying subtypes of patients [6].

Possibilities of novel biomarkers have been proposed in light of the advance of microarray, sequencing technologies, and mass spectrometry technologies. These new high-throughput genomic technologies have facilitated the identification of *–omics* based biomarkers such as gene expression profiles or *molecular signatures* composed of several dozen to several thousand genes [7]–[9], several of which have been made commercially available (Table 1.1). In one of the pioneering studies, a molecular signature among breast cancer patients was used to predict recurrence of cancer after surgery with significant accuracy[10]. This diagnostic test is helpful in guiding the decision of whether the patients require continuing chemotherapy. A very recent study [11] also demonstrated the potential of molecular subtyping to guide therapy in early-stage, invasive breast cancer. In this study, Gluck and colleagues performed molecular subtyping of early-stage breast cancer with the MammaPrint [12] and BluePrint [13] tests to identify a group of patients, Luminal A, who do not benefit from neoadjuvant (preoperative) chemotherapy and show a high five-year metastases-free survival rate. This group could not be identified using traditional clinical tests such as immunohistochemistry and fluorescence in situ hybridization (IHC/FISH). We list some of the current commercially available molecular diagnostic tests in Table 1.1. The 70-gene assay (MammaPrint, Agendia, Netherlands) and the 21-gene assay (Oncotype DX, Genomic Health, USA) are the most widely used breast cancer multigene classifier assays. A 50-gene assay (PAM50, NanoString, USA) has also shown promise. Excellent summaries and review can be found in [14].

Encouraged by the recent success of molecular diagnostics, we developed a molecular biomarker discovery framework that could be used as the first step toward disease subtyping. In the next sections, I provide the biological background necessary for further discussion, including the measurements and regulation of gene expression.

Table 1.1 A list of commercially available molecular diagnostic tests

Test name	Platform	Features	References
MammaPrint	Microarray/Agendia BV	70-gene signature; categorizes good/poor prognosis	[12], [15]
Oncotype DX	qRT-PCR/Genomics Health	21-gene signature; recurrence score predicts likelihood of recurrence in 10 years	[10], [16]
PAM50	qRT-PCR/NanoString	50-gene assay; risk of relapse and likelihood of relapse	[17]

1.2 BIOLOGICAL BACKGROUND

1.2.1 Gene expression and microarray

The abundance of mRNA products is called *gene expression*. Gene expression measurement is a popular and cheap way to infer the state of cells under a given condition. A genome wide measurement of transcription is called an *expression profile* and reflects the transcription level of the genes in the particular condition in which they are extracted from. While in general the gene expression measurement is directly correlated with the level at which the genes are being transcribed, they do not reflect other aspects of the biological processes such as protein levels and their activity. However, mRNA levels are easier to measure than protein levels, thus we use gene expression level as a reasonable substitute. As mentioned in the previous section, the availability of microarray technology has also enabled the search for molecular biomarkers.

In the last two decades, technical advances have lead to development of gene expression microarrays. Expression microarrays have enabled researchers to measure the abundance of thousands of mRNA targets simultaneously [18], [19], providing a genomic, holistic view of gene expression. Microarray technology is based on hybridization: a process in which a strand binds to its unique complementary strand. On a microarray, a set of probes is attached to a solid surface (chip). A sample containing fluorescently tagged sequences are allowed to interact with the probes, and based on intensity of the fluorescence, the (relative) abundance of the targets of interests can be determined. There are two broad categories of microarrays: two-channel and single channel. Two-channel microarrays allow two individually labeled samples to hybridize competitively on the same surface to determine the relative abundance of the genes between the two samples. In contrast, one-color microarray only one sample is used. An illustration of the two-channel microarray is shown in Figure 1.1.

The biomedical community has witnessed an exponential growth in gene expression profiled from clinical samples, and several large consortiums [20]–[22] in addition to the Gene Expression Omnibus(GEO)[23] have systematically collected gene expression measurements from patients. Clinical gene expression data that we work with in this dissertation are generally extracted from biopsies of tumor samples and healthy tissues taken from patients or volunteers that participate in the studies. The gene expression profiles are typically measured on the same platforms across samples and common pre-processing steps including background correction, imputation, and normalization are performed together.

Gene expression is a tightly regulated process. The expression of a given gene at any given time depends on a complicated series of feedback and regulation that are controlled by many factors, including chromatin states, methylation status, transcription factor binding and suppression by a family of small RNAs. In the next sections, we focus on the last two types of regulation.

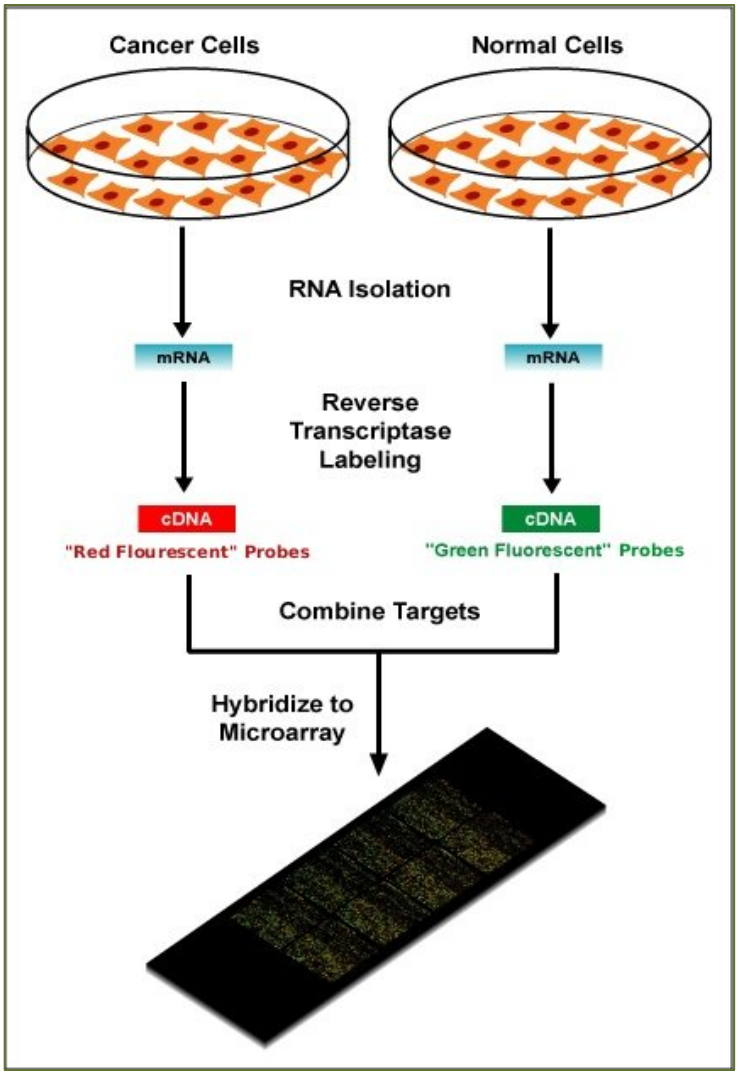


Figure 1.1 Illustration of a two-color microarray experiment
http://upload.wikimedia.org/wikipedia/en/c/c8/Microarray-schema.jpg

1.2.2 Transcriptional regulation

In transcriptional regulation, proteins known as transcription factors (TFs) are recruited by a set of protein complexes and Pol II [24] to bind a promoter region of protein coding genes to either initiate(activators) or block(repressors) the activation of the gene. The core promoter region typically consists of several hundred base pairs surrounding the transcription start site of a gene, and encompass a wide range of characteristics such as the presence of CpG islands, TATA box, methylation, and various other sequence elements [25], [26]. The regulatory region outside of the core promoter can be bound by TFs that typically contain a DNA binding domain that binds to specific set of sequence motifs 6-15bp in length [27]. There are over 2000 TFs that can be organized into families based on their structural properties and corresponding binding motifs [28], [29]. The TF can regulate its targets alone, in conjunction with other co-regulators, or in competition [30]. Identification of transcription factor bindings sites (TFBS) has been a major topic of interests for many years in the computational biology community. The short and degenerate nature of the motifs makes it a challenging task to identify them in the genomic region. Nevertheless, a number of tools and publications have resulted, often taking advantage of information such as conservation across species, presence in promoters of co-regulated genes, in addition to sequence specificity [31]–[33]. Several databases additionally contain curated experimental information that further supports the regulatory relationships of TFs and targets [34]–[36].

TFs are known to participate in cancer and disease formation. A large number of transcription factors involved in cell differentiation and apoptosis have been identified over the years [37]–[40], perhaps most famous of which is p53 [41], a tumor-suppressor gene, whose inactivation is one of the key hallmarks of a tumor.

1.2.3 Post-transcriptional regulation

First identified in 1993 by Lee *et al.* [42], microRNAs(miRNAs) are small (20-23 nt), non-coding single stranded RNA molecules that play an important role in post-transcriptional regulation of protein-coding genes. The regulations are post-*transcriptional*, as their regulatory event occurs after mRNAs have been transcribed, by binding to the target sites of the 3'untranslated regions of protein coding genes. miRNAs are believed to be mostly transcribed by RNA Polymerase II [43] and less frequently, RNA Polymerase III [44]. The initial full-length miRNA forms a hairpin structure, which is then processed by two proteins, *Drosha* and *Dicer*, to form the final ~22nt product associated with a protein complex containing *Argonaut*. The miRNAs bind to the target sites on the 3'UTRs of target genes through base complementarity [45]. This binding can either lead to full degradation of the target mRNA transcript, or the blocking of its translation. Exact mechanisms for both are still under investigation, and current evidence seems to support both forms of suppression. Each miRNA can target many mRNAs, and each mRNA can in turn be the targets of multiple miRNAs [46].

Many miRNA target prediction algorithms have been developed [47]–[49]. In general, they combine sequence information, energy calculations, and various sequence contextual information such as position and nucleotide compositions, as well as conservation across species to infer possible binding sites of a given miRNA. A comprehensive review and discussion of various target prediction methods can be found in [50].

miRNAs have been found in all animal lineages, and have been implicated as critical regulators during disease formation and tumorgenesis [51]. In this dissertation, we are interested in both the abundance of miRNAs in disease samples, as well as the target genes and potential oncogenes that they regulate. There is mounting evidence that miRNAs can be useful for cancer prognosis. miRNA expression profiles for different tumor subtypes are unique due to tissue specificity. Using miRNA profiles, Lu et al. were able to correctly classify 12 of 17 poorly differentiated carcinomas [52]. The Table 1.2 highlights some of the miRNAs and targets found to be associated with tumors.

Table 1.2 Tumor-associated microRNAs and their validated target genes [51]
NSCLC: Non-small cell lung cancer; CLL: chronic lymphocytic leukemia; GBM:Glioblastoma multiforme.

miRs	Tumor Type	Expression	Target Genes
let-7	NSCLC	Down	RAS
miR-15a,miR-16	CLL	Down	BCL2
miR17-92 polycistron	Breast, B-cell lymphomas	Up	AIB1,E2F1,TGFBR2,Tspi,CTGF
miR-21	Breast, GBM	Up	TPM1
miR-106a	Colon, pancreas, prostate	Up	TPM1
miR-221-222,miR-146b	Tyroid, papillary	Up	KIT
miR-372-373	Testis, germ cell tumors	Up	LATS2

Having introduced the motivation behind biomarker discovery and associated introductory concepts in biology, we now turn to the computational aspects of biomarker discovery and discuss some of its current limitations.

1.3 LIMITATIONS OF CURRENT BIOMARKER DISCOVERY METHODS

We aim to develop biomarker discovery methods that could be used as the first step toward disease subtyping. From a statistical perspective, biomarker discovery can be best cast as a variable selection problem, and identification of cancer subtype can be viewed as the associated classification step. The variables under selection are the molecular attributes of interest, in our case genes, genetic variations, or metabolites; the observations are samples from which the variables are measured e.g. patients. The goal is to search for the most discriminating features with respect to the labels for the observations.

Aside from its usefulness in extracting biological information, variable selection is critical in our application from a computational perspective. High-throughput genomics data is high dimensional, often with tens of thousands of variables measured simultaneously. However, the sample size is severely limited compared to the size of the variables. This is known as a phenomenon called curse-of-dimensionality [53], where the dimension of the variable space increases so fast that the available data becomes extremely sparse in this space. This sparsity is problematic for many methods that require statistical significance [54], [55]. Dimensionality reduction methods or variable selection are often performed as the first steps in analysis of omics data.

Many variable selection methods have been applied toward biomarker discovery using omics data. A review of the existing variable selection methods can be found in Section 3.1. Even though numerous computational methods have been proposed for this purpose, clinical adoption of these biomarker discovery methods have been slow and limited due to a lack of reproducibility of the results. We detail several computational challenges and sources of non-reproducibility in biomarker discovery for omics data.

- **Heterogeneity**

Cancer is highly heterogeneous with respect to molecular alterations, cellular compositions, and clinical outcome [56]. This creates a principal challenge in biomarker discovery. Individual tumors are defined by distinct molecular changes and mechanisms. Further complicating the picture is the fact that tumors have a complex tissue structure comprised of malignant cells, tumor stromal components, host cells, and adjacent normal tissues. This molecular heterogeneity, along with the complex micro environment in which the tumor resides, makes analysis of high-throughput measurements taken from pooled samples of tumor a very challenging task.

Statistically, heterogeneity presents us with the problem of high level of noise. For example, we may not see perfect differential expression between normal tissues or patient samples, even if stratified with the most discriminative predictor.

- **Multicollinearity (correlation)**

Another intricate challenge in omics data variable selection is that cellular processes are often coupled and synchronized due to internal cellular regulation or external signals and stimulations. Variables of interest can display similar behavior. For example, many genes regulated by the same activator/repressor or whose protein products physically interacting with each other would display similar expression patterns across different conditions or across time points. This results in a correlation structure among the variables. This correlation structure would break down the assumptions of a lot of traditional variable selection methods designed for uncorrelated data [57], rendering them unsuitable for the task of biomarker discovery.

- **Multiplicity and Instability**

Two other problems that impede meaningful biomarker discovery are: **gene multiplicity** and **instability**. Gene multiplicity alludes to the fact that several maximally predictive solutions (gene sets) can co-exist [58], [59]. This may be due to the multicollinearity problem alluded to previously, but coexisting maximally predictive solutions may not necessarily be correlated. Instability refers to the phenomenon that inconsistent gene sets are selected from different research groups, different experiments conducted in the same lab, or even among different subsets of the data [60]. Many existing variable selection algorithms are designed with no regard to stability, as they seek to optimize only the predictive performance. These two issues are tightly coupled with the problem of multicollinearity, heterogeneity, and the high dimension of the data relative to sample size, and are possibly the main contributors to a lack of reproducibility in biomarker discovery.

In addition to these computational challenges, two additional properties are often overlooked by biomarker discovery methods developed from a purely computational perspective:

- **Network context**

In recent years, the systems biology community has shifted toward a network-centric view on pathogenesis. It has become widely accepted that pathways rather than individual genes dictate the course of carcinogenesis and complex human disease formation [61]. Different mutations in the same pathway can all result in dysregulation, such as excessive cell proliferation, which forms the basis of tumor growth [62], [63]. This fact unfortunately implies that the traditional paradigm that relies on features being over-represented in disease samples would fail to recognize biomarkers that are only present in subsets of the disease samples.

- **Clinical relevance**

In addition, traditional approaches are mainly concerned with statistical significance and often neglect to consider the clinical relevance of the selected biomarkers. While they have been applied to the problem of biomarker discovery to varying degree of success, they are usually done to optimize toward statistical significance without considering biological importance of the features. As a result, a gene with no biological relevance to the specific target variable may be selected simply because its expression pattern is similar to the expression pattern of truly important genes (multiplicity) and running the same algorithm on different sets of data could result in discrepancy in genes selected (instability).

We provide a motivating example of these issues in the following section.

1.4 MOTIVATING EXAMPLE

We provide an example (Table 1.3 and Figure 1.2) to illustrate some of the challenges alluded to in the pervious section. In this breast cancer gene expression data, we are interested in identifying genes that differentiate the two cancer subtypes: Basal (24 samples) and Luminal (36 samples). In 10-fold cross validation, a total of 11 candidate genes are selected by a feature selection method (HITON-PC [64]). The top candidate gene, MSN, is consistently selected in all cross-validation iterations (Table 1.3), yet it does not seem to directly play a role in tumor growth. When we examine its closest neighbors (genes) in terms of similarity in expression across the samples, we observe that several are indeed tumor suppressors (CAV1, CAV2, CD44). Similarly, XBP1 is selected in several rounds. While it is not directly known to be involved with breast cancer, its closest neighbor FOXA1 is known to be involved in ESR-mediated transcription in breast cancer cells (Figure 1.2). Interestingly, when we examined the local potential regulatory relationships between the selected genes and their top neighbors, we found potential XBP1 transcription factor binding sites in the promoter of FOXA1 (Figure 1.2). This observation suggests that a method that performs variable selection on groups of variables and additionally provides contextual information around the selected groups could provide more biologically robust and meaningful biomarkers.

Table 1.3 A list of candidate genes that define Basal versus Luminal breast cancer subtypes. Marked in red are genes known to be involved with tumorgenesis. Marked in orange are genes known to have potential roles in tumor growth and energetic. Rows highlighted in the same color are gene groups that cluster together based on their expression profiles.

Selected Genes	Top Neighbors	Cross-validation iteration
MSN	CAV1 CD44 AKR1B1 FOSL1 PTRF CAV2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
XBP1	FOXA1 SPDEF ATAD4 TTC39A TOB1 AGR2 SLC9A3R1 TJP3 ARHGAP8	2, 3, 4, 5, 8, 10
AKR1B1	MSN CAV1 CAV2 CD44 UPP1 FOSL1 PTRF	1, 2, 3, 4, 5, 7
CAV2	CAV1 AKR1B1 MSN PTRF RIN3 CD44 EPHA2 ADORA2B	3
ETV5	IGF2BP2 UPP1 WWTR1 ELK3 SGK1 FOSL1 COTL1 IGF2BP3	8
CLDN3	SLC9A3R1 TTC39A FOXA1 GATA3 ANXA9 XBP1	6
PTRF	CAV1 FOSL1 MSN CAV2 LOXL2 AXL AKR1B1 ACSL4	10
AGR2	ATAD4 SPDEF XBP1 TJP3 FOXA1 EMP2 TTC39A MYO5C	6, 7, 9
MBL1P1	CCRN4L EML2 H6PD ACRV1 HBG1 HBG2 PIK3R5 IFNA4 CKM ARSF MYL7	9
SPDEF	FOXA1 XBP1 ATAD4 ARHGAP8 AGR2 EFHD1 SELENBP1 RAB17	6, 8 , 9
GPD1L	SPDEF TTC39A TBC1D30 ATAD4 SLC35A1 XBP1 ALDH6A1 REEP5 SLC9A3R1	9

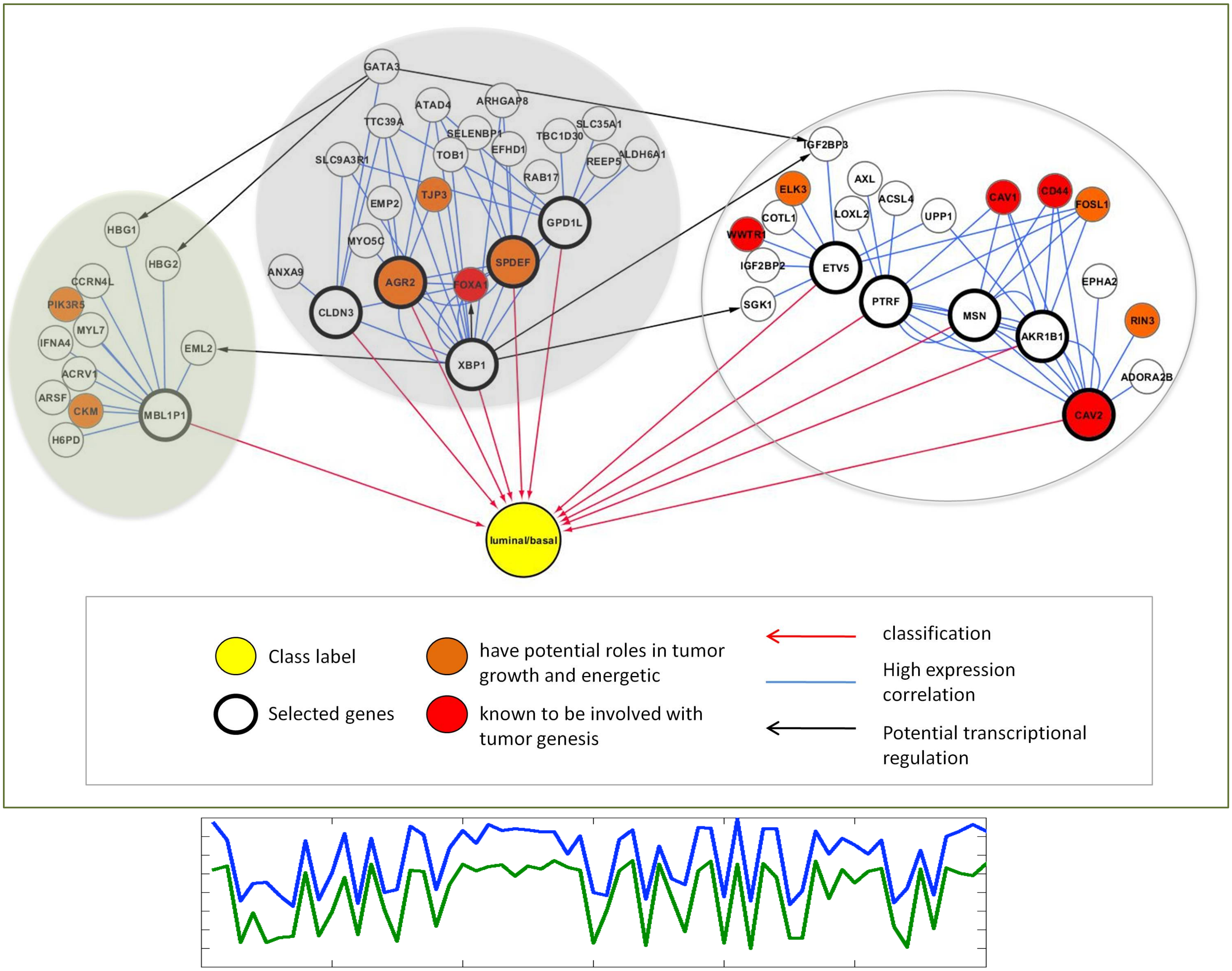


Figure 1.2 An example of gene multiplicity and its implications.

(Top) Selected candidate genes and their top neighbors placed in context of regulatory network. Genes in bold borders with a red arrow pointing to the class label (Luminal/Basal, yellow node) are those selected, as shown in Table1.3. Their top neighbors with expression highly correlated to theirs are connected to them in blue edges. Black arrows indicate potential regulatory relationships based on presence of binding motifs in promoters. Orange and red coloring of the nodes indicate their potential role in tumor genesis, consistent with color scheme Table 1.3. Finally, the grouping of the nodes and their respective background colors indicate potential clusters, consistent with color scheme in Table 1.3. (Bottom) Expression levels of XBP1 (blue) and FOXA1 (green) are plotted across samples. XBP1 and FOXA1 have very similar expression patterns, but XBP1 is chosen as the candidate gene.

1.5 OUTLINE OF OUR APPROACH

This dissertation illustrates my attempt to address each of these issues in the form of a three-component, module-based biomarker discovery framework.

We conjecture that due to the complex nature of pathogenesis, several sets of molecular signatures could be equally predictive of disease state. Furthermore, we hypothesize that biologically meaningful variables can be highly correlated with other less relevant but statistically discriminative variables, even if the biologically relevant variables themselves are not maximally predictive. Since high-dimensional genomics data exhibit an intrinsic correlation structure among variables, we argue that it is beneficial to incorporate this information in the variable selection process.

In response to our hypothesis, we aimed to exploit the correlation structure of the variables and organize them into modules. Subdividing variables into modules greatly reduces the complexity of the model space, partly addressing the dimensionality problem. In addition, this organization can offer insights into the resulting network structure, and points to potential molecular functions for the lesser-known members in the system. We approach this task with a recursive spectral clustering strategy. Spectral methods are appropriate here since data heterogeneity can be somewhat reduced by transformation of the correlation matrix. In addition, a recursive design speeds up computation and summons a natural representation of the partition in a hierarchical, multi-scale structure.

We proceed to take advantage of this tree structure to achieve the goal of group feature selection. The group feature selection framework directly tackles the multiplicity and instability issues. By treating clusters of variables as single entities, multiplicity can be eliminated as redundant variables are now summarized by a single group variable. This approach also affords a greater stability in the system, since single, unstable variables will eventually converge to larger groups that can be expected to get selected more consistently. Two conditional independence tests are designed to collectively determine whether we can accept a substitution of a single predictive variable with a group variable without losing significant predictive accuracy. The thresholds for these tests can then be used to fine-tune the resolution of the predictive group variables we output.

Finally, we address the issue of clinical relevance in all three components of the framework. In the clustering step, a prior incorporation scheme is developed to formally incorporate expert prior knowledge. The group feature selection procedure allows the selection of biologically informative genes by virtue of association with statistically predictive variables. In the final step, we enrich the selected variables with relevant contextual information, including regulatory relationships between TFs, miRNAs and genes, and deliver them in the form of an integrated network through a user-friendly web interface. The outline of this approach is presented in Figure 1.3.

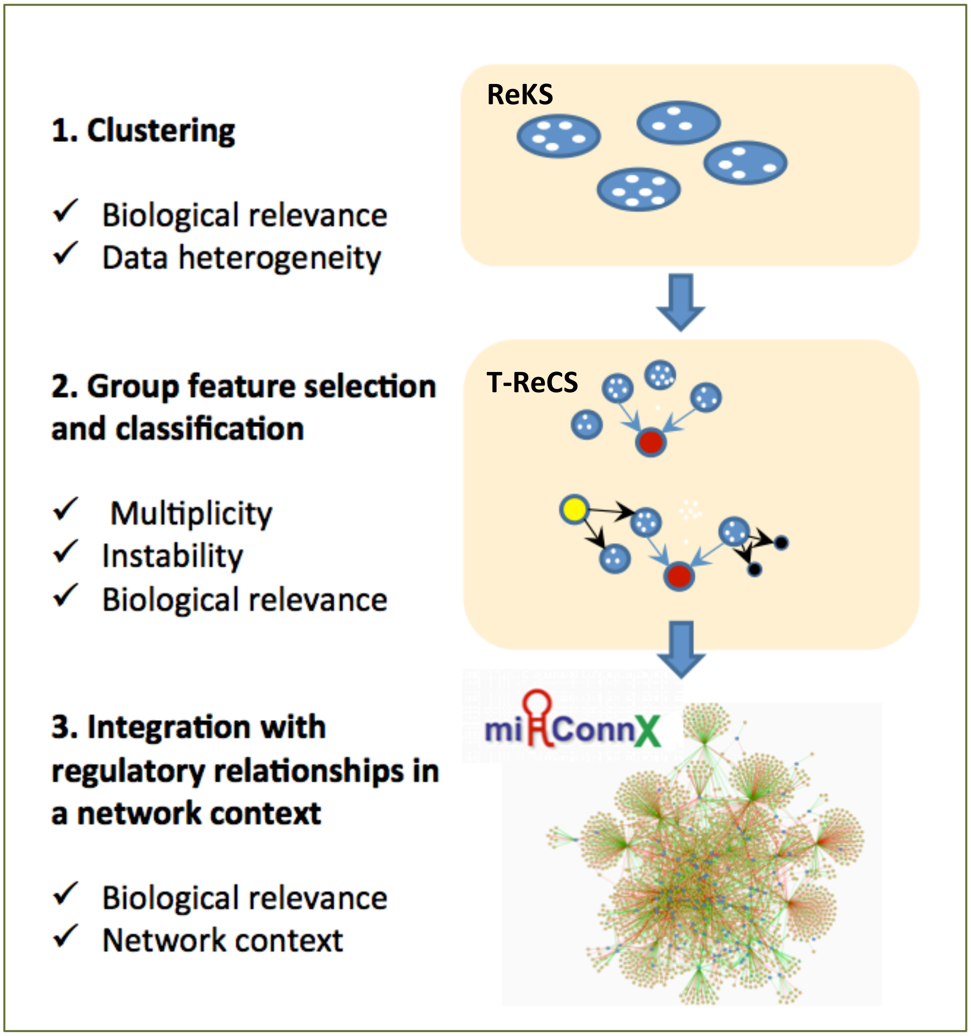


Figure 1.3 Overall approach of the proposed module-based biomarker discovery framework

1.6 CONTRIBUTIONS

Motivated by the abundance of recently available large-scale clinical data, and a need for a more biologically informed biomarker discovery method, an integrative, module-based framework for biomarker selection is developed and outlined in this dissertation. We highlight some of its key contributions.

Computationally, this work contributes two novel algorithms to the community, both building on sound existing algorithms. The clustering algorithm, Recursive K-means Spectral Clustering (**ReKS**) (Figure 1.3 [1]), is one of the first to apply a recursive form of spectral clustering algorithm on high-dimensional clinical expression data. The second algorithm, Tree-guided Recursive Cluster Selection (**T-ReCS**) (Figure 1.3 [2]), offers a novel group variable selection framework based on local causal discovery theories. The two algorithms are integrated; the output of the clustering algorithm can be used to guide the feature selection process. Nevertheless, each can also be used separately as standalone algorithms. Aside from the application in high-throughput genomics data emphasized in this dissertation, both of these algorithms are general-purpose methods also applicable to other high-dimensional datasets with correlated variables.

To serve the biomedical community, we developed an interactive web-server, **mirConnX** (Figure 1.3 [3]), to present an integrated transcriptional and post-transcriptional network. The mirConnX network is constructed from a comprehensive compilation of prior regulatory relationships and computational predictions, and integrated with user supplied condition-specific expression data. Since its introduction, mirConnX has assisted numerous[1] users to explore their datasets in the context of regulatory relationships. An upcoming release of mirConnX 2.0 will further integrate with the aforementioned biomarker discovery framework. The selected discriminative gene and miRNA clusters are annotated with curated and putative regulatory relationships, and presented in a network context. We hope that through our proposed module-based biomarker discovery framework, integrated in the mirConnX environment, we will further

[1] As of September 2013, mirConnX has had 2407 unique visitors

[2] Their rule of thumb is that the G^2 test is reliable if there are five or more instances per degree of freedom of the test

assist the biomedical community in generating actionable biological hypothesis and advancing toward the ultimate goal of improving understanding of complex disease formation.

1.7 OVERVIEW AND ORGANIZATION

This dissertation is organized as follows.

In Chapter 2, ReKS, a recursive spectral clustering method developed to partition genes into a tree structure is described in detail. The algorithm is evaluated against other popular clustering methods on several metrics and on a benchmarking dataset. Its applications to large-scale clinical datasets are presented. We also described a formal prior information incorporation framework for incorporating prior knowledge such as protein-protein interaction, domain knowledge or pathway information. We additionally pointed to a potential future strategy for improving stability using a perturbation algorithm.

Chapter 3 presents the group variable selection algorithm, T-ReCS, which exploits the tree structure generated previously to guide its search for discriminative group variables. Relevant background concepts are introduced and the rationale for the method is explained in detail. The performance of the algorithm is evaluated on simulated, benchmarking and real data.

In Chapter 4, we describe the details of the method for constructing the backend of the integrative web server mirConnX, and highlight several examples of its applications. We also provide a demonstration of the proposed integrative analysis on a set of melanoma gene and miRNA expression dataset to reveal the utility of our integrated framework, and illustrate our vision for the upcoming release of the web server.

Finally, we provide discussion and future directions in Chapter 5.

2.0 REKS: RECURSIVE K-MEANS SPECTRAL CLUSTERING

Clustering of gene expression data simplifies subsequent data analyses and forms the basis of numerous approaches for biomarker identification, prediction of clinical outcome, and therapeutic strategies. The most popular clustering methods such as K-means and hierarchical clustering are intuitive and easy to use, but they require arbitrary choices on their various parameters (number of clusters for K-means, and a threshold to cut the tree for hierarchical clustering). Human disease gene expression data are in general more difficult to cluster efficiently due to background (genotype) heterogeneity, disease stage and progression differences and disease subtyping; all of which cause gene expression datasets to be more heterogeneous. Spectral clustering has been recently introduced in many fields as a promising alternative to standard clustering methods. The idea is that pairwise comparisons can help reveal global features through the *eigen* techniques. In this paper, we developed a new method (ReKS) for clustering disease gene expression data based on a recursive spectral clustering algorithm. We benchmarked ReKS on three large-scale cancer datasets and we compared it to different clustering methods with respect to execution time, background models and external biological knowledge. We found ReKS to be superior to the hierarchical methods and equally good to *K*-means, but much faster than them and without the requirement of a priori knowledge of K. Overall, we believe that recursive spectral clustering offers an attractive alternative for efficient clustering of human disease data.

2.1 CLUSTERING OVERVIEW AND MOTIVATING EXAMPLE

The explosion of gene expression and other data collection from thousands of patients of several diseases has created novel questions about their meaningful organization and analysis. The Cancer Genome Atlas (TCGA) [22] initiative for example provides large heterogeneous datasets from patients with different types of cancers including breast, ovarian and glioblastoma. However, unlike data from model organisms and cell lines that inherently contain uniform genetic background, and where experiments are conducted under controlled conditions, disease samples are typically much more heterogeneous. Differences in the genetic background of the subjects, disease stage, progression, and severity as well as the presence of disease subtypes contribute to the overall heterogeneity. Discovering genes or features that are most relevant to the disease in question and identifying disease subtypes from such heterogeneous data remains an open problem.

Clustering, the unsupervised grouping of data vectors into classes with similar properties is a powerful technique that can help solve this problem by reducing the number of features one has to analyze and by extracting important information directly from data when prior knowledge is not available. As such, it has formed the basis of many feature selection and classification methods [65], [66]. Hierarchical and data partitioning algorithms (like *K*-means) have been used widely in many domains [67] including biology [68], [69]. They have become very popular due to their intuitiveness, ease of use, and availability of software. Their biggest drawbacks come from the usually arbitrary selection of parameters, such as the optimal number of clusters (for *K*-means) or an appropriate threshold for cutting the tree (for hierarchical clustering).

When applied to datasets from model organisms and cell lines, these clustering approaches have been quite successful in identifying biologically informative sets of genes [68],

[69]. However, the heterogeneity of the disease samples hinders their efficiency in them. Figure 2.1 shows an example of such a dataset; a dendrogram produced from the breast cancer TCGA data, in comparison to dendrogram generated from the less heterogeneous yeast expression data. It is obvious that the structure of the data makes it difficult to find a threshold to prune the tree to produce a satisfactory number of clusters, since every newly formed cluster is joined with a singleton node each time. Thus, despite its popularity, classical hierarchical clustering frequently performs poorly in discovering a satisfactory group structure within gene expression data. Tight clustering [70] and fuzzy clustering [71] attempt to build more biologically informative clusters either by focusing only on closely related genes while ignoring the rest, or by allowing overlap in cluster memberships. However, both methods suffer from long execution times. Similarly, Affinity Propagation [72] has been applied on gene clustering successfully but a significant execution time trade-off exists.

More recently, spectral clustering approaches have been used for data classification, regression and dimensionality reduction in a wide variety of domains, and have also been applied to gene expression data [73]. The spectral clustering formulation requires building a network of genes, encoding their pairwise interactions as edge weights, and analyzing the eigenvectors and eigenvalues of a matrix derived from such a network. To our knowledge, no systematic attempt has been made to-date to test and compare the performance of existing clustering methods in large-scale disease gene expression data, perhaps due to unavailability of suitable size datasets. In this paper, we evaluate the standard *K*-means and hierarchical clustering methods on three large TCGA datasets. The evaluation is performed using intrinsic measures and external information. We introduce ReKS (Recursive *K*-means Spectral clustering), and compare it to the two aforementioned methods on the TCGA data. ReKS leverages the global similarity structure that spectral clustering provides, while saving on computing time by performing recursion. At each recursion step, we exploit the distribution of eigenvalues to select the optimal number of partitions, thus eliminating the need for pre-specifying K. We show that ReKS is very useful in deriving important biological information from patient gene expression data. Furthermore, we show how to add prior information from KEGG [74] pathway to refine the cluster boundaries.

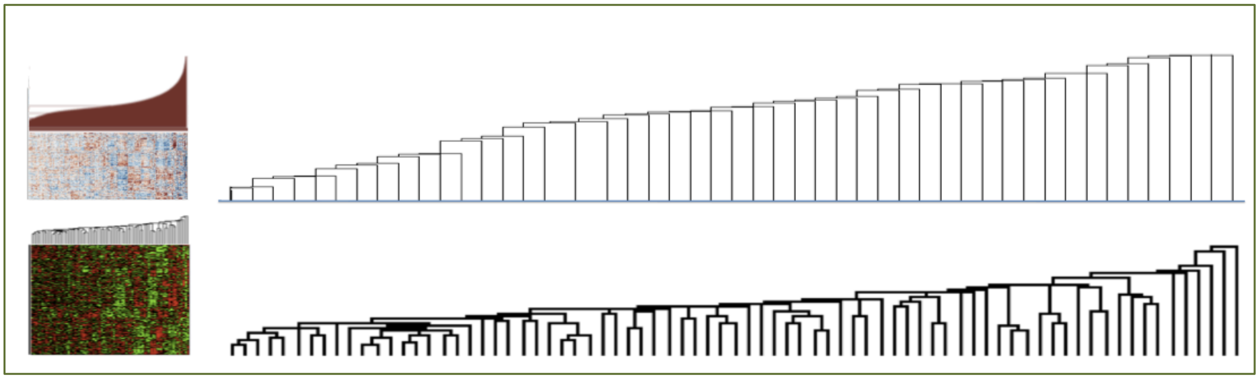


Figure 2.1 Clustering patient data is more difficult than cell-based data.
Partial views of dendrograms constructed from hierarchical clustering of the TCGA Breast Cancer expression data (top) and the yeast expression data (from Spellman *et al.* [75]). The dendrograms suggest that it is easier to select a threshold to prune the tree and generate potentially meaningful clusters for the yeast data but not so for the breast cancer data.

2.2 SPECTRAL CLUSTERING

The spectral clustering formulation requires building a network of genes, encoding their pairwise interactions as edge weights, and analyzing the vectors and eigenvalues of a matrix derived from such a network. This procedure is well established in the literature [76] so here we limit our discussion to the main points of the algorithm and use a Markov chain perspective to help us reason further about the idiosyncrasies of the algorithm when applied to cancer expression data.

A convenient framework for understanding the spectral method is to consider the partitioning of an undirected graph $G =< V, E >$ into a set of distinct clusters. Here the genes are represented as vertices v_i for $i = 1 \ldots N$ where N is the total number of genes and network edges have weights w_{ij} that are non-negative symmetric ($w_{ij} = w_{ji}$) to encode the strength of interaction between a given pair of genes. Affinities denote how likely it is for a pair of genes to belong to the same group. Here we used as affinities a modified form of the correlation coefficient ρ_{ij}, calculated on the gene expression vectors:

$$w_{ij} = exp\left(-\left(sin\frac{arccos(\rho_{ij})}{2}\right)^2\right) \qquad (2.1)$$

This is distance measure previously found to give empirical success in the clustering of gene expression data [73]. Note that high affinities correspond to pairs of genes that are likely to belong in the same group (e.g., participate in a pathway). In this paper, we ensured that the network is connected so that there is a path between any two nodes of the network. Our goal is to group genes into distinct clusters so that genes within each group are highly connected to each other, while genes in distinct clusters are dissimilar.

Spectral methods use local (pairwise) similarity (affinity) measurements between the nodes to reveal global properties of the dataset. The global properties that emerge are best understood in terms of a random walk formulation on the network [77]–[79]. The random walk is initiated by constructing a Markov transition matrix over the edge weights. Representing the matrix of affinities w_{ij} by W and defining the degree of a node by $d_j = \sum_i w_{ij}$, a Markov transition matrix M can be defined over the edge weights by

$$M = WD^{-1} \qquad (2.2)$$

where D is a diagonal matrix stacked with degree values d_j. The transition matrix M can be used to set up a diffusion process over the network. In particular, a starting distribution p^0 of the Markov chain evolves to $p = M^\beta p^0$ after β iterations. As β approaches infinity, the Markov chain can be shown to approach a stationary distribution:$M^\infty = \pi 1^T$ is an outer product of 1 (a column vector of N 1s) and π (column vector of length N). It is easy to show that π is uniquely given by: $\pi_i = d_i / \sum_j d_j$ and is the leading eigenvector of M: $M\pi = \pi$ with eigenvalue 1.

We can analyze the diffusion process analytically by using the eigenvectors and eigenvalues of *M*. From an eigen perspective the diffusion process can be seen as [78]:

$$p^\beta = \pi + \sum_2^n \lambda_j{}^\beta D^{0.5} u_j u_j{}^T D^{-0.5} p^0 \qquad (2.3)$$

where the eigenvalue $\lambda_1 = 1$ is associated with stationary distribution π. The eigenvectors are arranged in decreasing order of their eigenvalues, so the second eigenvector u_2 perturbs the stationary distribution the most as $\lambda_2 \geq \lambda_k$ for $k > 2$. The matrix $u_2 u_2{}^T$ has elements $u_{2,i} \times u_{2,j}$, which means the genes that share the same sign in u_2 will have their transition probability increased, while transitions across points with different signs are decreased. A straightforward strategy for partitioning the network is to use the sign of the elements in u_2 to cluster the genes into two distinct groups.

Ng *et al.* [80] showed how this property translates to a condition of piecewise constancy on the form of leading eigenvectors, i.e. elements of the eigenvector have approximately the same value with-in each putative cluster. Specifically, it was shown that for K weakly coupled clusters, the leading K eigenvectors of the transition matrix M will be roughly piecewise constant. The *K*-means spectral clustering method is a particular manner of employing the standard *K*-means algorithm on the elements of the leading K eigenvectors to extract K clusters simultaneously. We follow the recipe in Ng *et al.* where instead of using a potentially non-symmetric matrix M, a symmetric normalized graph Laplacian $L = D^{-0.5}WD^{-0.5}$, whose eigenvalues and eigenvectors are similarly related to M, is used for partitioning the graph.

Spectral approaches have also some drawbacks. Their basic assumption of piecewise constancy in the form of leading eigenvectors need not hold on real data. Much work has been done to make this step robust, including the introduction of optimal cut ratios [81] and relaxations [82], [83] and highlighting the conditions under which these methods can be expected to perform well [78]. Spectral methods can be slow as they involve eigen decomposition of potentially large matrices ($O(n^3)$). Recent attempts at addressing this issue include implementing the algorithm in parallel [84], speeding eigen decomposition with Nystrom approximations [85], building hierarchical transition matrices [86] and embedding distortion measures for faster analysis of large-scale datasets [87].

2.3 METHOD OVERVIEW

In this paper, we will pursue a recursive form of *K*-means spectral clustering (ReKS), apply it on cancer expression data from patients and understand the intrinsic structure of the data by establishing a baseline clustering result. ReKS first defines an affinity matrix of all pairwise similarities between genes. We reduce the computational burden with sparse matrices, such that each gene is connected to a small number of its neighbors (default: 15) with varying affinities, and extract only a small subspace of eigenpairs (default: 20). In each recursion step, we determine the most appropriate subspace in which to run *K*-means using the eigengap heuristic, which is to compute the ratio of successive eigenvalues and pick K of $argmax_i\, \lambda_i/\lambda_{i+1}$, for $i =$

1 to 20. We apply the eigengap heuristic at each recursion level to determine the optimal number of partitions at that level. In addition, to improve the convergence of the *K*-means algorithm we initiate the algorithm with orthogonal seed points. For each newly formed cluster, we extract the corresponding affinity sub-matrix and repeat the procedure.

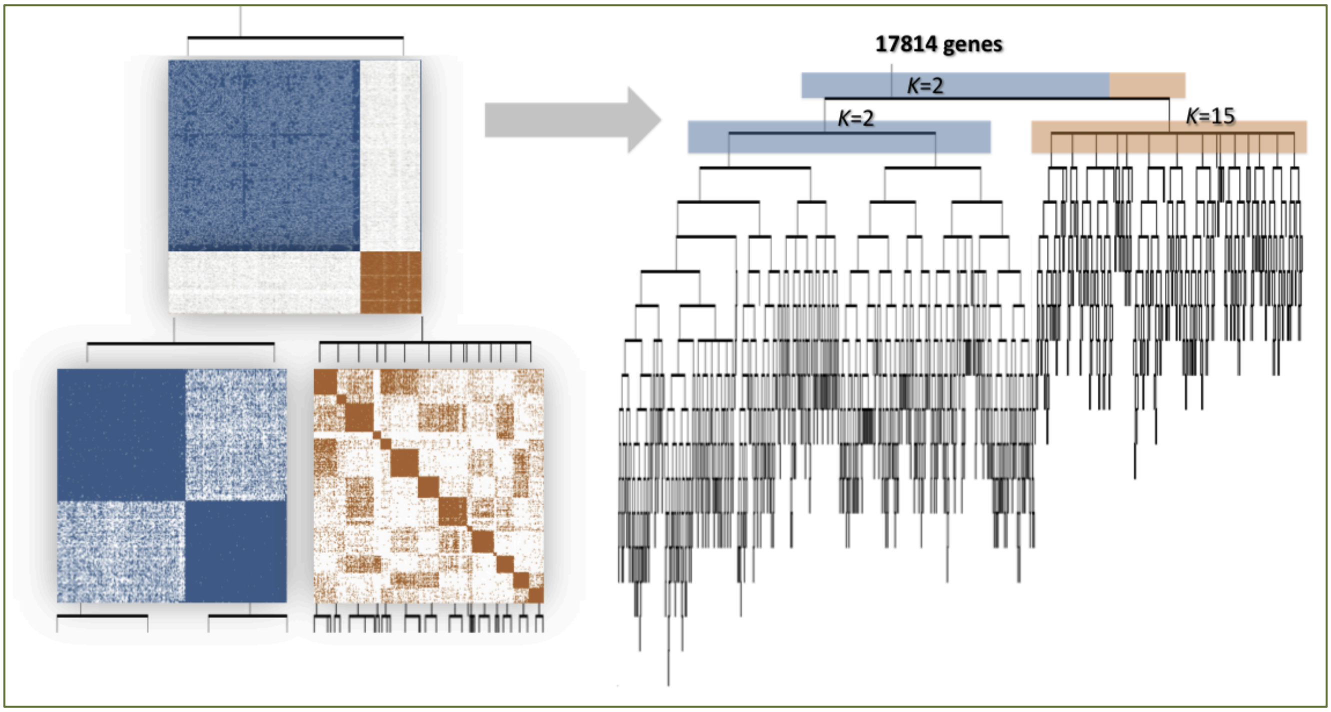


Figure 2.2 Demonstration of the ReKS method on the GBM dataset.
(Left) The first two iterations of *K*-means spectral decomposition recursions: two clusters are visible in the affinity map constructed from the entire dataset at the first level. From each, a new affinity matrix is constructed and spectral clustering repeated on the sub-affinity matrix. (Right) Complete tree obtained by ReKS iterations. Each leaf node corresponds to a gene cluster in the final partition.

In Figure 2.2(Left) we illustrate the top two levels of ReKS recursion on the GBM dataset. At level-1 an obvious partition exists for the original affinity matrix. The genes are split into two clusters at this node, and for each cluster, a new affinity matrix is computed. ReKS performs this procedure iteratively stopping when further split would cause all clusters to be 35 or smaller in size. The stopping threshold corresponds to the average number of genes that participate in a KEGG pathway. In the end, we arrive at a tree where each leaf node represents a gene cluster. Note that with this procedure clusters of smaller than 35 genes could be obtained, for example due to an early split off the tree, as long as there is a cluster that is large in size. Figure 2.2(Right) presents the full tree generated by ReKS on the GBM dataset.

The complexity of ReKS is roughly $O(N^2)$, N being the total number of genes to cluster. At every node of the tree, an SVD is performed at $O(dn^2)$, n being the number of genes at the node, and k-means is performed at $O(i, k, n, d)$, where i is the number of iterations, d is the reduced dimension capped at 20, and k is the corresponding number of clusters <= 20. Since i is bounded and d and k are fixed to 20 and less, the k-means step is essentially linear to n. Assuming a balanced tree with each node having $k=20$ children, the overall complexity is $\sum_{d=1}^{\infty} k^d O(d\left(\frac{N}{k^d}\right)^2) \approx O(N^2)$, and $O(N^3)$ in the worst case scenario with an extremely unbalanced tree.

2.4 PERFORMANCE EVALUATION

2.4.1 Comparison to other methods on TCGA cancer data

2.4.1.1 Data description

We applied ReKS on the three most complete TCGA gene expression datasets to date: Glioblastoma multiform (GBM) with a total of 575 tumor samples, Ovarian serous cystadenocarcinoma (OV) with a total of 590 tumor samples, and Breast invasive carcinoma (BRCA) with a total of 799 tumor samples. The level 3, normalized and gene-collapsed data obtained from the TCGA portal were downloaded and no further normalization was performed.

We compare our method against four other partition solutions: (1) average linkage hierarchical clustering, (2) average linkage hierarchical clustering on the spectral space, (3) K-means and (4) K-means on the spectral space. These algorithms are chosen to cover a range of common clustering techniques and clustering assumptions.

2.4.1.2 Comparison of ReKS and other clustering strategies on TCGA data

Agglomerative clustering methods build a hierarchy of clusters from bottom up. It is perhaps the most popular on gene expression data analysis [88], due to its ease of use and readily available implementations. We performed hierarchical agglomerative clustering using Euclidean distance and average linkage. A maximum number of clusters is specified to be comparable to the number of clusters K obtained when running ReKS. Since this choice might be considered favorable to ReKS, we also performed hierarchical clustering on the top three eigenvectors in the spectrum, using cosine distances to measure the distance on the resulted unit sphere. Note that hierarchical clustering is done from bottom up, using local similarities, and does not embed the global structure in its tree.

Similarly, standard K-means and K-means performed on the spectral space are included for benchmarking purposes. Given a number of clusters, K, the algorithm iteratively assigns members to centroids and re-adjusts the centroids of the clusters. K-means tends to perform well as it directly optimizes the intra-cluster distances, but tends to be slow especially as K increases. Here we used the default implementation of the K-means clustering algorithm in Matlab, with Euclidean distance, again using the K obtained from ReKS. We also ran K-means on the spectral space, effectively performing ReKS only once without choosing an optimal number of eigenvectors to use, but instead using K top eigenvectors.

Shown in Figure 2.3 are the distributions of the cluster sizes when applying the five methods to the three TCGA datasets. Hierarchical clustering, whether in the original or the eigenspace, produces a very skewed distribution of cluster sizes that is possibly an artifact of focusing on only local similarities. The K-means methods and ReKS produce cluster sizes that span roughly the same range. However, the K-means methods produce distributions that are artificially Gaussian, with relatively little clusters that contain small number of genes.

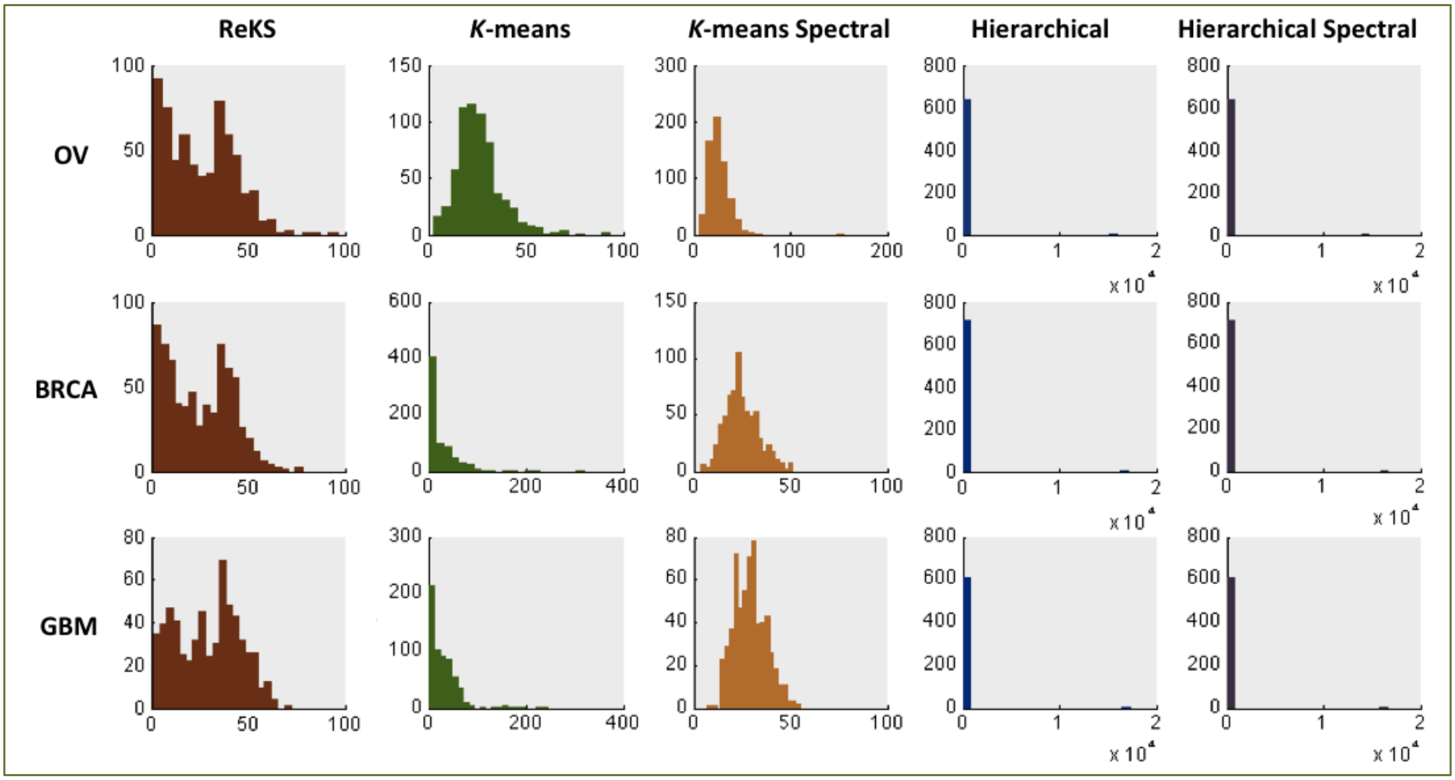


Figure 2.3 Distribution of cluster sizes produced by ReKS and by other methods

2.4.1.3 Cluster quality evaluation

We evaluate the quality of the clusters obtained from each of the five methods (ReKS, K-means, K-means spectral, Hierarchical, Hierarchical spectral) using both intrinsic, statistical measures as well as external biological evidence, as detailed in the sections below.

- **Calinski-Harabasz**

To evaluate the quality of the clusters, we used the Calinski-H[arabasz measure [89], defined by:

$$CH = \frac{traceB/(K-1)}{traceM/(m-K)} \qquad (2.4)$$

where $traceB$ denotes the error sum of squares between different clusters, $traceM$ is the intra-cluster square differences, m is the number of objects assigned to the i^{th} cluster, and K is number of clusters. This statistic is effectively an adjusted measure of the ratio of between- vs. within- group dispersion matrices. A larger value denotes a higher compactness of the cluster compared to the inter-cluster distances. Figure 2.4(Left) shows the performance of ReKS compared across other methods. Not surprisingly, ReKS outperforms hierarchical clustering in both the original data space as well as the spectral space, as hierarchical clustering produces some very large clusters with no apparent internal cohesion. The K-means based methods and ReKS are comparable in terms of cluster separation across the datasets.

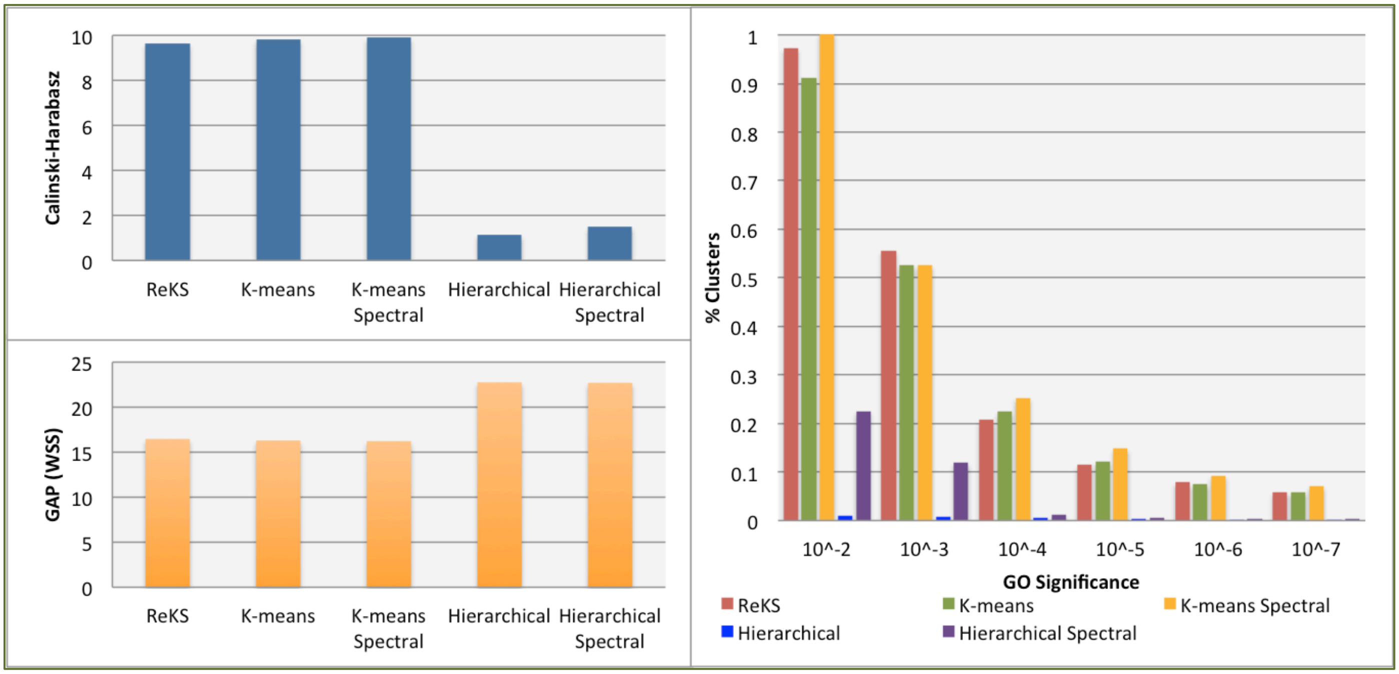


Figure 2.4 Performance of ReKS compared to other methods.
(Left) Cluster validity comparison with other methods using the Calinski-Harabasz and the GAP statistics (Right) Gene Ontology(GO) enrichment across different range of p-values

- **GAP Statistic**

The Gap statistic was proposed as a way to determine optimal cluster size [90]. In short, it is the log ratio of a reference within-cluster sum of square errors over the observed within-cluster sum of squares errors. The reference is usually built from a permutated set of genes that form K random clusters. Since we are comparing the (five) methods across the same dataset with the same K, it is fair to compare the performance of the observed within sum-of squares error only. With this direct proxy, ReKS performs at the same level as K-means based methods (shown in Figure 2.4(Left), and achieved a significantly lower sum-of-square distances than the hierarchical methods.

- **Gene Ontology Enrichment**

Since no ground truth exists for gene cluster partition, we examine the overall quality of the clusters in terms of the amount of enrichment for Gene Ontology (GO) annotations. For each cluster, we test for GO enrichment using a variant of the Fisher's exact test, as described in the *weight01* algorithm of the topGO [91] package in R. The significance level of the test indicates the degree a particular GO annotation is over-represented in a given cluster. For a partition, we calculate the proportion of clusters annotated with a GO term at a p-value threshold. If a cluster has less than five members, the test is not performed. As shown in Figure 2.4(Right), compared to hierarchical clustering, we observe that ReKS contains higher percentage of clusters that are significant at the specified levels, and especially so with more stringent p-value thresholds, and performs roughly the same as K-means methods. Finally, we observe that the spectral methods tend to perform better than their non-spectral counter-parts.

- **Execution Time**

Table 2.1 shows the execution time of the five methods on a 3.4 GHz Intel Core i7 CPU. ReKS is slower than hierarchical clustering but compares favorably to K-means methods.

Table 2.1 ReKS average execution time compared to other methods

Methods	ReKS	*K*-means	*K*-means Spectral	Hierarchical	Hierarchical Spectral
Execution time	373s	6000s	1774s	90s	22s

2.4.2 Benchmarking against patient data

Since gold standard for gene clustering does not exist, we resort to benchmarking ReKS on a set of well established microarray data where the goal is to cluster patients into known disease subtypes. de Souto *et al.* [88] compiled a list of 35 datasets from Affymetrix and cDNA microarrays . They performed a comprehensive analysis of seven different clustering methods and coupled them with seven definitions of proximity measure for clustering cancer tissues.

We ran ReKS on each of the datasets compiled in this study, and calculated the corrected Rand (*cR*) index [92] by comparing the actual classes of the tissue samples with the cluster assignments of the tissue samples. *cR* measures the success of algorithm in recovering the true partition of the datasets. It takes on values from -1 to 1, with 1 indicating perfect agreement between the partitions and 0 being the cluster agreement found by chance. It is defined to be:

$$cR = \frac{\Sigma_i^R \Sigma_j^C \binom{n_{ij}}{2} - \binom{n}{2}^{-1} \Sigma_i^R \binom{n_i}{2} \Sigma_j^C \binom{n_j}{2}}{\frac{1}{2}\left[\Sigma_i^R \binom{n_{i\cdot}}{2} + \Sigma_j^C \binom{n_{\cdot j}}{2}\right] - \binom{n}{2}^{-1} \Sigma_i^R \binom{n_i}{2} \Sigma_j^C \binom{n_j}{2}} \tag{2.5}$$

where n_{ij} is the number of members in cluster u_i and v_j, $n_{i\cdot}$ represents the number of members in cluster u_i, and n being the total number of objects.

We calculate the mean of the *cR* by enforcing the same number of actual classes present in the samples. For a tree, this is accomplished by identifying a level at which the partition would yield the closest number to the number of classes, and merging the clusters with the lowest eigenvalues. As shown in Figure 2.5, ReKS outperformed all the other methods in combination with the proximity measures among the Affy samples, and is comparable to the best performing clustering combination- *K*-means clustering with ranked Euclidean proximity measure, among the cDNA samples.

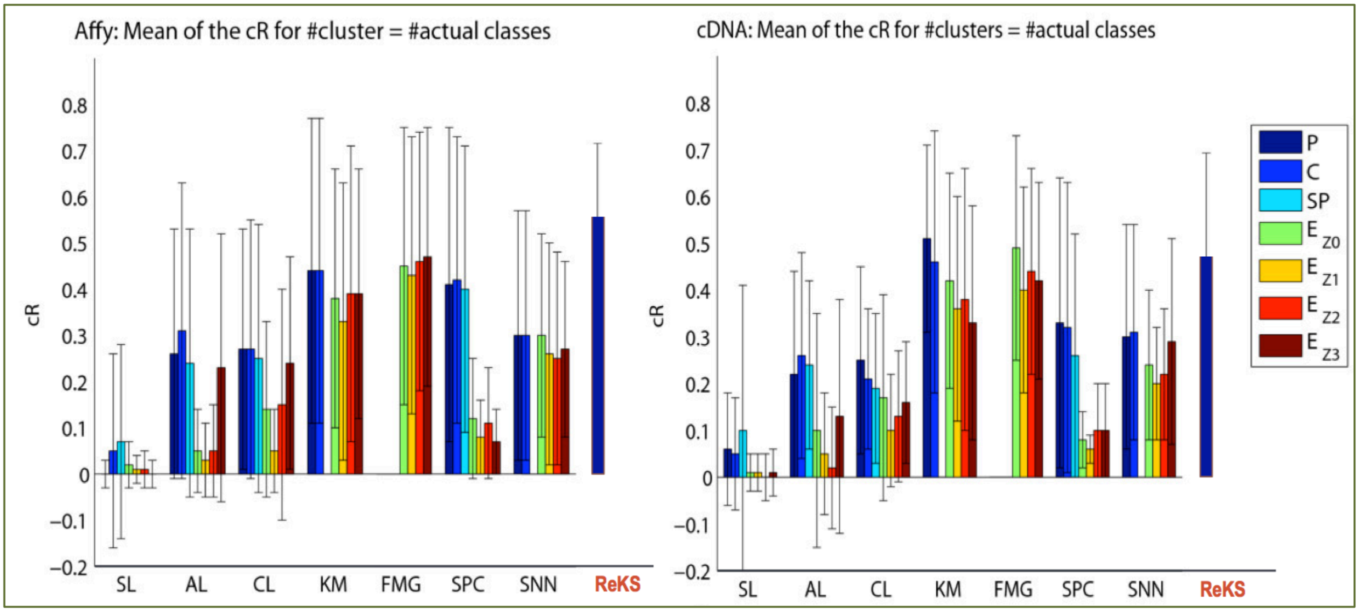


Figure 2.5 Benchmarking ReKS on patient data.
On 35 benchmarking datasets, we calculated the mean of the cR(corrected Rand) index for ReKS (labeled in red) partition results and compared it to the other seven clustering methods: single linkage (SL), complete linkage (CL), average linkage (AL), *K*-means (KM), mixture of multivariate Gaussians (FMG), spectral clustering (SPC) and shared nearest neighbor-based clustering (SNN). Four proximity measures are used together with these methods: Pearson's Correlation coefficient (P), Cosine (C), Spearman's correlation coefficient (SP) and Euclidean Distance (E). Regarding Euclidean distance, we employ the data in four different versions: original (Z0), standardized (Z1), scaled (Z2) and ranked (Z3) versions. For ReKS, we only include results for Pearson's Correlation.

2.5 PRIOR INCORPORATION

We use existing expert knowledge as prior information (from KEGG pathway [74]) to guide our clustering method, aiming to generate partitions that are even more biologically meaningful. The KEGG database includes a collection of manually curated pathways constructed from knowledge accrued from the literature. For the purposes of ReKS, we assume that the genes in a KEGG pathway are fully connected to each other (i.e., should belong in the same cluster). We code this prior knowledge in a constraint matrix U in which each column U^c is a pathway, and $u_{ic} = 1, u_{jc} = -1$ if a pair of genes i, j participate in the same KEGG pathway c. Similar to what was detailed in Ji *et al.* [93], where they supplied a prior for document clustering using *K*-means spectral decomposition, we apply a penalty term to the normalized graph Laplacian as follows:

$$L' = D^{-0.5}(W + \beta U^T U)D^{-0.5} \tag{2.6}$$

where $\beta \geq 0$ controls the degree of enforcement of the KEGG prior knowledge. As shown in Ji *et al.*, the eigenvectors of the K smallest eigenvalues of L' form the eigen-space represents a transformation of the affinity space embedded with prior information. We then proceeded to apply the *K*-means algorithm within the eigenspace, and iterate recursively as we did with ReKS. As shown in Figure 2.6(Left), when we use a large amount of prior, not surprisingly the GO significance becomes very large. We observe the significance of the clusters do not drop very fast as β decreases. Therefore, small amount of prior at roughly β = 0.2 may be enough to enhance the biological significance of the ReKS clustering results.

We applied ReKS on the TCGA datasets at $\beta=0.2$. A total of 715, 639, and 610 clusters are obtained for BR, OV, and GBM respectively. As shown in Figure 2.6(Right), we observe that there exists a slight anti-correlation between how early a cluster splits off the tree and how significant the cluster is (ρ = -0.2112, p <10-7). As a preliminary observation, how early a cluster is formed seems to imply the "tightness" of the cluster, this result seems to suggest that there is a slightly higher chance the clusters that form early to be more biologically significant. For example, in Figure 2.7(Right) there is a tight histone H1 cluster that splits off the BRCA tree at the third level on the top. It has been shown that EB1089 treatment of breast cancer cell lines (MCF-7, BT20, T47D, and ZR75) is correlated with a reduction in CdK2 kinase activity towards phosphorylation of histone H1 and a decrease in DNA synthesis [94]. This cluster does not exist in *K*-means spectral, *K*-means, and spectral hierarchical clustering results, and only exists in a mega-cluster in hierarchical clustering partition. Additionally, upon examining the resulted tree closely, we found that a few genes that have been implicated for breast cancer [95] cluster together or close to each other on the tree, as shown in Figure 2.7(Left). When considering a few of these sub-clusters together, the top functional categories that emerged are indeed caner and p53 pathways. We found several of these examples throughout the tree, all within 12 levels up to which the composition of the clusters remains stable when splitting the data into training and test sets. We note that PIK3CA, RB1, and RUNX1 do not cluster together in any of the other methods we compared to, nor does the rest of the genes we examined. This example suggests that the tree structure could be useful for inferring additional previously unknown biomarkers.

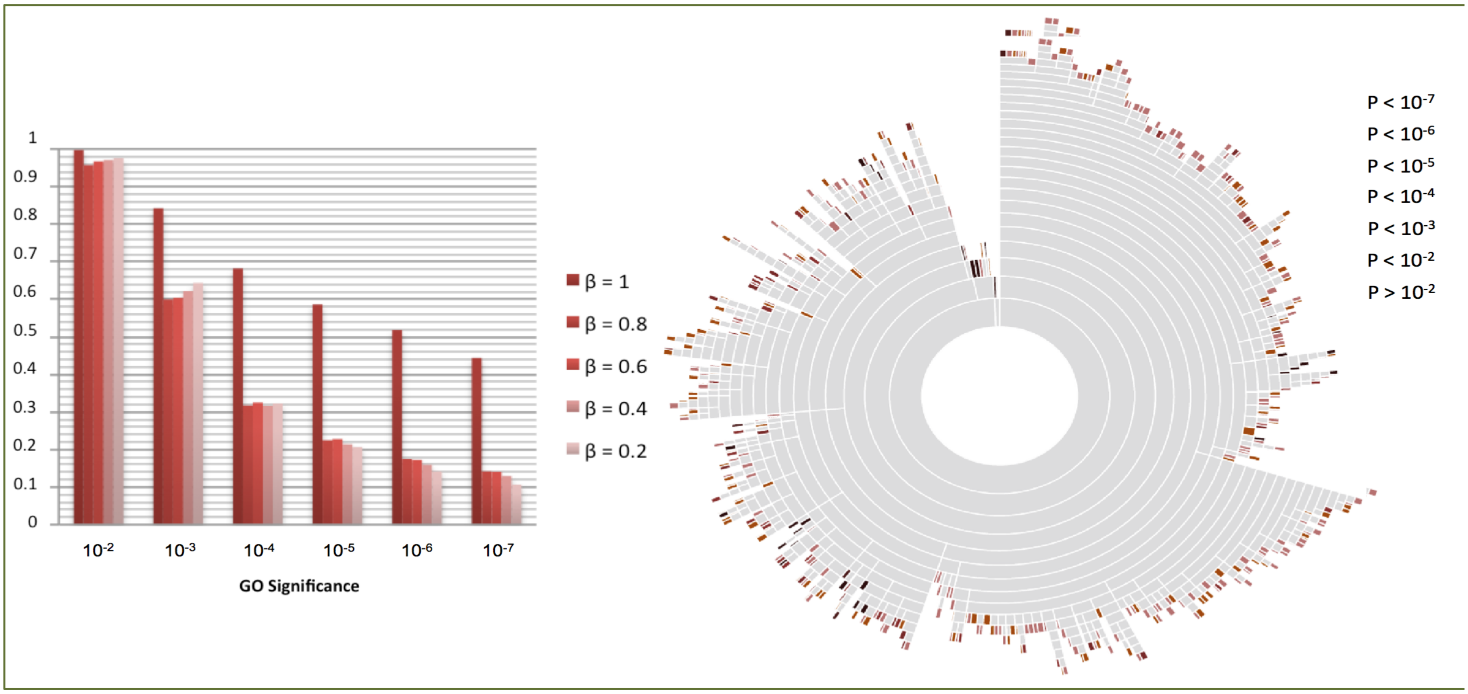
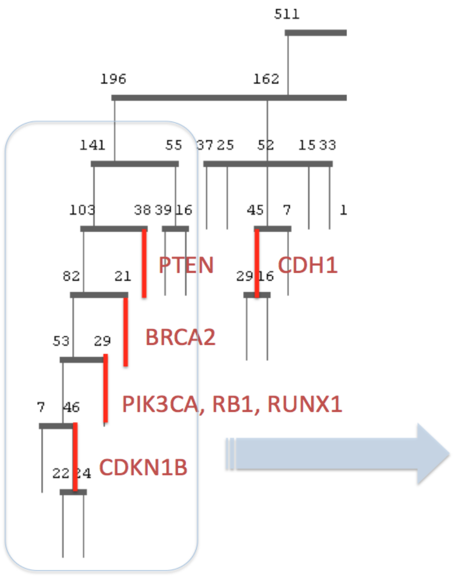


Figure 2.6 ReKS with prior incorporation applied on BRCA dataset.
(Left) Effect of incorporation of prior information on the GO significance of the obtained clusters. $\boldsymbol{\beta}$ controls the degree of enforcement of the KEGG prior knowledge (Right) A sunburst diagram for the BRCA dataset. In this alternative representation of the ReKS clustering results, each concentric circle represents a level of the tree. Each ring is sub-divided into clusters. The color of a leaf node denotes the GO significance of the cluster. There exists a small anti-correlation ($\boldsymbol{\rho}$ = -0.2112, p < 10-7) between the level from which a cluster splits off, and its GO significance



Functional Category	E-value
Pathway in cancer	1.90E-71
p53 signaling pathway	2.60E-31
regulation of apoptosis	2.10E-26
regulation of programmed cell death	1.70E-26
regulation of cell death	1.30E-26
cell cycle process	2.60E-23
cell cycle	8.20E-22
cell cycle phase	7.00E-21
DNA recombination	1.10E-19
regulation of cell proliferation	4.70E-18

Figure 2.7 Detailed look of BRCA results.
(Left) A part of the tree enriched with genes implicated for breast cancer (level 2 and down). (Right) The GO significance and categories of the 169 gene super-cluster (grey box).

2.6 IMPROVING REKS STABILITY USING EIGENCUTS

We notice that with the high level of heterogeneity in the clinical expression data, the affinity maps we observe are in reality very noisy and are presented with a large amount of cluster crosstalks. These crosstalks contribute in part to *K*-means' inability to consistently create stable partitions, which accounts for all of the instability of the ReKS algorithm.

Chennubhotla *et al.* [78] proposed a perturbation algorithm, EigenCuts, for identifying and removing such crosstalks. This formulation takes advantage of the Markov transition perspective of spectral theories. Recall that we can view a given graph G as a Markov transition matrix. The flows of probability along the edges of the graph, which is governed by the eigenvectors of the Markov transition matrix, are referred to as *eigenflows*. We can view G as a set of coupled clusters with eigenflows flowing between the clusters. In this formulation, we can view the aforementioned crosstalks as "bottlenecks" in a system with severely restricted eigenflows. Graphs with weakly coupled clusters have eigenflows that can be characterized by a slow decay, or a long half-life. Therefore, identification of these bottlenecks can be achieved by identifying regions of the graph that is particular sensitive to perturbations to the edge weights. If changing the edge weights causes a large change to the eigenflows of the system, we can infer that this edge is a bottleneck in the system.

This idea is captured in the EigenCuts algorithm. It computes the eigenvectors of the graph Laplacian and calculates their respective half-lives. For eigenvectors with long enough half-lives, their half-life sensitivities for each edge in the graph is computed. Within a given neighborhood, an edge with lowest sensitivity that is also below a certain threshold is removed. This process can be repeated iteratively until well-separated clusters form. At this point, we are presented with a modified affinity matrix with crosstalks removed.

We omit the derivations here and instead refer the readers to [78] for detailes. The inclusion of the EigenCuts algorithm to ReKS is still ongoing work, and we present here a snippet of the preliminary results, shown in Figure 2.8. In this example, EigenCuts was applied as a preprocessing step before performing ReKS on the ALS benchmarking dataset described in Section 2.4.2. We started out with the original, noisy affinity map on the top right and "cleaned up" the matrix in successive iterations. We can see that the crosstalks between the clusters are gradually removed. We can compare the performance of ReKS (measured by *cR* described in the previous section) against that of ReKS with the EigenCuts preprocessing step, across different parameter combinations (Figure 2.9). In this particular data, we can see that ReKS performance with EigenCuts preprocessing improve substantially toward the lower left as well as another isolated region of the parameters. Initial analysis reveals rather inconsistent patterns both across parameter landscape and across datasets. Further work is required to determine the suitability of EigenCuts for clinical expression data, and to determine proper parameters and interpretations.

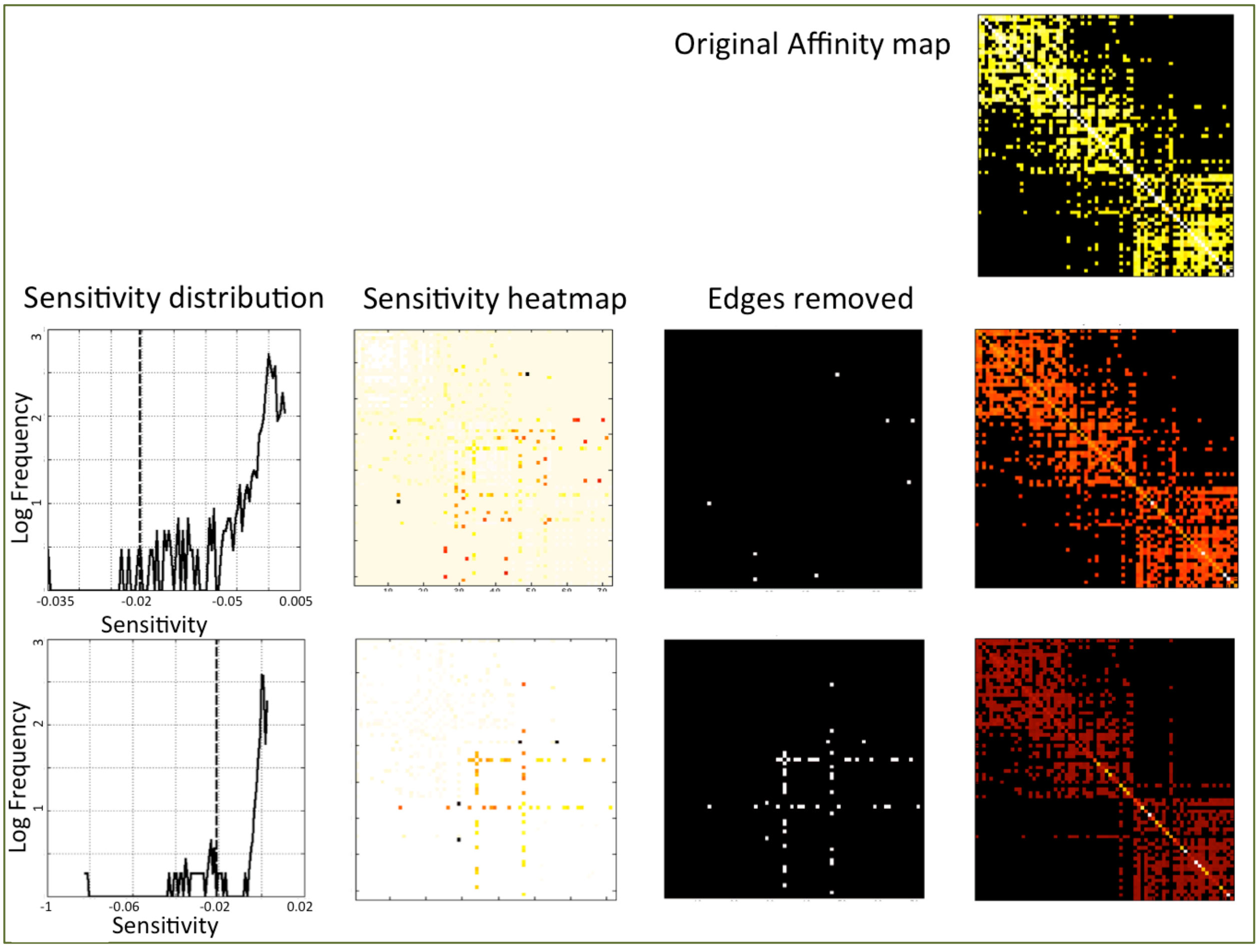


Figure 2.8 An example of the EigenCuts algorithm.

The original affinity map for the ALS dataset is displayed on the upper right corner. A distribution of the sensitivity values of all edges is plotted, and all the edges below the cutoff threshold (dotted line) and are strictly minimal within a neighborhood are removed. The corresponding sensivity are plotted on the the affinity map, and the edges flagged to be removed are indicated in white. The resulting affinity map from this iteration is plotted on the far right. Iterations 1 and 5 are plotted.

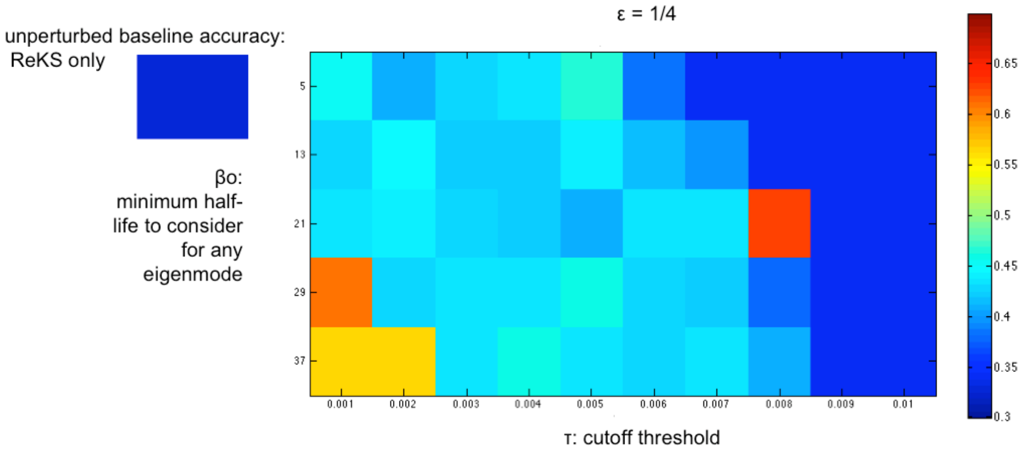


Figure 2.9 An example of EigenCuts performance across two parameters. EigenCuts was applied across five minimum half life parameters (5 to 37) and ten sensitivity cutoff thresholds (0.001 to 0.01). The performance of the original ReKS without perturbation is plotted on the left for reference.

2.7 DISCUSSION AND FUTURE DIRECTIONS

In this study, we demonstrate the utility of a new recursive spectral clustering method we proposed as an alternative to traditional methods for clustering large-scale, human disease expression data. Consistent with previous findings [88], hierarchical methods are faster but perform relatively poorly. *K*-means methods can be accurate when the number of groups *K* is known. However, in the case of gene clustering of disease samples we are rather agnostic as to the number of the clusters we should expect. ReKS does not require the number of clusters to be known *a priori*, and is an order of magnitude faster than the original *K*-means algorithm. Also, compared to *K*-means spectral, ReKS enjoy a considerable speed gain by performing the decomposition and clustering iteratively, while maintaining a comparable performance even without directly minimizing the overall inter- and intra- cluster distances. We demonstrated the superior performance on clustering patients using expression data, and we briefly introduced an algorithm, EigenCuts, that can be used to further improve stability of ReKS by removing noise in the affinity map.

By incorporating prior pathway information in the algorithm, ReKS additionally guides the clustering process toward a more biologically meaningful partition. We showed that the clusters obtained are biologically relevant in their enrichment in GO terms, and the size of the clusters has a more natural distribution than that of *K*-means or hierarchical clustering partitions. The clusters, being rather compact and constrained in size, could then be used in subsequent studies, where clusters of genes could potentially be used as predictors for disease classification. Not only does ReKS provide a partition of the gene space, the resulting tree structure provides a hint to the relative tightness of the clusters and potential targets. In the future, we wish to investigate the relationship between the relative position of the cluster in the tree and their potential strengths in classifying disease labels and other clinical variables. Also, it is possible to automatically calculate the optimal number of neighbors to be considered in each recursion level. For example, we can use an approach similar to eigengap, where the distribution of similarities for each node will be compared to the global distribution to identify the optimal number of informative neighbors. The above results indicate that, when applied to large clinical datasets, recursive spectral clustering offers an attractive alternative to conventional clustering methods.

3.0 T-RECS: TREE-GUIDED CLUSTER SELECTION

Large-scale, high dimensional datasets have become increasingly abundant in the biomedical community. A variety of feature selection methods have been developed to tackle the issue of high dimensionality, with a goal of extracting a set of features that is minimal but still maximally predictive of the outcome.

An issue that has been relatively neglected in feature selection algorithms is the *stability* of the methods. Stability is a measure of the sensitivity of a method to variations in the training set. Traditional feature selection algorithms applied on high-dimensional, noisy systems are known to lack stability [96]. In addition, in the biological system that we study, a large amount of co-linearity (redundant information) exists in the variables. It is possible that different training samples lead to vastly different variable sets that regardless yield the same predictive performance.

High-throughput measurements of clinical samples have very recently been made available through the efforts of several large consortiums with unprecedented sample sizes, ranging from hundreds to thousands of patients [20], [22]. The motivation for developing a more stable feature selection stems from the need to provide biomedical domain experts with features that are minimal, discriminative, and relatively robust to variations in training samples. This is of paramount importance as the selected features could provide a basis to distinguish different types of pathologies in the clinical samples, or deliver prognosis of patient outcome and survival.

Additionally, they pave a way for hypothesis construction for the underlying mechanisms of biological processes for pathogenesis, a starting point often followed by considerable amount of efforts and time in laboratory research. The burden falls on the shoulders of computational biologists to provide feature sets that are stable, minimal in size and maximally discriminative.

The goal of this work is two-fold: to improve the understanding of the underlying system through the process of stable feature selection, and to provide a predictive model of disease classification or survival estimation. To this end, we propose a novel feature selection algorithm that accepts continuous data as input, and produces clusters that are predictive of categorical or survival outcome labels. We apply this algorithm to gene expression input data from patients and use disease subtypes or patient survival as the output variables. To bypass multiplicity and promote stability, our proposed framework treats clusters as single entities for feature selection, and builds upon an existing feature selection algorithm based on local causal Markov Blanket induction called Max-Min Parents and Children (MMPC) [97], [98].

Our framework is motivated by a key hypothesis that biologically meaningful biomarkers may not be maximally discriminative, but could be highly correlated with predictive features that lack biological interpretation. By first clustering the variables, we anticipate that these meaningful biomarkers will be revealed by virtue of association with statistically discriminative variables. The resulting algorithm, termed Tree-guided Cluster Selection (T-ReCS), is sound and can efficiently process large datasets in the range of tens of thousands of variables. Additionally, it is computationally efficient without imposing strict requirement for training size, which makes it suitable for high-throughput biological data.

The chapter is organized as follows. We briefly survey existing single and group feature selection methods in Section 3.1. A special type of feature selection, local causal discovery, and relevant concepts are introduced in Section 3.2 as preliminaries for our algorithm. In particular, we provide a review of the MMPC algorithm that our work builds upon in Section 3.2.5. In Sections 3.3 and 3.4, we introduce our algorithm and evaluation metrics. Results from simulated experiment and application to clinical data are provided in Section 3.5. Finally, we provide conclusions and future directions in 3.6.

3.1 VARIABLE SELECTION METHODS

3.1.1 Traditional variable selection methods

Traditionally, feature selection has been viewed as a problem of searching for an optimal subset of features in order to maximize some evaluation measures. Feature selection methods broadly fall into three categories: filter, wrapper, and embedded methods [99]. Filter (univariate) methods use measures of intrinsic data characteristics and select subsets of variables as pre-processing step, independent of the classifier. *t-test*, Pearson correlation, entropy and other similar statistics computed from empirical distributions generally fall into this category and have been applied toward biomarker discovery due to the ease of use and the rather straightforward intuition behind them [7], [100]. This strategy does not work well when features highly correlate or interact with each other, however. Wrapper methods, on the other hand, integrate the classification/prediction step and score subsets of variables according to their predictive power, selecting the joint set of variables with maximum performance in cross validation [101]. Examples include Linear Discriminative Analysis (LDA) and logistic regressions [102]. Since searching through the space of all combination of features is generally computationally infeasible, heuristics are often utilized to prioritize the feature sets tested. Embedded methods such as random forests and Support Vector Machine (SVM) perform feature selection in the process of the training step, and are often specific to the classifier in use [103], [104]. A representative model is Recursive Feature Elimination using SVM (SVM-RFE) [104].

3.1.2 Efforts toward stable, non-redundant, or group feature selection

While all of these methods have enjoyed varying degree of success, none of them was deliberately designed to achieve stable results. Some efforts have been extended toward improving this aspect of feature selection process [96]. Statnikov *et al.* described the TIE* algorithm [59] for resolving *in silico* redundancy for Markov Boundary discovery. In this paper, they provide a careful treatment of the concept of "bioequivalence" and a detailed discussion of the notion of molecular signature multiplicity. The method they proposed iteratively discovers maximally predictive Markov boundary and then removes the set from consideration in the next iteration. Similarly, Tuv *et al.* [105] introduced a redundancy elimination procedure that also directly aims to generate a compact set of non-redundant features. While it is possible to detect weaker features through this process, the goal of this type of methods differs from ours in that relationships between redundant features are not explicitly defined thus it is more difficult to yield biological interpretation, and the issue of stability is not directly addressed. Yu *et al.* proposed a strategy for selecting stable features via dense feature groups identified by kernel density estimation [106]. This strategy, while novel, is sensitive to the bandwidth of the kernel estimation and is limited to the dense feature groups selected and may not include some of the most relevant features in individual feature rankings, especially in the sparse regions of the data.

Group Lasso [107], [108], which estimates sparse linear models by minimizing an empirical error penalized by a regularization term, is a very popular and successful approach in statistics and machine learning. This formulation attempts to balance a trade-off between accuracy in data fitting and parameter regularization. The sparse solution is suitable as the number of variables (genes) far exceeds the number of samples. A popular formulation is given by Lasso with constraints on the parameter vector using L-1 type penalty. The group Lasso is an extension where the variables are partitioned into groups and the goal is to select groups of covariates, as we wish to do with the modified MMPC algorithm. Yuan and Lin [107] proposed the group lasso penalty where constraints are imposed on the sum of the L-2 norms of the parameter vectors of the different groups of covariates, where the L-2 norm ensures that sparsity is induced on the group level. This penalty could be viewed as an intermediate between L-1 and L-2 type penalties. The drawback of this approach is that an explicit partition (clustering) has to be supplied to the model.

Perhaps a strategy most analogous to ours is a method proposed by Hastie *et al.* [109]. They first create a clustering tree over the variables using hierarchical clustering. Next, using a forward stepwise selection method, they gradually include clusters represented by average expression profiles into the linear model. However, this method requires iterating over all clusters (roughly twice the number of variables) and relies on hierarchical clustering which we have shown in the previous chapter to be inferior to our clustering method for high-throughput clinical data. Additionally, the level at which clusters are selected are based purely on the predictive performance in cross validation, and does not provide any biological insights as to whether or not the features are indeed coherent at that level. More importantly, this method, along with several similar ones [65], [110]–[112], does not address the stability of feature groups.

Finally, ensemble methods have been proposed. In this approach, instead of relying on a single (unstable) set of feature selection result, a committee of feature selection results is built to find the optimal feature set. This can be done either by a) perturbing the data (using random samplings of the original data) or b) using a diversity of feature selection algorithms to aggregate results. Two recent studies investigated the stability issue of feature selection under small sample size, and recommended empirically choosing the most stable feature set by repeated sampling of the training data. The computational overhead is high, and this strategy is at best only as stable as the pooled results of existing feature selection algorithms employed [113], [114].

3.2 LOCAL CAUSAL INDUCTION

Causal structure learning is a particular flavor of feature. This is an emerging, successful approach that performs variable selection in the form of identifying the *Markov Blanket*, or a minimally predictive variable subset, of the target variable. This subset can be regarded as the variables we wish to select for a given target variable. We first introduce the basic concepts and general algorithmic framework for this formulation, and present an overview and theories of a particular algorithm that falls into this category: the MMPC (Max Min Parent-Children) algorithm [98].

3.2.1 Bayesian networks

To facilitate our discussions, we must first briefly introduce the definition and semantics of Bayesian network [115], the representation framework in which our algorithm is based on. A

Bayesian network is a probabilistic graphical model in which variables appearing in a dataset and their conditional independencies are statistically represented through a directed, acyclic graph (DAG) $G =< \Phi, E >$. We show an example of such Bayesian network in Figure 3.1(Left). Here, the nodes Φ are random variables that could either be observed or latent variables, parameters, or even hypotheses. Conditional independences between these variables are encoded by edges; nodes that are not connected by edges are (conditionally) independent of each other. Each node has a probability distribution that depends on the instantiation of its parents. In other words, given the values of the node's parent variables, a probability function can be defined to output the distribution of values that this node can take on. One implication of this definition is that an edge between two nodes corresponds to causal influence under broad conditions. Every edge from a variable X to a variable Y encodes a probabilistic and direct cause from X to Y.

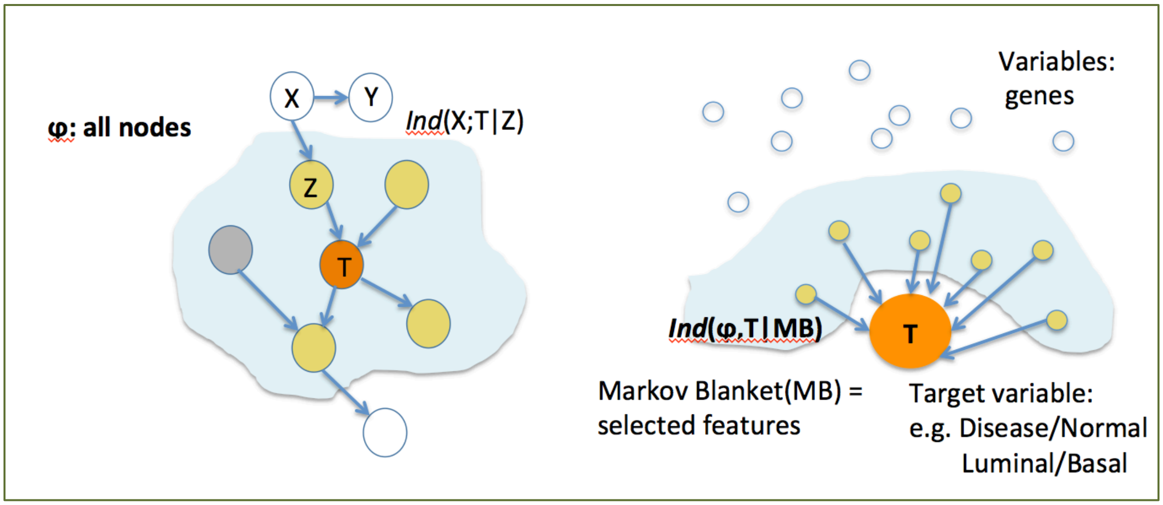

φ: all nodes
X
Y
Ind(X;T|Z)
Z
T
Variables: genes
Ind(φ,T|MB)
T
Markov Blanket(MB) = selected features
Target variable: e.g. Disease/Normal Luminal/Basal

Figure 3.1 Illustrations of a Bayesian Network and Markov Blanket.
(Left) An example of a Bayesian Network, and an example of conditional independence between X and T given Z. The Parent and Children set of target variable T(in red) are the variables in yellow, and the Markov Blanket(shaded in blue) additionally includes the variable in gray. (Right) Biomarker selection represented as a Markov Blanket discovery problem. Genes to be selected are represented as nodes, while edges represent causal relationship between the expression level of genes and a clinical variable to predict (T). The goal is to discover the subset of genes that lie within the Markov Blanket (shaded in blue).

A basic component of the probabilistic relations in a Bayesian network is *conditional independence* between variables. We denote conditional independence between two nodes X and T given a set of variables Z as $Ind(X;T|Z)$, and analogously, *conditional dependence* between them given Z as $Dep(X;T|Z) \equiv \neg Ind(X;T|Z)$. Two variables, X and T, are conditionally independent given Z if and only if $P(T|X,Z) = P(T|Z)$. A stronger criterion for independence is d-separation.

Formally, the joint probability distribution J of the data is related to the graph G of a Bayesian network through the *Markov Condition* property [116]. This property states that a node is conditionally independent of its non-descendants given its parents. Additionally, a node in a Bayesian network is conditionally independent of the entire network given its *Markov Blanket* - a special set of neighbors that include its parents, children, and spouses. A *faithful* Bayesian network is one in which only the independencies that are entailed by the Markov condition hold in J. There can be many graphs G that are faithful to a given distribution J. However, they all share a unique set of parents and children (neighbors) of a variable T.

A particular property of the Bayesian network that we would like to exploit is that, for every variable T in the network, we will find a special subset of variables that, given knowledge of their values, the probability distribution of the variable of interest T can be determined, and knowledge about other variables becomes redundant. This set of variables coincides with the MB of T that we just introduced [117]. Since all information for optimally predicting T is contained within the MB(T), it seems that the task of feature selection can be roughly translated into discovering the MB of a given variable T. In the next section, we demonstrate how we can pose the feature selection problem as a Markov blanket discovery process.

3.2.2 Feature selection as a Bayesian network structure learning problem

How can we define the biomarkers selection problem in the context of Bayesian network? In this representation, all the features to be selected (e.g., genes), are nodes in a network, and an *output* or *target* variable T (e.g., disease subtypes or survival time) that we attempt to predict is also a node in the network, as demonstrated in Figure 3.1(Right). Directed edges between nodes represent causal relations. As stated earlier, the goal is to discover a minimal variable set that is within the Markov Blanket (MB) of *T*. Markov Blanket of a variable T consists of the parents, children, and spouse nodes of T. All the nodes outside of MB are conditionally independent of T when conditioned on the MB. In other words, the MB(T) shields T from the influence of the rest of the network. Given the states of nodes within the MB, we will be able to predict the state of T, without knowledge of the rest of the nodes. In this way, identification of the MB could be viewed as a feature selection process. In fact, it was shown that under certain broad conditions, the Markov Blanket is the solution to the variable selection problem [118].

There are multiple advantages in using this kind of framework for our problem. The graphical representation lends an intuitive and visually descriptive way to summarize the result. We also believe that its graphical representation is particularly suitable for incorporating context information such as those demonstrated in Chapter 1, given that additional regulatory information that will be added to the selected features are best represented as directional graphs. As we will see later, popular MB induction methods employ heuristics that provide for a modular environment in which different types of hypothesis tests could be easily plugged in to suit the different types of datasets one is interested in. This is especially welcomed for the purpose of easy code maintenance and extension. Finally, the theories and nature of the MB induction heuristics facilitates the development of our computationally efficient extension to group feature selection.

Algorithms for learning the structure of a Bayesian network generally fall into two broad categories: *constraint-based algorithms* and *search-and-score* algorithms. Constraint-based algorithms view a Bayesian network structure as a representation of independencies, and rely on statistical tests to identify structures that are consistent with the conditional independencies encoded in the data. The drawback, however, is that the success of the method hinges on the accuracy of the conditional independence tests, and failure of even one test could lead to an inaccurate structure. Search-and-score algorithms, on the other hand, view a Bayesian network structure as parameters for a statistical model and cast the structure-learning task as an optimization problem. First, a *hypothesis space* of potential models is defined, representing the set of possible network structures we consider. Given a scoring metric that measures how well the model fits the observed data, the goal is to search through the hypothesis space for maximum-scoring structure(s) using heuristic search techniques. This type of methods maximizes a score for the entire structure, therefore it is less prone to local mistakes, and provides a direct way for regularization to prevent over-specifying the model by incorporating too many edges. The problem with this type of methods is that the solutions are often not efficient, when searching through a combinatorial space with a super-exponential number of structures.

Algorithms for learning a complete Bayesian network generally do not scale up beyond the range of hundred to thousand variables [119]. Moreover, in the case of variable selection we are only interested in learning the structure around the variable of our interest. Thus, we focus our attention on *local* structure learning, which not only provides for a scalable alternative to learning the entire Bayesian network, but also directly addresses our pursuit of variable selection for classification.

3.2.3 Local structure learning

The goal of local structural learning is to discover only the local causal structure around a variable of interest T. We state the goal formally: given a probability distribution J faithful to some BN and a node of interest T, identify 1) the set of parents and children of T, $PC(T)$, or 2) the Markov Blanket $MB(T)$. Identifying the $PC(T)$ is equivalent to identifying the direct causes and effects of T. In the BN representation, this is equivalent to identifying the incoming and outgoing edges to T. Aliferis *et al.* [120] presented a comprehensive overview of the general algorithmic framework for learning such local causal structure around the target variable of interest.

While we are interested in discovering the Markov Blanket, we are not as interested in defining the exact causal relationship between members of the MB and T. In other words, we are focusing on discovering the skeleton of a BN only, and not orienting the edges. As an example, we may discover a biomarker gene set G that is represented as the MB of a clinical label T. While the identities of the gene set G is important to us, deciphering exactly which genes cause the observed clinical label T, and which gene expression changes are caused by T, is of less interest to the biomedical community, and is in reality difficult to distinguish in practice. Furthermore, whether one should undertake a strict causal interpretation for the exact directions of the edges inside the MB is open to debate, and is beyond the scope of this work. Therefore, this work is focused on only identifying the set G (*existence* of an edge connecting to T) and not on establishing the direct cause and effects (*direction* of the edges) of T. In other words, we are only interested in identifying the skeleton of a Bayesian network – the set of undirected edges encoding potential dependencies between the nodes.

3.2.4 Skeleton identification methods

A number of undirected skeleton identification methods have been proposed. We briefly introduce three that are highly representative of their respective strategies: PC [121], Max-Min Parents Children (MMPC) [97], [98], and Three-Phase Dependency Analysis [122]. Spirtes and Glymour proposed the PC algorithm as one of the first skeleton identification methods. It considers the hypothesis tests in increasing order of conditioning set size until tests can no longer be performed due to restrictions of sample size[2] at which point an edge is included by default. MMPC is the skeleton identification component of a family of algorithms [64], [98] and pioneered the two-phase approach for identifying the parents and children of T. It uses the Max-Min heuristic to include variables to the conditioning set, and eliminate any false positives in a second stage. The TPDA algorithm is unique in that it uses tests of conditional mutual information instead of hypothesis tests to determine independencies. It exhaustively runs all the tests through a three-phase process. All three strategies have been adopted and applied to different areas [123], [124].

Constraint-based skeleton identification algorithms such as MMPC and PC utilize a series of statistical decisions, or hypothesis tests, to inform the addition or exclusion of an edge to the skeleton. In hypothesis testing, we have a null hypothesis that is usually denoted by H_0. In the particular case of the conditional independence tests, the null hypothesis is that the variables are independent given a set of variables. We want to test whether the data support this null hypothesis. Specifically, the hypothesis test takes as input data D, and outputs an "*accept*" or "*reject*" decision. To evaluate this decision, one can analyze the probability of false rejection probability p of the null hypothesis. A standard significance threshold of $p \leq 0.05$ is usually given for a hypothesis test. With this significance threshold, we reject the null hypothesis that the variables are independent if the probability of observing the event by random is smaller than 0.05. If we fail to reject the null hypothesis, then we determine that the variables are independent, given the conditioning variable(s). This framework allows for a decision rule that

[2] Their rule of thumb is that the G^2 test is reliable if there are five or more instances per degree of freedom of the test

acts as building blocks of the local learning algorithms. We delay the discussion of how to design such a rule in Section 3.3.2.

3.2.5 Max-Min Parent Children (MMPC)

We introduced in the previous section the general framework of local causal discovery. MMPC is one such algorithm for learning local causal structure around target variable of interest. Given a set of features Φ and a target variable T, MMPC aims to discover $PC(T)$, the parent and children of T using a two-stage process.

The reader might wonder why we only focus on discovering the $PC(T)$, and not the full set of MB. It was shown that the full MB does not improve predictions substantially when compared to only the parent and children set, while requiring significant computational overhead [98]. Up to 200-fold increase in computing time was observed when running MMMB, the algorithm for discovering the full set of MB, compared to MMPC. We determine that MMPC should achieve the balance between approximating the minimum set of variables that best predict the target variable T and computational efficiency. We also choose to adopt the MMPC algorithm as 1) extensive testing was done over a large collection of datasets of different sizes and characteristics, and it was shown to have comparable or superior performance over other structure learning algorithm such as Incremental Association Markov Blanket (IAMB) [125], the Grow-Shrink (GS) algorithm [126], and the Koller-Sahami algorithm (KS) [127] it is sound in the sample limit and scales up to datasets with thousands of variables, and 3) the code for MMPC was open-source and made readily available for easy modification and adaptation.

MMPC uses a two-stage strategy for PC induction. In Phase I, a series of local statistical decisions are used to efficiently identify conditional independence relations among variables.

The idea is that if one could find a set of variables such that two variables could be shown to be independent conditioned on this set, there should not be an edge connecting these variables in the final model structure. Therefore, we exclude from our consideration structures that contain an edge between these variables. If we cannot demonstrate that two variables could be made conditionally independent, then an edge is added to the skeleton between the variables. The skeleton structure elimination process acts as a way to constraint the heuristic search, and is thus considered constraint-based in this phase of the algorithm.

We present the pseudocode of MMPC in Figure 3.2. Initially, the candidate Parents and Children (CPC) set is empty. In Phase I, or the *forward phase*, we seek to 1) eliminate any variables from consideration that achieve *minimum association* (independence) with T even without conditioning on any variables, and 2) produce a candidate list of CPC by including variables that are most associated with T conditioning on s, some subset of CPC. We use $assoc(X; T|s)$ to represent the strength of association of X and T given s, and denote

- $Ind(X; T|s) \Leftrightarrow (assoc(X; T|s) = 0)$
- $Dep(X; T|s) \Leftrightarrow (assoc(X; T|s) \neq 0)$

Calculation of $assoc(X; T|s)$ is detailed in Section 3.3.2. The rationale for the forward phase is the following: if a variable could be made independent of T conditioned on some subset, it does not belong in the MB, and will not be considered again. On the other hand, variables that are highly associated with T despite our best effort to make them conditionally independent of T should be included into the CPC.

```
MMPC($D; T; k; a$)
// **Input: Data** $D$ with all variables $\phi$; Target $T$; maximum conditioning set size $k$;
threshold for rejecting independence $a$
// **Output:** Parent and Children set $PC$

// **Phase I (forward)**
$CPC = \emptyset$ // Initialize the temporary $PC$ set
$R = \phi$ // variables to consider
**if** $\exists\ s \subseteq CPC, s.t. Ind(X; T|\emptyset)$
    $R = R \setminus \{X\}$// remove $X$ from consideration
**end if**
**repeat**
  **for** every variable $X$ in $\phi$ find
    $minAssocSet(X)$ =subset $s$ of $CPC$ that minimize $assoc(X; T|s)$, $|s| \le k$
  **end for**
  $F$= variable of $\phi \setminus (\{T\} \cup CPC)$ that maximizes $assoc(F; T|minAssocSet(F))$
  **if** $Dep(F; T|minAssocSet(F))$
     $CPC = CPC \cup F$ // Include F into $CPC$
     $R = R \setminus \{X\}$// remove $X$ from consideration
  **end if**
**until** $CPC$ has not changed

// **Phase II (backward)**

**for** all $X \in CPC$
  **if** $\exists\ s \subseteq CPC, s.t. Ind(X; T|s)$, $|s| \le k$ // if $X$ can be made *d-sep* from $T$
     $CPC = CPC \setminus \{X\}$ // Remove $X$ from $CPC$
  **end if**
**end for**
$PC = CPC$
**Return** $PC$
```

Figure 3.2 The Max-Min Parents and Children (MMPC) Algorithm

The algorithm of forward phase translates into the following procedure. Initially, univariate association between each variable and T is calculated to determine an initial set of variables to seed the CPC, as well as eliminate any variables that have zero association with T ($assoc(X; T|s) = 0$). Next, even with a conditioning set that allows for the maximum independence, variables that still have high association with T are likely to be the parents and children of T, and enter the CPC. This procedure is repeated until no variables outside of the CPC set is eligible for inclusion, and this part of the algorithm terminates. This heuristic is admissible - all members of the $PC(T)$ will be included - since parents and children of T will always have dependence (i.e., non-zero association) relationship with T given *some* subset of variables. Thus, it will eventually be incorporated into CPC.

In Phase II, the backward phase, we seek to reduce any false positives in the CPC set by searching for those variables in the CPC that are independent of T conditioned on some subset s of the CPC. This elimination strategy identifies variables within the CPC that could still be made conditionally independent of T, which disqualifies their membership in the MB. This procedure is done iteratively until no variable in the CPC can be removed, at which point the algorithm terminates. At the end of the algorithm, $CPC = PC(T)$, and we arrive at a set of parent and child nodes that are features most predictive of T.

The original MMPC includes a symmetry corrections step that attempts to remove a particular type of false positive that could arise. We do not include this step in our implementation as Aliferis *et al.* [59] determined empirically that these cases are rare, and the extra computational burden of symmetry correction does not justify for the theoretical benefits. We therefore use the implementation of MMPC without the symmetry correction step.

Two parameters are required in MMPC. The parameter a controls the level of dependence we are willing to accept. The conditional test of independence returns a *p*-value; the lower the *p*-value, the higher the association. Parameter a defines the *p*-value threshold for rejection of the null hypothesis of independence. Thus, the lower the threshold a is, the more stringent the criteria for inclusion into CPC. The parameter k controls the maximum size of the conditioning set we are willing to consider. Ideally, we would like to test all possible subsets of

CPC, from the null set in the univariate case to sets of increasing complexity. In reality, however, k is limited by the number of training instances available to reliably measure association; large k would result in inadequate statistical power. To see this, if the data and the target variable are both binary, a variable with n parents will have 2^n combinations of values that the parents can take on. We very quickly run out of training instances we could use to construct a reliable measure of association. The consequence of this limit is that false positives may enter CPC. To see this, imagine that a variable requires m members of the CPC to be *d-separated* from T. If we only search through subsets of CPC up to a maximum size of $m-1$, the variable would have been associated with T without the last member of the m conditioning variables, and the variable would have been erroneously included into the CPC. Based on experiments in [98], most variables not in $PC(T)$ can be made *d-separated* from T using less than 4 variables. Also, most distributions can be described by sparse networks, which suggests that a small value of k is sufficient for the algorithm[3].

The complexity of MMPC is roughly $O(|\phi||PC(T)|^k)$ tests of conditional independence, assuming the final PC output is roughly the maximum size of s in any iteration. Assuming a sparse structure (small number of variables in the PC set), the complexity grows almost linearly to the number of variables.

[3] Based on conclusion from [134], MMPC does not perform independence tests unless there are at least five training instances on average per parameter to be estimated.

3.3 TREE-GUIDED CLUSTER SELECTION (T-RECS)

3.3.1 Method overview

In this Section, we describe how we adapt an existing variable selection method, MMPC (described in detail in Section 3.2.5), to construct a group variable selection framework, T-ReCS.

T-ReCS has two components. The first is a clustering step that partitions the feature space into a tree structure. At the root of the tree is the set of all variables, and each leaf at the bottom of the tree is a single variable. Each internal node in the tree represents a cluster of variables – in our case genes – whose values (expression) are correlated. The lower in the tree a node is, the more similar the patterns of its members are. Conversely, the nodes close to the root of the tree are large agglomerates and consequently these members are not expected to display similar behavior across samples. We can indeed observe this trend from a later example in Figure 3.8. For our implementation, we use the tree produced by ReKS from Chapter 2 for this part of the algorithm.

The second component of T-ReCS is feature selection operating on the tree structure obtained in the first part of the algorithm. The idea is that the clusters obtained previously are treated as entities for feature selection, and we iteratively explore supersets of a selected feature to see if the same predictive properties are retained as we traverse up the tree. Every node of the tree is "collapsed" into a single vector representing a latent variable X'. This latent variable is used to represent all the members of the nodes. This is akin to the concept of "eigengene" that Langfelder and Horvath have proposed [128]. The difference is that here, a partition is first created and then a function is applied to members of the partitions to represent the resulting latent variable. We expect that the latent variables will cease to represent a predictive feature adequately and would lose its predictive property as we traverse up the tree. Details of the various cluster representation methods are found in Section 3.3.3.

An obvious way to accomplish this goal is to perform feature selection at varying variable granularity, i.e. select latent variables that are created at different levels of the tree, and then construct a predictive model for each combination to determine performance, as Hastie proposed [109]. However, as we mentioned earlier this approach is very inefficient given that a ReKS tree typically produces 15-25 levels. Furthermore, it is not clear what is the best way to yield a partition from a given ReKS tree for feature selection purposes. In Chapter 2, we used the maximum cluster size to guide us in choosing an appropriate partition. For the purpose of selecting features that remain predictive of T, there is no reason to believe clusters of certain size would remain congruent.

Instead, we take advantage of the Markov Blanket/Parent-Children induction heuristics in the original MMPC algorithm, and employ a strategy that not only automatically determines a suitable partition level for the tree, but also gives us remarkable savings in computational efficiency. The strategy is detailed as follows: a round of MMPC is first performed at the leaf level of the tree. This is equivalent to conducting the original MMPC to select predictive single variables. For each predictive leaf X selected, we test to see if they can be replaced by their parent node X'. This is accomplished by concurrently evaluating two tests of conditional independence: $dep(X'; T|S)$, where $S \subseteq \{ PC\backslash\{X\}\}$, and $Ind(X; T|X')$. Essentially, we are testing to see (1) whether or not X' remains associated with T, conditioning on subsets of the parent and children set excluding its descendant, X. If X' can be made conditional independent of T given the current conditioning set, then X' does not belong to MB, thus X' is not an adequate replacement for X; and (2) whether or not X can be made *d*-separated from T conditioning on X'.

If X becomes independent of T given all the information X' already provides, that means X' contains information equivalent to X, and we can safely substitute X with X'. In the algorithm, we will only replace X with X' and advance to the next level of the tree if both of these conditions hold true. If a X' meet the criteria of both of the tests, we then check to see if the same conditions hold true for its parents X'', relative to X. This procedure is continued iteratively until at least one of the conditions is violated, at which point the procedure terminates and returns the last group variable that is still representative of X. After this process is repeated for each selected leaf, the algorithm returns a set of selected single or group features that are seeded from the single variables selected in the MMPC round, each with varying sizes and located at different levels of the tree. Note that with this strategy, we will only explore the parent nodes of the variables that were selected at the initial round of MMPC and not the rest, operating on the assumption that other leafs will never become predictive of T since 1) if a variable is un-predictive, the only way it could become predictive is to become aggregated with predictive features. However, those predictive features would have been selected in the MMPC round, and we would have arrived at the same node from those predictive features anyway; and 2) if a variable is predictive, it would only lose its predictive property as more and more members enter and "average out" its signal. This reasoning guarantees that we would not miss a predictive group feature by disregarding any clusters that stem from the single variables initially deemed un-predictive.

An example of T-ReCS is shown in Figure 3.3. We also include the pseudocode in Figure 3.4.

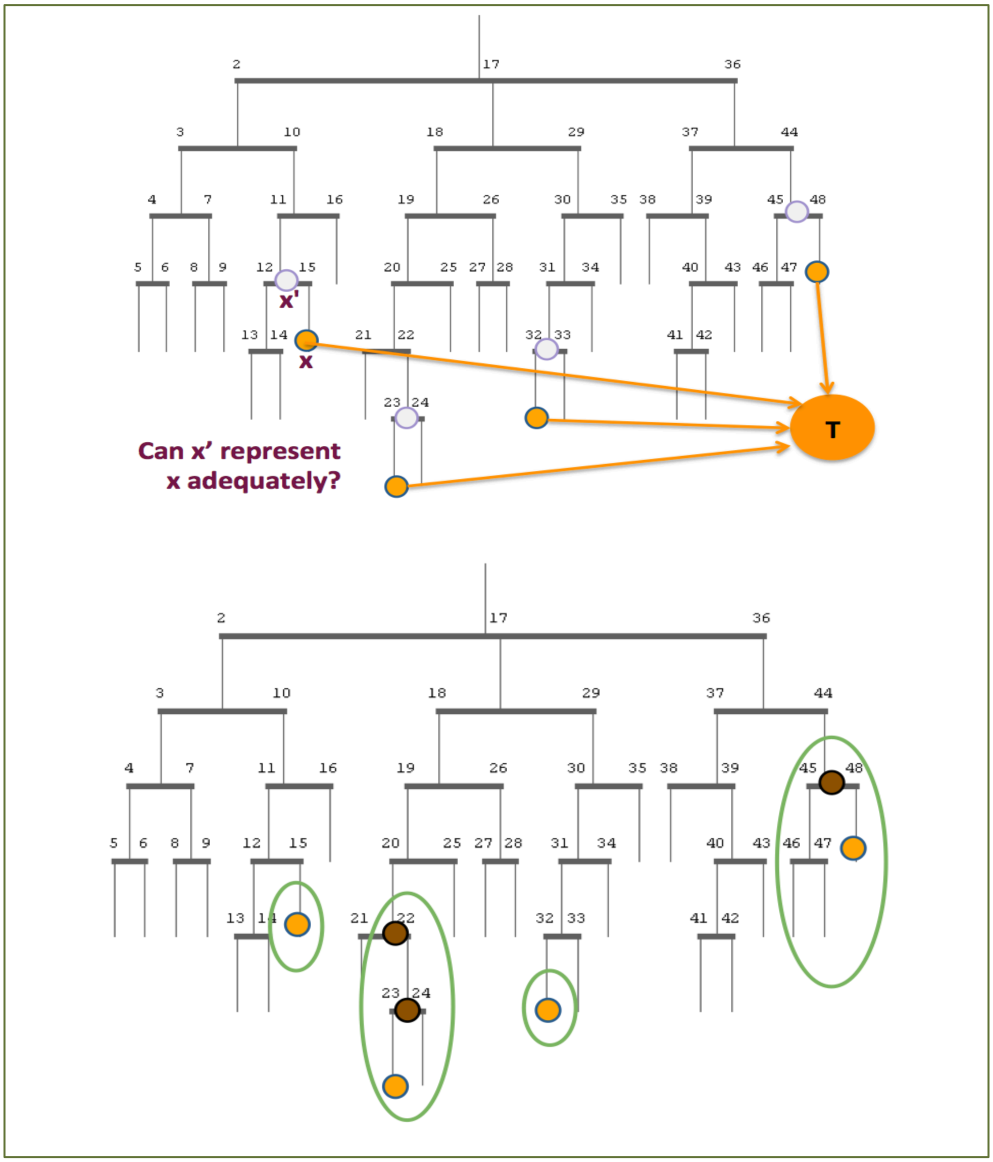


Figure 3.3 An example to illustrate the selection procedure of T-ReCS.

(Top) We first generate a tree structure from ReKS to partition the variable space. Single variable selection is conducted and single variables(leafs) that are predictive of a target variable T are marked in orange. For a predictive variable X, we test to see if its parent node represented by X' still retain the same predictive property of X. (Bottom) We employ two tests of conditional independence to determine if X' can replace X. This is done repeatedly for each selected single variable until one of the conditions is not met. The algorithm then returns the group variables marked in green circle.

```
MMPConReKS($D$; $T$; $k$; $a$, $a_{Dep(X';T|S)}$, $a_{Ind(X;T|X')}$)

// Input: Data $D$ with all variables $\phi$; Target $T$; maximum conditioning set size $k$; threshold
for single variable test $a_{single}$, thresholds for group variable tests $a_{Dep(X';T|S)}$, $a_{Ind(X;T|X')}$
// Output: set of single or group variables $G$

$P$ = ReKS($D$) // call ReKS to obtain $P$, a tree partition of the variables $\phi$
$PC$ = MMPC ($D$; $T$; $k$; $a_{single}$) // obtain predictive single variables $PC$

for every variable $X$ in $PC$
  $X' = X$ // initialize both current and parent node to the starting leaf node
  while true
    $X^c$=X' // set current node to the former parent node
    $X'$= $ParentNode_P(X')$ // define new parent node

    // first group variable conditional independence test
    for all $S \subseteq PC\backslash\{x\}, s.t.\ |s| \leq k$
      if $Ind(X';T|S)$, or $P(X';T|S)$ > $a_{Dep(X';T|S)}$
        break // do not replace $X$ with this group variable
      end if
    end for

    // second group variable conditional independence test
    if $Dep(X;T|X')$ , or $P(X;T|X')$ < $a_{Ind(X;T|X')}$
      break // do not replace $X$ with this group variable
    end if
  end while
  $G = G \cup X^c$
end for
return G
```

Figure 3.4 The T-ReCS algorithm

An observation that can be made is a phenomenon that is analogous to the concept of bias and variance tradeoff familiar to many in the Machine Learning field. The cluster size of the features selected by T-ReCS is controlled by two parameters: the thresholds for the two tests of conditional independence. Varying these thresholds result in final outputs whose cluster sizes vary accordingly. A stringent set of thresholds would prevent the procedure from advancing far beyond the single variables we start with, while moderate thresholds allow larger group variables to be selected. It is easy to see that there are two forces at play that dictate the proper range for these thresholds. At the bottom of the tree, stability of the variable selection should be low. However, the accuracy of the predictive model should be relatively high, since variables with the highest signals are used to construct the model. The other extreme is the root of the tree; stability is very high as variables are lumped into only a few clusters, yet accuracy suffers because the signals of the strong predictors have been "diluted" by other variables that join the clusters, and would not be as informative as the single variables they start out with. The goal is to find an optimal tradeoff that achieves higher stability while minimizing the loss of accuracy of the predictive model. To this end, we performed cross validation to identify the range of thresholds that produce the best accuracy-stability tradeoff. This procedure is detailed in Section 3.4.4.

For the clustering component, we reused the code from ReKS described in Chapter 2. For the single variable selection component, we used the code available from the *Mens X Machina Probabilistic Graphical Model Toolbox* (http://www.mensxmachina.org/software/). The source code for T-ReCS will be made available for download at the laboratory's web site (http://www.benoslab.pitt.edu/) and at Github.

The complexity of T-Recs is roughly $O(|\phi|^2)$: $O(|\phi|^2)$ for ReKS (See chapter 2), $O(|\phi||PC(T)|^k)$ tests of conditional independence for MMPC (the single variable selection round), and $O(log|\phi||PC(T)|)$ tests of conditional independence for the group selection.

The algorithm we described here is completely modular, and can be applied with 1) alternative clustering scheme that gives a tree structure, such as hierarchical clustering, 2) alternative statistical tests for conditional independence suitable for the data types, and 3) alternative latent variable representation.

3.3.2 Tests of conditional independence

Both the original MMPC and the T-ReCS depend on a series of statistical decisions to determine *d-separability* relations for graphical structure. In MMPC and T-ReCS, these decisions take the form of conditional independence tests. The better the test is in capturing the conditional independence relations, the better the performance one could expect from the algorithms.

Suppose we want to test whether X and Y are independent given Z. The null hypothesis is:

$$H_0: Ind(X;T|Z), or\, P(X,Y,Z) = P(Z)P(X|Z)P(Y|Z) \qquad (3.1)$$

In Section 3.2.4, we briefly described the evaluation of such test. Here, we provide a more detailed discussion of how the decision rules are designed. A common framework for creating this decision rule is to measure the deviance d from the null hypothesis of independence. A large deviance value implies that the data has little support for the null hypothesis. Another interpretation for the deviance is a measure of strength of association between pairs of variables. The larger the deviance, the greater the departure the data is from independence, and the stronger the association is between the variables.

The measure of deviance can be captured in the framework of a Likelihood Ratio test [129]. Intuitively, the Likelihood Ratio test measures the fit of two competing models. Recall that the null hypothesis of an independence conditional test $Ind(X;T|Z)$ suggests that once information about Z is given, X is not necessary for predicting T. With this interpretation, we can view a conditional independence test exactly as selection procedure between two competing models: a predictive model for T using both Z and X, and a model constructed using only Z as predictors. A log-likelihood ratio, which captures how much more likely the data is under one model versus the other, can be constructed between the two models. This ratio, or equivalently its *logarithm*, can be used to compute the p-value.

The log likelihood ratio statistic between the null and alternative hypothesis is defined as:

$$\Lambda = -2log\ (L_0/L_a) \tag{3.2}$$

where $L_{0,}\ L_a$ are the maximum likelihood for the null $(Ind(X;T|Z))$ and alternative non-independent, or $Dep(X;T|Z)$ hypotheses, respectively. Essentially, they correspond to the maximum likelihood when we fit a model using only Z as predictors $(L_0\ )$, versus the maximum likelihood of a model fitted using both Z and X as predictors (L_a). Asymptotically, Λ has a χ^2 distribution with df=1 for continuous data. The p-value can be calculated from the distribution, and the null hypothesis rejected if $p\ \leq$significance threshold a.

MMPC utilizes the Likelihood ratio framework for its skeleton identification step. As a natural extension we also base our group variable selection strategy on the same framework. We detail the tests for single variable MMPC and group variable MMPC below for select data types below.

3.3.2.1 Conditional independence tests employed by single variable MMPC

For the original, single variable MMPC, we highlight two conditional independence tests that are implemented for two types of data that are common in the task of biomarker identification. The tests for categorical data in the original MMPC implementation is described in Appendix A. We do not include them here, as they are not suitable for biomarker discovery unless discretization is first performed on the gene expression values.

- **Continuous data, categorical outcome (target variable)**

Suppose we would like to test $H_0\colon Ind(X;T|Z)$ for the case where the data (D; containing X and Z) is continuous, and the target variable T is categorical. This is the test of choice when we attempt to select genes whose expression values (continuous) are predictive of categorical clinical labels such as disease subtypes (categorical). Tsamardinos *et al.* [130] developed a conditional independence test that follows the Likelihood Ratio framework described above. The model used here is multinomial logistic regression that maps a set of continuous predictors to categorical target variable. When the target variable is binary, we have a special case of the regular logistic regression. The test statistic can be calculated by comparing the deviance of fit between the null logistic model constructed with Z only, versus the alternative logistic model built with both X and Z. Specifically, the likelihood function is:

$$L(b) = \sum_{i=1}^{n}\left(\sum_{j=0}^{J} I(y_i = j)\, v_{ij} \cdot b^j - log\ \left(\sum_{j=0}^{J} e^{v_{ij}\cdot b^j}\right)\right) \qquad \textit{(3.3)}$$

where n is the number of training instances, J is the number of categories the target variable could take on, $I(\cdot)$ is the indicator function, v_{ij} is a row vector of data, and b^j is a vector of parameters specific to category j. The test statistic then follows the $\mathcal{X}_1^2$distribution:

$$\Lambda = -2\, log(L_Z/L_{Z\cup X}) \sim\ \mathcal{X}_1^2 \qquad \textit{(3.4)}$$

Again, p-value can be calculated using this distribution, and it is negatively related to the strength of association. The higher the p-value, the lower the association. For a detailed description of logistic regression and its extension to multinomial data, please refer to [131], [132].

- **Continuous data, censored survival outcome (target variable)**

This test was introduced by Tsamardinos *et al.* in their variant of MMPC for survival data, SMMPC [133]. The test was developed specifically for right-censored survival data. This test is useful for the scenario where we attempt to select genes whose expression values (continuous) are predictive of the survival time of patients (continuous, right-censored). Again, it follows the Likelihood Ratio framework exactly as described earlier, computing a p-value that reflects the deviance statistic between the predictive model built with Z only versus one built with $Z \cup X$, and uses the Cox regression model for calculation of deviance of fit. We delay the discussion of Cox regression model to Section 3.4.1. Assuming that the reader is familiar with the notion of censoring, the likelihood function is:

$$L(b) = \sum_{i=1}^{n} \delta_i \left(v_i \cdot b - log \sum_{j \subseteq R(f_i)} e^{v_j \cdot b} \right) \tag{3.5}$$

where n is the number of training instances, δ_i is the censoring indicator, f_i is the follow-up time for individual i, and $R(f_i)$ is the set of subjects still at risk at time f_i. The test statistic can be calculated using Equation 3.4 and follows the same $\mathcal{X}^2_{df=1}$ distribution, and the p-value can be calculated using this distribution[4].

3.3.2.2 Conditional independence tests employed by T-ReCS

Conditional independence tests are also used in our T-ReCS framework to determine whether a group variable could replace a single variable while still retaining its predictive property. We use two conditional independence tests to determine if a group variable still carries the same information as the single predictive feature it includes. The two tests represent slightly different criteria: the first checks to see if the group variable carries enough extra information than the original PC set (excluding the single predictive variable this group variable contains). The second test ensures that the group variable holds information that is equivalent to that of the

[4] A local score test can also be used instead of a log-likelihood ratio test, but Tsamardinos *et al.* showed that log-likelihood ratio test produces better results than the local score test.

original single variable. A group variable has to meet both criteria to warrant a substitution of the single variable. Depending on the type of data, appropriate prediction model and test statistic described in the previous section are employed to compute the p-values.

- **Test for** $Dep(X';T|S),\ S \subseteq PC\backslash\{X\}$

This test was designed to check to see whether the dependency between the group variable X' and target T still exist, even after our best effort to condition out the association using subsets of the parent and children set PC that does not include X. If the dependency vanishes, we know that X' does not belong in MB (or PC). This is exactly the procedure that we performed in the second (backward) phase of the single variable MMPC, where we remove any variables that become d-separated from T given some subset of CPC. We can think of it as having included X' in a forward phase, and we are trying to determine if it is a false positive. Again, the p-value computed from the test statistic corresponds to the opposite of dependence. The lower the p-value, the more associated X' is to T, and the more likely X' still belongs in PC. Therefore, in order to move up the tree and have a single variable replaced by its cluster counterpart, we would like to see a low p-value output from the test. The other way to understand this test is from the perspective of the likelihood ratio test. Recall that the test statistic represents the deviance from the two competing models: $[S \rightarrow T]$ and $[\{X' \cup S\} \rightarrow T]$. A large difference implies that X' is providing extra information than the conditioning set alone. Since a large value of the test statistic corresponds to a low p-value, while the p-value remains within a defined threshold $a_{Dep(X';T|S)}$, we can safely assume that X' should still belong to the PC set.

- **Test for** $Ind(X;T|X')$

This test is designed to check that the single variable X and group X' contain the same information with regard to predicting T. It is rather intuitive to assume that if X' and X are similar enough with respect to their predictive properties, we can replace X with X'. To determine their similarity, we can again take advantage of the likelihood ratio framework. If the deviance between the two competing models $[X' \rightarrow T]$ and $[\{X \cup X'\} \rightarrow T]$ is small enough, then we know X and X' are redundant, and using both of them to predict T does not produce any extra information than using one of them alone. Since small deviance corresponds to large p-value, we can keep advancing up the tree while the p-value remains larger than a defined threshold $a_{Ind(X;T|X')}$. In other words, X' needs to be conditionally independent of X in order for the substitution to be valid.

3.3.3 Choosing latent variable representation

Up until now, we have not specified how a group variable X' can be constructed from its members. In this section, we describe several strategies that one can use to collapse a set of variables into a single-vector representation. In some cases, we explicitly think of this representation as a latent variable that generates the "observations" that we can measure from its members. Note that other representations are possible.

3.3.3.1 Centroid

Centroid is perhaps the most obvious way to represent the average behavior of a cluster. Formally, the centroid of a group of variables $\{X_1, X_2, \ldots X_n\}$ in $\mathbb{R}^n$ is:

$$C_X = \frac{X_1 + X_2 + \cdots + X_k}{k} \qquad (3.6)$$

It is the geometric center of the n-dimensional space that each variable (gene) lies on. It has previously been applied on gene expression data, for example in a *fuzzy* *K*-means clustering algorithms [134] for representing a group of similarly expressed genes. The common concern with using the centroid is that it could be subject to the influence of an outlier member. In our case, we are less concerned with this problem as the cluster was created from our previous clustering algorithm and we expect members of a cluster to display similar behavior especially at the bottom portion of the tree.

3.3.3.2 Medoid

Medoid is another way to represent a cluster. It is a member of the cluster whose average dissimilarity to all the members in the group is minimal. It is similar in concept to centroid, but note that a medoid is always an actual member of the cluster. The advantage of this is that we have an explicit interpretation of the cluster representation, but as we will see later this property may invalidate some of the assumptions we make with our method. The concept of a medoid of a group of genes has been applied in the computational biology community as well [135]. The medoid is defined as:

$$M_X = argmin_{m \in X} \ \textstyle\sum_{i=1:k} d(m, X_i) \tag{3.7}$$

where d is a distance metric.

3.3.3.3 Principle Component Analysis (PCA)

Principle component analysis (PCA) [136] is a dimensionality reduction technique that takes a set of higher-dimensional data and transforms them into a lower dimensional form, while preserving most of the information. Mathematically, PCA tries to find an orthogonal linear transformation W that projects the data onto a lower dimensional linear space, such that the variance of the projected data is maximized. The projection is performed such that the greatest variance lies on the first principle component, and the second greatest variance on the second principle component, and so on. In our case, we want to represent a group of genes using only one vector. Therefore we use the first principle component, which contains the highest amount of variance, to represent the group. The first principle component is given by

$$P^{(1)} = UW^{(1)} \tag{3.8}$$

Where U is the samples (patients) in k-dimensional space (k genes), and $W^{(1)}$ is the first loading vector that can be found by maximizing the projected variance subject to the constraint that w is a unit vector:

$$W^{(1)} = argmax_{||W||=1}\{||UW^{(1)}||^2\} \tag{3.9}$$

W is essentially the eigenvectors of $U^T U$ and $W^{(1)}$ corresponds to the eigenvector with the largest eigenvalue. PCA was used to construct eigengenes mentioned earlier [128].

3.4 EVALUATION METHODS

In the following section, we will discuss how we could evaluate the performance of T-ReCS and introduce several properties of interest. The primary goal of T-ReCS is to increase the stability of traditional single variable selection method while incurring minimal loss of predictive accuracy. We would like to investigate how varying the significance thresholds of the conditional independence tests, $a_{Dep(X';T|S)}$ and $a_{Ind(X;T|X')}$, affects the performance of T-ReCS, against the baseline performance of single variable selection. Thus, we are interested in measuring 1) accuracy, 2) stability, and 3) cluster size. We describe in detail how each of these is defined in the following section. Additionally, we describe the cross validation procedure for parameter selection.

3.4.1 Accuracy

We are interested in how good a set of selected features is in predicting T. In order to do so, we need to couple the feature selection method with a regression or classification method to produce a predictive model. Depending on the types of data, the predictive models and performance metrics differ. The predictive models and the corresponding accuracy measure for the two types of data introduced in Section 3.3.2 are defined below.

3.4.1.1 Continuous data, categorical target variable : SVM and AUC

With categorical target variables, we pair our feature selection method with a *classifier* that takes continuous predictors as input and outputs a label assignment. Many classifiers exist that accomplish this goal. Here, we adopt Support Vector Machine (SVM) as our classifier of choice, as it is practical, scalable, and was shown to have competitive performance for this type of data [120]. Given a set of training samples that we can imagine as points in space, SVM attempts to identify a decision boundary that maximally separates the data points of different categories with a margin that is as wide as possible. For detailed description of SVM, please refer to [137]. In our experimentation, for every training set we train a SVM model using the selected features as predictors. The features can either be single variables or group variables represented in the collapsed form.

A commonly used classifier performance measure is the Area Under the ROC Curve (AUC) [138]. A receiver operating characteristic (ROC) is a graphical plot that describes the performance of a binary classifier system as its discrimination threshold is varied. On the X-axis of the plot is the False Positive Rate (1-*specificity*, or fraction of false positive out of total actual positives*),* and the Y-axis is the True Positive Rate (*sensitivit*y, fraction of true positive out of total actual positives). Each point on the ROC curve is a set of the TPR and FPR measured at a specific value of the set of parameters. Intuitively, we would like to achieve a high TPR and a low FPR. In other words, we would like for many of the points on the ROC curve to lie close to the upper left quadrant of the plot, where the TPR is high and FPR is low. To compare a ROC curve to another, we can calculate the area under the ROC curve. The larger the area, the close a ROC curve lies toward the upper left quadrant, and the better the performance. A random guess would result in a diagonal line running from 0% FPR and TPR to 100% of both, corresponding to an AUC value of 0.5.

Another measure of performance is simply the classification accuracy rate defined as the sum of number of true positives and true negatives over total number instances. There exist some debates as to which of the two measures is more appropriate under different circumstances [139]. Since the types of data we target vary in size and class label proportions (balance), we calculate both measures for our experimentation.

3.4.1.2 Continuous data, censored survival target variable: Cox regression and CI

In clinical settings, we are frequently interested in relating biomarkers to survival time of patients. In this type of problem, the target variable T is the survival time (time to event, the event could be death or relapse, for example) of a particular patient. A characteristic of this data is that at the time of data collection, some patients may still be event-free (for example living or relapse free), and the record would only contain the follow-up time but not the exact time the event would actually occur. This type is data is termed *right-censored survival* data and typical regression methods cannot be applied to model the time-to-event T. A regression method specifically designed to handle survival data is Cox regression. Cox regression (or Cox Proportional Hazards Model) [140] relates the time that passes before an event occurs to predictor variables. In the proportional hazard model, the effect of one unit increase in a predictor variable is proportional to the hazard rate. For example, the increase of expression of a biomarker by one unit may double the hazard rate (of death) of the patient. We can define survival function, or the probability that someone with predictors X_i survive past time t as

$$P(t_i > t) = S_i(t) = e^{-\int_0^t H_i(u)du} \tag{3.10}$$

where $H_i(t)$ is the hazard function which expresses the event rate for subject i at time t and is defined as

$$H_i(t) = H_0(t) \cdot e^{x_i \cdot b} \tag{3.11}$$

b is the vector of coefficients we try to learn, and H_0 is the baseline hazard function that is shared by all the subjects. The beauty of this model is that we do not actually need to learn H_0 to be able to compute the proportional hazard between individuals, as the baseline function cancel out in the computation. The model can be learned by maximizing the likelihood function defined in Section 3.3.2.

Compounded by censorship, measuring the performance of a survival regression model is also a more difficult task, as the error can be computed exactly only for the uncensored cases. Several performance measures have been proposed to deal with the skewed distribution of survival times [141]–[143]. We select the Concordance Index (CI) [141] to measure the performance of the selected features, as it is one of the most commonly used measures for survival models. Intuitively, the CI measures the fraction of all pairs of patients whose predicted survival time are correctly ordered by the regression model. There are scenarios where order of observed survival cannot be determined due to censorship. This can be easily represented by an order graph in Figure 3.5(Left). These scenarios are excluded from the calculation. This probability of concordance between the predicted and the observed survival can be written as:

$$CI = \frac{1}{|\varepsilon|}\sum_{\varepsilon_{ij}} I\left(f(x_i) < f(x_j)\right) \tag{3.12}$$

where $|\varepsilon|$ denotes the number of edges in the order graph, $I(\cdot)$ is the indicator function, and $f(x_i)$ is the predicted survival time for subject i by the model f. Without the censored data, CI is a generalization of the Wilcoxon-Mann-Whitney statistic and can be shown to be equivalent to the AUC [144], [145], and we would expect a CI value of 0.5 for a model that orders the patient survival by random.

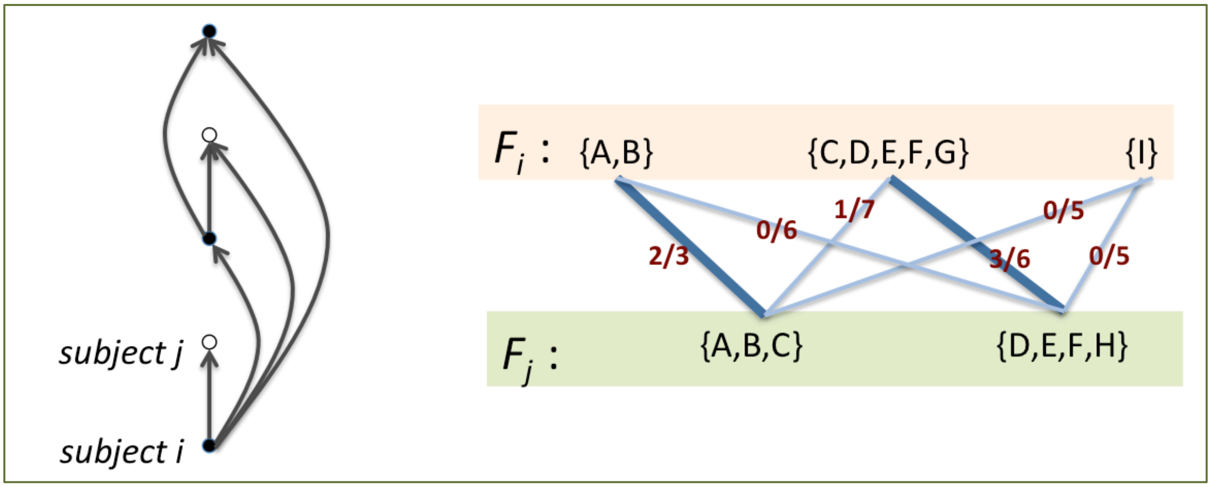


Figure 3.5 An order graph for Concordance Index calculation and bipartite maximum weight matching for cluster stability calculation.
(Left) Two subjects' survival times can be ordered if 1) both of them are uncensored or if 2) the uncensored time of one is smaller than the censored survival time of the other. We represent this by means of an order graph. The set of vertices represents all the individuals ordered by increasing value of their survival times with lowest being at the bottom. Each filled vertex indicates an uncensored survival time, and an empty circle denotes a censored observation. Existence of an edge $\boldsymbol{\varepsilon_{ij}}$ implies that observed survival time $\boldsymbol{T_i}$ for individual $\boldsymbol{i}$ is smaller than that for individual $\boldsymbol{T_j}$. An edge cannot originate from a censored point. (Right) $\boldsymbol{F_i}$ and $\boldsymbol{F_j}$ are two selected group or single variable sets from cross validation runs $\boldsymbol{i}$ and $\boldsymbol{j}$. The elements in brackets are members of a given group variable. We can perform a maximum weight matching to identify a mapping between them, with darker edges indicating the mapping selected.

3.4.2 Stability

For feature selection methods, we are interested in its stability, or how consistently the method selects the same variables across different cross validation runs. Typically, stability measure such as the Tanimoto set-similarity [146] is used to characterize the agreement or percentage of overlap between two sets of variables output from the different runs. Tanimoto set-similarity for two sets C_1 and C_2 is defined as the intersection over union:

$$S(C_1, C_2) = \frac{|C_1 \cap C_{12}|}{|C_1 \cup C_2|} \tag{3.13}$$

where a value of 0 indicates an absence of overlap, and a value of 1 suggests the two sets are identical. In our case, however, the set of variables output from each cross validation run could contain both single- and group-variables, and the Tanimoto set-similarity alone would not suffice. How can we determine the amount of overlap between two such variable sets F_i and F_j from cross validation runs i and j? We need to first define a mapping between elements in F_i and elements in F_j, where these elements can have size greater or equal to one. To do so, we use *maximum weight matching* [147] to build a bipartite graph where one side of the graph contains the members of F_i, and the other contains the members of F_j. We attempt to match each member in F_i to another member in F_j that is maximally similar to it. This is done as follows: for each member s in F_i and w in F_j, we create an edge between them with weight defined by the Tanimoto set difference $S(s, w)$. We find the best bipartite matching, and take the normalized sum of weights of the selected edges of this matching. This gives us a mapping between the members of F_i and F_j. Once we have the mapping, we can compute the stability between F_i and F_j using the weights, taking the average of sum over all pairs. This definition gives us an opportunity to compare the stability of our method across cross validation runs. The final stability is calculated as the average of all pairwise combinations of run i and j, which reflects an estimation of the average agreement over the runs. An example is shown in Figure 3.5(Right).

3.4.3 Cluster Size

Naturally, the size of the clusters grows as the algorithm ascends the tree. We are interested in monitoring how fast the cluster size increases, and how that relates to stability and accuracy. We calculate both the average and the maximum cluster size over cross validation runs.

3.4.4 Choosing suitable parameters: cross validation

A classical framework for parameter selection is cross-validation. The data is divided into N-folds, each with a set of training set and a test set. For each training set, group variable selection is performed and a predictive model learned using the selected group variables. Combinations of different parameters are used for each training set. Stability is calculated across all pairwise combinations, and accuracy is evaluated on the corresponding test set. The set of parameters that gives rise to optimal performance is used for learning a final model over the entire set of data.

3.5 RESULTS

As a proof of concept, we applied T-ReCS to a set of simulated data, a set of six benchmarking gene expression data sets, and a set of large-scale clinical data that includes mRNA and miRNA expression. Both categorical and survival target variables are covered in these datasets. In all cases, we compare the group variable selection performance to a baseline produced by single variable selection. Since no group variable selection method is directly comparable to our method, we will only compare our method against simple ensemble constructed from features selected from different folds of cross validation data. Wherever possible, we perform a 10-fold cross validation as long as the sample size allows. For datasets with sample size less than 200, we perform two repetitions of 5-fold cross validation. For a fair comparison, the single variable MMPC component was run with the same significance threshold a and size of maximum conditioning set k.

3.5.1 Simulated data and results

In order to evaluate T-ReCS, we created a set of simulated data where the structure of the network is known *a priori*. We generate a linear Gaussian Bayesian network with a target variable T, a set of 25 variables that are ancestors of T, a set of 25 variables that are descendants of T, and 44 variables that do not have a path to T. Among the ancestors and descendants of T, three are parents and three are children of T, and the reset has an average out-degree of 2. The parent and children set, PC, has direct edges connecting to T and is the set of variables that we wish to recover through the algorithm. This network structure is shown in Figure 3.6. In this particular set up, we expect to have variables that have small or random correlation with T (unconnected varaibles), variables with varying degree of correlation with T (ancestors and descendants), as well the PC set that together will hopefully be maximally predictive of T. Each node has continuous values analogous to that of gene expression data. The target variable we observe is binary; this is akin to observing cancer subtypes in patients.

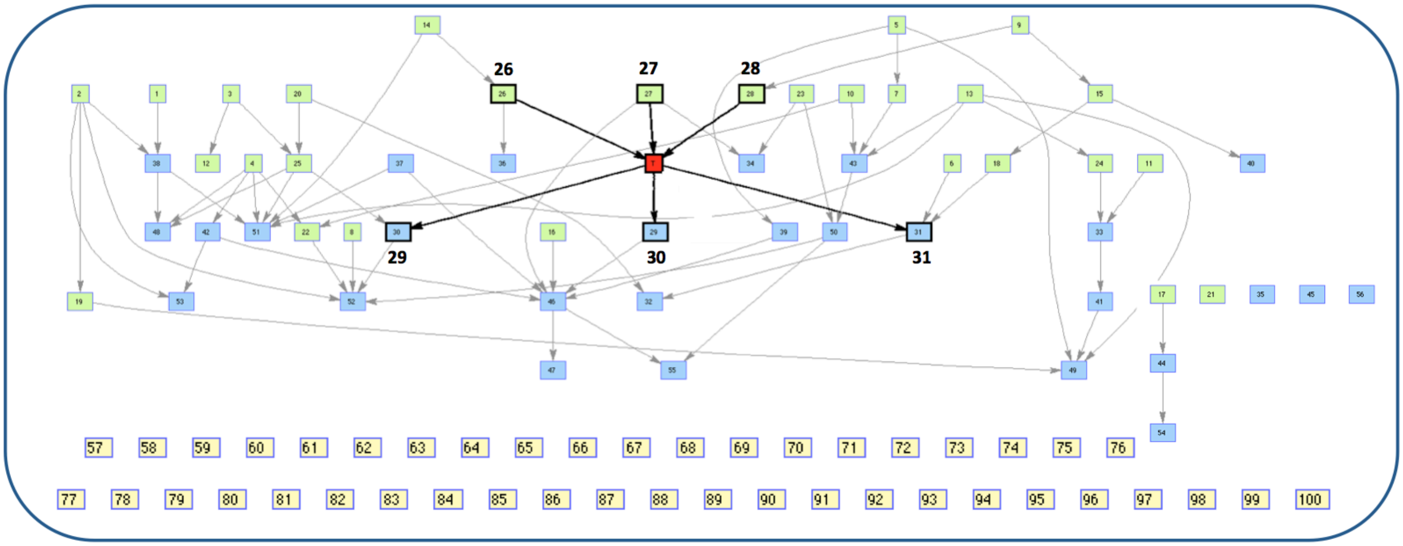


Figure 3.6 A simulated linear Gaussian Bayesian network.
The target variable T is colored in red. Three children (nodes 29,30,31) and three parents (nodes 26,27,28) are planted, in addition to 50 other ancestors (green) and descendants (blue). A set of 44 variables are not connected to T (yellow).

To generate the data, we model the value of each variable as a linear function of its parents with equal weights, with a Gaussian noise of $N(0,1)$. In order to simulate the effects of collinearly between variables that one often observes in biological data, for every variable in the dataset we create nine additional copies of itself with increasing amount of Gaussian noise, ranging from $N(0,0.05)$ to $N(0,2.5)$. We point out a particular subtlety of the design – since we do not have a convenient way of generating the values of the binary variable T from its continuous parents, or generating the continuous values of T's children from the mixed discrete and continuous values of T and its other parents, we resort to modeling the target T as a latent variable with continuous values, with an observed counter part T_{binary} that we generate by thresholding T. A potential caveat of this design is that T's parents and children may still be d-connected when conditioning on T_{binary}, instead of T. However, this is only a problem for constructing a global Bayesian Network or learning the local causal structure around T's parents and children, and would not cause errors in the case of learning the PC of T.

Using the procedure described, we generate 10 training sets with 1000 samples each, and 1 test set with 5000 samples. Using the p-value distribution of the two conditional independence tests we gathered from a preliminary run, we created 10 different significance thresholds a for each of the two tests (Appendix B). For each group variable representation, and for each combination of significance thresholds, we run T-ReCS to select a set of group variables and train an SVM model from the group variables. We then test the trained model on the corresponding test set. We calculated the average pairwise stability between all training set pairs, and the average accuracy as well as AUC. Additionally, we monitor the growth of cluster size across the parameter combinations.

The result of the simulated experiment can be found in Figure 3.7. We first check to see whether T-ReCS can recover the variables in the PC set (nodes 26, 27, 28, 29, 30, 31) . Indeed, we recovered subsets of the PC set in all folds. Half of the cross validation runs contain false positives. However, the false positives are almost always the least significant selected variables. This confirms that the single variable MMPC is indeed performing as expected and is successfully recovering the planted variables. Next, we check that ReKS is correctly clustering the noisy copies of the variables together. Again, we confirm that most of the noisy copies of the variables do appear in the same clusters selected, and when a cluster contains more than one "group variable", they are often connected by an edge, indicating that there is high correlation between them and the clustering is justified. In fact, of all the unique clusters that selected are under the most lenient parameter combination, 75.7% are homogenous in that they only contain copies of a single "seed" variable, while 18.6% contain "foreign" variables seeded from a different variable, and a mere 5.7% of the selected clusters have copies of variables from more than one foreign seed variables. This result confirms that ReKS is indeed creating valid partitions on which our method will build on.

	centroid: selected group variables								
cv1	29(3)	31(1)	30(10)	26(6)36(2)	27(10)				
cv2	29(4)	31(5)	30(10)	27(5)34(1)	36(4)	29(1)	77(1)		
cv3	29(5)	31(5)	30(10)9(6)28(10)	26(10)14(10)36(10)	27(9)	28(6)			
cv4	29(5)	31(1)	30(3)	26(10)14(10)36(10)65(10)	28(5)	27(1)	48(4)		
cv5	29(5)	31(1)	30(4)	26(8)36(3)	27(6)	13(1)	31(1)	70(1)	78(1)
cv6	29(6)	30(8)	31(1)	26(7)36(1)	27(10)34(1)	28(7)			
cv7	29(4)	31(1)	30(1)	26(6)36(2)	27(4)	28(5)	29(1)		
cv8	29(5)	31(4)	30(10)	26(5)	28(10)	82(1)			
cv9	29(5)	31(5)	26(1)	30(10)	27(10)	73(1)			
cv10	29(6)	31(1)	30(8)	26(9)36(2)91(1)	27(6)	29(1)	28(6)		
	medoid: selected group variables								
cv1	29(9)...(42)	31(1)	30(10)6(10)32(4)	26(6)36(2)	27(10)				
cv2	29(5)	31(5)	30(10)	27(5)34(1)	36(5)	29(1)	77(1)		
cv3	29(5)	31(6)29(5)	30(10)	26(1)	27(9)	28(10)...(63)			
cv4	29(5)	31(2)32(2)	30(5)	26(10)14(10)36(10)65(10)	28(5)	27(1)34(1)	48(5)25(6)38(2)51(5)		
cv5	29(5)30(4)	31(1)	30(4)	26(8)36(3)	27(10)23(10)	13(1)	31(1)	70(1)	78(1)
cv6	29(7)94(1)	30(10)95(8)	31(1)	26(10)14(10)36(10)	27(10)34(1)	28(7)			
cv7	29(10)...(70)	31(10)6(10)32(10)	30(1)	26(10)...(28)	27(10)91(1)	28(10)9(10)	29(1)		
cv8	29(5)	31(4)	30(10)16(2)55(10)	26(5)	28(10)	82(1)			
cv9	29(10)...(123)	31(5)	26(1)	30(10)	27(10)	73(1)			
cv10	29(10)...(100)	31(1)	30(10)29(2)	26(10)...(22)	27(6)	29(1)	28(1)		
	PCA: selected group variables								
cv1	29(3)	31(1)	30(10)6(10)32(4)	26(6)36(2)	27(10)				
cv2	29(4)	31(5)	30(10)	27(5)	36(4)	29(1)	77(1)		
cv3	29(5)	31(5)	30(10)9(6)28(10)	26(10)14(10)36(10)	27(7)	28(10)9(6)			
cv4	29(5)	31(1)	30(3)	26(10)14(10)36(10)65(10)	28(5)	27(1)	48(4)		
cv5	29(5)	31(1)	30(4)	26(8)36(3)	27(6)	13(1)	31(1)	70(1)	78(1)
cv6	29(6)	30(8)	31(1)	26(7)36(1)	27(10)...(452)	28(7)			
cv7	29(4)	31(1)	30(1)	26(6)36(2)	27(4)	28(5)	29(1)		
cv8	29(5)9(8)70(4)	31(4)	30(10)	26(5)	28(10)	82(1)			
cv9	29(5)	31(5)	26(1)	30(10)	27(8)	73(1)			
cv10	29(6)	31(1)	30(8)	26(9)36(2)91(1)	27(5)	29(1)	28(6)64(1)70(9)74(1)		

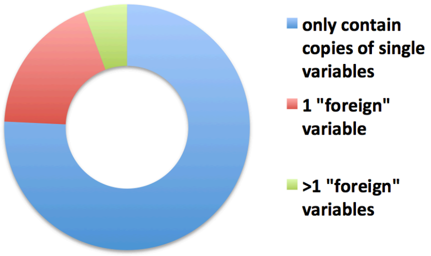


Figure 3.7 Variables selected by T-ReKS over cross validation.
Group variables selected using three different cluster representation methods are listed. Each row contains group variables selected from a given fold. Each cell in the row contains the members of that group variable. The group variables (cells) are arranged in the order of significance (left is most significant). The number in the cell indicates the identity of the seed variable (see node IDs from Figure 3.6). The number in the bracket immediately after the number indicates how many noisy copies of this variable is also present in the cluster. Cells with … contain members too numerous to list. Variables marked in red are outside of the Parent and Children set and are considered false positives. This table is produced at the most relaxed set of thresholds tested, p(x',T|S) < 8*10^{-4} AND p(x,T|x')>10^{-4} . (Bottom) Of all the clusters below this threshold, the majority contains only copies of a single seed variable, while a small portion (5.7%) contains copies of more than two seed variables, indicating that ReKS is indeed producing a valid partition of the variable space.

Having confirmed the efficiency of MMPC and ReKS in identifying the correct single variables around T and partitioning the data in a meaningful way, we next examine the performance of the statistical tests that the group variable selection component of the algorithm relies on. Recall that we employ two tests of independence to aid us in making the decision of whether or not to replace a single variable by its parent, group variable. What should we expect to see, if the two tests are indeed behaving the way we expect them to? For the test for $Dep(X'; T \,\middle|\, S), S \subseteq PC \setminus \{X\}$, we expect the p-value to increase as we ascend the tree, as the benefit of including X' in the PC declines, since the signal contained in X' will decrease with increase in its size. On the other hand, for the test for $Ind(X; T|X')$, we anticipate the p-value to decrease as we ascend the tree and for X and T to become associated given X', since X' would carry decreasing amount of signal and would no longer predict T equally well as X. Here, we check to see if these two trends indeed exist in our simulated dataset.

In Figure 3.8, we plot a ReKS tree example generated from a run of cross validation. We use the centroid method to collapse the group variables in this example, and highlighted in green is the trace of the variable as it gets collapsed and advances toward the top of the tree. We plot the three statistics of interest: log $(P(X';T|S))$, log $(P(X;T|X'))$, and the size of the cluster. We verify that we indeed observe the pattern we were expecting as mentioned. At the bottom of the tree, we have only 3 members in the cluster, and $P(X';T|S)$ gives a value of 0, indicating that X' and T are strongly dependent, despite our best effort to condition out the dependency. In the mean while, $P(X;T|X')$ yields a value of 0.9827, displaying a high level of conditional independence between X and T given X'. These observations suggest that X can be replaced by

X' without loss of classification accuracy, thus, we advance one level up the tree. As we ascend the tree, the first p-value increases, and the second p-value decreases. The cluster size also increases from 3 members to 163 at the second level from the top. If we were to place a set of significance thresholds 10^{-4} and 10^{-3} on this path as stopping criteria, for example, we would stop at the third level from the bottom, yielding a group variable consisting of 7 members.

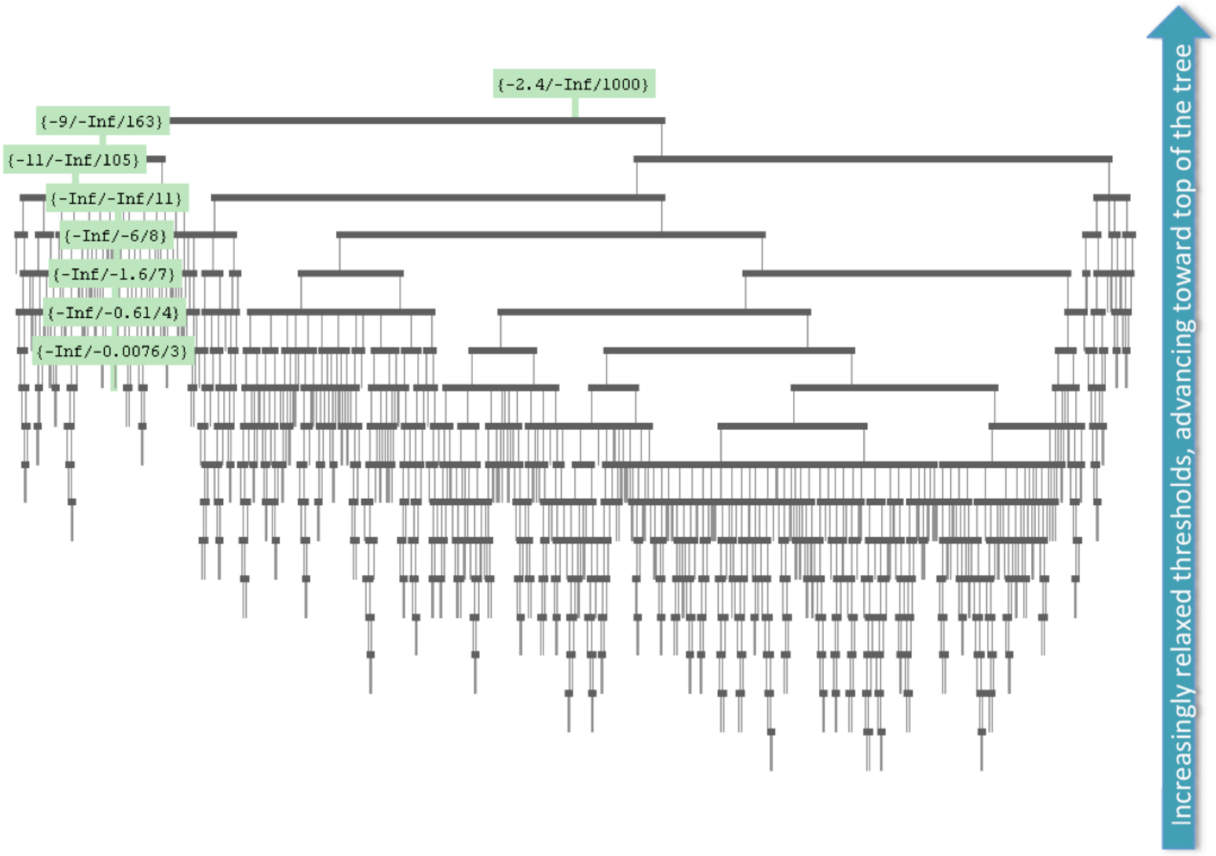


Figure 3.8 An example of the ReKS tree on simulated data.
We follow the trace of the most significant variable, marked in green, advancing from the bottom of the tree all the way to the root. The numbers in the bracket contain three statistics of interest: $\log(P(X';T \mid S))$, $\log(P(X;T \mid X'))$, and the size of the cluster. Consistent with our expectations, the p-value from the first test increases, while the p-value from the second test decreases as cluster size increases. The more relaxed the significance thresholds we use, the higher up the tree we allow the cluster to grow.

Finally, we investigate the effect of the significance thresholds on algorithm performance. In Figure 3.9, we plot the average cross validation accuracy, stability, and cluster size against different threshold combinations, first on a YYY plot(top) and then on heatmaps showing the pattern across the parameter landscape(bottom). We generally see a trend of increasing stability and cluster size toward the top of the tree, with the accuracy displaying more subtle variations with a slight spike in the middle region. We plot the baseline stability and accuracy in dotted lines in corresponding colors. These are the average performance of single variable MMPC across the cross validation runs, and indeed stability is improved dramatically

(95% confidence interval 0.331-0.551), while accuracy enjoys a very subtle boost (95% confidence interval 0.0121-0.0289). We also investigate the performance of T-ReCS against another baseline – simple ensemble of the single variables selected from the 10 cross validation runs. We use the union set of these variables as predictors and train SVM models across the 10 training sets. The average test accuracy is plotted in purple dotted line, and we can see that T-ReCS's performance does not depart significantly from the ensemble method (p=0.75) in terms of accuracy. All of these methods outperform the baseline model when no variable selection is performed (Accuracy = 0.8786±0.014, brown dotted line), and also the baseline model where a random set of variables of the same size as the selected variables are used for training a model for cross validation (Accuracy = 0.6283±0.119). Lastly, we observe that centroid and PCA seems to produce very similar results, while medoid allows for larger clusters to be included, possibly because the same member continues to be the "medoid" of the cluster as it advances up the tree, masking the "noise" that other members of the cluster may otherwise introduce.

Having concluded that T-ReCS performs as expected on the simulated data, we now turn our attention to its application on real datasets.

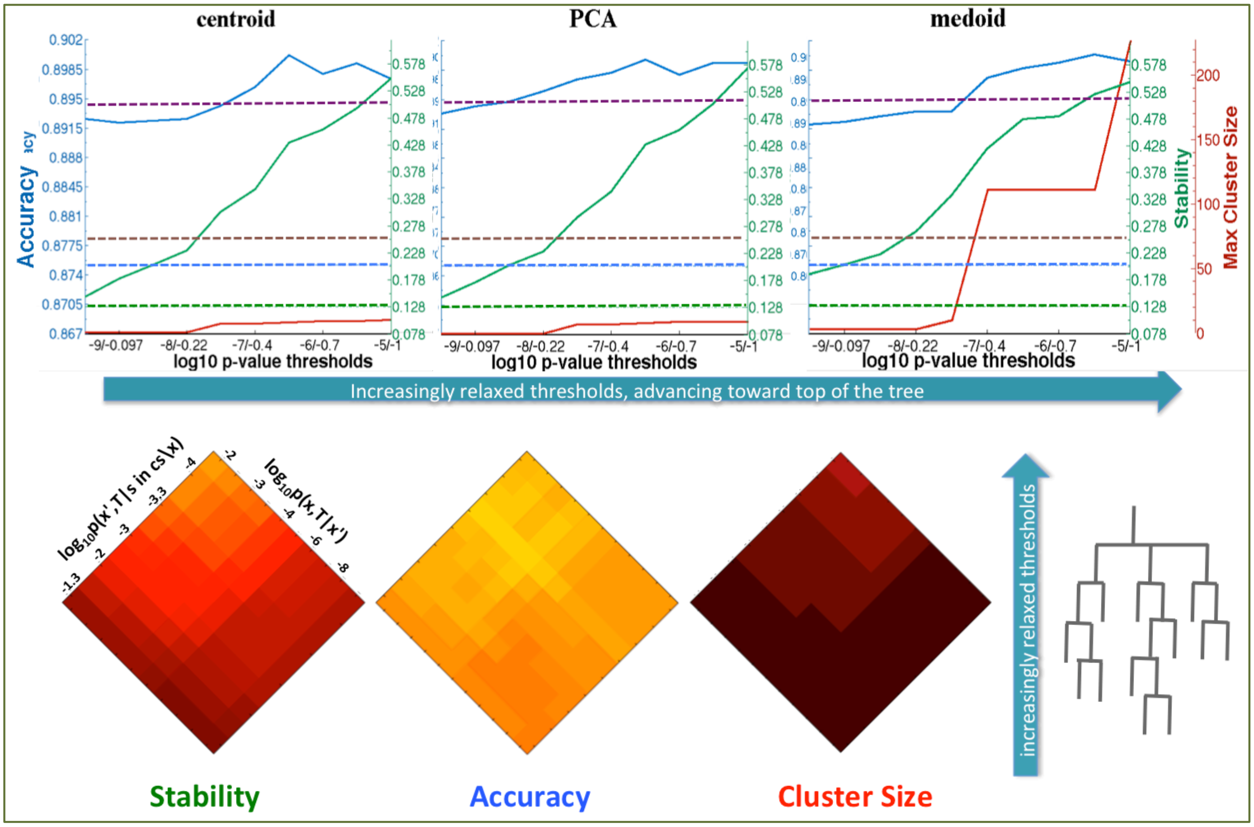


Figure 3.9 T-ReCS performance on simulated data.
(Top) Accuracy(blue), Stability(green) and Cluster size(red) across 10 parameter combinations (left to right, most stringent to most permissive) for three different cluster representation methods applied on simulated data. Plotted in dotted lines in corresponding colors are single variable selection baseline results. The purpose dotted line is the ensemble baseline accuracy, and the brown dotted line the no variable selection baseline. (Bottom) The same results plotted across all 10 by 10 threshold combinations. The bottom combinations correspond to restricting the group variables to the bottom portion of the tree (shown to the right)

Since the simulated data is for binary target variable, so far we have yet to test the performance of the conditional independence tests as well as the selected variables on survival data. In 2010, Tsamardinos *et al*. [133] introduced the *Survival MMPC* (SMMPC) and tested extensively its performance against other variable selection methods, and in conjunction with various regression techniques, on six clinical data sets [9], [15], [148]–[151]. To see if our method could improve on this baseline, we run T-ReCS on the same datasets for comparison. The six sets of censored survival data range in size from 86 to 295 cases with 70 to 8810 variables, and the events of interests are either metastasis or survival (Table 3.1). The YYY plots are shown in Appendix C. In general, stability improves from the baseline by a large margin for several datasets. Accuracy hovers around baseline, with small increase or decrease across parameter combinations, none statistically significant. The size of the group variables chosen largely stays within the range of 10 members. We conclude from this set of experiments that our method enjoys gain in stability without severe loss of accuracy, compared to the single variable selection baseline.

Table 3.1 A list of benchmarking datasets used in this evaluation (*significant at 0.05)

Dataset	#Cases	#Cens	#Vars	Event	%Stability	%Accuracy
Vijver	295	207	70	metastasis	+19~27.5%*	-1.2~1.7%
Veer	78	44	4751	metastasis	+0~9%	-1.8~10.6%
Ros02	240	102	7399	survival	+23.5~40.8%*	-2.7~0.64%
Ros03	92	28	8810	survival	+100~194%*	-5.3~1.2%
Bullinger	116	49	6283	survival	+6.7~19%	-1.3~3%
Beer	86	62	7129	survival	+39~80%*	-7.3~8.5%

3.5.2 Application: Melanoma gene and miRNA expression

In this section, we showcase an application of T-ReCS on a large-scale clinical dataset for which the algorithm is designed. mRNA and miRNA gene expression were collected from a cohort of patients (ECOG-E2603 [152]) with unresectable locally advanced or stage IV Melanoma. A total of 105 samples were available with censored progression free survival. We applied T-ReCS on the mRNA and miRNA expression data separately, against progression free survival follow-up time of the patients. A list of the selected genes for both data can be found in Appendix D. We highlight a few instances where we were able to discover biologically meaningful results using the group selection strategy we proposed.

Among the single mRNA genes selected, Plxnb1 has the most significant p-value, and has been shown to act as a tumor suppressor of Melanoma [153]. The group variable additionally recruited Rad23, a protein involved in the nucleotide excision repair (NER) mechanism. This protein was found to be a component of the protein complex that specifically complements the NER defect of Xeroderma Pigmentosum group C (XP-c) cell extracts *in vitro*. Patients with Xeroderma Pigmentosum have mutations in NER pathway, and have a 1000-fold increase in the incidence of skin cancers including melanoma, suggesting a significant role NER plays in melanoma genesis and a potential role of Rad23 [154]. Another example is BCMO1, a protein that is a key enzyme in beta-carotene metabolism to vitamin A important for skin protection. BCMO1 is part of a very significant group variable whose seed variable is LOC389936, an shRNA construct with no known association with melanoma.

Results from group variable selection on microRNA data are equally promising. With standard MMPC, only a single miRNA was selected: hsa-miR659-3p. This miRNA is significant in that it targets RAS, one of the most important common oncogenes in human cancer [155], and BRAF, an oncogene that regulates the MPA kinase pathway and affects cell division, differentiation, and secretion, and whose mutations is undergoing active investigation in the melanoma community [156]. After running our algorithm, this single selected miRNA was grown into a group variable containing three members. The additional members are hsa-miR219-1-3p and hsa-miR516a-5p. hsa-miR219-1-3p is of special interest to us, as it targets several genes of interest, including TGFBR2, a well known signal transduction protein whose mutations have been associated with tumor progression in solid tumors [157], and FGFR1. A number of functional studies have implicated FGFR1 signaling in melanoma progression [158]. Introduction of antisense oligonucleotides targeted toward FGFR1 into metastatic cell lines resulted in decreased proliferation and signs of differentiation [159], [160]. Injection of an antisense FGFR1 construct into primary and metastatic melanomas grown in nude mice has also been shown to result in inhibition of tumor growth and induction of apoptosis [161], [162]. We examined the Cox regression coefficients of this selected cluster to interrogate our results. Indeed, all of these miRNAs are positively associated with survival; higher miRNA expression levels are associated with lower expression of their targets, which in this case are all oncogenes. This is consistent with our expectation that lower expression of the oncogenes lead to longer survivals.

We note that hsa-miR219-1-3p was never selected in the single variable portion of the 10 cross validation runs. This suggests that a simple approach such as an ensemble method that simply take the union of single variable selection results not have been able to uncover this miRNA.

3.6 DISCUSSION AND FUTURE DIRECTIONS

In this chapter, we introduced a novel feature selection algorithm for learning predictive *group* features of a target variable, T. This algorithm builds on top of an existing local causal discovery algorithm. We developed two conditional independence tests to identify groups of predictive features that are statistically equivalent to single predictive features they contain. A definition for group variable stability is proposed to characterize relative stability of this group variable selection scheme. We provide implementations of this algorithm for both categorical and censored, survival outcome T. To our knowledge, this is one of the very few that achieves this goal without requiring the user to supply an explicit partition of the data *a priori*, such as the strategy employed by group lasso. Instead, we only require a clustering structure in the form of a tree, and the size of the clusters selected will be automatically determined. The algorithm is sound and can be run efficiently on datasets in the range of tens of thousands of variables.

Additionally, it is computationally efficient without imposing strict requirement for training size, which makes it suitable for high-throughput biological data.

We demonstrated the stability improvement of the algorithm over single variable selection and ensemble baseline on simulated data. Significant stability improvement was achieved while minimum change in accuracy occurred. We also investigated its performance over a range of parameter combinations using three distinctive cluster representation methods. Similar performance was observed between centroid and PCA, while medoid tends to produce slightly more dissimilar behaviors. We suspect that this is because a medoid does not represent an "average" behavior of a cluster; it is merely a member of the cluster that is most similar to everyone else. As the cluster size increases, the identity of this member could remain unchanged, in which case the cluster may be allowed to grow very large without affecting the predictive performance, and too many noisy members could be erroneously recruited. On the other hand, medoid could also be susceptible to fluctuations of the member composition in the scenario that a current cluster joins with a larger, dissimilar cluster and the identity of medoid switches all of a sudden. For this reason, we recommend centroid and PCA as the preferred collapsing methods since they produce more gradual change in stability across many parameter ranges. We observed gain in stability consistent with our expectation as we relax the parameters, with a minimal loss in accuracy. The accuracy reflects a tradeoff between overfitting (from the more stringent range of the parameters) and loss of predictive signals (in the more relaxed range of the parameters), with fluctuations in between. We note that we do often observe a slight peak in the middle, suggesting that this may be a parameter region that is more suitable for the two conditional independence tests. For most of the datasets we investigated, this region usually corresponds to 10^{-2} to 10^{-4} for both thresholds. A closer look in the distribution of p-values (Appendix B) that result from these two tests also confirm that this parameter range is most effective in thresholding the clusters in the bottom portion of the tree. Additional tests on large number of clinical datasets for both categorical data survival data are required to provide recommendations for the most appropriate parameter range for the two tests. Note that we would not expect the parameters to be necessarily the same for categorical versus survival data. Alternatively, cross validation can be performed on all input datasets and a set of parameters can be selected manually by the users, or output automatically based on a statistic that combines the accuracy and stability based on a predefined weight.

We demonstrated the potential benefits of discovering biologically meaningful biomarkers on a set of real clinical data. Results are delivered in the form of clusters of maximally predictive genes (includes TFs and miRNAs) for a given disease. This provides a small list of candidate genes to be tested on the bench. As a by-product of the algorithm, given a new patient sample, we may be able to predict disease classification or clinical outcome using the molecular signatures identified.

We have also begun systematically applying our method on a number of large-scale studies, including several in the TCGA initiative as well as that of the Metabric [21] and LGRC [20] consortiums. We expect to apply our method on more datasets as they become available, and collaborate with domain experts to identify potential areas of improvements. Future directions include incorporation of more sophisticated latent variable representation such as Factory Analysis and Canonical Correlation analysis; incorporating prior in the ReKS step using the prior incorporation scheme we proposed, and investigating its effect on the p-values and predictive performances as we vary the amount of prior network information incorporated.

While in this chapter we tested our method in high-throughput gene expression datasets only, it can be easily adapted to other high-dimensional systems such as methylation and SNP data and beyond to provide predictive models as well as biological intuition. Additionally, the modular structure of the algorithms paves the way for a novel group feature selection framework in which alternative clustering step, hypothesis tests, and different variants of the causal discovery algorithm can be employed. The results presented are promising both in computational performances as well as biological implications.

4.0 MIRCONNX: CONDITION-SPECIFIC MRNA-MIRNA NETWORK INTEGRATOR

In order to enrich computational results with biological meanings, we developed mirConnX, a user-friendly web interface for inferring, displaying and parsing mRNA and microRNA (miRNA) gene regulatory networks. mirConnX combines sequence information with gene expression data analysis to create a disease-specific, genome-wide regulatory network. A prior, static network has been constructed for all human and mouse genes. It consists of computationally predicted transcription factor (TF) – gene associations and miRNA target predictions. The prior network is supplemented with known interactions from the literature. Dynamic TF-gene and miRNA-gene associations are inferred from user-provided expression data using an association measure of choice. The static and dynamic networks are then combined using an integration function with user-specified weights. Visualization of the network and subsequent analysis are provided *via* a very responsive graphic user interface. Two organisms are currently supported: *Homo sapiens*, and *Mus musculus*. The intuitive user interface and large database make mirConnX a useful tool for clinical scientists for hypothesis generation and explorations. mirConnX is freely available for academic use at http://www.benoslab.pitt.edu/mirconnx.

We have integrated mirConnX with ReKS and T-ReCS to create a fully integrated, biologically informed biomarker discovery environment. In the upcoming release, mirConnX2.0, users will have the option of uploading clinical labels, in addition to mRNA and/or miRNA expression data, to explore regulatory relationships between computationally prognostic mRNA and miRNA clusters.

4.1 MOTIVATION

Since its discovery two decades ago, it has become increasingly clear that microRNAs (miRNAs) play a crucial role in modulating gene expression at the post-transcriptional level. The small, 22 nucleotide long RNA molecules fine-tune gene expression by base pairing to target messenger RNAs, resulting in its degradation or causing translational repression. As Pandit *et al.* [163] has shown, deregulation of even a single miRNA may cause complex human diseases. Regulatory network reconstruction methods have traditionally involved transcriptional regulation only. Incorporating miRNAs thus becomes the next natural step. Only few tools have explored ways to associate mRNA and miRNA expression to infer regulations. MMIA [164] and MAGIA [165], for example, utilize association metrics such as correlation and mutual information. In a different context, Huang et al. [166] employed a Bayesian model to identify miRNA targets from sequence features and expression data. However, there are several limitations to these tools. MMIA only examines a subset of the miRNAs that are significantly up- or down- regulated, and omits those that could potentially be significantly correlated with their targets if they are not considered to be differentially expressed, based on the specific threshold. This only limits the data to those with a control/disease contrast, excluding possible use of time-series data. GenMir++ [166] is a more sophisticated algorithm, but it becomes computationally inefficient when a large number of genes are considered. Furthermore, it does not take into account other supporting information such as transcriptional regulation. In fact, none of these tools incorporates the full set of transcription factors (TFs) in global network construction. Additionally, network motifs such as feed-back and feed-forward loops that are known to have an important role in cancer development and other diseases are usually not identified as part of the routine analyses of the currently available tools.

To this end, we developed *mirConnX* to attempt to address some of the above concerns. mirConnX (http://www.benoslab.pitt.edu/mirconnx) takes advantage of prior knowledge (from sequence data), and incorporates evidence from gene expression data to create condition-specific genome-wide regulatory networks. mirConnX also aims to identify gene network motifs, involving transcription factors and miRNAs, that are associated with the corresponding diseases, pathogenesis or phenotype of interest.

4.2 METHOD OVERVIEW

4.2.1 Method overview

mirConnX aims to provide an integrated environment that allows the user to infer genome-wide transcriptional (TF-gene/miRNA) and post-transcriptional (miRNA-gene/TF) regulatory networks for a particular disease or condition. We consider mRNA and miRNA expression data measured under the same set of conditions, or at the same time points, or from the same corresponding diseased or normal samples (matching samples). The mRNA and miRNA expression data are pre-processed to remove genes that are lowly expressed with limited variance overall. Then, we connect TFs and miRNAs to genes using a statistical association measure. The *association network* that is constructed reflects the disease status or the condition of interest. This network is an undirected graph, in which an edge exists between two nodes (genes) if an interaction has been detected. Note that such association networks do not discriminate between direct and indirect interactions. This network is then superimposed to a pre-compiled, species-specific *prior network*, which is derived from TF motif scanning and binding, miRNA target predictions, and literature evidence. The prior network is a directed, weighted graph, in which an edge between a TF or miRNA and a gene exists if the former is predicted to regulate the latter. All the connections in the prior network correspond to direct, predicted or verified interactions. Superimposing the two networks *via* an integration function results in a directed network, which is expected to contain significantly fewer indirect interactions (depending on the weight the user assigns on the prior network). mirConnX web tool allows easy visualization and exploration of the network, and identifies network motifs. In the following sections, we describe the construction of the context-dependent (dynamic) association network, the construction of the prior (static) network, and their integration. Figure 4.1 presents an overview of the mirConnX pipeline.

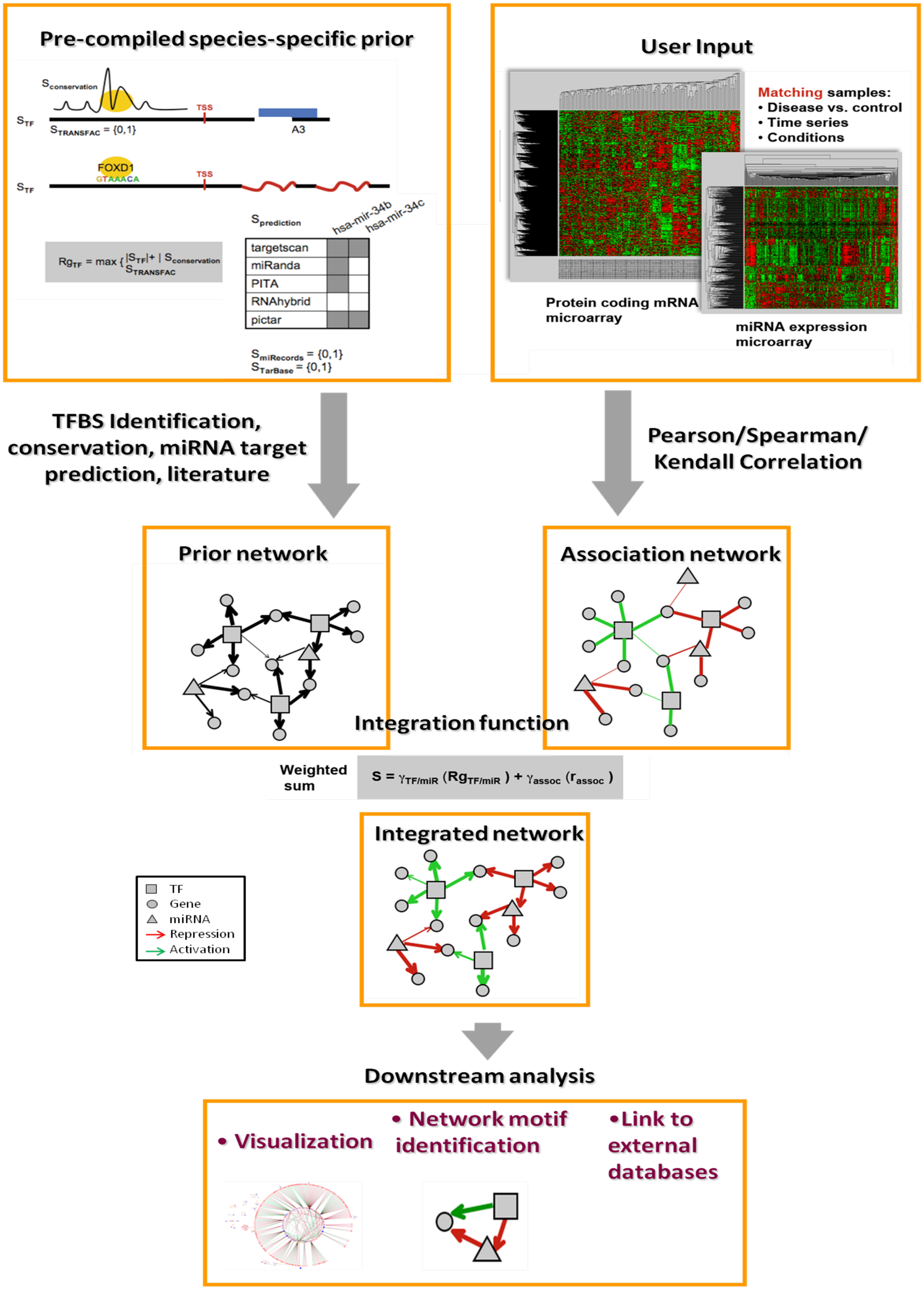


Figure 4.1 Overview of the integrated analysis in mirConnX.

4.2.2 Building a prior network

The prior network is constructed by combining all predictions of TF to gene and TF to miRNA interactions and all miRNA target predictions. The network is then enhanced by literature evidence that confirms the existence of an edge. This results in a directed network that represents the collection of prior knowledge on regulatory potentials between genes.

4.2.2.1 TF to gene/miRNA regulations

We define the binding potential (Rg_{TF}) of a promoter sequence for a given gene/miRNA as the maximum score between literature evidence (S_{Lit}) and binding score of a TF (S_{TF}). The binding score is calculated using a sliding window method [27] on the promoters of genes and miRNAs, The JASPAR [34] and TRANSFAC [35] position weight matrices (PWMs) are used for the scanning. A subsequence is considered a binding site for a TF if its PWM score is on the top 1% of all scores for this PWM. In addition, UCSC Regulation track Conserved TFBS (S_{Cons}) scores are added to enhance the confidence. The sum of S_{TF} and S_{Cons} are normalized to a score between 0 and 1. Finally, if an experimentally verified binding motif for a given TF is available (e.g., in TRANSFAC), we increase the binding potential automatically to 1, as shown below,

$$Rg_{TF} = max\{(|S_{TF}| + |S_{Cons}|)\,, S_{Lit}\}$$

$$S_{Lit} \subseteq \{0,1\} \qquad \textit{(4.1)}$$

Regular gene promoters were defined 5kb upstream of TSS obtained from Database of Transcription Start Sites (DBTSS) [167], The Eukaryotic Promoter Database (EPD) [168], and UCSC genome browser Regulation-Transcription track (Eponine and SwitchGear TSS) [169]. miRNA TSSs are defined using a combination of predictions and experiments from

CoreBoost_HM [169], Marson *et al.* [170], and Corcoran *et al*. [171]. Human (NCBI36/Hg18) and mouse (NCBI37/mm9) sequence data were downloaded from UCSC genome browser [36].

4.2.2.2 miRNA to gene/TF regulations

miRNA target prediction algorithms generally do not agree very well. Thus, we used a combination of five target prediction algorithms that take into account of seed sequence, flanking sequences and context, binding energy, and conservation. These algorithms are: PITA [172], miRANDA [46], TargetScan 5.0 [173], RNAhybrid [174], and Pictar [48]. If predictions for corresponding genome versions are not available, we ran the algorithms using default parameters and cutoffs. We define the regulatory potential (Rg_{miR}) of a miRNA for a gene as the proportion of the target prediction algorithms predicting the gene to contain at least one miRNA target site. In addition, if the 3'UTR of a gene contains an experimentally verified site from TarBase [175] or miRecords [176], the regulatory potential of the gene for a given miRNA is increased to 1, as shown below,

$$Rg_{miR} = max\ \{| |S_{prediction}|, S_{TarBase}, S_{miRecords}\}$$

$$S_{TarBase}, S_{miRecords} \subseteq \{0,1\} \tag{4.2}$$

Human and mouse 3'UTRs were downloaded from UCSC genome browser. The list of mature and complement miRNAs, as well as their sequences, were obtained from miRBase v.14 [46].

4.2.2.3 Gene expression preprocessing

Standard gene symbol and miRNA ID are used as our primary identifier. Genes and miRNAs with multiple probes on the array, or those converted to the same gene symbol/miRNA ID, are collapsed into a single medium value. The normalized mRNA and miRNA expression data are pre-processed using three filters for low (1) absolute expression, (2) variance, and (3) entropy. A cutoff of 5% is used for mRNA and miRNA expression data individually to remove data that are not likely to be important for the network. A list of the genes filtered and excluded from the analysis is available for user to download. Finally, all matching conditions or samples between the mRNA and miRNA data matrices are retained for analysis. In case of multiple replicates for the same condition, the median value between replicates is used.

4.2.2.4 Constructing association network from gene expression data

We construct an association network from the user-supplied expression data by measuring the strength of all pair-wise interactions between TFs, miRNAs and genes across the samples/replicates. A number of parametric and non-parametric association metrics are available to the user for defining these interactions. Correlation coefficient is one of the most intuitive, and most well received. The different flavors of correlation (Pearson, Spearman, and Kendall) have been used successfully in the past and achieved different levels of success, for example in the WGCNA R package [177]. Pearson correlation coefficient is often used when a linear dependence between the variables exists. By contrast, Spearman rho correlation coefficient applies the Pearson formula on the ranks of the values of the two variables and can detect similarities even if non-linear (but monotonic) association exists. Kendall tau rank correlation coefficient also operates on the ranks, but it calculates the probability of concordance or discordance of any pair of observations. In general, Spearman and Kendall give similar results, but they differ on the magnitude (for more details on correlation measures, please see [178]). We implemented these three correlation measures and applied them on pairs of gene, TF, or miRNA expression values across matching conditions. The absolute magnitude reflects the level of correlation, and the sign suggests positive or negative interaction. Mutual Information is the non-parametric counterpart of the correlation coefficient and it has been implemented in algorithms such as ARACNE [179] as the measure of association for genome-wide two-way interactions. Mutual Information does not provide information about the sign of interaction (it is non-negative) and is generally computationally intensive and sample-size sensitive, since it requires estimation of marginal and joint probabilities of the variables. As a result, we have not implemented the mutual information statistic in miRconnX, although we might do so in the future. The degree of association, r_{assoc}, is defined as the probability that two genes are correlated. We used the inverse of correlation coefficient significance $(1-p)$ as the probability of nonrandom association. The use of significance, instead of the coefficient itself, takes into account of the sample size and allows a fair comparison between networks generated by different size of data.

4.2.3 Network integration

We currently implement a simple weighted sum of the regulatory potentials (Rg_{TF} or Rg_{miR}) from the prior network and association score (rassoc) as integrated binding score (S) between any two genes, as shown below,

$$S = \gamma_{prior}(Rg_{TF}, Rg_{miR}) + \gamma_{assoc}(r_{assoc}) \qquad (4.3)$$

where γ_{prior} is a user defined parameter between 0 and 1, and γ_{assoc} = 1 - γ_{prior}. The default for γ_{prior} is 0.3 (*i.e.*, 30%) and a value less than 0.5 is recommended for the prior information. We also allow the user to set a cutoff for the minimum integrated regulation score for an interaction to be displayed in the output. This is also number between 0.0 - 1.0, but the higher the more stringent the criteria (reduced false positive interactions reported). A value of 0.7 - 0.99 is recommended. We do cap the number of interactions to be displayed on screen at 3000, as beyond that the network becomes too large to be efficiently visualized.

4.3 USER INTERFACE

4.3.1 Input format

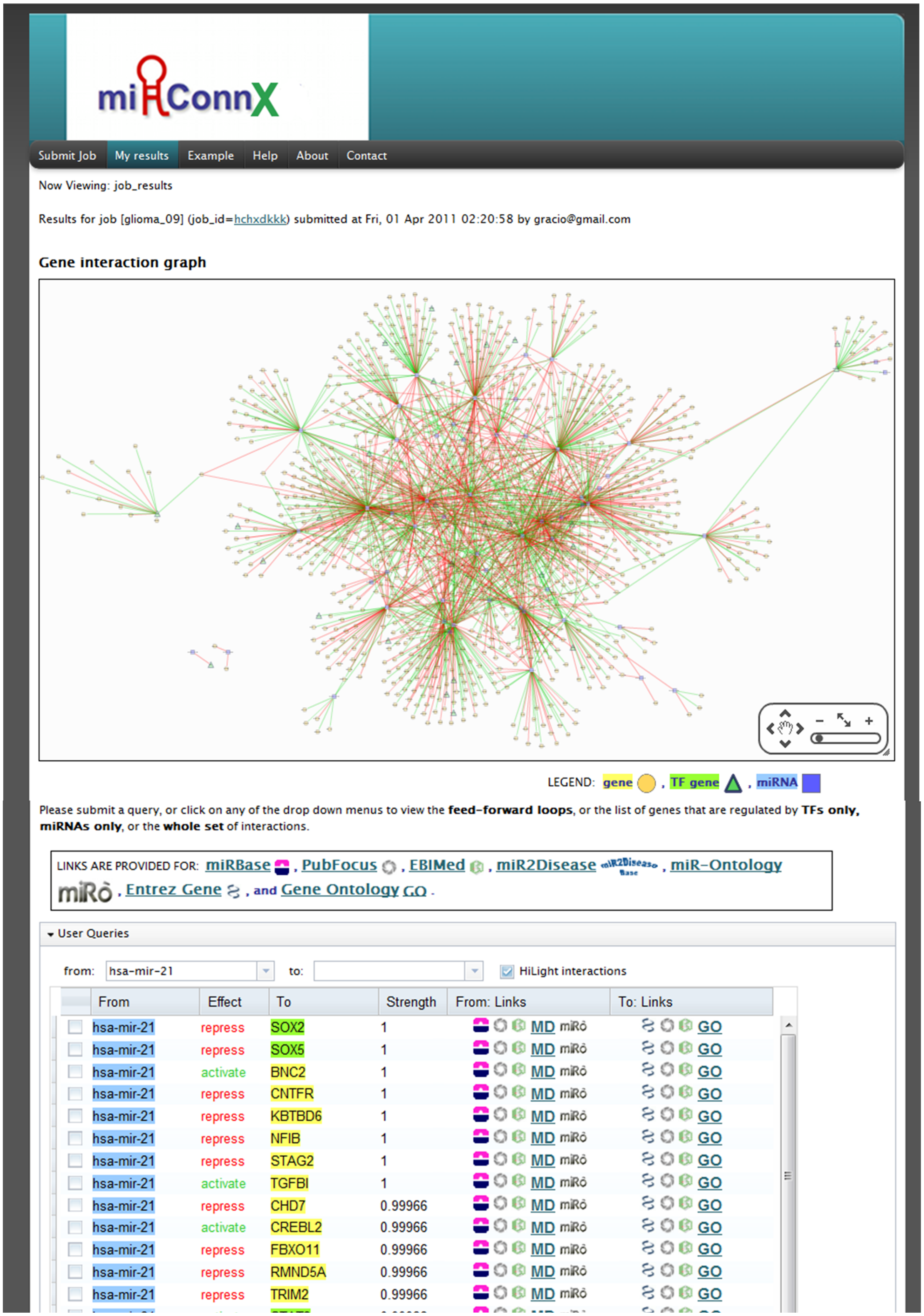
mirConnX accepts normalized mRNA and miRNA expression data in tab-delimited files where the first row contains sample IDs and the first column contains mRNA or miRNA IDs. mirConnX supports gene symbols, Ensembl Gene ID, Ensembl Transcript ID, Entrez Gene ID, RefSeq DNA ID, and Unigene ID as mRNA identifiers; and miRBase miRNA ID and Accession numbers as miRNA identifiers. An example of the matching mRNA-miRNA datasets can be found and pre-loaded on the front page. Note that the sample IDs for mRNA and miRNA data should match. Any unmatched samples are discarded. mirConnX allows multiple columns with the same header in case of biological or technical replicates. The input data sets are stored only during a user's session and are used to construct the association network. If no miRNA data file is included, the resulting network will show only TF-gene interactions. We currently support two organisms: human (*Homo sapiens*), and mouse (*Mus musculus*), as genome annotation and prior information is most abundant for these species.

4.3.1.1 Submission and waiting time

Depending on the size of the files (number of genes analyzed) and types of analysis chosen, the analysis could take anywhere from minutes to up to an hour. As an example, for 20,000 genes and 500 miRNA, the computing time is roughly 15 minutes using Pearson correlation. While the job is running, an execution log will be displayed. The user can close the browser window. When the job finishes, the user will receive an email notification and retrieve the results from the link provided.

4.3.1.2 mirConnX output

Following the link to the result, a visualization of the network is displayed, as shown in Figure 4.2. Cytoscape Web v0.7.2 [180] is used for network display. The rendering time for a network with 1,500 nodes and 2,500 connections is about 15 sec. Once uploaded, browsing the various areas of the network is instantaneous. Users can use the tools at the bottom right corner to zoom in/out and edit node placements on the visualization page, and output the visualization as graphics. The list of interactions is also displayed with links to external databases such as miRBase and Entrez Gene [181] for annotation, PubFocus [182], EBIMed [183], miR2Disease [184], and miRo [185] to facilitate clinical research by sifting through large body of literature and records, as well as Gene Ontology [186] terms for each genes. We also make available for download: **(1)** list of interactions above the user-defined display cutoff, ranked by regulatory score in tab-delimited text file; **(2)** list of nodes, ranked by degree centrality in text-delimited text file; and **(3)** the network in pdf or GRAPHML graph formats compatible with Cytoscape for further exploration. The user can **(4)** search for a particular node and its targets/regulators (through the "List of gene interactions: filtered" drop down menu), a set of particular interaction, and highlight or select the corresponding nodes and edges on the graph display. Finally, we display all **(5)** feed-forward loops and their neighbors at the given threshold. In addition, a summary of statistics, including the actual number of TFs, miRNAs and genes can be retrieved under "execution log".



Now Viewing: job_results

Results for job [glioma_09] (job_id=hchxdkkk) submitted at Fri, 01 Apr 2011 02:20:58 by gracio@gmail.com

Gene interaction graph

LEGEND: gene ●, TF gene ▲, miRNA ■

Please submit a query, or click on any of the drop down menus to view the **feed-forward loops**, or the list of genes that are regulated by **TFs only**, **miRNAs only**, or the **whole set** of interactions.

LINKS ARE PROVIDED FOR: miRBase, PubFocus, EBIMed, miR2Disease, miR-Ontology miRò, Entrez Gene, and Gene Ontology GO.

User Queries

from: hsa-mir-21 to: ☑ HiLight interactions

	From	Effect	To	Strength	From: Links	To: Links
□	hsa-mir-21	repress	SOX2	1	MD miRò	GO
□	hsa-mir-21	repress	SOX5	1	MD miRò	GO
□	hsa-mir-21	activate	BNC2	1	MD miRò	GO
□	hsa-mir-21	repress	CNTFR	1	MD miRò	GO
□	hsa-mir-21	repress	KBTBD6	1	MD miRò	GO
□	hsa-mir-21	repress	NFIB	1	MD miRò	GO
□	hsa-mir-21	repress	STAG2	1	MD miRò	GO
□	hsa-mir-21	activate	TGFBI	1	MD miRò	GO
□	hsa-mir-21	repress	CHD7	0.99966	MD miRò	GO
□	hsa-mir-21	activate	CREBL2	0.99966	MD miRò	GO
□	hsa-mir-21	repress	FBXO11	0.99966	MD miRò	GO
□	hsa-mir-21	repress	RMND5A	0.99966	MD miRò	GO
□	hsa-mir-21	repress	TRIM2	0.99966	MD miRò	GO

Figure 4.2 Snapshot of the glioblastoma case study output.

An example search for the downstream targets of miR-21, a key player in glioblastoma development, is shown in the middle. Feed-forward loops are displayed on a separate tab. Links to external databases are provided for every coding gene or miRNA.

4.4 APPLICATION: TCGA

The main idea behind mirConnX was first used to analyze lung epithelial gene expression data few years ago [163]. In that study we were able to identify a feed-forward-loop that included SMAD TFs, let-7d, and HMGA2 gene, which was central in the regulation of epithelial to mesenchymal transition (EMT). Furthermore, we later found that knocking down of let-7d in the trachea of mice can cause lung fibrosis few days later [163].

Here we present a case study that demonstrates the utility of mirConnX. We downloaded a set of publicly available mRNA and miRNA expression profiles from The Cancer Genome Atlas (TCGA) pilot project (http://cancergenome.nih.gov/), where a large compendium of tumor and normal Glioblastoma mutlforme (GBM, primary brain tumor) expression data is available. The choice of this disease is two-fold: in this repository, this is one of the two diseases with both tumor and normal cells. Furthermore, recent studies have revealed distinct patterns of miRNA expression in tumor compared to normal brain [187], and several miRNA targets have in fact been experimentally verified [188]–[190]. The disease samples are characterized by rapid proliferation and stem-cell like behavior which is possibly caused by malfunctioning of characteristic pathways [191]. Mutations in miRNA and miRNA targets have been postulated to be involved in tumorgenesis, but have not been specifically identified in GBM.

The expression profiles downloaded consist of a total of 58 matched mRNA and miRNA samples from the Agilent 244k aCGH platform at data level 3. We used the following parameters on mirConnX: Gene Symbols, miRBase ID, Pearson correlation with a prior weight of 0.3, and

0.9 as the display cutoff threshold. A total of 56 miRNAs, 29 TFs, and 1180 genes form a network with a total of 1851 connections. Of these interactions, 43 are miR-TF regulations, 34 TF-gene connections, and 1774 miRNA-gene connections.

Among the top interactions, we were able to identify two hubs miR-21, miR-326 and miR-34a and miR-137 that have been verified to be miRNAs involved in Glioblastoma. These two miRNAs are also hubs with some of the highest degree centrality [192], sharing many targets and TFs with other hubs. miR-21 has been found to be one of the most highly expressed miRNAs in many cancer types, and it has been shown that miR-21 acts as an oncogene in Glioblastoma by suppressing apoptosis [193]. Among the highest ranking targets we predict for miR-21, SOX2 [194] and TGFB pathway [195] were shown to be regulated by the miRNA. RECK and PDCD4 have been experimentally verified, in vivo and in vitro, to be involved in proliferation [196]. In addition, PELI1 and CDC25A have been shown in other cancer types to play a role in apoptosis [197]–[199]. Similarly, miR-137 has been shown to be involved in proliferation and neuronal differentiation in vitro [119]. Indeed, both CDK6 and MITF, the experimentally verified targets from the study were also predicted in our network.

A thorough literature search on all of the predicted interactions for Glioblastoma is not possible here, but we demonstrated that mirConnX is useful for identifying hub genes, their regulators, and their targets involved in diseases, the pathways involved, and could potentially be a powerful tool for clinical scientists to create a list of top candidate genes and forming hypotheses.

4.5 MIRCONNX 2.0

While each of the methods proposed in the first two aims constituted standalone studies and the source code will be made available separately, they were all developed toward the same goal of deciphering disease complexities and should be integrated. We expanded the mirConnX web environment to streamline all the methods to provide an integrated view to enable further user exploration and hypothesis generation. Figure 4.3 demonstrates the vision of the end product of this effort. As before, users supply mRNA and/or miRNA gene expression. In addition, they have the option of uploading clinical labels of interest that correspond to the same samples in the gene expression profiles. Examples include disease subtypes, normal/control labels, progression free survival, or relapse events.

Gene expression and miRNA expression, if supplied, are clustered separately by ReKS to produce coherent clusters that are candidate molecular signatures, potentially taking advantage of pathway information from KEGG pathway using the prior incorporation scheme. Next, using the group variable selection developed in Chapter 3, molecular signatures that are predictive of disease subtypes or user-input labels are identified. Finally, transcriptional and post-transcriptional regulatory information are provided to genes within and between the molecular signatures. In summary, a network with genes, miRNAs and TFs is generated with the following: (1) expression correlation edges indicating **strength and sign of association** (2) **dynamic cluster boundary** set by user-defined significance thresholds (3) **transcription factor regulations** and strength, between selected TFs-selected genes/miRs, otherTFs-selected miRS, as well as selected TFs-otherTFs (4) **miRNA regulations and strength**, between selected miRs-selected genes/TFs, selected miRs-co regulators or targest of other selected miRs/genes, and potentially (5) protein-protein interaction or (6) pathway boundary and interaction.

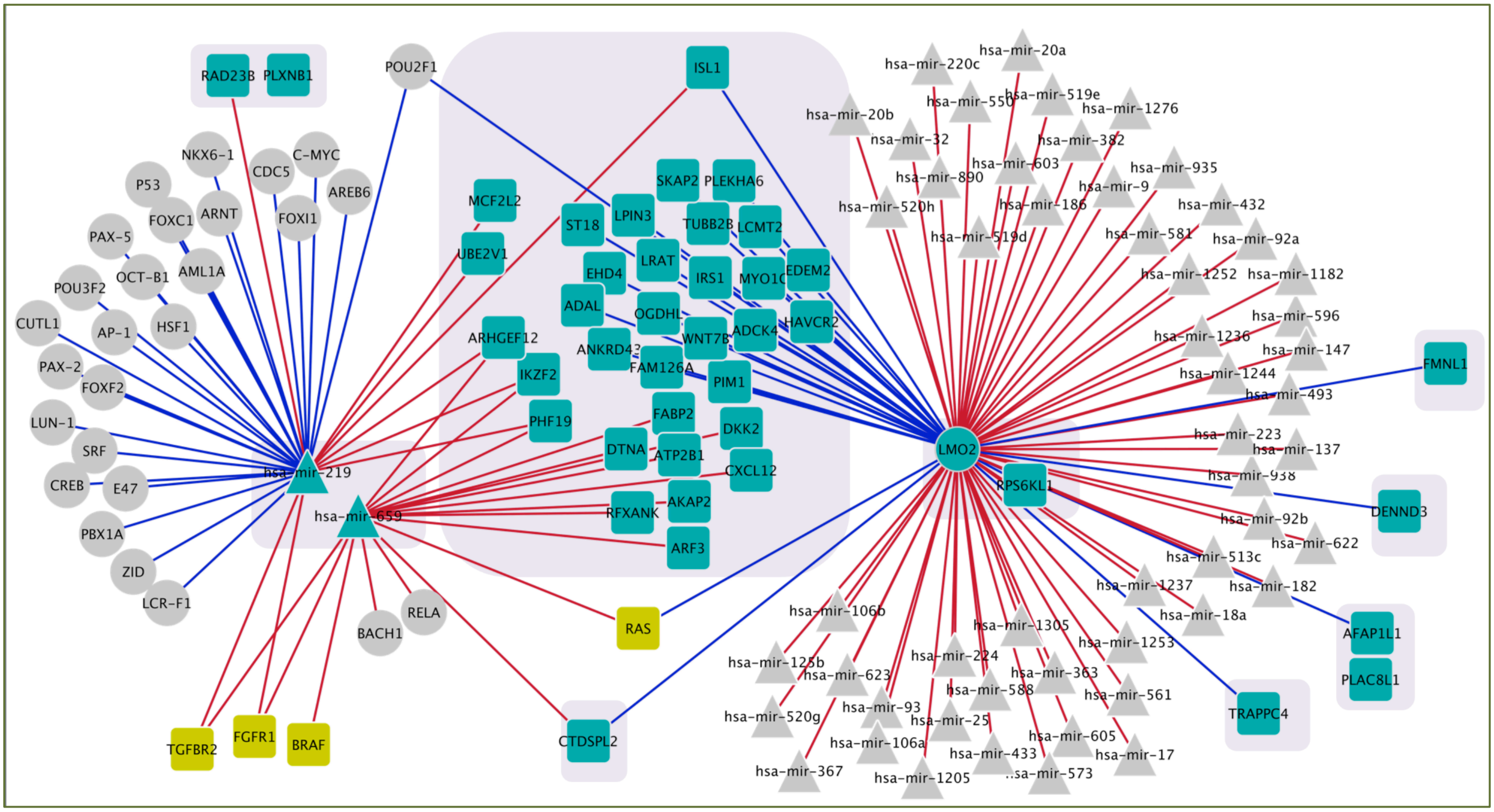


Figure 4.3 Integrative analysis with mirConnX2.0 on melanoma.
mirConnX 2.0, an integrated framework that includes clustering, feature selection, and regulatory relationship enrichment, was applied on the Melanoma data detailed in Section 3.5.2. Genes and miRNA clusters that are predictive of patient survival are selected (green). Cluster membership is indicated in light purple boxes. Several oncogenes known to be involved in melanoma disease mechanisms are highlighted in gold. Genes are represented by squares, miRNAs by triangles, and TFs by circles. Blue edges indicate TF-> gene regulatory interactions supported by literature or computational predictions. Red edges indicate miRNA-> gene targeting supported by literature or computational predictions. For visual simplicity, strength of association (regulatory strength plus correlation) and sign of association (repression or activation) are omitted in this figure, but are available to display as an option.

As a proof of concept, we illustrate the full power of mirConnX 2.0 by revisiting the melanoma mRNA and miRNA expression datasets used in the case study in Section 3.5.2. We applied mirConnX 2.0 to this dataset, using no prior clustering information. As shown in Figure 4.3, the nine groups of selected miRNAs and genes are shown in green, inside light purple boxes indicating their group memberships. mirConnX 2.0 automatically extract the biological context around these variables, including TF-> gene regulatory relationships and miRNA-> gene targeting relationships.

Several preliminary observations can be made from this powerful analysis: 1) we notice several known oncogenes to be jointly regulated by prognostic biomarkers, for example RAS by has-miR-659 and LMO2, and a few others jointly by has-miR-219 as well as has-miR-659. 2) POU2F1(also known as OCT1) acts a the master regulator in this snapshot of the network, regulating both of the hub clusters (has-miR-219/has-miR-659, and LMO2/RPS56KL1). It binds to an octamer DNA sequence and is known to have cell type-specific effects on differentiation, but no clear association with Melanoma. However, another protein in the same family, POU3F2(also known as BRN2/OCT5), has been shown to be linked to melanoma proliferation by participating in the Wnt signaling as an early factor in melanoblasts that negatively regulated differentiation [230,231]. This observation presents POU2F1 and reinforces POU2F2 to be interesting targets for investigation, and one can already start generating interesting hypothesis about possible feedback loops they may form with the two hub regulators. 3) On the other hand, ISL1 and CTDSPL2 are regulated by both of the selected hub clusters. Neither have apparent association with melanoma, but ISL1 is phosphorylated by Rho kinase whose dysregulation contributes to the metastatic behavior of many tumor types including melanoma [232,233] and this pathway has been targeted by several clinical studies for anticancer therapeutics. Thus, we may want to extend our network to additionally include interactive partners of these genes.

These simple, preliminary observations demonstrate that by placing the selected biomarkers in their biological context and linking them through regulatory interactions, a much simpler, cleaner organization of the biomarkers and regulatory partners surface. In addition, relative biological importance of the biomarkers are immediately clear in a network context, with hub clusters being primary interests to many as they are ideal candidates for understanding dynamics and mechanism of disease formation, and they present possible points of attacks for potential intervention. Even for a cluster with hundreds of members shown in this example, only a subset will be of immediate interests to bench scientists and failure in properly defining a set of computational thresholds for cluster selection by the user can be in most part remedied by this type of context information “filtering”. Finally, an advantage mirConnX 2.0 has over its predecessor is that the size of the final network is now very manageable, as we now focus on the part of the network containing mainly prognostic clusters.

We expect that users will be able to adjust size of the network by adjusting the significance thresholds (*p*-valule for conditional independence tests) in Section 3.3.2.2. The basic set of selected genes and miRNAs will remain unchanged. However, the size of the cluster will grow and shrink, and members of the clusters as well as regulatory relationships will be shown or hidden accordingly.

Additionally, users will be allowed to upload additional prior grouping information to be used in the prior incorporation scheme in Section 2.5. Example of the prior information that a user may want to supply include pathway information (from KEGG [88] or Ingenuity[234], for example), Protein-protein interaction, regulatory modules or co-regulation information, and domain expertise. Not all the prior information would be necessarily suitable, but it will be up to the user’s discretion.

4.6 DISCUSSION AND FUTURE DIRECTIONS

In recent years, with the availability of condition-specific high-throughput mRNA and miRNA expression data, there is an increasing need of integrated environment that combines data analyses and visualization in the form of constructing hypothesized networks. While many methods exist for either network generation using only expression data, only binding affinity experiments such as ChIP-chip, or even manually curated data from expert knowledge databases, an integrated network that maximally exploits information in both domains is lacking. Additionally, there has not been many attempts to incorporate both TF and miRNA regulations, yet it has become increasingly clear that miRNAs play a crucial role in human diseases. mirConnX is a novel web tool developed specifically to fill the niche. The utility of mirConnX lies in its ability to integrate user-supplied data with pre-compiled information of miRNA targeting and TF binding, and generate a network that reflects characteristics specific to the data guided by some prior beliefs. The user-friendly display of interaction networks and other downstream analyses also provides an integrated environment for clinical researchers to perform further investigation and exploration.

We present mirConnX, a web tool developed to integrate user-supplied expression data with pre-compiled information of miRNA targeting and TF binding, and generate a network that reflects characteristics specific to the data guided by some prior beliefs. Coupled with the clustering methods developed in Chapter 2 and feature selection method from Chapter 3, we expect to have a powerful pipeline for analyzing disease expression data. To the best of our knowledge, mirConnX 2.0 will be the only interactive web-stool to date that allows the user to explore *groups* of prognostic biomarkers in the context of regulatory relationships. The user-friendly display of the resulting networks and other downstream analyses will also provide an integrated environment for clinical researchers to perform further investigation and exploration.

5.0 CONCLUSION

5.1 CONCLUDING REMARKS

The emerging model of personalized medicine aims to personalize disease risk assessment and improve responsiveness to treatments. Treatment plans will only be successful if appropriate biomarkers are identified to help guide the selection of the most beneficial treatment for a given patient. Traditional biomarkers have been inadequate in providing proper disease subtyping, and the community has been looking beyond the traditional assays and diagnostic tests in search for novel prognostic biomarkers.

The advances in molecular profiling technologies have changed our understandings of cancer and led to the identification of such prognostic/predictive gene signatures. Despite the huge quantity of information gleaned from these profiling technologies and the increasing number of gene signatures proposed, their validation and incorporation into clinical decision making is a slow and limited. This is in part due to a number of challenging issues impeding the adoption of traditional feature selection algorithms to high-dimensional genomics datasets.

In my view, current biomarker discovery methods are hampered by several complications. Computationally, significant correlation structure exists between the variables of interests. Partly as a result, the selected features often occur in redundancy and are highly unstable. Clinically, the data displays a high degree of heterogeneity. Furthermore, existing biomarker discovery methods does not adequately represent the increasingly popular pathway-centric view of disease formation, and selected variables often lack biological relevance. As a whole, these problems present the current challenges of applying feature selection and other computational methods to high-throughput genomics data. Individually, many of these issues are prevalent among other noisy, high-dimensional systems as well.

We demonstrated in this dissertation our attempts to address these key issues through a novel module-based, biologically informed biomarker discovery framework. Specifically, we tackle these issues by developing (1) an efficient clustering algorithm suitable for heterogeneous clinical expression data, (2) a novel feature selection method that operates on groups of variables, with results delivered in the form of clusters of maximally predictive genes (includes TFs and miRNAs) for a given clinical label of interest, and (3) an integrated environment for these two methods, enhanced with relevant biological information, delivered in the form of a user-friendly web server.

The deliverables of this combined framework include a small list of candidate genes to be tested on the bench, as well as relevant biological organization and regulatory information to prioritize the targets of interests. As a by-product of the framework, given a new patient sample, we would be able to predict disease classification or clinical outcome using the molecular signatures identified. The publicly available web-tool will enable researchers to confirm existing biomarkers and generate hypothesis about novel causes for diseases.

Our algorithms are designed specifically to tackle high-throughput clinical data. While we primarily used gene expression data in this dissertation to demonstrate the utility of our methods, the same framework could be augmented to achieve variable space partition and group variable selection using other types of genomic measurements as features.

5.2 FUTURE DIRECTIONS

The work described here can be refined in several ways.

Each of these algorithms requires further testing for one to gain a deeper understanding of their behaviors across different types of datasets and parameter range. For ReKS, different distance measures and affinity definitions will invariably affect the clustering result, and a thorough investigation is necessary. Development of a parallelized SVD (personal communications with Chennubhotla and Quinn *et al.*) is currently under way and once available, ReKS can adopt the parallelization to allow partitioning of data beyond the current size limits. I would like to take the perturbation-based algorithm further to interrogate optimum thresholds for producing the most stable partitions. Finally, this approach can potentially be extended toward biclustering of heterogeneous clinical genomics datasets. Similarly, for T-ReCS a specific parameter range for which desirable algorithmic behaviors can be expected need to be more precisely defined. As feature selection performance is tightly coupled with classification or regression methods, a comprehensive study of T-ReCS using combinations of different classification and regression methods should be conducted. mirConnX can benefit from integration with a wider variety of data types and databases, and addition of common bioinformatics practices such as Gene Ontology analysis.

From a more holistic perspective, I believe that our work paves the way for a larger effort in which a variety of genomics data types such as mutation, copy number variation, methylation and genome aberrations can be incorporated to create a more comprehensive view on disease formation and progression. My long-term vision for our existing web-server is for it to become an open-source repository where networks and biomarker clusters generated by users can be deposited and queried, and prognostic analysis automatically generated with each addition of a new sample.

Through the proposed framework, we took a small step toward improving molecular biomarker discovery. My hope is that our tools will enable other researchers to take a larger stride toward the ultimate goal of improving disease prognosis and deciphering the complex underpinning of human diseases. .

APPENDIX A

CONDITIONAL INDEPENDENCE TESTES FOR CATEGORICAL DATA

Suppose we would like to test $H_0\colon Ind(X;T|Z)$ for the case where both the data (D; containing X and Z) and the target variable T are both categorical. We can create a contingency table where each cell in the table holds S_{XTZ}^{abc} , the counts of the number of times $X = a, T = b$, and $Z = c$ appear in the data. Similarly, we can define marginal counts S_{XZ}^{ac}, S_{TZ}^{bc}, and S_{Z}^{c}, and we can calculate the expected count under the null hypothesis of conditional independence as:

$$P(X,Y,Z) = P(Z)P(X|Z)P(Y|Z)$$

$$=(S_{XZ}^{ac} \cdot S_{TZ}^{bc})/S_{Z}^{c}$$

The deviance between the observed counts and the expected count can be defined either though the G^2 statistic under the likelihood ratio framework [134]:

$$G^2 = 2\sum Observed\ ln\frac{Observed}{Expected} = 2\sum_{a,b,c} S_{XTZ}^{abc}\ ln\frac{S_{XTZ}^{abc}}{(S_{XZ}^{ac} \cdot S_{TZ}^{bc})/S_{Z}^{c}}$$

Or the χ^2 statistic as a direct measure of goodness-of-fit:

$$\chi^2 = \sum\frac{(Observed - Expected)^2}{Expected} = \sum_{a,b,c}\frac{(S_{XTZ}^{abc} - (S_{XZ}^{ac} \cdot S_{TZ}^{bc})/S_{Z}^{c})^2}{(S_{XZ}^{ac} \cdot S_{TZ}^{bc})/S_{Z}^{c}}$$

They are asymptotically distributed as χ^2 with degree of freedom df:*(foot note: assuming no structure zeros. If there are cells with counts of zero, one can either follow the practice of Spirtes GS or [125] to calculate the effective number of parameters)

$$df = (|D(X)| - 1) \cdot (|D(T)| - 1) \prod_{k \subseteq Z} |D(k)|$$

Where $D(X)$ is the domain of variable X, i.e. the values that X can take on. The statistic with the given degree of freedom produces a p-value that corresponds to the opposite of strength of association. If the p-value is less than the significance level a, the null hypothesis of conditional independence is rejected.

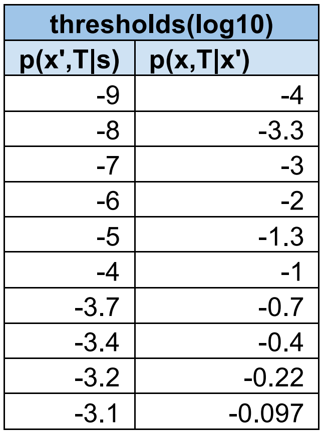
In the original implementation of MMPC, the χ^2 statistic is used since it is asymptotically correct for discrete multinomial distribution, and is easy to compute. One can see that the number of cells increase exponentially with the number of conditioning variables, and the number of training samples can become inadequate. For reliable estimation, it was recommended that the sample size be at least five times the number of cells. This test was implemented in the original MMPC[5].

[5] Note that there are better alternative tests of conditional independence for categorical data, for example mutual information and Bayesian test proposed by Margaritis and Thrun [97].

APPENDIX B

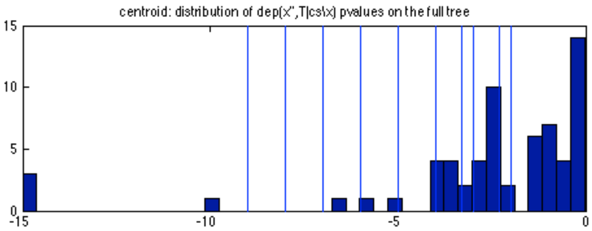
T-RECS SIGNIFICANCE THRESHOLDS

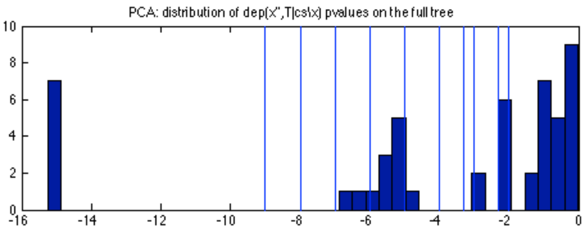
The following thresholds are used for T-ReCS on the simulated data.

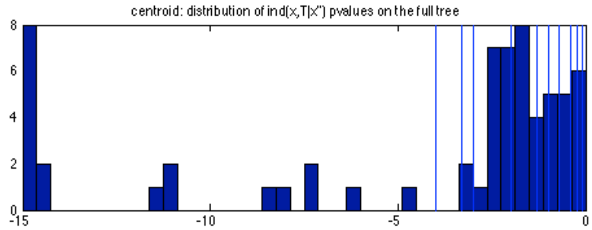


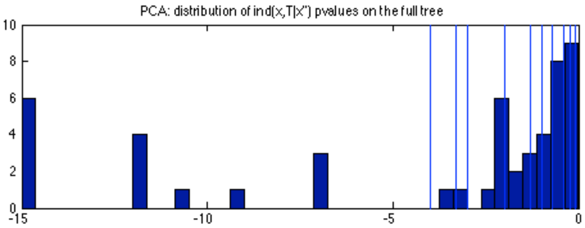
thresholds(log10)	
p(x',T|s)	p(x,T|x')
-9	-4
-8	-3.3
-7	-3
-6	-2
-5	-1.3
-4	-1
-3.7	-0.7
-3.4	-0.4
-3.2	-0.22
-3.1	-0.097

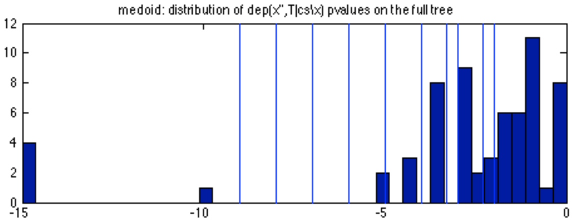
The following histograms are *P*-value distributions of the T-ReCS conditional independence tests performed on the simulated data, plotted on log10 scale, based on three different collapsing methods. The significance levels tested are plotted as blue vertical lines over the histograms.

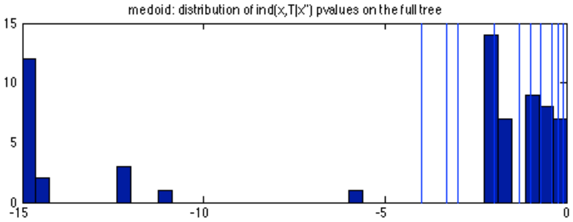

centroid: distribution of dep(x'',T|cs\x) pvalues on the full tree
15
10
5
0
-15
-10
-5
0


PCA: distribution of dep(x'',T|cs\x) pvalues on the full tree
10
8
6
4
2
0
-16
-14
-12
-10
-8
-6
-4
-2
0


centroid: distribution of ind(x,T|x'') pvalues on the full tree
8
6
4
2
0
-15
-10
-5
0


PCA: distribution of ind(x,T|x'') pvalues on the full tree
10
8
6
4
2
0
-15
-10
-5
0

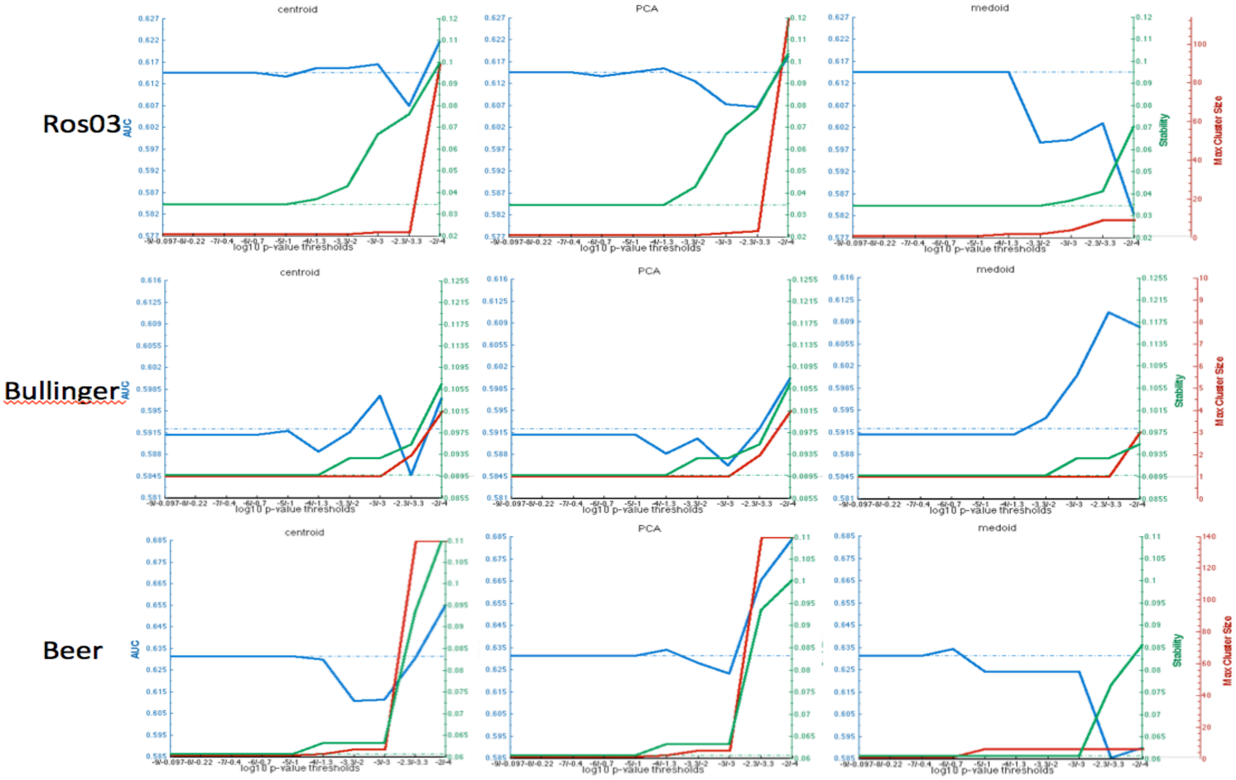

medoid: distribution of dep(x'',T|cs\x) pvalues on the full tree
12
10
8
6
4
2
0
-15
-10
-5
0

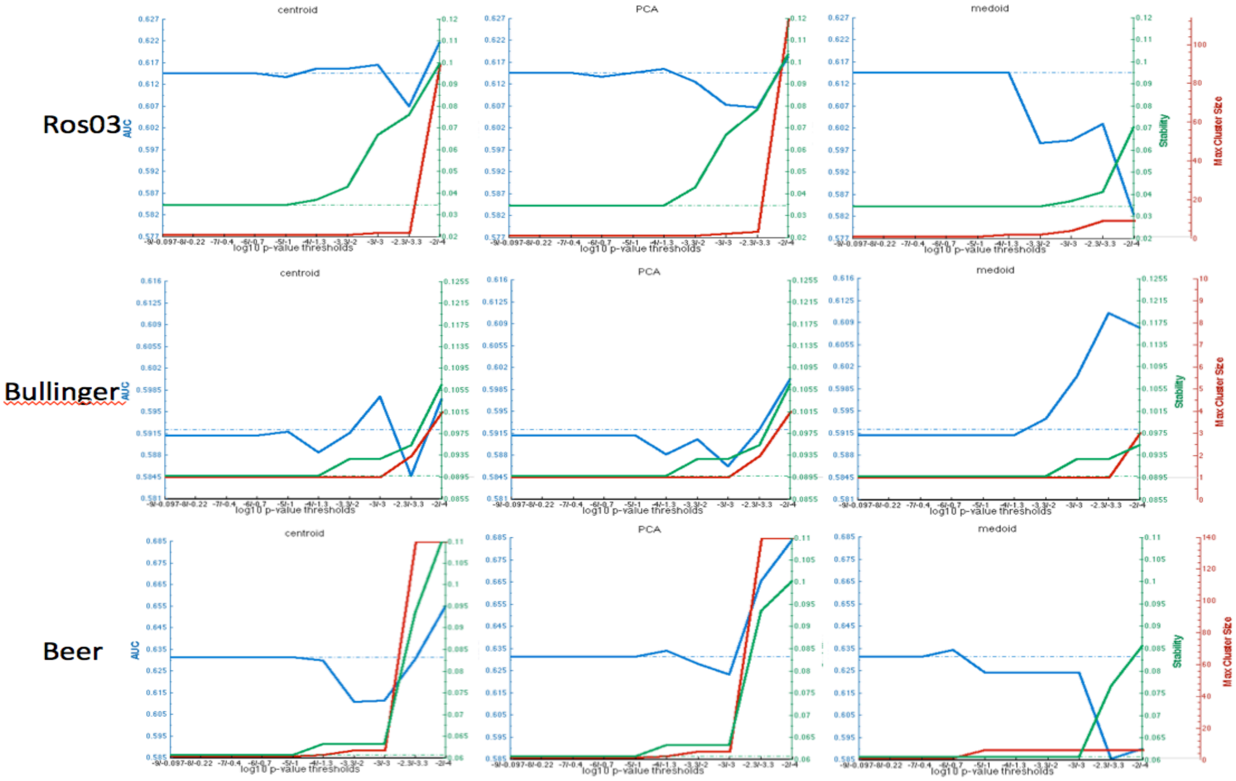

medoid: distribution of ind(x,T|x'') pvalues on the full tree
15
10
5
0
-15
-10
-5
0

APPENDIX C

SURVIVAL BENCHMARKING DATA YYY PLOTS

Performance of T-ReCS across 6 benchmarking survival datasets.




Ros03
Bullinger
Beer
centroid
PCA
medoid
AUC
Stability
Max Cluster Size
log10 p-value thresholds
-9/-0.097 -8/-0.22 -7/-0.4 -6/-0.7 -5/-1 -4/-1.3 -3.3/-2 -3/-3 -2.3/-3.3 -2/-4

APPENDIX D

SELECTED GENES AND MIRNAS FROM MELANOMA DATASET

The following gene clusters are selected from the melanoma dataset thresholds **dep(x',T|s)=**10^{-2} and **ind(x,T|x’)=**10^{-4}**.** Highlighted in red are the seed single variables from which they grew from. The clusters are presented in the order of significance.

{**AFAP1L1**,PLAC8L1},{**PLXNB1**,RAD23B},{ADO,BCMO1,BEGAIN,C20ORF7,C21ORF84,C9ORF61,CCT8L2,ERLIN2,LMO2,**LOC389936**,LOC440053,MIR99A,OPN1SW,OR9Q2,OSM,PSD4,RNASE9,RPS6KL1,SMPD1,TBCEL,TLR5,TMEM121,ZNF552},{AAGAB,ACRBP,ACTR3B,ADAL,ADCK4,ADCY10,ADD1,AFP,AK3,AKAP2,AMY2A,ANKRD37,ANKRD43,AP2B1,APOBEC3C,ARF3,ARHGEF12,ARHGEF7,ARVP6125,ARX,ASB11,ATAD3B,ATE1,ATP2B1,ATP8A1,ATXN8OS,AUH,BATF3,BDP1,BLVRA,BST2,BTK,C10ORF113,C11ORF54,C12ORF73,C14ORF112,C14ORF174,C15ORF39,C15ORF42,C15ORF62,C17ORF85,C19ORF21,C1ORF165,C20ORF86,C21ORF100,C21ORF42,C4ORF15,C4ORF37,C5ORF54,C7ORF63,CACNA1I,CALD1,CALML5,CAMKK1,CAPN6,CBL,CCDC54,CCRK,CDC42SE1,CDH23,CEACAM18,CHAT,CHFR,CHMP4B,CHP2,CIDECP,CISD2,CLIP3,CLN8,CLOCK,CLRN2,CMA1,COX16,COX7A2L,CSF2,CTTNBP2NL,CXCL12,CXXC1,DECR1,DEK,DENND3,DKK2,DNAJC27,DPF2,DPM2,DTNA,ECGF1,EDEM2,EDIL3,EHD4,ELK4,ELOVL1,EME2,EPHX2,ERCC-00131,ERCC00136,EVI1,FABP2,FADS2,FAM108B1,FAM126A,FAM75B,FBRS,FBXO31,FGF3,FKBP7,FLAD1,FLJ41649,FLJ45079,FNDC3B,FOXB2,FRMD1,FUS,GAS2L1,GDF7,GFRA1,GJC3,GLMN,GLT1D1,GNPDA1,GPR1,GRRP1,GUCA1B,HAND1,HAVCR2,HBM,HDGFRP3,HIST1H2BD,HNF1B,HNRNPH3,HOOK1,HPN,HRG,HS3ST6,HSD11B1,ID1,IKZF2,IRS1,ISL1,ITGB4,JUND,KCNQ2,KLK1,KLRC3,KREMEN2,LBXCOR1,LCMT2,LILRA5,LILRA6,LIN7A,LOC100133144,LOC374395,LOC388955,LOC400986,LOC401286,LOC554235,LOC641522,LONRF2,LPIN3,LRAT,LRP5L,MANBA,MAPK7,MCF2L2,MEA1,MED14,MIMT1,MIST,MMP14,MMP24,MOCS1,MORN1,MRPL20,MTF2,MTNR1A,MYBPC2,MYO1C,MYOF,NARS2,NCCRP1,NDC80,NDUFA1,NFAM1,NFYA,NHLH1,NIF3L1,NKX6-2,NMBR,N-PAC,NUDT4P1,ODF2,OGDHL,OPRK1,OR11G2,OR11L1,OR12D2,OR51A2,OR5M11,OR5V1,OR7A10,OR7G2,P2RY12,PBK,PCBD1,PCOTH,PDCD1LG2,PDCD4,PDE6C,PDILT,PEX3,PFN3,PGM5P2,PHF19,PIM1,PIP4K2B,PKD1L2,PLCB3,PL

EKHA6,PLEKHA8,POF1B,POL3S,PRAMEF13,PRH1,PROCA1,PRSSL1,PSEN2,PSTPIP2,PTPN14,PVRIG,PWWP2B,RAP1GAP,RASAL3,RBMY1J,RELL1,RFXANK,RG9MTD1,RNF133,RNU15,RNU1A3,ROD1,RPAIN,RPL34,RPS16,RPS3,RXFP3,SAMD13,SDCCAG10,SDK2,SDSL,SEC24C,,SERAC1,SERPINA3,SERPINA7,SF3B14,SFXN2,SH3BP5,SHD,**SKAP2**, SLC12A5,SLC12A7,SLC12A9,SLC17A2,SLC17A8,SLC1A6,SLC25A32,SLC4A5,SLC7A6OS,SNORA7A,SNORD114-13,SNORD11548,SNORD72,SOCS6,SORCS3,SOST,SOX30,SPACA1,SPATA3,SPNS3,SRP14P1,SST,ST18,STEAP4,TAL2,TBX5,TCEA3,TCL1A,TIA1,TIMM8A,TK1,TLR9,TMEM132E,TMEM39B,TMEM66,TRAF2,TRERF1,TRIM32,TRIP4,TTTY17B,TUBB2B,UBE1,UBE2CBP,UBE2V1,UBXN10,UFD1L,ULK2,UTX,VAC14,VSIG1,VWA2,VWF,WDR33,WDR37,WDR52, WNT7B,ZBTB16,ZFAND6,ZNF131,ZNF526,ZNF639,ZNF703},{AGR2,C16ORF35,CCDC64,CNTD2,**CRYBB3**,CTDSPL2,HSPBAP1,OPN1SW,TAAR8},{OR8H3,**TRAPPC4**},{**C6ORF224**},{**USP51**},{FAM23A,FMNL1,**LRDD**}

The following miRNA cluster is the only one selected from the melanoma dataset at thresholds **dep(x',T|s)=10**$^{-1}$ and **ind(x,T|x’)=10**$^{-2}$**.** Highlighted in red are the seed single variables from which they grew from.

{hsa-miR219-1-3p, hsa-miR516a-5p, **hsa-miR659-3p**}

BIBLIOGRAPHY

[1] R. L. Schilsky, “Personalized medicine in oncology: the future is now,” *Nat Rev Drug Discov*, vol. 9, no. 5, pp. 363–366, May 2010.

[2] M. A. Hamburg and F. S. Collins, “The Path to Personalized Medicine,” *New England Journal of Medicine*, vol. 363, no. 4, pp. 301–304, 2010.

[3] “Post-Treatment Biomarker Changes Improved Breast Cancer Survival - Cancer Network,” 11-Sep-2013. [Online]. Available: http://www.cancernetwork.com/conference-reports/asco2013/breast-cancer-symposium/content/article/10165/2157691. [Accessed: 24-Sep-2013].

[4] T. Sørlie, R. Tibshirani, J. Parker, T. Hastie, J. S. Marron, A. Nobel, S. Deng, H. Johnsen, R. Pesich, S. Geisler, J. Demeter, C. M. Perou, P. E. Lønning, P. O. Brown, A.-L. Børresen-Dale, and D. Botstein, “Repeated observation of breast tumor subtypes in independent gene expression data sets,” *PNAS*, vol. 100, no. 14, pp. 8418–8423, Jul. 2003.

[5] B. P. Schneider, E. P. Winer, W. D. Foulkes, J. Garber, C. M. Perou, A. Richardson, G. W. Sledge, and L. A. Carey, “Triple-Negative Breast Cancer: Risk Factors to Potential Targets,” *Clin Cancer Res*, vol. 14, no. 24, pp. 8010–8018, Dec. 2008.

[6] R. W. Dunstan, K. A. Wharton, C. Quigley, and A. Lowe, “The Use of Immunohistochemistry for Biomarker Assessment—Can It Compete with Other Technologies?,” *Toxicol Pathol*, vol. 39, no. 6, pp. 988–1002, Oct. 2011.

[7] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander, “Molecular classification of cancer: class discovery and class prediction by gene expression monitoring,” *Science*, vol. 286, no. 5439, pp. 531–537, Oct. 1999.

[8] 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.*, vol. 98, no. 26, pp. 15149–15154, Dec. 2001.

[9] L. J. van ’t Veer, H. Dai, M. J. van de Vijver, Y. D. He, A. A. M. Hart, M. Mao, H. L. Peterse, K. van der Kooy, M. J. Marton, A. T. Witteveen, G. J. Schreiber, R. M. Kerkhoven, C. Roberts, P. S. Linsley, R. Bernards, and S. H. Friend, “Gene expression profiling predicts clinical outcome of breast cancer,” *Nature*, vol. 415, no. 6871, pp. 530–536, Jan. 2002.

[10] S. Paik, S. Shak, G. Tang, C. Kim, J. Baker, M. Cronin, F. L. Baehner, M. G. Walker, D. Watson, T. Park, W. Hiller, E. R. Fisher, D. L. Wickerham, J. Bryant, and N. Wolmark, “A multigene assay to predict recurrence of tamoxifen-treated, node-negative breast cancer,” *N. Engl. J. Med.*, vol. 351, no. 27, pp. 2817–2826, Dec. 2004.

[11] S. Glück, F. de Snoo, J. Peeters, L. Stork-Sloots, and G. Somlo, “Molecular subtyping of early-stage breast cancer identifies a group of patients who do not benefit from neoadjuvant chemotherapy,” *Breast Cancer Res. Treat.*, vol. 139, no. 3, pp. 759–767, Jun. 2013.

[12] M. E. Straver, A. M. Glas, J. Hannemann, J. Wesseling, M. J. van de Vijver, E. J. T. Rutgers, M.-J. T. F. D. Vrancken Peeters, H. van Tinteren, L. J. Van’t Veer, and S. Rodenhuis, “The 70-gene signature as a response predictor for neoadjuvant chemotherapy in breast cancer,” *Breast Cancer Res. Treat.*, vol. 119, no. 3, pp. 551–558, Feb. 2010.

[13] O. Krijgsman, P. Roepman, W. Zwart, J. S. Carroll, S. Tian, F. A. de Snoo, R. A. Bender, R. Bernards, and A. M. Glas, “A diagnostic gene profile for molecular subtyping of breast cancer associated with treatment response,” *Breast Cancer Res. Treat.*, vol. 133, no. 1, pp. 37–47, May 2012.

[14] M. Dowsett and A. K. Dunbier, “Emerging Biomarkers and New Understanding of Traditional Markers in Personalized Therapy for Breast Cancer,” *Clin Cancer Res*, vol. 14, no. 24, pp. 8019–8026, Dec. 2008.

[15] M. J. van de Vijver, Y. D. He, L. J. van ’t Veer, H. Dai, A. A. M. Hart, D. W. Voskuil, G. J. Schreiber, J. L. Peterse, C. Roberts, M. J. Marton, M. Parrish, D. Atsma, A. Witteveen, A. Glas, L. Delahaye, T. van der Velde, H. Bartelink, S. Rodenhuis, E. T. Rutgers, S. H. Friend, and R. Bernards, “A Gene-Expression Signature as a Predictor of Survival in Breast Cancer,” *New England Journal of Medicine*, vol. 347, no. 25, pp. 1999–2009, 2002.

[16] J. A. Sparano and S. Paik, “Development of the 21-gene assay and its application in clinical practice and clinical trials,” *J. Clin. Oncol.*, vol. 26, no. 5, pp. 721–728, Feb. 2008.

[17] J. S. Parker, M. Mullins, M. C. U. Cheang, S. Leung, D. Voduc, T. Vickery, S. Davies, C. Fauron, X. He, Z. Hu, J. F. Quackenbush, I. J. Stijleman, J. Palazzo, J. S. Marron, A. B. Nobel, E. Mardis, T. O. Nielsen, M. J. Ellis, C. M. Perou, and P. S. Bernard, “Supervised Risk Predictor of Breast Cancer Based on Intrinsic Subtypes,” *JCO*, vol. 27, no. 8, pp. 1160–1167, Mar. 2009.

[18] D. J. Lockhart, H. Dong, M. C. Byrne, M. T. Follettie, M. V. Gallo, M. S. Chee, M. Mittmann, C. Wang, M. Kobayashi, H. Horton, and E. L. Brown, “Expression monitoring by hybridization to high-density oligonucleotide arrays,” *Nat. Biotechnol.*, vol. 14, no. 13, pp. 1675–1680, Dec. 1996.

[19] J. L. DeRisi, V. R. Iyer, and P. O. Brown, “Exploring the metabolic and genetic control of gene expression on a genomic scale,” *Science*, vol. 278, no. 5338, pp. 680–686, Oct. 1997.

[20] “LGRC - Lung Genomics Research Consortium.” [Online]. Available: https://www.lung-genomics.org/research/. [Accessed: 19-Sep-2013].

[21] C. Curtis, S. P. Shah, S.-F. Chin, G. Turashvili, O. M. Rueda, M. J. Dunning, D. Speed, A. G. Lynch, S. Samarajiwa, Y. Yuan, S. Gräf, G. Ha, G. Haffari, A. Bashashati, R. Russell, S. McKinney, Metabric Group, A. Langerød, A. Green, E. Provenzano, G. Wishart, S. Pinder, P. Watson, F. Markowetz, L. Murphy, I. Ellis, A. Purushotham, A.-L. Børresen-Dale, J. D. Brenton, S. Tavaré, C. Caldas, and S. Aparicio, “The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups,” *Nature*, vol. 486, no. 7403, pp. 346–352, Jun. 2012.

[22] “The Cancer Genome Atlas - Data Portal.” [Online]. Available: https://tcga-data.nci.nih.gov/tcga/tcgaHome2.jsp. [Accessed: 31-Jul-2012].

[23] R. Edgar, M. Domrachev, and A. E. Lash, “Gene Expression Omnibus: NCBI gene expression and hybridization array data repository,” *Nucleic Acids Res.*, vol. 30, no. 1, pp. 207–210, Jan. 2002.

[24] S. T. Smale and J. T. Kadonaga, “The RNA polymerase II core promoter,” *Annu. Rev. Biochem.*, vol. 72, pp. 449–479, 2003.

[25] P. Carninci, A. Sandelin, B. Lenhard, S. Katayama, K. Shimokawa, J. Ponjavic, C. A. M. Semple, M. S. Taylor, P. G. Engström, M. C. Frith, A. R. R. Forrest, W. B. Alkema, S. L. Tan, C. Plessy, R. Kodzius, T. Ravasi, T. Kasukawa, S. Fukuda, M. Kanamori-Katayama, Y. Kitazume, H. Kawaji, C. Kai, M. Nakamura, H. Konno, K. Nakano, S. Mottagui-Tabar, P. Arner, A. Chesi, S. Gustincich, F. Persichetti, H. Suzuki, S. M. Grimmond, C. A. Wells, V. Orlando, C. Wahlestedt, E. T. Liu, M. Harbers, J. Kawai, V. B. Bajic, D. A. Hume, and Y. Hayashizaki, “Genome-wide analysis of mammalian promoter architecture and evolution,” *Nat. Genet.*, vol. 38, no. 6, pp. 626–635, Jun. 2006.

[26] S. Y. Ng, P. Gunning, S. H. Liu, J. Leavitt, and L. Kedes, “Regulation of the human beta-actin promoter by upstream and intron domains,” *Nucleic Acids Res.*, vol. 17, no. 2, pp. 601–615, Jan. 1989.

[27] G. D. Stormo, “DNA binding sites: representation and discovery,” *Bioinformatics*, vol. 16, no. 1, pp. 16–23, Jan. 2000.

[28] A. Sandelin and W. W. Wasserman, “Constrained binding site diversity within families of transcription factors enhances pattern discovery bioinformatics,” *J. Mol. Biol.*, vol. 338, no. 2, pp. 207–215, Apr. 2004.

[29] S. Mahony, P. E. Auron, and P. V. Benos, “DNA familial binding profiles made easy: comparison of various motif alignment and clustering strategies,” *PLoS Comput. Biol.*, vol. 3, no. 3, p. e61, Mar. 2007.

[30] M. Umetani, C. Mataki, N. Minegishi, M. Yamamoto, T. Hamakubo, and T. Kodama, “Function of GATA transcription factors in induction of endothelial vascular cell adhesion molecule-1 by tumor necrosis factor-alpha,” *Arterioscler. Thromb. Vasc. Biol.*, vol. 21, no. 6, pp. 917–922, Jun. 2001.

[31] D. L. Corcoran, E. Feingold, J. Dominick, M. Wright, J. Harnaha, M. Trucco, N. Giannoukakis, and P. V. Benos, “Footer: a quantitative comparative genomics method for efficient recognition of cis-regulatory elements,” *Genome Res.*, vol. 15, no. 6, pp. 840–847, Jun. 2005.

[32] P. Huggins, S. Zhong, I. Shiff, R. Beckerman, O. Laptenko, C. Prives, M. H. Schulz, I. Simon, and Z. Bar-Joseph, “DECOD: fast and accurate discriminative DNA motif finding,” *Bioinformatics*, vol. 27, no. 17, pp. 2361–2367, Sep. 2011.

[33] T. L. Bailey, N. Williams, C. Misleh, and W. W. Li, “MEME: discovering and analyzing DNA and protein sequence motifs,” *Nucleic Acids Res*, vol. 34, no. Web Server issue, pp. W369–W373, Jul. 2006.

[34] E. Portales-Casamar, S. Thongjuea, A. T. Kwon, D. Arenillas, X. Zhao, E. Valen, D. Yusuf, B. Lenhard, W. W. Wasserman, and A. Sandelin, “JASPAR 2010: the greatly expanded open-access database of transcription factor binding profiles,” *Nucleic Acids Res*, vol. 38, no. Database issue, pp. D105–110, Jan. 2010.

[35] V. Matys, E. Fricke, R. Geffers, E. Gößling, M. Haubrock, R. Hehl, K. Hornischer, D. Karas, A. E. Kel, O. V. Kel-Margoulis, D.-U. Kloos, S. Land, B. Lewicki-Potapov, H. Michael, R. Münch, I. Reuter, S. Rotert, H. Saxel, M. Scheer, S. Thiele, and E. Wingender, “TRANSFAC®: transcriptional regulation, from patterns to profiles,” *Nucleic Acids Research*, vol. 31, no. 1, pp. 374 –378, Jan. 2003.

[36] W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and and D. Haussler, “The Human Genome Browser at UCSC,” *Genome Research*, vol. 12, no. 6, pp. 996 –1006, Jun. 2002.

[37] O. Elemento, M. A. Rubin, and D. S. Rickman, “Oncogenic transcription factors as master regulators of chromatin topology: a new role for ERG in prostate cancer,” *Cell Cycle*, vol. 11, no. 18, pp. 3380–3383, Sep. 2012.

[38] S. Stolzenburg, M. G. Rots, A. S. Beltran, A. G. Rivenbark, X. Yuan, H. Qian, B. D. Strahl, and P. Blancafort, “Targeted silencing of the oncogenic transcription factor SOX2 in breast cancer,” *Nucl. Acids Res.*, vol. 40, no. 14, pp. 6725–6740, Aug. 2012.

[39] C. C. Wolford, S. J. McConoughey, S. P. Jalgaonkar, M. Leon, A. S. Merchant, J. L. Dominick, X. Yin, Y. Chang, E. J. Zmuda, S. A. O’Toole, E. K. A. Millar, S. L. Roller, C. L. Shapiro, M. C. Ostrowski, R. L. Sutherland, and T. Hai, “Transcription factor ATF3 links host adaptive response to breast cancer metastasis,” *Journal of Clinical Investigation*, vol. 123, no. 7, pp. 2893–2906, Jun. 2013.

[40] A. T. Look, “Oncogenic transcription factors in the human acute leukemias,” *Science*, vol. 278, no. 5340, pp. 1059–1064, Nov. 1997.

[41] S. Strano, S. Dell’Orso, S. Di Agostino, G. Fontemaggi, A. Sacchi, and G. Blandino, “Mutant p53: an oncogenic transcription factor,” *Oncogene*, vol. 26, no. 15, pp. 2212–2219, 2007.

[42] R. C. Lee, R. L. Feinbaum, and V. Ambros, “The C. elegans heterochronic gene lin-4 encodes small RNAs with antisense complementarity to lin-14,” *Cell*, vol. 75, no. 5, pp. 843–854, Dec. 1993.

[43] Y. Lee, M. Kim, J. Han, K.-H. Yeom, S. Lee, S. H. Baek, and V. N. Kim, “MicroRNA genes are transcribed by RNA polymerase II,” *EMBO J*, vol. 23, no. 20, pp. 4051–4060, Oct. 2004.

[44] G. M. Borchert, W. Lanier, and B. L. Davidson, “RNA polymerase III transcribes human microRNAs,” *Nat Struct Mol Biol*, vol. 13, no. 12, pp. 1097–1101, Dec. 2006.

[45] D. P. Bartel, “MicroRNAs: genomics, biogenesis, mechanism, and function,” *Cell*, vol. 116, no. 2, pp. 281–297, Jan. 2004.

[46] S. Griffiths-Jones, H. K. Saini, S. van Dongen, and A. J. Enright, “miRBase: tools for microRNA genomics,” *Nucleic Acids Res*, vol. 36, no. Database issue, pp. D154–158, Jan. 2008.

[47] M. Rehmsmeier, P. Steffen, M. Hochsmann, and R. Giegerich, “Fast and effective prediction of microRNA/target duplexes,” *RNA*, vol. 10, no. 10, pp. 1507–1517, Oct. 2004.

[48] A. Krek, D. Grün, M. N. Poy, R. Wolf, L. Rosenberg, E. J. Epstein, P. MacMenamin, I. da Piedade, K. C. Gunsalus, M. Stoffel, and N. Rajewsky, “Combinatorial microRNA target predictions,” *Nat. Genet*, vol. 37, no. 5, pp. 495–500, May 2005.

[49] B. P. Lewis, C. B. Burge, and D. P. Bartel, “Conserved seed pairing, often flanked by adenosines, indicates that thousands of human genes are microRNA targets,” *Cell*, vol. 120, no. 1, pp. 15–20, Jan. 2005.

[50] N. Rajewsky, “microRNA target predictions in animals,” *Nat Genet*, vol. 38, pp. S8–S13, May 2006.

[51] W. Zhang, J. E. Dahlberg, and W. Tam, “MicroRNAs in Tumorigenesis,” *Am J Pathol*, vol. 171, no. 3, pp. 728–738, Sep. 2007.

[52] J. Lu, G. Getz, E. A. Miska, E. Alvarez-Saavedra, J. Lamb, D. Peck, A. Sweet-Cordero, B. L. Ebert, R. H. Mak, A. A. Ferrando, J. R. Downing, T. Jacks, H. R. Horvitz, and T. R.

Golub, “MicroRNA expression profiles classify human cancers,” *Nature*, vol. 435, no. 7043, pp. 834–838, Jun. 2005.

[53] R. B. Marimont and M. B. Shapiro, “Nearest Neighbour Searches and the Curse of Dimensionality,” *IMA J Appl Math*, vol. 24, no. 1, pp. 59–70, Aug. 1979.

[54] G. Hughes, “On the mean accuracy of statistical pattern recognizers,” *IEEE Transactions on Information Theory*, vol. 14, no. 1, pp. 55–63, 1968.

[55] M. E. Houle, H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek, “Can Shared-Neighbor Distances Defeat the Curse of Dimensionality?,” in *Scientific and Statistical Database Management*, M. Gertz and B. Ludäscher, Eds. Springer Berlin Heidelberg, 2010, pp. 482–500.

[56] R. Fisher, L. Pusztai, and C. Swanton, “Cancer heterogeneity: implications for targeted therapeutics,” *Br J Cancer*, vol. 108, no. 3, pp. 479–485, Feb. 2013.

[57] *New Perspectives in Partial Least Squares and Related Methods*. .

[58] L. J. Frey, S. R. Piccolo, and M. E. Edgerton, “Multiplicity: an organizing principle for cancers and somatic mutations,” *BMC Medical Genomics*, vol. 4, no. 1, p. 52, Jun. 2011.

[59] A. Statnikov and C. F. Aliferis, “Analysis and Computational Dissection of Molecular Signature Multiplicity,” *PLoS Comput Biol*, vol. 6, no. 5, p. e1000790, May 2010.

[60] X. Qiu, Y. Xiao, A. Gordon, and A. Yakovlev, “Assessing stability of gene selection in microarray data analysis,” *BMC Bioinformatics*, vol. 7, no. 1, p. 50, Feb. 2006.

[61] L. D. Wood, D. W. Parsons, S. Jones, J. Lin, T. Sjöblom, R. J. Leary, D. Shen, S. M. Boca, T. Barber, J. Ptak, N. Silliman, S. Szabo, Z. Dezso, V. Ustyanksky, T. Nikolskaya, Y. Nikolsky, R. Karchin, P. A. Wilson, J. S. Kaminker, Z. Zhang, R. Croshaw, J. Willis, D. Dawson, M. Shipitsin, J. K. V. Willson, S. Sukumar, K. Polyak, B. H. Park, C. L. Pethiyagoda, P. V. K. Pant, D. G. Ballinger, A. B. Sparks, J. Hartigan, D. R. Smith, E. Suh, N. Papadopoulos, P. Buckhaults, S. D. Markowitz, G. Parmigiani, K. W. Kinzler, V. E. Velculescu, and B. Vogelstein, “The Genomic Landscapes of Human Breast and Colorectal Cancers,” *Science*, vol. 318, no. 5853, pp. 1108–1113, Nov. 2007.

[62] W. C. Hahn and R. A. Weinberg, “Modelling the molecular circuitry of cancer,” *Nat. Rev. Cancer*, vol. 2, no. 5, pp. 331–341, May 2002.

[63] B. Vogelstein and K. W. Kinzler, “Cancer genes and the pathways they control,” *Nat. Med.*, vol. 10, no. 8, pp. 789–799, Aug. 2004.

[64] C. F. Aliferis, I. Tsamardinos, and A. Statnikov, “HITON: A Novel Markov Blanket Algorithm for Optimal Variable Selection,” *AMIA Annu Symp Proc*, vol. 2003, pp. 21–25, 2003.

[65] R. Butterworth, G. Piatetsky-Shapiro, and D. A. Simovici, “On Feature Selection through Clustering,” in *Data Mining, IEEE International Conference on*, Los Alamitos, CA, USA, 2005, pp. 581–584.

[66] T. Sørlie, C. M. Perou, R. Tibshirani, T. Aas, S. Geisler, H. Johnsen, T. Hastie, M. B. Eisen, M. van de Rijn, S. S. Jeffrey, T. Thorsen, H. Quist, J. C. Matese, P. O. Brown, D. Botstein, P. E. Lønning, and A. L. Børresen-Dale, “Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications,” *Proc. Natl. Acad. Sci. U.S.A.*, vol. 98, no. 19, pp. 10869–10874, Sep. 2001.

[67] A. Anagnostopoulos, A. Dasgupta, and R. Kumar, “Approximation algorithms for co-clustering,” in *Proceedings of the twenty-seventh ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, New York, NY, USA, 2008, pp. 201–210.

[68] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein, “Cluster analysis and display of genome-wide expression patterns,” *Proc Natl Acad Sci U S A*, vol. 95, no. 25, pp. 14863 – 14868, 1998.

[69] S. Tavazoie, J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church, “Systematic determination of genetic network architecture,” *Nat. Genet.*, vol. 22, no. 3, pp. 281–285, Jul. 1999.

[70] G. C. Tseng and W. H. Wong, “Tight clustering: a resampling-based approach for identifying stable and tight patterns in data,” *Biometrics*, vol. 61, no. 1, pp. 10–16, Mar. 2005.

[71] J. C. Bezdek, S. K. Pal, and IEEE Neural networks society, *Fuzzy models for pattern recognition : methods that search for structures in data*. Piscataway NJ: IEEE Press, 1992.

[72] B. J. Frey and D. Dueck, “Clustering by Passing Messages Between Data Points,” *Science*, vol. 315, no. 5814, pp. 972–976, Feb. 2007.

[73] R. Braun, G. Leibon, S. Pauls, and D. Rockmore, “Partition decoupling for multi-gene analysis of gene expression profiling data,” *BMC Bioinformatics*, vol. 12, no. 1, p. 497, Dec. 2011.

[74] M. Kanehisa and S. Goto, “KEGG: kyoto encyclopedia of genes and genomes,” *Nucleic Acids Res.*, vol. 28, no. 1, pp. 27–30, Jan. 2000.

[75] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher, “Comprehensive Identification of Cell Cycle–regulated Genes of the Yeast Saccharomyces cerevisiae by Microarray Hybridization,” *Mol. Biol. Cell*, vol. 9, no. 12, pp. 3273–3297, Dec. 1998.

[76] U. V. Luxburg, M. Belkin, O. Bousquet, and Pertinence, “A tutorial on spectral clustering,” *Stat. Comput*, 2007.

[77] M. Meila and J. Shi, “A Random Walks View of Spectral Segmentation,” 2001.

[78] C. Chennubhotla and A. D. Jepson, “Half-lives of eigenflows for spectral clustering,” in *In NIPS*, 2002, pp. 689–696.

[79] R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker, “Geometric diffusions as a tool for harmonic analysis and structure definition of data: Diffusion maps,” *PNAS*, vol. 102, no. 21, pp. 7426–7431, May 2005.

[80] A. Y. Ng, M. I. Jordan, and Y. Weiss, “On Spectral Clustering: Analysis and an algorithm,” in *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS*, 2001, pp. 849–856.

[81] J. Shi and J. Malik, “Normalized Cuts and Image Segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[82] F. R. Bach and M. I. Jordan, “Learning Spectral Clustering,” in *Advances in Neural Information Processing Systems 16*, 2003.

[83] D. A. Tolliver, “Graph partitioning by spectral rounding: Applications in image segmentation and clustering,” in *In CVPR*, 2006, pp. 1053–1060.

[84] Y. Song, W. Chen, H. Bai, C. Lin, and E. Y. Chang, *Parallel Spectral Clustering*. .

[85] P. Drineas and M. W. Mahoney, “On the Nystrom Method for Approximating a Gram Matrix for Improved Kernel-Based Learning,” *J. Mach. Learn. Res.*, vol. 6, pp. 2153–2175, Dec. 2005.

[86] C. Chennubhotla and A. D. Jepson, “Hierarchical eigensolver for transition matrices in spectral methods,” in *NIPS*, 2005, pp. 273–280.

[87] D. Yan, L. Huang, and M. I. Jordan, “Fast approximate spectral clustering,” in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, 2009, pp. 907–916.

[88] M. C. de Souto, I. G. Costa, D. S. de Araujo, T. B. Ludermir, and A. Schliep, “Clustering cancer gene expression data: a comparative study,” *BMC Bioinformatics*, vol. 9, no. 1, p. 497, Nov. 2008.

[89] T. Calinski and J. Harabasz, “A dendrite method for cluster analysis,” *Communications in Statistics - Theory and Methods*, vol. 3, no. 1, pp. 1–27, 1974.

[90] R. Tibshirani, G. Walther, and T. Hastie, “Estimating the number of clusters in a dataset via the Gap statistic,” vol. 63, pp. 411–423, 2000.

[91] A. Alexa, J. Rahnenführer, and T. Lengauer, “Improved scoring of functional groups from gene expression data by decorrelating GO graph structure,” *Bioinformatics*, vol. 22, no. 13, pp. 1600–1607, Jul. 2006.

[92] W. M. Rand, “Objective Criteria for the Evaluation of Clustering Methods,” *Journal of the American Statistical Association*, vol. 66, no. 336, pp. 846–850, Dec. 1971.

[93] X. Ji and W. Xu, “Document clustering with prior knowledge,” in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, New York, NY, USA, 2006, pp. 405–412.

[94] G. Wu, R. S. Fan, W. Li, T. C. Ko, and M. G. Brattain, “Modulation of cell cycle control by vitamin D3 and its analogue, EB1089, in human breast cancer cells,” *Oncogene*, vol. 15, no. 13, pp. 1555–1563, Sep. 1997.

[95] T. C. G. A. Network, “Comprehensive molecular portraits of human breast tumours,” *Nature*, 2012.

[96] Z. He and W. Yu, “Stable feature selection for biomarker discovery,” *Computational Biology and Chemistry*, vol. 34, no. 4, pp. 215–225, Aug. 2010.

[97] I. Tsamardinos, C. Aliferis, and A. Statnikov, “Time and Sample Efficient Discovery of Markov Blankets And Direct Causal Relations,” in *Proceedings of the 9th CAN SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003, pp. 673–678.

[98] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, “The max-min hill-climbing Bayesian network structure learning algorithm,” *Mach. Learn.*, vol. 65, no. 1, pp. 31–78, Oct. 2006.

[99] I. Guyon and A. Elisseeff, “An Introduction to Variable and Feature Selection,” *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Mar. 2003.

[100] R. J. Fox and M. W. Dimmic, “A two-sample Bayesian t-test for microarray data,” *BMC Bioinformatics*, vol. 7, p. 126, 2006.

[101] R. Kohavi and G. H. John, “Wrappers for feature subset selection,” *Artificial Intelligence*, vol. 97, no. 1–2, pp. 273–324, Dec. 1997.

[102] M. Xiong, X. Fang, and J. Zhao, “Biomarker identification by feature wrappers,” *Genome Res.*, vol. 11, no. 11, pp. 1878–1887, Nov. 2001.

[103] R. Díaz-Uriarte and S. A. de Andrés, “Gene selection and classification of microarray data using random forest,” *BMC Bioinformatics*, vol. 7, no. 1, p. 3, Jan. 2006.

[104] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, “Gene Selection for Cancer Classification using Support Vector Machines,” *Machine Learning*, vol. 46, no. 1–3, pp. 389–422, Jan. 2002.

[105] E. Tuv, A. Borisov, G. Runger, and K. Torkkola, “Feature Selection with Ensembles, Artificial Variables, and Redundancy Elimination,” *Journal of Machine Learning Research*, vol. 10, pp. 1341–1366, Jul. 2009.

[106] L. Yu, C. Ding, and S. Loscalzo, “Stable feature selection via dense feature groups,” in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, 2008, pp. 803–811.

[107] M. Yuan and Y. Lin, “Model selection and estimation in regression with grouped variables,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, Feb. 2006.

[108] L. Meier, S. Van De Geer, and P. Bühlmann, “The group lasso for logistic regression,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 70, no. 1, pp. 53–71, Feb. 2008.

[109] T. Hastie, R. Tibshirani, D. Botstein, and P. Brown, *Supervised Harvesting of Expression Trees*. 2000.

[110] W.-H. Au, K. C. C. Chan, A. K. C. Wong, and Y. Wang, “Attribute clustering for grouping, selection, and classification of gene expression data,” *IEEE/ACM Trans Comput Biol Bioinform*, vol. 2, no. 2, pp. 83–101, Jun. 2005.

[111] H. Steck and T. S. Jaakkola, “On the Dirichlet Prior and Bayesian Regularization,” Sep. 2002.

[112] R. Jörnsten and B. Yu, “Simultaneous gene clustering and subset selection for sample classification via MDL,” *Bioinformatics*, vol. 19, no. 9, pp. 1100–1109, Jun. 2003.

[113] C. A. Davis, F. Gerick, V. Hintermair, C. C. Friedel, K. Fundel, R. Küffner, and R. Zimmer, “Reliable gene signatures for microarray classification: assessment of stability and performance,” *Bioinformatics*, vol. 22, no. 19, pp. 2356–2363, Oct. 2006.

[114] A. Kalousis, J. Prados, and M. Hilario, “Stability of feature selection algorithms: a study on high-dimensional spaces,” *Knowl. Inf. Syst.*, vol. 12, no. 1, pp. 95–116, May 2007.

[115] R. E. Neapolitan, *Probabilistic reasoning in expert systems: theory and algorithms*. New York: Wiley, 1990.

[116] J. Pearl, *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

[117] J. Pearl, “Probabilistic reasoning in intelligent systems: networks of plausible inference,” 1988.

[118] I. Tsamardinos and C. Aliferis, “Towards Principled Feature Selection: Relevancy, Filters and Wrappers,” in *in Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, 2003.

[119] D. Heckerman, “A Tutorial on Learning with Bayesian Networks,” in *Innovations in Bayesian Networks*, P. D. E. Holmes and P. L. C. Jain, Eds. Springer Berlin Heidelberg, 2008, pp. 33–82.

[120] C. F. Aliferis, A. Statnikov, I. Tsamardinos, S. Mani, and X. D. Koutsoukos, “Local Causal and Markov Blanket Induction for Causal Discovery and Feature Selection for Classification Part II: Analysis and Extensions,” *J. Mach. Learn. Res.*, vol. 11, pp. 235–284, Mar. 2010.

[121] P. Spirtes, C. N. Glymour, and R. Scheines, *Causation, prediction, and search.* Cambridge, Mass.: MIT Press, 2000.

[122] J. Cheng, D. Bell, and W. Liu, *Learning Bayesian Networks from Data: An Efficient Approach Based on Information Theory*. 1997.

[123] Z. Li, P. Li, A. Krishnan, and J. Liu, “Large-scale dynamic gene regulatory network inference combining differential equation models with local dynamic Bayesian network analysis,” *Bioinformatics*, vol. 27, no. 19, pp. 2686–2691, Oct. 2011.

[124] G. Morota, B. d. Valente, G. j. m. Rosa, K. a. Weigel, and D. Gianola, "An assessment of linkage disequilibrium in Holstein cattle using a Bayesian network," *Journal of Animal Breeding and Genetics*, vol. 129, no. 6, pp. 474–487, 2012.

[125] I. Tsamardinos, C. Aliferis, A. Statnikov, and E. Statnikov, "Algorithms for Large Scale Markov Blanket Discovery," in *In The 16th International FLAIRS Conference, St*, 2003, pp. 376–380.

[126] D. Margaritis and S. Thrun, "Bayesian Network Induction via Local Neighborhoods," in *Advances in Neural Information Processing Systems 12*, pp. 505–511.

[127] D. Koller and M. Sahami, "Toward Optimal Feature Selection," 1996, pp. 284–292.

[128] P. Langfelder and S. Horvath, "Eigengene networks for studying the relationships between co-expression modules," *BMC Systems Biology*, vol. 1, no. 1, p. 54, Nov. 2007.

[129] J. Neyman and E. S. Pearson, "On the Problem of the Most Efficient Tests of Statistical Hypotheses," in *Breakthroughs in Statistics*, S. Kotz and N. L. Johnson, Eds. Springer New York, 1992, pp. 73–108.

[130] V. Lagani, G. Kortas, and I. Tsamardinos, "Biomarker signature identification in 'omics' data with multi-class outcome," *Comp & Struct Biotech*, vol. 6, no. 0, Jun. 2013.

[131] J. S. Long, *Regression Models for Categorical and Limited Dependent Variables*. SAGE, 1997.

[132] P. McCullagh and J. A. Nelder, *Generalized linear models*. London; New York: Chapman and Hall, 1989.

[133] V. Lagani and I. Tsamardinos, "Structure-Based Variable Selection for Survival Data," *Bioinformatics*, vol. 26, no. 15, pp. 1887–1894, Aug. 2010.

[134] A. P. Gasch and M. P. Eisen, *Exploring the Conditional Coregulation of Yeast Gene Expression Through Fuzzy K-Means Clustering*. 2002.

[135] R. Kashef and M. S. Kamel, "Efficient Bisecting k-Medoids and Its Application in Gene Expression Analysis," in *Image Analysis and Recognition*, A. Campilho and M. Kamel, Eds. Springer Berlin Heidelberg, 2008, pp. 423–434.

[136] K. Pearson, "{On lines and planes of closest fit to systems of points in space}," *Philosophical Magazine*, vol. 2, no. 6, pp. 559–572, 1901.

[137] C. Cortes and V. Vapnik, "Support-Vector Networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[138] T. Fawcett, *ROC Graphs: Notes and Practical Considerations for Data Mining Researchers*. 2003.

[139] B. Hanczar, J. Hua, C. Sima, J. Weinstein, M. Bittner, and E. R. Dougherty, "Small-sample precision of ROC-related estimates," *Bioinformatics*, vol. 26, no. 6, pp. 822–830, Mar. 2010.

[140] D. R. Cox, "Regression Models and Life-Tables," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 34, no. 2, pp. 187–220, Jan. 1972.

[141] P. J. Heagerty, T. Lumley, and M. S. Pepe, "Time-Dependent ROC Curves for Censored Survival Data and a Diagnostic Marker," *Biometrics*, vol. 56, no. 2, pp. 337–344, Jun. 2000.

[142] R. Dybowski, "Neural Computation in Medicine: Perspectives and Prospects," in *Proceedings of the ANNIMAB-1 Conference (Artificial Neural Networks in Medicine and Biology*, 2000, pp. 26–36.

[143] E. Graf, C. Schmoor, W. Sauerbrei, and M. Schumacher, "Assessment and comparison of prognostic classification schemes for survival data," *Statistics in Medicine*, vol. 18, no. 17–18, pp. 2529–2545, 1999.

[144] H. B. Mann and D. R. Whitney, “On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other,” *Ann. Math. Statist.*, vol. 18, no. 1, pp. 50–60, Mar. 1947.

[145] F. Wilcoxon, “Individual Comparisons by Ranking Methods,” *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, Dec. 1945.

[146] D. J. Rogers and T. T. Tanimoto, “A Computer Program for Classifying Plants,” *Science*, vol. 132, no. 3434, pp. 1115–1118, Oct. 1960.

[147] A. Gibbons, *Algorithmic graph theory*. Cambridge: Cambridge University Press, 1985.

[148] A. Rosenwald, G. Wright, W. C. Chan, J. M. Connors, E. Campo, R. I. Fisher, R. D. Gascoyne, H. K. Muller-Hermelink, E. B. Smeland, J. M. Giltnane, E. M. Hurt, H. Zhao, L. Averett, L. Yang, W. H. Wilson, E. S. Jaffe, R. Simon, R. D. Klausner, J. Powell, P. L. Duffey, D. L. Longo, T. C. Greiner, D. D. Weisenburger, W. G. Sanger, B. J. Dave, J. C. Lynch, J. Vose, J. O. Armitage, E. Montserrat, A. López-Guillermo, T. M. Grogan, T. P. Miller, M. LeBlanc, G. Ott, S. Kvaloy, J. Delabie, H. Holte, P. Krajci, T. Stokke, and L. M. Staudt, “The Use of Molecular Profiling to Predict Survival after Chemotherapy for Diffuse Large-B-Cell Lymphoma,” *New England Journal of Medicine*, vol. 346, no. 25, pp. 1937–1947, 2002.

[149] A. Rosenwald, G. Wright, A. Wiestner, W. C. Chan, J. M. Connors, E. Campo, R. D. Gascoyne, T. M. Grogan, H. K. Muller-Hermelink, E. B. Smeland, M. Chiorazzi, J. M. Giltnane, E. M. Hurt, H. Zhao, L. Averett, S. Henrickson, L. Yang, J. Powell, W. H. Wilson, E. S. Jaffe, R. Simon, R. D. Klausner, E. Montserrat, F. Bosch, T. C. Greiner, D. D. Weisenburger, W. G. Sanger, B. J. Dave, J. C. Lynch, J. Vose, J. O. Armitage, R. I. Fisher, T. P. Miller, M. LeBlanc, G. Ott, S. Kvaloy, H. Holte, J. Delabie, and L. M. Staudt, “The proliferation gene expression signature is a quantitative integrator of oncogenic events that predicts survival in mantle cell lymphoma,” *Cancer Cell*, vol. 3, no. 2, pp. 185–197, Feb. 2003.

[150] L. Bullinger, K. Döhner, E. Bair, S. Fröhling, R. F. Schlenk, R. Tibshirani, H. Döhner, and J. R. Pollack, “Use of gene-expression profiling to identify prognostic subclasses in adult acute myeloid leukemia,” *N. Engl. J. Med.*, vol. 350, no. 16, pp. 1605–1616, Apr. 2004.

[151] E. Bair and R. Tibshirani, “Semi-Supervised Methods to Predict Patient Survival from Gene Expression Data,” *PLoS Biol*, vol. 2, no. 4, Apr. 2004.

[152] L. B. Jilaveanu, F. Zhao, C. R. Zito, J. M. Kirkwood, K. L. Nathanson, K. D’Andrea, M. Wilson, D. L. Rimm, K. T. Flaherty, S. J. Lee, and H. M. Kluger, “Expression of Drug Targets in Patients Treated with Sorafenib, Carboplatin and Paclitaxel,” *PLoS ONE*, vol. 8, no. 8, p. e69748, Aug. 2013.

[153] G. M. Argast, C. H. Croy, K. L. Couts, Z. Zhang, E. Litman, D. C. Chan, and N. G. Ahn, “Plexin B1 is repressed by oncogenic B-Raf signaling and functions as a tumor suppressor in melanoma cells,” *Oncogene*, vol. 28, no. 30, pp. 2697–2709, Jul. 2009.

[154] T. Budden and N. Bowden, “The Role of Altered Nucleotide Excision Repair and UVB-Induced DNA Damage in Melanomagenesis,” *International Journal of Molecular Sciences*, vol. 14, no. 1, pp. 1132–1151, Jan. 2013.

[155] A. E. Karnoub and R. A. Weinberg, “Ras oncogenes: split personalities,” *Nat Rev Mol Cell Biol*, vol. 9, no. 7, pp. 517–531, Jul. 2008.

[156] B. Homet and A. Ribas, “New Drug Targets in Metastatic Melanoma,” *J. Pathol.*, Sep. 2013.

[157] A. M. Fernández-Peralta, N. Nejda, S. Oliart, V. Medina, M. M. Azcoita, and J. J. González-Aguilera, “Significance of mutations in TGFBR2 and BAX in neoplastic progression and patient outcome in sporadic colorectal tumors with high-frequency microsatellite instability,” *Cancer Genetics and Cytogenetics*, vol. 157, no. 1, pp. 18–24, Feb. 2005.

[158] M. G. Gartside, H. Chen, O. A. Ibrahimi, S. A. Byron, A. V. Curtis, C. L. Wellens, A. Bengston, L. M. Yudt, A. V. Eliseenkova, J. Ma, J. A. Curtin, P. Hyder, U. L. Harper, E. Riedesel, G. J. Mann, J. M. Trent, B. C. Bastian, P. S. Meltzer, M. Mohammadi, and P. M. Pollock, “Loss-of-function fibroblast growth factor receptor-2 mutations in melanoma,” *Mol. Cancer Res.*, vol. 7, no. 1, pp. 41–54, Jan. 2009.

[159] D. Becker, P. L. Lee, U. Rodeck, and M. Herlyn, “Inhibition of the fibroblast growth factor receptor 1 (FGFR-1) gene in human melanocytes and malignant melanomas leads to inhibition of proliferation and signs indicative of differentiation,” *Oncogene*, vol. 7, no. 11, pp. 2303–2313, Nov. 1992.

[160] L. Xerri, Z. Battyani, J. J. Grob, P. Parc, J. Hassoun, J. J. Bonerandi, and D. Birnbaum, “Expression of FGF1 and FGFR1 in human melanoma tissues,” *Melanoma Res.*, vol. 6, no. 3, pp. 223–230, Jun. 1996.

[161] Y. Wang and D. Becker, “Antisense targeting of basic fibroblast growth factor and fibroblast growth factor receptor-1 in human melanomas blocks intratumoral angiogenesis and tumor growth,” *Nat. Med.*, vol. 3, no. 8, pp. 887–893, Aug. 1997.

[162] M. Valesky, A. J. Spang, G. W. Fisher, D. L. Farkas, and D. Becker, “Noninvasive dynamic fluorescence imaging of human melanomas reveals that targeted inhibition of bFGF or FGFR-1 in melanoma cells blocks tumor growth by apoptosis.,” *Mol Med*, vol. 8, no. 2, pp. 103–112, Feb. 2002.

[163] K. V. Pandit, D. Corcoran, H. Yousef, M. Yarlagadda, A. Tzouvelekis, K. F. Gibson, K. Konishi, S. A. Yousem, M. Singh, D. Handley, T. Richards, M. Selman, S. C. Watkins, A. Pardo, A. Ben-Yehudah, D. Bouros, O. Eickelberg, P. Ray, P. V. Benos, and N. Kaminski, “Inhibition and role of let-7d in idiopathic pulmonary fibrosis,” *Am. J. Respir. Crit. Care Med*, vol. 182, no. 2, pp. 220–229, Jul. 2010.

[164] S. Nam, M. Li, K. Choi, C. Balch, S. Kim, and K. P. Nephew, “MicroRNA and mRNA integrated analysis (MMIA): a web tool for examining biological functions of microRNA expression,” *Nucleic Acids Res*, vol. 37, no. Web Server issue, pp. W356–362, Jul. 2009.

[165] G. Sales, A. Coppe, A. Bisognin, M. Biasiolo, S. Bortoluzzi, and C. Romualdi, “MAGIA, a web-based tool for miRNA and Genes Integrated Analysis,” *Nucleic Acids Res*, vol. 38, no. Web Server issue, pp. W352–359, Jul. 2010.

[166] J. C. Huang, Q. D. Morris, and B. J. Frey, “Bayesian Inference of MicroRNA Targets from Sequence and Expression Data,” *Journal of Computational Biology*, vol. 14, no. 5, pp. 550–563, Jun. 2007.

[167] R. Yamashita, H. Wakaguri, S. Sugano, Y. Suzuki, and K. Nakai, “DBTSS provides a tissue specific dynamic view of Transcription Start Sites,” *Nucleic Acids Res*, vol. 38, no. Database issue, pp. D98–104, Jan. 2010.

[168] C. D. Schmid, R. Perier, V. Praz, and P. Bucher, “EPD in its twentieth year: towards complete promoter coverage of selected model organisms,” *Nucleic Acids Res*, vol. 34, no. Database issue, pp. D82–85, Jan. 2006.

[169] X. Wang, Z. Xuan, X. Zhao, Y. Li, and M. Q. Zhang, “High-resolution human core-promoter prediction with CoreBoost_HM,” *Genome Res*, vol. 19, no. 2, pp. 266–275, Feb. 2009.

[170] A. Marson, S. S. Levine, M. F. Cole, G. M. Frampton, T. Brambrink, S. Johnstone, M. G. Guenther, W. K. Johnston, M. Wernig, J. Newman, J. M. Calabrese, L. M. Dennis, T. L. Volkert, S. Gupta, J. Love, N. Hannett, P. A. Sharp, D. P. Bartel, R. Jaenisch, and R. A. Young, “Connecting microRNA genes to the core transcriptional regulatory circuitry of embryonic stem cells,” *Cell*, vol. 134, no. 3, pp. 521–533, Aug. 2008.

[171] D. L. Corcoran, K. V. Pandit, B. Gordon, A. Bhattacharjee, N. Kaminski, and P. V. Benos, “Features of mammalian microRNA promoters emerge from polymerase II chromatin immunoprecipitation data,” *PLoS ONE*, vol. 4, no. 4, p. e5279, 2009.

[172] M. Kertesz, N. Iovino, U. Unnerstall, U. Gaul, and E. Segal, “The role of site accessibility in microRNA target recognition,” *Nat. Genet*, vol. 39, no. 10, pp. 1278–1284, Oct. 2007.

[173] R. C. Friedman, K. K.-H. Farh, C. B. Burge, and D. P. Bartel, “Most mammalian mRNAs are conserved targets of microRNAs,” *Genome Research*, vol. 19, no. 1, pp. 92 –105, Jan. 2009.

[174] J. Krüger and M. Rehmsmeier, “RNAhybrid: microRNA target prediction easy, fast and flexible,” *Nucleic Acids Res*, vol. 34, no. Web Server issue, pp. W451–454, Jul. 2006.

[175] G. L. Papadopoulos, M. Reczko, V. A. Simossis, P. Sethupathy, and A. G. Hatzigeorgiou, “The database of experimentally supported targets: a functional update of TarBase,” *Nucleic Acids Res*, vol. 37, no. Database issue, pp. D155–158, Jan. 2009.

[176] F. Xiao, Z. Zuo, G. Cai, S. Kang, X. Gao, and T. Li, “miRecords: an integrated resource for microRNA-target interactions,” *Nucleic Acids Res*, vol. 37, no. Database issue, pp. D105–110, Jan. 2009.

[177] P. Langfelder and S. Horvath, “WGCNA: an R package for weighted correlation network analysis,” *BMC Bioinformatics*, vol. 9, no. 1, p. 559, 2008.

[178] S. E. Fienberg, R. R. Sokal, F. J. Rohlf, F. J. Rohlf, and R. R. Sokal, “Biometry. The Principles and Practice of Statistics in Biological Research,” *Biometrics*, vol. 26, no. 2, p. 351, Jun. 1970.

[179] A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Favera, and A. Califano, “ARACNE: An Algorithm for the Reconstruction of Gene Regulatory Networks in a Mammalian Cellular Context,” *BMC Bioinformatics*, vol. 7, no. Suppl 1, p. S7, 2006.

[180] C. T. Lopes, M. Franz, F. Kazi, S. L. Donaldson, Q. Morris, and G. D. Bader, “Cytoscape Web: an interactive web-based network browser,” *Bioinformatics*, vol. 26, no. 18, pp. 2347–2348, Sep. 2010.

[181] D. Maglott, J. Ostell, K. D. Pruitt, and T. Tatusova, “Entrez Gene: gene-centered information at NCBI,” *Nucleic Acids Res*, vol. 39, no. Database issue, pp. D52–57, Jan. 2011.

[182] M. V. Plikus, Z. Zhang, and C.-M. Chuong, “PubFocus: semantic MEDLINE/PubMed citations analytics through integration of controlled biomedical dictionaries and ranking algorithm,” *BMC Bioinformatics*, vol. 7, p. 424, 2006.

[183] D. Rebholz-Schuhmann, H. Kirsch, M. Arregui, S. Gaudan, M. Riethoven, and P. Stoehr, “EBIMed--text crunching to gather facts for proteins from Medline,” *Bioinformatics*, vol. 23, no. 2, pp. e237–244, Jan. 2007.

[184] Q. Jiang, Y. Wang, Y. Hao, L. Juan, M. Teng, X. Zhang, M. Li, G. Wang, and Y. Liu, "miR2Disease: a manually curated database for microRNA deregulation in human disease," *Nucleic Acids Res*, vol. 37, no. Database issue, pp. D98–104, Jan. 2009.

[185] A. Lagana, S. Forte, A. Giudice, M. R. Arena, P. L. Puglisi, R. Giugno, A. Pulvirenti, D. Shasha, and A. Ferro, "miRo: a miRNA knowledge base," *Database*, vol. 2009, no. 0, pp. bap008–bap008, Aug. 2009.

[186] 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*, vol. 25, no. 1, pp. 25–29, May 2000.

[187] H. Dong, H. Siu, L. Luo, X. Fang, L. Jin, and M. Xiong, "Investigation gene and microRNA expression in glioblastoma," *BMC Genomics*, vol. 11 Suppl 3, p. S16, 2010.

[188] M. F. Corsten, R. Miranda, R. Kasmieh, A. M. Krichevsky, R. Weissleder, and K. Shah, "MicroRNA-21 Knockdown Disrupts Glioma Growth In vivo and Displays Synergistic Cytotoxicity with Neural Precursor Cell–Delivered S-TRAIL in Human Gliomas," *Cancer Research*, vol. 67, no. 19, pp. 8994 –9000, Oct. 2007.

[189] G. Gabriely, T. Wurdinger, S. Kesari, C. C. Esau, J. Burchard, P. S. Linsley, and A. M. Krichevsky, "MicroRNA 21 promotes glioma invasion by targeting matrix metalloproteinase regulators," *Mol. Cell. Biol*, vol. 28, no. 17, pp. 5369–5380, Sep. 2008.

[190] Y. Li, F. Guessous, Y. Zhang, C. Dipierro, B. Kefas, E. Johnson, L. Marcinkiewicz, J. Jiang, Y. Yang, T. D. Schmittgen, B. Lopes, D. Schiff, B. Purow, and R. Abounader, "MicroRNA-34a inhibits glioblastoma growth by targeting multiple oncogenes," *Cancer Res*, vol. 69, no. 19, pp. 7569–7576, Oct. 2009.

[191] F. B. Furnari, T. Fenton, R. M. Bachoo, A. Mukasa, J. M. Stommel, A. Stegh, W. C. Hahn, K. L. Ligon, D. N. Louis, C. Brennan, L. Chin, R. A. DePinho, and W. K. Cavenee, "Malignant astrocytic glioma: genetics, biology, and paths to treatment," *Genes & Development*, vol. 21, no. 21, pp. 2683 –2710, Nov. 2007.

[192] U. Brandes and T. Erlebach, *Network Analysis: Methodological Foundations*. Springer, 2005.

[193] J. A. Chan, A. M. Krichevsky, and K. S. Kosik, "MicroRNA-21 is an antiapoptotic factor in human glioblastoma cells," *Cancer Res*, vol. 65, no. 14, pp. 6029–6033, Jul. 2005.

[194] X. Fang, J.-G. Yoon, L. Li, W. Yu, J. Shao, D. Hua, S. Zheng, L. Hood, D. R. Goodlett, G. Foltz, and B. Lin, "The SOX2 response program in glioblastoma multiforme: an integrated ChIP-seq, expression microarray, and microRNA analysis," *BMC Genomics*, vol. 12, p. 11, 2011.

[195] T. Papagiannakopoulos, A. Shapiro, and K. S. Kosik, "MicroRNA-21 targets a network of key tumor-suppressive pathways in glioblastoma cells," *Cancer Res*, vol. 68, no. 19, pp. 8164–8172, Oct. 2008.

[196] Y. Chen, W. Liu, T. Chao, Y. Zhang, X. Yan, Y. Gong, B. Qiang, J. Yuan, M. Sun, and X. Peng, "MicroRNA-21 down-regulates the expression of tumor suppressor PDCD4 in human glioblastoma cell T98G," *Cancer Lett*, vol. 272, no. 2, pp. 197–205, Dec. 2008.

[197] R. T. Marquez, E. Wendlandt, C. S. Galle, K. Keck, and A. P. McCaffrey, "MicroRNA-21 is upregulated during the proliferative phase of liver regeneration, targets Pellino-1, and inhibits NF-kappaB signaling," *Am. J. Physiol. Gastrointest. Liver Physiol*, vol. 298, no. 4, pp. G535–541, Apr. 2010.

[198] P. Wang, F. Zou, X. Zhang, H. Li, A. Dulak, R. J. Tomko, J. S. Lazo, Z. Wang, L. Zhang, and J. Yu, “microRNA-21 negatively regulates Cdc25A and cell cycle progression in colon cancer cells,” *Cancer Res*, vol. 69, no. 20, pp. 8157–8165, Oct. 2009.

[199] J. Silber, D. A. Lim, C. Petritsch, A. I. Persson, A. K. Maunakea, M. Yu, S. R. Vandenberg, D. G. Ginzinger, C. D. James, J. F. Costello, G. Bergers, W. A. Weiss, A. Alvarez-Buylla, and J. G. Hodgson, “miR-124 and miR-137 inhibit proliferation of glioblastoma multiforme cells and induce differentiation of brain tumor stem cells,” *BMC Med*, vol. 6, p. 14, 2008.