2018

DISCOVERING DRIVER MUTATIONS IN BIOLOGICAL DATA

Yahya Bokhari

DISCOVERING DRIVER MUTATIONS IN BIOLOGICAL DATA

A Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University.

by

YAHYA BOKHARI

Master of Science in Bioinformatics from Virginia Commonwealth University, 2013

Bachelor of Science in Medical Technology Sciences from King Abdulaziz University, 2004

Director: Dr. Tomasz Arodz,

Associate Professor, Department of Computer Science

Virginia Commonwealth University

Richmond, Virginia

November, 2018

Acknowledgements

First, I would like to thank God Almighty for everything including the success finishing my Ph.D. dissertation. Then, I would like to express my deepest gratitude to my advisor, Dr. Tomasz Arodz for his valuable support, continuous guidance and help throughout this research project and thesis writing. I'm really fortunate to have such a great, kind available and caring advisor. I also want to extend my thanks to the committee members, Dr. J. Paul Brooks, Dr. Maria C Rivera, Dr. Bridget McInnes, and Dr. Alberto Cano. Finally, my greatest appreciation and thanks go to my family for their unlimited love, supplications, and support throughout my life.

TABLE OF CONTENTS

List of Algorithms

LIST OF TABLES

LIST OF FIGURES

Abstract

DISCOVERING DRIVER MUTATIONS IN BIOLOGICAL DATA

By Yahya Bokhari

A Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy at Virginia Commonwealth University.

Virginia Commonwealth University, 2018.

Director: Dr. Tomasz Arodz,

Associate Professor, Department of Computer Science

The genetic material we carry today is different from that we were born with since our cells are prone to mutations. Some mutations can make a cell divide without control, resulting in a growing tumor – these are called cancer driver mutations. Typically, in a cancer sample from a patient, a large number of mutations can be detected, and only a few of those are drivers, which contribute to cancer development. The majority are passenger mutations that either accumulated before the onset of the disease but did not cause it, or are byproducts of the genetic instability of cancer cells. One of the key questions in understanding the process of cancer development is which mutations are drivers, and should be analyzed as potential diagnostic markers or targets for therapeutics, and which are passengers that do not contribute to oncogenesis.

My research focuses on answering this question by analyzing mutation data from large groups of cancer patients and using optimization-based approaches to find sets of genes that fit the characteristic driver mutation pattern of high patient coverage but low excess coverage. My general approach is to improve state-of-the-art by focusing on two aspects of optimization-based driver detection: the form of the objective function, and the techniques for minimizing it.

In this work, I present two methods. The first method, QuaDMutEx, incorporates two novel elements: a new gene set penalty that includes non-linear penalization of excess mutations in a single patient, and a computationally efficient method for finding gene sets that minimize the penalty through a combination of stochastic search and exact binary quadratic programming. Compared to state-of-the-art methods, QuaDMutEx algorithm finds sets of putative driver genes that show higher coverage and lower excess coverage in datasets of mutations from brain, breast and colon tumors. The second method, QuaDMutNetEx is built on QuaDMutEx method, where I extended QuaDMutEx by incorporating biological networks as additional source of information to be taken into account when discovering driver mutations. QuaDMutNetEx has the ability to discover biologically connected set of driver genes that is also mutually exclusive.

CHAPTER 1

INTRODUCTION

1.1 Motivation

If we live long enough, we will eventually get cancer. That is because our cells are mutating from the time we are born. Fortunately, not all mutations cause cancer. Only a small fraction of mutations contribute to oncogenesis, that is, the development of cancer - these are called driver mutations. Genes in which driver mutations may occur are called driver genes. The rest of mutations are called passenger mutations and do not contribute to oncogenesis. Efforts with different approaches are ongoing to delay or to cure cancer. One important step to beat cancer is understanding its mechanism and discovering how it starts and progresses. Sequencing the genome of cancer samples and discovering which of the mutations that are present in it are driver mutations is a crucial way of doing that.

The Cancer Genome Atlas (TCGA) is an example of a cancer genome sequencing project. Previous analyses of the data TCGA generated indicate that driver mutations are scarce. Specifically, depending on the tumor type, a patient can have 10 to more than 100 passenger mutations in comparison to 2 to 6 driver mutations. The scarcity of driver mutation among passenger mutations makes it difficult to determine which mutations are drivers, and multiple approaches are being proposed to offer solution to this problem. Driver discovery approaches include statistical models, models that analyze functional impact of mutations, and approaches based on mutual exclusivity pattern observed in driver mutations. In my work, I adopted the mutual exclusivity approach. One reason of choosing mutual exclusivity approach is that the other approaches still suffer from lack of biological information that is necessary to achieve

reasonable accuracy.

Mutual exclusivity describes the combinatorial pattern of a minimal set of genes required for oncogenesis. In actual patient data, additional mutations in driver genes may occur, especially for slow growing tumors or in older patients. Also, some of mutations may be missed due to observation errors. Thus instead of detecting the presence or absence of mutual exclusivity in a set of genes, driver detection algorithms involve a score that penalizes deviations from a driver pattern, that is, a penalty for zero mutations in a patient, or for more than one mutation. Then, a search procedure is utilized to find a set of genes closest to mutual exclusivity pattern. It is an NP-hard problem, thus various approximation algorithms can be proposed.

1.2 Contributions of the Dissertation

My research focuses on improving cancer driver discovery by analyzing mutation data from large groups of cancer patients, and using optimization-based approaches to find sets of genes that fit the characteristic driver mutation pattern of high patient coverage but low excess coverage. My general approach is to improve state-of-the-art by focusing on two aspects of optimization-based driver detection: the form of the objective function, and the techniques for minimizing it.

In this work, we present two methods. The first method is QuaDMutEx (Quadratic Driver Mutation Explorer), a method that incorporates two novel elements. First, instead of a linear penalty for excess coverage used in tools like Dendrix, QuaDMutEx, uses a quadratic penalty that provides a more realistic penalty for sets with an excessive number of mutations. Second, QuaDMutEx uses a combination of optimal search through a series of subproblems, allowing for more effective and efficient search through the space of possible driver genes. In addition, the method allows for user-specified trade-offs between increasing coverage and decreasing excess coverage allowing for tailoring the method to fast- or slow-evolving tumors. In the second method, QuaDMut-

NetEx(Quadratic Driver Mutation Network Explorer), we improved the first method by using preexisting knowledge about which genes interact together in biological processes, namely, biological networks of protein-protein interaction. QuaDMutNetEx is built on top of QuaDMutEx to discover a biologically connected set of driver genes that are mutually exclusive.

For QuaDMutEx, I evaluated the method on two sources of data. The first source is The Cancer Genome Atlas, using Glioblastoma Mutiforme (GBM), Breast Invasive Carcinoma (BRCA) and Colon Adenocarcinoma (COAD) cancers. The second source of data is from DriverNet method[1], we used four genomic-transcriptomic datasets: triple negative breast cancer (eTNB), glioblastoma multiforme (eGBM), high-grade serous ovarian cancer (eHGS), and METABRIC breast cancer (eMTB) datasets.

In the other method, QuaDMutNetEx, we also used two sources of data. the first one is from DriverNet, namely, eTNB, eGBM, eHGS and eMTB. The other source of data is a big datset, Pan12, from HotNet2 [2]. Pan12 data set consists of a pool of 12 cancer type. In both data sources we used human protein-protein interactions networks used in HotNet2, namely, iRefIndex, HINT+HI2012 and MultiNet.

QuaDMutEx shows higher coverage and higher mutual exclusivity than the current state-of-the-art tool Dendrix. QuaDMutNetEx results in high-quality gene sets that are known to be biologically connected and have the features of high coverage and high exclusivity.

1.3 Structure of the Proposal

The rest of the proposal is organized as follows. First, in Chapter 2, I review relevant facts about cancer and experimental methods for studying it. I also present the state-of-the-art in methods for discovering cancer driver mutations. Chapter 3 presents the algorithmic and mathematical background behind my proposed methods, which are introduced in Chapter 4 and 5. In section 4.3, I present results of using QuaDMutEx on

different data from TCGA and DriverNet. The same chapters compare QuaDMutEX to established state-of-the-art methods. In Chapter 5, I present the other method, QuaDMutNetEx. The discussion of QuaDMutNetEx results and the comparison with state-of-the-art methods, DriverNet and HotNet2, is in section 5.3.

CHAPTER 2

MUTATIONS: BIOLOGY AND BIOINFORMATICS

The study of genes and their variation in humans is important to understand a lot of diseases including heart diseases, diabetes, hereditary diseases and cancer. There are multiple ways to study human genetics. One way is through sequencing of the human genome. This approach has been boosted by the introduction of next-generation sequencing (NGS) techniques, also known as high-throughput or second-generation sequencing. The cost and time required for genome data generation is significantly reduced compared to first generation sequencing, also known as Sanger sequencing. In fact, issues relating to time and cost of sequencing have been replaced with computational and big data analysis problems. NGS data generated in a week are equivalent to entire genome centers few years ago. Specifically, a single sequencer can output 40 Gb per day as opposed to 10 Mb per day in earlier sequencers. This huge growth translates into the need to process terabytes of data [3]. It also brings the opportunity to better understand how variations in sequence may impact human health and disease.

2.1 Human Genome Variation

One of the main reasons for genome sequencing is to uncover variation in the human genome. All human body cells descend from the mitotic cell division of the first fertilized egg. Dividing cells accumulate a number of alterations in DNA bases compared to that first fertilized egg. Most DNA damages are repaired except for a small fraction that may be converted into fixed mutations. Depending on the type of cell the mutations occur, we differentiate two types of mutations - somatic and germline [4]. Germline mutations are those occurring in sperm and egg cells - these will be carried on by the parents to the next generation. Effects of germline mutations, accumulated over generations, are

partially responsible for the diversity of the human population. Mutations in all other cells are called somatic mutations - they will not be passed on to offspring, but may impact the person in which they occur - many will have no detectable impact, but some may lead to cancer. Projects detecting those mutations are using NGS as a method of choice. One limitation of NGS technologies is errors in detecting or reading nucleotides or bases. Hence, given detected mutation in a base discovered by those technologies needs further analysis and invistigation to decide if the mutation is somatic, germline or a false positive. Additionally sometimes these errors are missed and classified as a mutation.

Changes in genetic sequence can be classified into three categories: 1) single nucleotide polymorphisms (SNPs), where a single DNA base is replaced by another; 2) INDELs (INsertion/DELetion), where between 1 to 10,000 base pairs (bps) are inserted or deleted from the genome 3) structural variation, where more than 10,000 bps are inserted or deleted from the genome [5]. Often, SNPs and INDELs involving just one base pair are called point mutations. These can occur by base substitutions and base additions or deletions, and may have various impact on the phenotype or protein functions. One simple way to categorize the impact of a point mutation is to look at its effect on the translated protein sequence of amino acids. Four scenarios are possible: a silent substitution, missense mutation, nonsense mutation, and a frameshift mutation. Silent substitution results in a mutation in the DNA sequence with no changes in the amino acid, hence, no change in the protein function. A missense mutation changes the amino acid sequence at a single point, that is, results in amino acid substitution. Nonsense mutations result in a codon that terminates translation, and thus a missense mutation has a more global effect, leading to a truncated protein. Another type of mutation with a non-localized effect is frameshift mutation, in which an insertion or a deletion of a single nucleotide results in change in all the amino acids between the place of the mutation and the end of the protein [6].

2.2 Cancer Genome

One of the potential consequences of changes and mutations in the genome is cancer. Cancer is a complex and heterogeneous disease to which all body organs and tissues are susceptible. The term cancer is given to any disease in which abnormal cells divide indefinitely and have the ability to invade other tissues. The American Cancer Society estimates the diagnosis of 1,735,350 new cases of cancer in the U.S. and cancer will account for 609,640 deaths in 2018 [7].

Formation of cancer (carcinogenesis) typically involves the following chain of mutations that deregulates cellular proliferation: m1: inactivation of a tumor suppressor gene results in cell proliferation; m2: mutation(s) that inactivates a DNA repair pathway; m3: Generation of an *oncogene* as a result of a mutation in a proto-oncogene; and m4: mutation(s) that inactivates additional tumor suppressor genes resulting in cancerous proliferation [8]. Proto-oncogenes are normal genes that control cellular proliferation. A mutation in a proto-oncogene might trigger uncontrolled cellular growth. In contrast, tumor suppressor genes inhibit tumor formation as the name suggests [9].

2.3 Driver and Passenger Mutations in Cancer

Knowing that cancer arises as an effect of some mutations leads to a question: which specific mutations contribute to cancer? Depending on their contribution to cancer development or lack of it, somatic mutations can be divided into two classes, namely, *driver* and *passenger* mutations [4]. Driver mutations, which happen in cancer genes, perturb normal cell control of proliferation, differentiation and death. Thus, driver mutations provide survival and growth advantage leading to clonal proliferation of these mutated cancerous cells [10]. Eventually, these cells may advance to surrounding tissue and metastasize. Proto-oncogenes and tumor suppressor genes, when mutated, are called driver mutations. In contrast, passenger mutations, which account for the majority of somatic mutations, do not confer a growth advantage. It is thought that

some of these mutations were already present in the ancestor of the cancer cell when it acquired any of its driver mutations [10]. Others arise from mutational exposures, genome instability or from increased cell division that gives rise to a clinically detectable cancer from a single transformed cell [11]. Cancer genomes can carry up to thousands of somatic substitutions including driver and passenger mutations [10]. Specifically, a tissue sample from a cancer patient has on average about 700 mutations in coding and non-coding regions of the genome, about 130 of them located in the coding region. Among these 130 mutations, typically only between 2 and 6 are driver mutations [12, 13]. Also, two patients with the same cancer type are likely to have completely different sets of driver mutations present. Thus, discovering driver mutations responsible for cancer is difficult due to the coexistence of large number of passenger mutations in the cancer genome, and high variability of the driver mutations among patients.

2.4 Computational Approaches for the Identification of Driver Genes

Passenger mutations accumulate from the time of egg fertilization to the existing cancer cell and do not play a role in cancer development. Subsequently, isolating driver mutations that are important for cancer growth from passenger mutations is often challenging. In this section we will discuss different approaches to discover driver mutations and we will provide details of some methods for illustration. In general, there are three main approaches to discover driver mutations. Namely, functional impact prediction, recurrent somatic mutation identification and discovery using biological pathways.

2.4.1 Mutation Functional Impact Prediction

This indirect approach applies discovered biological information into protein sequences of mutated genes to predict the functional impact of the mutation. If the predicted impact on protein function is negligible, the mutation is likely to be a passenger. If the impact is high, it may be a potential driver mutation, although in many

cases the function alternation will not be related to cancer, so it may still be a passenger mutation. Several prediction tools that use different known biological features are available, including SIFT [14, 15], CHASM [16], Polyphen2 [17] and MutationAssessor [18]. I present SIFT and Polyphen2 in the next two subsections.

2.4.1.1 SIFT

Conserved proteins are a family of important and essential proteins common between similar species. SIFT [14, 15] uses the known conserved proteins along with chemical features of amino acids to predict functional impact of mutations. The software assumes that the substitution of an amino acid with one having a reverse chemical feature may not be tolerated in converved proteins. For instance, if a protein family has hydrophobic or charged amino acid replaced by a hydrophilic or polar one, respectively, then SIFT will predict the mutation to have high impact. SIFT predicts the effect on the protein function by aligning the query protein sequence against related known protein sequences. Based on how amino acid substitutions at each aligned position in the related proteins are being tolerated, SIFT gives a probability of whether the substitution in the query sequence is tolerated, and if not, it predicts high functional impact of the mutation.

2.4.1.2 PolyPhen2

PolyPhen2 [17] is a functional impact prediction tool that uses Naive Bayes classifier on the distributions of eleven known sequences and structural features that contributes to protein function. Position-Specific Independent Count (PSIC) score is used in sequence feature calculations. PSIC captures the likelihood of an amino acid to be in a specific position of protein sequence relying on the distribution of amino acids in multiple sequence alignments. For structural features PolyPhen2 included the hydrophobic properties and whether the amino acid residue can access the surface area of

the protein. Once the sequence and structural features are calculated, Naive Bayes is used as a supervised classification method for discriminating tolerated and not tolerated mutations.

2.4.2 Recurrent Somatic Mutation Identification

One of the most intuitive direct approaches is to consider gene mutation frequency in observed sequenced cancer samples. In a dataset of cancer samples, driver genes are expected to accumulate a higher number of mutations than other genes - while mutations can be seen as a consequence of a random process that does not favor driver genes, the observed dataset is biased, because it only includes cancer samples, where mutations in some driver genes did occur in each patient. We can define a baseline mutation rate, which is simply the expected number of mutated bases per total bases in a given gene sequence. Features such as gene length, type of mutation, and DNA sequence context are the main features that impact the background mutation rate. Subsequently, if a gene exceeds the expected rate given these sequence features, it is labeled as frequently mutated, and is considered a driver gene. The background mutation frequency estimation has low accuracy due to the high variability of genome mutations. Thus, methods that use background mutation frequency tend to include other factors to improve the accuracy. For instance, MutSig used replication time of the DNA region and the gene expression level to gain more accuracy of prediction [19] [20]. We chose two classical tools to elaborate in explaining this approach, namely, MuSiC [21] and MutSigCV [22].

2.4.2.1 MuSiC

Significantly mutated genes are used to point out genes that show significantly higher mutation rate than the background mutation rate. MuSiC considers multiple mutational mechanisms, as well as gene location, to improve calculations of background

mutation rate. Background mutation rate of a tested sample or a group of similar samples compared with appropriate background mutation rate and p-value of each gene produced is then used to decide if a gene is a driver based on the p-value significance [21].

2.4.2.2 MutSigCV

MutSigCV estimates the background mutation rate with respect to each gene based on synonymous mutations within the genes and non-coding mutations in surrounding genes. Background mutation rate suffers from low accuracy, and to obtain more accurate estimation MutSigCV pools data from other genes having similar properties to the gene of interest. For instance, MutSigCV uses the properties of gene replication time and expression level as features to improve the estimation accuracy. Finally, p-values are calculated to determine if the observed mutation in a given gene is not likely random in comparison to the background model, hence, concluding that the gene is a putative driver gene in cancer development [22].

2.4.3 Pathway-centric Driver Mutation Discovery

Genes and proteins interact and impact each other through signaling processes, regulatory interactions, and metabolic reactions. Mutation in a driver gene located in an important pathway perturbs the pathway, and that perturbation might lead to cancer. Observations show that cancer development is likely to depend on pathway perturbation, which can result from mutation in any gene involved in the pathway, instead of depending on a mutation in a single specific gene. Knowing that oncogenesis most likely depends on pathways rather than on particular genes has shifted the direction towards discovering driver mutation members of pathways instead of individual genes [23][24].

In general, there are two sub-approaches with respect to biological pathways. The

first is based on using existing prior knowledge of biological pathways, including the networks of protein-protein interactions and transcription regulation, and pre-defined pathways [25, 26, 27, 28, 29]. Three examples of this approach, PathScan, HotNet and DriverNet are presented below.

The second approach does not use any pathway information, instead it relies on discovering members of an important cancer-driving pathways de novo from mutation data, based on a pattern of mutations that would be expected in genes from a single pathway [30, 31, 32]. This approach is the one being adopted in this work, so below we explain in detail how Dendrix, a current state-of-the-art method that uses the de novo approach, works.

2.4.3.1 PathScan

PathScan [24, 25] is an extension of an approach that relies on background mutation rate that assesses the increase in frequency not for a single gene, but for a whole user-supplied pathway. Using a whole pathway helps deal with overall low frequency of driver mutations, and their random distribution among genes in a pathway.

2.4.3.2 HotNet

HotNet [23, 26, 27] aims at detecting sub-networks of genes, where most of the genes have many mutations. The main problem in this approach is that biological networks have highly skewed degree distribution, with hub genes that are connected to many other genes, and may thus be chosen to be part of the sub-networks even if they are not driver genes. To deal with this problem, HotNet uses a heat or fluid diffusion model on biological network. HotNet method infuses heat to each gene in proportion to the frequency of mutation of the gene. The heat then diffuses through the edges of the network for a certain period of time. Low-degree nodes have limited number of neighbors to diffuse the heat to and will remain hot if they were highly mutated. On

the other hand, high-degree nodes diffuse the heat to a large number of neighbors, and thus will not be able to keep their heat. Observing two highly mutated genes connected by a single low-degree node is of great interest. It is less interesting to have a single high-degree node connecting several highly mutated genes. Afterward, the network will break into small sub-networks according to heat distribution. Sub-networks are then subjected to novel statistical tests that avoid multiple hypotheses testing for a huge number of networks, and assessing the possibility of having similar sub-networks by chance at the same time.

2.4.3.3 DriverNet

DriverNet discover drivers genes by evaluating their effect on gene expression. This algorithm uses a binary gene mutation data matrix, real-valued gene expression data and an influence graph. Influence graph is a mix of multiple preexisting biological knowledge, including protein-protein interaction, gene coexpression and others. Essentially, a given query gene consider driver if it is: a) mutated, b) the mutation affects the expected gene expression of some genes in multiple patients c) the over-expressed or under-expressed gene has to be connected to the considered query gene. The problem is formulated in a bipartite graph where the nodes on the left represent all mutated query genes and nodes on the right are multiple sets of expression data, where each set represent a patient. Nodes on the right are labeled if the expression pattern is abnormal. Edge is drawn if a gene in the left partition is known to have interaction with a gene in the right that is expressed abnormally. DriverNet tries to find genes in the left partition that are highly connected to the nodes in the right. DriverNet algorithm uses a greedy approximation algorithm to solve the optimization problem since it is similar to the minimum set cover problem, which is NP-hard.

2.4.3.4 Dendrix

In contrast to PathScan, HotNet and related approaches, de novo mutated pathway discovery operates using only mutation data, that is, a list of mutated genes in a set of samples from cancer patients, without any a priori knowledge of biological networks or pathways. The discovery is predicated on the assumptions that only a small number of driver mutations are present in any given tumor [33], and a set of specific pathways must be perturbed by driver mutations to cause cancer [9]. These two assumptions suggest that driver mutations are scarce, a tumor rarely has more than one driver gene mutated per pathway, and within a pathway different samples might have different driver genes mutated [23]. Indeed, observations suggest that in many types of tumors, only one mutation per pathway, or functionally related group of genes, is needed to drive oncogenesis. Thus, the minimal set of mutated genes required for cancer to develop would consists of several sets of genes, each corresponding to a crucial pathway such as angiogenesis. Within each gene set, in each patient exactly one gene would be mutated. That is, all patients would be covered by a mutation in a gene from the set, and there would be no coverage overlap, that is, no patient will have more mutations than one in the genes from the set [24]. This pattern has been often referred to as mutual exclusivity within a gene set.

The De novo Driver Exclusivity (Dendrix) [31] algorithm summarizes how well a potential set of genes conforms to the driver mutation pattern by introducing two quantities, total coverage and coverage overlap:

- Total coverage = number of patients covered by at least one gene from the given gene set
- Coverage overlap = total count of all mutations in genes from the set that are in excess of one mutation per patient

These two quantities give rise to a Dendrix score, defined as total coverage minus

coverage overlap. This score is the objective function that is maximized by Dendrix. Dendrix uses Markov Chain Monte Carlo approach to maximize it, given a requested size of the set of genes, typically a number between three and ten.

CHAPTER 3

TECHNIQUES FOR SOLVING HARD OPTIMIZATION PROBLEMS

As we have seen in Chapter 2.4.3.4, finding driver mutations using de novo pathway-centric approach involves solving an optimization problem. The approach proposed here also employs optimization techniques - specifically, techniques for solving binary quandratic programs. Thus, this chapter will provide background on quadratic optimization including quadratic programming, binary quadratic programming, and heuristic methods for solving it.

3.1 Optimization with Quadratic Programming

In general, an optimization problem is called quadratic programming if it has convex, quadratic objective function, possibly with a set of affine constraints. A quadratic program can be expressed in the following form:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \frac{1}{2}\mathbf{x}^T Q\mathbf{x} + \mathbf{c}^T\mathbf{x} + \alpha \\ \text{subject to} \quad & A\mathbf{x} \leq \mathbf{b}, \end{aligned} \tag{3.1}$$

where Q is an element of symmetric $n \times n$ positive semidefinite matrices set, or $Q \in S_+^n$. A on the other hand is an element of real $m \times n$ matrices set, or $A \in R^{m \times n}$. $\mathbf{c}$ is a vector of coefficients, and $\mathbf{x}$ is the unknown real-valued vector that we seek.

3.1.1 Unconstrained Binary Quadratic Programming

In many practical applications, including the one studied here, the vector x we seek has to be a binary vector, because each member of x has the option be in the solution set, giving rise to Binary Quadratic Programs. One subclass of these are problems without constraints: the Unconstrained Binary Quadratic Programs (UBQP). UBQP

has been used in several applications including graphs and clustering problems. A UBQP can be represented as:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & \frac{1}{2}\mathbf{x}^T Q\mathbf{x} + c^T\mathbf{x} + \alpha \\ \text{subject to} \quad & x_i \in \{0, 1\}, \end{aligned} \tag{3.2}$$

where $Q \in S_+^n$. UBQPs has been studied since early work by Hammer and Rudeanu in 1968 [34, 35]. Although UBQP looks simple, it is known to be NP-hard [34]. Nonetheless, for small problems, the global optimum can be obtain using currently available solvers such as Gurobi or CPLEX that utilize branch-and-bound approach [34]. For larger problems, including those from genomics where the number of variables can be in thousands, solving UBQP requires heuristic methods such as simulating annealing or genetic algorithms, that do not guarantee optimality of the returns solution. Certain class of UBQP problems have combinatorial equivalents, and in some cases solving the UBQP is more efficient than using combinatorial methods [34].

3.2 Heuristics for Solving Unconstrained Binary Quadratic Programs

3.2.1 Genetic Algorithms

Genetic algorithms (GAs) are programming approaches to mimic a simple version of biological evolution theory towards better fitness, for the purpose of solving scientific and engineering problems. The majority of GAs methods have at least four fundamental elements: dynamically changing chromosome population in which each chromosome encodes one solution to the original problem; crossover between chromosomes to construct new chromosomes; random mutation of new chromosomes; and selection of chromosomes based on fitness [36]. In the simplest setting, each chromosome is a binary string encoding a solution. Also, a fitness function is defined to quantify how good the solution encoded by a chromosome is. Then, GA incorporates three operations: 1) *Selection*: chromosomes with high fitness values will be selected to produce offspring ; 2)

Crossover, which produces offspring of two chromosomes by taking parts of chromosome from each parent and merging them to form a new chromosome; 3) *Mutation*, in which the offspring are subject to small random changes. For instance, a crossover between sequence 11110000 and 10101010 can result in two new chromosomes: 11111010 and 10100000. These can then by mutated by flipping a random bit, resulting in 11111110 and 10110000. We can escape local minima in GAs by using memetic algorithms [37]. memetic algorithms can escape local minima by accepting a less-fit neighbor.

3.2.2 Markov Chain Monte Carlo Algorithms

MCMC was ranked as one of the top 10 most important algorithm in the 20th century [38]. The goal of the algorithm is to sample from a given distribution $p^*(x)$ by constructing a Markov chain on the state space χ that has stationary distribution $p^*(x)$. By drawing enough $x_0, x_1, ..., x_n$, samples from the chain, we can reach the stationary distribution p^* [39].

3.2.2.1 Metropolis-Hastings Algorithm

Metropolis-Hastings algorithm (MH) [40, 41] is a known MCMC method that is used in multiple applications [42], including finding minima of a black-box function $f(x)$, where the state space is the domain of the function. The core concept of MH-based optimization is to design a chain in which points x with low value of the function correspond to states in the state space that have high probability in the stationary distribution. Transition from state x_i to state x_{i+1} is achieved by randomly sampling a new state, and going through accept-or-reject filtering based on the value of the function at both points. Particularly, we have certainty of accepting the new solution x_{i+1} if the value of the function decreases, and low but not null probability to acceptance if the value of the function increases, which helps to escape the local minima. After enough number of transitions MH converges into desired stationary distribution, hence, global

minima are more likely to be sampled[39]. Algorithm 1 illustrates the Metropolis-Hastings algorithm in more detail.

Algorithm 1 Metropolis-Hastings algorithm

1: Initialize with a solution x_1
2: **for** $n \leftarrow iteration\{1, ..., N\}$ **do**
3: $x_{cand} = RandomSolution(x_n)$;
4: $W_n = [f(x_n) - f(x_{cand})]$;
5: Compute $p = min[1, e^{W_n}]$;
6: Sample $u \sim U(0, 1)$;
7: **if** $u < p$ **then**
8: Set x_{n+1} as x_{cand}
9: **else**
10: Set x_{n+1} as x_n
11: **end if**
12: **end for**

Often, the best solution encountered during the iterations is stored, and returned instead of the last solution.

3.2.2.2 Simulated Annealing

Simulated annealing is used to find minimum of a black box function $f(x)$. It is directly related to Metropolis Hastings algorithm in terms of random sampling from unknown probability distribution. It simulates the effect of annealing, that is heating and cooling a material to achieve a minimum energy state. For optimization, the value of the function $f(x)$ takes the role of energy of the state x. At fixed temparature T, the algorithm behaves just like the MH algorithm, and x approches samples from a

stationary distribution with state probabilities:

$$p(x) \propto \exp(-f(x)/T). \tag{3.3}$$

The convergence to the stationary distribution is achieved through randomly sampling a new state based on the current one, and accepting it or rejecting depending on coefficient α

$$\alpha = \exp([f(x) - f(x')]/T). \tag{3.4}$$

As in Metropolis Hastings, we accept the new state x_{t+1} with probability $min(1, \alpha)$.

The chances of accepting an state that is farther from optimum than the current state depend on the value of the temperature. At high temperature, acceptance of solutions that are worse than the current one is more likely than at low temperatures. Simulated Annealing starts with high temprature, to allow for relatively free exploration of the state space. SA then reduces the temperature gradually, which shifts the behavior towards accepting better, or only slightly worse solutions, and favors local exploration. Algorithm 2 shows Simulated Annealing in detail.

Algorithm 2 Simulated Annealing

Initialize with a solution x_1

for $n \leftarrow iteration\{1, ..., N\}$ **do**

    $x_{cand} = RandomSolution(x_n)$;

    $W_n = [f(x_n) - f(x_{cand})]/T$;

    Compute $p = min[1, e^{W_n}]$;

    Sample $u \sim U(0, 1)$;

    if $u < p$ **then**

        Set x_{n+1} as x_{cand}

    else

        Set x_{n+1} as x_n

    end if

    Decrease temperature T

end for

CHAPTER 4

DETECTING DRIVER MUTATIONS USING BINARY QUADRATIC PROGRAMMING

4.1 Proposed Method

The proposed approach for discovering driver mutations is based on optimization techniques, and belongs to the category of Pathway-centric Driver Mutation Discovery methods that were summarized in Chapter 2. I focus on two aspects of optimization-based driver detection: the form of the objective function, and the techniques for minimizing it.

In this proposal, I present Quadratic Driver Mutations Explorer (QuaDMutEx) [43], a method that incorporates two novel elements: a new gene set penalty that includes non-linear penalization of excess mutations in a single patient, and a computationally efficient method for finding gene sets that minimize the penalty through a combination of stochastic search and exact binary quadratic programming. It also allows for adjusting the desired behavior through parameters that control the solution size, and the trade-off between coverage and mutual exclusivity.

The proposed algorithm for detecting driver mutations in cancer operates at the gene level. That is, on input, we are given an n by p mutation matrix G, where n is the number of cancer patients with sequenced cancer cell DNA, and p is the total number of genes explored. The matrix is binary, that is, $G_{ij} = 1$ if patient i has a non-silent mutation in gene j; otherwise, $G_{ij} = 0$. A row vector G_i represents a row of the matrix corresponding to patient i. The solution we seek is a sparse binary vector x of length p, with $x_j = 1$ indicating that mutations of gene j are cancer driver mutations. In the proposed approach, the solution vector should capture driver genes that are functionally

related, e.g. are all part of a pathway that needs to be mutated in oncogenesis. If we want to uncover all driver genes, we should apply the algorithm multiple times, each time removing the genes found in prior steps from consideration. We will often refer to the nonzero elements of x as the mutations present in x.

In designing the algorithm for choosing the solution vector x, we assume that any possible vector is penalized with a penalty score based on observed patterns of driver mutations in human cancers. We expect that each patient has at least one mutation in the set of genes selected in the solution; however, in some cases, the mutation may not be detected. Also, while several distinct pathways need to be mutated to result in a growing tumor, typically one mutation in each of those pathways suffices. The chances of accumulating additional mutations in the already mutated pathway are low and decrease with each additional mutation. Not all tumors grow at the same pace, slow-growing tumors have a higher chance of accumulating additional mutations. In those types of tumors, we needed to have some flexibility to reduce the overlap penalty to allow some overlaps. We introduce parameter k that capture this decreasing odds through a quadratic penalty associated with x given the observed mutations G_i in patient i

$$L(G_i, x) = \frac{1+k}{2}\left(G_i x - 1\right)\left(G_i x - \frac{2}{1+k}\right) \tag{4.1}$$

The term $G_i x$ captures the number of mutations from solution x present in patient i. The penalty is parameterized by a non-negative real number k to be chosen by the user. It captures the ratio of penalty for exactly two mutations from set x present in patient i to penalty for no mutation from set x present in patient i. We incur no penalty if the number of mutated genes from x in a given patient is one. The effect of k on the penalty can be seen in Figure 1. For example, for a tumor with strong mutator phenotype where more mutations are present one can set k to a low value, lowering the

penalty for multiple mutations in genes from set x present in a patient.

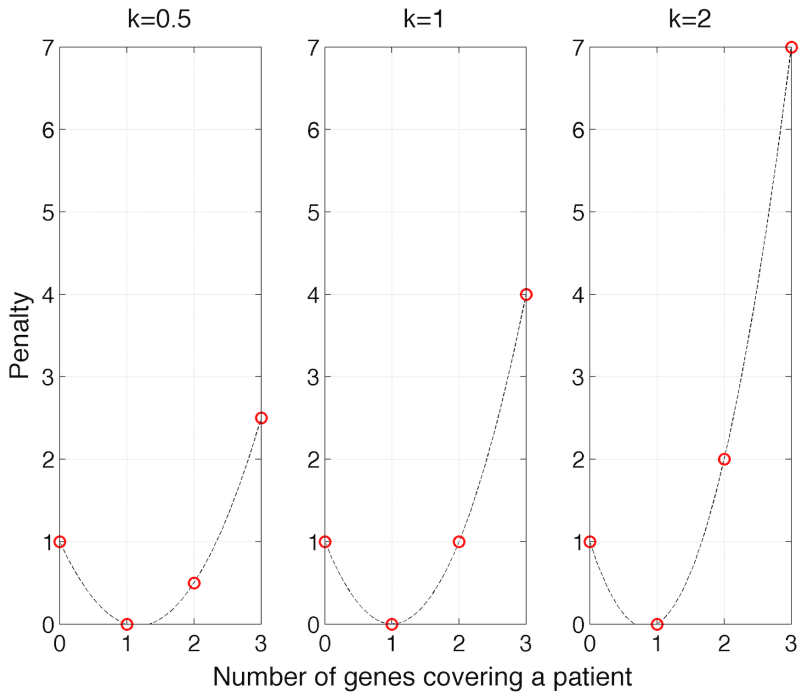


Figure 1: Effect of different values of k on penalty $L(G_i, x)$, in a function of $G_i x$, i.e., the number of mutations from solution x present in patient i. As k gets bigger, the resulting solution has more preference towards exclusivity

In addition, we expect the number of genes harboring driver mutations in a given pathway is small. Hence, we introduce a penalty on the number of genes selected in the solution, in a form of L_0 pseudo-norm, $L_0(x) = ||x||_0$. The effect of introducing the penalty can be seen in Figure 2.

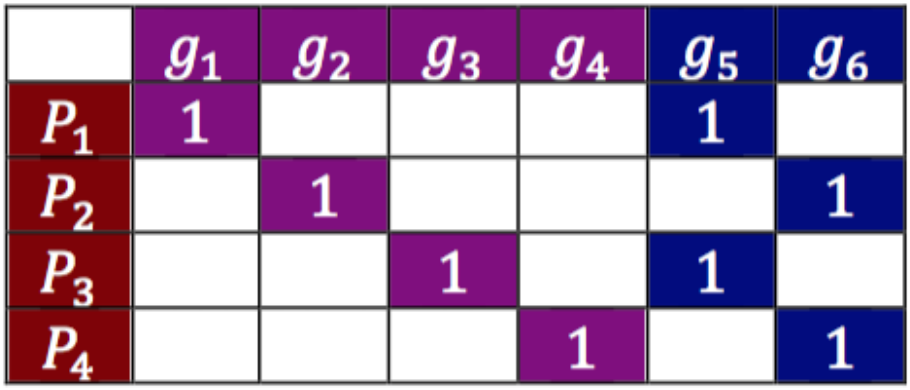


Figure 2: Illustration of the driver selection problem with six genes and four patients. Without the L_0 term, either violet $(g_1 - g_4)$ or blue (g_5, g_6) genes are both globally optimal solutions. Inclusion of L_0 pseudo-norm makes the blue solution a preferred one.

The total penalty for a possible solution vector x is a sum of per-patient penalties and the solution-size penalty:

$$L(G, x) = CL_0(x) + \sum_{i=1}^{n} L(G_i, x). \tag{4.2}$$

The parameter C controls the trade-off between minimization of $L(G_i, x)$ terms and of the L_0 pseudo-norm. It can alternatively be seen as the penalty incurred by increasing the number of genes in the solution x by one.

The result of the transformation is an unconstrained binary quadratic problem with the solution space involving binary vectors x of length p:

$$\begin{aligned}
&\underset{x}{\text{minimize}} && x^T Q x - f^T x && (4.3)\\
&\text{subject to} && 0 \le x \le 1 && (4.4)\\
& && x \in \mathbb{Z} && (4.5)\\
&\text{where} && Q = \frac{k+1}{2} G^T G && (4.6)\\
& && f = \frac{k+3}{2} G^T \mathbf{1}_n - C\mathbf{1}_p && (4.7)
\end{aligned}$$

where $\mathbf{1}_n$ represents a unit vector of length n. Binary quadratic problems are known to be NP-hard in general [34]. However, the optimal solution can be obtained quickly for problems of small size. Our approach in solving this problem involves a meta-heuristic based on Markov-Chain-Monte-Carlo search combined with optimal local search for small subproblems. The subproblems algorithm is presented below.

The main QuaDMutEx algorithm goes through T iterations, and in each considers a solution x containing up to ν genes. In each iteration, a new candidate solution is generated by randomly modifying the current solution vector. The new candidate solution is then modified by dropping some genes, based on exact binary quadratic optimization (eq. 4.3) involving ν genes present in the candidate solution. If the optimized solution is better than the solution from previous iteration, it is accepted. If not, it is accepted with probability depending on the difference in quality of the previous and the current solution. Throughout iterations, the solution x^* with the lowest value of the objective function (eq. 4.3) is kept.

Algorithm 3 QuaDMutEx

procedure QuaDMutEx($G, C, k, \nu, T, \Gamma, \sigma$)
    $x^0 = 0$
    $L^* = L^0 = \infty$
    for $t \leftarrow 1, ..., T$ **do**
        x=RandomGenerateNewSolution(x^{t-1}, ν, Γ)
        x, L=LocalOptimizeSolution(G, x, C, k)
        if $L < L^*$ **then**
            $L^* = L$
            $x^* = x$
        end if
        P=exp($-\frac{L-L^{t-1}}{\sigma}$)
        r=RandomUniform[0,1]
        if $r < P$ **then**
            $L^t = L$
            $x^t = x$
        else
            $L^t = L^{t-1}$
            $x^t = x^{t-1}$
        end if
    end for
    return x^*
end procedure

The random process generating a new candidate solution based on current solution always returns a solution with exactly ν genes. If the current solution already has ν genes, one of them will be randomly replaced with a gene not in the solution. The gene to be removed is chosen at random with uniform probability of $1/\nu$. The gene

to be added is chosen by random sampling from a distribution $\Gamma_{\sim x}$, which is defined through a user-supplied distribution Γ over all genes, modified to have 0 probability for the genes currently in solution x. If the current solution contains less then ν genes, the solution is expanded to include ν genes, and the $\nu - ||x||_0$ genes to be added are sampled without replacement according to $\Gamma_{\sim x}$. In our experiments, we used Γ proportional to the logarithm of the frequency of a mutation in a given gene among patients in the dataset.

Algorithm 4 QuaDMutEx: RandomGenerateNewSolution

```
1: procedure RandomGenerateNewSolution(x, ν, Γ)
2:     if ||x||_0 = ν then
3:         x=RandomReplaceOne(x, Γ_{∼x})
4:     else
5:         x=x + RandomSample(ν − ||x||_0, Γ_{∼x})
6:     end if
7:     return x
8: end procedure
```

The local search for an improved new solution returns an optimized solution x and its penalty score, L. It operates by limiting the problem to the ν genes present in the new candidate solution. That is, we create a n by ν submatrix G_x by choosing from G columns for which $x = 1$. Thus, we have an NP-hard binary QP problem with small number of variables, which can be solved using standard techniques. In our experiments, for datasets with below 500 patients, values of ν up to 50 lead to problems where global optimum could be reached in less than a second on a desktop workstation.

Algorithm 5 QuaDMutEx: LocalOptimizeSolution

procedure LocalOptimizeSolution(G, x, C, k)
    G_x = SubMatrix(G, x)
    x, L = BinaryQP(G_x, C, k)
    return x, L
end procedure

4.2 Method Complexity

Preparing the data to be solved by our algorithm is $O(n^3)$ that is because of the quadraic term G^TG needed in BQP. Inside the algorithm, generating a new random solution is $O(n)$. Solving the BQP locally is NP-hard but for small problems, the global optimum can be obtain using currently available solvers such as Gurobi.

4.3 Results and Discussion

4.3.1 Evaluation on Real Cancer Datasets

Table 1.: Summary of mutation-only datasets used in experimental validation of QuaDMutEx.

Dataset	samples (n)	genes (p)	mutations
GBM	84	178	809
OV	316	312	3004
LUNG	163	356	979
BRCA	771	13,582	33,385

We evaluated the proposed algorithm using four somatic mutation datasets (see Table 1), one from the Cancer Genome Atlas (TCGA) database and three from literature. Two datasets were originally used by the authors of Dendrix: somatic mutations in lung cancer (LUNG), and a dataset relating to Glioblastoma Multiforme (GBM)

that includes not only somatic mutations but also copy number alternations. The ovarian cancer dataset (OV) was originally used by the authors of TiMEx tool [44]. A larger dataset of mutations in samples from Breast Invasive Carcinoma (BRCA) was downloaded from TCGA. We have no missing data in any of the datasets. Following standard practice, in the BRCA dataset we removed known hypermutated genes that have no role in cancer [45], including olfactory receptors, mucins, and a few other genes such as titin. For each dataset, each gene in each patient was marked with one if it harbored one or more mutation, and with zero otherwise, resulting in the input matrix G for QuaDMutEx.

4.3.2 Quantitative Evaluation of QuaDMutEx Results

We ran QuaDMutEx on the four datasets: GBM, OV, LUNG, and BRCA. In the tests, we set the maximum size of the gene set to be $\nu = 30$. We set $k = 1$, indicating neutral stance with respect to the trade-off between coverage and excess coverage. The value of C, the weight of the gene solution size penalty, was set to 0.5 for GBM, the dataset with the smallest number of genes measured, to 1 for the LUNG and OV datasets which have twice the number of genes compared to GBM, and to 1.5 for BRCA, the dataset with much larger number of genes. We ran QuaDMutEx for $10,000$ iterations, which corresponds to running times below 10 minutes for each dataset. For GBM and BRCA, we also ran additional experiments with the default parameter values: $k = C = 1$.

Table 2.: Quantitative characteristics of QuaDMutEx results. For all four datasets, the solutions are statistically significant at $p < 0.05$.

Dataset	Parameters	Genes	Quadratic penalty	Estimated p-value
GBM	$k = 1, C = 0.5$	12	18	0.023
GBM	$k = C = 1$ (default)	7	20.5	0.001
OV	$k = C = 1$ (default)	3	17	0.010
LUNG	$k = C = 1$ (default)	15	59	0.036
BRCA	$k = 1, C = 1.5$	20	393	0.002
BRCA	$k = C = 1$ (default)	26	399	0.002

To assess statistical significance of the results returned by QuaDMutEx, we used permutation test proposed in [46]. In short, we randomly permuted the contents of each column of the input patient-gene matrix, which results in randomized dataset in which, for each gene, the number of patients harboring a mutation in the gene is preserved, but any pattern of mutation within a row, that is, within each single patient, is lost. We created 1000 randomized datasets and ran QuaDMutEx on each dataset. The value of the objective function observed on the original dataset was then compared with the distribution of objective function values on the randomized datasets to obtain a p-value estimate. The results of the tests, presented in Table 2, show that for all four datasets, QuaDMutEx returns gene sets that are statistically significant at 0.05.

The quadratic penalty provides a single-metric measure for what is essentially a two-criterion optimization problem involving simultaneous maximization of coverage and mutual exclusivity. To capture each of these independently, we used two metrics, coverage and excess coverage:

- $coverage = \frac{\text{number of patients covered by at least one gene from the set}}{\text{total number of patients}}$
- $excess\ coverage = \frac{\text{number of patients covered by more than one gene from the set}}{\text{number of patients covered by at least one gene from the set}}$

These metrics together capture how well a gene set conforms to the pattern expected of driver genes. Both of the metrics range from 0 to 1. A perfect pattern would have coverage of 1 and excess coverage of 0, indicating full mutual exclusivity.

4.3.3 Comparison with other Mutual-Exclusivity-based Methods

For comparison, we used RME [47], TiMEx [44], CoMEt [48] and Dendrix [46] as they are all from the de novo discovery family of methods [49] for driver detection, and all utilize only genetic data, same as QuaDMutEx. We ran the four tools on the same four datasets: GBM, OV, LUNG, and BRCA. For TiMEx, which does not require the user to specify the number of genes in the solution, we ran the tool with default parameters. Dendrix, RME and CoMEt require the user to provide the desired solution size. For Dendrix, we performed 29 runs for each dataset, with the solution size parameter ranging from 2 genes to 30 genes, and picked the solution size with the best Dendrix score. Each run involved 10^7 iterations. For CoMEt, the running time increases steeply with the requested solution size, thus we used sizes for which a single run finishes in less than 48 hours; in result, we tested solution sizes 2, 3, 4 ,5 for GBM and OV, between 2 and 6 for LUNG, and between 2 and 10 for BRCA. For RME, we used solution sizes between 2 and 5 genes, as recommended by the authors of the tool. For BRCA dataset, RME invoked with default parameters does not return any valid solution; to circumvent this problem, we executed RME for BRCA with the minimum gene frequency parameter lowered to 0.02 from the default value of 0.1. For the other three datasets, we used the default value.

We used the objective function maximized by Dendrix, which can be expressed by the notation introduced in the Methods section as $\textit{Dendrix score} = n - \sum_{i=1}^{n} |G_i x - 1|$, as the metric for evaluating the tool. Essentially, the Dendrix score equals to total coverage minus coverage overlap, where total coverage is the number of patients covered by at least one gene from the given gene set, and coverage overlap is total count of all

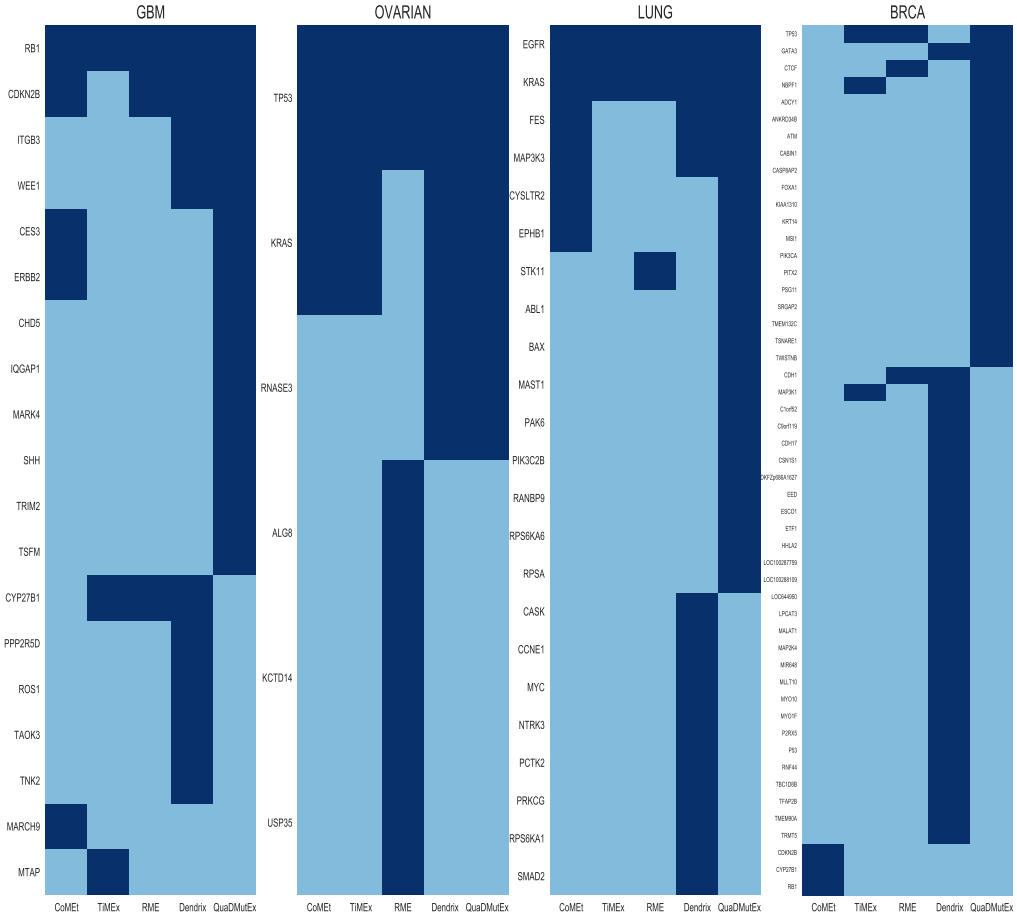
mutations in genes from the set that are in excess of one mutation per patient. High-quality solutions should have high Dendrix score.

The results of the tests, presented in Table 3, show that QuaDMutEx consistently returns higher quality solutions than all other methods. Only on the OV dataset, Dendrix discovers the same set of genes as QuaDMutEx. Remarkably, the quality of solutions from QuaDMutEx is higher even though the score used as the metric, the Dendrix score, is not function optimized by QuaDMutEx, but is the objective function of Dendrix. These results show that the proposed optimization scheme that combines stochastic heuristic approach with exact solution to a series of tractable subproblems is more efficient than the heuristic approach employed in Dendrix. The putative cancer driver gene sets discovered by QuaDMutEx are mostly different than sets returned by other tools (see Figure 3).

Table 3.: Comparison between QuaDMutEx and other methods. For QuaDMutEx, we used default parameter values $k = 1$ and $C = 1$ unless specified otherwise.

Method	Genes	Coverage	Excess coverage	Dendrix score
GBM: Glioblastoma multiforme				
TiMEx	3	0.7857	0.0606	62
RME	3	0.7857	0.0606	62
CoMEt	5	0.8452	0.0845	65
Dendrix	9	0.8571	0.0556	68
QuaDMutEx (C=0.5)	12	0.9286	0.0769	**72**
QuaDMutEx	7	0.8690	0.0822	67
OV: Ovarian Cancer				
TiMEx	2	0.9525	0	301
RME	5	0.9494	0.1	62
CoMEt	2	0.9525	0	301
Dendrix	3	0.9557	0	**302**
QuaDMutEx	3	0.9557	0	**302**
LUNG: Lung Adenocarcinoma				
TiMEx	2	0.5521	0	90
RME	3	0.6748	0.1273	96
CoMEt	6	0.6196	0	101
Dendrix	12	0.6809	0.0270	108
QuaDMutEx	15	0.8160	0.1053	**119**
BRCA: Breast Invasive Carcinoma				
TiMEx	3	0.4202	0.1006	289
RME	3	0.3865	0.0268	290
CoMEt	3	0.2620	0	202
Dendrix	29	0.5811	0.09598	402
QuaDMutEx (C=1.5)	20	0.6109	0.1338	408
QuaDMutEx	26	0.6342	0.1595	**411**

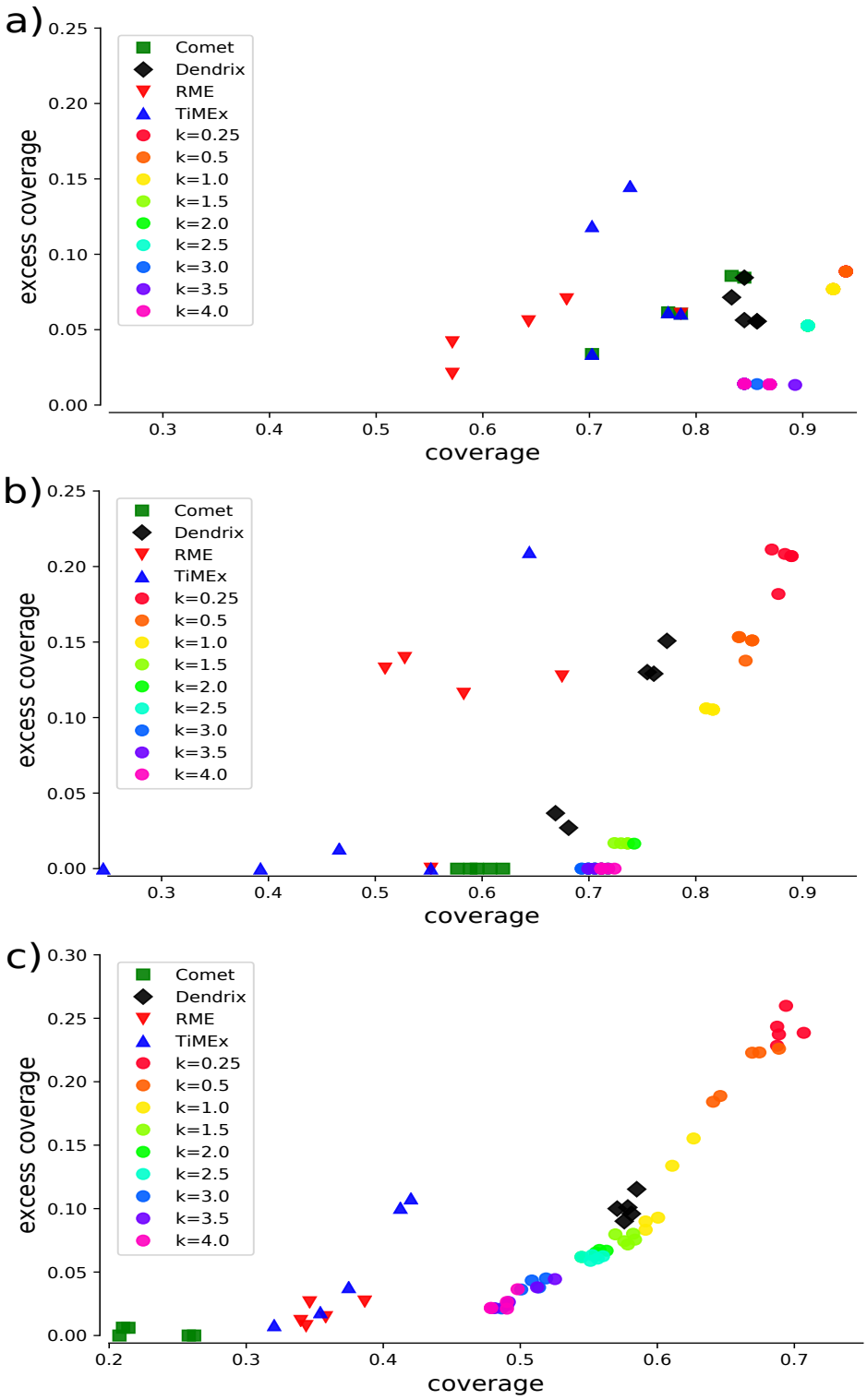
Figure 3: Comparison of putative cancer driver gene sets returned by QuaDMutEx and the other tools. Genes found by a tool are in dark blue.



We also checked how QuaDMutEx performs with respect to coverage and excess coverage, and compared the results with those of Dendrix, RME, TiMEx, and CoMEt. One of the features of QuaDMutEx is the flexibility in choosing the parameter k, which controls the trade-off between high coverage but higher excess coverage solutions and low excess coverage but lower coverage solutions. Thus, we ran QuaDMutEx with a range of values of parameter $k = 0.25, 0.5, 1, 1.5, 2, 2.5, 4$. As previously, the value of C

was set to 0.5 for GBM, to 1 for the LUNG and OV, and to 1.5 for BRCA. The number of iterations was again set to 10,000. For each parameter setting, we ran QuaDMutEx 5 times. We also gathered results from 5 runs of Dendrix for the best-performing solution size. For RME, TiMEx, and CoMEt the results do not vary from run to run, so we instead picked top five solution from a single run. Then, we quantified coverage and excess coverage. The results in Figure 4 show that QuaDMutEx solutions are on the Pareto-optimality frontier of all (RME, TiMEx, CoMEt, Dendrix and QuaDMutEx) solutions. For each Dendrix, TiMEx and CoMEt solution, there is a QuaDMutEx solution that is better: has higher coverage and lower excess coverage. These results further confirm results from Table 3 showing that the proposed tool improves upon the state-of-the-art. Data for OV are not shown graphically, as there is very little variability in solutions returned by the methods and the plot only confirms what is presented in Table 3.

Figure 4: Comparison of results from QuaDMutEx using different values of parameter k with results from other tools, in terms of coverage and excess coverage: a) GBM; b) LUNG; c) BRCA. In all three datasets, QuaDMutEx results are on the Pareto frontier.



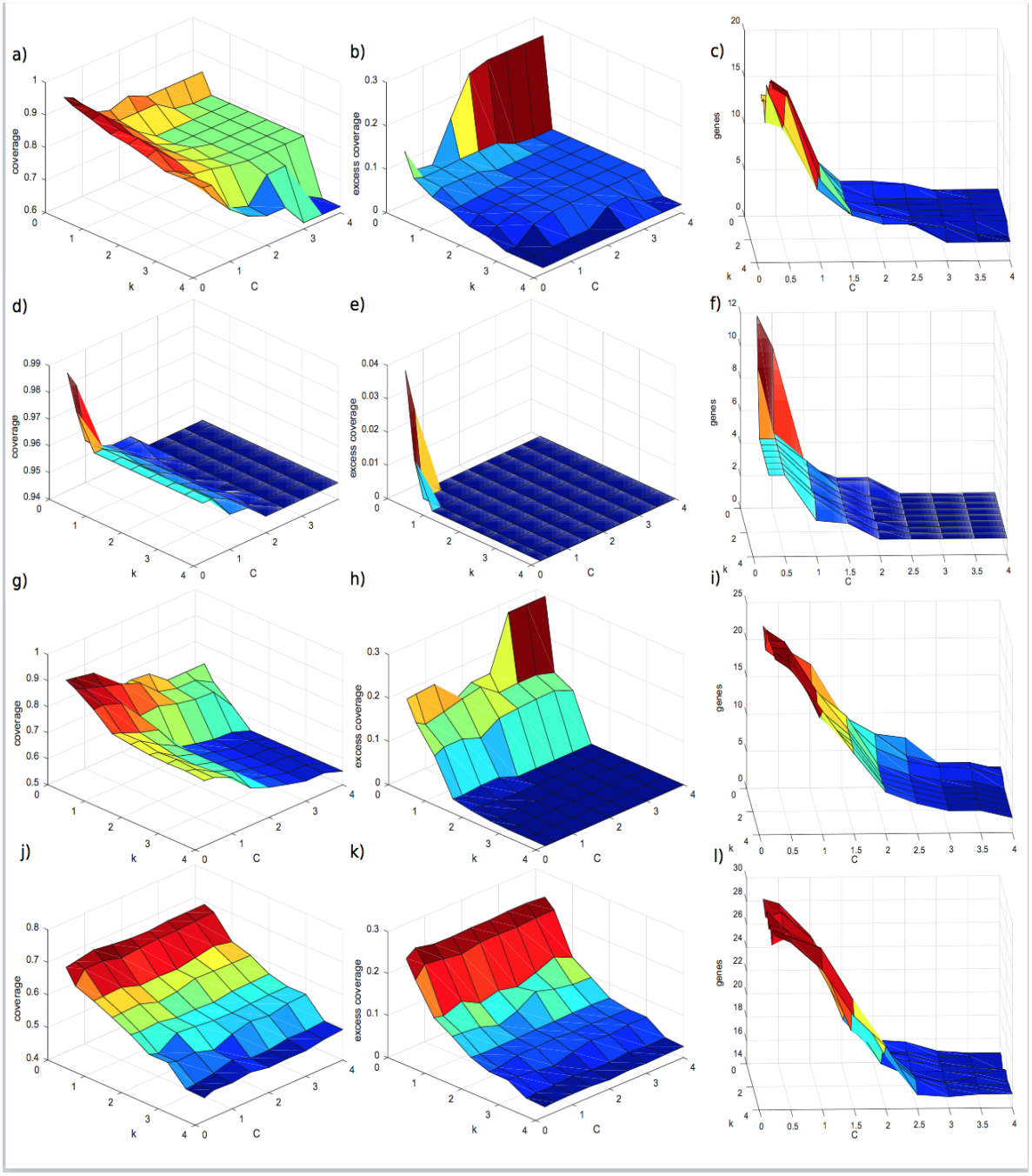
4.3.4 Effects of Parameters on QuaDMutEx

The proposed method allows for adjusting the penalty for expanding the solution size, through a parameter C that corresponds to the additional penalty for increasing the number of genes in the solution by one. It also allows for tweaking the trade-off between coverage and mutual exclusivity, through a parameter k that captures the ratio of penalty for one excess mutation in a patient to penalty for the patient not being covered by any mutation. We have analyzed how these two parameters affect the solution by running QuaDMutEx for 10,000 iterations for parameters $C = 0.25, 0.5, 1, 1.5, 2, 2.5, 4$ and $k = 0.25, 0.5, 1, 1.5, 2, 2.5, 4$.

Figure 5 shows that the parameter C achieves its design goal, that is, solutions with higher C include fewer genes. The figures also show that as the penalty for the size of the solution set is lowered, by specifying lower value of C, the coverage of patients by genes in the solution tends to increase for the three small datasets, where high values of C reduce the solution size to only a few genes and thus necessarily lower coverage. This effect is not present in the large dataset, BRCA, where C does not impact coverage. Changing C does not show any impact on excess coverage.

Changes in parameter k result in changes in coverage and excess coverage, but has no substantial impact on the number of genes in the solution. The results show that, as intended, lower values of k lead to higher coverage, at the cost of higher excess coverage, than high values of k. Thus, for slow growing tumors, tumors with elevated mutator phenotypes, or tumors in old patients, where many mutations may occur by chance and higher excess coverage is expected, low values of k is preferred over high k values.

Figure 5: Effects of parameters C and k on QuaDMutEx results, i.e., coverage (a,d,g,j), excess coverage (b,e,h,k), and genes in solution (c,f,i,l), for GBM dataset (a,b,c), OV dataset (d,e,f), LUNG dataset (g,h,i), and BRCA dataset (j,k,l).



4.3.5 Qualitative Assessment of QuaDMutEx Results

To validate the ability of QuaDMutEx to take only mutation data and discover rare putative cancer driver genes, which are the most hard to find using traditional methods that rely on mutation frequency in patient population, in each of the four datasets we focused on the genes in the solution with the fewest number of mutations. See Table 4 for a complete list of all genes in the solution, and for the number of mutations for each gene in each dataset. In addition to literature review, we also used DriverDBv2 [50], a database of previously discovered cancer driver genes, to further validate the quality of QuaDMutEx solutions. The solution set resulting from a bigger or new dataset will potentially have novel driver genes.

Table 4.: Putative driver gene sets discovered by QuaDMutEx. For each gene, in parentheses, we provide the number of patients in the dataset that harbored a mutation in that gene. Genes in bold are present int the DriverDBv2 [50] database of previously discovered cancer drivers.

Putative driver genes discovered by QuaDMutEx	Estimated p-value
GBM: Glioblastoma multiforme	
CDKN2B (43) **TSFM** (16) **RB1** (10) **ERBB2** (7)	
ITGB3 TRIM2 WEE1 CHD5 MARK4 CES3 SHH IQGAP1 (1)	0.023
OV: Ovarian cancer	
TP53 (299) **KRAS** (2) RNASE3 (1)	0.010
LUNG: Lung Adenocarcinoma	
KRAS (60) **STK11** (34) **EGFR** (30) **EPHB1** (4) **MAP3K3** (3)	
ABL1 PAK6 MAST1 CYSLTR2 RPS6KA6 FES (2)	
BAX PIK3C2B RANBP9 RPSA (1)	0.036
BRCA: Breast Invasive Carcinoma	
TP53 (194) **PIK3CA** (138) **GATA3** (80) NBPF1 (27)	
CTCF (18) **ATM** (16) **FOXA1** (15) **TMEM132C** (6)	
CABIN1 SRGAP2 KIAA1310 (5) **CASP8AP2 TSNARE1** (4)	
ADCY1 PITX2 PSG11 (3) **ANKRD34B KRT14 MSI1 TWISTNB** (2)	0.002

In the brain tumor dataset, eight identified genes are each mutated in only 1 out of 84 patients. Out of these, ITGB3 has known role in multiple cancers [51, 52], TRIM2 has tumor suppressing function in ovarian cancer [53] and plays a role in brain, the source of the analyzed tissue [54], WEE1 is already a target for cancer therapy [55], and CHD5 is a known tumor suppressor [56]. Changes in expression of MARK4 have been observed in glioblastomas [57]. While no cancer role has been so far identified

for carboxylesterase 3 (CES3), it is known to be expressed in the source tissue of our samples, the brain [58]. SHH gene has been linked to glioma growth [59], as well as to other cancers [60]. Finally, IQGAP1 is believed to play a role in cell proliferation and cancer transformation [61].

In the ovarian cancer dataset, KRAS, a known proto-oncogene, was found mutated in two patient. Eosinophil cationic protein (RNase 3) was found in only one patient. The protein, while not present in DriverDBv2 and not directly related to oncogenesis, has cytotoxic activity and was recently shown to inversely affect viability of cancer cell lines [62] and thus its mutations may affect human tumor growth.

In the QuaDMutEx solution for the lung datasets, six putative cancer driver genes are each mutated in only two of the 356 patients, and additional four are mutated in single patients. Among these, role of ABL1 in cancer is well established. PAK6 has been shown to be involved in prostate cancer [63], and presence of MAST1 mutations has been detected in lung samples [64]. The expression of CYSLTR2 gene is a prognostic marker in colon cancer [65]. RPS6KA2 gene is a putative tumor suppressor gene in ovarian cancer [66], and FES is a known proto-oncogene [67]. BAX is an oncoprotein with known role in cancers [68], including lung cancer [69]. Mutations in the PIK3C2B gene were previously observed in lung and other tumors [70, 71]. There is emerging evidence of a role of RANBP9 gene in lung cancer [72]. The 67-kDA laminin receptor gene RPSA, while not present in DriverDBv2, is known to play a role in tumor growth [73, 74].

Among the putative driver genes discovered by QuaDMutEx in the BRCA samples, nine were mutated in four or fewer of the 771 patients. Two among the genes that were mutated in more than four patients were not present in the DriverDBv2 database: NBPF1 and KIAA1310. However, NBPF1 has recently been identified as tumor suppressor gene [75]. KIAA1310 (KANSL3) is a member of KANSL family which plays a role in cell cycle and reduction of its function is associated with cancer [76]. Of the

rarely mutated genes, only TSNARE1 gene is likely to be a false positive. CASP8AP2 gene has been previously linked to cancer [77, 78]. No direct role in oncogenesis for ADCY1 gene has been reported, however it has been found downregulated in osteosarcomas [79]. PITX2 is a recurrence marker in breast cancer [80]. PSG11 gene has been shown to be correlated with survival in ovarian cancer [81]. Ankyrin repeat proteins, though not ANKRD34B specifically, have been previously reported as promoting cancer development [82]. KRT14 gene dysregulation was recently linked with breast cancer metastases [83]. MSI1 is putative therapeutic target in colon cancer [84]. TWISTNB is a component of the RNA polymerase I complex, and while TWISTNB gene has not been previously linked to cancer, mutations in polymerase subunits, cofactors, and mediators are known factors in malignancy [85]. Together, these results confirm that QuaDMutEx is effective in identifying cancer driver mutations even if they are rare in the analyzed patient group.

4.3.6 Comparison with Gene Expression-based Driver Discovery

In addition to methods that use only genomic mutation data, we also compared QuaDMutEx to DriverNet [1], a method that uses a biological network and gene expression data in addition to mutation data. We used four genomic-transcriptomic datasets that are provided with the DriverNet tool: triple negative breast cancer (eTNB), glioblastoma multiforme (eGBM), high-grade serous ovarian cancer (eHGS), and METABRIC breast cancer (eMTB) datasets. The summaries of the datasets are provided in Table 5.

DriverNet was executed using default parameters on the full information contained in the dataset, that is, the genomic, transcriptomic, and biological network information. The solution gene sets include all genes found by DriverNet to be statistically significant at the 0.05 p-value threshold. QuaDMutEx was executed using only the genomic data describing presence or absence of a mutation in a given gene in a given patient. We

Table 5.: Summary of genomic-transcriptomic datasets used in comparison with DriverNet.

Dataset	samples (n)	genes (p)	mutations
eTNB	94	4594	6007
eGBM	120	3747	8141
eHGS	316	13278	22897
eMTB	696	13076	51255

used the default value of $k = 1$, and set the value of C to 1.5, with the exception of the smallest dataset, eTNB, for which we used $C = 1$. We compared the putative cancer driver gene sets discovered by the two tools using coverage, excess coverage, and the Dendrix score, as described above.

For the eGBM dataset, QuaDMutEx shows much higher coverage and much lower excess coverage (see Table 6). For the other three datasets, QuaDMutEx shows much lower excess coverage than DriverNet, at the cost of a moderate decrease in coverage. These results reflect the fact that DriverNet is not designed to take mutual exclusivity of genes into consideration. On the other hand, DriverNet return many more genes than QuaDMutEx. A single run of QuaDMutEx is designed to return a single set of genes with low excess coverage, and does not include all putative driver genes - these can be detected with another run of QuaDMutEx.

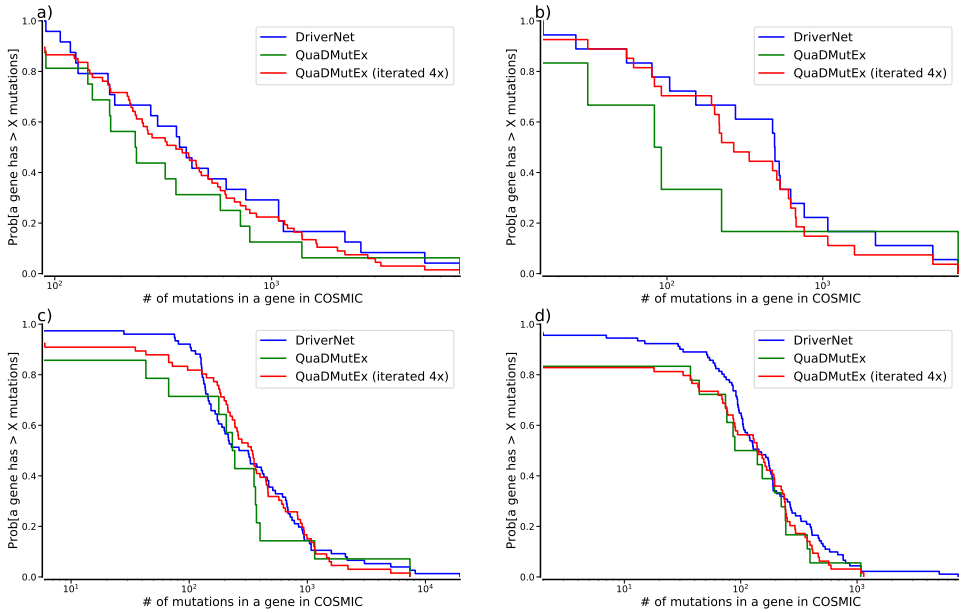
To provide a comparison that does not involve mutual exclusivity, we used the COSMIC database of mutations in cancer, and we introduced iterated QuaDMutEx, which increases the number of genes found by QuaDMutEx to the numbers similar to DriverNet. We performed four executions of QuaDMutEx, after each run removing the genes discovered so far from the dataset, so that they do not prevent discovery of additional genes that are not mutually exclusive with previously discovered ones. We then pooled the four high-exclusivity gene sets into a single high-coverage set.

Table 6.: Comparison between QuaDMutEx and DriverNet.

Method	Genes	Coverage	Excess coverage	Dendrix score
eTNB: Triple negative breast cancer				
DriverNet	64	0.6809	0.4688	18
QuaDMutEx (C=1)	16	0.8315	0.0270	**72**
eGBM: Glioblastoma multiforme				
DriverNet	19	0.9412	0.8839	-183
QuaDMutEx (C=1.5)	6	0.8067	0.0938	**87**
eHGS: high-grade serous ovarian cancer				
DriverNet	77	0.9430	0.6946	-110
QuaDMutEx (C=1.5)	14	0.8734	0	**276**
eMTB: METABRIC breast cancer				
DriverNet	92	0.4670	0.7785	-1151
QuaDMutEx (C=1.5)	18	0.4071	0.0876	**250**

Since mutual exclusivity can be expected only for a set of functionally-related genes, for example genes from a single cancer-related pathway, a single call to QuaDMutEx corresponds to a single-pathway query, and calling QuaDMutEx iteratively corresponds to a multi-pathway query, facilitating comparison with DriverNet which does not have a single-pathway focus.

Figure 6: Complementary cumulative distribution function plots for QuaDMutEx, iterated QuaDMutEx, and DriverNet, for eTNB (a), eGBM (b), eHGS (c), and eMTB (d) datasets.



To measure the quality of solutions returned by DriverNet and iterated QuaDMutEx in a way independent of any mutual exclusivity of gene mutations, we compared the numbers of COSMIC occurrences of mutations in genes returned by DriverNet with occurrence numbers for QuaDMutEx gene sets. Specifically, for each gene in a discovered gene set, we queried COSMIC for the number of observed mutations in that gene. We then plotted a complementary cumulative distribution function (CCDF) over the

numbers over the whole gene set. For example, for the eHGS dataset, for both QuaDMutEx and DriverNet, the CCDF value at 1,000 is approximately 0.14, indicating that for both methods, 14% of the genes in the solution set have more than 1,000 mutation each in COSMIC, while for 86% of genes in the solution set a COSMIC query for the gene results in at most 1,000 mutations. The results in Figure 6 indicate that iterated QuaDMutEx and DriverNet perform similarly on eTNB and eGBM datasets, and on eHGS and eMTB both perform similarly for majority of the mutation counts range, with DriverNet having an edge at the numbers below that threshold.

Genes returned by QuaDMutEx are to large extent different than those returned by DriverNet (see Figure 7), showing that the expression-based approach used in DriverNet and the mutation-only approach used in QuaDMutEx are complementary. We validated the genes discovered by QuaDMutEx (Table 7) in DriverDB2, a database of genes previously discovered as cancer drivers. For eTNB and eGBM datasets, all the genes discovered by QuaDMutEx are present in DriverDB2 database. In eHGS dataset, only ANKRD36B was not found in DriverDB2. However, ANKRD36B gene was identified in rare germline copy number variations in renal clear cell carcinoma [86], and also correlates with cellular sensitivity to chemotherapeutic agents [87]. In eMTB dataset, TRA@ gene is not present in DriverDB2, but it has been previously found to be linked to breast cancer [88]. TRA@ is also one of the genes that were discovered both by DriverNet and by QuaDMutEx. TBC1D3P2 is recurrently mutated in meningioma cell lines [89] and is a pseudogene for TBC1D3, a known oncogene [90]. There is no information available about AC116655.7-12 and AC116165.7-3, and at this point we classify both as false positives.

Figure 7: Comparison of putative cancer driver gene sets returned by QuaDMutEx, iterated QuaDMutEx, and DriverNet. Genes found by a tool are in dark blue.

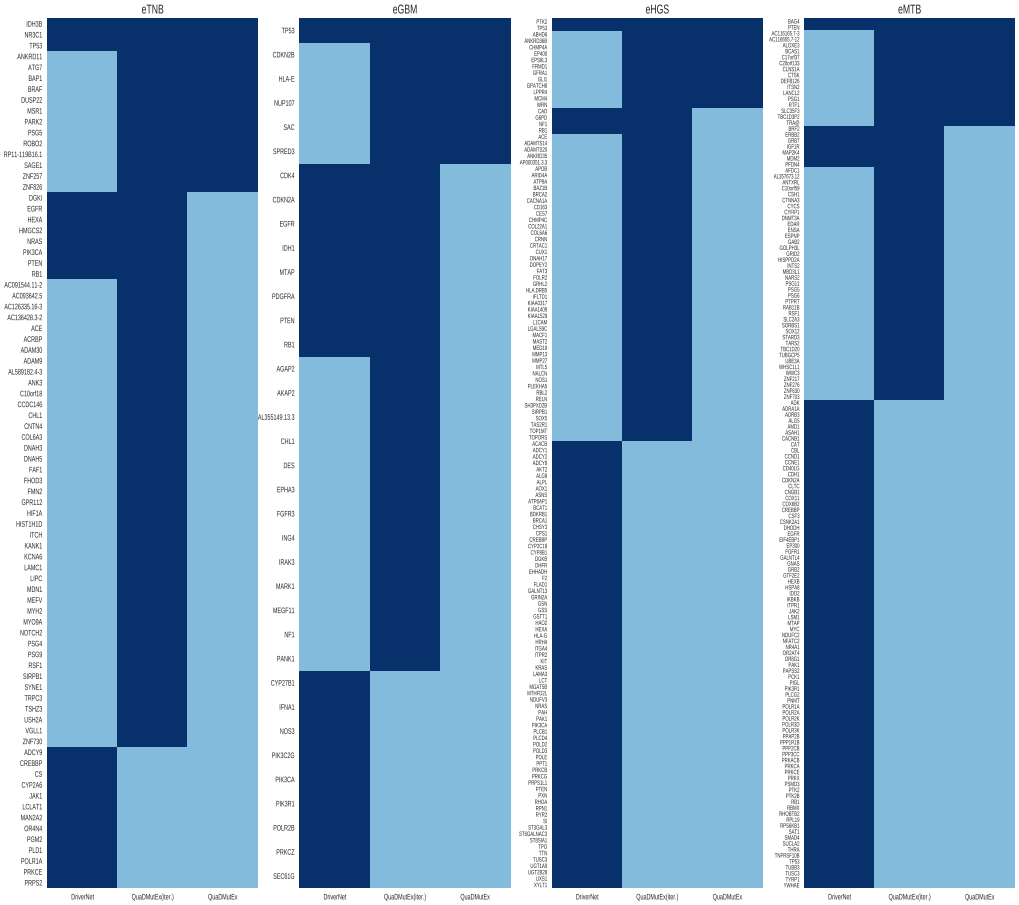


Table 7.: Putative driver gene sets discovered by QuaDMutEx. For each gene, in parentheses, we provide the number of patients in the dataset that harbored a mutation in that gene. Genes in bold are present in the DriverDBv2 [50] database of previously discovered cancer drivers.

Putative driver genes discovered by QuaDMutEx	Estimated p-value
eTNB: Triple negative breast cancer	
TP53 (35) **PARK2** (6) **ROBO2 DUSP22** (4)	
SAGE1 ANKRD11 NR3C1 (3) **BAP1 BRAF ATG7** (2)	
ZNF257 IDH3B ZNF826 RP11-119B16.1 PSG5 MSR1 (2)	0.001
eGBM: Glioblastoma multiforme	
CDKN2B (52) **TP53** (38) **NUP107** (9) **HLA-E SAC SPRED3**	0.001
eHGS: high-grade serous ovarian cancer	
TP53 (249) **GLI1** (3) **ABHD6 CHMP4A EP400 EPS8L3 FRMD1** (2)	
GPATCH8 MCM4 GFRA1 LPPR4 PTK2 WRN ANKRD36B (2)	0.001
eMTB: METABRIC breast cancer	
C17orf37(MIEN1) (82) **BAG4** (52) **CLNS1A** (37) **PSG1** (24)	
C20orf133(MACROD2) (19) **BCAS1** (17) **PTEN**(16) **RTF1 ALOXE3** (7)	
TRA@ AC116165.7-3 (6) TBC1D3P2 (5) **CTSK** AC116655.7-12 **LANCL2** (4)	
ITSN2 (3) **DEFB126** (3) **SLC35F3** (2)	0.001

4.3.7 Stability test

To evaluate the stability of our method to see how the resulted objective function values are close to each other, we repeated our method on GBM data for 100 times with $c = 1.5$ and $k = 1$ 10,000 iterations each. Figure 8 shows that the objective function values were stable and centered in the middle distribution with a mean equal to 146.1 and a standard deviation equal to 1.75. Since the objective function values were stable, the coverage and excess coverage were stable as well. Figure 8 also show a stable distribution for both coverage and excess coverage.

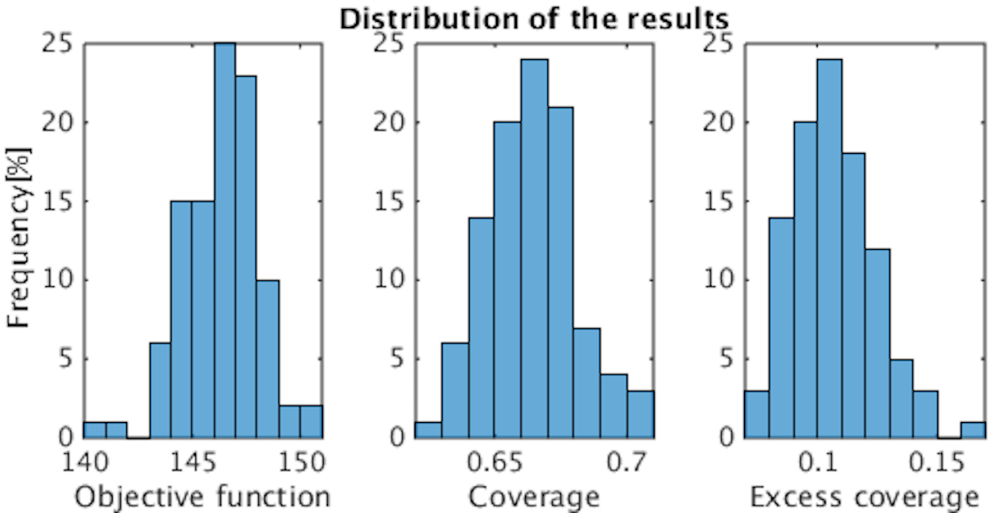


Figure 8: Shows the distribution of the percentage of the genes coverage.

4.4 Conclusions

Superior ability to improve on both coverage and excess coverage of the detected driver gene sets on datasets from different types of cancer shows that QuaDMutEx is a tool that should be part of a state-of-the-art toolbox in the driver gene discovery pipeline. It can help detect low-frequency driver genes that can be missed by existing methods.

CHAPTER 5

DETECTING DRIVER MUTATIONS USING BINARY QUADRATIC PROGRAMMING AND BIOLOGICAL NETWORKS

QuaDMutEx shows that an improved objective function and improved optimization method can both lead to improvement in detecting driver genes. However, it follows the de novo approach, which does not take advantage of existing prior knowledge. In this chapter, I propose a new method, QuaDMutNetEx, into which I integrated additional terms into the objective function that are present protein-protein interactions. Proteins perform almost all biological functions within the cell. Hence, using protein-protein interactions gives a comprehensive image of cellular processes, which lead to a better interpretation of any results in the context of the cellular system [91].

5.1 Integration of Biological Networks

Genes are elements of a complex system, a cell. One way to represent this complex system is through networks or graphs. These networks capture prior knowledge about interactions between genes, or their product, proteins. Since cancer is thought to be a disease of pathways, in which pathway functioning is perturbed by mutations, using information about edges forming those pathways in the algorithm will likely improve driver gene detection. If a gene in a given pathway is a driver, that would increase the chances that other genes in that pathway are also drivers

5.1.1 Biological Networks

Graphs are one way to represent complicated systems such as social, biological, and computer networks. Although the exact connectivity of biological networks is not fully known yet, their analysis and visualization are essential to understanding such a

complicated system. DNA, RNA, proteins and metabolites interact with each other and play different roles in biological systems. Therefore, different biological networks exist to illustrate different levels of biological interactions. For instance, *gene regulatory networks* explain regulation of protein production by gene activation and repression at any given time. On the other hand, *protein-protein interaction networks* show a more general view of which proteins make functional complexes with other proteins to facilitate biological processes including gene expression or cell growth. *Metabolic networks* are another type of biological networks that represent chemical substances transformation from one form to another [92].

In QuaDMutNetEx we used the three human protein-protein interactions networks previously used in HotNet2 [2]. The first network is the iRefIndex network, which consists of 91,872 interactions among 12,338 proteins. The second network is Multi-Net network which consists of 109,597 interactions among 14,445 proteins. The last network is HINT+HI2012 which is created by considering two interactome databases: HI-2012 prepublication data in human HI2 Interactome database (HI2012) and high-quality interactomes database (HINT). The HINT+HI2012 network consists of 40,783 interactions among 10,008 proteins.

5.2 Proposed Method

The aim of my second method is to include protein-protein interactions in the objective function. The external knowledge carried in the network helpes not only in improving the algorithm but also in the interpretability of the resulting gene set. The network can be introduced in various ways. In our method, we introduce the network as a term $N(A, x)$ that works as a reward term to the objective function if two genes in a solution are connected. Here, x is the gene solution set and A is the network, encoded

as an undirected adjacency matrix:

$$A_{ij} = \begin{cases} 1 & \text{if gene } i \text{ is the immediate neighbor of gene } j \text{ or vice versa,} \\ 0 & \text{otherwise.} \end{cases} \tag{5.1}$$

The additional term $N(A, x)$ will be defined as $-x^T Ax$. $N(A, x)$ can be expanded as $-\sum_{i,j} A_{ij} x_i x_j$, where x_i and x_j are binary. Thus, adding a negative term to the objective function corresponds to providing a reward every time two genes, x_i and x_j in the solution vector x are connected by an edge, that is, when $A_{ij} = 1$.

Frurther, we added parameter α to the term N to control the reward size. A high value of α results in a solution set with more connected genes. Therefor, the reward term N is now updated to be $-x^T \alpha Ax$.

The quadratic term $-x^T \alpha Ax$ can be integrated to the QuaDMutEx objective function (eq. 4.2) to have a new objective function as follow:

$$L(G, A, x) = \sum_{i=1}^{n} L(G_i, x) + CL_0(x) + N(A, x) \tag{5.2}$$

$$= \sum_{i=1}^{n} \frac{1+k}{2} (G_i x - 1) \left(G_i x - \frac{2}{1+k} \right) + C||x||_0 - \alpha x^T Ax. \tag{5.3}$$

Minimization of $L(G, A, x)$ can be viewed as an unconstrained binary quadratic problem (BQP) with the solution space involving binary vectors x of length p:

$$\underset{x}{\text{minimize}} \quad x^T Qx - f^T x \tag{5.4}$$

$$\text{subject to} \quad 0 \leq x \leq 1 \tag{5.5}$$

$$x \in \mathbb{Z} \tag{5.6}$$

$$\text{where} \quad Q = \frac{k+1}{2} G^T G - \alpha A \tag{5.7}$$

$$f = \frac{k+3}{2} G^T \mathbf{1}_n - C\mathbf{1}_p \tag{5.8}$$

BQPs are known to be NP-hard in general [93]. Except for the new parameters,

we used the same algorithm of QuaDMutex to solve this problem, i.e., a meta-heuristic based on Markov-Chain-Monte-Carlo search combined with optimal local search for small subproblems. The algorithm is presented below.

The main QuaDMutNetEx algorithm goes through T iterations, and in each considers a solution x containing up to ν genes. In each iteration, a new candidate solution is generated by randomly modifying the current solution vector. The new candidate solution is then modified by dropping some genes, based on exact binary quadratic optimization (eq. 5.2) involving ν genes present in the candidate solution. If the optimized solution is better than the solution from previous iteration, it is accepted. If not, it is accepted with probability depending on the difference in quality of the previous and the current solution. Throughout iterations, the solution x^* with the lowest value of the objective function (eq. 5.1) is kept.

The random process generating a new candidate solution based on current solution always returns a solution with exactly ν genes. If the current solution already has ν genes, one of them will be randomly replaced with a gene not in the solution. The gene to be removed is chosen at random with uniform probability of $1/\nu$. The gene to be added is chosen by random sampling from a distribution $\Gamma_{\sim x}$, which is defined through a user-supplied distribution Γ over all genes, modified to have 0 probability for the genes currently in solution x. If the current solution contains less than ν genes, the solution is expanded to include ν genes, and the $\nu - ||x||_0$ genes to be added are sampled without replacement according to $\Gamma_{\sim x}$. In our experiments, we used Γ proportional to the logarithm of the frequency of a mutation in a given gene among patients in the dataset.

The local search for an improved new solution returns an optimized solution x and its penalty score, L. It operates by limiting the problem to the ν genes present in the new candidate solution. That is, we create a n by ν submatrix G_x by choosing from G columns for which $x = 1$. Similarly, we create A_x from A by selecting rows

Algorithm QuaDMutNetEx

procedure QuaDMutNetEx($G, A, C, k, \alpha, \nu, T, \Gamma, \sigma$)
  $x^0 = 0$
  $L^* = L^0 = \infty$
  for $t \leftarrow 1, ..., T$ **do**
    x=RandomGenerateNewSolution(x^{t-1}, ν, Γ)
    x, L=LocalOptimizeSolution(G, A, x, C, k, α)
    if $L < L^*$ **then**
      $L^* = L$
      $x^* = x$
    end if
    P=exp($-\frac{L-L^{t-1}}{\sigma}$)
    r=RandomUniform[0,1]
    if $r < P$ **then**
      $L^t = L$
      $x^t = x$
    else
      $L^t = L^{t-1}$
      $x^t = x^{t-1}$
    end if
  end for
  return x^*
end procedure

Algorithm QuaDMutNetEx: RandomGenerateNewSolution

procedure RandomGenerateNewSolution(x, ν, Γ)
    if $||x||_0 = \nu$ **then**
        x=RandomReplaceOne($x, \Gamma_{\sim x}$)
    else
        x=x + RandomSample($\nu - ||x||_0, \Gamma_{\sim x}$)
    end if
    return x
end procedure

and columns with $x = 1$. Thus, we have an NP-hard binary QP problem with number of variables small enough that problem can be quickly solved to the optimum using standard techniques.

Algorithm QuaDMutNetEx: LocalOptimizeSolution

procedure LocalOptimizeSolution(G, A, x, C, k, α)
    G_x, A_x = SubMatrix(G, A, x)
    x, L = BinaryQP(G_x, A_x, C, k, α)
    return x, L
end procedure

In the proposed approach, the solution vector x from a single run of QuaDMutNetEx will capture a set of driver genes not only functionally related but also exhibit mutual exclusivity pattern, for example genes that are all part of a pathway that needs to be mutated in oncogenesis. To uncover a comprehensive set of driver genes for a specific cancer type, spanning multiple functional subsystems vital to oncogenesis, the algorithm should be applied multiple times, each time removing the genes found in prior runs from consideration.

5.3 Results and Discussion

5.3.1 Evaluation on Real Cancer Datasets

We evaluated the proposed algorithm using five new datasets. Four datasets are genomic-transcriptomic, that are provided with the DriverNet tool: triple negative breast cancer (eTNB), glioblastoma multiforme (eGBM), high-grade serous ovarian cancer (eHGS), and METABRIC breast cancer (eMTB) datasets. These data set have used in chapter 4 to evaluate QuaDMutEx. In the fifth dataset, we used Pan12 dataset from Hotnet2, which consists of 12 cancer types from The Cancer Genome Atlas (TCGA) [2]. We chose datasets from the tools that we wanted to compare our method with. That would obtain a better comparison as some discovered networks in those datasets are already evaluated by those tools. Also, DriverNet uses expression data, which we do not have in previous method dataset. The summaries of the datasets are provided in (see Table 8). We have no missing data in any of the datasets. Following standard practice, we removed known hypermutated genes that have no role in cancer [45], including olfactory receptors, mucins, and a few other genes such as titin. For each dataset, each gene in each patient was marked with one if it harbored one or more mutation, and with zero otherwise, resulting in the input matrix G for QuaDMutNetEx.

Table 8.: Summary of mutation-only datasets used in experimental validation of QuaDMutNetEx.

Dataset	samples (n)	genes (p)	mutations
eTNB	94	4594	6007
eGBM	120	3747	8141
eHGS	316	13278	22897
eMTB	696	13076	51255
Pan12	3281	19325	518742

5.3.2 Quantitative Evaluation of QuaDMutNetEx Results

We ran QuaDMutNetEx on the five datasets: eTNB, eGBM, eHGS, eMTB and Pan12. As with QuaDMutEx, we set the maximum size of the gene set to be $\nu = 50$. We set $k = 1$, indicating neutral stance with respect to the trade-off between coverage and excess coverage. The value of C, the weight of the gene solution size penalty, was set to 1.5 to limit the number of genes in the solution set. We experimentally set the network parameter to $\alpha = 0.15$. We ran QuaDMutNetEx for $100,000$ iterations, which corresponds to running times below 30 minutes for all datasets except the big dataset Pan12, which took 60 minutes to finish the run. Finally, the whole problem with all p genes can be run in a solver such as Gurobi, which uses a linear-programming based branch-and-bound algorithm to solve these family of problems. Gurobi has the option to run the problem for a certain amount of time which results in a solution and lower and upper bounds. Noticeably, Subproblems solution is much better than a solution generated by running the whole problem in Gurobi for one day. Also, the lower bound is a negative value, which can not be interpreted or compared to the current best solution.

As in QuanDMutEx method, we used permutation test proposed in [46] to assess statistical significance of the results returned by QuaDMutNetEx. In short, we randomly permuted the contents of each column of the input patient-gene matrix, which results in randomized dataset in which, for each gene, the number of patients harboring a mutation in the gene is preserved, but any pattern of mutation within a row, that is, within each single patient, is lost. We created 1000 randomized datasets and ran QuaDMutNetEx on each dataset. The value of the objective function observed on the original dataset was then compared with the distribution of objective function values on the randomized datasets to obtain a p-value estimate. The results of the tests, presented in Table 9 show that eGBM, eHGS, eMTB and Pan12 datasets,returns gene sets that are statistically significant at 0.05. eTNB results in gene set with p-value = 0.0634 , which is slightly higher than 0.05

Table 9.: Quantitative characteristics of QuaDMutNetEx results. Parameters were set to default, i.e, $k = 1, C = 1.5, \alpha = 0.15$. Except eTNB, all datasets solutions are statistically significant at $p < 0.05$.

Dataset	Genes	Quadratic penalty	Estimated p-value
eTNB	23	54	0.0634
eGBM	6	89	0.0175
eHGS	25	263	0.0019
eMTB	26	285	0.0000
Pan12	25	1894	0.0000

5.3.3 Comparison with other Methods

For comparison, we used DriverNet and HotNet We ran the three tools on the same five datasets: eTNB, eGBM, eHGS, eMTB and Pan12. For both tools we used the default values.

DriverNet and HotNet were executed using default parameters on the full information contained in the dataset, that is, the genomic, transcriptomic, and biological network information for DriverNet and only genomic and biological network information for HotNet.We used the default value of $\alpha = 0.15$, $k = 1$, and set the value of C to 1.5. We compared the putative cancer driver gene sets discovered by the three tools using coverage, excess coverage as described in QuaDMuteEx.

We used the objective function maximized by Dendrix to provide a comparison that is independent and involve mutual exclusivity, which can be expressed as *Dendrix score* $= n - \sum_{i=1}^{n} |G_i x - 1|$, as the metric for evaluating the tool. Essentially, the Dendrix score equals to total coverage minus coverage overlap, where total coverage is the number of patients covered by at least one gene from the given gene set, and coverage overlap is total count of all mutations in genes from the set that are in excess

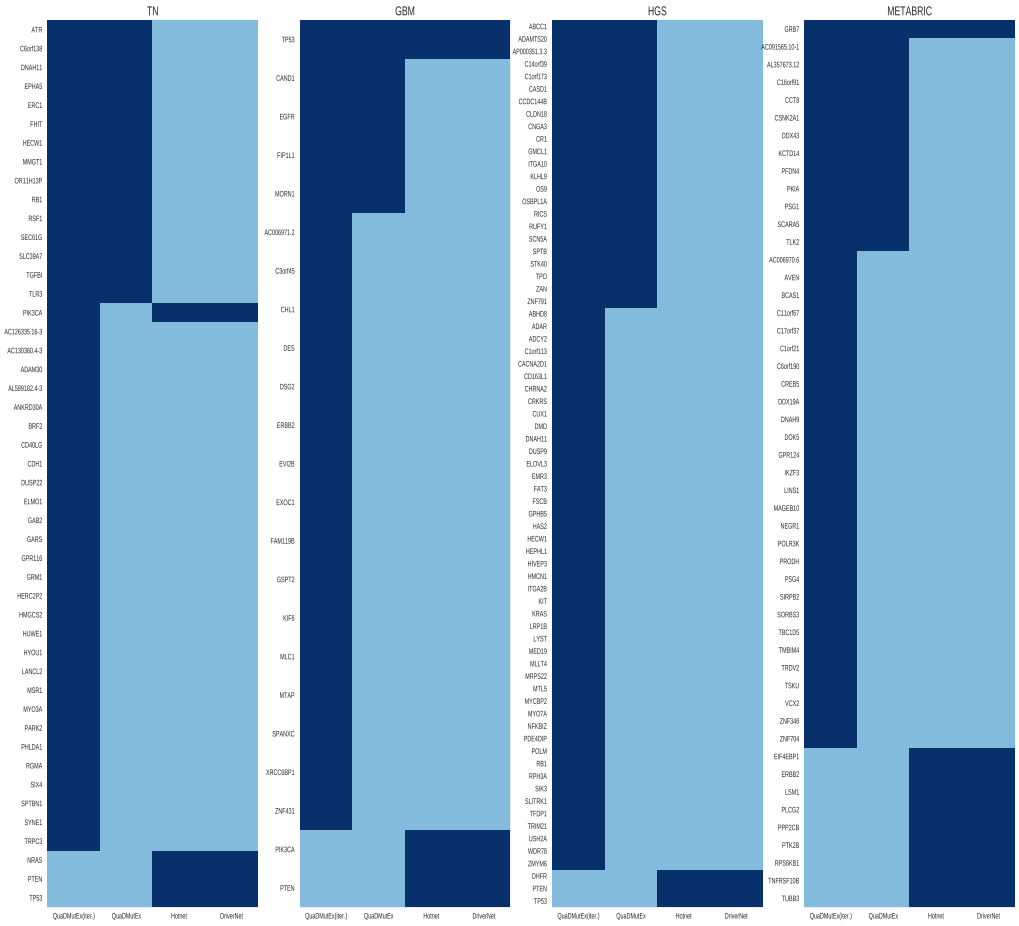
of one mutation per patient. High-quality solutions should have high Dendrix score.

The results of the tests, presented in Table 10 show that QuaDMutEx consistently returns higher quality solutions than DriverNet and HotNet. For the eGBM dataset, QuaDMutNetEx shows the same coverage and much lower excess coverage. For all datasets, QuaDMutNetEx shows much lower excess coverage than DriverNet and HotNet, at the cost of a moderate decrease in coverage. These results reflect the fact that DriverNet and HotNet are not designed to take mutual exclusivity of genes into consideration. On the other hand, DriverNet and NotNet return many more genes than QuaDMutNetEx. A single run of QuaDMutNetEx is designed to return a single set of genes with low excess coverage, and does not include all putative driver genes - these can be detected with several iterations of QuaDMutNetEx. Accordingly, we introduced iterated QuaDMutNetEx, which increases the number of genes found by QuaDMutNetEx to the numbers similar to DriverNet and HotNet. We performed four executions of QuaDMutNetEx, after each run removing the genes discovered so far from the dataset, so that they do not prevent discovery of additional genes that are not mutually exclusive with previously discovered ones. We then pooled the four high-exclusivity gene sets into a single high coverage set. Since mutual exclusivity can be expected only for a set of functionally-related genes, for example genes from a single cancer-related pathway, a single call to QuaDMutNetEx corresponds to a single-pathway query, and calling QuaDMutNetEx iteratively corresponds to a multi-pathway query, facilitating comparison with DriverNet which does not have a single-pathway focus. Genes returned by QuaDMutNetEx are to large extent different than those returned by DriverNet. Figure 9 shows that approaches used in DriverNet and HotNet are complementary to the approach used in QuaDMutNetEx.

Table 10.: Comparison between QuaDMutNetEx, HotNet and DriverNet. C=1.5, K=1, α=0.15 .

Method	Genes	Coverage	Excess coverage	Dendrix score	Quadratic penalty
eTNB: Triple negative breast cancer					
HotNet2	128	0.6809	0.7969	-118	1330.30
DriverNet	21	0.6383	0.4667	23	118.6
QuaDMutNetEx	23	0.6809	0.1563	**54**	**70.88**
eGBM: Glioblastoma multiforme					
HotNet2	37	0.7833	0.4149	10	368.10
DriverNet	17	0.9333	0.8661	-140	806.6
QuaDMutNetEx	6	0.7667	0.0326	**89**	**39.86**
eHGS: high-grade serous ovarian cancer					
HotNet2	58	0.8449	0.4307	83	1040.4
DriverNet	72	0.9335	0.6373	-35	1310.3
QuaDMutNetEx	25	0.8291	0.0878	**263**	**122.54**
eMTB: METABRIC breast cancer					
HotNet2	224	0.4424	0.7394	-1694	61393
DriverNet	90	0.4683	0.7785	-1130	14966
QuaDMutNetEx	26	0.4582	0.1038	**285**	**447.72**
Pan12: TCGA pan-cancer of 12 cancer types					
HotNet2	136	0.6809	0.7290	-5235	138640
DriverNet	NA	NA	NA	NA	NA
QuaDMutNetEx	25	0.6586	0.1191	**1894**	**1434.30**

Figure 9: Comparison of putative cancer driver gene sets returned by QuaDMutNetEx, iterated QuaDMutNetEx, and DriverNet. Genes found by a tool are in dark blue. Genes do not exsist in both DriverNet and HotNet are removed



5.3.4 Effects of Parameters on QuaDMutNetEx

The proposed method prefer solutions with connected genes. The model gives a negative rewards to the objective function if it has connected solution. The reward can be adjusted by parameter α, which leads to intresting behavior with respect to

covarage and overlap that controls by parameters C and k. Parameter C corresponds to the additional penalty for increasing the number of genes in the solution by one, and parameter k captures the ratio of penalty for one excess mutation in a patient to penalty for the patient not being covered by any mutation. We analyzed how these parameters affect the solution by running QuaDMutNetEx for 100,000 iterations with fixed $k = 1$ and parameters $\alpha = 0, 0.01, 0.05, 0.1, 0.15, 0.3, 0.5$ and $C = 0.25, 0.5, 1, 1.5, 2$. Also, we analyzed the effect on the solution set if we set $C = 1.5$ and varies n and k as follows, $\alpha = 0, 0.01, 0.05, 0.1, 0.15, 0.3, 0.5$ and $k = 0.25, 0.5, 1, 1.5, 2$.

Figure 10 shows that the parameter α achieves its design goal, that is, solutions with higher α include fewer connected components and prefer connected network. The α parameter has the following effect on coverage and excess coverage: As the value of α increases, the coverage decreases and the excess coverage increases. Furthermore, as the value of α increases, it decreases the effect of C and k. Setting α to a low value, such as 0.001, makes the effect of C and k to be more dominant. Specifically, increasing C leads reduce the solution size to only a few genes and thus necessarily lower coverage, whereas, lowering the value of C leads increase the coverage of patients by increasing the genes in the solution set. On the other hand, results in changes in parameter k result in changes in coverage and excess coverage. Specifically, lower values of k lead to higher coverage, at the cost of higher excess coverage, than high values of k.

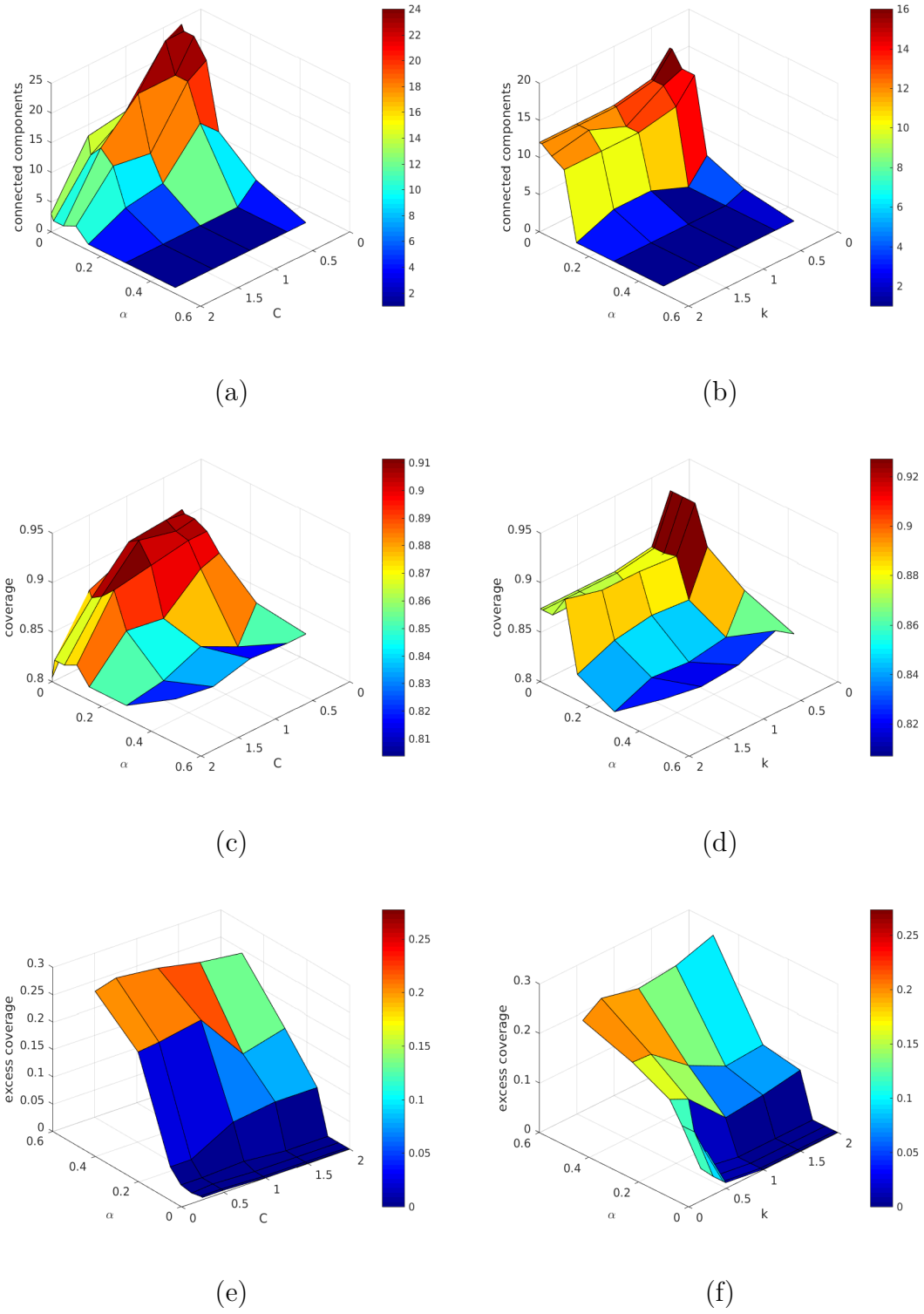


Figure 10: Effects of parameters on QuaDMutNetEx. (a), (b) Effect on connected components. (c), (d) Effect on coverage. (e), (f) Effect on excess covarage.

5.3.5 Qualitative Assessment of QuaDMutNetEx Results

To validate the ability of QuaDMutNetEx to take only mutation data and discover rare putative cancer driver genes, which are the most hard to find using traditional methods that rely on mutation frequency in patient population, in each of the datasets we focused on the genes in the solution with the fewest number of mutations. See Table 11 for a complete list of all genes in the solution, and for the number of mutations for each gene in each dataset. In addition to literature review, we also used DriverDBv2 [50], a database of previously discovered cancer driver genes, to further validate the quality of QuaDMutNetEx solutions.

In the breast tumor dataset, eTNB, TP53BP1 function as a suppressor gene [94]. ATM found to be breast cancer susceptibility alleles [95]. NCOR1 identified as driver and can be used as an independent prognostic factor for breast cancer [96, 97]. ZNF148 and SMARCA2 expression are associated with breast and other cancer types [98, 99, 100]. CCND1 is a known driver gene [97], JUN, stat3 and MYC are oncogenes [101, 102, 103]. HDAC1 is a driver and a known drug target [104, 105]. SMARCA4 is discovered as a driver in a rare type of small cell carcinoma [106]. HSPA8 is an attractive target for drug design and a potential risk factors and/or prognostic markers for breast cancer [107, 108].

In the brain tumor dataset, eGBM, MEGF11 is found to be highly expressed in GBM [109] EPHA3 is one of the most frequently mutated genes in lung cancer, it attenuates the tumor-suppressive effects of normal EPHA3 [110]. FGFR3 plays an important role in tumorigenesis [111].

In the ovarian tumor dataset, eHGS, PTPN11 overexpressed in ovarian cancer and play a big rule in cell proliferation and tumorigenesis [112]. CBLB is a proto-oncogen [113]. EGFR and MDM2 are oncogens [114, 115]. EPOR is involved in ovarian cancer growth and angiogenesis hence can be targeted for therapy [116]. PDGFRB is a target for tyrosine kinase inhibitor drugs [117]. Mutation in PLCG2 causes drug resistance in

chronic lymphocytic leukemia [118]. SH2B3 acts as a signal transduction modulator in ovarian cancers [119]. PLCG1 has a potential oncogenic role in different types of cancer [120].

In the other breast tumor dataset, eMTB, PIK3R1 is an oncogene [121]. FKBP1C is a pseudogene. KCNE1 regulates the activity of potassium channels and has no obvious role in oncogensis [122].

In Pan12, which is a pool of 12 different cancer types, the genes that occur in low frequency did not found to be oncogene or proto-oncogene.

Together, these results confirm that QuaDMutNetEx is effective in identifying cancer driver mutations even if they are rare in the analyzed patient group.

Table 11.: Putative driver gene sets discovered by QuaDMutNetEx. For each gene, in parentheses, we provide the number of patients in the dataset that harbored a mutation in that gene. Genes in bold are present in the DriverDBv2 [50] database of previously discovered cancer drivers.

Putative driver genes discovered by QuaDMutNetEx	Estimated p-value
eTNB: Triple negative breast cancer	
TP53(35) **PARK2**(6) **SAGE1 NR3C1** (3) **CREBBP NCOA1 SLC39A7** **RUNX2 TAF9 HIF1A CREB5 MLL** (2) **TP53BP1 ATM NCOR1 ZNF148** **SMARCA2 CCND1 JUN STAT3 HDAC1 SMARCA4 HSPA8 MYC** (1)	0.0634
eGBM: Glioblastoma multiforme	
CDKN2A(55) **CDK4**(18) **RB1**(12) **MEGF11**(4) **EPHA3 FGFR3**(3)	0.0175
eHGS: high-grade serous ovarian cancer	
TP53(249) **PSD3 SOS1**(3) **ERBB2 GRB2 PIK3R1 PTK2 UBC VAV3 ZAP70** **ERBB3 NTRK2 SPRY2 JAK2**(2) **PTPN11 CBLB EGFR** **EPOR PDGFRB PLCG2 SH2B3 SHC1 MDM2 SRC PLCG1**(1)	0.001
eMTB: METABRIC breast cancer	
ERBB2(84) **LETM2**(52) **CLNS1A**(37) **PSG11**(28) **MACROD2**(19) **PTEN**(16) **CYP24A1**(16) ESPNP(13) **IGF1R** (10) **TUBGCP5**(8) **EEF1A1**(6) **ADAMTSL4 ELF5**(5) AC116655.7-12 **EGFR ZNF277**(4) **ANTXRL NDN PRR12**(4) **C4orf29** **SKAP2** AE000660.1-14(3) **PIK3R1 FKBP1C KCNE1**(2)	0.0000
Pan12: TCGA pan-cancer of 12 cancer types	
TP53(1393) **PIK3CA**(611) **VHL**(228) **NPM1**(65) **CEBPAJ**(18) **ZNF672**(14) GUSBP1(9) **CHMP1B MT-CO2**(8) **SDC4 UTS2R**(7) **MRC1 TREX1**(6) **DNAJC19 SLC2A4RG**(5) **G0S2 MT1X**(4) **ANXA8L1 HSPB1 KRTAP17-1 LYL1 MRPS16 ZFHX2**(3) **MZT2B PRAMEF14**(2)	0.0000

5.4 Conclusions

In addition to to superior ability to improve on both coverage and excess coverage of the detected driver gene sets in QuaDMutEx, QuaDMutNetEx produces results that are more interpretable in terms of biological pathways.

CHAPTER 6

CONCLUSIONS

6.1 Comparison between QuaDMutEx and QuaDMutNetEx

In this dissertation, I proposed two novel methods for discovering driver mutations that are mutually exclusive. The first method introduced in this dissertation was QuaDMutEx in which we followed the de novo Pathway-centric Driver Mutation Discovery. We modeled the problem as a quadratic objective function as opposed to classical linear objective function used in some of state-of-art methods. Additionally, we used an efficient method for finding gene sets that minimizes the objective function through a combination of stochastic search and exact binary quadratic programming. QuaDMutEx also has the flexibility to adjust for the desired behavior of through parameters that control the solution size, and the trade-off between coverage and mutual exclusivity. Flexibility helps detecting low-frequency driver genes that can be missed by existing methods. The second method introduced in this dissertation was QuaDMutNetEx, in which we integrated additional terms into the objective function to represent protein-protein interactions. Essentially, QuaDMutNetEx inherits QuaDMutEx ability to find gene sets that have high coverage and mutually exclusive. In addition to high coverage and mutual exclusivity, QuaDMutNetEx prefers biologically connected genes with respect to protein-protein interaction networks. It is interesting to see how the two methods stack up against each other.

We used two criteria to compare the two methods: *a*) objective function that capture the coverage and the excess coverage that is used in Dendrix method, which is just coverage minus coverage overlap; *b*) the number of connected components, which is the number of sets of vertices in a graph that are connected to each other by paths.

Results in Table 12 shows that QuaDMutNetEx tries to maintain the coverage and the mutual exclusivity achieved by QuaDMutEx while minimizing the number of connected

Table 12.: Comparison between QuaDMutEx and QuaDMutNetEx. For both QuaDMutEx and QuaDMutNetEx, we used default parameter values $k = 1$ and $C = 1.5$ $\alpha = 0.15$ unless specified otherwise.

Method	Genes	Coverage	Excess coverage	Dendrix score	Connected components
eTNB: Triple negative breast cancer					
QuaDMutEx	19	0.8	0.03	**73**	16
QuaDMutNetEx	23	0.68	0.16	54	**2**
eGBM: Glioblastoma multiforme					
QuaDMutEx	6	0.77	0.03	89	5
QuaDMutNetEx	6	0.77	0.03	89	**3**
eHGS: high-grade serous ovarian cancer					
QuaDMutEx	14	0.87	0	**276**	12
QuaDMutNetEx	25	0.83	0.09	236	**1**
eMTB: METABRIC breast cancer					
QuaDMutEx	18	0.43	0.10	263	18
QuaDMutNetEx	25	0.44	0.10	**278**	20
QuaDMutNetEx ($\alpha = 0.3$)	28	0.37	0.16	212	**4**
Pan12: TCGA pan-cancer of 12 cancer types					
QuaDMutEx	15	0.65	0.12	1870	12
QuaDMutNetEx	25	0.66	0.12	1894	20
QuaDMutNetEx ($\alpha = 0.3$)	26	0.67	0.12	**1903**	19
QuaDMutNetEx ($\alpha = 0.6$)	25	0.64	0.15	1766	**2**

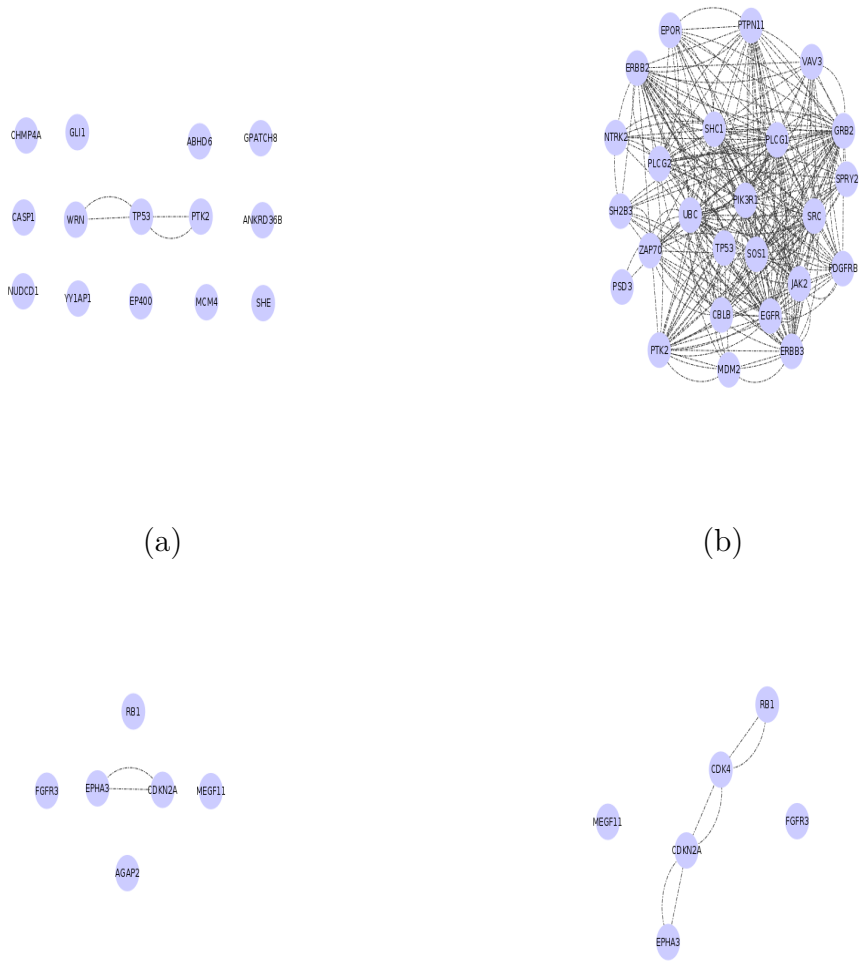


Figure 11: Effects of α on the connected components (a) eHGS run on QuaDMutEx, (b) eHGS run on QuaDMutNetEx, (c) eGBM run on QuaDMutEx ,(d) eGBM run on QuaDMutNetEx

components. The effect of the network term N in the big data sets, eMTB and Pan12, was moderate. Thus, we increased α to give more weight for connectivity. As a result, we have the desired goal of low number of connected components. Noticeably, in eGBM dataset we have identical number of genes, coverage, excess coverage and Dendrix score,

but a lower number of connected components, which clearly shows there is a difference between QuaDMutEx and QuaDMutNetEx.

We can visualize the difference between the two methods in Figure 11 that shows that the new network term N in QuaDMutNetEx has achieved its purpose. The gene sets in QuaDMutNetEx are different from the gene sets in QuaDMutEx. Also, the gene set in QuaDMutNetEx have fewer connected components.

Comparing two methods, there is no clear answer which one is better. De novo method gives results that is independent from biological networks. It is an advantage if we consider that knowledge of biological network is incomplete. More importantly, the available data involve biases which might affect the true structure of the network in a way that can not be measured. Hence, biological networks must be used with caution. On the other hand, using biological network leads to better interpretability of results especially if the network used has high quality.

6.2 Contibution of QuaDMutEx and QuaDMutNetEx

QuaDMutEx incorporates novel gene set penalty that includes non-linear penalization of excess mutations in a single patient. Also, QuaDMutEx controls the solution size, and has the flexibility to tradeoff between gene coverage to the patients and the exclusivity of the coverage using the two-parameters k and C. Additionally, QuaDMutEx uses a computationally efficient method for finding gene sets that minimize the penalty through a combination of stochastic search and exact binary quadratic programming.

QuaDMutNetEx inherits all the functionality of QuaDMutEx and adds the feature of biological networks rewards. In other words, QuaDMutNetEx prefers solution sets that have adjacent genes in biological networks.

6.3 Conclusion

In previous chapters, we showed that both methods are an improvement compared to existing methods. QuaDMutEx and QuaDMutNetEx have a superior ability to improve on both coverage and excess coverage of the detected driver gene sets on datasets from different types of cancer. Additionally, QuaDMutNetEx discovers sets with a low number of connected components. This indicates the effectiveness of both tools in driver genes discovery for any new datasets. In particular, these two tools help in prioritizing of putative cancer genes for further in-depth extended study. QuaDMutEx and QuaDMutNetEx should be part of a state-of-the-art toolbox in the driver gene discovery pipeline since both can help in detecting low-frequency driver genes that can be missed by existing methods.

CHAPTER 7

FUTURE WORK

The main direction of future work could center on alternative approaches to solve the optimization problems.

7.0.1 Genetic Algorithm

Genetic algorithm (GA) is a heuristic programming approach that mimics a simple version of biological evolution theory towards better fitness or objective function. Typically GA relies on the following elements: mutation, crossover, and selection. We introduced genetic algorithm in section 3.2.1 as one of the methods to solve Unconstrained Binary Quadratic Programs (UBQP). Here, we are proposing using the genetic algorithm to solve our UBQP problem. We can encode the current solution as a binary genotype whose size is equal to the total number of genes. i.e. if ith gene is 1, it indicates that ith gene is selected to be in the solution set and vice versa. Then, we can evaluate different fitness functions, namely, Dendrix, QuaDMutEx, and QuaDMutNetEx. Also, we can use multi-objective optimization. In such encoding, GA operators such as mutation can be easly implented in the GA's individual representation. Genetic algorithm is implemented efficiently in multiple off-the-shelf software such as a "plug and play" JCLEC software system for Evolutionary Computation (EC) research [123]. Solving our problem using genetic algorithm is promising and might give a better objective function value in a reasonable time.

7.0.2 Quantum annealing

Quantum computation is a new field that is currently being explored to solve certain fundamental computational problems faster than any existing algorithms. Recently,

quantum annealing (QA) or adiabatic quantum optimization was introduced as an approach that potentially can outperform classical optimization algorithms. QA is physically implemented in quantum annealing processors or D-Wave (DW) processors [124]. Our objective function involves only pairwise terms and thus can be in principle optimized using quantum annealing.

REFERENCES

[1] Ali Bashashati et al. "DriverNet: uncovering the impact of somatic driver mutations on transcriptional networks in cancer". In: *Genome Biology* 13.12 (2012), R124.

[2] Mark DM Leiserson et al. "Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes". In: *Nature Genetics* 47.2 (2015), pp. 106–114.

[3] Scott D Kahn. "On the Future of Genomic Data". In: *Science* 331.6018 (2011), pp. 728–729.

[4] Michael R. Stratton, Peter J. Campbell, and P. Andrew Futreal. "The Cancer Genome". In: *Nature* 458.7239 (2009), pp. 719–724.

[5] Julienne M Mullaney et al. "Small Insertions and Deletions (INDELs) in Human Genomes". In: *Human Molecular Genetics* 19.R2 (2010), R131–R136.

[6] Anthony JF Griffiths et al. *How DNA changes affect phenotype*. WH Freeman, 2000.

[7] American Cancer Society. *Cancer Facts and Figures 2016*. `http://www.cancer.org/research/cancer-facts-statistics/all-cancer-facts-figures/cancer-facts-figures-2018/`.

[8] Peter Devilee and Cees J Cornelisse. "Somatic Genetic Changes in Human Breast Cancer". In: *Biochimica et Biophysica Acta (BBA)-Reviews on Cancer* 1198.2 (1994), pp. 113–130.

[9] Douglas Hanahan and Robert A Weinberg. "The Hallmarks of Cancer". In: *Cell* 100.1 (2000), pp. 57–70.

[10] Michael R. Stratton. "Exploring the Genomes of Cancer Cells: Progress and Promise". In: *Science* 331.6024 (2011), pp. 1553–1558.

[11] Daniel A Haber and Jeff Settleman. "Cancer: Drivers and Passengers". In: *Nature* 446.7132 (2007), pp. 145–146.

[12] Michael S Lawrence et al. "Discovery and Saturation Analysis of Cancer Genes Across 21 Tumour Types". In: *Nature* 505.7484 (2014), pp. 495–501.

[13] Christopher D McFarland et al. "Impact of Deleterious Passenger Mutations on Cancer Progression". In: *Proceedings of the National Academy of Sciences* 110.8 (2013), pp. 2910–2915.

[14] Pauline C Ng and Steven Henikoff. "Predicting Deleterious Amino Acid Substitutions". In: *Genome Research* 11.5 (2001), pp. 863–874.

[15] Pauline C Ng and Steven Henikoff. "SIFT: Predicting Amino Acid Changes that Affect Protein Function". In: *Nucleic Acids Research* 31.13 (2003), pp. 3812–3814.

[16] Hannah Carter et al. "Cancer-specific High-throughput Annotation of Somatic Mutations: Computational Prediction of Driver Missense Mutations". In: *Cancer Research* 69.16 (2009), pp. 6660–6667.

[17] Ivan A Adzhubei et al. "A Method and Server for Predicting Damaging Missense Mutations". In: *Nature Methods* 7.4 (2010), pp. 248–249.

[18] Boris Reva, Yevgeniy Antipin, and Chris Sander. "Predicting the Functional Impact of Protein Mutations: Application to Cancer Genomics". In: *Nucleic Acids Research* (2011), 39:e118.

[19] Julia R Pon and Marco A Marra. "Driver and Passenger Mutations in Cancer". In: *Annual Review of Pathology: Mechanisms of Disease* 10 (2015), pp. 25–50.

[20] David Tamborero, Abel Gonzalez-Perez, and Nuria Lopez-Bigas. "Identification of Oncogenic Driver Mutations". In: *Exp Med* 32 (2014), pp. 1–7.

[21] Nathan D Dees et al. “MuSiC: Identifying Mutational Significance in cancer Genomes”. In: *Genome research* 22.8 (2012), pp. 1589–1598.

[22] Michael S Lawrence et al. “Mutational Heterogeneity in Cancer and The Search for New Cancer-associated Genes”. In: *Nature* 499.7457 (2013), pp. 214–218.

[23] Benjamin J Raphael et al. “Identifying Driver Mutations in Sequenced Cancer Genomes: Computational Approaches to Enable Precision Medicine”. In: *Genome Med* 6.5 (2014).

[24] Jiajia Chen, Maomin Sun, and Bairong Shen. “Deciphering Oncogenic Drivers: From Single Genes to Integrated Pathways”. In: *Briefings In Bioinformatics* (2014), bbu039.

[25] Michael C Wendl et al. “PathScan: A Tool for Discerning Mutational Significance in Groups of Putative Cancer Genes”. In: *Bioinformatics* 27.12 (2011), pp. 1595–1602.

[26] Fabio Vandin, Eli Upfal, and Benjamin J Raphael. “Algorithms for Detecting Significantly Mutated Pathways in Cancer”. In: (2010), pp. 506–521.

[27] Fabio Vandin et al. “Discovery of Mutated Subnetworks Associated with Clinical Data in cancer”. In: *Pac Symp Biocomput*. Vol. 2012. 2012, pp. 55–66.

[28] Giovanni Ciriello et al. “Mutual Exclusivity Analysis Identifies Oncogenic Network Modules”. In: *Genome Research* 22.2 (2012), pp. 398–406.

[29] Mark DM Leiserson et al. “Pan-cancer Network Analysis Identifies Combinations of Rare Somatic Mutations Across Pathways and Protein Complexes”. In: *Nature Genetics* 47.2 (2015), pp. 106–114.

[30] Christopher A Miller et al. “Discovering Functional Modules by Identifying Recurrent and Mutually Exclusive Mutational Patterns in Tumors”. In: *BMC Medical Genomics* 4.1 (2011), p. 34.

[31] Fabio Vandin, Eli Upfal, and Benjamin J Raphael. “De novo Discovery of Mutated Driver Pathways in Cancer”. In: *Genome Research* 22.2 (2012), pp. 375–385.

[32] Mark DM Leiserson et al. “Simultaneous Identification of Multiple Driver Pathways in Cancer”. In: *PLoS Comput Biol* 9.5 (2013), e1003054.

[33] Bert Vogelstein and Kenneth W Kinzler. “Cancer Genes and The Pathways they Control”. In: *Nature Medicine* 10.8 (2004), pp. 789–799.

[34] Gary Kochenberger et al. “The Unconstrained Binary Quadratic Programming Problem: A Survey”. In: *Journal of Combinatorial Optimization* 28.1 (2014), pp. 58–81.

[35] Peter L Hammer and Sergiu Rudeanu. *Boolean Methods in Operations Research and Related Areas*. Vol. 7. Springer Science & Business Media, 2012.

[36] Melanie Mitchell. “Genetic Algorithms: An Overview”. In: *Complexity* 1.1 (1995), pp. 31–39.

[37] Natalio Krasnogor and Jim Smith. “A memetic algorithm with self-adaptive local search: TSP as a case study”. In: *Proceedings of the 2nd Annual Conference on Genetic and Evolutionary Computation*. Morgan Kaufmann Publishers Inc. 2000, pp. 987–994.

[38] Barry A Cipra. “The Best of The 20th Century: Editors Name Top 10 Algorithms”. In: *SIAM News* 33.4 (2000), pp. 1–2.

[39] Kevin P Murphy. *Machine Learning: A Probabilistic Perspective*. MIT press, 2012.

[40] Nicholas Metropolis et al. “Equation of State Calculations by Fast Computing Machines”. In: *The Journal of Chemical Physics* 21.6 (1953), pp. 1087–1092.

[41] W Keith Hastings. “Monte Carlo Sampling Methods Using Markov chains and Their Applications”. In: *Biometrika* 57.1 (1970), pp. 97–109.

[42] Dana Randall. “Rapidly Mixing Markov Chains with Applications in Computer Science and Physics”. In: *Computing in Science & Engineering* 8.2 (2006), pp. 30–41.

[43] Yahya Bokhari and Tomasz Arodz. “QuaDMutEx: quadratic driver mutation explorer”. In: *BMC bioinformatics* 18.1 (2017), p. 458.

[44] Simona Constantinescu et al. “TiMEx: a waiting time model for mutually exclusive cancer alterations”. In: *Bioinformatics* 32.7 (2015), pp. 968–975.

[45] Michael S Lawrence et al. “Mutational heterogeneity in cancer and the search for new cancer-associated genes”. In: *Nature* 499.7457 (2013), pp. 214–218.

[46] Fabio Vandin, Eli Upfal, and Benjamin J Raphael. “De novo discovery of mutated driver pathways in cancer”. In: *Genome Research* 22.2 (2012), pp. 375–385.

[47] Christopher A Miller et al. “Discovering functional modules by identifying recurrent and mutually exclusive mutational patterns in tumors”. In: *BMC Medical Genomics* 4.1 (2011), p. 1.

[48] Mark DM Leiserson et al. “CoMEt: a statistical approach to identify combinations of mutually exclusive alterations in cancer”. In: *Genome Biology* 16.1 (2015), p. 1.

[49] Christos M Dimitrakopoulos and Niko Beerenwinkel. “Computational approaches for the identification of cancer genes and pathways”. In: *Wiley Interdisciplinary Reviews: Systems Biology and Medicine* 9.1 (2017).

[50] I-Fang Chung et al. “DriverDBv2: a database for human cancer driver gene research”. In: *Nucleic Acids Research* 44.D1 (2016), pp. D975–D979.

[51] Bingtian Zhao et al. “MicroRNA let-7c inhibits migration and invasion of human non-small cell lung cancer by targeting ITGB3 and MAP4K3”. In: *Cancer Letters* 342.1 (2014), pp. 43–51.

[52] Yunlong Lei et al. “Proteomics identification of ITGB3 as a key regulator in reactive oxygen species-induced migration and invasion of colorectal cancer cells”. In: *Molecular & Cellular Proteomics* 10.10 (2011), pp. M110–005397.

[53] Xiaobo Chen et al. “MicroRNA-145 targets TRIM2 and exerts tumor-suppressing functions in epithelial ovarian cancer”. In: *Gynecologic Oncology* 139.3 (2015), pp. 513–519.

[54] Martin Balastik et al. “Deficiency in ubiquitin ligase TRIM2 causes accumulation of neurofilament light chain and neurodegeneration”. In: *Proceedings of the National Academy of Sciences* 105.33 (2008), pp. 12016–12021.

[55] Khanh Do, James H Doroshow, and Shivaani Kummar. “Wee1 kinase as a target for cancer therapy”. In: *Cell Cycle* 12.19 (2013), pp. 3348–3353.

[56] Anindya Bagchi et al. “CHD5 is a tumor suppressor at human 1p36”. In: *Cell* 128.3 (2007), pp. 459–475.

[57] Alessandro Beghini et al. “The neural progenitor restricted isoform of the MARK4 gene in 19q13.2 is upregulated in human gliomas and overexpressed in a subset of glioblastoma cell lines”. In: *Oncogene* 22.17 (2003), pp. 2581–2591.

[58] Roger S Holmes, Laura A Cox, and John L VandeBerg. “Mammalian carboxylesterase 3: comparative genomics and proteomics”. In: *Genetica* 138.7 (2010), pp. 695–708.

[59] Virginie Clement et al. “HEDGEHOG-GLI1 signaling regulates human glioma growth, cancer stem cell self renewal, and tumorigenicity”. In: *Current Biology* 17.2 (2007), pp. 165–172.

[60] Young A Yoo et al. “Sonic hedgehog pathway promotes metastasis and lymphangiogenesis via activation of Akt, EMT, and MMP-9 pathway in gastric cancer”. In: *Cancer Research* 71.22 (2011), pp. 7061–7070.

[61] Michael Johnson, Manisha Sharma, and Beric R Henderson. “IQGAP1 regulation and roles in cancer”. In: *Cellular Signalling* 21.10 (2009), pp. 1471–1478.

[62] Priscila Oliveira de Lima et al. “Effect of eosinophil cationic protein on human oral squamous carcinoma cell viability”. In: *Molecular and Clinical Oncology* 3.2 (2015), pp. 353–356.

[63] Tong Liu et al. “p21-Activated kinase 6 (PAK6) inhibits prostate cancer growth via phosphorylation of androgen receptor and tumorigenic E3 ligase murine double minute-2 (Mdm2)”. In: *Journal of Biological Chemistry* 288.5 (2013), pp. 3359–3369.

[64] Koichi Tomoshige et al. “Germline mutations causing familial lung cancer”. In: *Journal of Human Genetics* 60.10 (2015), pp. 597–603.

[65] Dingzhi Wang and Raymond N DuBois. “Eicosanoids and cancer”. In: *Nature Reviews Cancer* 10.3 (2010), pp. 181–193.

[66] Paola A Bignone et al. “RPS6KA2, a putative tumour suppressor gene at 6q27 in sporadic epithelial ovarian cancer”. In: *Oncogene* 26.5 (2007), pp. 683–700.

[67] Jack M Lionberger and Thomas E Smithgall. “The c-Fes protein-tyrosine kinase suppresses cytokine-independent outgrowth of myeloid leukemia cells induced by Bcr-Abl”. In: *Cancer Research* 60.4 (2000), pp. 1097–1103.

[68] Maria Ilaria Del Principe et al. “Clinical significance of BAX/BCL-2 ratio in chronic lymphocytic leukemia”. In: *Haematologica* 101.1 (2016), pp. 77–85.

[69] Rosa M Apolinario et al. “Prognostic value of the expression of p53, bcl-2, and bax oncoproteins, and neovascularization in patients with radically resected non-small-cell lung cancer”. In: *Journal of Clinical Oncology* 15.6 (1997), pp. 2456–2466.

[70] Pengyuan Liu et al. “Identification of somatic mutations in non-small cell lung carcinomas using whole-exome sequencing”. In: *Carcinogenesis* 33.7 (2012), pp. 1270–1276.

[71] Stella Koutros et al. “Pooled analysis of phosphatidylinositol 3-kinase pathway variants and risk of prostate cancer”. In: *Cancer Research* 70.6 (2010), pp. 2389–2396.

[72] Dario Palmieri et al. “Ran Binding Protein 9 (RanBP9) is a novel mediator of cellular DNA damage response in lung cancer cells”. In: *Oncotarget* 7.14 (2016), p. 18371.

[73] Motofumi Kumazoe et al. “67-kDa laminin receptor increases cGMP to induce cancer-selective apoptosis”. In: *Journal of Clinical Investigation* 123.2 (2013).

[74] Shan-Chun Zhang et al. “RPSA gene mutants associated with risk of colorectal cancer among the Chinese population”. In: *Asian Pacific Journal of Cancer Prevention* 14.12 (2013), pp. 7127–7131.

[75] Vanessa Andries et al. “NBPF1, a tumor suppressor candidate in neuroblastoma, exerts growth inhibitory effects by inducing a G1 cell cycle arrest”. In: *BMC Cancer* 15.1 (2015), p. 391.

[76] Sylvain Meunier et al. “An epigenetic regulator emerges as microtubule minus-end binding and stabilizing factor in mitosis”. In: *Nature Communications* 6 (2015), p. 7889.

[77] Zhi-Gang Li et al. “Hypermethylation of two CpG sites upstream of CASP8AP2 promoter influences gene expression and treatment outcome in childhood acute lymphoblastic leukemia”. In: *Leukemia Research* 37.10 (2013), pp. 1287–1293.

[78] Maria Sokolova et al. “Genome-wide screen of cell-cycle regulators in normal and tumor cells identifies a differential response to nucleosome depletion”. In: *Cell Cycle* 16.2 (2017), pp. 189–199.

[79] Y Li et al. “Comparative proteomics analysis of human osteosarcomas and benign tumor of bone”. In: *Cancer Genetics and Cytogenetics* 198.2 (2010), pp. 97–106.

[80] Sabine Maier et al. “DNA-methylation of the homeodomain transcription factor PITX2 reliably predicts risk of distant disease recurrence in tamoxifen-treated, node-negative breast cancer patients–technical and clinical validation in a multicentre setting in collaboration with the European Organisation for Research and Treatment of Cancer (EORTC) PathoBiology group”. In: *European Journal of Cancer* 43.11 (2007), pp. 1679–1686.

[81] Qingyang Zhang, Joanna E Burdette, and Ji-Ping Wang. “Integrative network analysis of TCGA data for ovarian cancer”. In: *BMC Systems Biology* 8.1 (2014), p. 1338.

[82] Junan Li, Anjali Mahajan, and Ming-Daw Tsai. “Ankyrin repeat: a unique motif mediating protein-protein interactions”. In: *Biochemistry* 45.51 (2006), pp. 15168–15178.

[83] Kevin J Cheung et al. “Polyclonal breast cancer metastases arise from collective dissemination of keratin 14-expressing tumor cell clusters”. In: *Proceedings of the National Academy of Sciences* 113.7 (2016), E854–E863.

[84] Dawei Li et al. “Msi-1 is a predictor of survival and a novel therapeutic target in colon cancer”. In: *Annals of Surgical Oncology* 18.7 (2011), pp. 2074–2083.

[85] Megan J Bywater et al. “Dysregulation of the basal RNA polymerase transcription apparatus in cancer”. In: *Nature Reviews Cancer* 13.5 (2013), pp. 299–314.

[86] Eric R Gamazon et al. “Copy number polymorphisms and anticancer pharmacogenomics”. In: *Genome Biology* 12.5 (2011), R46.

[87] Richard W Park et al. “Identification of rare germline copy number variations over-represented in five human cancer types”. In: *Molecular Cancer* 14.1 (2015), p. 25.

[88] Jinliang Huan et al. “Insights into significant pathways and gene interaction networks underlying breast cancer cell line MCF-7 treated with 17β-estradiol (E2)”. In: *Gene* 533.1 (2014), pp. 346–355.

[89] Yu Mei et al. “Genomic profile of human meningioma cell lines”. In: *PloS ONE* 12.5 (2017), e0178322.

[90] Marisa J Wainszelbaum et al. “The hominoid-specific oncogene TBC1D3 activates Ras and modulates epidermal growth factor receptor signaling and trafficking”. In: *Journal of Biological Chemistry* 283.19 (2008), pp. 13233–13242.

[91] Allen D Bosley, Sudipto Das, and Thorkell Andresson. “A Role for Protein–Protein Interaction Networks in the Identification and Characterization of Potential Biomarkers”. In: *Proteomic and Metabolomic Approaches to Biomarker Discovery*. Elsevier, 2013, pp. 333–347.

[92] Roberto Tamassia. *Handbook of Graph Drawing and Visualization*. CRC press, 2013.

[93] Gary Kochenberger et al. “The unconstrained binary quadratic programming problem: a survey”. In: *Journal of Combinatorial Optimization* 28.1 (2014), pp. 58–81.

[94] Xiaoyan Li et al. “53BP1 functions as a tumor suppressor in breast cancer via the inhibition of NF-κB through miR-146a”. In: *Carcinogenesis* 33.12 (2012), pp. 2593–2600.

[95] M Ahmed and N Rahman. “ATM and breast cancer susceptibility”. In: *Oncogene* 25.43 (2006), p. 5906.

[96] Zhenhuan Zhang et al. “NCOR1 mRNA is an independent prognostic factor for breast cancer”. In: *Cancer letters* 237.1 (2006), pp. 123–129.

[97] Philip J Stephens et al. “The landscape of cancer genes and mutational processes in breast cancer”. In: *Nature* 486.7403 (2012), p. 400.

[98] Tanya Gritsko et al. “Persistent activation of stat3 signaling induces survivin gene expression and confers resistance to apoptosis in human breast cancer cells”. In: *Clinical cancer research* 12.1 (2006), pp. 11–19.

[99] Zhijiu Zhong et al. “Cyclin D1/cyclin-dependent kinase 4 interacts with filamin A and affects the migration and invasion potential of breast cancer cells”. In: *Cancer research* (2010), pp. 0008–5472.

[100] Deng Pan, Masha Kocherginsky, and Suzanne D Conzen. “Activation of the glucocorticoid receptor is associated with poor prognosis in estrogen receptor-negative breast cancer”. In: *Cancer research* (2011).

[101] Peter Angel et al. “The jun proto-oncogene is positively autoregulated by its product, Jun/AP-1”. In: *Cell* 55.5 (1988), pp. 875–885.

[102] Jacqueline F Bromberg et al. “Stat3 as an oncogene”. In: *Cell* 98.3 (1999), pp. 295–303.

[103] JM Varley et al. “Alterations to either c-erbB-2 (neu) or c-myc proto-oncogenes in breast carcinomas correlate with poor short-term prognosis.” In: *Oncogene* 1.4 (1987), pp. 423–430.

[104] Tony Kouzarides. “Histone acetylases and deacetylases in cell proliferation”. In: *Current opinion in genetics & development* 9.1 (1999), pp. 40–48.

[105] Ricky W Johnstone. “Histone-deacetylase inhibitors: novel drugs for the treatment of cancer”. In: *Nature reviews Drug discovery* 1.4 (2002), p. 287.

[106] Douglas I Lin et al. “Comprehensive genomic profiling reveals inactivating SMARCA4 mutations and low tumor mutational burden in small cell carcinoma of the ovary, hypercalcemic-type”. In: *Gynecologic oncology* 147.3 (2017), pp. 626–633.

[107] Michael Y Sherman and Vladimir L Gabai. “Hsp70 in cancer: back to the future”. In: *Oncogene* 34.32 (2015), p. 4153.

[108] Flora Zagouri et al. “HSP90, HSPA8, HIF-1 alpha and HSP70-2 polymorphisms in breast cancer: a case–control study”. In: *Molecular biology reports* 39.12 (2012), pp. 10873–10879.

[109] Feng Yu and Wei Ming Fu. “Identification of differential splicing genes in gliomas using exon expression profiling”. In: *Molecular medicine reports* 11.2 (2015), pp. 843–850.

[110] Guanglei Zhuang et al. “Effects of cancer-associated EPHA3 mutations on lung cancer”. In: *Journal of the National Cancer Institute* 104.15 (2012), pp. 1183–1198.

[111] Jun-Hyeog Jang, Ki-Hyuk Shin, and Jae-Gahb Park. “Mutations in fibroblast growth factor receptor 2 and fibroblast growth factor receptor 3 genes associated with human gastric and colorectal cancers”. In: *Cancer research* 61.9 (2001), pp. 3541–3543.

[112] ZhongQian Hu et al. “shP2 overexpression enhances the invasion and metastasis of ovarian cancer in vitro and in vivo”. In: *OncoTargets and therapy* 10 (2017), p. 3881.

[113] Qian Dong et al. “MicroRNA 891b is an independent prognostic factor of pancreatic cancer by targeting Cbl-b to suppress the growth of pancreatic cancer cells”. In: *Oncotarget* 7.50 (2016), p. 82338.

[114] Masaki Mandai et al. “Expression of metastasis-related nm23-H1 and nm23-H2 genes in ovarian carcinomas: correlation with clinicopathology, EGFR, c-erbB-2, and c-erbB-3 genes, and sex steroid receptor expression”. In: *Cancer research* 54.7 (1994), pp. 1825–1830.

[115] John E Landers, Suzanne L Cassel, and Donna L George. “Translational enhancement of mdm2 oncogene expression in human tumor cells containing a stabilized wild-type p53 protein”. In: *Cancer research* 57.16 (1997), pp. 3562–3568.

[116] Yoshiko Yasuda et al. “Erythropoietin is involved in growth and angiogenesis in malignant tumours of female reproductive organs”. In: *Carcinogenesis* 23.11 (2002), pp. 1797–1805.

[117] Rosemarie E Schmandt et al. “Expression of c-ABL, c-KIT, and platelet-derived growth factor receptor-β in ovarian serous carcinoma and normal ovarian surface epithelium”. In: *Cancer* 98.4 (2003), pp. 758–764.

[118] D Jones et al. “PLCG2 C2 domain mutations cooccur with BTK and PLCG2 resistance mutations in chronic lymphocytic leukemia undergoing ibrutinib treatment”. In: *Leukemia* 31.7 (2017), p. 1645.

[119] Ling-Wen Ding et al. “LNK (SH2B3): paradoxical effects in ovarian cancer”. In: *Oncogene* 34.11 (2015), p. 1463.

[120] Rebeca Manso. “PLCG1 (Phospholipase C, Gamma 1)”. In: (2014).

[121] Amanda J Philp et al. “The phosphatidylinositol 3-kinase p85α gene is an oncogene in human ovarian and colon tumors”. In: *Cancer research* 61.20 (2001), pp. 7426–7429.

[122] Heimo Ehmke. “Physiological functions of the regulatory potassium channel subunit KCNE1”. In: *American Journal of Physiology-Regulatory, Integrative and Comparative Physiology* 282.3 (2002), R637–R638.

[123] Alberto Cano et al. “A classification module for genetic programming algorithms in JCLEC”. In: *The Journal of Machine Learning Research* 16.1 (2015), pp. 491–494.

[124] Richard Y Li et al. “Quantum annealing versus classical machine learning applied to a simplified computational biology problem”. In: *NPJ quantum information* 4.1 (2018), p. 14.

VITA

Yahya Abdulfattah Bokhari was born on November 29, 1981, in Makkah, Saudi Arabia. He received his Bachelor of Science in Medical Technology Sciences, from King Abdulaziz University, Riyadh, Saudi Arabia in 2004 and subsequently worked as a medical technologist in a clinical cytogenetics laboratory in King Abdulaziz Medical City, Riyadh, Saudi Arabia for four years. In 2008, he obtained the American Society Of Clinical Pathology Board of Certification in Cytogenetics CG(ASCP). He joined the Bioinformatics program in Virginia Commonwealth University in 2010 and earned his Master's of Science degree in Bioinformatics in 2013. On the same year, he received his Post-baccalaureate Certificate in Computer Science. He joined the department of Computer Science in 2014. In 2016, While he is seeking his PhD degree in Computer Science, he earned his Post-baccalaureate Certificate in statistics from department of Statistical Sciences and Operations Research.