Novel Approaches for Cancer Subtypes Discovery and Pathway Analysis

by

Phi Hung Bya

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
May 4, 2024

Keywords: Cancer Subtyping, Integrative Analysis, Pathway Analysis, Gene Set Enrichment Analysis

Approved by

Tin Chi Nguyen, Chair, Associate Professor of Computer Science and Software Engineering
Wei-Shinn Ku, Professor of Computer Science and Software Engineering
Gerry Vernon Dozier, Professor of Computer Science and Software Engineering
Anh Nguyen, Assistant Professor of Computer Science and Software Engineering
Nam Tran, University Reader, Research Professor of National Center for Asphalt Technology

Abstract

Complex diseases, particularly cancer, encompass a wide range of disorders, from aggressive and lethal to indolent lesions with low or delayed potential for progression to death. Treatment options and success heavily depend on the disease subtype of individual patients, which are often determined based on their molecular features. The advent of high-throughput platforms in the past decade has generated a wealth of molecular data, not only for gene expression but also for other molecular data, including DNA methylation and non-coding microRNA. This has significantly increased the number of samples to cover the heterogeneity of the diseases and allowed for subtyping from a more holistic perspective, considering phenomena at different molecular levels in a single analysis. However, the stochastic nature of omics data and its high dimensionality have hindered consensus among different omic levels and the interpretability of the discovered subtypes, necessitating a powerful integrative technique to handle the noise, high dimensionality, and large sample sizes for improved subtyping.

Following subtyping, understanding the biological mechanisms driving subtype differences in complex diseases remains crucial for developing effective treatments and therapies. Pathway analysis and gene set enrichment analysis are widely used to determine significantly impacted biological processes between conditions. However, current pathway analyses are biased towards well-studied diseases, sensitive to noise, and have limited validation across diverse datasets and conditions, making their effectiveness unclear in analyzing new diseases, complex etiologies, or in analyzing data with weak signals compared to controls. Moreover, inconsistencies among different methods hinder interpretation and confidence in the results for downstream analyses.

This dissertation addresses these challenges by investigating a wide range of techniques for integrating multi-omics data to subtype cancer patients, including matrix factorization, genetic algorithms, and similarity-based methods. We introduce several novel subtyping frameworks,

including Multi-objective Genetic K-means clustering Algorithm (MGKA), Disease Subtyping using Community detection from Consensus networks (DSCC), PINSPlus, and Subtyping Multi-omics using a Randomized Transformation (SMRT). MGKA utilizes a multi-objective genetic algorithm to refine the k-means clustering algorithm and automatically determine the optimal number of subtypes. DSCC employs a consensus network approach, building patient similarity networks from individual data types and using community detection to identify robust subtypes. PINSPlus is an extension of the original PINS method, integrating multiple data types and providing a more accurate and efficient subtyping analysis. SMRT is capable of integrating a large number of omics data types to subtype cancer patients. Through an extensive analysis of over 11,000 patients across 37 cancer types, we demonstrate the ability of these methods to detect cancer subtypes with significant differences in patient risk and survival. Notably, these methods easily handle a large number of data types and patients, are robust against noise and missing data, and gain accuracy as more data types are integrated.

For the second challenge, we first introduce a web interface that offers pathway analysis using multiple methods and datasets in a single session with rich visualization features, allowing life scientists to easily conduct pathway analysis, compare results from different methods and datasets, and reach better consensus for downstream analyses. We then introduce a novel consensus pathway analysis approach, Perturbation-based Gene Set Analysis (PGSA), which efficiently determines significantly impacted pathways across a wide range of diseases. We analyzed 421 datasets from more than 30 diseases, demonstrating PGSA's superior performance compared to state-of-the-art methods in identifying significantly impacted pathways. This marks the first time a pathway analysis method has been tested on such a large number of datasets and diseases to prevent bias and overfitting to well-studied diseases.

Acknowledgments

I would like to express my deepest gratitude to my advisor, Dr. Tin Nguyen, for his exceptional guidance, unwavering support, and continuous encouragement throughout my doctoral journey. His mentorship, patience, and steadfast belief in my abilities have been instrumental in shaping my academic growth. Dr. Nguyen's passion for research and commitment to excellence have been a constant source of inspiration, motivating me to push the boundaries of my knowledge and strive for greatness. His invaluable insights, thought-provoking discussions, and intellectual challenges have broadened my perspectives and enriched my research experience.

I also extend my sincere thanks to my esteemed committee members, Dr. Anh Nguyen, Dr. Wei-Shinn Ku, Dr. Gerry Dozier, and Dr. Nam Tran for their invaluable feedback and suggestions. Their expertise, time, and effort in reviewing my dissertation and providing constructive input have been crucial to the successful completion of this work. I am truly grateful for their guidance and the significant impact they have had on my research.

My heartfelt appreciation goes out to my colleagues at Auburn University, especially Ha Viet Nguyen, who have become dear friends along the way. Their camaraderie, friendship, and unwavering support have made this journey more enjoyable and memorable. I am also thankful for the friendship and collaboration of Zeynab Maghsoudi from the University of Nevada, Reno. Her contributions to my research and her personal support have been invaluable.

Words cannot express the depth of my gratitude towards my family for their unconditional love, encouragement, and unwavering support throughout my academic journey. My parents and sisters have been my pillars of strength, and their belief in me has been a constant source of motivation. I am forever indebted to them for their understanding, sacrifices, and the endless love they have showered upon me.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ARI	Adjusted Rand Index
CPA	Consensus Pathway Analysis
DE	Differentially Expressed
DSCC	Disease Subtyping using Community detection from Consensus networks
ER	Estrogen Receptor
FCS	Functional Class Scoring
FDR	False Discovery Rate
FKNN	Fast k-nearest neighbor
GDC	Genomic Data Commons Data Portal
GEO	Gene Expression Omnibus
GO	Gene Ontology
GSA	Gene Set Analysis
GSEA	Gene Set Enrichment Analysis
HER2	Human Epidermal Growth Factor Receptor 2
HMP	Harmonic Mean P-value
KEGG	Kyoto Encyclopedia of Genes and Genomes

KS-test Kolmogorov–Smirnov test

METABRIC Molecular Taxonomy of Breast Cancer International Consortium

MGKA Multi-objective Genetic K-means clustering Algorithm

NNMF Non-negative Matrix Factorization

NSGA-II Non-dominated Sorting Genetic Algorithm

ORA Over-Representation Analysis

PAM Partition Around Medoids

PGSA Perturbation-based Gene Set Analysis

PR Progesterone Receptor

RSVD Randomized Singular Value Decomposition

SMRT Subtyping Multi-omics using a Randomized Transformation

SVD Singular Value Decomposition

TB Topology-Based

TCGA The Cancer Genome Atlas

TPM Transcripts Per Million

WCSS Within-Cluster Sum of Squares

Overview

The dissertation is divided into three parts. The first two parts describe the two distinctively different research areas, cancer subtyping and pathway analysis, and my contribution to the respective fields. The last part provides a summary of my past and future directions in bioinformatics research. The complete list of papers I published during my PhD study can be found in the last part, which consists of 18 journal articles [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18] and 8 conference papers [19, 20, 21, 22, 23, 24, 25, 26].

Part I: Integrative Cancer Subtyping describes computational methods and integrative techniques for cancer subtype discovery. This part consists of six chapters. The first two chapters provide an introduction and background information about cancer subtype discovery and current approaches. The last four chapters describe in-depth the four computational methods I developed during my PhD study. Each of the four chapters has its own sections for literature review, method development, experimental results, and conclusion.

Part II: Pathway Analysis describes the computational methods for signaling pathway analysis. This part consists of four chapters. The first chapter introduces computational methods in pathway analysis and describes the outstanding challenges faced by researchers working in the field. The next two chapters describe in-depth the methods I developed during my PhD study. The last chapter summarizes my contribution to this research area.

Part III: Summary and Future Research summarizes the contribution and research I have done in cancer subtyping and pathway analysis, together with my publications. It also describes the future research directions that I intend to take.

Part I

Integrative Cancer Subtyping

Chapter 1

Introduction

Cancer is a heterogeneous disease known to evolve through various pathways [27]. A tumor is a complex ecosystem comprising not only tumor cells but also various infiltrating cell types, such as endothelial, hematopoietic, stromal, and others, that can influence the overall function of the tumor [28, 29, 30]. Due to the diversity of mutations and molecular mechanisms, the behavior of individual tumors and their response to treatment can vary significantly [31]. In other words, even if it originates from the same tissue, the underlying molecular and cellular mechanisms can vary dramatically among patients. This has greatly contributed to the failure of many cancer therapies despite advances in cancer prognosis and treatment [32, 33], resulting in disease progression, recurrence, and reduced overall survival [27, 34]. For an accurate prognosis and improved treatment, the correct identification of cancer subtypes is essential.

Since the advent of gene expression profiling technologies, cancer subtyping has been a widely studied topic. The first successful subtyping of cancer was performed on breast cancer, which led to the identification of four main clinical subtypes based on the expression of the estrogen receptor (ER), progesterone receptor (PR), and human epidermal growth factor receptor 2 (HER2) [35]. These subtypes are known as Luminal A, Luminal B, HER2-enriched, and Basal-like. However, several studies have shown that there is still heterogeneity within each subtype, with patients that have very different survival rates and different responses to treatments [36, 37]. This suggests that further studies are needed to refine the current subtypes, even for a well-studied disease such as breast cancer.

With the advancement of high-throughput platforms with the ability to measure a wide range of molecular features from patients, including gene expression, DNA methylation, and

microRNA expression, it is now possible to identify cancer subtypes that share common molecular features on multiple molecular levels to further refine the current subtypes. The development of subtyping methods has also shifted toward multi-omics integration in order to differentiate between subtypes from a more holistic perspective [4, 38, 39, 40]. In addition, vast amounts of molecular data have accumulated in public repositories, including The Cancer Genome Atlas datasets (TCGA) [41], Genomic Data Commons Data Portal (GDC) [42], Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) [43], and UK Biobank [44]. This demands powerful yet fast analysis methods to leverage large multi-omics datasets for a more accurate subtype discovery.

In this chapter, we present the current status of intrinsic cancer subtyping, i.e., subtyping based on molecular features, which focuses on the integration of multi-omics. We also present the current challenges and limitations of the current subtyping methods and the motivation for a series of four studies that aim to address these challenges. In the first study, we developed a genetic algorithm-based clustering technique, called Multi-objective Genetic K-means clustering Algorithm (MGKA) to refine the k-means clustering algorithm and to automatically determine a suitable number of clusters, i.e., the number of subtypes [20]. In the second study, we introduce a subtyping framework using community detection on a consensus patient-patient network, named Disease Subtyping using Community detection from Consensus networks (DSCC), that is robust against noise with matrix factorization-based gene filtering [19]. In the third study, we introduce PINSPlus uses perturbation clustering to identify robust subtypes across multiple omics data types [4]. PINSPlus enhances the previous method PINS [38] by providing a more efficient and accurate subtyping analysis that can integrate data types with different numbers of patients. In the fourth study, we developed a novel subtyping framework, named Subtyping Multi-omics using a Randomized Transformation (SMRT), that is able to integrate a large number of omics data to subtype cancer patients while being robust against noise and missing data [2]. For each study, we compare the performance of the proposed method with the current state-of-the-art methods on a wide range of cancer datasets.

Chapter 2

Background

Cluster analysis has become a widely used tool for the exploration of high-dimensional data and cancer subtyping. Cluster analysis is an unsupervised approach to categorize objects without any predefined standards or knowledge for classification. In general, clustering methods aim to recognize the differences and similarities between objects so that the most similar objects will be grouped into one cluster and vice versa. Advances in high-throughput technologies, which produce a huge amount of genomic information, put a high demand on clustering methods that analyze gene expression data with disease subtypes and cell types discovery, two of the main application areas for clustering using genomic data.

Due to the noisy nature of genomic data and its undefined structures, it is impossible to find a universal clustering approach that works efficiently on a wide range of genomic data. Therefore, many clustering methods have been developed to tackle the clustering problems of genomic data. These methods can be categorized into two main categories: single-omic data clustering and multi-omics data clustering. In the following sections, we will review the current status of method development in both categories, the challenges and limitations of the current methods, and the motivation for the proposed methods in this dissertation.

2.1 Disease subtyping using single omic data

In principle, any clustering method can be utilized to perform disease subtyping by segregating patients into different groups. The accuracy of the discovered subtypes can be validated by comparing them with the clinical subtypes and correlating with the overall or disease-free

survival of the patients. Along with classical clustering methods such as k-means [45], partition around medoids [46], and hierarchical clustering, many other modern techniques have also been developed recently to tackle the clustering problems of genomic data [47]. The k-means clustering method, which is a broadly used and well-considered clustering technique, was found to be efficient for clustering cancer datasets using gene expression [47].

While classical clustering methods were widely used in clustering gene expression data, each of the methods has many drawbacks in their applications. For example, hierarchical clustering is sensitive to noise and requires the number of clusters to be provided as a parameter, which is not always available and is difficult to determine. K-means clustering also suffers from the same problems as hierarchical clustering and tends to produce non-optimal solutions when using random starting points, especially when the number of clusters is high. Since k-means received a lot of attention, several techniques have been introduced to refine the starting points for the k-means clustering method [48, 49, 50, 51, 52]. There are also many other studies that have tried to combine k-means with other heuristic algorithms to prevent k-means from converging into local minima including simulated annealing [53, 54] and genetic algorithm [55, 56, 57, 58, 59, 60, 61]. Partition around medoids (PAM), on the other hand, is less sensitive to noise and outliers when compared to k-means but still produces non-optimal solutions in most cases. It is important for a subtyping method to produce stable solutions so that the subtypes can be used in other downstream analyses to investigate the underlying mechanisms further. Unfortunately, k-means, PAM, and many other classical methods (e.g., Gaussian mixture model, Self-organizing maps) are unable to deal with the noisy nature of gene expression.

Many methods were also developed dedicated to cluster gene expression data. Some of the most widely used methods are Consensus Clustering (CC) [62] and its successor ConsensusClusterPlus (CC+) [63]. These algorithms cluster patients by iteratively subsampling both the patients and genes, then cluster the sampled data using a based clustering algorithm input by users. This process generates the consensus connectivity between every pair of patients. The algorithm then uses hierarchical clustering to cluster the connectivity matrix. The methods

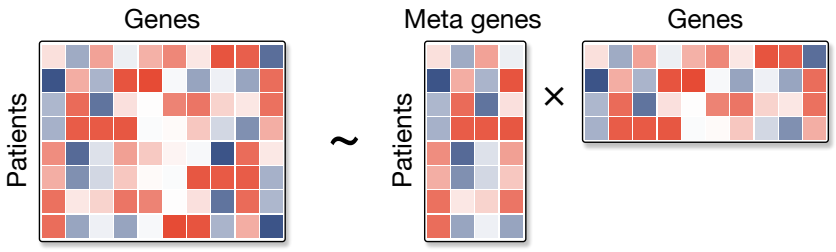

Figure 2.1: Abstract representation of matrix factorization.

allow users to examine the consensus distributions for different numbers of clusters to select the most optimal results.

Matrix factorization is another widely used class of methods that many methods have adopted [64, 65, 66, 67, 68]. This class of method decomposes the gene expression matrix into the multiplication of two lower-rank matrices that represent the compressed profile of genes and patients (Figure 2.1). These methods differ from each other in how they constrain the values of the compressed profile (e.g., binary, non-negative) and in how they optimize the factorization process (e.g., minimize l_2 norm, Kullback-Leibler distance, or maximize correntropy [69]). The dimension of the lower-rank matrices represents the number of desired clusters or "meta genes", and the compressed profile of patients shows the expression of the patients for each meta gene. The final cluster assignments can be determined by further applying a clustering algorithm on the meta genes or by directly examining the meta genes of the patients.

Many other methods have used genetic algorithms in the hope of finding the optimal clusters from gene expression data [70, 71, 72, 73, 74]. Briefly, a genetic algorithm is an algorithm that generates a population of solutions and then iteratively crossovers and mutates them in hopes of making a better generation after every iteration. The quality of a solution is evaluated by a fitness function. For clustering, methods applied genetic algorithms [75, 76, 77, 78, 56, 58, 61] by encoding the candidates for the optimal cluster assignments (label-based) or the centrals of the clusters (i.e., medoid or centroid) as the population. The latter representation is more efficient in terms of the size of the search space compared to the former one, especially when the number of clusters increases. However, the benefits of each

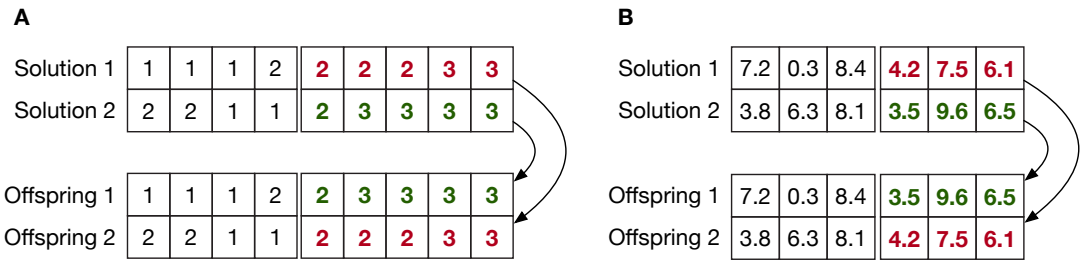

Figure 2.2: One point crossover for (A) integer encoding (label-based) and (B) real-number encoding (centroid-based).

representation are difficult to evaluate because the performance also heavily depends on the design of the fitness function and the crossover function. Figure 2.2 shows the original one-point crossover operation on the label-based representation and centroid-based presentation.

K-means is also used in several methods to refine the genetic algorithm generated results [78, 61, 76]. With each generation, one or multiple steps of k-means are applied to certain solutions during the mutation process or to all individuals in the population. This operation is especially helpful in urging the genetic algorithm to converge and in refining the ultimate solutions. However, it can also trap solutions at local optima.

Using these clustering approaches, however, only allows us to perform subtyping on individual data types. Although one can concatenate the feature space of different data types for the clustering input, it is advised to against this approach since different data types have different scales and different numbers of features.

2.2 Multi-omics integration methods

Several methods have been developed based on multi-omics data integrations, and the number is growing day by day. While with the single-omic data type, most methods were developed for a more general clustering purpose and adopted for gene expression clustering, most of the methods described in this section were originally developed dedicated to discovering disease subtypes.

Current approaches for multi-omics integration and cancer subtyping can be categorized into four categories based on their integration strategy. The first strategy is to concatenate different types of data into a single matrix and then partition the patients using the concatenated

data. For example, users can normalize and concatenate multiple data types (e.g., mRNA, methylation, miRNA, etc.) into one single matrix and then apply well-known methods developed for single-omics analysis, such as ConsensusClusterPlus [63], to determine the subtypes. Such approaches are simple and computationally efficient. In principle, any clustering method can be utilized to perform such a task as long as the number of samples is the same across different data types and the method can handle the large feature space. It is apparent that such strategies do not account for data heterogeneity, e.g., different data types might have different scales and dimensions and might require different normalization procedures.

The second strategy is to model the multi-omics data as a mixture of statistical models. Methods in this category include LRACluster [79], rMKL-LPP [80], iClusterPlus [81], iClusterBayes [82], OTRIMLE [83], SBC [84], BCC [85], MID [86], JIVE [87], MCIA [88], moCluster [89], and sMBPLS [90]. These methods typically maximize a joint likelihood function to determine the model parameters and the subtypes. These models usually assume the distribution of the inputs or require users to input the distributions (e.g., Gaussian distribution, Bernoulli distribution, Poisson distribution). This might also limit the applications of these methods on new data. Even for the same data types, different data normalization and transformation methods applied to the same data can lead to different distributions. This is a common problem when dealing with multi-omics data, especially with expression data when the data are generated from different platforms. Though statistically sound when the assumptions are met, the methods in this category need to estimate a large number of parameters that often lead to overfitting. A more serious problem is that these methods are not scalable to the whole genome scale, as any omic data can have tens of thousands to millions of features. Therefore, an added step of gene filtering or data transformation is often applied before the statistical analysis. However, this step can lead to the loss of important information and the introduction of bias to the final results. For example, to have a reasonable runtime with the iClusterPlus method, users were recommended to use the top 2,000 most variable features [81]. With more than 450,000 features in the Methylation 450K array, this means that more than 99% of the features were discarded.

The third strategy is to project all data types into a joint latent space. A common technique used for this strategy is non-negative matrix factorization. Methods in this category include MvNMF [91], MultiNMF [92], IntNMF [93], iNMF [94], jointNMF [95]. Another method is MCCA [96], which performs correlation analysis and then concatenates the correlation matrices into one single matrix. After projecting the data onto a joint space, cluster analysis is performed to determine the final subtypes. While methods in this category do not assume the distribution of the data, they essentially model the data as a mixture of Gaussian distributions in the latent space as the final clustering is performed using k-means or hierarchical clustering. As the methods often treat each data type equally, i.e., the loss function is the same for all data types, data types with more features can dominate the final clustering. This can be mitigated by applying a weighting scheme to the loss function, but the weighting scheme is often difficult to determine. Another approach is to use gene filtering to reduce the dimension of the data to similar scales across different data types. This can also help with the computational complexity of the methods, as methods in this category often have excessive computational complexity and cannot be applied on the whole genome scale. However, similar to the second strategy, gene filtering can cause information loss and bias in the final results.

In the fourth category, similarly-based methods usually construct the pairwise connectivity between patients for each data type, and a similarity matrix is formed by merging all connectivity matrices. The connectivity indicates how often the patients are grouped. Similarity-based algorithms are often used on the similarity matrix to discover the subtypes. The computational complexity mostly depends on the number of patients or features. SNF [97], PSDF [98], PFA [99], IS-Kmeans [100], NEMO [101], PINS [38, 4], SCFA [14], and CIMLR [31] are some methods of this category. SNF forms a similarity network by integrating multi-omics data from the connectivity matrices for each data type and uses spectral clustering to partition the network. NEMO builds an inter-patient similarity matrix for each data type and integrates them into a similarity network. It also uses spectral clustering to cluster the network. NEMO is also able to work on partial datasets (i.e., datasets where each patient has a different number of data types). PINS is robust against data perturbation and identifies how frequently the patients are grouped with each other across multiple data types when data is perturbed. CIMLR

creates the similarity matrix using multiple Gaussian kernels per data type and uses k-means to discover the subtypes. SCFA filters the unnecessary gene using an autoencoder, reduces the dimension by factor analysis, and finally uses a consensus ensemble to find subtypes shared across all lower representations. The similarity-based methods do not assume the distribution of the data, are often computationally efficient, and can easily support different omic types. As a result, there are an increasing number of methods in this category, and the number is still growing. Most methods in this category, however, are not able to handle partial datasets, i.e., datasets where each patient has a different number of data types. It is required to keep only patients with all data types of interest before the analysis. As a result, the number of patients in the analysis is greatly reduced, and the statistical significance of the discovered subtypes also decreases.

Chapter 3

MGKA: A genetic algorithm-based clustering technique for genomic data

This chapter is based on the following publication: ***Hung Nguyen****, Sushil J. Louis, and Tin Nguyen. MGKA: A genetic algorithm-based clustering technique for genomic data. IEEE Congress on Evolutionary Computation (CEC). 2019. DOI: 10.1109/CEC.2019.8790225*

While it is true that disease subtyping has adopted multi-omics data integration, the majority of the methods in the literature are still based on single omic data types. This is because many cohort studies have only one type of omic data available, especially for older datasets or datasets with large numbers of patients. Single-omic data clustering methods are still widely used in practice and are still the first choice for many researchers. It is still the building block for multi-omics data clustering methods, and can easily be applied to perform clustering on data from other fields, such as single-cell RNA-seq data, social network data, and image data.

In this chapter, we present a genetic algorithm-based clustering technique, called Multi-objective Genetic K-means clustering Algorithm (MGKA), to refine the k-means clustering algorithm and to automatically determine a suitable number of clusters, i.e., the number of subtypes. This chapter is organized as follows. In Section 3.1, we present a more detailed review of genetic algorithm-based clustering methods. In Section 3.2, we present the proposed MGKA method. In Section 3.3, we present the experimental results of the proposed method on simulated datasets and on real cancer datasets. We also extend the analysis to single-cell RNA-seq data to show the general applicability of the proposed method. In Section 3.4, we conclude the chapter.

3.1 Related work in genetic clustering algorithms

Along with classical clustering methods such as k-means [45], partition around medoids [46], and hierarchical clustering, many other modern techniques have been developed recently to tackle the clustering problems of genomic data [47]. The k-means clustering method, which is a broadly used and well-considered clustering technique, was found to be efficient for clustering cancer datasets [47]. The k-means clustering technique is simple to use, easy to implement, and one of the most straightforward algorithms to understand. With a predefined number of clusters k, the algorithm tries to find k centroids in the multiple-dimensional space from a set of random centers so that every data point is allocated to an adjacent centroid. A detailed discussion about the algorithm can be found in [102]. However, the k-means algorithm is known to be sensitive to initial conditions and does not guarantee to produce global optimal clusters. The clustering results heavily depend on the starting center points, which are (usually) randomly initialized. Therefore, the algorithm is susceptible to converge into a local optimum. Furthermore, the number of clusters must be given as an input parameter for the k-means clustering technique. Without any prior knowledge of the data, determining the appropriate number of clusters is considered a difficult task.

A few efforts have been accounted for to take care of the clustering initialization problem. The most common and naive technique is to attempt the k-means algorithm multiple times with different initial seeds and gather the best result. However, the best-obtained solution from this stochastic procedure does not often produce globally optimal clusters. Note that finding globally optimal clusters is known to be an NP-hard problem. Several techniques have been introduced to refine the starting points for the k-means clustering method [48, 49, 50, 51, 52]. On the other hand, many other studies have tried to combine k-means with other heuristic algorithms to prevent k-means from converging into local minima, including simulated annealing [53, 54] and genetic algorithm [55].

The genetic algorithm (GA) is a powerful technique for optimization problems based on natural selection and genetics. GAs have been applied to many function optimization problems

and have been shown to be good at finding optimal and near-optimal solutions. The basic methods of the genetic algorithm are designed to reproduce processes in normal systems necessary for evolution based on the principle of survival of the fittest.

Although the initial population is randomized, the GA is by no means random. It chronologically directs the population in the search space by probabilistic applying genetic operators, including selection, crossover, and mutation. In general, the selection operator selects individuals from the current population for the next population with a probability proportional to the individual's fitness relative to the fitness of the rest of the population. Crossover operates on two individuals (parents) to produce two new individuals (offspring) inheriting some of the attributes from their parents. The mutation operator changes the genomic structure of an individual at some point in the hope that the mutated individual will help maintain diversity for crossover and selection to exploit. Depending on the specificity of the problem, the selection, the crossover, and the mutation procedures in the GA may vary.

Several studies address genetic algorithms to solve clustering problems using label-based representation for solution [75, 56, 58, 61]. Label-based representation uses integer encoding to present cluster membership. For example, providing k number of clusters (e.g., $k = 3$), the integer vector $[111222233]$ indicates that the first three data points belong to the cluster $\#1$, the next four data points belong to cluster $\#2$, and the last two data points belong to cluster $\#3$. This encoding is, however, redundant. For example, the cluster membership integer vector $[111222233]$ is equivalent to $[222111133]$. With the same solution, there will be $k!$ different encodings. Therefore, the size of the search space for the genetic algorithm significantly increases when the number of clusters k increases, which may reduce the efficiency of the genetic algorithm.

Compared to label-based representation, medoid-based and centroid-based representations, which encode only the centers of the clusters, are more efficient in terms of the size of the search space. However, the ultimate benefits of each representation are still hard to evaluate and compare because performance also greatly depends on the design of the fitness function. Several methods make use of medoid-based representation using integer encoding to encode the solution [103, 104]. The previous cluster membership example $[111222233]$ can

be encoded as $[2\,5\,8]$ in k-medoids approach in which the second, fifth, and eighth data points are three centers represented for three clusters. Other data points are then assigned to each cluster using these centers. Centroid-based representation, on the other hand, uses real-number encoding to represent the center of clusters. Unlike medoid-based representation, which uses data points in the input data as cluster centers, cluster centers in centroid-based representation can be any point in the multi-dimensional space. Therefore, a solution is now represented by a set of coordinates. For example, the real-number vector $[7.2\ 0.3\ 8.4\ 4.2\ 7.5\ 6.1]$ illustrates three cluster centers $A(7.2, 0.3)$, $B(8.4, 4.2)$, and $C(7.5, 6.1)$. This representation is adopted by Maulik and Bandyopadhyay [78] and several other papers [77, 76, 105].

Traditional genetic crossover is strongly adopted in genetic-based clustering algorithms. Many studies applied one-point crossover to produce offspring for both integer encoding solutions and real-number encoding solutions [75, 76, 77, 78]. Figure 3.1 describes one-point crossover for integer encoding (A) and real-number encoding (B). However, the naive one-point crossover can produce invalid offspring, as described in Figure 3.1C. On the other hand, one-point crossover on a real-number encoding can be very destructive to the population since it can generate significantly different offspring compared to its parents. In high-dimension data, this operation tends to swap the centers between parents rather than moving them in the high-dimensional space. Crossover can also be omitted, such as in Krishna and Murty's method [56].

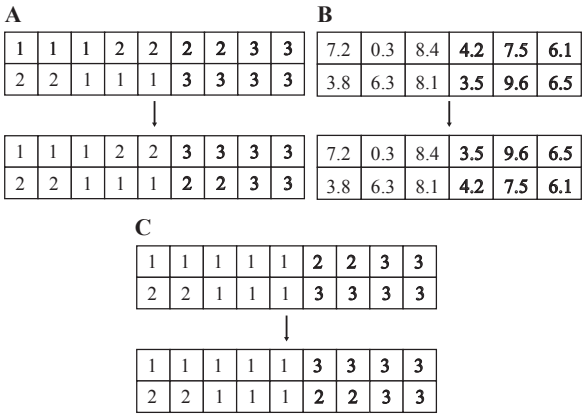

Figure 3.1: One point crossover for (A) integer encoding and (B) real-number encoding. (C) Invalid offspring from one point crossover for integer encoding.

With label-based representation, mutation can be as simple as assigning a data point to a random cluster. However, it can generate invalid solutions. Krishna and Murty [56] design a mutation operator that assigns a new cluster for a data point based on the distances of the cluster centroids from the corresponding data point. The cluster that has the centroid closer to the data point will have a higher probability of being assigned to that data point. This mutation principle is also adopted by Lu et al. [58, 61]; however, such methods have been found to create empty clusters. Other papers using real-number encoding [76, 77, 78] operate mutation by slightly modifying the centroids. By modifying the centroids, this mutation may change the membership of some data points in relation to the clusters represented by the solution. It can also shake the centers out of the local optimum.

K-means is also used in several methods to refine the genetic algorithm generated results [78, 61, 76]. With each generation, one or multiple steps of k-means are applied to certain solutions during the mutation process or to all individuals in the population. This operation is especially helpful in urging the genetic algorithm to converge and in refining the ultimate solutions. However, it can also trap solutions at local optima.

3.2 Multi-objective genetic k-means clustering algorithm

Many studies have attempted to apply GA to refine the k-means algorithm. However, many of them omit the crossover procedure and greatly depend on the selection and mutation operator. On the other hand, none of those methods has taken into account the problem of choosing the appropriate number of clusters for the k-means algorithm. In this study, we make use of real-number encoding to encode the cluster centers in a way that makes the number of clusters encoded by a solution dynamic. We use simulated binary crossover [106], which applies crossover for every dimension of the centers in the solution. This crossover operation will generate offspring close to their parents. Besides adding noise to the cluster centers to avoid the convergence of the genetic algorithm to a local optimum, our mutation operation can also change the number of clusters that a solution represents. We also use the k-means operator to refine the solutions. In the following sections, we describe the encoding of the solution, the genetic operators, and the fitness function of the proposed method.

3.2.1 Chromosome encoding

In our encoding, each solution (chromosome) has two encoded regions, as shown in Figure 3.2. The first region encodes the status of each center, which is either active (1) or disabled (0). The second region encodes the coordinates of each center. Figure 3.2 represents a solution for two-dimensional input data where the number of clusters is two. In this example, the maximum number of clusters is three. However, this number can be higher. The number of coordinates for each center depends on the number of dimensions in the input data.

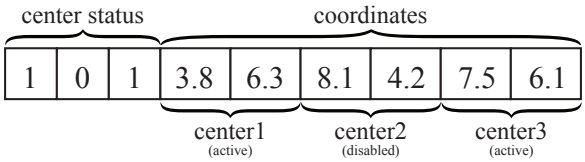

Figure 3.2: Chromosome encoding of a two-cluster solution for two-dimension data with the maximum cluster it can encode is three.

This chromosome encoding allows us to hold solutions with different numbers of clusters in the same population. It also allows us to change the number of clusters of a solution through mutation, which prevents solutions with the same number of clusters from dominating the population.

The population is randomly initialized by selecting random data points and assigning their coordinates to cluster centers. With predefined maximum cluster numbers $kMax$ from users, each number of clusters k is initialized with the same number of solutions.

3.2.2 The fitness functions

The fitness of individuals is evaluated using three different criteria including: i) within-cluster sum of squares, ii) Davies and Bouldin index [107], and iii) Silhouette index [108]. We describe each of these in turn below.

Within cluster sum of squares (WCSS) is the objective function of the original k-means.z Denoting k as the number of clusters, $\{c_i, i \in [1..k]\}$ as the cluster centers, and $\{C_i, i \in [1..k]\}$ as the k clusters (each cluster consists of many data points), the within-cluster sum of squares is defined as:

$$WCSS = \sum_{i=1}^{k} \sum_{j \in C_i} \|x_j - c_i\|^2$$

where $\|x_j - c_i\|^2$ is the Euclidean squared distance between data point x_j and center c_i. A better solution will have a smaller $WCSS$ value.

Davies and Bouldin (DB) index is a function of the sum of within-cluster scatter to between-cluster separation. A better solution will have a smaller $DB(k)$ value. DB index is calculated as follows:

$$DB(k) = \frac{1}{k} \sum_{i=1}^{k} \underset{i \neq j}{max}(\frac{\delta_i + \delta_j}{d_{ij}})$$

where

- k is the number of clusters,
- i, j are the i^{th} and j^{th} cluster respectively,
- d_{ij} is the distance between centers c_i and c_j,
- δ_i and δ_j are the dispersion measure of a cluster C_i and C_j, respectively. For example, C_i is the standard deviation of the distance of data points in cluster C_i to the center of this cluster c_i.

Silhouette index (SI) measures how similar an object is to its own cluster (cohesion) compared to other clusters (separation). A better solution will have a higher SI value. The silhouette index is computed as follows:

$$SI = \frac{\sum_{i=1}^{n} \frac{b(i) - a(i)}{max\{a(i); b(i)\}}}{n}$$

where

- $a(i) = \frac{\sum_{j \in \{C_r \setminus i\}} d_{ij}}{n_r - 1}$ is the average dissimilarity of the i^{th} object to all other objects of cluster C_r,

- $b(i) = \underset{s \neq r}{min}\{d_{iC_s}\}$, in which $d_{iC_s} = \frac{\sum j \in C_s d_{ij}}{n_s}$ is the average dissimilarity of the i^{th} object to all objects of cluster C_s.

By using all of the three metrics (WCSS, DB, and SI), the fitness function will evaluate how similar each member is in the same cluster and how well the clusters are separated.

3.2.3 The crossover operator

The crossover operator is performed by using the simulated binary crossover proposed by Agrawal, R. B. et al. [106] on parents that have the same number of clusters. The crossover operator is performed by using the simulated binary crossover proposed by Agrawal et al. [106]. The crossover procedure is described in Figure 3.3. For each pair of parents ($Parent_1$ and $Parent_2$) that have the same number of clusters selected randomly from the population, the simulated crossover is applied to each coordinate of the centers in $Parent_1$ with the corresponding coordinate of the centers in $Parent_2$.

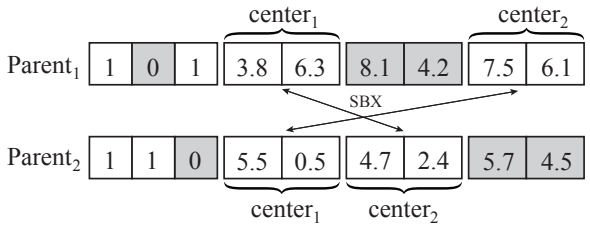

Figure 3.3: Crossover procedure between two parents. Simulated binary crossover is applied to (1) each coordinate of $center_1$ of $Parent_1$ with the corresponding coordinate of $center_2$ of $Parent_2$, and (2) each coordinate of $center_2$ of $Parent_1$ with the corresponding coordinate of $center_1$ of $Parent_2$.

First, the Euclidean distance is calculated between any centers of $Parent_1$ and $Parent_2$. A simulated binary crossover is then applied to each coordinate of the corresponding dimension of the closest centers between two parents. These centers are then removed from the crossover center list. The procedure is applied to the rest of the centers of the two parents until no center is left. The results of another example of the crossover operator can be seen in Figure 3.4.

3.2.4 The mutation operator

Mutation shakes the centers out of a local optimum and moves them, hopefully, toward the global optimum. Within the mutation operator, there are two functions that can be applied to

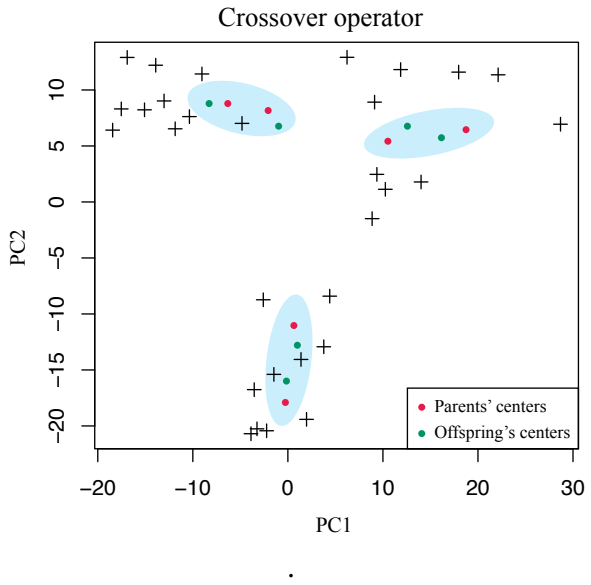

Figure 3.4: Offspring resulted from simulated binary crossover. Red dots represent the centers of two parents, and green dots represent the centers of two corresponding offspring

each solution, including (1) adding noise to the centers of each cluster and (2) changing the number of clusters.

Noise is added to the centers of each cluster using Gaussian noise. The noise added to the data will have the variance equal to the variance of the data. By setting the variance of the added noise equal to the median variance of the data, we aim to sufficiently shake the centers out of local optima. If the added noise is considerably higher, the new centers will be moved further from the original points, which can destroy the solution. On the other hand, if the noise is low, the new centers will only move close to the original centers, which can result in them being trapped in a local optimum.

The mutation operator can also change the number of clusters and solutions by activating or disabling a center. Activating a center will select a random data point and add its coordinates to a new center. Disabling a center will select a random center in the solution and mark it as disabled.

3.2.5 The k-means operator

The k-means operator is applied to speed up the convergence of the algorithm by applying one step of the k-means algorithm to the solution. For each generation, the k-means operator is applied with a predefined probability by the users for each solution. The procedure starts with

assigning each data point to the closest center in the solution; the centers are then adjusted using the mean of the data points assigned to that center.

The k-means operator, however, can produce an illegal solution with empty clusters. If the adjusted solution contains empty clusters, a random data point will replace the center with empty members. The k-means operator is then re-applied until a valid solution is produced.

3.2.6 The selection operator

The goal of the selection operator is to find the Pareto front of the three objective functions. In this paper, we make use of the selection procedure proposed by Deb, K. et al. [109]: Non-dominated Sorting Genetic Algorithm (NSGA-II). The principle of the selection is to arrange the population into a hierarchy of non-dominated Pareto fronts and use a crowding distance to prevent solutions from concentrating in the region at the level of the Pareto fronts (Figure 3.5). The detailed implementation of the algorithm is described in [109].

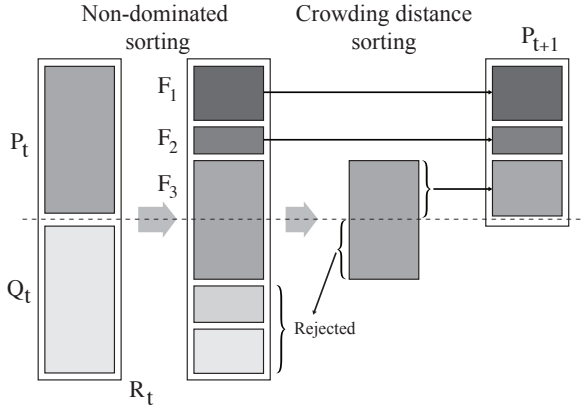

Figure 3.5: Non-dominated Sorting Genetic Algorithm (NSGA-II) where t is the population generation, P is the parents, Q is the offspring, and F_i is the Pareto front level i^{th}.

3.2.7 Evaluating the ultimate solution

Although NSGAII was designed to produce a dispersed Pareto front, and in practice, we can present the entire Pareto front to a domain expert user to choose from, we may still wish to identify an "ultimate" solution as the result of our algorithm. We start by considering all individuals in the final Pareto front. We then select the best solution for each objective. If the best solutions for two index values are different, each solution will be ranked based on its other

index value compared to other solutions in our Pareto set. The solution that has a better rank will be extracted as the ultimate solution.

3.3 Experimental results

3.3.1 Results on simulation

We first validate the framework from a theoretical perspective by comparing the new method with the original k-means. In this section, we compare the performance of MGKA with k-means on generated datasets with a large number of clusters. It is known that k-means does not produce a global optimum. Therefore, we run k-means multiple times in order to obtain results that are at least close to the global optimum. Here, we set the number of times we run k-means equal to the population size of MGKA, which is 50. The simulation generates datasets with clusters of 10 to 15; each cluster is well separated and has ten members. The landscape of the simulated data with $k = 10$ is described in Figure 3.6. We use the $kmeans$ function in the $stats$ package, R programming language, to obtain the clustering result from the k-means algorithm.

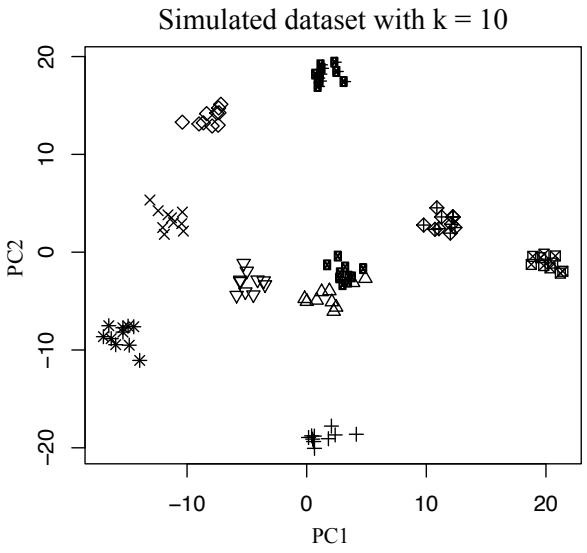

Figure 3.6: The landscape of simulated dataset with the number of cluster is ten. Each cluster is well separated to each other and has ten members.

The average result of 30 runs for each k is represented in Table 3.1. We use the within-cluster sum of square errors and the Adjusted Rand Index (ARI) to compare the results between the two algorithms. Table 3.1 shows that MGKA outperforms k-means in all of the datasets.

Table 3.1: Within cluster sum of square errors and Adjust Random Index (ARI) of clustering result produced by MGKA and k-means with restarts.

#k	#Samples	WithinSS		ARI	
		MGKA	k-means	MGKA	k-means
10	100	457.237	782.051	1	0.963
11	110	461.326	996.554	1	0.954
12	120	520.686	913.989	1	0.939
13	130	598.247	910.19	0.993	0.914
14	140	547.731	1136.477	1	0.931
15	150	630.188	1074.967	1	0.929

Adjust Rand Index (ARI) values for clusters produced by MGKA in all datasets show that MGKA can easily achieve the global optima in all simulated datasets. K-means, on the other hand, produces sub-optimal solutions most of the time. The average ARI of 30 runs also shows that MGKA is much more stable compared to k-means. The within-cluster sum of squares shows significant differences among clusters produced by MKGA and k-means. The results from k-means are also too far away from optimal solutions.

3.3.2 Results on cancer omics data

Table 3.2: Description of the eight mRNA datasets used in our analysis. The top five datasets were downloaded from the Gene Expression Omnibus. The bottom three datasets were downloaded from the Broad Institute website.

Datasets	#Class	#Sample	#Feature	Platform	Description
GSE10245 [110]	2	58	19851	hgu133plus2	40 adenocarcinomas and 18 squamous cell carcinomas
GSE19188 [111]	3	91	19851	hgu133plus2	45 adenocarcinomas, 19 large cell carcinomas, and 27 squamous cell carcinomas
GSE43580 [112]	2	150	19851	hgu133plus2	77 adenocarcinomas and 73 squamous cell carcinomas
GSE14924 [113]	2	20	19851	hgu133plus2	10 acute myeloid leukemia CD4 T cell and 10 CD8 T cell
GSE15061 [114]	2	366	19851	hgu133plus2	202 acute myeloid leukemia samples and 164 myelodyplastic syndrome samples
Lung2001 [115]	4	237	8641	hgu95a	190 adenocarcinomas, 21 squamous cell carcinomas, 20 carcinoid, and 6 small-cell lung carcinomas
AML2004 [116, 117]	3	38	5000	hgu6800	11 acute myeloid leukemia, 19 acute lymphoblastic leukemia B cell, and 8 T cell
Brain2002 [118]	5	42	5299	hgu6800	10 meduloblastomas, 10 malignant gliolas, 10 atypical teratoid/rhaboid tumors, 4 normal cerebellums, and 8 primitive neuroectodermal tumors

Here, we demonstrate the application of MGKA in the context of cancer subtyping using multi-omics data. In order to assess the performance of MGKA, we compare the results of MGKA with those of widely used methods in this field, including Consensus Clustering

(CC) [62] – a resampling-based approach, Similarity Network Fusion (SNF) [97] – a graph-theoretical approach and iClusterPlus [81] – a mixture model approach. With CC, SNF, and iClusterPlus, we use the default parameter settings. The parameters for MGKA after this section are: population size = 20, the number of generations = 20, crossover probability = 1, mutation probability = 0.01, and k-means operator probability = 0.5.

First, we compare the four methods using eight mRNA gene expression datasets with known disease subtypes. The 5 datasets with accession id GSE10245, GSE19188, GSE43580, GSE15061, and GSE14924 were downloaded from Gene Expression Omnibu (`www.ncbi.nlm.nih.gov/geo/`). The other three datasets were downloaded from the Broad Institute: Lung200 (`www.broadinstitute.org/mpr/lung/`), AML200 (`www.broadinstitute.org/cancer/pub/nmf`), and Brain200 (`www.broadinstitute.org/MPR/CNS/`). Details of the eight datasets are described in Table 3.2. The results of clustering for eight mRNA datasets are represented in Tabl 3.3. We use the Adjusted Rand Index (ARI) to assess the performance of the resulting subtypes. Among the eight datasets that we tested, MGKA outperforms other methods in six methods. SNF and iClusterPlus, however, crashed with GSE14924 and AML2004 and are represented with *NA* in the table.

Table 3.3: The performance of MGKA, Consensus Clustering (CC), Similarity Network Fusion (SNF), and iClusterPlus in discovering subtypes from gene expression data. For each dataset (row), cells highlighted in green have the highest Adjusted Rand Index (ARI).

Dataset	Samples	#Class	MGKA	CC	SNF	iCluster+
GSE10245	58	2	0.80	0.32	0.38	0.22
GSE19188	91	3	0.84	0.6	0.12	0.19
GSE43580	150	2	0.44	0.37	0.15	0.21
GSE15061	366	2	0.78	0.43	0.05	0.15
GSE14924	20	2	1.00	0.25	NA	0.73
Lung2001	237	4	0.54	0.11	0.28	0.11
AML2004	38	3	0.41	0.56	0.17	NA
Brain2002	42	5	0.15	0.46	0.13	0.32

Secondly, we compare the four methods using DNA methylation datasets from The Cancer Genome Atlas (TCGA). In the comparison, we use eight datasets downloaded from the TCGA website (cancergenome.nih.gov and firebrowse.org). Eight datasets include Glioblastoma multiforme(GBM), Thymoma (THYM), Glioma (GBMLGG), Kidney renal papillary cell

carcinoma (KIRP), Kidney Chromophobe (KICH), Uveal Melanoma (UVM), Pancreatic adenocarcinoma (PAAD), and Adrenocortical carcinoma(ACC). These datasets, however, do not contain subtypes for each disease. Instead, with known survival outcomes, we use Cox regression to assess the survival difference of the discovered subtypes. The Cox p-values of the subtypes discovered by each of the four approaches are presented in table 3.4. Again, among eight datasets, MGKA outperforms other methods in six datasets. Moreover, while MGKA can discover subtypes with significant cox-p value (at the threshold of 5%) for all datasets, CC, SNF, and iClusterPlus can only discover subtypes with significant cox-p value for three, seven, and five datasets respectively.

Table 3.4: The performance of MGKA, Consensus Clustering (CC), Similarity Network Fusion (SNF), and iClusterPlus in discovering subtypes from DNA methylation data. Cells highlighted in yellow have significant Cox p-values at the threshold of 5%. For each dataset (row), cells highlighted in green have the most significant Cox p-value.

Dataset	Samples	MGKA	CC	SNF	iCluster+
GBM	273	1.2e−4	0.075	0.017	0.103
THYM	119	0.006	0.053	0.04	0.068
GBMLGG	510	3.3e−16	3e−9	1.9e−12	5.4e−14
KIRP	271	5.1e−18	0.299	2.8e−13	0.013
KICH	65	1e−4	0.88	1e−4	0.788
UVM	80	7.1e−4	9.8e−4	0.005	0.003
PAAD	178	0.002	6.6e−4	0.346	3.8e−4
ACC	79	6.2e−4	0.06	0.047	6.6e−5

3.3.3 Results on single-cell transcriptomics data

We also test our method on four different single-cell datasets with known cell types (Table 3.5). Yan's dataset contains 90 human embryo samples in six different stages. Goolam's and Deng's datasets contain mouse embryo samples in different stages. Pollen's dataset contains 301 samples of different human tissues. The references for each dataset are given in Table 3.5. We compare our method with SC3[119] method - a consensus clustering method of single-cell RNA-seq data, and SEURAT[120] - a graph-based clustering approach for single-cell RNA-seq data. Table 3.5 shows the ARI values obtained by MGKA, SC3, and SEURAT on those four datasets. MGKA produces the best clusters in three out of four tested datasets.

Table 3.5: The performance of MGKA, SC3, and SEURAT in discovering cell types from gene expression data. For each dataset (row), cells highlighted in green have the highest Adjusted Rand Index (ARI). MGKA produces clusters with highest ARI value for three out of four datasets.

Dataset	Samples	#Class	MGKA	SC3	SEURAT
Yan (GSE36552)[121]	90	6	0.67	0.63	0.53
Goolam (E-MTAB-3321)[122]	124	5	0.72	0.63	0.57
Deng (GSE45719)[123]	268	6	0.60	0.55	0.51
Pollen (SRP041736)[124]	301	11	0.88	0.93	0.70

3.4 Conclusion (MGKA)

K-means clustering is a simple, fast, and unsupervised approach. However, it suffers from some limitations, such as the initial centroids problem and the selection of the appropriate number of clusters. This study describes and evaluates a new approach that uses an evolutionary multi-objective algorithm to find a set of Pareto optimal solutions along three measures of cluster goodness. A new representation directly addresses the initial centroid problem, and the non-dominated sorting genetic algorithm maintains a population with a diverse number of high-performing clusters. That is, while many current approaches integrate genetic algorithms with k-means to find the global optimum for a fixed number of clusters, our method, MGKA, is able to maintain and evaluate solutions with different numbers of clusters at the same time. By using simulated binary crossover, our crossover operator is less destructive compared to naive one-point crossover and generates offspring close to the parents rather than exchanging dataset members or center coordinates.

The multi-objective genetic algorithm allows us to optimize the solution with different cluster validity indexes so that, at the same time, we can also evaluate the appropriate number of clusters. By using the Davies & Bouldin index and Silhouette index, the best solutions will have the most similar members in the same cluster and have well-separated clusters. Our experiment on different simulated datasets shows that MGKA is better than naive k-means in finding the global optimum. Other experiments on 16 disease datasets and five single-cell datasets indicate that MGKA outperforms other state-of-the-art algorithms in discovering disease subtypes. We also demonstrate that MGKA can be used to discover cell types from single-cell RNA-seq data and displays a better performance compared to SC3 and SEURAT, which are specifically

designed for single-cell RNA-seq data. This provides strong evidence of the viability of our approach for clustering applications, especially in the biomedical domain.

Chapter 4

DSCC: Disease subtyping using community detection from consensus networks

This chapter is based on the following publication: ***Hung Nguyen****, Bang Tran, Duc Tran, Quang-Huy Nguyen, Duc-Hau Le, and Tin Nguyen. Disease subtyping using community detection from consensus networks. 12th International Conference on Knowledge and Systems Engineering (KSE). 2020. DOI: 10.1109/KSE50997.2020.9287843*

In the previous chapter, we introduced a general clustering method that uses a multi-objective genetic algorithm to find a better solution for k-means clustering. The method is suitable for clustering a single type of data. In this chapter, we focus on the problem of disease subtyping using multi-omics data. Here, we introduce DSCC (Disease Subtyping using Community detection from Consensus networks), which exploits the local relationships between patients from each data type to build a consensus network from patient connectivities. It then uses a community detection technique to discover different groups within patients that have significantly different survival profiles. In an extensive analysis using simulation studies and 5,782 real patients related to 20 cancer datasets from The Cancer Genome Atlas, we demonstrate that DSCC is robust against noise and outperforms state-of-the-art methods in identifying known patient classes and novel subtypes with significantly different survival profiles.

This chapter is divided into three main sections. In the first section, we describe the methodology of DSCC. In the second section, we present the results of DSCC on both simulated and real data. In the third section, we discuss the results and the potential of DSCC in the context of disease subtyping.

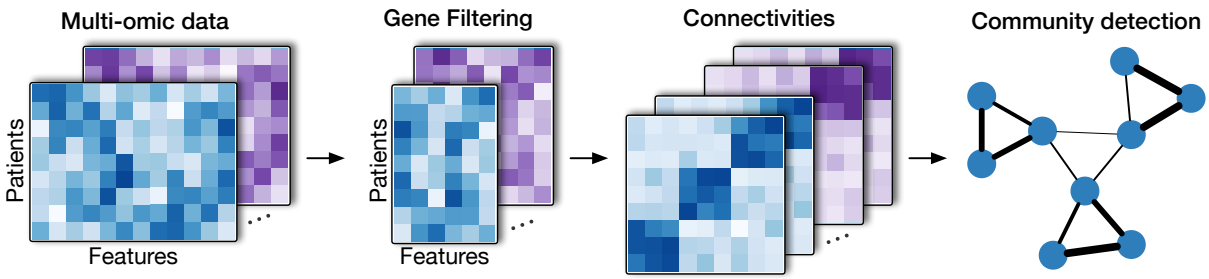

Figure 4.1: The overall workflow of DSCC. The method consists of three main steps: i) gene filtering using non-negative matrix factorization, ii) building patients connectivities using k-means with different numbers of clusters, and iii) clustering using community detection.

4.1 Methodology

Figure 4.1 shows the overall workflow of DSCC. The method requires a list of data matrices (mRNA, methylation, miRNA, etc.). In each matrix, rows represent samples/patients, and columns represent genes/features. For each matrix, the method first applies gene filtering using non-negative matrix factorization and then builds connectivities between patients using k-means clustering. Finally, the method applies community detection on the combined connectivity using Louvain modularity [125] to cluster patients.

4.1.1 Gene filtering using Non-negative Matrix Factorization

Our hypothesis is that although the total number of features in omics data is large (e.g. $\sim$20k for mRNA data), only a subset of them truly differentiates among cancer subtypes. Therefore, we first focus on filtering out genes that are not likely to play a major role in subtyping. Figure 4.2 shows the workflow of our gene filtering approach using 1-factor Non-Negative Matrix Factorization (NNMF). Briefly, Matrix Factorization is a technique that decomposes a matrix into the product of two lower dimensionality matrices:

$$V = W \times H + E$$

where in the context of this article:

- V is a matrix of size $p \times g$ (the original omic data, e.g. gene expression matrix), in which p is the number of patients and g is the number of genes;

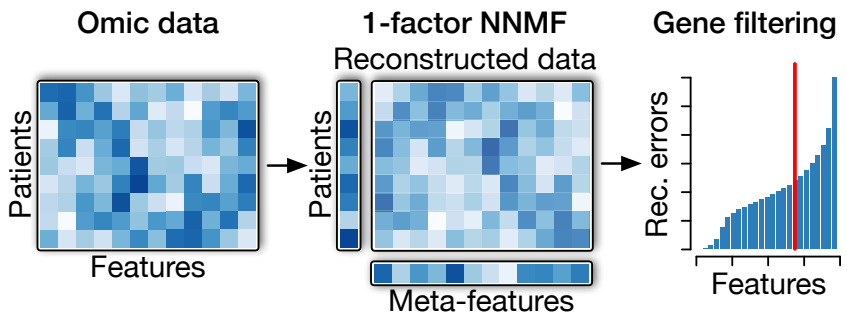

Figure 4.2: Gene filtering using non-negative matrix factorization. The original data matrix is decomposed into two vectors representing patients and their features in 1-dimensional latent space. The error of the reconstructed data using these two vectors is used to rank each gene. Only 30% of genes that have the largest error are kept for the next steps.

- W is a matrix of size $p \times k$, a representation of patients in a latent space with the number of factors is k;
- H is a matrix of size $k \times g$ representing meta-gene matrix in the latent space; and
- E is a matrix of size $p \times g$, the error between the original data and the reconstructed data from W and H.

The number of latent factors k has been used as the number of clusters in a number of clustering methods [117, 126]. If a dataset consists of k subtypes, it is expected that genes contributing to differentiating subtypes will have different expression patterns among subtypes, and these patterns can be captured in each latent factor. In our method, we use NNMF to filter features that have insignificant contributions to differentiating subtypes rather than directly assign clusters using data from NNMF. Here, we choose the number of factors $k = 1$. This makes it difficult to fit the model for genes that have significantly different expression patterns on different subtypes. As a result, genes that have a significant contribution to differentiating subtypes will have more errors in the reconstructed data. We then rank the genes by their total absolute error $\sum |E_{.g}|$ and keep only 30% of genes that have the largest error for the next steps.

After filtering unimportant features, the number of remaining features is still on the scale of hundreds or even thousands. It is necessary to perform dimension reduction to reduce the

time complexity for network construction. Therefore, we finally use principal component analysis to perform dimension deduction on each filtered data with the number of principal components at 20. This data is then used to generate connectivities between patients in the next step.

4.1.2 Consensus network generation and subtyping

To generate the overall connectivities for patients in each data type, we run k-means on the 20-dimension data with different numbers of clusters. The connectivities between patients are defined as a square matrix where both rows and columns represent patients. Its values are 1 when two patients are clustered into the same group and otherwise 0.

In this step, we aim to group a certain number of patients into the same clusters. This can be achieved by adjusting the number of clusters inputted for the k-means algorithm. For example, if the number of patients is p and the number of clusters is k, it is expected that each cluster will have an average of $\frac{p}{k}$ patients, assuming that the clustering yields balanced clusters. We choose the number of clusters k so that each cluster will have the number of members from 2 to 50. Our assumption is that if a group of patients belongs to the same subtypes, then they will tend to establish connections regardless of the predefined number of clusters. Also, by using a large number of clusters, we expect that both local and global connections between patients will be established.

It is known that the k-means algorithm often converges at local minima, especially with big numbers of clusters. However, the more time that samples clustered into the same groups, the more chance these samples belong to the same cluster in the final assignments. Therefore, for each number of clusters k, we run k-means $1,000$ times. The final patient connectivity matrix for each data type is the average of connectivities from all runs of all k.

Finally, we create an undirected weighted graph from the average of all patient connectivities across all data types. We apply community detection using the Louvain method to discover communities from the graph as the final clusters. The Louvain algorithm [125] optimizes a modularity quality function in two elementary phases: i) local moving of nodes, and ii) network

aggregation. The modularity function measures the edge density within communities compared to those between communities and is computed as follows:

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j)$$

where

- A_{ij} is the edge weight between the two nodes i and j;
- $k_i = \sum_j A_{ij}$;
- $m = \frac{1}{2} \sum_{ij} A_{ij}$;
- c_i is the community that node i is assigned; and
- $\delta(u, v) = 1$ if $u = v$ and 0 otherwise.

First, each node in the graph is assigned to a community. In the nodes moving phase, each node is moved to one of its neighbor communities, which yields the largest increase in the quality function. If no increase is gained from all moves, the node remains in its original community. This process repeats until no increase in the quality function occurs. In the network aggregation phase, each community in the first phase becomes a node to form an aggregate network. The two phases are repeated until the modularity quality function converges. The final detected communities for the network are the output clusters for all data types.

4.2 Results

In this section, we assess the performance of the proposed method using i) simulation studies and ii) 20 real datasets from TCGA. We compare DSCC with four other state-of-the-art methods in cancer subtyping, including Consensus Clustering (CC), Similarity Network Fusion (SNF), iClusterBayes (iCB), and Cancer Integration via Multikernel LeaRning (CIMLR). Among the four methods, CC is the only method that does not inherently support multiple data type integration. Therefore, in each analysis, we concatenate all data types for the integrative analysis.

We note that our method is completely unsupervised learning, in which, besides input is multi-omics data, no additional knowledge is provided for our clustering method. To make it fair with all other methods, we let each method detect the true number of clusters from the input data and use that number to generate the final cluster assignments.

4.2.1 Simulation study

To generate data for the simulation study, we use three different models to simulate different types of omics data, including Gaussian, Beta-like, and Binary models. These simulation models are inspired by Pierre-Jean et al. [127].

We simulate three different data types using the three models. Each data type has 100 samples and 10,000 features and is split into five groups, each of which has 50 differential features that distinguish between clusters. The parameters for each model are as follows: for the Gaussian model, $\mu = 2$ and $\sigma = 1$; for Beta-like model, $\mu_1 = -2$, $\sigma_1 = 0.5$, $\mu_2 = -2$, and $\sigma_2 = 0.5$; and for Binary model, we set the probability $p = 0.6$ for a value to be 1.

We also simulated noise in each data type, in which we define the based-noise for each model as follows: for Beta-like model, we add noise using normal distribution with $\mu = 0$ and $\sigma = 1$; for Beta-like model, we also add noise using normal distribution with $\mu = 0$ and $\sigma = 0.1$; and for Binary model, we add randomly add a value of 1 to the data with probability $p = 0.1$. With this level of noise, clusters of all data types are well separated. We finally simulate a total of 20 datasets, each of which consists of three data types from the three distributions with different noise levels. The noise level is adjusted by increasing σ and p in the noise added to the data from 10% to 200%.

Since the true cluster assignments are known, we use Adjusted Rand Index (ARI) [128] to assess the performance of the methods. Briefly, ARI measures the similarity between two cluster assignments with correction for chance. ARI values range from -1 to 1 where $ARI = 1$ indicates a perfect match between two cluster assignments, $ARI = 0$ indicates the agreements are expected to be the same with random cluster assignments, and negative ARI indicates that the agreement is less than what is expected from a random result.

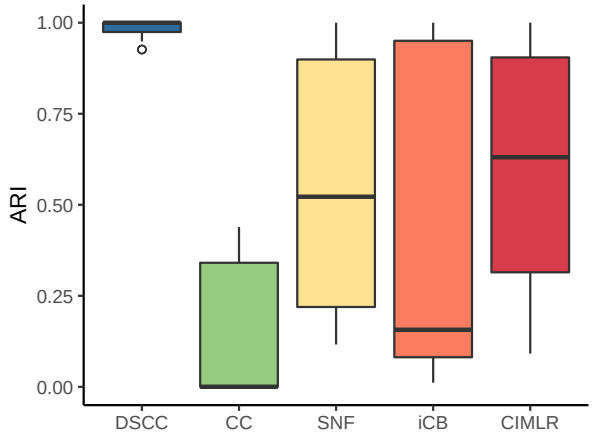

Figure 4.3: ARI values of clusters produced by DSCC, Consensus Clustering (CC), Similarity Network Fusion (SNF), iClusterBayes (iCB) and Cancer Integration via Multikernel LeaRning (CIMLR) using 20 simulated data.

Figure 4.3 shows the distribution of ARIs for the 20 simulated datasets for the five methods. CC produces clusters with the lowest ARI values since this method fails to detect the true number most of the time. SNF, iCB, and CIMLR can reach ARI values of 1 when the level of noise $< 30\%$. However, when the noise level increases more, their performance drastically decreases. While iCB can still detect the true number of clusters when the noise level increases, SNF and CIMLR fail to do so when the noise level is $> 100\%$. On the other hand, it is clear that DSCC can easily maintain the ARI values close to 1 in all datasets. The performance of DSCC is slightly affected only when the noise level is $> 150\%$.

4.2.2 Performance on TCGA data

To better assess the performance of DSCC, we compare DSCC and CC, SNF, iCB, and CIMLR on 20 TCGA datasets. The 20 datasets include Adrenocortical carcinoma (ACC), Bladder Urothelial Carcinoma (BLCA), Breast invasive carcinoma (BRCA), Cervical squamous cell carcinoma and endocervical adenocarcinoma (CESC), Cholangiocarcinoma (CHOL), Colon adenocarcinoma (COAD), Colon and Rectum adenocarcinoma (COADREAD), Diffuse Large B-cell Lymphoma (DLBC), Esophageal carcinoma (ESCA), Glioblastoma multiforme (GBM), Glioma (GBMLGG), Head and Neck squamous cell carcinoma (HNSC), Kidney Chromophobe (KICH), Pan-Kidney cohort (KIPAN), Kidney renal clear cell carcinoma (KIRC), Kidney renal papillary cell carcinoma (KIRP), Brain Lower Grade Glioma (LGG), Acute Myeloid Leukemia

(LAML), Liver hepatocellular carcinoma (LIHC), and Lung adenocarcinoma (LUAD). Each dataset consists of three data types, including mRNA, miRNA, and DNA Methylation.

Since the true subtypes are not available for any of the datasets, we use the Cox Proportional-Hazards Model [129] to validate the subtypes produced by each method. The p-values from this regression model represent the association between the survival time of patients with the subtype they are assigned. Table 4.1 shows the Cox p-values of subtypes produced by the five methods on the 20 datasets.

Table 4.1: Cox p-values of subtypes identified by DSCC, CC, SNF, iClusterBayes (iCB), and CIMLR for 20 TCGA datasets. Cells in yellow indicate significant p-values (< 0.05). Cells in green indicate the most significant p-value for each dataset.

#	Dataset	DSCC	CC	SNF	iCB	CIMLR
1	ACC	6.0e-05	8.7e-04	4.3e-05	5.4e-03	1.3e-01
2	BLCA	7.2e-05	1.1e-01	1.1e-01	2.1e-01	4.4e-01
3	BRCA	1.7e-03	1.0e-02	1.2e-01	2.7e-02	5.2e-03
4	CESC	1.6e-02	2.2e-01	5.1e-01	2.0e-02	1.9e-01
5	CHOL	5.9e-01	7.9e-02	5.7e-01	7.0e-01	3.4e-01
6	COAD	2.6e-01	5.5e-01	1.3e-01	4.2e-01	2.6e-01
7	COADREAD	6.6e-01	7.2e-01	6.6e-01	8.0e-01	3.3e-01
8	DLBC	8.8e-01	5.1e-01	7.5e-01	1.9e-01	7.4e-01
9	ESCA	3.1e-01	8.1e-01	3.9e-01	1.9e-01	5.6e-01
10	GBM	5.0e-03	7.5e-01	2.1e-02	2.6e-01	5.4e-02
11	GBMLGG	2.6e-16	4.9e-04	4.8e-14	8.0e-02	3.7e-10
12	HNSC	1.5e-03	5.1e-01	3.7e-01	7.8e-02	4.0e-01
13	KICH	5.1e-01	9.3e-01	7.0e-01	1.4e-01	4.6e-01
14	KIPAN	6.3e-19	5.3e-08	2.1e-07	1.4e-01	9.8e-05
15	KIRC	1.7e-03	8.3e-01	6.9e-01	2.1e-01	2.9e-01
16	KIRP	7.0e-03	2.2e-02	5.3e-03	4.9e-02	1.9e-02
17	LAML	3.6e-04	2.0e-01	1.7e-03	8.7e-03	8.7e-01
18	LGG	2.4e-19	1.3e-06	1.6e-14	2.3e-05	7.1e-15
19	LIHC	3.2e-04	8.2e-01	3.3e-01	2.0e-01	1.3e-01
20	LUAD	7.5e-03	7.6e-01	5.0e-01	2.2e-02	3.7e-01
	#Sigificant	14	6	7	7	5

Among 20 datasets, there are seven datasets (CHOL, COAD, COADREAD, DLBC, ESCA, KIRC and LIHC) for which none of the five methods is able to discover subtypes with significant survival differences. In the remaining 14 datasets, DSCC identifies subtypes with significantly different survival profiles on all 14 datasets. That number for CC, SNF, iCB, and CIMLR is 5, 7, 5, and 5, respectively. Moreover, DSCC has the most significant p-values for 12 out of 14 datasets.

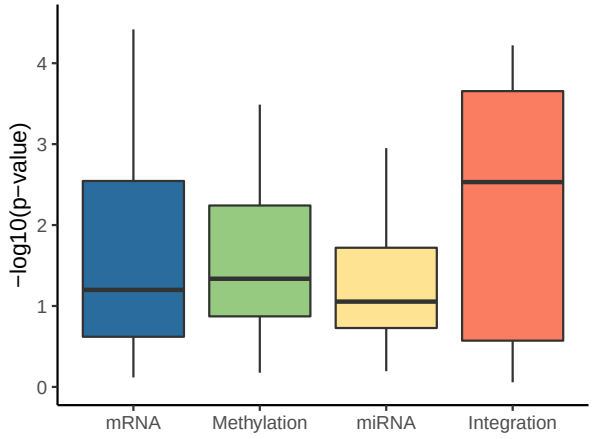

Figure 4.4: Cox p-values of subtypes identified by DSCC on each single data type and on integrated data. Overall, data integration yields better results than single data type only.

To further investigate the effect of data integration on the clustering results using DSCC, we also perform subtyping analysis for each data type and gather p-values from the produced clusters. Figure 4.4 shows the distribution of $-\log10$(p-value) by each data type and also integrated data (mRNA, Methylation, and miRNA combine together). Among 20 datasets, the Cox p-values obtained from integrated data have the median $-\log10$(p-value) of 2.5, compared to 1.2, 1.3, and 1.0 from mRNA, Methylation, and miRNA, respectively. With a significant threshold of p-value $= 0.05$ or $-\log10$(p-value) $= 1.3$, subtyping using integrated data shows that it can identify subtypes with significant differences in survival profiles, while subtyping using single data type fails to do so.

4.3 Conclusion (DSCC)

In this study, we developed a novel method, DSCC, for disease subtyping and data integration. DSCC is robust against noise and can efficiently identify cancer subtypes with significantly different survival profiles. We validated our method using 20 simulated datasets and 20 real datasets from TCGA with a total of $5,782$ patients. Our simulation study shows that DSCC can work well with data that have different distributions. It can precisely detect the true number of clusters and is robust against noise. Our evaluation of real data shows that DSCC is able to discover subtypes with significantly different survival profiles, while many other state-of-the-art fail to do so. It also shows that subtyping using data integration produces better subtypes

compared to subtyping using only a single data type. The developed method is flexible and can be applied in a wide range of applications, including cancer subtyping, single-cell RNA-seq data analysis, and other multi-omics data analysis.

Chapter 5

PINSPlus: A novel perturbation clustering method for cancer subtyping

This chapter is based on the following publication: ***Hung Nguyen****, Sangam Shrestha, Sorin Draghici, and Tin Nguyen. PINSPlus: a tool for tumor subtype discovery in integrated genomic data. Bioinformatics. 2018. DOI: 10.1093/bioinformatics/bty1049*

In this chapter, we introduce PINSPlus, a novel perturbation clustering method for cancer subtyping that radically differs from existing approaches. PINSPlus is built upon the resilience of patient connectivity and cluster ensembles to ensure robustness against noise and bias, and is an extension of the original PINS method [38], which can identify subtypes with significantly different survival profiles. PINSPlus improves the original PINS method by integrating multiple data types and providing a more accurate and efficient subtyping analysis. We demonstrate the effectiveness of PINSPlus using 37 TCGA datasets and two METABRIC datasets, analyzing a total of 3,653 cancer samples. Our results show that PINSPlus overwhelmingly outperforms existing approaches in identifying known subtypes and in discovering novel patient subgroups with significant survival differences. Furthermore, PINSPlus is more efficient than the original PINS method and is able to analyze hundreds of samples in minutes.

5.1 Methodology

PINSPlus is an unsupervised approach for subtype discovery without using any *a priori* knowledge (such as clinical variables or known subtypes). The method is based on the observation that small changes in quantitative assays will be inherently present between individuals, even in a truly homogeneous population. If distinct molecular subtypes do exist, they must be stable

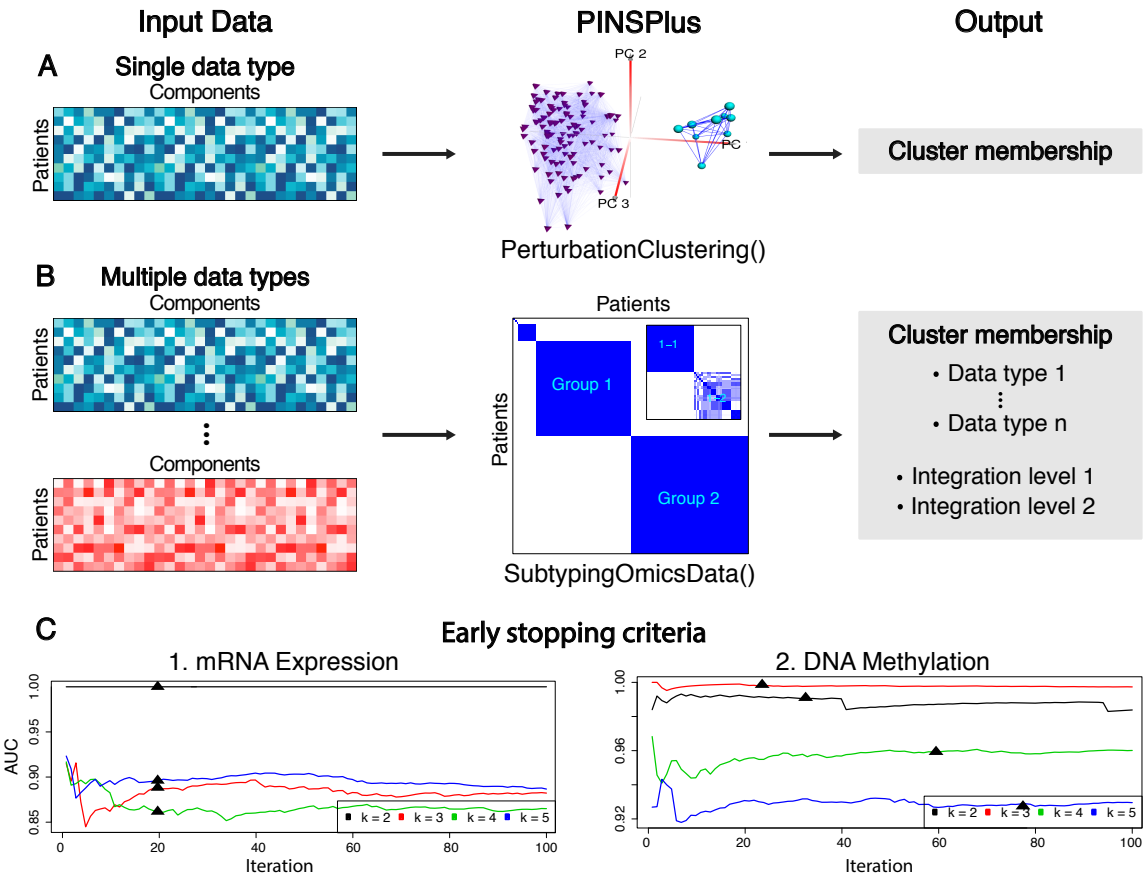

Figure 5.1: Overall workflow of PINSPlus. (A) Subtyping using a single data type. The function *PerturbationClustering()* reads a single matrix and yields the optimal number of subtypes, as well as cluster membership for each patient. (B) Subtyping using multi-omics data. The input consists of multiple matrices (data types) for the same set of patients (rows). The function *SubtypingOmicsData()* clusters each data type and combines the connectivities to subtype the multi-omics data in stage I. In stage II, the algorithm also attempts to split each discovered group. The output is the cluster membership of each patient, for each data type, and for each of stage I and II. (C) Early stoppage criterion. Beside parallel computing, PINSPlus also implemented an early stoppage criterion to speed up the analysis without compromising the results. The triangle symbols indicate the early stop point for each k in PINSPlus. We stop the iterations when the AUC values for a given number of cluster (k) converge.

with respect to small changes in quantitative assays. In order to discover reliable subtypes, we estimate how often each pair of patients is grouped together in the following scenarios: i) when data are perturbed, ii) when using different data types, and iii) when using different clustering techniques. We then partition patients into subgroups that are strongly connected in all scenarios.

PINSPlus optimizes two algorithms of PINS [38, 130]: i) *PerturbationClustering()* to cluster a single data type (Fig. 5.1A), and ii) *SubtypingOmicsData()* to integrate omics data (Fig. 5.1B). Given a single data type, the function *PerturbationClustering()* repeatedly perturbs

the data (by adding Gaussian noise) and partitions the patients using different values for cluster number. The number of clusters that gives the most stable connectivity is considered the optimal. The corresponding connectivity is considered the optimal connectivity.

For data integration, the input of *SubtypingOmicsData()* consists of multiple matrices for the same set of patients (rows) where each matrix represents a data type. The function outputs: i) subtyping results using each data type, ii) subtyping results using multi-omics data in stage I, and iii) subtyping results in stage II (Fig. 5.1B).

In order to integrate omics data, we represent patient connectivity from each data type as a graph, with patients as nodes and connectivity as edges. Our goal is to identify subgraphs that are strongly connected across all data types. We merge the connectivities of all data types into a similarity matrix that represents the overall connectivity between patients (Fig. 5.1B). We use several similarity-based algorithms to cluster the similarity and choose the partitioning that agrees the most with the partitionings of individual data types. This ensemble strategy ensures that the identified subtypes are consistent across all data types, and are robust against the choice of clustering algorithms. This completes stage I.

We also add an additional step to check whether the data has a hierarchical structure, i.e. there are subgroups of patients within discovered subtypes. Since our approach is an unsupervised approach, we do not have prior information to take into account important covariates, such as gender, race, or demographics. If these signals are predominant, we are likely to miss the real subtypes. Another motivation is that there are often heterogeneous subgroups of patients that share clinically relevant characteristics even within a subtype. For example, in breast cancer, Luminal A and Luminal B are both estrogen receptor positives and are likely to be grouped together. One-round clustering would likely overlook the subgroups within the groups identified in stage I. In stage II, we attempt to split each discovered group individually, based on reasonable conditions set to avoid over-splitting: i) stage I clustering has to be extremely imbalanced, and ii) the splitting must be supported by a strong signal across all data types.

PINSPlus also implements an early stopping criterion for the process of generating perturbed connectivity matrices (Fig. 5.1C). At each iteration, we check whether the AUC values converge. The two panels in Fig. 5.1C show an example using kidney renal clear cell carcinoma

(KIRC) data. Each curve represents the AUCs for a value of k (number of clusters). The triangle symbols in each panel indicate the early stopping point for each k in PINSPlus. For mRNA data, the AUC values for $k = 2$ are consistently larger than the rest and thus we terminate all iterations for all values of k after only 20 iterations. For methylation, each curve converges before reaching the maximum number of iterations.

It currently only takes several minutes for the software to cluster hundreds of patients with three or more types of data and tens of thousands of features. The parallel computing allows users to efficiently analyze datasets with tens of thousands of patients. The software uses k-means as the default clustering algorithm. We strongly suggest that users run PINSPlus with this setting since it has been extensively tested. However, we also provide hierarchical clustering and partitioning around methods as built-in alternatives. Users can also incorporate their own algorithm, distance metrics, or customized perturbation techniques into PINSPlus.

In the following subsections, we describe the details of the PINSPlus algorithm.

5.1.1 Connectivity resilience

Our hypothesis is that if well-defined subtypes of a disease exist, these subtypes have to be stable with respect to small changes in the measured values. This is indeed the case and we will demonstrate that the pair-wise connectivity between patients that truly belong to the same subtype tends to be preserved when the data is perturbed (Figure 5.2). In this example, we have three distinct classes of patients (Figure 5.2a). We aim to discover the subtypes with an algorithm as simple as k-means. Assuming that we do not know the correct number of subtypes, we set the number of subtypes to $k = 2$. The upper panel in Figure 5.2b shows the connectivity between patients after clustering: blue when they belong to the same cluster, and white otherwise. Now we perturb the molecular measurements and repeatedly perform clustering and partition the patients (with $k = 2$). The lower panel in Figure 5.2b shows the combined connectivity of all perturbed connectivities between patients. The visualization of the perturbed connectivity matrix clearly suggests that the larger cluster is not stable. Similarly, we partition the patients using $k = 10$ as the number of subtypes (Figure 5.2c). The discordant connectivity again states that this partitioning does not reflect the true structure of the data. More interestingly,

the perturbed connectivity matrices for both cases (lower panels in Figure 5.2b,c) clearly suggest that there are three distinct classes of patients. Finally, when we set $k = 3$ as the number of subtypes, the perturbed and the original connectivity matrices are identical (Figure 5.2d). This resilience of the patient connectivities occurs consistently regardless of the clustering algorithm being used (e.g., k-means, hierarchical clustering, partitioning around medoids, etc.), or the distribution of the data.

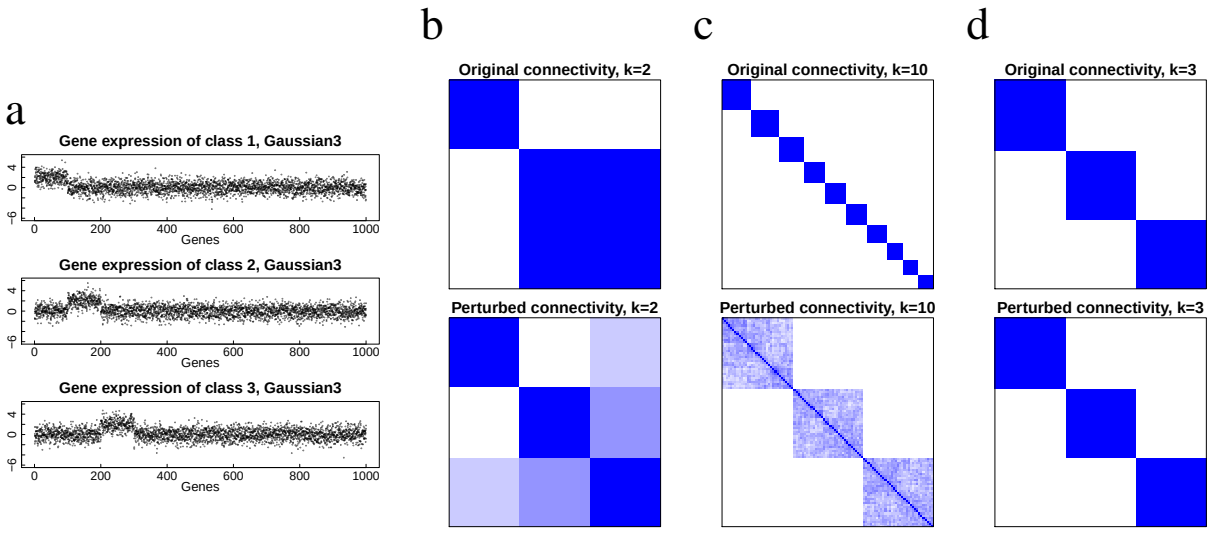

Figure 5.2: The resilience of pair-wise connectivity. (a) The dataset consists of three classes of patients: the first class has genes $1-100$ up-regulated, the second class has genes $101-200$ up-regulated, and the third class has genes $201-300$ up-regulated. (b) The original connectivity matrix (upper panel) and perturbed connectivity matrix (lower panel) for $k = 2$ clusters. Despite setting the wrong number of subtypes ($k = 2$), the perturbed connectivity matrix suggests that the data consists of three groups of samples, which is the true structure of the data. (c) The original (upper panel) and the perturbed connectivity (lower panel) matrices for $k = 10$. Again, even if the number of clusters is incorrect ($k = 10$ this time), the perturbed connectivity matrix still has three big blocks suggesting that the data consists of three groups of samples. (d) The original and perturbed connectivity matrices for $k = 3$. The agreement between the original and perturbed connectivity strongly suggests the structure of the data.

5.1.2 Perturbation clustering and stopping criterion

In the perturbation clustering algorithm proposed by Nguyen et al. [38], for each number of cluster $k \in \{2, 3, ...10\}$, perturbation process perturbs the original data then performs clustering on perturbed data in a finite number of times n, for example, $n = 200$, to generate the perturbed connectivity matrices. The algorithm then calculates the difference between the original and the perturbed connectivity matrices and computes the empirical cumulative distribution functions

of the difference matrix (CDF-DM). The area under the CDF-DM curve AUC_k is used to assess the stability of the partitioning. In the ideal case when the original and the perturbed connectivity matrices are identical, the difference matrix consists of only zero values, yielding a CDF-DM that jumps from 0 to 1 at the origin, and an AUC value of 1.

The perturbation clustering is very robust against noisy high-throughput data. However, the algorithm is slow due to the large number of perturbations needed to obtain the optimal k and AUC_k. For example, it takes 25 minutes to analyze mRNA, methylation, and miRNA data of the kidney renal clear cell carcinoma (KIRC) dataset with 124 patients. Here we optimize the algorithm to significantly reduce the analysis time. For the same dataset (KIRC, 124 patients), the running time is reduced to less than a minute.

Figure 1C in the main text shows the AUC values after each iteration for mRNA and methylation data of the KIRC dataset. For each data type, the AUC values tend to converge after a certain number of iterations, which means that at some point, additional iterations are not necessary. PINSPlus makes use of this advantage in order to determine an early stopping point for the perturbation clustering. As a result, the iteration can stop much earlier before it reaches the maximum number of iterations but still guarantees the quality of perturbed connectivity matrices. More specifically, the perturbation process will stop if: i) after the first 20 iterations, there exists a k for which $AUC_k = 1$, or ii) within all values of k, the variance of the last 20 iterations is smaller than 10^{-6}, i.e., $\frac{\sum_{i-20}^{i}(AUC_i-\mu)^2}{20} < 10^{-6}$ where $\mu = \frac{\sum_{i-20}^{i}(AUC_i)}{20}$. Figure 1C1 shows the first scenario, for which all perturbation processing for every k stops when the number of iterations $i = 20$ because $AUC_2 = 1$. Figure 1C2 shows the second scenario for which the AUC values barely change after 20 iterations before the stopping points (triangle symbols).

5.1.3 Parallel programming

PINSPlus makes use of multi-core processing to speed up the perturbation processing. The iterations in the perturbation processing are now assigned for different cores of the CPU. Many existing clustering approaches are sensitive to the number of threads being used, leading to

different results with different numbers of threads. PINSPlus implements multi-core feature in a way such that the result is stable regardless of the number of cores being used.

5.1.4 Customizable algorithm

By default, PINSPlus uses k-means as the basic clustering algorithm and Gaussian noise as the method of perturbation. To make PINSPlus more flexible, we also implemented hierarchical clustering and partitioning around medoids [131] as built-in alternatives to k-means. For advanced users, PINSPlus allows passing any customized clustering function as a parameter. For data perturbation, we also implemented a subsampling approach as an alternative method to Gaussian noise. Advanced users can also pass a customized perturbation function as a parameter.

5.1.5 Cluster ensemble and two-stage clustering

Let us consider T data types from N patients. In the first stage, PINSPlus works with each data type to build T connectivity matrices, one for each data type. A connectivity matrix can be represented as a graph, with patients as nodes, and connectivity between patients as edges. Our goal is to identify subgraphs that are strongly connected across all data types. We merge the T connectivity matrices into a combined similarity matrix that represents the overall connectivity between patients. This matrix is used as an input for similarity-based clustering algorithms, such as hierarchical clustering and partitioning around medoids [131]. We then choose the partitioning most agrees with the partitionings of individual data types [132]. This completes Stage I.

In Stage II, we consider each group one at a time and decide whether to split it further. We expect the splitting algorithm to work effectively when the data has a hierarchical structure, i.e., there are subgroups of patients within discovered subtypes. Since our method is an unsupervised approach, we do not have prior information to take into account important covariates, such as gender, race, or demographic. If these signals are predominant, we are likely to miss the real subtypes. Another motivation is that there are often heterogeneous subgroups of patients

that share clinically relevant characteristics even within a subtype. One example is that Luminal A and Luminal B are both estrogen receptor positives. If the data follows a hierarchical structure, the distances between subgroups at the second level are smaller than those between groups at the first level. Therefore, one-round clustering would likely overlook the subgroups within the groups identified in Stage I. To avoid over-splitting the subtypes, we impose some conditions before proceeding to Stage II. First, Stage I clustering has to be extremely imbalanced. Second, the splitting must be supported by a strong signal across all data types. In both cases, it is worth reviewing the data to see if each of the discovered groups can be further split. The software returns the result of both rounds, so users can investigate both groupings for discovery. Figure 5.3 demonstrates and example using the dataset KIRC.

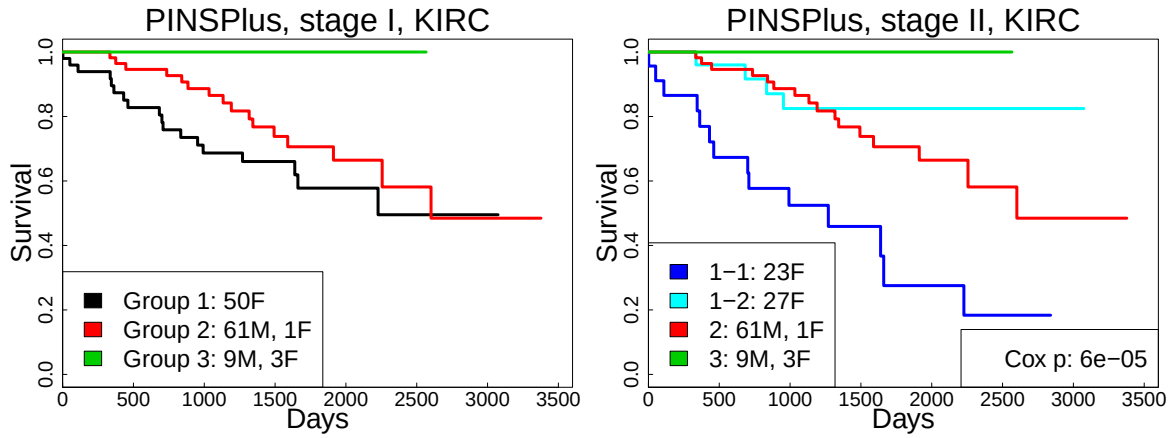

Figure 5.3: Kaplan-Meier survival analysis for kidney renal clear cell carcinoma (KIRC). The horizontal axes represent the time passed after entry into the study while the vertical axes represent estimated survival percentage. The left and right panels show subtypes discovered by PINS in stage I and stage II, respectively.

5.1.6 Choosing a suitable clustering method

PINSPlus uses k-means as default, and it has been shown to work well in our analysis of 8 mRNA and 36 omics datasets. However, in theory, k-means is not without flaws. For example, k-means might be sensitive to outliers and is not designed to discover hierarchical data structures. Therefore, we provide pam and hclust as alternative build-in algorithms. We will show examples in which one method performs well in one scenario might not be the best choice in

another scenario. We note that these examples are not supposed to provide a thorough comparison between the three methods (k-means, hclust, and pam), but to provide some guidance for a better use of PINSPlus.

Generally, if the groups are well separated, any clustering algorithm would perform well. This ideal scenario is shown in Figure 5.4a. In this example, there are 3 groups of samples with different sets of up-regulated genes. The expression values of the up-regulated genes are very different from those of un-regulated genes. As shown in the first 2 principal components, the groups are well separated. All of the three methods perform well in this ideal case.

When the distances between the groups decrease, we notice that k-means is a more robust choice. As shown in Figure 5.4b, k-means performs very well even when the three groups are close to one another. One likely reason is that the cluster centers are very stable to data perturbation. When the data is perturbed, each data point moves around its original position. However, these random effects from multiple data points are canceled out and the cluster centers do not vary drastically, leading to a very stable k-means grouping. Since hclust tries to force the data into a hierarchy, the structure changes every time the data is perturbed. Therefore, hclust tends to increase the number of clusters to seek for stability. The algorithm pam differs from k-means in the way that it uses medoids to represent clusters (instead of arithmetic centers). When the data is perturbed, the medoids move around and are unstable, leading to unstable pam groupings.

In some cases, when the data has a hierarchical structure, hclust is expected to perform better than k-means. If the data follows a hierarchical structure, the distances between subgroups at the second level are smaller than the distances between groups at the first level. Therefore, k-means probably can only identify the groups at the first level. Figure 5.4c shows an example in which the distance between groups 2 and 3 and between groups 4 and 5 are much smaller than the distance between group 1 and the rest. As shown in the principal components, the difference between groups 2 and 3 are not distinguishable when we look at the data altogether. In this case, both k-means and pam are unable to discover the true structure of the data. On the contrary, hclust perfectly separates the groups.

Figure 5.4d shows an example in which pam is the best choice. Note that in this scenario, the data are well separated and each group has approximately the same number of data points. We added some outliers in order to test the robustness of each clustering method. In this case, pam provides a perfect grouping while k-means and hclust are sensitive to outliers and are unable to identify the correct number of groups.

5.2 Results

5.2.1 Gene expression data

In order to validate PINSPlus with single data type analysis, we first tested it using eight real datasets with known subtypes from Gene Expression Omnibus and Broad Institute. We use the 8 gene expression datasets described in Table 3.2. We use the Rand Index (RI) and Adjusted Rand Index (ARI) to assess the performance of the resulted subtypes. Table 5.1 presents the results produced from PINSPlus, CC, SNF, and iClusterPlus. We note that for $iClusterPlus$ datasets, only the top 4000 components were used due to its time complexity.

Table 5.1: The performance of PINS, PINSPlus, Consensus Clustering (CC), Similarity Network Fusion (SNF), and iClusterPlus in discovering subtypes from gene expression data. For each dataset (row), cells highlighted in green have the highest Rand Index (RI), and Adjusted Rand Index (ARI). For all 8 datasets, PINSPlus outperforms its competitors by having the highest RI and ARI. SNF produced an error for GSE14924, and iClusterPlus produced an error for AML2004, shown as an NA value.

Dataset			PINS/PINS+			CC			SNF			iCluster+		
Name	Samples	#Class	k	RI	ARI	k	RI	ARI	k	RI	ARI	k	RI	ARI
GSE10245	58	2	2	0.90	0.80	6	0.64	0.32	2	0.69	0.38	4	0.58	0.22
GSE19188	91	3	3	0.84	0.66	4	0.82	0.6	4	0.61	0.12	9	0.67	0.19
GSE43580	150	2	2	0.72	0.44	3	0.68	0.37	2	0.58	0.15	5	0.61	0.21
GSE15061	366	2	2	0.83	0.65	6	0.72	0.43	2	0.53	0.05	10	0.57	0.15
GSE14924	20	2	2	1.00	1.00	7	0.64	0.25	NA	NA	NA	3	0.87	0.73
Lung2001	237	4	2	0.82	0.54	8	0.46	0.11	3	0.62	0.28	7	0.45	0.11
AML2004	38	3	4	0.85	0.65	5	0.81	0.56	2	0.59	0.17	NA	NA	NA
Brain2002	42	5	7	0.89	0.61	5	0.8	0.46	2	0.57	0.13	4	0.74	0.32

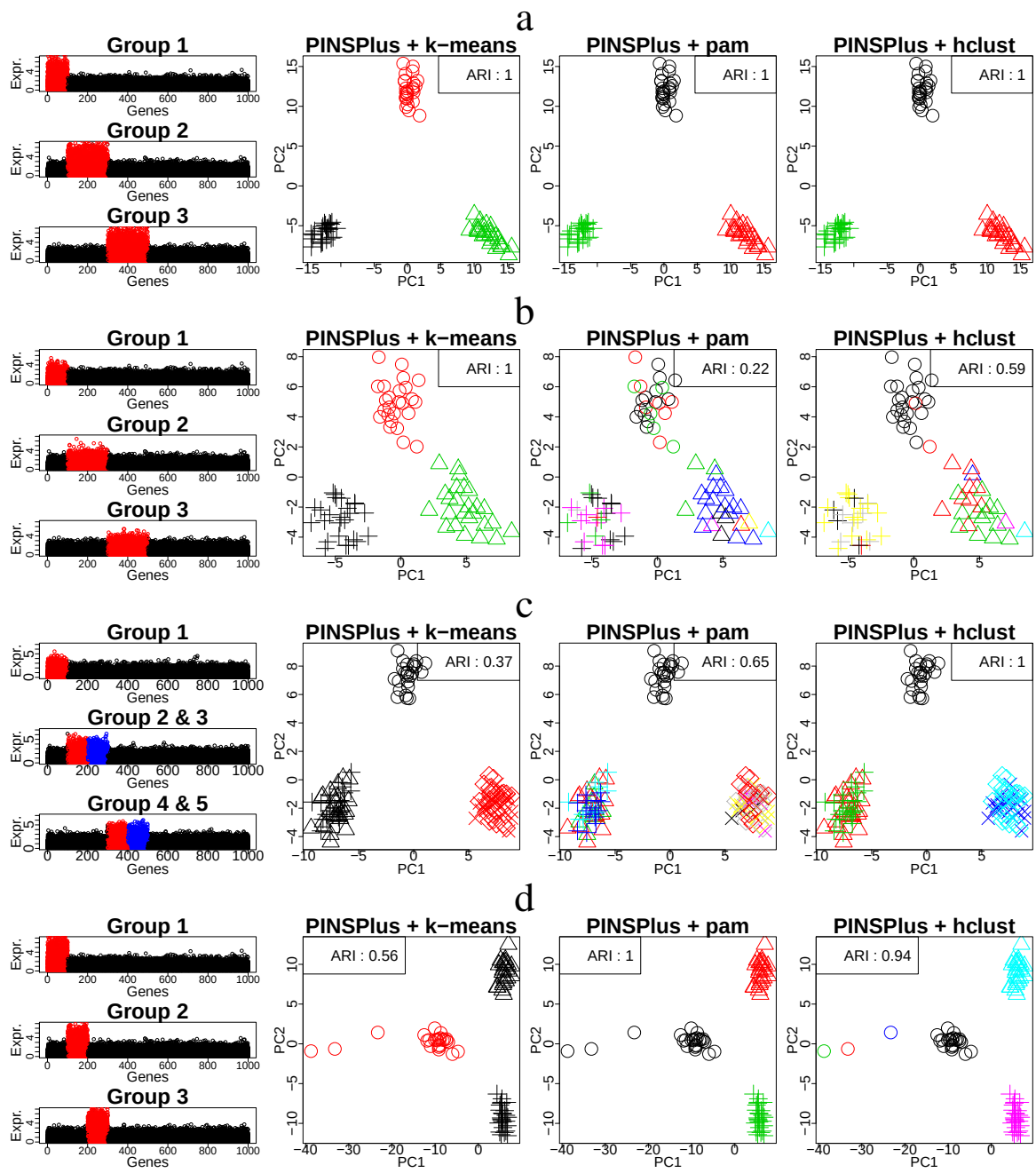

Figure 5.4: Examples to demonstrate the strength and weakness of each clustering method. In each row, the most left panel shows the data while the three remaining panels show the clustering results of PINSPlus in conjunction with k-means, pam, and hclust, respectively. (a) All clustering methods perform well when the clusters are well-separated. (b) k-means outperforms other methods when the clusters are close to one another. (c) When the data has a hierarchical structure, hclust should be the best choice. (d) In presence of outliers, pam outperforms k-means and hclust.

5.2.2 TCGA and METABRIC data

To validate PINSPlus using multi-omics data, we tested it using 34 TCGA datasets and two METABRIC datasets. The 34 TCGA datasets are: Kidney renal clear cell carcinoma (KIRC),

Glioblastoma multiforme (GBM), Acute Myeloid Leukemia (LAML), Lung squamous cell carcinoma (LUSC), Bladder Urothelial Carcinoma (BLCA), Head and Neck squamous cell carcinoma (HNSC), Liver hepatocellular carcinoma (LIHC), Stomach adenocarcinoma (STAD), Thymoma (THYM), Glioma (GBMLGG), Brain Lower Grade Glioma (LGG), Pancreatic adenocarcinoma (PAAD), Skin Cutaneous Melanoma (SKCM), Colorectal adenocarcinoma (COAD-READ), Uterine Corpus Endometrial Carcinoma (UCEC), Cervical squamous cell carcinoma and endocervical adenocarcinoma (CESC), Colon adenocarcinoma (COAD), Breast invasive carcinoma (BRCA), Stomach and Esophageal carcinoma (STES), Kidney renal papillary cell carcinoma (KIRP), Kidney Chromophobe (KICH), Uveal Melanoma (UVM), Adrenocortical carcinoma (ACC), Sarcoma (SARC), Mesothelioma (MESO), Rectum adenocarcinoma (READ), Uterine Carcinosarcoma (UCS), Ovarian serous cystadenocarcinoma (OV), Esophageal carcinoma (ESCA), Paraganglioma (PCPG), Lung adenocarcinoma (LUAD), Prostate adenocarcinoma (PRAD), Thyroid carcinoma (THCA), and Testicular Germ Cell Tumors (TGCT). We use mRNA expression, DNA methylation, and miRNA expression data for each of the 37 cancers. For each data type, we select the platform such that it gives the largest number of patients when intersecting with patients of other data types. In the preprocessing step, only log transformation (base 2) is used if the range of the data is larger than 100 to prevent the domination of genes with extreme expression values. Table 5.2 provides more details on the TCGA datasets.

We also analyze two METABRIC datasets [133], including a discovery cohort (997 patients) and a validation cohort (983 patients). For each of these patients, matched DNA and RNA were subjected to copy number analysis and transcriptional profiling on the Affymetrix SNP 6.0 and Illumina HT 12 v3 platforms, respectively. We download the mRNA and copy number variation (CNV) data from the European Genome-Phenome Archive (`www.ebi.ac.uk/ega/`) and high-quality follow-up clinical data from cBioPortal (`www.cbioportal.org`). There are patients who have been followed up upon for almost 30 years. The only preprocessing done is mapping CNVs to genes using the CNTools package [134]. Table 5.3 provides more details on the METABRIC datasets.

Table 5.2: Description of the 34 datasets from The Cancer Genome Atlas (TCGA)

Dataset	#Sample	mRNA	Methylation	miRNA
KIRC	124	HiSeq RNASeq	Methylation27	GASeq miRNASeq
GBM	273	HT HG-U133A	Methylation27	HiSeq miRNASeq
LAML	164	GASeq RNASeq	Methylation27	GASeq miRNASeq
LUSC	110	HT HG-U133A	Methylation27	GASeq miRNASeq
BLCA	404	HiSeq RNASeq v2	Methylation450	GASeq miRNASeq
HNSC	228	HiSeq RNASeq	Methylation450	HiSeq miRNASeq
LIHC	366	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
STAD	362	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
THYM	654	HiSeq RNASeq v2	Methylation450	GASeq miRNASeq
GBMLGG	654	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
LGG	510	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
PAAD	178	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
SKCM	439	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
COADREAD	294	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
UCEC	234	GASeq RNASeq v2	Methylation450	HiSeq miRNASeq
CESC	304	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
COAD	220	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
BRCA	622	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
STES	545	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
KIRP	271	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
KICH	65	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
UVM	80	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
ACC	79	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
SARC	257	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
MESO	86	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
READ	74	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
UCS	56	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
OV	286	HiSeq RNASeq v2	Methylation27	HiSeq miRNASeq
ESCA	183	HiSeq RNASeq	Methylation450	HiSeq miRNASeq
PCPG	179	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
LUAD	428	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
PRAD	493	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
THCA	499	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
TGCT	134	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq

Table 5.3: Description of the 2 datasets from The Molecular Taxonomy of Breast Cancer International Consortium (METABRIC): METABRIC discovery and METABRIC validation.

Dataset	#Sample	mRNA	CNV
Discovery	997	Illumina HT 12 v3	Affymetrix SNP 6.0
Validation	983	Illumina HT 12 v3	Affymetrix SNP 6.0

The results are reported in Table 5.4. There are 9 datasets for which no method is able to identify subtypes with significantly different survival (READ, UCS, OV, ESCA, PCPG, LUAD, PRAD, THCA, TGCT). For the remaining 27 datasets, PINSPlus has significant p-values in all of them whereas CC, SNF, and iClusterPlus has significant p-values in only in 8, 14, and 9 datasets, respectively. More importantly, PINSPlus has the most significant p-values in 23 datasets (out of 27).

5.2.3 Running time

Table 6.4 shows the running time of each method for the 34 datasets. For gene expression data, PINSPlus, CC, and SNF can finish each analysis in less than a minute while it takes iClusterPlus several hours. The gap in running time is much larger for data integration. PINSPlus, CC, and SNF can integrate omics data and partition hundreds of patients in minutes while iClusterPlus (with 60 cores) takes up to many hours to analyze large datasets.

5.3 Conclusion (PINSPlus)

As an unsupervised approach, PINSPlus relies solely on molecular data to discover disease subtypes. One caution is that a cluster of samples could be determined not only by molecular measures but also by other variables like environmental or clinical variables. These variables could represent confounders and they should be considered explicitly when available. This problem can be addressed in a number of ways, for instance by integrating the connectivity matrices obtained from clinical variables. We plan to extend PINSPlus in the future to exploit clinical data whenever possible.

Nevertheless, PINSPlus is a fast and powerful software for subtype discovery. PINSPlus overwhelmingly outperforms established approaches in identifying known subtypes and discovering novel subgroups of patients with significant survival differences. The software is flexible enough to be applied in many areas to tackle unsupervised machine learning problems involving either single or multiple types of high-dimensional data.

Table 5.4: Cox p-values of subtypes discovered by PINSPlus, CC, SNF, and iClusterPlus for two METABRIC breast cancer datasets and 34 TCGA datasets. Cells highlighted in yellow have significant Cox p-values at the threshold of 5%. Cells highlighted in green have the most significant Cox p-value. PINSPlus substantially outperforms the other methods in identifying subtypes with significant survival differences.

Dataset	Size	PINS+	CC	SNF	iCluster+
METABRIC					
1. Discovery	997	1.8e−9	0.022	2.3e−5	0.378
2. Validation	983	3.4e−5	0.096	0.010	0.031
TCGA					
1. KIRC	124	6e−5	0.118	0.691	0.058
2. GBM	273	8.7e−5	0.014	0.021	0.103
3. LAML	164	8.7e−4	0.292	0.002	0.083
4. LUSC	110	0.008	0.688	0.087	0.224
5. BLCA	404	0.019	0.089	0.109	0.17
6. HNSC	228	0.046	0.428	0.366	0.364
7. LIHC	366	0.03	0.622	0.334	0.072
8. STAD	362	0.002	0.428	0.041	0.434
9. THYM	119	0.013	0.139	0.097	0.24
10. GBMLGG	510	7.5e−17	5.2e−4	4.8e−14	5.4e−14
11. LGG	510	7.7e−25	2e−6	1.6e−14	2.7e−14
12. PAAD	178	2.5e−4	0.013	7.4e−4	6.3e−4
13. SKCM	439	0.048	0.604	0.478	0.108
14. COADREAD	294	0.003	0.946	0.66	0.178
15. UCEC	234	0.001	0.105	0.018	0.619
16. CESC	304	0.03	0.376	0.51	0.201
17. COAD	220	0.001	0.419	0.128	0.884
18. BRCA	622	0.007	0.008	0.119	0.014
19. STES	545	0.007	0.301	0.157	0.46
20. KIRP	271	1.1e−9	0.367	0.005	0.013
21. KICH	65	0.028	0.955	0.701	0.788
22. UVM	80	7.5e−4	0.005	1.7e−4	0.003
23. ACC	79	0.007	0.014	4.3e−5	7.1e−4
24. SARC	257	0.03	0.148	0.044	4e−4
25. MESO	86	7.3e−4	0.272	4.2e−4	2.2e−4
26. READ	74	0.649	0.737	0.762	0.249
27. UCS	56	0.458	0.207	0.859	0.983
28. OV	286	0.319	0.859	0.445	0.062
29. ESCA	183	0.33	0.791	0.392	0.16
30. PCPG	179	0.866	0.938	0.332	0.55
31. LUAD	428	0.099	0.926	0.501	0.118
32. PRAD	493	0.349	0.638	0.475	0.879
33. THCA	499	0.166	0.64	0.62	0.111
34. TGCT	134	0.531	0.758	0.838	0.58

Table 5.5: Running time of each subtyping method. The time is rounded to minutes (min). CC and SNF can only run on 1 core while PINSPlus and iClusterPlus allow for parallel computing.

Consortium	Dataset	#Patient	PINS 1 core	PINS+ 2 cores	CC 1 core	SNF 1 core	iCluster+ 60 cores
GEO&Broad	GSE10245	58	<1m	<1m	<1m	<1m	19m
	GSE19188	91	1m	<1m	<1m	<1m	29m
	GSE43580	150	2m	<1m	<1m	<1m	50m
	GSE15061	366	12m	<1m	<1m	<1m	100m
	GSE14924	20	<1m	<1m	<1m	<1m	9m
	Lung2001	237	5m	<1m	<1m	<1m	58m
	AML2004	38	<1m	<1m	<1m	<1m	NA
	Brain2002	42	<1m	<1m	<1m	<1m	16m
TCGA	KIRC	124	6m	<1m	<1m	<1m	95m
	GBM	273	53m	1m	<1m	<1m	190m
	LAML	164	10m	<1m	<1m	<1m	123m
	LUSC	110	5m	<1m	<1m	<1m	59m
	BLCA	404	112m	6m	3m	3m	433m
	HNSC	228	32m	4m	3m	2m	101m
	LIHC	366	96m	5m	4m	3m	263m
	STAD	362	97m	5m	4m	3m	299m
	THYM	119	6m	1m	2m	1m	95m
	GBMLGG	510	192m	7m	7m	4m	392m
	LGG	510	188m	12m	8m	6m	274m
	PAAD	178	20m	3m	2m	1m	176m
	SKCM	439	144m	8m	3m	3m	202m
	COADREAD	294	61m	5m	4m	3m	157m
	UCEC	234	34m	4m	4m	2m	201m
	CESC	304	60m	7m	5m	2m	203m
	COAD	220	30m	3m	3m	2m	126m
	BRCA	622	236m	16m	10m	5m	285m
	STES	545	171m	12m	14m	5m	324m
	KIRP	271	33m	3m	3m	1m	184m
	KICH	65	4m	1m	1m	<1m	58m
	UVM	80	3m	1m	1m	1m	71m
	ACC	79	3m	1m	1m	<1m	63m
	SARC	257	43m	5m	3m	1m	201m
	MESO	86	4m	1m	2m	<1m	72m
	READ	74	3m	1m	2m	<1m	52m
	UCS	56	2m	1m	1m	<1m	32m
	OV	286	52m	2m	2m	1m	188m
	ESCA	183	23m	5m	5m	2m	204m
	PCPG	179	16m	2m	3m	1m	244m
	LUAD	428	128m	8m	5m	3m	233m
	PRAD	493	205m	11m	10m	5m	276m
	THCA	499	213m	10m	5m	3m	251m
	TGCT	134	9m	2m	2m	1m	105m
METABRIC	Discovery	997	1153m	9m	15m	2m	350m
	Validation	983	581m	8m	14m	2m	348m

Chapter 6

SMRT: Randomized data transformation for cancer subtyping and big data analysis

This chapter is based on the following publication: ***Hung Nguyen****, Duc Tran, Bang Tran, Monikrishna Roy, Adam Cassell, Sergiu Dascalu, Sorin Draghici, and Tin Nguyen. SMRT: Randomized Data Transformation for Cancer Subtyping and Big Data Analysis. Frontiers in Oncology. 2021. DOI: 10.3389/fonc.2021.725133*

In this chapter, we introduce Subtyping Multi-omics using a Randomized Transformation (SMRT), a new method for cancer subtyping and big data analysis. This method offers important advantages over existing software: (i) it allows researchers to analyze hundreds of thousands of samples in minutes, (ii) it can integrate data types with different numbers of patients, (iii) it has the ability to integrate and analyze unmatched data of different types, and (iv) the web application offers a convenient data analysis pipeline. We also improve the efficiency of our ensemble-based perturbation clustering to support analysis on machines with memory constraints. Our extensive analysis on 37 TCGA and two METABRIC datasets shows that SMRT is more accurate than state-of-the-art subtyping methods in identifying subtypes with significantly different survival profiles. In addition, our simulations with big data show that SMRT is fast and many-fold more scalable than existing methods. Specifically, SMRT is able to analyze hundreds of thousands of samples in minutes.

6.1 Materials and Methods

6.1.1 The SMRT pipeline

The overall workflow of SMRT is presented in Figure 6.1. This workflow offers two different analysis pipelines for big data and data with a moderate size. In the first case, given a multi-omics dataset with a moderate size (e.g., less than 2,000 samples), SMRT performs subtyping as follows. It first projects each data type onto a lower-dimensional space using randomized singular value decomposition (RSVD) and then performs a perturbation clustering (PINS) [38, 4] to determine the subtypes within each data level. It also builds a pair-wise connectivity matrix for each data type that represents the connectivity between patients. Next, the method combines the connectivity matrices into a single similarity matrix and then determines the final subtypes using an ensemble of multiple similarity-based methods. In the second case, when the data has more than 2,000 samples, SMRT splits the data into two different sets of patients: a sampled set and a propagated set. It then performs the subtyping on the sampled set and then assigns the patients from the propagated set to the identified subtypes. Note that the number 2,000 is chosen to balance the accuracy and time complexity of the method. This moderate number of samples allows SMRT to perform a fast and accurate analysis in limited memory (see Section 6.3.6). Our simulation studies show that the results do not change when we vary this number. However, users are free to change this parameter when using the R package. Below is the description of each of these analysis modules.

6.1.2 Dimension reduction using randomized singular value decomposition

The goal of this step is to project the multi-omics data into a lower-dimensional space using randomized singular value decomposition (RSVD). For data with hundreds of thousands of dimensions (e.g., Illumina 450k), this step substantially reduces the required computational power while maintaining the clustering accuracy. Let us denote $X \in \mathbb{R}^{n \times m}$ as the input matrix, where n is the number of samples/patients, and m is the number of genes/features. Briefly, the RSVD method starts by generating a random projection matrix $P \in \mathbb{R}^{m \times r}$ from a standard normal distribution where $r \ll m$. It then projects $X \in \mathbb{R}^{n \times m}$ to the column space of P to get

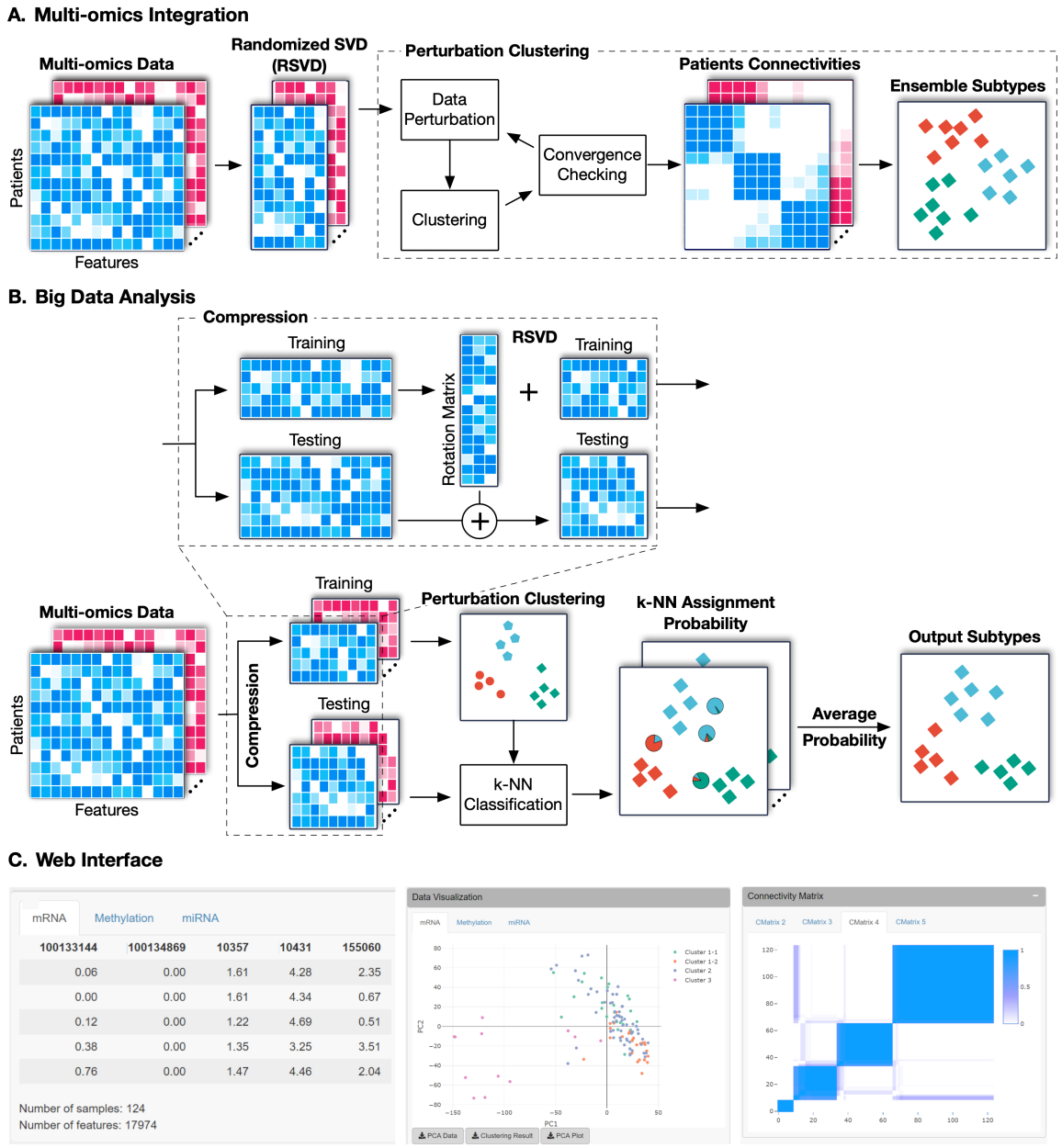

Figure 6.1: The overall workflow of SMRT. (A) Analysis pipeline for data with moderate size. First, SMRT projects each data type to a lower-dimensional space using randomized singular value decomposition (RSVD). Next, it performs perturbation clustering to determine the subtypes and build a pair-wise patient connectivity for each data type. Finally, it merges the connectivity matrices onto a single similarity matrix and then determines the final subtypes using a cluster ensemble. The output is the clustering results for each data type, as well as the results after the multi-omics data integration. (B) Analysis pipeline for big data. SMRT first splits the data into two different sets: a sampled and a propagated set. The method first determines the subtypes using the sampled set and then assigns the patients from the propagated set to subtypes identified using the sampled set. The sampled data is partitioned using the pipeline described in (A). These subtypes are used to generate multiple k-NN models – one per data type. The final subtype assignments for samples in the propagated set are determined by averaging the probabilities from all k-NN models. (C) An example of the subtypes discovered by the SMRT web service for the KIRC dataset. The left panel shows a preview of the uploaded data. The middle panel shows the visualization of the discovered subtypes and export functions. The right panel shows patient connectivity matrices for each data type.

a matrix Z such that $Z = XP$. Due to the random projection, Z and X will have approximately the same dominant columns (features). Now, we can obtain the orthogonalized matrix Q of Z by using QR decomposition, where Q has the same size as Z of $n \times r$. In the next step, the method projects X into a smaller space to get a matrix $Y \in \mathbb{R}^{r \times m}$ such that $Y = Q^T * X$ and then computes singular value decomposition (SVD) of Y as $Y = U\Sigma V^*$ using the traditional SVD method [135]. U and V matrices only keep at most r eigenvectors so the size of U is $r \times r$ and the size of V^* is $m \times r$. Finally, the low-rank rotated data of the original matrix X can be computed using: $X' = XV^*$.

In practice, RSVD is faster and requires less memory than the traditional SVD. To further speed up our approach, we implement a parallel version of RSVD that can efficiently utilize multiple cores available in modern processors. Note that when the input data is large (e.g., more than 2,000 samples), we do not perform RSVD on the whole input. Instead, we split the data into two sets of patients: a sampled set and a propagated set. We first perform RSVD on the sampled set, and then project the original data matrix (both sampled and propagated set) to the subspace of the *sampled set* by multiplying it with the rotation matrix obtained from the RSVD for the sampled set. This implementation allows us to perform SVD in, at most, a few seconds, even for datasets with hundreds of thousands of samples and features.

The output of this module is multiple matrices – one per data type. In each matrix, the rows represent patients, while the columns represent the principal components (PCA). These matrices will serve as input for the next module, perturbation clustering, which will be described in the next section. This will compute the perturbed connectivity matrices and determine the subtypes.

6.1.3 Subtype discovery using one data type

Given a single data type, SMRT utilizes our previously developed perturbation clustering (PINS) [38, 4] to partition the data. Briefly, we perturb the data (by adding Gaussian noise) and repeatedly partition the patients (using k-means by default). For each partitioning, we build a pair-wise connectivity matrix of 0's and 1's in which 1 means that the two patients belong to the same

cluster and 0 otherwise. By perturbing and clustering the data multiple times, we obtain multiple connectivity matrices that represent how stable the connectivity between patient pairs is. Finally, we choose the partitioning that is the most stable for data perturbation. This algorithm automatically determines the number of clusters and patient subgroups.

When the number of samples is large, the perturbation clustering becomes slow and memory-inefficient. The perturbation clustering algorithm relies on the pair-wise connectivity of size $n \times n$ for clustering (n is the number of patients). The time and space complexity (running time and memory usage) of this method increase quadratically when the number of samples increases. Therefore, when the number of samples is large (by default setting, when $n > 2,000$), we perform a sub-sampling process over the original data to obtain a subset of 2,000 patients/samples. Next, we transform the data into a lower-dimensional space and use the perturbation clustering to partition these patients. After this step, each of the 2,000 patients has a subtype. Let us refer to this selected set of 2,000 patients as the *sampled set*. The next step is to determine the subtypes for the rest of the patients, called the *propagated set*. For this purpose, we use the fast k-nearest neighbor searching algorithms (FKNN) algorithm [136, 137] to assign each patient from the propagated set to one of the subtypes in the sampled set. Briefly, the FKNN method calculates the distance between the new patient to the k nearest patients in the sampled set. Next, the FKNN method classifies the new patient using vote counting (i.e., it chooses the subtype with the most patients among the k neighbors). By default, k is determined using the Elbow method on the sampled set using 5-fold cross-validation. The sampled set is divided randomly into 5 equally smaller sets. In each round, the combination of 4 sets is used as the training set, and the other is used as the validation set for the KNN algorithm with k ranges from 5 to a maximum of 50. The k that yields the lowest average classification error rate will be used as the optimal k. However, users are also free to modify the value of this parameter. Section 6.3.7 provides more details on the performance of using the Elbow method versus using a fixed number of k.

One note of caution is that the number of dimensions of the data can be high, thus slowing the process of distance calculation and neighbor finding. Therefore, instead of calculating the distance between patients in the original space, we calculate the distance between patients in

the principal component (PC) space of the sampled set. As described above, we project the original data matrix (both sampled and propagated set) to the subspace of the *sampled set* by multiplying it with the projection matrix obtained from the RSVD for the sampled set. After this transformation, the pair-wise distance between patients will be calculated in the new space with a much lower number of dimensions.

6.1.4 Subtype discovery using multi-omics data

When the number of samples is small (by default, when $n \leq 2,000$), we utilize an ensemble strategy to partition the patients. The method first clusters each data type (using the algorithm described in Section 6.1.3) and constructs the perturbed connectivity matrices. It then merges the connectivity matrices of all data types into a single similarity matrix that represents the similarity between patients across all data types by averaging the connectivity values for each pair of samples. Next, to cluster the similarity matrix, it uses several similarity-based algorithms, including hierarchical clustering, partitioning around medoids [131], and dynamic tree cut [138] and then chooses the partitioning that agrees the most with the partitionings of individual data types. This ensemble strategy ensures that the identified subtypes are consistent across all data types and are robust against the choice of clustering algorithms.

When the number of samples is large (by default, when $n > 2,000$), we perform a subsampling and classifying procedure that is similar to the algorithm described in Section 6.1.3. The difference here is that multiple data types are involved. First, we randomly select 2,000 samples/patients and then apply the multi-omics algorithm described above to partition the selected samples. We refer to this selected set of 2,000 patients as the *sampled set* and the remaining patients as the *propagated set*. The next task is to determine the subtypes of patients in the propagated set. Given a patient in the propagated set, we perform the FKNN procedure for each data type to obtain the probability that it belongs to each subtype using the labels obtained from the nearest neighbors. The final probabilities are calculated by averaging the probabilities across all data types. Finally, we classify the patient to the subtype that has the highest probability. This strategy is also applied when integrating multi-omics data whose each data type has a different number of samples. Here, the sampled set will be the set of patients

(by default, a maximum of 2,000 patients) that have data in all data types, and the remaining patients will be in the propagated set.

6.2 Data and pre-processing

In this article, we analyze 39 cancer datasets: 37 datasets from The Cancer Genome Atlas datasets (TCGA), and two datasets from The Molecular Taxonomy of Breast Cancer International Consortium (METABRIC) [133].

We analyze 37 different types of cancer with curated level three data, available at the TCGA website (`cancergenome.nih.gov` and `firebrowse.org`): Kidney Renal Clear Cell Carcinoma (KIRC), Glioblastoma Multiforme (GBM), Acute Myeloid Leukemia (LAML), Lung Squamous Cell Carcinoma (LUSC), Bladder Urothelial Carcinoma (BLCA), Head and Neck Squamous Cell Carcinoma (HNSC), Liver Hepatocellular Carcinoma (LIHC), Stomach Adenocarcinoma (STAD), Thymoma (THYM), Glioma (GBMLGG), Brain Lower Grade Glioma (LGG), Pancreatic Adenocarcinoma (PAAD), Skin Cutaneous Melanoma (SKCM), Colorectal Adenocarcinoma (COADREAD), Uterine Corpus Endometrial Carcinoma (UCEC), Cervical Squamous Cell Carcinoma and Endocervical Adenocarcinoma (CESC), Colon Adenocarcinoma (COAD), Breast Invasive Carcinoma (BRCA), Stomach and Esophageal Carcinoma (STES), Kidney Renal Papillary Cell Carcinoma (KIRP), Kidney Chromophobe (KICH), Uveal Melanoma (UVM), Adrenocortical Carcinoma (ACC), Sarcoma (SARC), Mesothelioma (MESO), Rectum Adenocarcinoma (READ), Uterine Carcinosarcoma (UCS), Ovarian Serous Cystadenocarcinoma (OV), Esophageal Carcinoma (ESCA), Paraganglioma (PCPG), Lung Adenocarcinoma (LUAD), Prostate Adenocarcinoma (PRAD), Thyroid Carcinoma (THCA), and Testicular Germ Cell Tumors (TGCT), Cholangiocarcinoma (CHOL), Lymphoid Neoplasm Diffuse Large B-cell Lymphoma (DLBC), Pan-kidney (KIPAN). We applied the same data processing as describe in Section 5.2.2. Table 6.1 provides more details on the TCGA datasets.

For the METABRIC datasets, we use the same data as described in Section 5.2.2. The METABRIC dataset consists of two datasets: METABRIC_Discovery and METABRIC_Validation.

Table 6.1: Description of 37 datasets downloaded from The Cancer Genome Atlas (TCGA).

Dataset	#Samples	mRNA	Methylation	miRNA
ACC	79	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
BLCA	404	HiSeq RNASeq v2	Methylation450	GASeq miRNASeq
BRCA	622	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
CHOL	36	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
CESC	304	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
COAD	220	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
COADREAD	294	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
DBLC	47	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
ESCA	183	HiSeq RNASeq	Methylation450	HiSeq miRNASeq
GBM	273	HT HG-U133A	Methylation27	HiSeq miRNASeq
GBMLGG	510	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
HNSC	228	HiSeq RNASeq	Methylation450	HiSeq miRNASeq
KICH	65	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
KIPAN	654	HiSeq RNASeq	Methylation450	HiSeq miRNASeq
KIRC	124	HiSeq RNASeq	Methylation27	GASeq miRNASeq
KIRP	271	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
LAML	164	GASeq RNASeq	Methylation27	GASeq miRNASeq
LGG	510	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
LIHC	366	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
LUAD	428	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
LUSC	110	HT HG-U133A	Methylation27	GASeq miRNASeq
MESO	86	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
OV	286	HiSeq RNASeq v2	Methylation27	HiSeq miRNASeq
PAAD	178	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
PCPG	179	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
PRAD	493	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
READ	74	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
SARC	257	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
SKCM	439	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
STAD	362	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
STES	545	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
TGCT	134	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
THCA	499	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
THYM	119	HiSeq RNASeq v2	Methylation450	GASeq miRNASeq
UCEC	234	GASeq RNASeq v2	Methylation450	HiSeq miRNASeq
UCS	56	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq
UVM	80	HiSeq RNASeq v2	Methylation450	HiSeq miRNASeq

Table 6.2: Description of the 2 datasets from The Molecular Taxonomy of Breast Cancer International Consortium (METABRIC): METABRIC discovery and METABRIC validation.

Dataset	#Sample	mRNA	CNV
Discovery	997	Illumina HT 12 v3	Affymetrix SNP 6.0
Validation	983	Illumina HT 12 v3	Affymetrix SNP 6.0

The METABRIC_Discovery dataset contains 997 patients, while the METABRIC_Validation dataset contains 983 patients. Table 6.2 provides more details on the METABRIC datasets.

When performing disease subtyping analysis with SMRT, we suggest that users use standardized data normalization, e.g., RSEM for RNA-Seq data, to process their data. We also recommend that users use log transformation (base 2) to transform the data if the range of the data is large (e.g., > 100) to mitigate unexpected effects caused by extreme expression values on the clustering results.

6.3 Results

We perform an extensive analysis using 39 cancer datasets and simulated data. First, we demonstrate that SMRT is able to identify cancer subtypes with significantly different survival profiles. Second, we provide a comparative analysis between subtypes discovered by SMRT and those of the PAM50 classifier on three Breast cancer datasets (TCGA-BRCA, METABRIC_Discovery, and METABRIC_Validation). Third, we perform an in-depth analysis for the Glioma dataset. Fourth, we investigate the contribution of each data type to the subtyping results. Fifth, we perform clinical enrichment analysis to show the clinical relevance of the discovered subtypes. Finally, we illustrate the scalability of SMRT by analyzing simulated datasets with hundreds of thousands of samples. In addition, we investigate the effect of automatic k selection in the KNN algorithm on the performance of SMRT.

6.3.1 Experimental studies using 39 cancer datasets

We compare SMRT with eight state-of-the-art subtyping algorithms: SNF [97], CIMLR [31], NEMO [101], moCluster [89], iClusterBayes [82], LRACluster [79], MCCA [96], and IntNMF [93]. The following packages were used in our comparison: SNFtool v2.3.0 on CRAN for SNF, CIMLR v1.0.0 at `https://github.com/danro9685/CIMLR` for CIMLR, NEMO

v0.1.0 at https://github.com/Shamir-Lab/NEMO for NEMO, mogsa v1.16.0 on Bioconductor for moCluster, iClusterPlus on Bioconductor v1.18.0 for iClusterBayes, LRACluster v1.18.0 at http://bioinfo.au.tsinghua.edu.cn/member/jgu/lracluster/ for LRACluster, PMA v1.2.1 on CRAN for MCCA, and IntNMF on CRAN v1.2.0 for IntNMF. When the number of dimensions exceeded 2,000, we used only the top 2,000 variables with the largest variance for iClusterBayes, IntNMF, and MCCA, because these methods cannot analyze the data on the whole-genome scale. For all methods, we used default parameters and let all methods automatically determine the optimal number of clusters. For MCCA, which is not a clustering method itself, we follow the implementation at https://github.com/Shamir-Lab/Multi-Omics-Cancer-Benchmark for cluster analysis.

Using each method, we partition the patients in each dataset, and then assess the survival difference of the discovered patient groups using Cox regression [139]. Table 6.3 shows the Cox p-values obtained from each dataset and method. There are seven datasets in which no method is able to identify subtypes with significant Cox p-values. For the remaining 32 datasets, SMRT has significant p-values in 28 datasets, whereas NEMO has significant p-values in 19 datasets and all other methods have significant p-values in 15 datasets or less. SMRT has the most significant p-values in 12 datasets out of those 28 datasets, while SNF, CIMLR, NEMO, moCluster, iClusterBayes, LRACluster, MCCA, and IntNMF have the most significant p-values in 0, 3, 8, 4, 2, 0, 1, and 2 datasets, respectively.

Figure 6.2 shows the distributions of the Cox p-values in the -log10 scale while Figure 6.3-6.10 show the Kaplan-Meier survival analysis for each dataset. Overall, the median -log10 p-values of SMRT is close to 2 (i.e., median p-value of 0.01) whereas the median -log10 p-value of the second-best method (NEMO) is close to 1 (i.e., median p-value of 0.1). A Wilcoxon test also confirms that the p-values of SMRT are significantly smaller than the p-values obtained from other methods ($p = 0.0002$ using the one-tailed Wilcoxon test).

The running time of each method is shown in Table 6.4. The top 39 rows show the running time of each method in each dataset, while the last row shows the average running time. On average, SMRT, SNF, NEMO, and MCCA are fast and able to finish each analysis in less than

Table 6.3: Cox p-values of subtypes discovered by SNF, CIMLR, NEMO, moCluster, iClusterBayes (iCB), LRACluster (LRA), MCCA, IntNMF, and SMRT for 37 TCGA datasets and two METABRIC breast cancer datasets (M_Discovery and M_Validation). Cells highlighted in yellow have significant Cox p-values at the threshold of 5%. Cells highlighted in green have the most significant Cox p-value in their respective rows. No methods were able to yield subtypes with significantly different survival in 7 data sets (shown with red fonts). SMRT yields subtypes with significantly different survival profiles in 28 out of the 39 datasets. In 12 such datasets, SMRT also p-values more significant than any of those provided by the other eight methods.

Dataset	SNF	CIMLR	NEMO	moCluster	iCB	LRA	MCCA	IntNMF	SMRT
1. ACC	4.34e-05	3.96e-01	2.07e-04	2.63e-09	4.26e-03	2.46e-03	1.24e-08	6.11e-03	1.33e-02
2. BLCA	1.09e-01	3.09e-01	6.74e-02	3.13e-01	4.95e-01	7.42e-02	3.57e-01	3.43e-02	1.95e-02
3. BRCA	1.19e-01	4.95e-03	2.93e-02	2.58e-01	3.07e-02	3.90e-01	3.80e-04	2.53e-01	1.96e-03
4. CESC	5.10e-01	1.90e-01	3.33e-01	1.81e-01	1.69e-01	2.90e-01	6.69e-01	8.89e-01	2.95e-02
5. CHOL	5.72e-01	3.35e-01	3.02e-01	5.17e-01	6.51e-01	6.93e-01	4.50e-01	9.63e-01	3.01e-02
6. COAD	1.28e-01	2.52e-01	6.76e-01	3.73e-01	6.47e-01	5.05e-01	6.20e-01	5.35e-01	1.44e-03
7. COADREAD	6.60e-01	1.35e-01	8.11e-01	4.72e-02	2.55e-01	7.47e-01	7.87e-01	4.76e-01	2.89e-03
8. DLBC	7.55e-01	7.44e-01	3.53e-01	9.82e-01	7.42e-01	8.94e-01	8.15e-01	7.28e-01	4.74e-01
9. ESCA	3.92e-01	3.91e-01	3.92e-01	5.01e-01	3.75e-01	1.71e-01	2.25e-01	4.90e-01	3.30e-01
10. GBM	2.08e-02	8.11e-02	1.49e-04	5.12e-01	1.24e-01	5.37e-01	3.69e-01	7.04e-01	8.75e-05
11. GBMLGG	4.75e-14	6.36e-10	2.31e-17	6.46e-16	8.66e-12	8.04e-14	3.83e-07	1.25e-10	7.48e-17
12. HNSC	3.66e-01	6.19e-01	7.41e-05	2.44e-01	1.42e-01	3.27e-01	9.88e-01	1.55e-01	4.56e-02
13. KICH	7.01e-01	4.63e-01	8.14e-14	0.00e+00	4.03e-01	2.10e-01	8.08e-01	6.61e-01	2.77e-02
14. KIPAN	2.11e-07	9.84e-05	4.81e-08	4.04e-13	2.16e-08	4.21e-08	3.82e-03	4.36e-04	1.16e-11
15. KIRC	6.91e-01	9.79e-01	2.46e-01	1.76e-01	6.70e-01	1.76e-01	1.32e-01	7.29e-01	5.98e-05
16. KIRP	5.33e-03	1.85e-02	8.42e-18	1.00e+00	4.60e-02	5.97e-03	2.49e-02	1.93e-01	1.15e-09
17. LAML	1.73e-03	1.24e-02	5.14e-04	7.00e-01	9.38e-01	1.19e-01	1.75e-02	7.78e-02	8.72e-04
18. LGG	1.60e-14	7.14e-15	1.17e-17	3.52e-01	6.08e-03	1.01e-01	1.16e-09	4.04e-02	4.26e-15
19. LIHC	3.34e-01	1.28e-01	1.09e-03	8.25e-01	2.57e-01	2.93e-01	5.04e-01	8.80e-01	7.04e-01
20. LUAD	5.01e-01	3.73e-01	7.51e-03	5.92e-01	2.55e-02	1.49e-01	2.08e-01	8.21e-03	4.66e-01
21. LUSC	8.71e-02	3.91e-02	1.32e-01	7.04e-01	5.11e-01	9.05e-01	2.88e-01	6.75e-01	8.37e-03
22. MESO	4.24e-04	1.72e-02	7.94e-04	7.29e-02	8.66e-05	2.77e-01	5.53e-04	3.85e-04	7.34e-04
23. OV	4.45e-01	5.88e-01	6.95e-01	9.73e-01	4.35e-01	6.47e-01	7.78e-01	9.60e-01	6.81e-01
24. PAAD	7.36e-04	2.03e-03	1.44e-03	2.96e-03	4.19e-03	4.86e-04	3.18e-01	3.45e-02	2.73e-04
25. PCPG	3.32e-01	4.57e-01	2.57e-01	3.11e-01	3.39e-01	1.41e-01	6.63e-01	7.67e-01	8.66e-01
26. PRAD	4.75e-01	6.95e-01	6.61e-01	9.56e-01	3.73e-01	4.97e-01	7.07e-01	3.90e-01	3.49e-01
27. READ	7.62e-01	3.35e-01	6.27e-01	1.00e+00	5.68e-01	2.72e-01	3.53e-01	3.41e-01	2.35e-02
28. SARC	4.37e-02	5.58e-02	7.23e-02	3.37e-02	3.07e-01	6.36e-01	9.54e-02	2.83e-01	3.03e-02
29. SKCM	4.78e-01	7.54e-05	6.37e-04	4.30e-03	4.67e-03	3.92e-02	1.90e-01	1.48e-03	1.05e-01
30. STAD	4.07e-02	5.11e-01	1.02e-01	4.83e-01	6.25e-01	3.08e-01	3.16e-01	5.55e-01	1.86e-04
31. STES	1.57e-01	3.41e-02	1.18e-01	4.97e-01	4.13e-03	5.92e-01	6.35e-02	8.45e-02	1.51e-02
32. TGCT	8.38e-01	8.39e-01	8.38e-01	5.89e-01	2.96e-01	3.74e-01	5.65e-01	5.41e-01	5.31e-01
33. THCA	6.20e-01	8.62e-03	3.87e-02	5.11e-01	7.42e-01	5.51e-01	3.87e-01	1.75e-02	8.82e-02
34. THYM	9.69e-02	1.15e-01	7.11e-02	8.89e-05	7.06e-02	5.96e-01	5.47e-02	1.38e-01	1.33e-02
35. UCEC	1.81e-02	1.70e-01	1.64e-01	6.88e-01	1.65e-01	8.61e-01	1.58e-02	3.02e-03	4.83e-03
36. UCS	8.59e-01	3.59e-01	7.16e-01	1.68e-01	8.76e-01	8.34e-01	5.85e-01	6.27e-01	4.26e-01
37. UVM	1.67e-04	5.80e-04	1.67e-04	5.50e-01	9.19e-02	4.92e-03	2.06e-04	2.20e-05	6.43e-03
38. M_Discovery	2.26e-05	3.15e-12	1.16e-11	2.87e-01	9.16e-01	4.32e-06	4.59e-10	2.01e-07	3.25e-10
39. M_Validation	1.04e-02	4.68e-06	2.75e-07	1.57e-01	1.97e-01	1.28e-01	7.46e-04	9.16e-04	2.66e-05
#Significant	15	15	19	9	11	8	12	14	28

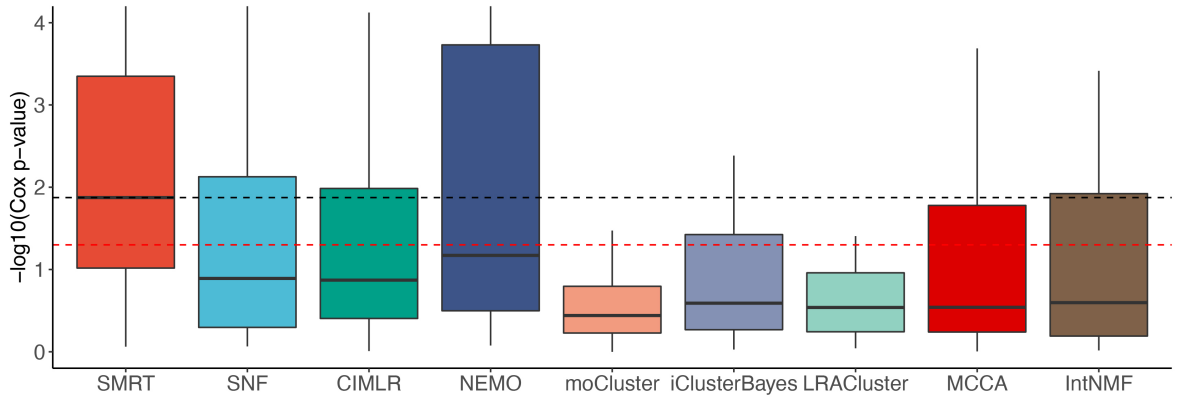

Figure 6.2: Distributions of Cox p-values (in $-log10$ scale, higher is better) of the subtypes discovered from 37 TCGA and 2 METABRIC datasets. The red dashed line shows the 5% significance level. Note that all existing methods do not reach this level of significance on average (median). Overall, the Cox p-values obtained from SMRT are substantially more significant than those of other methods ($p = 0.0002$ using the one-tailed Wilcoxon test).

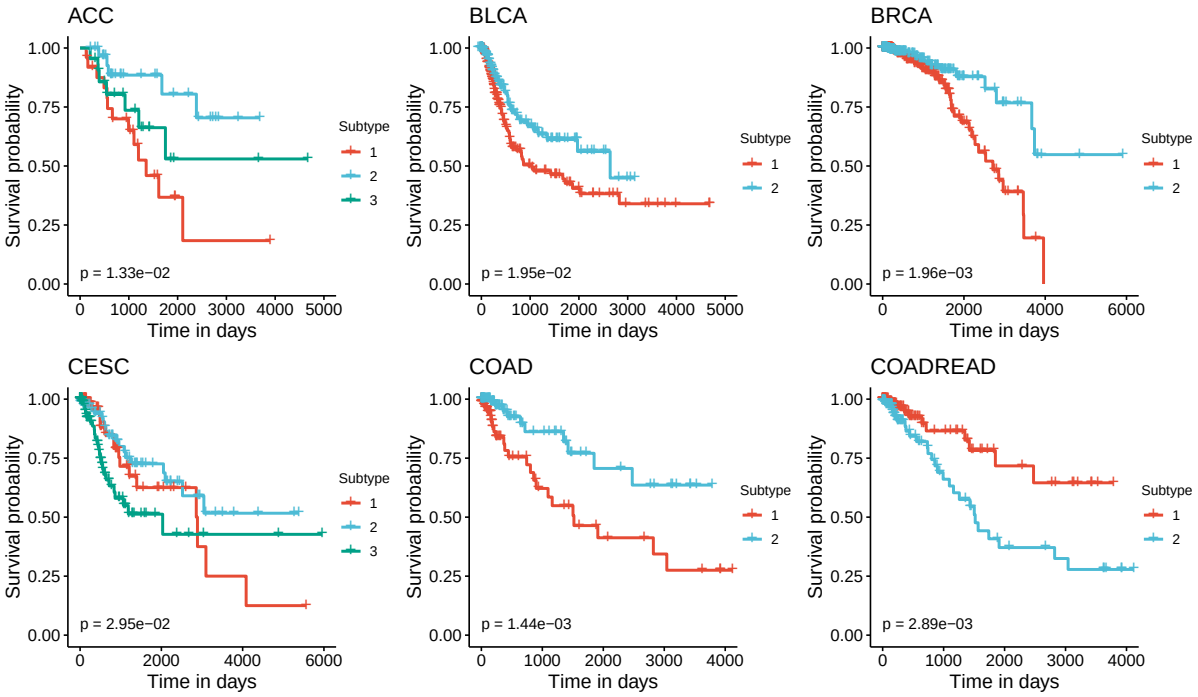

Figure 6.3: Kaplan-Meier survival analysis for TCGA-ACC, BLCA, BRCA, CESC, and COAD datasets.

a minute. The remaining methods are slower, especially iClusterBayes and IntNMF, although their analysis is limited to only 2,000 of the most varied genes.

Table 6.4: Running time (in minutes) of SNF, CIMLR, NEMO, moCluster, iClusterBayes (iCB), LRACluster (LRA), MCCA, IntNMF, and SMRT for 37 TCGA and two METABRIC datasets.

Dataset	Size	SNF	CIMLR	NEMO	moCluster	iCB	LRA	MCCA	IntNMF	SMRT
1. ACC	79	0.40	1.14	0.05	0.97	9.09	5.58	0.50	6.64	0.25
2. BLCA	404	0.73	3.71	0.28	7.85	29.57	34.92	0.83	21.94	1.30
3. BRCA	622	1.61	9.44	0.75	24.09	56.39	102.13	1.61	40.07	1.53
4. CESC	304	1.01	3.23	0.28	8.78	30.49	50.41	1.20	20.66	0.90
5. CHOL	36	0.33	0.60	0.02	0.38	5.23	2.02	0.53	4.77	0.10
6. COAD	220	0.93	1.84	0.20	5.28	23.77	30.81	1.07	16.44	0.67
7. COADREAD	294	0.98	4.41	0.30	9.14	29.81	40.10	1.17	21.07	0.96
8. DLBC	47	0.37	0.61	0.03	0.52	6.25	2.66	0.44	4.90	0.16
9. ESCA	183	0.75	2.44	0.14	4.45	16.91	27.54	0.84	12.93	1.20
10. GBM	273	0.05	2.15	0.02	0.46	20.30	1.02	0.19	15.03	0.91
11. GBMLGG	510	0.89	5.33	0.40	11.61	44.30	41.47	0.97	31.08	1.43
12. HNSC	228	0.84	2.24	0.18	5.41	16.32	32.22	1.06	13.51	0.77
13. KICH	65	0.37	1.13	0.03	0.70	5.93	3.47	0.47	4.93	0.33
14. KIPAN	654	1.14	13.77	0.49	14.90	41.54	63.67	1.16	31.39	3.51
15. KIRC	124	0.04	1.14	0.01	0.15	8.53	0.65	0.09	7.76	0.16
16. KIRP	271	0.61	3.93	0.15	3.96	16.85	18.91	0.70	15.96	0.94
17. LAML	164	0.04	1.57	0.01	0.20	10.84	0.68	0.10	8.13	0.13
18. LGG	510	1.29	7.60	0.60	13.95	33.18	83.92	1.37	28.77	1.76
19. LIHC	366	0.80	3.81	0.28	6.54	23.33	34.19	0.94	20.12	0.84
20. LUAD	428	0.81	4.42	0.28	7.95	34.64	39.17	1.02	29.77	1.26
21. LUSC	110	0.04	1.15	0.00	0.11	7.83	0.46	0.09	6.40	0.12
22. MESO	86	0.42	0.85	0.03	0.88	7.67	5.40	0.60	6.98	0.26
23. OV	286	0.36	2.37	0.10	3.14	19.37	16.24	0.53	16.99	0.72
24. PAAD	178	0.46	1.96	0.08	2.23	11.72	12.25	0.67	8.86	0.98
25. PCPG	179	0.55	2.35	0.12	2.52	15.98	14.51	0.64	11.79	0.52
26. PRAD	493	1.51	6.13	0.54	12.52	33.67	79.05	1.29	32.18	1.75
27. READ	74	0.39	0.86	0.03	0.64	6.32	4.24	0.59	5.88	0.22
28. SARC	257	0.54	3.07	0.14	3.29	18.00	17.82	0.63	12.64	1.40
29. SKCM	439	0.83	6.51	0.34	7.71	27.58	35.17	0.78	23.61	1.76
30. STAD	362	0.87	5.07	0.33	5.77	24.99	34.14	0.89	18.61	1.07
31. STES	545	1.55	8.79	0.53	14.11	37.81	88.00	1.22	28.85	1.85
32. TGCT	134	0.85	1.79	0.10	2.01	10.61	18.49	0.93	7.01	0.41
33. THCA	499	1.06	5.90	0.46	8.85	33.01	53.59	0.92	25.35	1.66
34. THYM	119	0.49	0.97	0.07	1.18	8.78	9.76	0.52	7.16	0.28
35. UCEC	234	1.04	2.57	0.19	4.60	19.61	34.42	1.08	14.78	0.88
36. UCS	56	0.47	0.64	0.04	0.49	6.18	3.92	0.62	4.58	0.19
37. UVM	80	0.41	0.73	0.04	0.61	7.91	5.27	0.60	6.25	0.24
38. M_Discovery	997	0.38	17.96	0.21	7.10	60.24	16.17	0.38	49.62	2.42
39. M_Validation	983	0.37	10.14	0.19	6.85	58.11	17.95	0.40	50.87	2.28
Mean	305	0.68	3.96	0.21	5.43	22.53	27.75	0.76	17.80	0.98

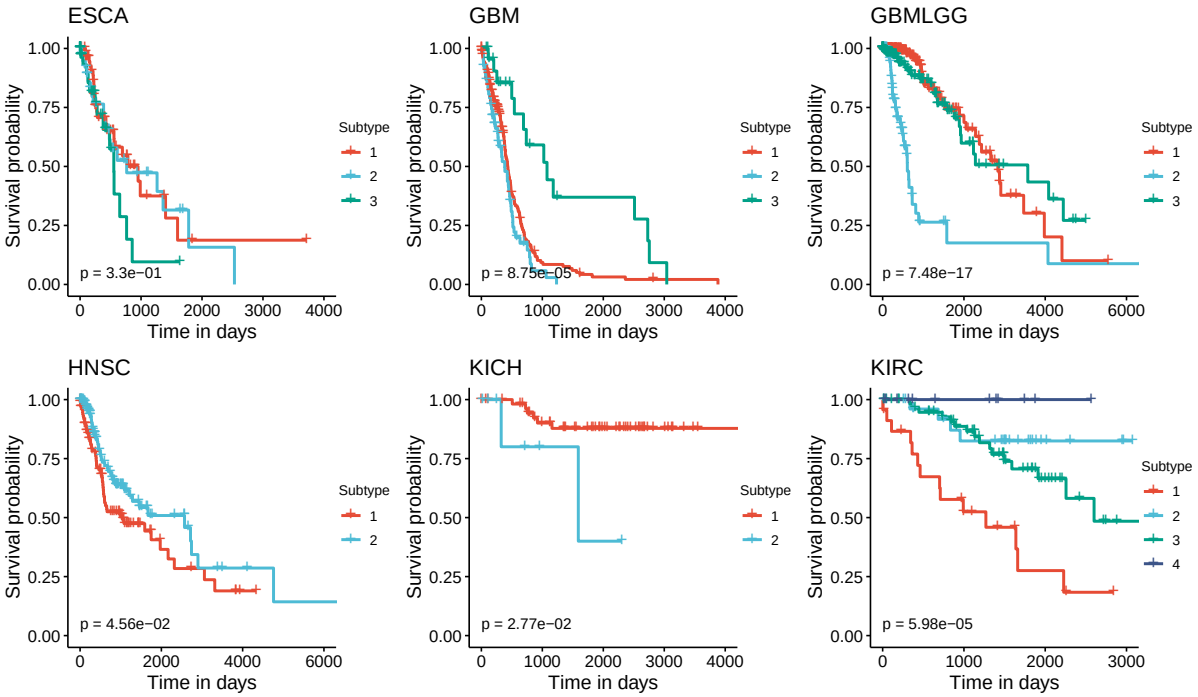

Figure 6.4: Kaplan-Meier survival analysis for TCGA-ESCA, GBM, GBMLGG, HNSC, KICH, and KIRC datasets.

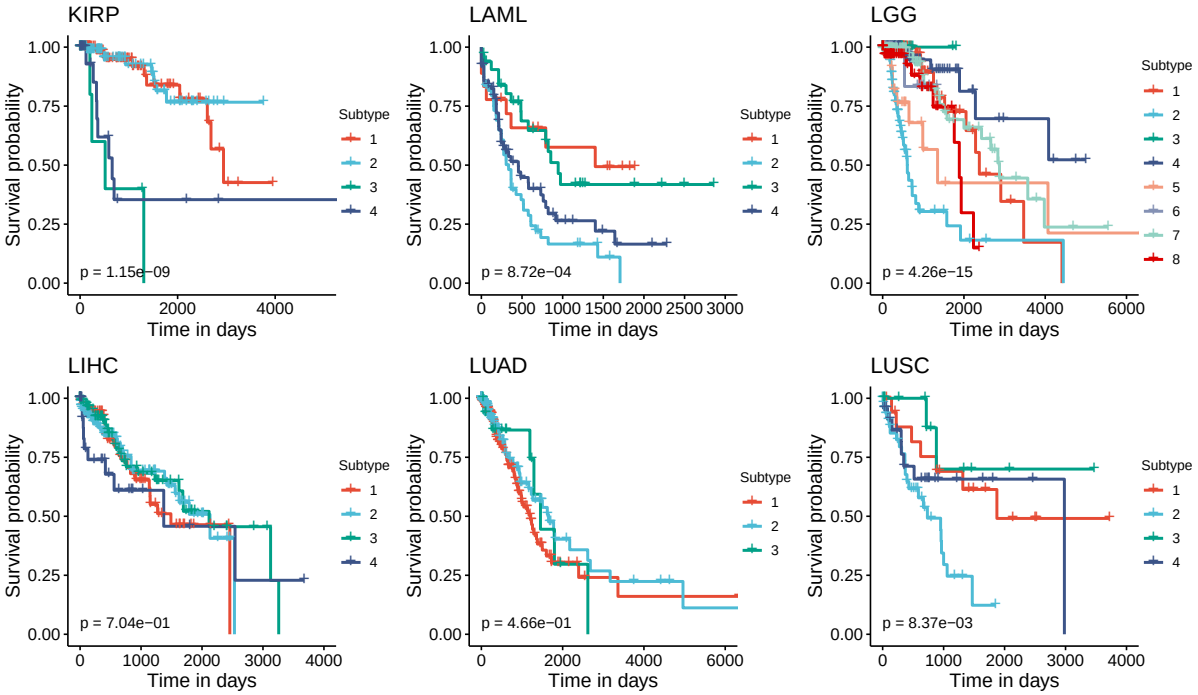

Figure 6.5: Kaplan-Meier survival analysis for TCGA-KIRP, LAML, LGG, LIHC, LUAD, and LUSC datasets.

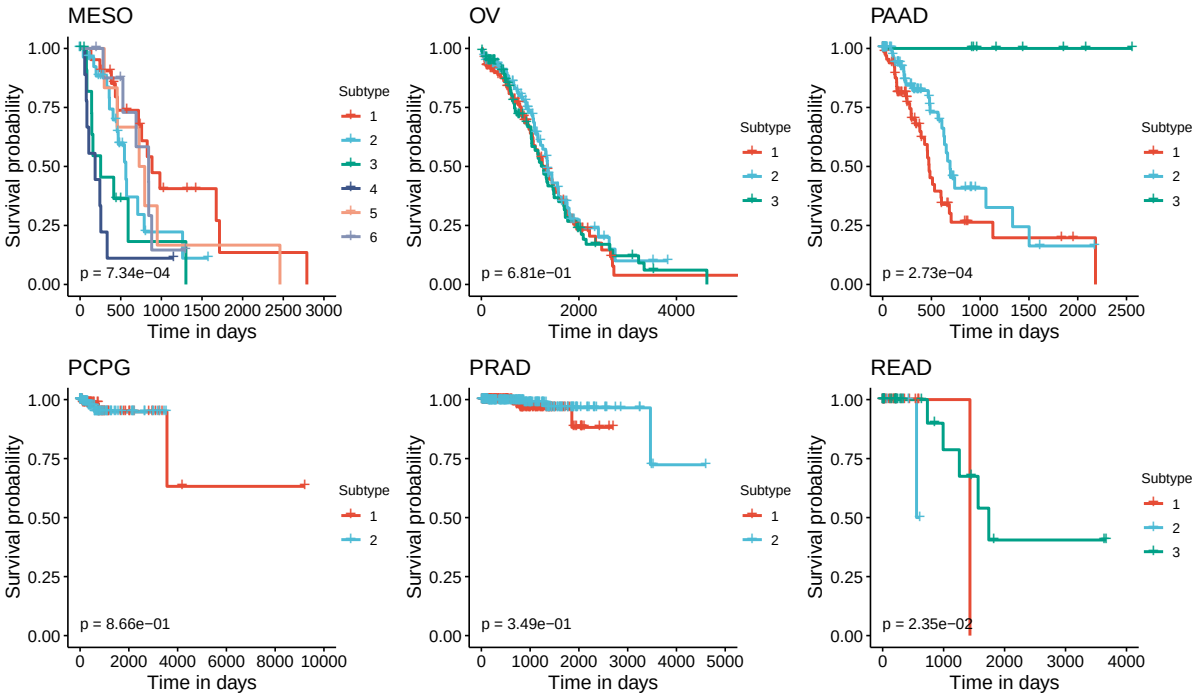

Figure 6.6: Kaplan-Meier survival analysis for TCGA-MESO, OV, PADD, PCPG, PRAD, and READ datasets.

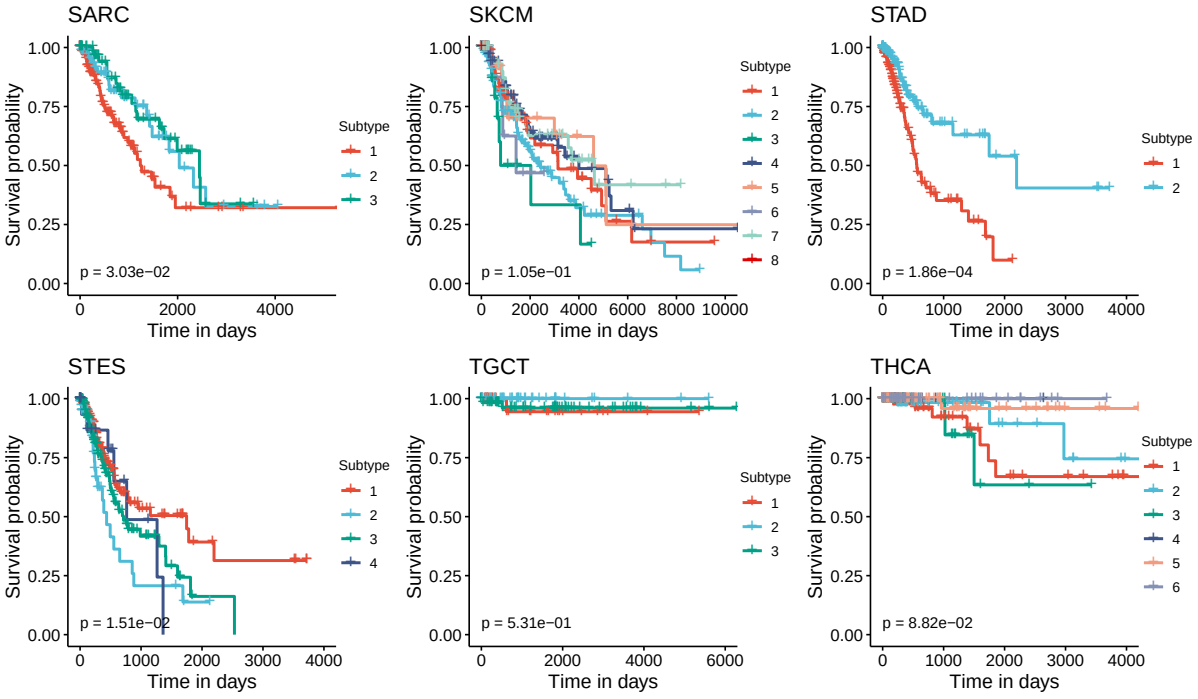

Figure 6.7: Kaplan-Meier survival analysis for TCGA-SARC, SKCM, STAD, STES, TGCT, and THCA datasets.

6.3.2 Analysis of subtypes from SMRT and PAM50 on Breast cancer datasets

In this analysis, we used the PAM50 classifier implemented in *genefu* R package [140] to classify patients using gene expression data in the three breast cancer datasets: TCGA-BRCA and

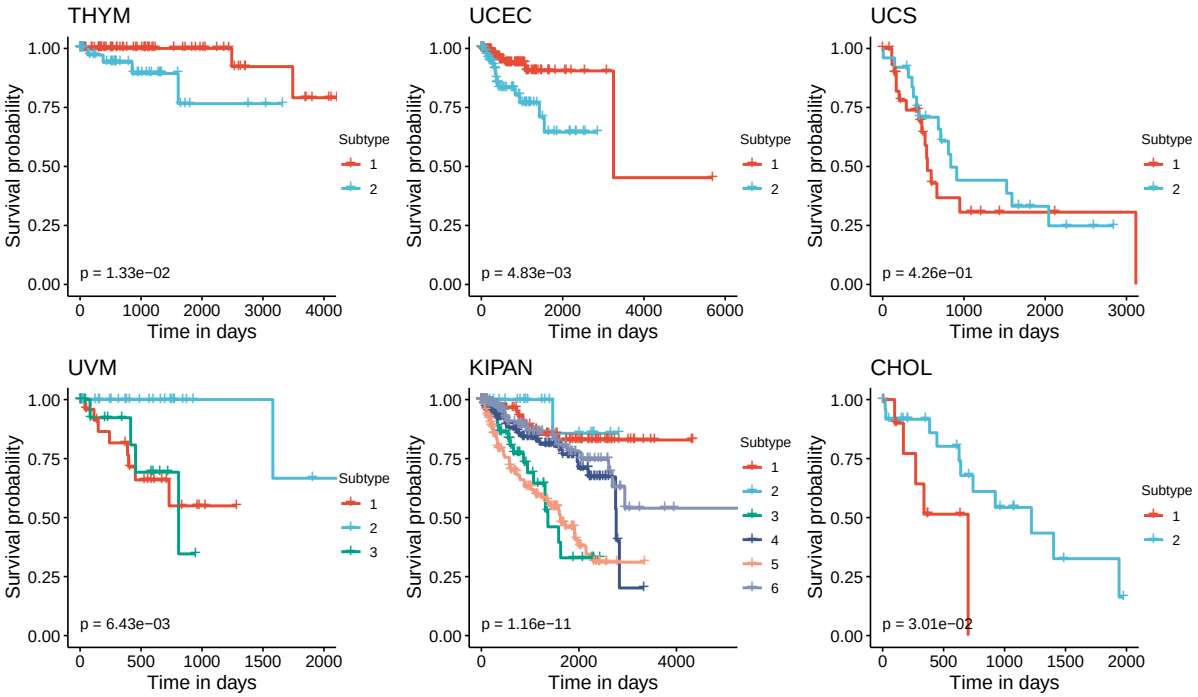

Figure 6.8: Kaplan-Meier survival analysis for TCGA-THYM, UCEC, UCS, UVM, KIPAN, and CHOL datasets.

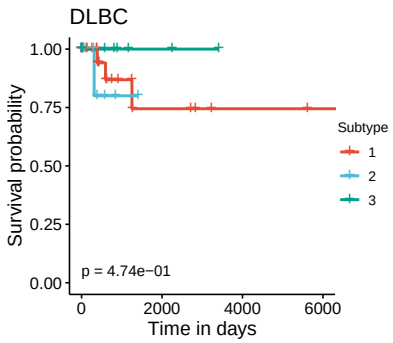

Figure 6.9: Kaplan-Meier survival analysis for TCGA-DLBC dataset.

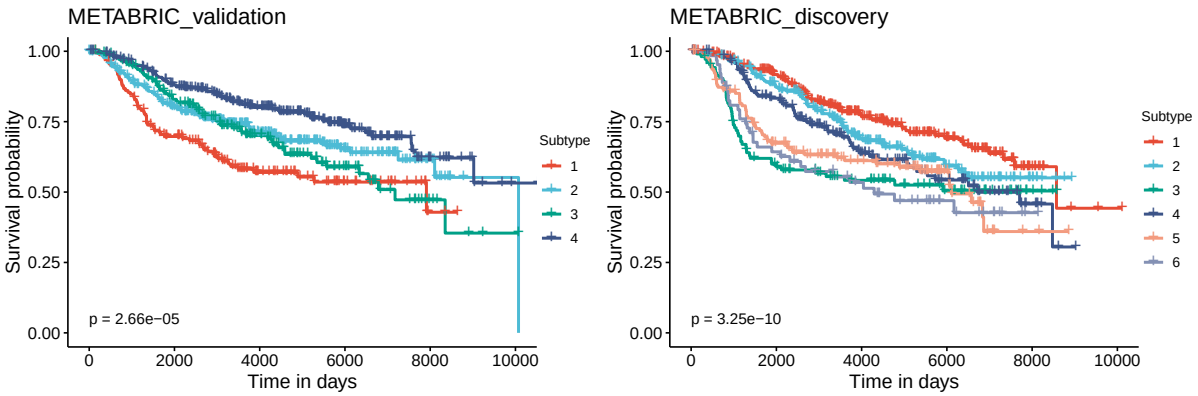

Figure 6.10: Kaplan-Meier survival analysis for METABRIC Validation and Discovery datasets.

the two METABRIC datasets. Tables 6.5, 6.6 and 6.7 show the confusion matrix between subtypes determined by PAM50 and those discovered by SMRT. Overall, both SMRT and PAM50 are able to identify patient subgroups with significantly different survival profiles in all three datasets. Note that the Cox p-values of SMRT are more significant than PAM50 in two out of three datasets: 1) 0.002 (SMRT) vs. 0.009 (PAM50) for TCGA-BRCA, and 2) 3.25e-10 (SMRT) vs. 8.32e-7 (PAM50) for METABRIC_Discovery. For the METABRIC_Validation dataset, PAM50 has a marginally more significant p-value (1.77e-05 vs 2.66e-05).

For the TCGA-BRCA dataset, SMRT discovers two subtypes. Most of PAM50's Normal-like and LumA samples fall into group 1 of SMRT. Most LumB are clustered into group 2. The two remaining PAM50 subtypes (Her2 and Basal) are split equally to SMRT's two clusters. Note that the subtypes discovered by SMRT have a more significant Cox p-value than PAM50's for this dataset (0.002 vs. 0.009)

For the METABRIC Discovery dataset, SMRT discovers 6 subtypes. Most samples of Normal-like, LumA, LumB, and Her2 are grouped into groups 1, 2, 3, and 4 of SMRT, respectively. Basal samples are split equally into groups 5 and 6. For this dataset, the Cox p-value of SMRT is more significant than that of PAM50 (3.25e-10 vs. 8.32e-7).

For the METABRIC Validation dataset, SMRT discovers 4 subtypes. Most Basal and Her2 samples belong to groups 1 and 4, respectively. The remaining subtypes (Normal-like, LumA, and LumB) are almost evenly distributed across SMRT's 4 groups. For this dataset, the Cox p-value of PAM50 is slightly more significant than that of SMRT (1.77e-5 vs. 2.66e-5).

Table 6.5: Confusion matrix between PAM50 subtypes and level-1 subtypes discovered by SMRT for TCGA-BRCA. The Cox p-values of subtypes obtained from PAM50 and SMRT are 0.009 and 0.002, respectively.

PAM50/SMRT	1	2
Normal-like	17	4
LumA	108	27
LumB	12	100
Her2	29	36
Basal	149	140

Table 6.6: Confusion matrix between PAM50 subtypes and level-1 subtypes discovered by SMRT for METABRIC Discovery dataset. Cox p-values obtained from PAM50 and SMRT are $8.32e-07$ and $3.25e-10$, respectively.

PAM50/SMRT	1	2	3	4	5	6
Normal-like	11	0	2	8	0	2
LumA	38	75	1	0	1	0
LumB	59	24	14	10	6	8
Her2	14	1	9	173	98	24
Basal	21	5	37	95	129	132

Table 6.7: Confusion matrix between PAM50 subtypes and level-1 subtypes discovered by SMRT for METABRIC Validation dataset. Cox p-values obtained from PAM50 and SMRT are $1.77e-05$ and $2.66e-05$, respectively.

PAM50/SMRT	1	2	3	4
Normal-like	19	20	24	69
LumA	92	114	61	69
LumB	2	5	57	88
Her2	4	1	0	41
Basal	168	35	18	96

6.3.3 Case study of the GBMLGG dataset

Here, we perform an in-depth analysis of the GBMLGG (Glioma). Figure 6.12A shows the Kaplan–Meier survival analysis of the discovered subtypes. For this dataset, SMRT discovers three subtypes in which one subtype (group 2) has a very low survival rate where at year 3, the survival probability of patients this group is only at 26% while that number for the patients in the other two subtypes (groups 1 and 3) is 84%. Figure 6.11 shows the age distribution of each subtype, in which patients in Subtype 2 (low survival) are older than patients in Subtypes 1 and 3 (high survival).

We also perform a variant analysis for the dataset in order to find mutations that highly occur in the short-term-survival patient group (group 2) but not in the long-term-survival patient group (groups 1 and 3) and vice versa. Figure 6.12B shows the mutations of each group in which each point is a gene, and its coordinates represent the number of patients that have that mutation in the corresponding group. In principle, we want to investigate the mutated genes in the top left or bottom right of the figure. In this figure, we can easily identify four marker genes that are associated with GBMLGG disease: IDH1, TP53, PTEN, and EGFR. Among those, IDH mutant (bottom-right) is known as a factor driving Low-Grade Glioma (LGG) and has

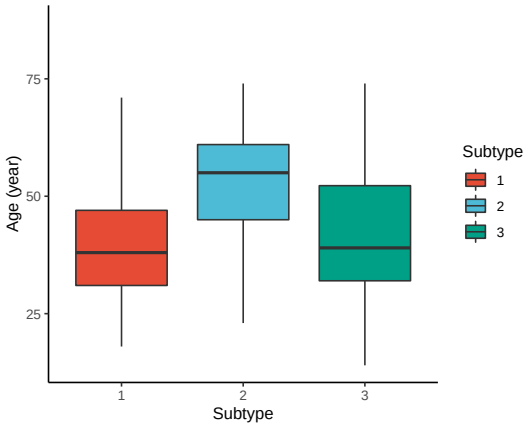

Figure 6.11: Age distribution for each subtype of the GBMLGG dataset.

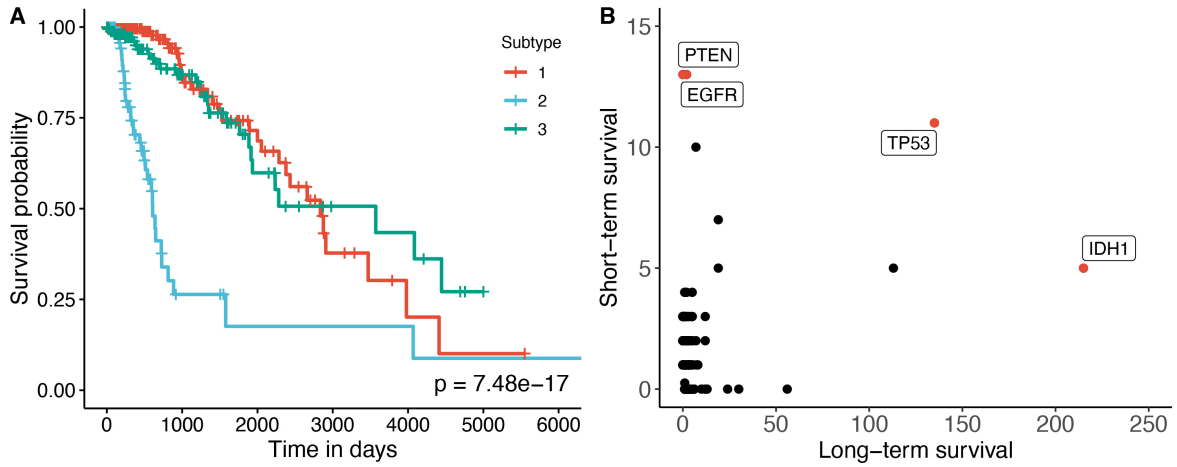

Figure 6.12: A) Kaplan–Meier survival analysis of the GBMLGG dataset. The horizontal axis represents the time (days) while the vertical axis represents the estimated survival probability. B) Number of patients in each group for each mutated gene in GBMLGG dataset. The horizontal axis shows the count for other subtypes with high survival rates, and the vertical axis represents the count in the subtype with low survival rates.

been used in the WHO classification system [141] to classify IDH-mutant and IDH-wildtype, which has worse prognoses. On the other hand, EGFR is not a common mutation in LGG but in GBM (Glioblastoma) [142] which has a very low survival rate [143]. The amplification of EGFR can cause the mutation of PTEN gene [144], which is a tumor suppressor gene [145]. Interestingly, no patient in the long-term survival group has a PTEN mutation. The occurrence of EGFR-mutated genes may be another cause of the low survival rate of patients in the short-term survival group.

We further investigated the contribution of genes and data types using this dataset as a case study. For this dataset, SMRT identified three subtypes using multi-omics data with a

p-value of 7.48e-17. First, we map the features in miRNA data and Methylation data onto genes. For miRNA data, the mapping is done using the miRTarBase database. For methylation data, we map the methylation probes to their corresponding gene using probe positions in the reference genome. Next, for each of the three data types, we use an ANOVA test to calculate the significance of each gene. This results in 3 p-values for each gene. We then combine these p-values using Fisher's method, adjust them for multiple comparisons using false discovery rate (FDR), and rank them according to their p-values.

Next, we performed a gene set analysis using the whole ranked list of gene and KEGG pathways. For this purpose, we used the FGSEA method [146] implemented in our web-based platform named Consensus Pathway Analysis (CPA) [1]. Figure 6.13 shows the pathways that are significant with a significance threshold of 0.5%. In this connected network, each node is a pathway, and there is an edge between two pathways if they have common genes. As shown in the figure, the Glioma pathway is significantly impacted. Other pathways that have common components with the Glioma pathway, including the MAPK signaling pathway, ErbB signaling pathway, Calcium signaling pathway, and Pathway in cancer, are also significantly impacted. This confirms that the subtypes discovered by SMRT have significant differences in the activity of Glioma- and cancer-related pathways. In other words, genes that belong to these pathways significantly contribute to differentiating the three subtypes. In fact, when we intersected the list of significantly differentially expressed genes from the pathways above, 5 out of 9 of the intersected list are oncogenes, including EGFR, PDGFA, PDGFB, PDGFRA, and PDGFRB [147, 148, 149, 150, 151, 152, 153] (see Table 6.8 for the p-value and description of each gene).

6.3.4 Contribution of indivivual omic types

To reveal the contribution of each data type, we used SMRT to partition the patients using each of the data types independently. Next, we calculated the Cox p-values obtained from each data type and compared them with those obtained from subtyping the multi-omics data. Figure 6.14 shows the distribution of -log10 p-values of subtypes by each data type for 37 TCGA datasets. The p-values obtained from multi-omics data are substantially more significant

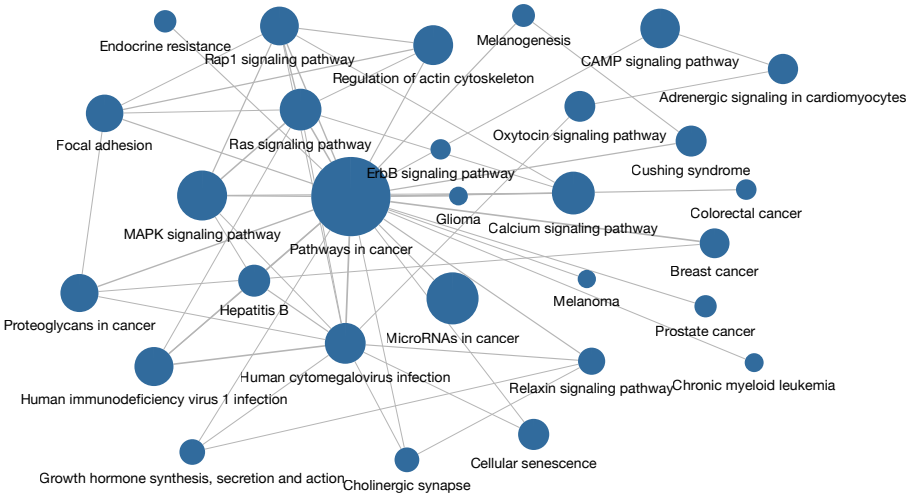

Figure 6.13: The largest connected component of the significant impacted pathways network resulted from pathway analysis on the subtypes discovered by SMRT using GBMLGG dataset.

Table 6.8: The common signficantly differential expressed genes and their p-value from the Glioma pathway, MAPK signaling pathway, Calcium signaling pathway, and Pathway in cancer.

Gene	Description	p-value.FDR
EGF	Epidermal growth factor	7.29e-43
EGFR	Epidermal growth factor receptor	2.95e-37
PDGFA	Platelet derived growth factor subunit A	1.17e-46
PDGFB	Platelet derived growth factor subunit B	6.85e-20
PDGFRA	Platelet derived growth factor receptor alpha	1.49e-49
PDGFRB	Platelet derived growth factor receptor beta	1.45e-30
PRKCA	Protein kinase C alpha	1.02e-46
PRKCB	Protein kinase C beta	4.60e-73
PRKCG	Protein kinase C gamma	2.38e-50

than those obtained from individual data types. The median p-value obtained from multi-omics data is close to 0.01 (-log10 values are close to 2), while the median p-values of each data type are even higher than 0.1 (-log10 values are close to 1). This demonstrates that SMRT is able to exploit the complementary information available in each data type to determine subtypes with significant survival differences.

Table 6.9 reports the Cox p-values obtained for each data type of the 37 TCGA datasets. The data shows that mRNA plays a very important role in subtyping ACC, BLCA, LAML, MESO, PAAD, SARC, and SKCM datasets. In these cancers, subtypes discovered from mRNA data have more significant Cox p-values than those from other data types (methylation and miRNA). For BLCA, LAML, SARC, and SKCM, mRNA is the only data type for which SMRT

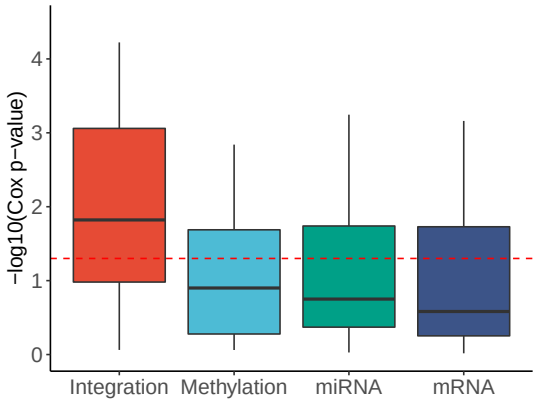

Figure 6.14: Distribution of -log10 Cox p-values for each data type of the 37 TCGA datasets. The horizontal red line indicates the significant threshold of $p - value = 0.05$. The p-values of subtypes discovered using multi-omics integration are substantially more significant than those obtained from individual data types (mRNA, methylation, miRNA).

can discover subtypes with significant survival differences. The second data type, DNA methylation, is very important for CHOL, GBM, GBMLGG, KICH, KIRP, LGG, THYM, and UCEC. Subtyping using methylation yields more significant Cox p-values than mRNA and miRNA. In fact, methylation is the only data type that provides significant Cox p-values among the three data types for CHOL, KICH, and UCEC. The third data type, miRNA, is important for BRCA, CESC, COAD, COADREAD, KIPAN, PAAD, READ, STES, and UVM. The Cox p-values of miRNA are more significant than those of mRNA and methylation in these datasets.

While each data type contributes differently to the integrated subtypes in each dataset, it is clear that the number of datasets with significant p-values for individual data types is substantially smaller than that obtained from data integration. These numbers are 12, 14, and 13 for mRNA, methylation, and miRNA, respectively, compared to 26 for data integration. More importantly, the p-values obtained from data integration are more significant in most of those datasets (20 out of the 26 significant datasets). In some datasets (e.g., HNSC, KIRC, LUSC), none of the data types provide sufficient information to determine subtypes with significantly different survivals. However, when we integrate these data types, SMRT is able to exploit the complementary information available in each data type to determine subtypes with significant survival differences.

Table 6.9: Cox p-values of clustering results by SMRT for each data type of 37 TCGA datasets.

Dataset	mRNA	Methylation	miRNA	Integration
1. ACC	3.55e-03	2.81e-02	3.81e-01	1.33e-02
2. BLCA	1.95e-02	7.11e-02	7.30e-02	1.95e-02
3. BRCA	3.81e-01	4.02e-01	1.96e-03	1.96e-03
4. CESC	2.90e-01	4.88e-02	2.95e-02	2.95e-02
5. CHOL	9.62e-01	3.01e-02	5.56e-01	3.01e-02
6. COAD	6.13e-01	3.05e-01	1.44e-03	1.44e-03
7. COADREAD	5.82e-01	8.68e-01	2.89e-03	2.89e-03
8. DLBC	7.28e-01	4.12e-01	9.37e-01	4.74e-01
9. ESCA	2.58e-01	5.27e-01	3.92e-01	3.30e-01
10. GBM	4.08e-01	1.25e-04	5.19e-02	8.75e-05
11. GBMLGG	2.16e-13	3.29e-16	8.26e-03	7.48e-17
12. HNSC	2.84e-01	5.83e-01	2.61e-01	4.56e-02
13. KICH	1.88e-01	1.02e-04	1.88e-01	2.77e-02
14. KIPAN	2.58e-06	4.25e-02	1.16e-07	1.16e-11
15. KIRC	1.76e-01	1.11e-01	1.38e-01	5.98e-05
16. KIRP	2.38e-03	1.24e-05	4.45e-02	1.15e-09
17. LAML	3.47e-03	4.42e-01	7.24e-02	8.72e-04
18. LGG	1.13e-04	3.29e-16	8.88e-16	4.26e-15
19. LIHC	2.62e-01	2.86e-01	6.83e-01	7.04e-01
20. LUAD	1.25e-01	8.49e-01	4.16e-01	4.66e-01
21. LUSC	1.25e-01	1.57e-01	1.17e-01	8.37e-03
22. MESO	6.69e-03	2.05e-02	1.96e-02	7.34e-04
23. OV	8.01e-01	1.22e-01	6.76e-01	6.81e-01
24. PAAD	6.91e-04	1.44e-03	6.91e-04	2.73e-04
25. PCPG	7.44e-01	8.66e-01	4.09e-01	8.66e-01
26. PRAD	4.97e-01	7.25e-01	8.93e-01	3.49e-01
27. READ	6.49e-01	6.27e-01	8.37e-03	2.35e-02
28. SARC	4.06e-02	8.17e-02	7.35e-01	3.03e-02
29. SKCM	5.16e-03	7.69e-01	1.78e-01	1.05e-01
30. STAD	8.43e-01	4.12e-01	3.71e-01	1.86e-04
31. STES	3.78e-01	1.66e-01	1.82e-02	1.51e-02
32. TGCT	3.89e-01	5.31e-01	5.90e-01	5.31e-01
33. THCA	5.59e-01	1.26e-01	4.93e-01	8.82e-02
34. THYM	1.87e-02	5.60e-03	1.87e-01	1.33e-02
35. UCEC	2.12e-01	4.83e-03	5.92e-01	4.83e-03
36. UCS	8.34e-01	8.13e-01	4.26e-01	4.26e-01
37. UVM	3.37e-01	2.53e-03	5.69e-04	6.43e-03

6.3.5 Clinical variables enrichment analysis

Next, we investigated the association between discovered subtypes and clinical variables. We performed our analysis on gender, age, cancer stage, and tumor grade, which are available for at least 15 datasets. We perform the following analyses: (1) Fisher's exact test to assess the

significance of the association between gender (male and female) and the discovered subtypes; (2) ANOVA to assess the age difference between discovered subtypes; and finally (3) calculate the agreement between the discovered subtypes and known cancer stages and tumor grades using Normalized Mutual Information (NMI). The distributions of $-log10$ of p-values for gender and age are shown in Figures 6.15 (see Tables 6.10- 6.11 for the exact p-values). With the exception of NEMO and iClusterBayes, the clustering methods do not generally yield differences in gender or age in their clustering. For gender, iClusterBayes has significant p-values in 17 out of 31 datasets. For age, NEMO and iClusterBayes have significant p-values in 17 and 15 out of 29 datasets, respectively. This result demonstrates that there are meaningful and survival-related molecular signatures inside the data to be discovered, and the methods do not simply separate patients based on some visible clinical variables such as gender or age. Figure 6.16 and Tables 6.12- 6.13 show the NMI values that represent the agreement between the discovered subtypes and known cancer stages and tumor grades. For the cancer stage, the median NMI values of SMRT and NEMO are comparable and are higher than the rest. For tumor grade, SMRT has the highest median NMI. However, for both cancer stage and tumor grade, the NMI values of all methods are low, meaning that there is a low agreement between the known stages/grades and the discovered subtypes using any of the subtyping methods. In conclusion, the discovered subtypes from SMRT and other subtyping methods have little agreement with clinical variables like gender, age, cancer stage, and tumor grade.

6.3.6 Simulation studies

In this simulation study, we generate multiple datasets with varying numbers of genes and samples. The general setup is that each dataset consists of three classes (equal size), each with a different set of up-regulated genes. Figure 6.17 shows an example dataset of size $1,000 \times 5,000$ (1,000 samples and 5,000 genes). This dataset has three classes (sample size of 333, 333, and 334), each with a different set of 100 genes that are up-regulated. The first class has the first 100 genes up-regulated; the second class has the second 100 genes up-regulated; the third class has the third 100 genes up-regulated. The expression values of un-regulated genes follow a

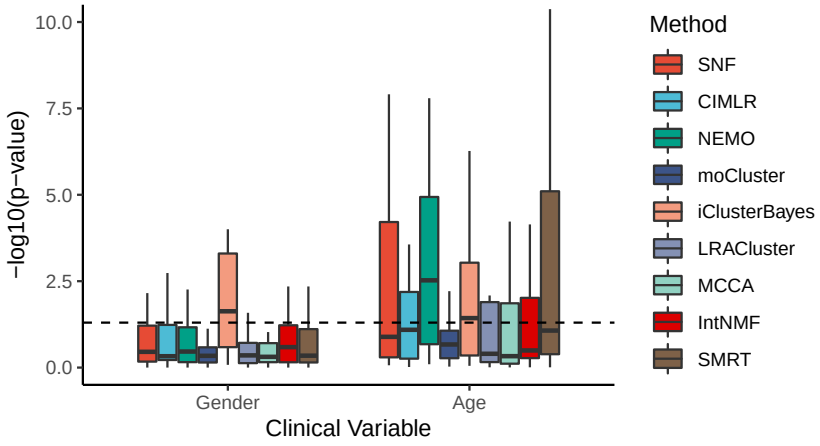

Figure 6.15: P-values obtained from comparing the discovered subtypes against gender, and age. Fisher's exact test was used to assess the statistical significance in the association between the discovered subtypes and gender while ANOVA was used to assess age difference. The horizontal axis shows the clinical variables while the vertical axis shows the minus $log10$ p-values. The horizontal dashed line denotes minus $log10$ of $p = 0.05$. With the exception of NEMO and iClusterBayes, the clustering methods do not generally yield differences in gender or age in their clustering.

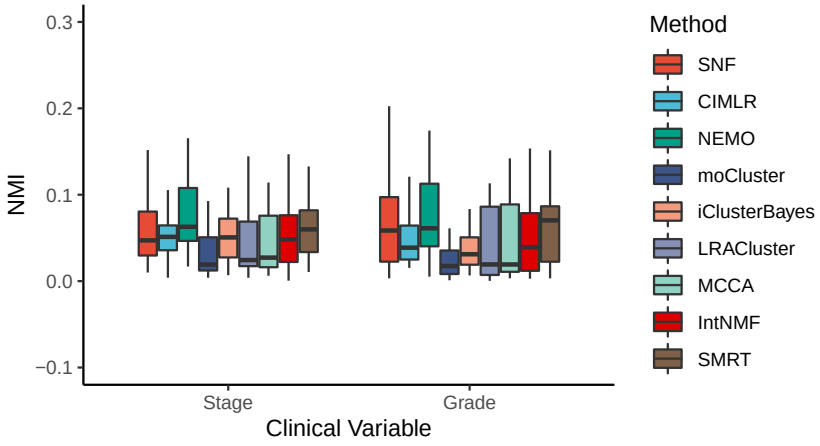

Figure 6.16: Normalized Mutual Information (NMI) values obtained from comparing the discovered subtypes against known cancer stages (left panel) and tumor grades (right panel). The clustering methods do not generally yield subtypes that are correlated with cancer stages and tumor grades.

distribution of $\mathcal{N}(0, 1)$ ($\mu_0 = 0$, $\sigma = 1$) while the expression values of up-regulated genes follow a distribution of $\mathcal{N}(2, 1)$ ($\mu_{DE} = 2$, $\sigma = 1$).

To test the scalability of the subtyping methods. We fix the number of genes (five thousand) but vary the number of samples (from 1,000 to 100,000). For each dataset, we use the nine subtyping methods, SNF, CIMLR, NEMO, moCluster, iClusterBayes, LRACluster, MCCA, IntNMF, and SMRT, to cluster the data. We monitor the running time and memory

Table 6.10: P-values obtained from Fisher's exact test that assesses the statistical significance of the association between the discovered subtypes and gender. NA indicates that there is not enough data to perform the test or all patients have the same gender.

Dataset	SNF	CIMLR	NEMO	moCluster	iClusterBayes	LRACluster	MCCA	IntNMF	SMRT
ACC	1.1e-01	1.0e+00	2.2e-01	8.4e-01	7.7e-01	1.7e-01	6.5e-01	6.4e-02	3.8e-01
BLCA	2.6e-01	5.4e-02	2.8e-01	6.3e-01	5.0e-04	2.1e-01	1.3e-01	2.6e-01	5.0e-01
BRCA	2.1e-01	8.1e-02	2.1e-01	2.7e-02	4.9e-02	3.8e-01	2.1e-01	5.6e-02	1.0e+00
CESC	NA	NA	NA	NA	NA	NA	NA	NA	NA
CHOL	3.4e-01	1.0e+00	7.2e-01	1.0e+00	2.6e-02	7.3e-01	4.8e-01	8.9e-02	4.8e-01
COAD	7.3e-01	5.0e-01	3.3e-01	7.8e-01	5.0e-04	1.0e+00	2.7e-01	5.0e-01	6.8e-01
COADREAD	7.4e-01	9.4e-01	4.3e-01	1.7e-01	5.0e-04	5.4e-01	5.4e-01	8.1e-01	1.0e+00
DLBC	1.0e+00	3.0e-01	9.8e-01	2.5e-01	2.5e-01	5.9e-01	5.5e-01	1.0e+00	7.8e-01
ESCA	1.0e+00	8.3e-01	1.0e+00	6.5e-01	8.3e-01	1.0e+00	1.0e+00	1.0e+00	1.0e+00
GBM	7.7e-01	3.5e-01	1.0e+00	1.6e-01	1.3e-02	1.3e-01	8.1e-01	3.0e-02	5.0e-04
GBMLGG	3.6e-01	7.6e-01	3.3e-01	4.0e-01	5.0e-04	1.0e+00	5.0e-01	2.4e-01	5.4e-01
HNSC	9.6e-03	5.8e-01	3.3e-02	1.0e+00	2.8e-01	4.5e-01	9.4e-02	5.5e-01	1.8e-01
KICH	2.0e-01	1.0e+00	4.7e-01	5.1e-01	5.3e-02	1.7e-01	6.1e-01	9.4e-02	4.5e-01
KIPAN	4.5e-02	5.7e-02	2.8e-02	1.0e+00	5.0e-04	3.4e-02	3.6e-01	7.5e-03	2.5e-03
KIRC	2.7e-01	6.6e-07	1.7e-02	3.0e-01	2.7e-01	3.0e-01	1.8e-01	1.3e-01	5.0e-04
KIRP	1.1e-03	7.0e-05	5.0e-04	4.6e-01	5.0e-04	1.2e-04	1.7e-05	5.0e-04	2.3e-02
LAML	4.3e-01	4.2e-01	1.8e-01	6.2e-01	8.6e-02	1.0e+00	5.2e-01	7.5e-01	9.4e-01
LGG	3.7e-01	4.9e-01	3.7e-01	2.5e-02	5.0e-04	4.4e-01	8.9e-01	5.3e-04	3.3e-01
LIHC	2.9e-05	5.0e-04	5.0e-04	5.8e-01	1.0e-03	7.1e-01	1.6e-03	1.7e-01	2.0e-01
LUAD	1.0e-06	5.0e-04	5.0e-04	5.9e-04	5.0e-04	5.0e-01	1.0e+00	5.2e-02	3.4e-02
LUSC	9.1e-02	5.0e-04	5.2e-01	1.0e+00	1.9e-01	4.3e-01	3.7e-01	3.9e-01	7.6e-02
MESO	7.6e-01	7.7e-02	1.2e-01	1.9e-01	7.0e-01	7.6e-01	5.0e-01	3.8e-01	1.0e+00
OV	NA	NA	NA	NA	NA	NA	NA	NA	NA
PAAD	5.0e-03	1.4e-01	3.8e-01	6.8e-02	3.1e-01	7.1e-01	8.5e-49	9.5e-01	3.3e-01
PCPG	5.3e-01	7.3e-01	7.6e-01	1.0e+00	5.0e-04	2.4e-01	4.5e-01	8.8e-01	1.0e+00
PRAD	NA	NA	NA	NA	NA	NA	NA	NA	NA
READ	1.0e+00	5.6e-01	2.5e-01	3.7e-01	1.5e-03	3.0e-01	9.5e-01	1.0e+00	5.9e-01
SARC	5.0e-04	1.5e-03	5.0e-04	4.6e-01	5.0e-04	1.5e-05	1.7e-03	7.7e-04	2.5e-03
SKCM	4.1e-01	5.3e-01	7.3e-01	5.5e-01	4.7e-02	1.4e-01	4.9e-01	4.0e-01	1.5e-03
STAD	5.6e-01	6.3e-01	9.1e-01	2.7e-01	1.7e-01	3.4e-01	1.0e+00	1.9e-01	1.2e-01
STES	6.0e-04	1.8e-03	3.5e-03	4.6e-01	9.9e-05	5.6e-02	3.1e-03	5.1e-04	9.1e-02
TGCT	NA	NA	NA	NA	NA	NA	NA	NA	NA
THCA	3.7e-01	4.4e-01	7.1e-01	5.5e-01	6.0e-01	7.9e-01	7.5e-01	7.9e-02	5.7e-01
THYM	6.4e-01	5.2e-01	1.9e-01	3.3e-01	6.0e-03	1.0e+00	2.4e-01	8.5e-01	7.1e-01
UCEC	NA	NA	NA	NA	NA	NA	NA	NA	NA
UCS	NA	NA	NA	NA	NA	NA	NA	NA	NA
UVM	1.0e+00	5.4e-01	1.0e+00	1.0e+00	6.4e-01	9.7e-01	1.0e+00	6.5e-01	7.5e-01
METABRIC validation	NA	NA	NA	NA	NA	NA	NA	NA	NA
METABRIC discovery	NA	NA	NA	NA	NA	NA	NA	NA	NA
# significant	8	7	8	3	17	3	5	6	7

usage of each analysis. To assess the accuracy of the clustering methods, we compare the clustering results with the ground truth (known class label) using the Adjusted Rand Index (ARI) [128]. The ARI takes values from -1 to 1, with the ARI expected to be 1 for a perfect agreement, and 0 for random clustering results.

Figure 6.18 shows the running time of the methods with varying numbers of samples. The detailed running time is shown in Tables 6.14. The time complexity of SNF, CIMLR, NEMO, and moCluster increases exponentially with respect to sample size. These methods are not able to analyze datasets with more than 30,000 samples (out of memory, produce errors, or take

Table 6.11: P-values obtained from ANOVA test that assesses the statistical significance of the association between the discovered subtypes and age. NA indicates that there is not enough data to perform the test.

Dataset	SNF	CIMLR	NEMO	moCluster	iClusterBayes	LRACluster	MCCA	IntNMF	SMRT
ACC	NA	NA	NA	NA	NA	NA	NA	NA	NA
BLCA	6.6e-03	7.1e-03	1.8e-07	2.0e-02	2.8e-02	2.8e-02	1.4e-02	8.4e-02	5.5e-02
BRCA	2.0e-01	2.7e-04	8.6e-02	2.1e-01	2.3e-05	1.3e-01	9.1e-02	7.1e-02	6.4e-02
CESC	3.8e-01	1.3e-01	5.9e-01	8.4e-01	5.9e-09	4.0e-01	7.7e-01	4.3e-01	7.6e-01
CHOL	NA	NA	NA	NA	NA	NA	NA	NA	NA
COAD	5.4e-01	2.2e-02	2.1e-01	9.3e-02	2.2e-01	8.5e-01	4.7e-01	3.8e-01	8.1e-01
COADREAD	7.6e-01	6.1e-01	8.0e-01	8.1e-01	6.4e-01	7.2e-01	6.8e-01	5.3e-01	5.2e-01
DLBC	8.6e-01	8.3e-01	7.1e-01	5.4e-01	6.5e-01	7.4e-01	8.7e-01	5.9e-01	4.1e-01
ESCA	NA	NA	NA	NA	NA	NA	NA	NA	NA
GBM	1.4e-02	2.9e-02	1.1e-03	4.1e-01	5.2e-05	9.0e-03	9.9e-01	9.6e-03	4.2e-11
GBMLGG	1.2e-17	2.8e-16	1.7e-17	8.7e-08	5.4e-07	4.5e-09	7.6e-10	7.3e-07	1.5e-13
HNSC	5.3e-01	5.6e-02	2.6e-03	9.4e-01	7.5e-01	5.5e-01	8.3e-01	7.3e-01	6.6e-01
KICH	3.0e-01	8.1e-02	1.4e-01	1.3e-01	7.3e-01	3.3e-01	2.2e-01	4.0e-01	3.3e-01
KIPAN	1.2e-08	8.8e-08	6.5e-07	8.6e-02	5.1e-10	3.9e-08	5.1e-01	1.1e-08	5.8e-08
KIRC	6.1e-01	5.9e-01	3.1e-01	5.6e-01	8.9e-01	5.6e-01	6.2e-01	7.8e-01	1.8e-02
KIRP	2.3e-01	9.6e-01	1.2e-05	1.1e-01	2.9e-02	9.8e-01	9.5e-01	5.5e-04	2.1e-03
LAML	6.1e-05	1.1e-01	2.2e-05	6.1e-03	5.8e-02	1.3e-02	6.3e-03	5.3e-03	7.9e-06
LGG	1.9e-18	3.3e-16	2.2e-19	6.7e-01	9.2e-04	8.8e-03	5.2e-13	6.1e-01	1.9e-09
LIHC	3.3e-05	6.5e-03	3.6e-04	2.9e-01	4.6e-03	9.0e-01	6.0e-05	9.1e-01	2.1e-04
LUAD	1.5e-02	2.7e-02	6.0e-02	3.1e-02	2.2e-02	3.3e-01	9.0e-01	2.3e-02	5.8e-01
LUSC	8.0e-01	4.0e-01	5.8e-01	2.0e-01	6.8e-01	6.9e-01	6.3e-01	5.1e-01	2.7e-01
MESO	NA	NA	NA	NA	NA	NA	NA	NA	NA
OV	1.3e-01	1.3e-01	2.0e-01	6.9e-01	1.1e-05	9.2e-01	4.4e-04	2.3e-01	9.8e-01
PAAD	5.0e-01	5.5e-01	4.4e-01	2.0e-01	3.4e-01	9.1e-02	8.7e-01	4.8e-01	1.6e-01
PCPG	NA	NA	NA	NA	NA	NA	NA	NA	NA
PRAD	1.6e-01	5.7e-01	6.0e-04	4.8e-01	1.6e-02	4.6e-01	3.1e-01	4.7e-01	8.5e-02
READ	7.8e-01	8.2e-01	2.2e-01	1.7e-01	1.7e-01	4.6e-01	2.5e-02	9.8e-01	8.2e-01
SARC	NA	NA	NA	NA	NA	NA	NA	NA	NA
SKCM	6.1e-03	1.4e-01	1.0e-03	2.0e-02	3.7e-04	8.2e-03	8.5e-01	7.2e-05	1.5e-01
STAD	1.2e-04	6.0e-03	3.8e-03	2.9e-02	5.3e-01	5.0e-01	4.0e-02	1.9e-02	1.3e-01
STES	5.1e-01	8.8e-01	3.9e-02	4.6e-01	4.5e-01	4.9e-01	7.2e-01	9.6e-01	3.6e-01
TGCT	NA	NA	NA	NA	NA	NA	NA	NA	NA
THCA	9.5e-02	1.3e-02	3.0e-03	5.6e-01	2.9e-01	3.4e-01	6.4e-01	3.2e-01	4.5e-04
THYM	NA	NA	NA	NA	NA	NA	NA	NA	NA
UCEC	1.3e-07	4.1e-01	1.6e-08	2.8e-01	3.7e-02	8.7e-01	3.4e-02	7.5e-02	3.6e-06
UCS	NA	NA	NA	NA	NA	NA	NA	NA	NA
UVM	NA	NA	NA	NA	NA	NA	NA	NA	NA
METABRIC validation	1.1e-16	1.6e-15	7.0e-14	3.1e-01	4.2e-01	1.1e-14	1.6e-19	1.0e-19	2.2e-15
METABRIC discovery	1.9e-17	1.5e-03	4.7e-17	3.3e-02	2.3e-02	3.2e-14	3.3e-13	1.1e-12	6.1e-20
# significant	13	13	17	7	15	9	11	10	12

more than 24 hours to analyze a single dataset). MCCA and LRACluster are able to analyze datasets with 50,000 samples but fail to analyze larger datasets. Only iCB and SMRT were able to analyze all datasets. However, it takes iCB more than three days to finish the analysis of the largest dataset. SMRT is much faster than other methods. It takes SMRT less than three minutes to analyze the biggest datasets while the running time of others increases exponentially. In addition, ARI values of SMRT are maintained at 1 for all datasets (Table 6.15).

Table 6.12: Normalized Mutual Information (NMI) vlaues obtained from comparing the discovered subtypes against known cancer stages. NA indicates that there is not enough data to perform the calculation.

Dataset	SNF	CIMLR	NEMO	moCluster	iClusterBayes	LRACluster	MCCA	IntNMF	SMRT
ACC	0.1	0.13	0.26	0.17	0.03	0.05	0.11	0.05	0.06
BLCA	0.03	0.04	0.06	0	0.02	0.03	0.05	0.03	0.04
BRCA	0.02	0.04	0.02	0.02	0.02	0.01	0.01	0.01	0.02
CESC	0.03	0.06	0.02	0.01	0.05	0.02	0.03	0.03	0.06
CHOL	0.08	0.16	0.23	0.18	0.21	0.12	0.18	0.22	0.13
COAD	0.06	0.05	0.05	0.02	0.06	0.02	0.02	0.03	0.03
COADREAD	0.04	0.05	0.03	0.03	0.06	0.02	0.02	0.03	0.01
DLBC	0.07	0.02	0.27	0.02	0.07	0.08	0.04	0.05	0.06
ESCA	0.09	0.09	0.09	0.09	0.09	0.08	0.09	0.09	0.08
GBM	NA	NA	NA	NA	NA	NA	NA	NA	NA
GBMLGG	NA	NA	NA	NA	NA	NA	NA	NA	NA
HNSC	0.01	0.04	0.07	0.03	0.02	0.02	0.02	0.02	0.01
KICH	0.1	0.04	0.24	0.11	0.09	0.02	0.01	0.07	0.06
KIPAN	0.06	0.05	0.06	0.01	0.03	0.06	0.02	0.05	0.07
KIRC	0.04	0	0.06	0.02	0.02	0.02	0.02	0.08	0.09
KIRP	0.1	0.07	0.12	0.01	0.07	0.1	0.08	0.07	0.13
LAML	NA	NA	NA	NA	NA	NA	NA	NA	NA
LGG	NA	NA	NA	NA	NA	NA	NA	NA	NA
LIHC	0.02	0.03	0.05	0.01	0.03	0.01	0.02	0.01	0.03
LUAD	0.01	0.03	0.06	0.01	0.03	0.02	0.01	0.03	0.02
LUSC	0.04	0.05	0.06	0.03	0.04	0.06	0.04	0.05	0.07
MESO	0.02	0.06	0.12	0.01	0.06	0.02	0.09	0.08	0.12
OV	0.05	0.04	0.05	0.02	0.05	0.02	0.02	0.02	0.04
PAAD	0.08	0.06	0.1	0.09	0.1	0.1	0.02	0.09	0.09
PCPG	NA	NA	NA	NA	NA	NA	NA	NA	NA
PRAD	NA	NA	NA	NA	NA	NA	NA	NA	NA
READ	0.17	0.14	0.17	0.06	0.11	0.14	0.21	0.15	0.12
SARC	NA	NA	NA	NA	NA	NA	NA	NA	NA
SKCM	0.03	0.05	0.08	0.02	0.02	0.02	0.01	0.05	0.07
STAD	0.05	0.02	0.02	0.01	0.04	0.02	0.02	0.02	0.02
STES	0.05	0.05	0.06	0.03	0.05	0.02	0.05	0.04	0.04
TGCT	0.09	0.08	0.09	0.07	0.1	0.06	0.08	0.08	0.1
THCA	0.03	0.05	0.04	0.01	0.03	0	0.03	0	0.05
THYM	NA	NA	NA	NA	NA	NA	NA	NA	NA
UCEC	0.04	0.05	0.03	0.04	0.05	0.03	0.04	0.06	0.05
UCS	0.15	0.11	0.13	0.06	0.2	0.08	0.11	0.16	0.08
UVM	0.08	0.11	0.08	0.02	0.08	0.09	0.07	0.06	0.07
METABRIC validation	0.02	0.02	0.03	0	0.01	0.02	0.01	0.01	0.03
METABRIC discovery	0.02	0.01	0.02	0.01	0.01	0.01	0.01	0.02	0.03
Mean	0.06	0.06	0.09	0.04	0.06	0.04	0.05	0.06	0.06

6.3.7 Performance of KNN with fixed K and using Elbow method

In this analysis, we compare the performance of KNN with a fixed number of neighbors (knn.k = 10) and using the Elbow method to determine k in terms of accuracy, memory usage, and running time using both simulated and real data. Table 6.16 shows the memory usage and running time using simulated datasets. In this simulation, we generate 12 datasets with a fixed number of genes (5,000) and varying numbers of samples from 3,000 to 100,000. We set the

Table 6.13: Normalized Mutual Information (NMI) values obtained from comparing the discovered subtypes against known tumor grades. NA indicates that there is not enough data to perform the calculation.

Dataset	SNF	CIMLR	NEMO	moCluster	iClusterBayes	LRACluster	MCCA	IntNMF	SMRT
ACC	NA	NA	NA	NA	NA	NA	NA	NA	NA
BLCA	0.1	0.09	0.12	0.02	0.02	0.09	0.09	0.06	0.07
BRCA	NA	NA	NA	NA	NA	NA	NA	NA	NA
CESC	0.01	0.04	0.01	0.01	0.04	0	0.01	0.01	0.03
CHOL	NA	NA	NA	NA	NA	NA	NA	NA	NA
COAD	NA	NA	NA	NA	NA	NA	NA	NA	NA
COADREAD	NA	NA	NA	NA	NA	NA	NA	NA	NA
DLBC	NA	NA	NA	NA	NA	NA	NA	NA	NA
ESCA	NA	NA	NA	NA	NA	NA	NA	NA	NA
GBM	NA	NA	NA	NA	NA	NA	NA	NA	NA
GBMLGG	0.06	0.06	0.06	0.06	0.08	0.1	0.01	0.08	0.07
HNSC	0.02	0.04	0.06	0.04	0.03	0.02	0.03	0.01	0.01
KICH	NA	NA	NA	NA	NA	NA	NA	NA	NA
KIPAN	0.02	0.02	0.03	0.02	0.05	0.02	0.02	0.01	0.07
KIRC	0.1	0.02	0.13	0.11	0.08	0.11	0.11	0.11	0.11
KIRP	NA	NA	NA	NA	NA	NA	NA	NA	NA
LAML	NA	NA	NA	NA	NA	NA	NA	NA	NA
LGG	0.06	0.06	0.06	0	0.02	0	0.01	0.01	0.03
LIHC	0.01	0.03	0.03	0.01	0.05	0	0.01	0	0.02
LUAD	NA	NA	NA	NA	NA	NA	NA	NA	NA
LUSC	NA	NA	NA	NA	NA	NA	NA	NA	NA
MESO	NA	NA	NA	NA	NA	NA	NA	NA	NA
OV	0.02	0.02	0.02	0.03	0.02	0.02	0.02	0.01	0.02
PAAD	0.1	0.06	0.08	0.09	0.07	0.08	0.02	0.04	0.08
PCPG	NA	NA	NA	NA	NA	NA	NA	NA	NA
PRAD	NA	NA	NA	NA	NA	NA	NA	NA	NA
READ	NA	NA	NA	NA	NA	NA	NA	NA	NA
SARC	NA	NA	NA	NA	NA	NA	NA	NA	NA
SKCM	NA	NA	NA	NA	NA	NA	NA	NA	NA
STAD	0.05	0.04	0.05	0.01	0.03	0.01	0.02	0.05	0
STES	0	0.02	0.05	0.02	0.01	0.01	0	0.01	0.01
TGCT	NA	NA	NA	NA	NA	NA	NA	NA	NA
THCA	NA	NA	NA	NA	NA	NA	NA	NA	NA
THYM	NA	NA	NA	NA	NA	NA	NA	NA	NA
UCEC	0.2	0.05	0.17	0.01	0.03	0	0.14	0.15	0.15
UCS	NA	NA	NA	NA	NA	NA	NA	NA	NA
UVM	NA	NA	NA	NA	NA	NA	NA	NA	NA
METABRIC validation	0.1	0.12	0.12	0.01	0.01	0.06	0.09	0.08	0.1
METABRIC discovery	0.09	0.1	0.11	0.01	0.01	0.11	0.11	0.09	0.09
Mean	0.06	0.05	0.07	0.03	0.04	0.04	0.05	0.05	0.06

dataset size to trigger the sampling process is 2,000 (e.g., if the dataset size is 3,000, then the size of the sampled set is 2,000, and the size of the propagated set is 1,000). It is clear that there is no difference in memory usage between the two cases. It is expected that using the Elbow method to determine k is slower than using a predefined k. However, the difference is marginal. We note that in this simulation, all ARI values are 1.

Table 6.17 in shows the memory usage and running time using 27 TCGA datasets. In these datasets, KNN is used to classify patients who do not have data for all data types. Again, the

Table 6.14: Running time (in minutes) of the subtyping methods for simulations with 5,000 genes and varying numbers of samples (1,000 to 100,000). SNF, CIMLR, NEMO, and moCluster were not able to analyze datasets with more than 30,000 samples (out of memory). LRACluster and MCCA are unable to finish the largest dataset with 100,000 samples. Only iCB and SMRT were able to analyze all datasets. SMRT can cluster 100,000 samples in under three minutes

#Samples	SNF	CIMLR	NEMO	moCl.	iCB	LRACl.	MCCA	IntNMF	SMRT
1000	0.05	9.35	0.04	1.12	57.44	1.05	0.42	NA	0.30
2000	0.20	36.08	0.19	5.71	110.12	5.12	1.18	219.57	0.87
5000	2.06	338.86	2.05	43.94	261.61	49.94	2.43	826.91	0.94
10000	13.23	2033.37	13.19	141.47	534.02	86.29	5.06	NA	0.98
20000	89.95	NA	94.21	507.81	1081.94	125.43	10.60	NA	1.18
30000	NA	NA	NA	1386.72	1607.05	169.87	16.42	NA	1.23
50000	NA	NA	NA	NA	2660.04	253.52	28.37	NA	1.54
100000	NA	NA	NA	NA	5121.43	NA	NA	NA	2.28

Table 6.15: The accuracy of the clustering results measured by ARI for simulations with 5,000 genes and varying numbers of samples (1,000 to 100,000).

#Samples	SNF	CIMLR	NEMO	moCl.	iCB	LRACl.	MCCA	IntNMF	SMRT
1000	1.00	1.00	1.00	1.0000	0.86	1.00	1.00	NA	1.00
2000	1.00	1.00	1.00	1.0000	1.00	1.00	0.57	1.00	1.00
5000	1.00	1.00	1.00	1.0000	1.00	1.00	1.00	0.29	1.00
10000	1.00	1.00	1.00	1.0000	0.00	1.00	1.00	NA	1.00
20000	1.00	NA	1.00	1.0000	0.00	1.00	0.57	NA	1.00
30000	NA	NA	NA	1.0000	1.00	1.00	1.00	NA	1.00
50000	NA	NA	NA	NA	0.00	1.00	0.57	NA	1.00
100000	NA	NA	NA	NA	0.00	NA	NA	NA	1.00

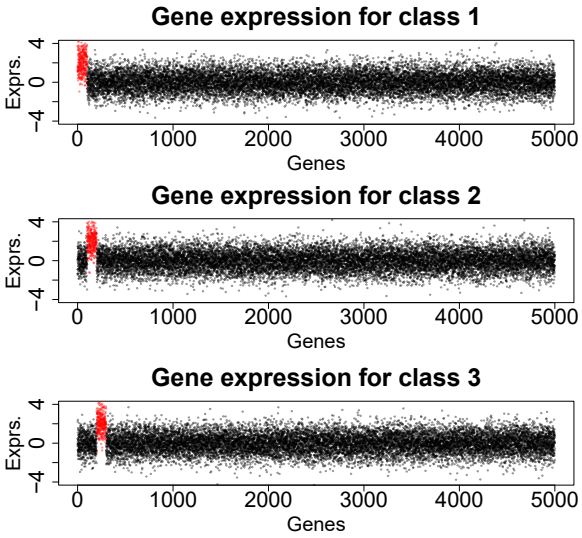

Figure 6.17: An example simulation. The dataset is represented as a matrix with size of $1,000 \times 5,000$ (1,000 samples and 5,000 features) divided into three classes of equal number of samples. Each class has a different set of 100 up-regulated genes (marked in red).

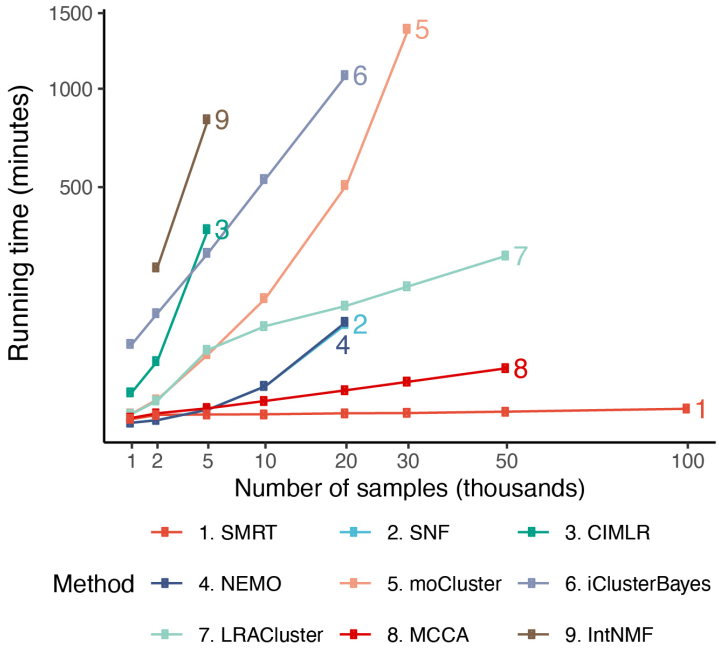

Figure 6.18: Running time of the nine subtyping methods with respect to varying numbers of samples and features. SMRT is the only method that can analyze all datasets. Even for large datasets with 100,000 samples, SMRT needs only a couple of minutes to finish the analysis.

differences in memory usage and running time between the two cases are marginal. Regarding accuracy, the two implementations have the sample Cox p-values in 13 datasets. Among the remaining 14, Cox p-values of $k = 10$ are better in 4 datasets, and Cox p-values of k determined

by the Elbow method are better in 10 datasets. Overall, using k determined by the Elbow method gives a better accuracy.

Table 6.16: Memory usage and running time of SMRT on simulation, for two settings of KNN: (1) fixed number of neighbors ($knn.k = 10$) and (2) the number of neighbors is determined by the Elblow method. The two methods produce comparable results.

#Samples	Memory (GB)		Running Time (minutes)	
	knn.k = 10	Elbow	knn.k = 10	Elbow
3,000	1.88	1.88	1.21	1.39
4,000	2.10	2.10	1.24	1.46
5,000	2.33	2.33	1.22	1.48
6,000	2.13	2.13	1.21	1.49
7,000	2.42	2.42	1.22	1.50
8,000	2.49	2.49	1.24	1.51
9,000	2.68	2.68	1.27	1.53
10,000	2.39	2.39	1.28	1.54
20,000	3.36	3.36	1.38	1.67
30,000	5.15	5.15	1.58	1.81
50,000	8.72	8.72	1.91	2.24
100,000	17.67	17.67	2.47	2.92

6.4 Conclusion (SMRT)

In this study, we introduced SMRT, a fast yet accurate method for data integration and subtype discovery. In an extensive analysis using 39 cancer datasets, we showed that SMRT outperformed other state-of-the-art methods in discovering novel subtypes with significantly different survival profiles. We also demonstrated that the method could accurately partition hundreds of thousands of samples in minutes with low memory requirements. At the same time, the provided web application will be extremely useful for life scientists who lack computational background or resources. Although the software was developed for the purpose of cancer subtyping, researchers in other fields can use the R package for unsupervised learning and data integration.

Table 6.17: Cox p-values, memory usage and running time of SMRT on TCGA data, for two settings of KNN: (1) fixed number of neighbors ($knn.k = 10$) and (2) the number of neighbors is determined by the Elblow method. All of 27 datasets contains patients that have data of some but not all data types. Cells highlighted in yellow have significant Cox p-values at the threshold of 5%. Cells highlighted in green have the most significant Cox p-value in their respective rows. The two methods produce comparable results.

Dataset	Size		Cox p-values		Memory (GB)		Running Time (m)	
	Match	Total	knn.k = 10	Elbow	knn.k = 10	Elbow	knn.k = 10	Elbow
1. ACC	79	80	7.74e-03	7.74e-03	2.75	2.75	0.52	0.46
2. BLCA	404	411	1.74e-02	1.74e-02	8.89	8.89	2.37	2.10
3. BRCA	622	1095	1.33e-02	1.76e-02	21.93	21.93	3.07	3.32
4. CESC	304	307	3.20e-02	3.20e-02	9.28	9.28	1.91	1.70
5. COAD	220	301	4.54e-03	1.37e-03	7.02	7.02	1.61	1.17
6. COADREAD	294	401	7.90e-03	4.85e-03	9.27	9.27	2.03	1.94
7. DLBC	47	48	4.69e-01	4.69e-01	1.90	1.90	0.45	0.37
8. ESCA	183	185	5.01e-01	5.01e-01	5.33	5.33	2.62	2.37
9. GBMLGG	510	754	0.00e-00	0.00e-00	14.04	14.04	1.76	2.25
10. HNSC	228	527	6.86e-02	4.15e-02	10.42	8.68	1.70	1.52
11. KIPAN	654	887	9.70e-14	3.06e-13	13.41	16.10	4.66	4.62
12. KIRP	271	288	1.87e-09	1.58e-09	6.49	5.95	1.64	1.66
13. LGG	510	514	3.93e-15	3.92e-15	13.83	13.83	3.01	2.93
14. LIHC	366	376	6.96e-01	7.27e-01	8.98	8.98	1.24	1.51
15. LUAD	428	500	7.71e-01	7.71e-01	9.60	9.74	2.04	2.15
16. OV	286	571	8.69e-01	5.22e-01	4.56	4.56	1.12	1.10
17. PAAD	178	184	1.79e-04	1.79e-04	4.57	4.57	1.16	1.17
18. PRAD	493	498	3.32e-01	3.32e-01	12.74	12.74	2.91	2.92
19. READ	74	100	2.45e-03	2.45e-03	5.39	5.39	0.34	0.35
20. SARC	257	261	4.11e-02	4.11e-02	6.15	6.23	1.98	1.78
21. SKCM	439	461	9.90e-02	8.92e-02	10.9	10.9	2.60	2.17
22. STAD	362	431	3.89e-03	6.44e-04	9.50	9.06	1.68	1.59
23. STES	545	616	3.77e-02	3.81e-02	16.29	16.29	3.13	3.10
24. THCA	499	502	8.76e-02	8.74e-02	11.3	11.28	2.47	2.47
25. THYM	119	123	1.11e-02	1.11e-02	6.89	7.45	0.56	0.46
26. UCEC	234	541	2.92e-04	3.28e-05	9.55	9.55	1.19	1.34
27. UCS	56	57	3.85e-01	3.85e-01	3.88	3.81	0.38	0.32

Part II

Pathway Analysis

Chapter 7

Pathway Analysis: Significance and Challenges

When the human genome was first sequenced, it was hailed as a major milestone in the field of biology [154]. It was expected to revolutionize our understanding of the genetic basis of diseases and pave the way for the development of new treatments and therapies. However, the reality has turned out to be more complex. While the human genome has provided valuable insights into the genetic basis of diseases, it has also revealed the complexity of biological systems [155].

The human genome is made up of approximately 20,000 genes, which encode the proteins that carry out the functions of the cell [156]. A single gene can have multiple variants, and the expression of these genes can be influenced by a wide range of factors, including environmental conditions, lifestyle, and genetic predisposition [157]. Typically, to understand the genetic basis of a disease or how a cell responds to an external stimulus, researchers measure the expression of thousands of genes simultaneously. By comparing the gene expression profiles of different conditions, researchers can identify the genes that are differentially expressed and gain insights into the underlying biological mechanisms [158]. Such comparative analysis is called differential gene expression analysis and is a fundamental step in gaining insights into the genetic basis of diseases and phenotypes [159].

However, the interpretation of the results of differential gene expression analysis is not straightforward. Typically, thousands of genes are differentially expressed, and it is not feasible to interpret the biological significance of each gene individually. Moreover, genes do not act in isolation but are part of a complex network of interactions [160]. These interactions, often referred to as pathways, are responsible for the regulation of various biological processes, such

as cell growth, metabolism, and immune response [161]. Disruptions in these pathways can lead to the development of diseases such as cancer, diabetes, and heart disease [162, 163, 164]. Therefore, analyzing the impact of different conditions on these pathways is a more informative approach than analyzing individual genes. Such an approach is called pathway analysis [165]. The conditions under which pathway analysis is applied can be diverse. They can range from a state of disease, such as cancer or diabetes, to a cellular response to an external stimulus, such as exposure to a drug or a change in environmental conditions [166, 167]. In each case, the goal is to identify the pathways or gene sets that are significantly impacted, providing a deeper understanding of the biological processes at play.

To conduct pathway analysis, researchers use statistical techniques to pinpoint pathways or gene sets that are significantly influenced by a particular biological condition. In other words, a prior list of pathways or gene sets must be provided for the pathway analysis method. These pathways are often curated from a vast array of biological literature and databases. Researchers have developed many knowledge bases and databases that contain information about the function, location, and other properties of the genes and gene products. One of the first such knowledge bases was the Gene Ontology (GO) [168]. GO consists of a controlled vocabulary of terms that describe biological processes, cellular locations, and biochemical functions, as well as the relationships between them. These together form an ontology. Furthermore, GO also provides associations between genes and these terms, thus capturing the knowledge about the gene functions and localization within the cell.

As soon as such annotations started to become available, analysis methods have been developed to take advantage of them. The first analysis approach was the over-representation analysis (ORA), which identifies the gene sets, such as GO terms, that are enriched in differentially expressed (DE) genes [169, 170, 171]. The drawbacks of ORA approaches include that they: (i) only consider the number of DE genes and ignore the actual expression changes, and (ii) assume that genes are independent, which is not true. Functional Class Scoring (FCS) approaches have been developed to address these drawbacks. These include the Gene Set Enrichment Analysis (GSEA) family of methods [172, 173, 174].

The main improvement is that these approaches can identify situations in which small but coordinated changes in the expression of functionally related genes are important. While GO captures the associations between genes and various biological processes, cellular locations, and biochemical functions, it does not provide any direct information about the interactions between the genes and/or gene products. Basically, each GO term can be seen as an unordered, unstructured set of genes that are associated with it. The next step was to try to describe the complex phenomena that take place in living organisms by describing the various signals and interactions between genes, gene products, and/or metabolites. These are captured in directed graphs that are commonly referred to as pathways. Examples include Kyoto Encyclopedia of Genes and Genomes (KEGG) [161] and Reactome [175]. Pathways can be further divided into gene signaling pathways and metabolic pathways. In gene signaling pathways, the nodes represent genes, and the edges represent signals or interactions between genes and/or gene products. In a metabolic pathway, nodes represent biochemical molecules, and edges represent reactions that take place between such biomolecules. The reactions are carried out by enzymes, which are coded by genes, so in a metabolic pathway, genes are associated with the edges rather than the nodes.

Once such sophisticated pathway models have become available, the challenge was to identify those pathways that are important in a given phenotype. The first analysis approaches for pathways were to simply consider the pathways as simple sets of genes and use the methods previously developed for gene set analysis: ORA and FCS. However, ORA and FCS are limited because they do not account for the hierarchical structure of pathways or interactions between genes. Topology-based (TB) approaches were developed to further incorporate knowledge about gene topology and network in their hypothesis testing [176, 177]. Topology-based approaches are able to consider all important elements ignored by ORA and FCS methods: the positions and roles of all the genes in every pathway, the direction and type of signals between them, etc. Because of their advantages, many more topology-based approaches have been proposed since [178, 179, 180, 181, 182, 183].

More than 70 pathway analysis methods have been developed to date. Despite the availability of many pathway analysis methods, there is no single method that is always superior to

others, as reported in a previous benchmarking article [184]. Such a situation is not surprising because the performance of a pathway analysis method is highly dependent on the characteristics of the data and the biological question being asked. For example, the performance of a pathway analysis method can be affected by the number of samples, the number of genes, the number of pathways, the number of differentially expressed genes, the effect size of the differentially expressed genes, the correlation structure of the genes, the noise level, etc. In addition, the performance of a pathway analysis method can be affected by the characteristics of the pathways, such as the number of genes in the pathway, the number of interactions between genes, the topology of the pathway, the number of differentially expressed genes in the pathway, etc. Unfortunately, the characteristics of the data and the biological question are often unknown, and the performance of a newly developed pathway analysis method is often evaluated on a small number of datasets and conditions. This makes it really difficult to determine the general performance of a pathway analysis method.

When performing pathway analysis, researchers are often overwhelmed by the number of methods available and the lack of a clear guideline on which method to use. This is further complicated by the fact that the results of different pathway analysis methods can be inconsistent. This inconsistency can be due to the different statistical models used by the methods, the different assumptions made by the methods, the different data preprocessing steps used by the methods, etc. Because of this inconsistency, researchers often use multiple pathway analysis methods and then compare the results to identify the pathways that are consistently identified by multiple methods. This approach is called consensus pathway analysis. Consensus pathway analysis is a powerful approach because it can identify the pathways that are robustly identified by multiple methods and are, therefore, more likely to be biologically meaningful. However, when performing consensus pathway analysis, researchers are faced with the challenge of combining the results of multiple pathway analysis methods. This is not a trivial task because the results of different pathway analysis methods are often not directly comparable. For example, the results of different pathway analysis methods can be in the form of p-values, z-scores, or ranks, and the scales of the results can be different. In addition, the results of different pathway analysis methods can be correlated.

In this dissertation, we address these challenges by first introducing a web interface that offers pathway analysis using multiple methods and datasets in a single session with rich visualization features. This allows life scientists to easily conduct pathway analysis, compare results from different methods and datasets, and reach better consensus for downstream analyses. We then propose a new pathway analysis method called Perturbation-based Gene Set Analysis (PGSA). PGSA is a gene set analysis method that is based on the idea of perturbing the input gene expression data and then performing gene set analysis on the perturbed data. The main advantage of PGSA is that it is able to account for the uncertainty and noise in the input gene expression data. This is important because gene expression data is often noisy, and the results of gene set analysis can be sensitive to the noise in the data. In an extensive benchmarking study with 421 datasets, we show that PGSA outperforms other methods, topological and non-topological alike, in identifying the impacted pathways by a large margin. This is the first time that such a large-scale benchmarking study has been conducted for pathway analysis methods.

This part of this dissertation is organized as follows. In Chapter 8, we introduce the web interface for consensus pathway analysis. In Chapter 9, we introduce the PGSA method and present the results of the benchmarking study. In Chapter 10, we conclude our work on pathway analysis.

Chapter 8

CPA: A web-based platform for Consensus Pathway Analysis and interactive visualization

This chapter is based on the following publication: ***Hung Nguyen****, Duc Tran, Jonathan M Galazka, Sylvain V Costes, Afshin Beheshti, Juli Petereit, Sorin Draghici, and Tin Nguyen. CPA: a web-based platform for consensus pathway analysis and interactive visualization. Nucleic Acids Research. 2021. DOI: 10.1093/nar/gkab421*

In molecular biology and genetics, there is a large gap between the ease of data collection and our ability to extract knowledge from these data. Contributing to this gap is the fact that living organisms are complex systems whose emerging phenotypes are the results of multiple complex interactions taking place on various pathways. This demands powerful yet user-friendly pathway analysis tools to translate the now abundant high-throughput data into a better understanding of the underlying biological phenomena. Here we introduce Consensus Pathway Analysis (CPA), a web-based platform that allows researchers to i) perform pathway analysis using eight established methods (GSEA, GSA, FGSEA, PADOG, Impact Analysis, ORA/Webgestalt, KS-test, Wilcox-test), ii) perform meta-analysis of multiple datasets, iii) combine methods and datasets to accurately identify the impacted pathways underlying the studied condition, and iv) interactively explore impacted pathways, and browsing relationships between pathways and genes. The platform supports three types of input: i) a list of differentially expressed genes, ii) genes and fold changes, and iii) an expression matrix. It also allows users to import data from NCBI GEO. The CPA platform currently supports the analysis of multiple organisms using KEGG and Gene Ontology, and it is freely available at `http://cpa.tinnguyen-lab.com`.

8.1 Introduction

Advanced high-throughput and sequencing technologies have transformed biological research by allowing scientists to monitor changes in living organisms and biological systems. Regardless of the assay technology used, a comparative analysis experiment often yields a set of differentially expressed (DE) genes or gene products. Though important, these lists of DE genes fail to reveal the mechanisms underlying the studied condition. To translate the differential expression to biological knowledge, researchers have been developing various knowledge bases that map genes and their products to functional modules and biological processes. These include KEGG [185], Reactome [186], Wikipathways [187], and Gene Ontology (GO) [168]. At the same time, pathway analysis methods have been developed to identify pathways that are impacted under certain conditions.

More than 70 pathway methods have been developed thus far [184, 188]. These methods can be categorized into three classes. The earliest approaches use Over-Representation Analysis (ORA) [171, 189, 190, 169, 191, 192] that identify the pathways in which the DE genes are over- or under-represented. The drawbacks of ORA include: i) they only consider the number of DE genes and completely ignore their expression changes, and ii) they assume the genes are independent, which they are not. Functional Class Scoring (FCS) approaches [174, 193, 194, 195, 196] have been developed to address some of the issues raised by ORA approaches. The main improvement of FCS is based on the observation that small but coordinated changes in expression of functionally related genes can have a significant impact on pathways. However, both ORA and FCS still ignore the direction and type of the signals between genes, the positions and roles of the genes on each pathway, as well as all the other information captured by the topology of the pathway. Topology-based (TB) approaches [197, 176, 177, 198, 178, 199, 179, 180] which fully exploit all the knowledge about how genes interact as described by pathways, have been developed more recently. Recent reviews included 22 TB-based methods [181, 188].

In spite of the availability of powerful pathway methods, understanding the phenomena that determine the measured changes is as challenging as ever, if not more so. First, the sheer

number of methods makes it challenging for life scientists to choose the correct method for their data and purpose. In a recent publication [184], we have shown that all existing methods often provide biased results. No single method is consistently superior to others. Second, many of these methods are software packages that require users to go through the burden of installation and updating (some are not even executable anymore due to outdated dependencies). This hinders the reproducibility and universal accessibility of analysis results. Finally, most tools do not offer interactive data visualizations that are important for users to deeply explore pathway connectivities and gene networks.

Recognizing these challenges, many web-based tools have been developed to assist researchers in their analysis. Tools such as EnrichNet [200], GENAVi [201], WebGestalt [202], WebGIVI [203], DAVID [204], INMEX [205], g:Profiler [206], and Enrichr [207] provide GUI interfaces for users to input gene lists and perform enrichment analysis. Other tools such as KaPPA-View [208], 3Omics [209], PaintOmics [210], IMPaLA [211], and GeneTrail2 [212] visualize enrichment results of multi-omics data. These tools, however, have a number of limitations: (1) they cannot combine, compare, and contrast results of different methods; (2) they lack integrative capability across multiple datasets; and (3) they are unable to comprehensively visualize pathway connectivity, gene networks, and expression change altogether.

Here, we introduce Consensus Pathway Analysis (CPA), a comprehensive web-based resource that allows users to compare and contrast analysis results across different methods and experiments. Specifically, CPA allows researchers to: i) perform pathway analysis using eight popular methods, GSEA [166], GSA [193], FGSEA [146, 213], PADOG [195], Impact Analysis [176], ORA/WebGestalt [214, 202], KS-test [215], and Wilcox-test [216], ii) perform meta-analysis of multiple datasets, iii) combine methods and datasets to find consensus results, and iv) interactively explore significantly impacted pathways across multiple analyses, and browsing relationships between pathways and genes. CPA currently supports the analysis of more than 1,000 organisms using KEGG and Gene Ontology databases.

8.2 Material and Methods

The CPA website is a cloud-computing service for pathway analysis. It provides functions to manage users' data, support multiple analysis sessions, and visualize results. All computations are performed on the CPA server hosted by UNR. Inputs, parameter settings, and analysis results are saved onto the user account and can be easily loaded and updated. Users can also switch between analysis sessions, as well as browse and export results at any time.

Figure 8.1A shows the overall workflow of an analysis session using CPA, while Figure 8.1B shows sample visualizations and analysis results. Overall, the analysis pipeline consists of three main modules: data input, parameter setting, and analysis and visualization. For input data, users can choose to input a gene list, a gene list and their fold changes, or a gene expression matrix from their local machine. The interface is designed so that users can flexibly analyze their own data. We also support a direct import from NCBI Gene Expression Omnibus (GEO) [217]. This is especially helpful if users are interested in taking advantage of existing data on NCBI GEO. In the parameter setting, users can choose the pathways of interest (GO/KEGG), analysis methods, and method parameters. Finally, in the analysis and visualization module, users can visualize and interactively explore and export analysis results. Figure 8.1B shows example visualizations and publication-ready figures generated by the platform. These include: sample landscape (using t-SNE), volcano plot, gene heatmap, pathway-pathway connectivity, and gene networks. We will describe in details each of the three modules in the following sections.

8.2.1 Input and data management

The CPA platform supports three different types of input including i) a list of differentially expressed (DE) genes, ii) genes and their fold changes, and iii) an expression matrix. The first two input types can be directly entered on the website or uploaded from the user's local machine as a .txt or .tsv file, in which each row represents a gene. For expression matrix input, a dataset can be represented by two .csv files (command-separated) – one for expression matrix and one for sample grouping. The sample grouping file has two columns; the first column includes samples, and the second column contains their corresponding groups (e.g., control or

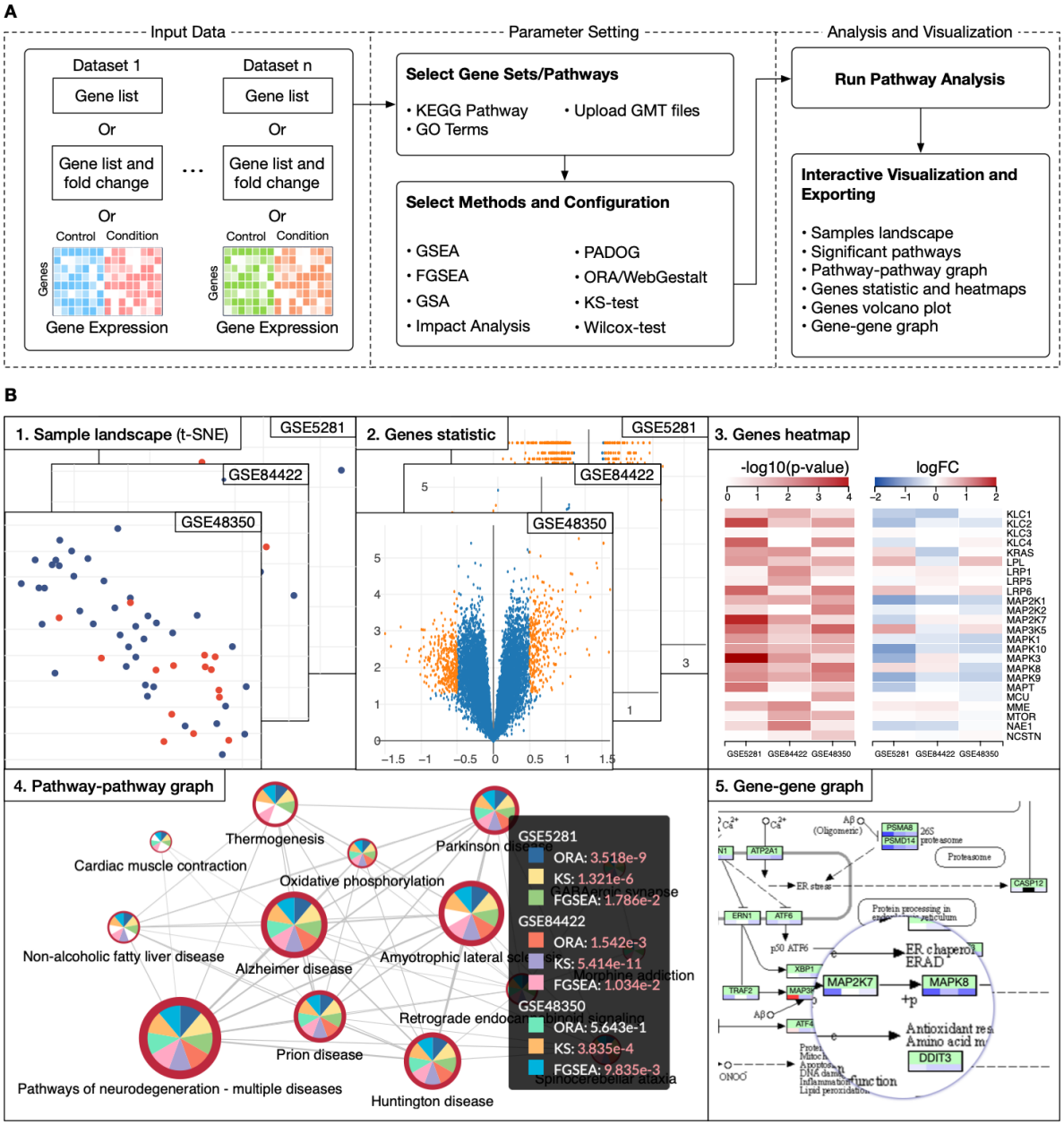

Figure 8.1: The overall workflow and data visualization using Consensus Pathway Analysis (CPA). A) The analysis pipeline consists of three modules: i) input data, ii) parameter setting, and iii) analysis and visualization. The input in one dataset can be a gene list, a gene list and their fold change or an expression matrix. In one analysis session, CPA allows users to analyze multiple datasets using multiple pathway analysis methods. B) Result visualization. Once the analysis is done, users can interactively explore and export the results. For example, they can export the samples landscape (B1), volcano plot (B2), and heatmaps showing p-values and log FC across all datasets (B3). At the pathway level, users can interactively visualize the pathway-pathway connectivity graph (B4) and KEGG pathways (B5). Users can see detailed analysis results and statistics by clicking on each node of graphs (B4). In this case, the analysis included 3 datasets and 3 methods. Analysis results, plots, and graphs can be exported as comma-separated values (.csv file) or publication-ready figures (.png, .svg, etc.).

disease). The sample grouping file is optional. If not provided, users need to manually select control and disease samples in the GUI interface. The platform supports ID conversion from other gene identifiers to Entrez IDs. The conversion is based on the ID mapping provided by the UniProt database with more than 90 ID types, and more than 200 annotation packages currently available from Bioconductor (`https://bioconductor.org/packages/3.12/data/annotation/`).

CPA provides an easy-to-use file manager for users to upload and manage files (upload, remove, rename, and download) for analysis using expression data. Users can upload expression data files from their local machine or import them from NCBI GEO. Data importation from GEO is based on the Bioconductor R package GEOquery [218]. A dataset can only be imported from GEO if the series matrix (pre-processed gene expression file) is available. Files uploaded and imported by anonymous users will be deleted after 24 hours. Users are encouraged to log onto CPA using a Google account so that they can permanently save data and get access to their analysis sessions across multiple devices.

8.2.2 Parameter setting for pathway analysis

Figure 8.2 shows the GUI interface for pathway analysis, in which users can select one or multiple datasets for an analysis session. For each dataset, users can choose the input type from the drop-down list. When users choose to provide a list of DE genes (gene list), ORA/Webgestalt is available for analysis. When genes and fold changes are chosen, Wilcox-test, KS-test, and FGSEA are available for analysis. When users provide an expression matrix, all of the eight pathway analysis methods are available for analysis: GSEA, GSA, FGSEA, PADOG, Impact Analysis, ORA/WebGestalt, KS-test, and Wilcox-test. Each of them is designed to find different patterns of the data. The purpose of consensus analysis is that users can explore the results of multiple analyses, including results of different datasets as well as of different methods. However, we would also like to note that a particular pathway is identified by multiple methods does not necessarily make it more biologically meaningful.

Currently, CPA supports the analysis of more than 1,000 organisms that have KEGG pathways [161] and GO terms [219, 168]. Users can also upload pathway annotations of other

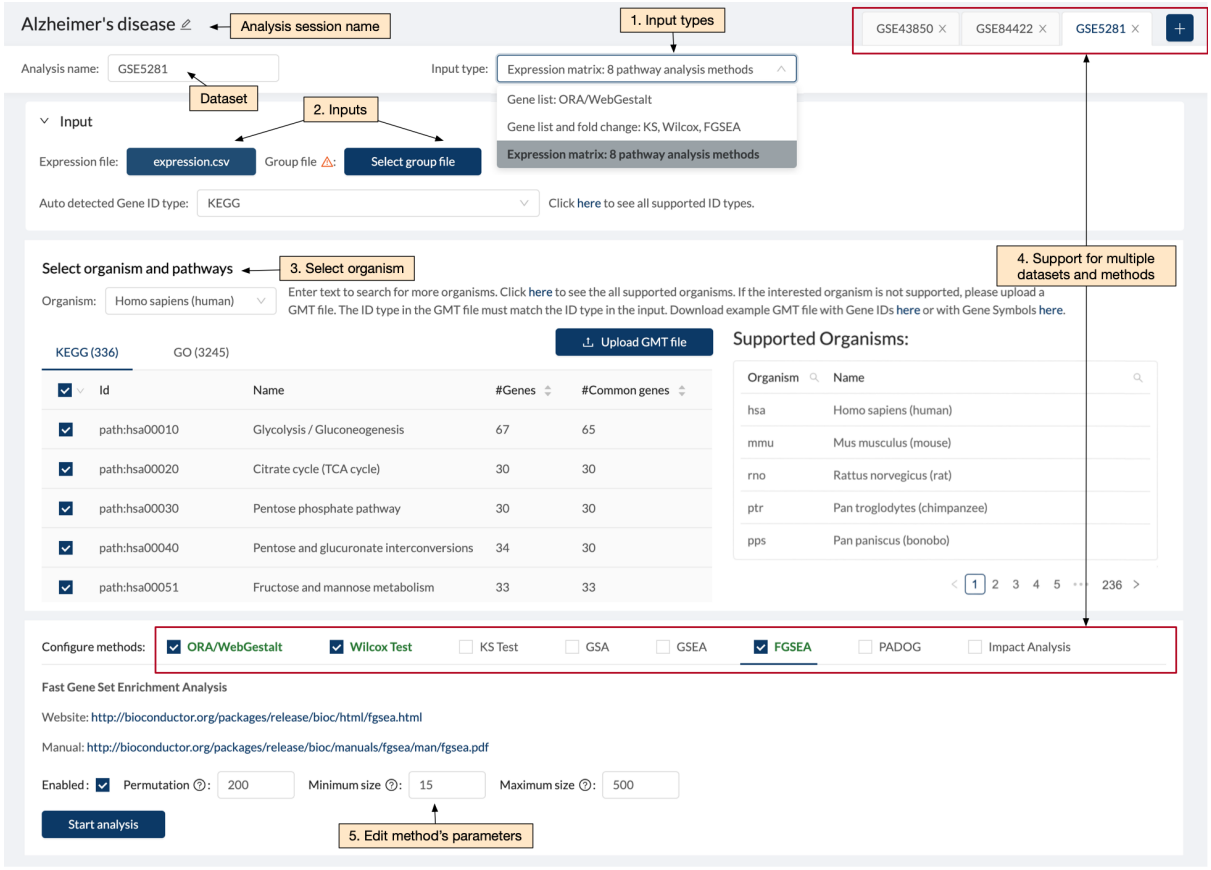

Figure 8.2: Main components of the pathway analysis page. Users are able to: (1) select the input type, (2) select the corresponding input with the input type, and (3) choose the organism and pathways to be analyzed. The website supports meta-analysis of multiple datasets and multiple methods (4). The website also allows users to change the parameters of individual methods if desired (5).

databases in the GMT file format. After choosing data, pathways, and methods, users can start the analysis by simply clicking the "Start analysis" button. Note that classical methods such as ORA, KS, or Wilcox test usually take a second to finish the analysis. However, methods such as PADOG or GSEA that involve permutation and bootstrapping usually take several minutes to finish an analysis, especially when analyzing multiple datasets. Analysis sessions are queued and updated in real time. Results and configurations are saved onto user accounts so that they can switch to any analysis session at any time.

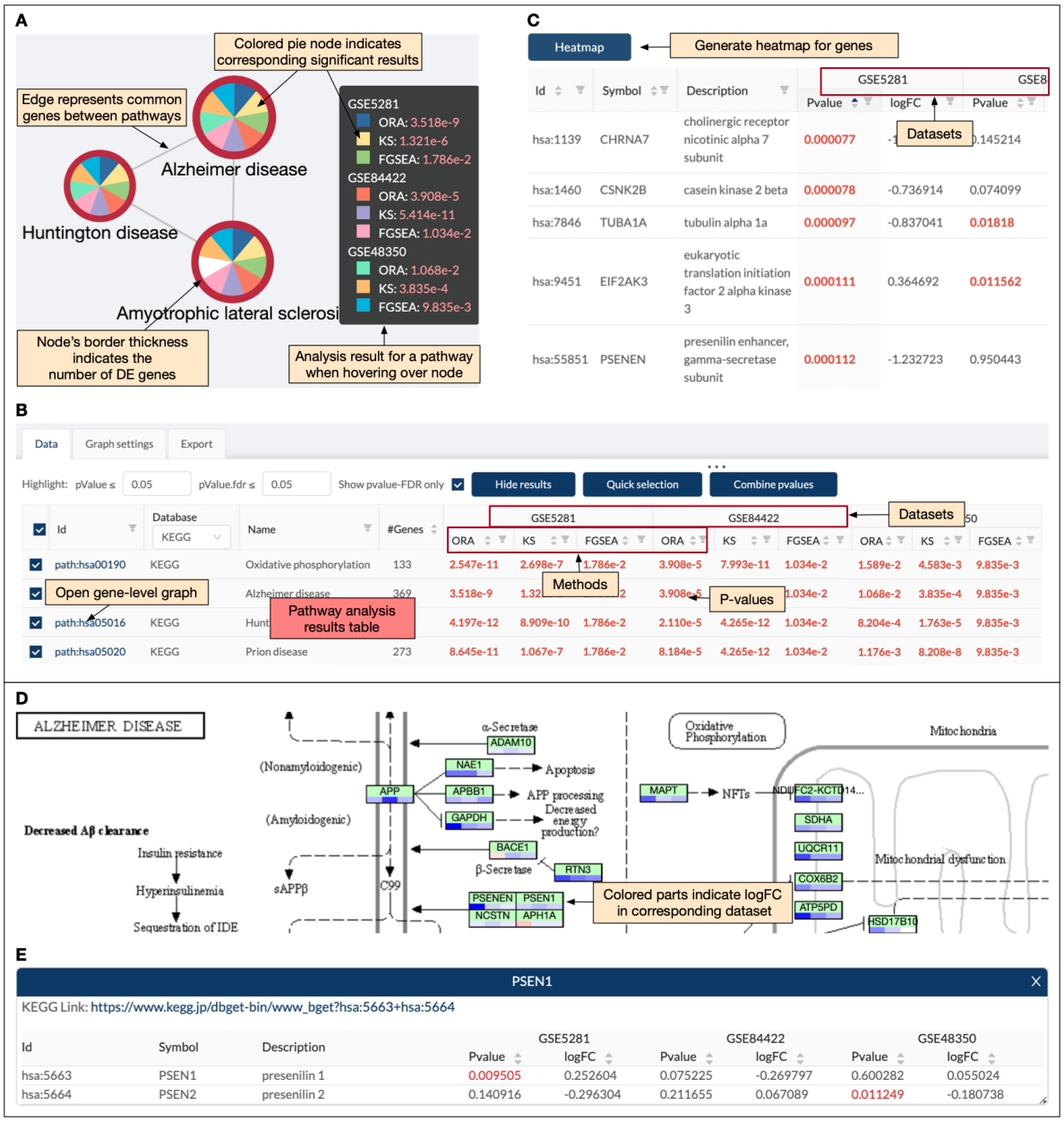

Figure 8.3: Pathway analysis and visualization using the CPA platform. **A**) Pathway-pathway connectivity graph where nodes represent pathways and edges represent that there are common genes between pathways. In this example, we analyze three datasets using three methods, making a total of nine analyses. Each node is a pie chart in which individual slices correspond to different analyses. A slice is colored if the corresponding p-value is significant. Node border thickness indicates the number of significantly differentially expressed (DE) genes in the pathway. **B**) The pathway panel shows the significant pathways and the adjusted p-values obtained in each dataset using each analysis method. For example, the *Alzheimer's disease* pathway is shown on top with significant p-values in all of the 9 analyses (p-values are colored in red when they are significant). This pathway panel is automatically populated, together with the pathway connectivity graph, after the analysis is performed. **C**) The gene panel appears when users left-click a node in the pathway connectivity graph (in panel A). This panel shows the genes of the pathways and their statistics (p-values and log FC) across all datasets. **D**) Gene network (KEGG pathway) and expression change. This panel appears when users right-click a node in the pathway connectivity graph (in panel A). Nodes in a KEGG pathway graph are divided equally into multiple colored parts representing expression change (up- or down-regulated). **E**) Gene panel that appears when users right-click on a node of the gene network (in panel D).

8.2.3 Analysis and visualization

Once the analysis is completed, the website displays the pathway-pathway connectivity graph (Figure 8.3A) in which nodes represent pathways and edges indicate that the connected pathways share a certain number of genes (defined by users). In this pathway graph, the size of a node is proportional to the number of genes in the pathways, while the border thickness is proportional to the total number of DE genes. As shown in the figure, each node is divided into multiple slices that represent the results of multiple analyses. For example, an analysis session with three datasets and three methods has a total of 9 slices (9 analyses). Users can change the number of nodes displayed by changing the significance threshold (p-value) and the number of analyses in which the p-values are significant. By default, the significance threshold is set to 5% (after adjustment using FDR), and a node appears only if the pathway is significant in at least one analysis. A slice is colored if the pathway has a significant p-value in the corresponding analysis. When users hover the mouse over a node, a small window will appear and show the p-values of the pathway in all analyses. In Figure 8.3A, the black window shows the p-values of the *Alzheimer's disease* pathway. All nine p-values of this pathway are significant ($FDR < 5\%$), and thus, all slices are colored. In contrast, the *Amyotrophic lateral sclerosis* pathway has a white slice because one analysis has a non-significant p-value. The graph is highly configurable inasmuch users can easily change the scale and color of all elements to export high-quality figures. Users can also choose to display pathways of only GO, or KEGG, or both.

A pathway table that accompanies the pathway graph shows the essential information of each pathway: ID, description, number of genes, and the p-values obtained in all analyses (Figure 8.3B). Using the editable fields and pop-up menus of this table, users can change the significance threshold, filter out pathways, or hide the results of any method or dataset. They can also interactively modify the graph by hiding unwanted pathways or adding pathways of interest. The table can also be used to select pathways with more than a certain number of significant results, or select pathways that are significant in some analyses but not in others. Users can also conduct meta-analysis by combining p-values of a pathway across multiple datasets using Fisher's [220], Stouffer's [221], addCLT [222], or minP method [223]. Note

that combining p-values obtained from different methods for the same dataset might lead to artificially low meta p-values. Therefore, it is more recommendable to combine the p-values obtained from independent datasets. When combining p-values using Fisher's or Stouffer's method, any individual p-value of zero will result in a combined p-value of zero. Therefore, by default, the platform will round the individual p-values up to 1e-10 before combining. The meta-analysis results will be added to the pathway table as a column and can also be used to manipulate the pathway graph. The meta-analysis results will be added to the pathway table as a column and can also be used to manipulate the pathway graph.

Besides the pathway table, the platform also creates a gene table (Figure 8.3C) that appears when users select one or more nodes of the pathway graph. The table shows the genes of the selected pathways, their description, and statistics obtained from all datasets. The table can be modified to show either the intersection or union of all pathways selected. Users can sort the genes, remove unwanted genes, or remove a dataset. The genes and their statistics can be exported. Users can also generate the heatmaps displaying log FC and p-values of the genes by just clicking the "Heatmap" button.

The platform also supports pathway visualization. When users right-click on a node of the pathway graph, they can choose to display the KEGG pathway (Figure 8.3D). In this presentation, each node is a compound. The bar under each node in the pathway is divided into smaller

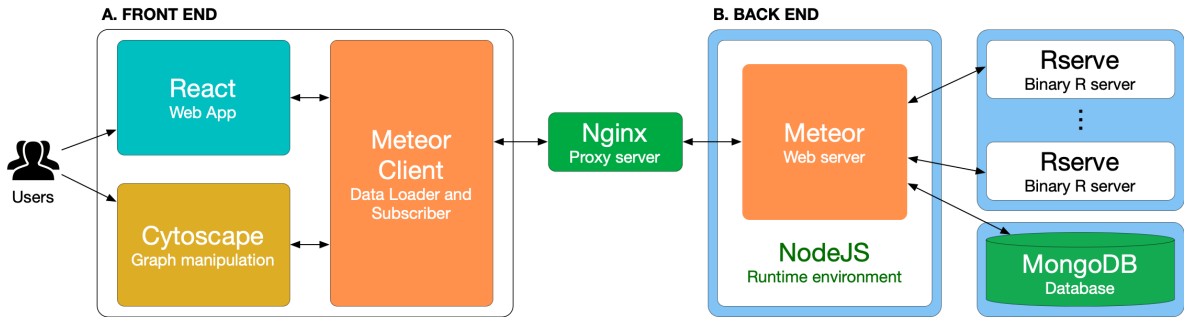

Figure 8.4: The architecture of the CPA platform. **A**) The front end consists of a graphic user interface (using React), a graph manipulation module (using Cytoscape), and a data handling module (Meteor client). **B**) The back end consists of a web server (Meteor web server), runtime environment (NodeJS), R servers (Rserve), and a database (MongoDB). Each backend module is containerized using Docker (blue boxes). The R servers are used to perform pathway analysis while the database is used to store user data and analysis results. User clients (from front end) communicate with backend servers through the Distributed Data Protocol (Meteor client) and a Nginx web proxy server.

parts that correspond to datasets included in the analysis session. Each part is colored based on its impact direction (i.e., up- or down-regulated). Users can easily find genes that are consistently up- or down-regulated in all datasets and relationships among them. Since each node in a KEGG pathway often includes multiple genes, the color of each part reflects the average FC of all genes in the node. By default, we also combine the p-values of all genes of the node to obtain a combined p-value. Users can choose to color the node based on this combined p-value. Users can also remove any unwanted datasets from the visualization. When users click on a KEGG node, they can see the genes belonging to the node. For example, when clicking on the PSEN1 node on the KEGG pathway, the gene table appears as shown in Figure 8.3E. This table displays the genes, their description, p-value, and log FC in all datasets.

While exploring the analysis results, users can export any graph as raster (.png) or vector (.svg) images. They can also export results obtained from differential analyses, gene information, and p-values from pathway analysis .csv files. Other plots in the pathway analysis page (e.g., sample landscape, volcano plot, etc.) can be exported as raster images (.png).

8.3 Implementation

Figure 8.4 shows the architecture and technologies used to build the CPA platform. We used MeteorJS (`https://www.meteor.com/`) – a full-stack JavaScript platform for modern web development – as the core web platform to create the web server and to communicate with user clients.

For the front end, we build the graphic user interface using React, which is a JavaScript library (`https://reactjs.org/`). The website is designed to be user-friendly and has three main pages: pathway analysis, pathway visualization, and data management. On the pathway analysis page, users can upload and choose datasets, select methods, and observe gene-level statistics. Gene-level plots are generated using the Plotly JavaScript graphing library (`https://plotly.com/javascript/`). In the visualization page, we implement the interactive network visualization using CytoscapeJS, which is a graph theory library (`https://js.cytoscape.org/`). Gene heatmaps are plotted using D3js (`https://d3js.org/`). In the data management page, we build the file manager using OpusCapita React File Manager

(https://www.npmjs.com/package/@opuscapita/react-filemanager). Data exchange between user clients and backend servers is accomplished using the Distributed Data Protocol (Meteor client) and a Nginx web proxy server (https://www.nginx.com).

For the back end, we build the web server using Meteor and NodeJS (https://nodejs.org). NodeJS is a JavaScript runtime environment built on Chrome's V8 JavaScript engine that allows JavaScript code to run outside the browser environments. Input files for analysis are stored on the server's storage for fast access. Other data, including user information, analysis sessions, analysis configuration, and results, are saved in a MongoDB database (https://www.mongodb.com). Once the requests for performing pathway analysis are received from clients and saved by the web server, they are passed to R servers created by Rserve (https://www.rforge.net/Rserve/) to perform pathway analyses. Multiple Reserve instances can be created to perform multiple analyses concurrently. All software and packages in the back end run in containerized environments using Docker (https://www.docker.com/).

8.4 Data source

CPA supports the analysis of more than 1,000 organisms using KEGG [161] and GO terms [168]. At the time of writing this article, the version of KEGG is 97.0 (released Jan. 1, 2021) and of GO terms is 1.16 (released Feb. 16, 2021). The automatic ID conversion in the CPA platform is based on the ID mapping from the UniProt database (current version: 2021_02) and more than 200 annotation packages from Bioconductor (version 3.12, released Oct. 28, 2020). ID mappings and databases will be updated twice a year (January and July).

8.5 Results

To show how the CPA platform can be used for pathway analysis, we have created an example analysis session and included it in our tutorial page. In this example session, we analyze three Alzheimer's datasets: GSE5281 [224], GSE84422 [225], and GSE48350 [226]. The three datasets consist of a total of 66 control and 57 disease samples (Table 8.1). We chose the Alzheimer's datasets because there is a target pathway in KEGG, *Alzheimer's disease*, that

describes the known mechanisms and biological processes involved in this disease. It is also well known that the pathways *Parkinson's disease*, *Huntington's disease*, and *Pathways of neurodegeneration - multiple diseases* share many genes and mechanisms with *Alzheimer's disease* [227, 228, 229, 230]. Therefore, we expect to identify all these neurological disorder pathways as statistically significant.

Table 8.1: Alzheimer's datasets used in our data analysis. The first two columns show the accession ID and tissue, while the last three columns show the number of controls, number of diseases, and assaying platforms, respectively.

Dataset	Tissue	C	D	Platform
GSE5281	Entorhinal Cortex	13	10	HG-U133+ 2.0
GSE84422	Sup. Tem. Gyrus	14	22	HG-U133A
GSE48350	Entorhinal cortex	39	15	HG-U133A

In this analysis, we include a total of 335 KEGG pathways and 2,508 GO terms. In the global pathway-pathway connectivity graph, we have a total of 2,843 nodes – one node per KEGG pathway or GO term. Each dataset is analyzed using three methods, ORA, KS-test, and FGSEA, using default parameters. For each analysis, we adjust the p-values using Benjamini-Hochberg's False Discovery Rate (FDR) [231]. The significance threshold is set to $FDR < 5\%$. Figure 8.5 shows the subnetwork obtained with the significant nodes. Nodes in the module are selected so that each pathway is significantly impacted in at least 5 analyses (out of 9 analyses).

The five pathways related to neurodegenerative diseases, *Pathways of neurodegeneration - multiple diseases*, *Alzheimer's disease*, *Huntington's disease*, *Parkinson's disease*, and *Prion disease*, are consistently identified as significant in all of the 9 analyses. The *Amyotrophic lateral sclerosis* pathway is significant in 8 out of 9.

Table 8.2 shows the FDR-corrected p-values of the 14 pathways. The first column shows the pathway name, while the next 9 columns show the p-values obtained from the 9 analyses. As the web interface also allows us to combine the p-values obtained for a pathway across multiple datasets, we use the addCLT method [222] to combine the p-values for each method. The meta-analysis results are presented in the three last columns in Table 8.2. The meta-analysis, as well as the results obtained from individual analyses, clearly shows that pathways related to neurodegenerative diseases are significantly impacted regardless of datasets and methods.

Table 8.2: FDR-corrected p-values of 14 pathways that are significantly impacted in three Alzheimer's datasets (GSE5281, GSE84422, and GSE48350). Each dataset is analyzed by three methods (CPA, ORA, and FGSEA), resulted in 9 analyses (columns 3–11). The last three columns show the meta-analysis results using the addCLT method. The results indicate that these pathways are consistently identified as significant across all analyses.

#	Pathway name	GSE5281			GSE84422			GSE48350			Meta-analysis		
		ORA	KS	FGSEA	ORA	KS	FGSEA	ORA	KS	FGSEA	ORA	KS	FGSEA
1	Alzheimer disease	4e-09	1e-06	2e-02	4e-05	5e-11	1e-02	1e-02	4e-04	1e-02	4e-07	9e-13	2e-06
2	Huntington disease	4e-12	9e-10	2e-02	2e-05	4e-12	1e-02	8e-04	2e-05	1e-02	6e-11	3e-17	2e-06
3	Parkinson disease	0	4e-13	2e-02	4e-07	0	1e-02	2e-03	1e-04	1e-02	6e-10	2e-14	2e-06
4	Prion disease	9e-11	1e-07	2e-02	8e-05	4e-12	1e-02	1e-03	8e-08	1e-02	2e-10	8e-24	2e-06
5	Pathways of neurodegeneration	1e-11	2e-08	2e-02	4e-06	4e-10	1e-02	2e-03	7e-06	1e-02	3e-10	1e-18	2e-06
6	Oxidative phosphorylation	3e-11	3e-07	2e-02	4e-05	8e-11	1e-02	2e-02	5e-03	1e-02	1e-06	1e-08	2e-06
7	Cardiac muscle contraction	2e-02	1e-02	2e-02	6e-01	3e-01	1e-02	4e-02	9e-02	1e-02	3e-02	1e-02	2e-06
8	Thermogenesis	4e-05	1e-02	2e-02	3e-01	5e-02	1e-02	3e-01	9e-03	1e-02	4e-02	6e-06	2e-06
9	Retrograde endocannabinoid s.	8e-06	1e-03	2e-02	4e-01	7e-04	1e-02	3e-04	3e-07	1e-02	4e-03	3e-11	2e-06
10	Amyotrophic lateral sclerosis	2e-09	2e-06	2e-02	2e-07	1e-10	1e-02	1e-01	2e-03	1e-02	3e-03	6e-10	2e-06
11	GABAergic synapse	8e-02	4e-02	3e-02	9e-02	4e-02	1e-02	4e-03	3e-03	1e-02	7e-05	7e-06	4e-06
12	Spinocerebellar ataxia	4e-03	1e-02	2e-02	3e-03	1e-05	1e-02	8e-02	3e-03	1e-02	3e-04	5e-08	2e-06
13	Non-alcoholic fatty liver d.	7e-06	6e-03	2e-02	3e-01	1e-04	1e-02	2e-01	1e-02	5e-02	2e-02	3e-07	8e-05
14	Morphine addiction	4e-01	7e-01	3e-02	1E+00	6e-01	4e-02	5e-03	9e-04	1e-02	3e-01	8e-01	3e-05

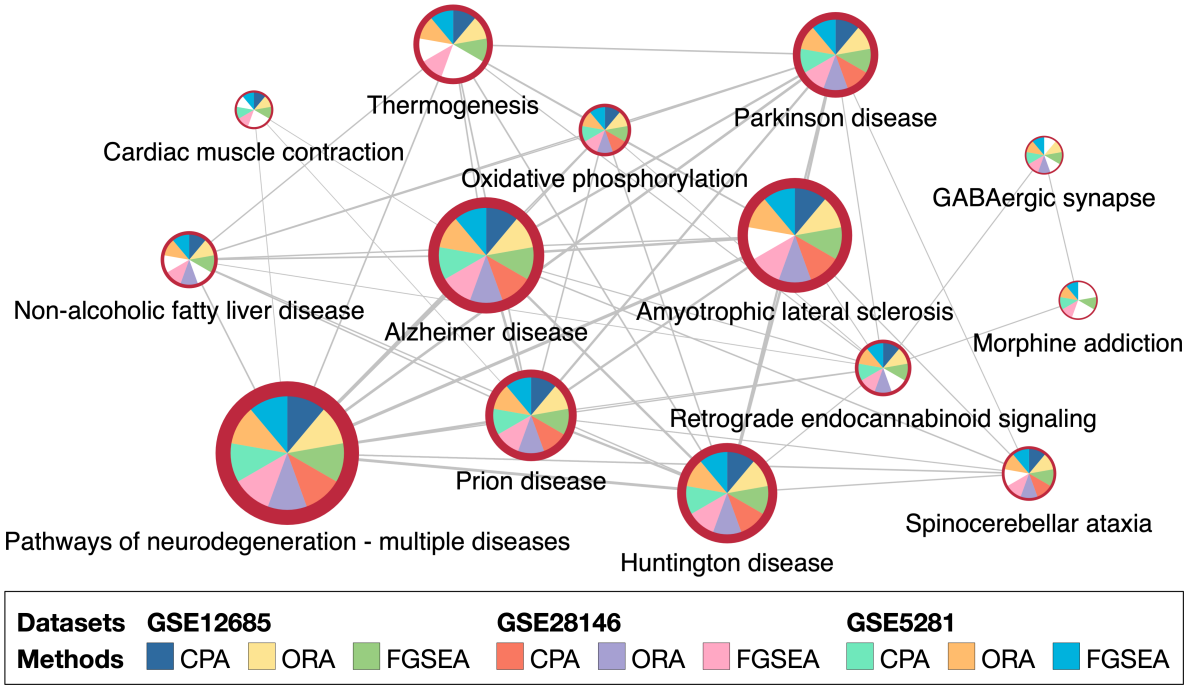

Figure 8.5: The connected module of pathways that are significantly impacted in Alzheimer's datasets GSE5281, GSE84422, and GSE48350. Each dataset is analyzed using three pathway methods, CPA, ORA, and FGSEA. Only pathways that are significantly impacted in at least 5 analyses (out of 9) are shown.

Using the website, we also perform a gene-level analysis to identify genes that can potentially play an important role in the dysregulation of the five neurodegenerative pathways. For that purpose, we intersect the genes that: (1) belong to all of the five pathways, and (2) are differentially expressed in all three datasets ($FDR < 5\%$). Figure 8.6A shows the heatmaps of the resulting 21 DE genes. Most of these genes belong to the components related to mitochondria, proteasome, and microtubule in all five pathways. Figure 8.6B shows the direct mapping of these genes to those components of the *Alzheimer's disease* pathway.

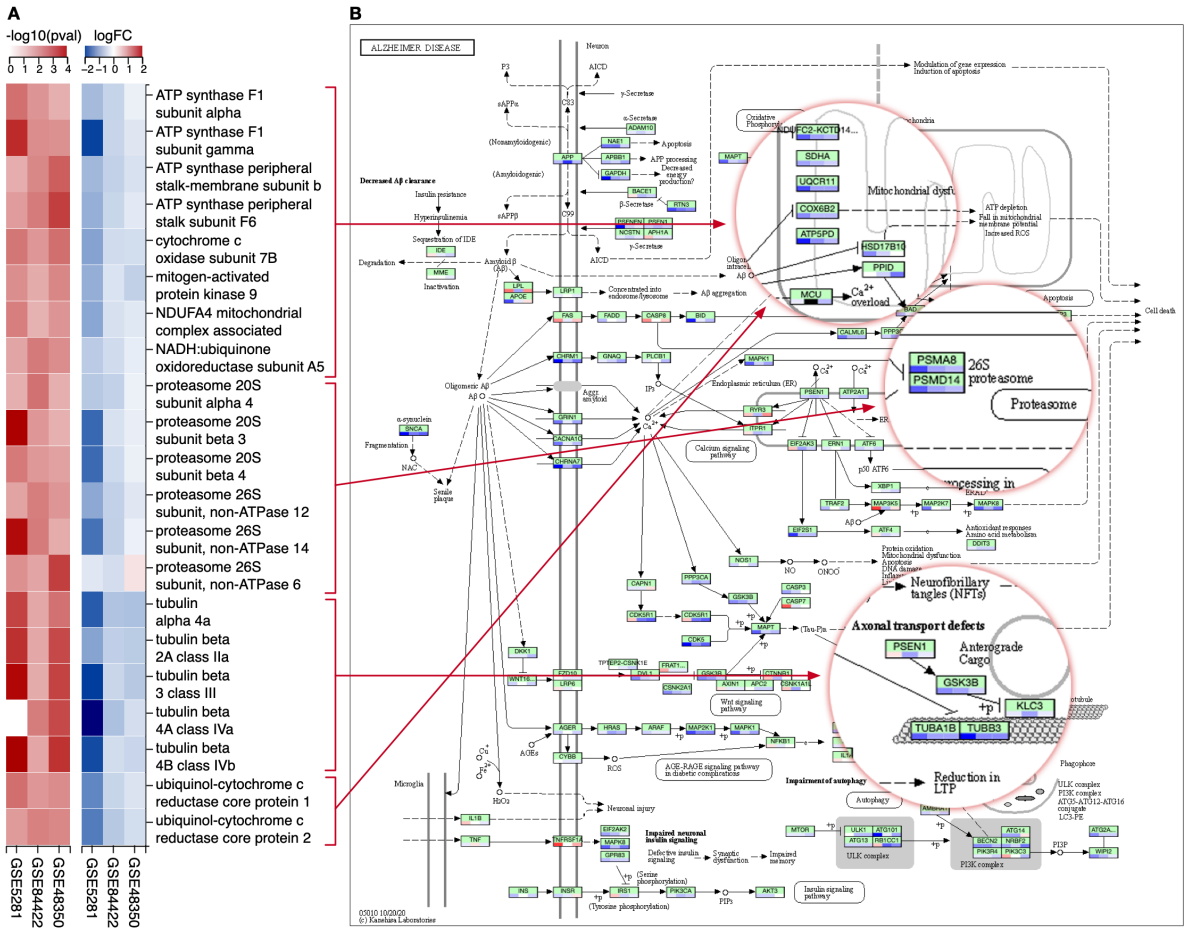

Figure 8.6: Differential analysis of genes that belong to five neurodegenerative pathways: *Pathways of neurodegeneration - multiple diseases*, *Alzheimer's disease*, *Huntington's disease*, *Parkinson's disease*, and *Prion disease*. **A.** Heatmaps of p-values and log FC of genes that are differentially expressed (DE) in all of the three Alzheimer's datasets (GSE5281, GSE84422, and GSE48350). **B.** Mapping of DE genes to mitochondria, proteasome, and microtubule components of the *Alzheimer's disease* pathway.

Chapter 9

PGSA: A consensus Perturbation-based Pathway Analysis

In this chapter, we introduce the Perturbation-based Gene Set Analysis (PGSA) method and present the results of the benchmarking study. PGSA is a non-topological pathway analysis method that are robust against noise and can efficiently identify the impacted pathways from complex diseases.

9.1 PGSA pipeline

The complete pipeline of PGSA is shown in Figure 9.1. The input is a matrix of gene expression values where each row represents a gene and each column represents a sample. The samples are labeled with a binary variable indicating the sample grouping (e.g., case or control). The input is assumed to be normalized (e.g., TPM normalized) and log-transformed. The input also includes a list of gene sets to be tested for enrichment. The pipeline consists of three main modules: (1) Perturbation, (2) Enrichment, and (3) Consensus. The following sections describe each module in detail.

9.1.1 Perturbation module

The perturbation module first adds noise to the input gene expression matrix. The noise is added to all expression values at once by sampling from a normal distribution with mean 0 and standard deviation $\tilde{\sigma}$, where $\tilde{\sigma} = \text{median}(\sigma_1, \sigma_2, \ldots, \sigma_n)$, and σ_i is the standard deviation of the expression values of gene i.

The module then performs clustering on the perturbed data using k-means clustering with the number of clusters $k = 2$. The initial centroids of the algorithm are the gene-wise mean

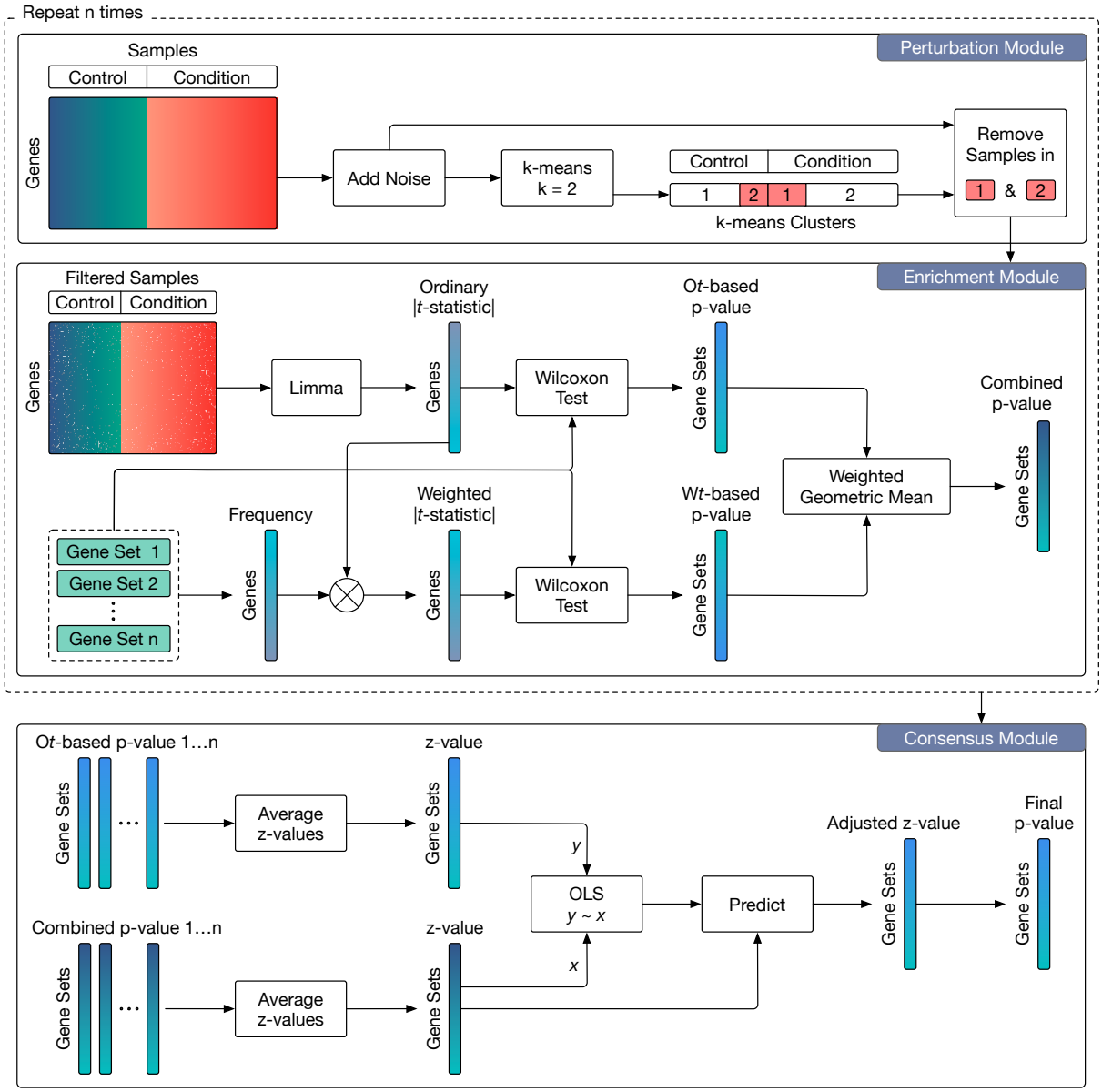

Figure 9.1: The complete pipeline of PGSA. The input is a matrix of gene expression values with labeled samples and a list of gene sets. The pipeline consists of three main modules: (1) Perturbation, (2) Enrichment, and (3) Consensus. The pipeline starts with the perturbation module, which adds Gaussian noise to the input gene expression matrix and performs clustering on the perturbed data using k-means. The module then compares the clustering labels to the input labels to filter out samples that are not consistently clustered with samples of the same label. The perturbed data with the filtered samples are then passed to the enrichment module. The enrichment module first computes the absolute t-statistic for each gene in the perturbed data using the limma package. The module then performs two gene set enrichment analyses for each input gene set using the Wilcoxon rank-sum test. The first analysis uses the original t-statistics to compute the enrichment p-values (Ot-based p-value) for the gene set, while the second analysis uses the weighted t-statistics to compute the enrichment p-values (Wt-based p-value) for the gene set. The gene weights for the weighted t-statistics are computed using the frequency of genes in the input gene sets. The module then combines the enrichment p-values from the two analyses using weighted geometric mean to compute the Combined p-value for the gene sets. The outputs of the enrichment module are two p-values for each gene set: the Ot-based p-value and the Combined p-value. PGSA repeats the perturbation and enrichment modules and passes the outputs of the enrichment module from all iterations to the consensus module. For each gene set, the consensus module computes the average z-statistic of the Ot-based p-values and the average of the Combined p-values across all iterations, respectively. The module then uses a linear regression model to adjust the average z-statistic of the combined p-values to the average z-statistic of the Ot-based p-values. The adjusted average z-statistic of the Combined p-values is then used to compute the final p-value for the gene set.

expression values of the two groups. The module then computes for each sample an agreement score using the connectivity of the sample to the two clusters and the input labels:

$$\text{score}_i = 1 - \frac{1}{n}\sum_{j=i}^{n}|l_{ij} - c_{ij}| \tag{9.1}$$

where

- n is the number of samples,
- $l_{ij} = 1$ if sample i and sample j have the same label in the input labels, and $l_{ij} = 0$ otherwise,
- $c_{ij} = 1$ if sample i and sample j are in the same cluster, and $c_{ij} = 0$ otherwise.

This score has a value between 0 and 1, where 1 indicates that the sample i and all samples that are in the same cluster with sample i have the same label in the input labels, and 0 indicates that all samples in the same cluster with sample i do not have the same label as sample i in the input labels. The module then filters out samples with agreement scores less than 0.5, i.e., more than half of the samples in the same cluster with sample i do not have the same label as sample i in the input labels. If there are no samples left after filtering, the module will keep all samples. If only one label is left after filtering, the module will keep all samples of the other label along with the samples kept from the first label. The output of this module is the perturbed expression data after filtering samples with low-agreement scores.

9.1.2 Enrichment module

Using the output of the perturbation module, the enrichment module computes the absolute t-statistic for each gene using the limma package [232]. For each gene set, the module performs two gene set enrichment analyses. In the first analysis, the module uses the original absolute t-statistics to compute the enrichment p-value (O*t*-based p-value) for the gene set. To compute the enrichment p-value, it uses the one-sided (greater) Wilcoxon rank-sum test to compare the absolute t-statistics of the genes in the gene set to the absolute t-statistics of the genes not in the gene set. In the second analysis, the module first computes the gene weights for all genes in

the expression data using the frequency of genes in the input gene sets. Essentially, a gene that appears in more input gene sets will have a higher weight. If a gene is not in any input gene set, its weight is set to 1. The module then multiplies the absolute t-statistics of each gene by its weight to compute the weighted t-statistics. It then uses the weighted t-statistics to compute the enrichment p-value (W*t*-based p-value) for the gene set using the one-sided Wilcoxon rank-sum test similar to the first analysis. Finally, the module combines the enrichment p-values from the two analyses using weighted geometric mean to compute the Combined p-value for the gene set. The weights for the geometric mean are set to 1 for the O*t*-based p-value and to 20 for the W*t*-based p-value. The output of this module is two p-values for each gene set: the O*t*-based p-value and the Combined p-value.

The pipeline repeats the perturbation and enrichment modules n times ($n = 20$ by default) and passes the outputs of the enrichment module from all iterations to the consensus module.

9.1.3 Consensus module

For each gene set, the consensus module converts the O*t*-based p-values and the Combined p-values from all iterations to one-tail z-statistics. To prevent the z-statistics from being too large or too small, the module clips the p-values at 10^{-16} and $1 - 10^{-16}$ before converting them to z-statistics. The module then computes the average z-statistic from the O*t*-based z-values and the average of the Combined z-values across all iterations, respectively. Next, the module fits an ordinary least square linear regression model $y = \beta_0 + \beta_1 x$, where y is the average O*t*-based z-statistic from all gene sets, and x is the average Combined z-statistic from all gene sets. The module then uses the fitted model to adjust the average Combined z-statistics. Finally, the module uses the adjusted average z-statistic to compute the final p-value for the gene set.

9.2 Data collection and processing

In this study, we utilized two distinct sources of gene expression data: the Gene Expression Omnibus (GEO) repository and the Genomic Data Commons (GDC) portal. We describe the data collection and processing for each source in the following subsections.

9.2.1 GEO repository

Data collection

The Gene Expression Omnibus (GEO) [233] is a publicly accessible repository that houses curated microarray, next-generation sequencing, and various other forms of high-throughput functional genomics data. To select datasets for our analysis, we conducted a comprehensive search of the GEO repository to identify gene expression datasets that encompass both normal and disease samples. Specifically, we targeted datasets that contain samples from human subjects and are associated with a wide range of diseases, such as various types of cancer, neurological disorders, viral infections, etc. For each disease, we use multiple related keywords to ensure the inclusion of a comprehensive set of datasets. For example, in the case of acute myeloid leukemia, we used keywords such as *AML*, *acute myeloid leukemia*, *acute leukemia*, and *blood cancer*. Along with the disease-specific keywords, we also included specific terms to further refine the search. Specifically, we screened datasets that contain any of the following keywords: *cancer, tumor, disease, and patient*, in conjunction with any of the followings: *control, normal, healthy, adjacent, non-tumor, non-cancer, and non-disease*. We meticulously selected datasets that are generated using microarray and RNA-Seq technologies to ensure the inclusion of both types of gene expression data in our benchmarks. We specifically chose datasets that contain a minimum of three normal and three disease samples to ensure the inclusion of datasets with sufficiently robust sample sizes for subsequent analyses.

For microarray datasets, we specifically targeted datasets that are generated using the Affymetrix platform, as it is one of the most widely used microarray platforms and has a large number of datasets available in the GEO repository. We only considered datasets that are deposited in GEO as raw data in the form of CEL files, which contain probe-level expression values. This is to ensure that we have control over the data processing steps, such as background correction, normalization, and summarization, which are crucial for the subsequent pathway analysis.

For RNA-Seq analyses, we did not restrict the search to any specific sequencing platform. The RNA-Seq datasets deposited in GEO can be in various formats, such as raw sequence

reads, processed gene expression values, or count data. To ensure consistency in the data processing, we utilized the publicly available ARCHS4 repository [234] to download the pre-processed gene expression. At the time of preparing this manuscript, we downloaded version 2.2 of human and mouse raw counts gene expression files, along with TPM (Transcripts Per Million) normalization files in HDF5 format from ARCHS4, and extracted the gene expression data for the interested datasets from the downloaded files.

Note that if a dataset contains samples from multiple diseases or tissues, we divided the dataset into smaller, disease/tissue-specific datasets. For example, with the dataset *GSE146889*, which includes samples from both colorectal cancer and endometrial cancer, we split this dataset into two sub-datasets: *GSE146889-Colorectal* and *GSE146889-Endometrial*. The full list of GEO datasets used in this study is available in Table 9.1.

Table 9.1: List of GEO datasets used in this dissertation for pathway analysis. The table includes the GEO ID, platform, platform name, technology, disease, and the number of samples. The platform name and technology are specific to the gene expression profiling technology used in the dataset. The disease column indicates the disease or condition that the samples in the dataset are associated with. The number of samples column indicates the total number of samples in the dataset.

GEO ID	**Disease**	**#Samples**	**GEO ID**	**Disease**	**#Samples**
GSE101432	Liver Cancer	39	GSE1418	Chronic Myeloid Leukemia	14
GSE101585	Dilated Cardiomyopathy	16	GSE14762	Renal Carcinoma	22
GSE102312	Chronic Myeloid Leukemia	32	GSE14924-CD4	Acute Myeloid Leukemia	20
GSE102485-I	Type 1 Diabetes	6	GSE14924-CD8	Acute Myeloid Leukemia	21
GSE102485-II	Type 2 Diabetes	22	GSE15471	Pancreatic Cancer	78
GSE102746	Inflammatory Bowel Disease	20	GSE16515	Pancreatic Cancer	52
GSE103001	Breast Cancer	44	GSE16759-mRNA	Alzheimer's	8
GSE104310	Liver Cancer	20	GSE17025-EE	Endometrial Carcinoma	90
GSE104836	Colorectal Cancer	20	GSE17025-PS	Endometrial Carcinoma	23
GSE105130	Liver Cancer	52	GSE17156-A	Influenza	18
GSE106119	Colorectal Cancer	6	GSE17156-S	Influenza	16
GSE106338	Gastric Cancer	15	GSE1751	Huntington's	31
GSE106608	Parkinson's	16	GSE18309	Alzheimer's	9
GSE107991-LTBI	Tuberculosis Infection	33	GSE18670-CTC	Pancreatic Cancer	12
GSE107991-TB	Tuberculosis Infection	33	GSE18670-G	Pancreatic Cancer	12
GSE111459	Tuberculosis Infection	39	GSE18670-T	Pancreatic Cancer	12
GSE112057	Inflammatory Bowel Disease	73	GSE18842	Lung Cancer	91
GSE112221	Liver Cancer	6	GSE19188-ADC	Lung Cancer	110
GSE113230	Breast Cancer	6	GSE19188-LCC	Lung Cancer	84
GSE113255-intestinal	Gastric Cancer	34	GSE19188-SCC	Lung Cancer	92
GSE113524	Alzheimer's	39	GSE19587-DMN	Parkinson's	12
GSE113617	Liver Cancer	78	GSE19587-ION	Parkinson's	10
GSE113942	Cervical Cancer	14	GSE19728-I	Brain Cancer	5
GSE114517-Amygdala	Parkinson's	23	GSE19728-II	Brain Cancer	8
GSE114517-MTG	Parkinson's	23	GSE19728-III	Brain Cancer	8
GSE114517-Nigra	Parkinson's	29	GSE19728-IV	Brain Cancer	8
GSE114564	Liver Cancer	78	GSE19804-1A	Lung Cancer	75
GSE114564-Advanced	Liver Cancer	60	GSE19804-1B	Lung Cancer	79
GSE114564-Early	Liver Cancer	33	GSE19804-2A	Lung Cancer	64
GSE114918	Parkinson's	52	GSE19804-2B	Lung Cancer	67
GSE117993-CD	Inflammatory Bowel Disease	147	GSE19804-3A	Lung Cancer	68
GSE117993-UC	Inflammatory Bowel Disease	98	GSE19804-3B	Lung Cancer	64

Table 9.1: Continued from the previous page.

GEO ID	Disease	#Samples	GEO ID	Disease	#Samples
GSE119630	Colorectal Cancer	60	GSE20146	Parkinson's	20
GSE119794-RNA	Pancreatic Cancer	20	GSE20153	Parkinson's	16
GSE119794-miRNA	Pancreatic Cancer	20	GSE20163	Parkinson's	17
GSE122401	Gastric Cancer	159	GSE20164	Parkinson's	11
GSE123141-CD	Inflammatory Bowel Disease	18	GSE20291	Parkinson's	35
GSE123141-UC	Inflammatory Bowel Disease	19	GSE20292	Parkinson's	29
GSE123658	Type 1 Diabetes	82	GSE21340	Type 2 Diabetes	20
GSE123972	Liver Cancer	40	GSE21354	Brain Cancer	18
GSE124939	Systemic lupus erythematosus	14	GSE21610-DCM	Dilated Cardiomyopathy	50
GSE125583	Alzheimer's	289	GSE21610-ICM	Dilated Cardiomyopathy	26
GSE126848	Non-alcoholic fatty liver disease	29	GSE21802	Influenza	40
GSE129473-BA9	Huntington's	81	GSE23878	Colorectal Cancer	59
GSE130279	Type 1 Diabetes	12	GSE24250	Huntington's	14
GSE131512-NoRec	Breast Cancer	100	GSE24739-G0	Chronic Myeloid Leukemia	12
GSE131512-Rec	Breast Cancer	60	GSE24739-G1	Chronic Myeloid Leukemia	12
GSE133039	Liver Cancer	63	GSE26910-Breast	Breast Cancer	12
GSE133101-AMG	Parkinson's	23	GSE26910-Prostate	Prostate Cancer	12
GSE133101-MTG	Parkinson's	21	GSE27131	Influenza	21
GSE133101-SN	Parkinson's	25	GSE28146	Alzheimer's	30
GSE133684	Pancreatic Cancer	267	GSE28735	Pancreatic Cancer	90
GSE135036	Parkinson's	36	GSE29819-DCM-LV	Dilated Cardiomyopathy	13
GSE135170	Inflammatory Bowel Disease	14	GSE29819-DCM-RV	Dilated Cardiomyopathy	13
GSE135223-CD	Inflammatory Bowel Disease	10	GSE30723-AMs	Influenza	12
GSE135223-UC	Inflammatory Bowel Disease	10	GSE30723-ATII	Influenza	12
GSE135251-Early	Non-alcoholic fatty liver disease	148	GSE32676	Pancreatic Cancer	32
GSE135251-Moderate	Non-alcoholic fatty liver disease	77	GSE33075-Diagnosis	Chronic Myeloid Leukemia	18
GSE136569	Pancreatic Cancer	10	GSE33075-Treated	Chronic Myeloid Leukemia	18
GSE136666-PU	Parkinson's	6	GSE34205-influenza	Influenza	50
GSE136666-SN	Parkinson's	10	GSE3467	Thyroid Cancer	18
GSE137327	Colorectal Cancer	18	GSE3585	Dilated Cardiomyopathy	12
GSE137344-CD	Inflammatory Bowel Disease	149	GSE36389	Endometrial Carcinoma	20
GSE137344-UC	Inflammatory Bowel Disease	81	GSE3678	Thyroid Cancer	14
GSE138485	Liver Cancer	64	GSE37517	Huntington's	13
GSE140089	Parkinson's	41	GSE3790-CB	Huntington's	133
GSE140343	Non-small cell lung cancer	100	GSE3790-CB	Huntington's	133
GSE141142	Breast Cancer	14	GSE3790-CN	Huntington's	140
GSE141746	Colorectal Cancer	10	GSE3790-CN	Huntington's	140
GSE142987	Liver Cancer	65	GSE3790-FC	Huntington's	131
GSE144119-Chronic	Chronic Myeloid Leukemia	65	GSE3790-FC	Huntington's	131
GSE144119-Remission	Chronic Myeloid Leukemia	49	GSE38642	Type 2 Diabetes	63
GSE144259	Colorectal Cancer	6	GSE40012-Influenza	Influenza	75
GSE144269	Liver Cancer	140	GSE40281	Influenza	9
GSE145645	Glioma	35	GSE4107	Colorectal Cancer	22
GSE146009-AA	Colorectal Cancer	30	GSE4183-CRC	Colorectal Cancer	23
GSE146009-CA	Colorectal Cancer	35	GSE4183-IDB	Inflammatory Bowel Disease	23
GSE146889-Colorectal	Colorectal Cancer	80	GSE4290-AC	Brain Cancer	49
GSE146889-Endometrial	Endometrial Carcinoma	72	GSE4290-GBM	Brain Cancer	100
GSE147352	Glioma	100	GSE4290-OG	Brain Cancer	73
GSE147352-low	Glioma	33	GSE45516	Huntington's	9
GSE148355	Gastric Cancer	74	GSE4757	Alzheimer's	20
GSE151347	Liver Cancer	18	GSE48350-central	Alzheimer's	68
GSE154272	COVID-19	45	GSE48350-entorhinal	Alzheimer's	54
GSE155454	COVID-19	58	GSE48350-frontal	Alzheimer's	69
GSE157240	Influenza	85	GSE48350-hippocampus	Alzheimer's	62
GSE157256	Renal Carcinoma	9	GSE48352-Hereditary	Renal Carcinoma	7
GSE158420	Non-small cell lung cancer	74	GSE48352-Sporadic	Renal Carcinoma	25

Table 9.1: Continued from the previous page.

GEO ID	Disease	#Samples	GEO ID	Disease	#Samples
GSE159260	Glioma	20	GSE48466	Influenza	12
GSE160501-D	Atopic dermatitis	12	GSE50161-EPN	Brain Cancer	59
GSE160501-E	Atopic dermatitis	12	GSE50161-GBM	Brain Cancer	47
GSE160501-SC	Atopic dermatitis	18	GSE50161-MED	Brain Cancer	35
GSE162515	Thyroid Cancer	57	GSE50161-PA	Brain Cancer	28
GSE164332	COVID-19	16	GSE50627-AD	Lung Cancer	12
GSE164541	Colorectal Cancer	10	GSE50627-SC	Lung Cancer	9
GSE164541-AD	Colorectal Cancer	10	GSE5281-EC	Alzheimer's	23
GSE164541-CRC	Colorectal Cancer	10	GSE5281-HIP	Alzheimer's	23
GSE165394	Gastric Cancer	12	GSE5281-MTG	Alzheimer's	28
GSE165512-CD	Inflammatory Bowel Disease	75	GSE5281-PC	Alzheimer's	22
GSE165512-CDi	Inflammatory Bowel Disease	55	GSE5281-SFG	Alzheimer's	34
GSE165512-UC	Inflammatory Bowel Disease	75	GSE5281-VCX	Alzheimer's	31
GSE165595	Glioma	30	GSE55945-NEG	Prostate Cancer	15
GSE166925-CD	Inflammatory Bowel Disease	76	GSE55945-POS	Prostate Cancer	14
GSE166925-UC	Inflammatory Bowel Disease	55	GSE57475	Parkinson's	142
GSE168496	Parkinson's	16	GSE58545-BRAF	Thyroid Cancer	36
GSE171415	Lung Cancer	62	GSE58545-RET	Thyroid Cancer	26
GSE174302-C	Colorectal Cancer	100	GSE58689	Thyroid Cancer	83
GSE174302-Li	Liver Cancer	73	GSE6044-ACC	Lung Cancer	15
GSE174302-Lu	Lung Cancer	81	GSE6044-SCC	Lung Cancer	15
GSE174302-S	Gastric Cancer	83	GSE6044-SCLC	Lung Cancer	14
GSE179252	Gastric Cancer	76	GSE6344-STG1	Renal Carcinoma	20
GSE180440	Colorectal Cancer	190	GSE6344-STG1	Renal Carcinoma	20
GSE181674	Type 1 Diabetes	15	GSE6344-STG2	Renal Carcinoma	20
GSE182424	Inflammatory Bowel Disease	12	GSE6344-STG2	Renal Carcinoma	20
GSE182923	Type 2 Diabetes	18	GSE6357	Renal Carcinoma	18
GSE183325	Inflammatory Bowel Disease	23	GSE63678-CC	Cervical Cancer	10
GSE183533	COVID-19	41	GSE63678-EC	Endometrial Carcinoma	12
GSE183947	Breast Cancer	43	GSE6613	Parkinson's	72
GSE184050	Type 2 Diabetes	116	GSE68172	Acute Myeloid Leukemia	77
GSE184336	Gastric Cancer	461	GSE6956	Prostate Cancer	84
GSE184950	Parkinson's	16	GSE71766-Influenza	Influenza	96
GSE185051	Non-alcoholic fatty liver disease	57	GSE71766-Rhino	Influenza	96
GSE190496	COVID-19	30	GSE7305	Endometrial Carcinoma	20
GSE191139	Gastric Cancer	8	GSE73655	Huntington's	12
GSE192804	Cervical Cancer	32	GSE73655-PRE	Huntington's	15
GSE193436	Type 2 Diabetes	12	GSE7621	Parkinson's	25
GSE193438-BG	Alzheimer's	8	GSE7803-HG	Cervical Cancer	17
GSE193438-Hi	Alzheimer's	8	GSE7803-IV	Cervical Cancer	31
GSE193438-PL	Alzheimer's	8	GSE781	Renal Carcinoma	24
GSE193438-SN	Alzheimer's	8	GSE781	Renal Carcinoma	24
GSE197698-C	Inflammatory Bowel Disease	44	GSE79962-DCM	Dilated Cardiomyopathy	20
GSE197698-I	Inflammatory Bowel Disease	106	GSE79962-ICM	Dilated Cardiomyopathy	33
GSE202151	Type 2 Diabetes	12	GSE8397-L1	Parkinson's	32
GSE202182	COVID-19	14	GSE8397-L2	Parkinson's	32
GSE203206	Alzheimer's	47	GSE8397-M1	Parkinson's	46
GSE207435	Liver Cancer	54	GSE8397-M2	Parkinson's	46
GSE211979	COVID-19	13	GSE8397-f1	Parkinson's	16
GSE213324	Renal Carcinoma	41	GSE8397-f2	Parkinson's	16
GSE214846	Liver Cancer	115	GSE84422-Am	Alzheimer's	51
GSE217948	COVID-19	467	GSE84422-AC1	Alzheimer's	118
GSE40710	Parkinson's	33	GSE84422-AC2	Alzheimer's	118
GSE48850	Thyroid Cancer	11	GSE84422-CN1	Alzheimer's	104
GSE50760	Colorectal Cancer	36	GSE84422-CN2	Alzheimer's	104
GSE52194	Breast Cancer	20	GSE84422-DPC1	Alzheimer's	114
GSE53695	Alzheimer's	17	GSE84422-DPC2	Alzheimer's	114
GSE53697	Alzheimer's	17	GSE84422-FP1	Alzheimer's	126
GSE58135	Breast Cancer	105	GSE84422-FP2	Alzheimer's	126
GSE63420	Liver Cancer	14	GSE84422-H1	Alzheimer's	110
GSE64810	Huntington's	69	GSE84422-H2	Alzheimer's	110
GSE66207	Inflammatory Bowel Disease	33	GSE84422-IFG1	Alzheimer's	106
GSE69197	Renal Carcinoma	6	GSE84422-IFG2	Alzheimer's	106
GSE69240	Breast Cancer	35	GSE84422-ITG1	Alzheimer's	116
GSE72820	Colorectal Cancer	14	GSE84422-ITG2	Alzheimer's	116
GSE73708	Liver Cancer	12	GSE84422-MTG1	Alzheimer's	116
GSE74369	Colorectal Cancer	16	GSE84422-MTG2	Alzheimer's	116
GSE77635	Colorectal Cancer	20	GSE84422-NA	Alzheimer's	51
GSE80183-ENA	Systemic lupus erythematosus	8	GSE84422-OVC1	Alzheimer's	106

Table 9.1: Continued from the previous page.

GEO ID	Disease	#Samples	GEO ID	Disease	#Samples
GSE80183-dsDNA	Systemic lupus erythematosus	8	GSE84422-OVC2	Alzheimer's	106
GSE80183-dsDNAENA	Systemic lupus erythematosus	8	GSE84422-PaG1	Alzheimer's	120
GSE80609	Prostate Cancer	45	GSE84422-PaG2	Alzheimer's	120
GSE81089	Non-small cell lung cancer	218	GSE84422-PCC1	Alzheimer's	116
GSE81928	Liver Cancer	26	GSE84422-PCC2	Alzheimer's	116
GSE83687-CDleft	Inflammatory Bowel Disease	12	GSE84422-PrG1	Alzheimer's	98
GSE83687-CDsmall	Inflammatory Bowel Disease	19	GSE84422-PrG2	Alzheimer's	98
GSE83687-LCCr	Inflammatory Bowel Disease	12	GSE84422-PrC1	Alzheimer's	112
GSE83687-LCUl	Inflammatory Bowel Disease	26	GSE84422-PrC2	Alzheimer's	112
GSE83687-RCUl	Inflammatory Bowel Disease	22	GSE84422-P1	Alzheimer's	104
GSE83687-SB	Inflammatory Bowel Disease	19	GSE84422-P2	Alzheimer's	104
GSE83687-UCleft	Inflammatory Bowel Disease	26	GSE84422-SPL1	Alzheimer's	100
GSE83687-UCright	Inflammatory Bowel Disease	22	GSE84422-SPL2	Alzheimer's	100
GSE84013	Huntington's	32	GSE84422-STG1	Alzheimer's	120
GSE84073	Liver Cancer	14	GSE84422-STG2	Alzheimer's	120
GSE87096	Colorectal Cancer	12	GSE84422-TP1	Alzheimer's	116
GSE87592	Liver Cancer	52	GSE84422-TP2	Alzheimer's	116
GSE88888	Alzheimer's	20	GSE85457	Thyroid Cancer	7
GSE89122	Renal Carcinoma	13	GSE8671-AC	Colorectal Cancer	8
GSE94660	Liver Cancer	42	GSE8671-DC	Colorectal Cancer	8
GSE95132	Colorectal Cancer	20	GSE8671-REC	Colorectal Cancer	14
GSE95437-CD	Inflammatory Bowel Disease	49	GSE8671-SC	Colorectal Cancer	32
GSE95437-UC	Inflammatory Bowel Disease	54	GSE8762	Huntington's	22
GSE95587	Alzheimer's	117	GSE92778	Acute Myeloid Leukemia	12
GSE97214	Liver Cancer	24	GSE9348	Colorectal Cancer	82
GSE97901	Prostate Cancer	45	GSE9476-BM	Acute Myeloid Leukemia	25
GSE100054	Parkinson's	19	GSE9476-PB	Acute Myeloid Leukemia	39
GSE12685	Alzheimer's	14	GSE99039	Parkinson's	558
GSE1297	Alzheimer's	31			

Data processing

For microarray datasets, we utilized the oligo package [235] in R to process the raw CEL files and obtain gene expression values. We applied the Robust Multi-array Average (RMA) method [236] for background correction, normalization, and summarization of the probe-level expression values. We then mapped the probe IDs to Entrez gene IDs for each dataset using annotation packages available in Bioconductor corresponding to the platform used in the dataset. For example, a dataset generated using the *Affymetrix Human Genome U133 Plus 2.0 Array* platform with GEO platform ID GPL570 was mapped to Entrez gene IDs using the *hgu133plus2.db* package. The full list of annotation packages used for each platform is available in Table 9.2. The ID mapping process consists of two steps. First, we performed differential analysis using the limma package [232] and obtained the probe-level differential expression results. Next, we mapped from probe IDs to Entrez IDs using the corresponding annotation package. This is a many-to-many mapping, as a single probe can map to multiple

Table 9.2: The annotation packages used to map from probe IDs to Entrez gene IDs for each microarray platform. Abbreviations: Affy.: Affymetrix.

GEO ID	Platform name	Annotation package
GPL96	Affy. Human Genome U133A Array	hgu133a.db
GPL97	Affy. Human Genome U133B Array	hgu133b.db
GPL570	Affy. Human Genome U133 Plus 2.0 Array	hgu133plus2.db
GPL6244	Affy. Human Gene 1.0 ST Array	hugene10sttranscriptcluster.db
GPL4866	Affy. Human Genome U133 Plus 2.0 Array	hgu133plus2.db
GPL16311	Affy. Human Genome U133 Plus 2.0 Array	hgu133plus2.db
GPL571	Affy. Human Genome U133A 2.0 Array	hgu133a2.db
GPL10739	Affy. Human Gene 1.0 ST Array	hugene10sttranscriptcluster.db
GPL201	Affy. Human HG-Focus Target Array	hgfocus.db
GPL80	Affy. Human Full Length HuGeneFL Array	hu6800.db
GPL13667	Affy. Human Genome U219 Array	hgu219.db
GPL23126	Affy. Human Clariom D Assay	clariomdhumantranscriptcluster.db

genes, and a single gene can be mapped by multiple probes. In cases where there is a one-to-many relationship from probe to genes, we only retained the first match. Subsequently, in cases of a one-to-many relationship from gene to probes, we kept the expression values of the most significant probe for each gene. This approach ensures the transformation of microarray data into gene-level information while preserving the relevance of the most significant probes in the subsequent analysis.

For RNA-Seq datasets, we have two types of gene expression data for each dataset: raw counts and TPM-normalized expression values. Both types of gene expression data have Ensembl gene IDs as the gene identifiers. For raw counts data, we normalized the counts data using the default median of ratios normalization method implemented in the DESeq2 package [237]. For TPM-normalized data, we only applied log (base 2) transformation to the expression values. We then map the Ensembl gene IDs to Entrez gene IDs for each dataset using the annotation package *org.Hs.eg.db*. Similar to microarray data processing, the ID mapping process consists of two steps. First, we performed a differential analysis of the gene-level differential expression results. For raw counts data, we used the DESeq2 package [237] to perform differential analysis, and for TPM-normalized data, we used the limma package. We then mapped from Ensembl IDs to Entrez IDs as described for the microarray datasets.

9.2.2 GDC repositories

Data collection

Using the Genomic Data Commons data portal, we obtained RNA-Seq datasets for specific project IDs: BLCA, BRCA, CESC, COAD, ESCA, GBM, KICH, KIRC, KIRP, LIHC, LUAD, LUSC, PAAD, PCPG, PRAD, STAD, and THCA. Other projects were excluded from the analysis due to a lack of sufficient samples labeled as normal or tumor, or a lack of explicit target pathways in the KEGG databases. We also acquired RNA-Seq datasets from other GDC projects, including REBC-THYR, ORGANOID-PANCREATIC, CPTAC-3, TARGET-AML, BEATAML1.0-COHORT, and OHSU-CNL. We downloaded the raw counts and aliquot files for each project from the GDC portal. The aliquot files contain metadata information, such as the sample type, sample ID, and project ID, while the raw counts files contain the gene expression values with Ensembl gene IDs.

Many of the TCGA projects have a notable imbalance in sample distribution among specific projects, such as CESC, GBM, PAAD, and PCPG, where the number of normal samples is considerably lower compared to the number of tumor samples. This sample imbalance raises concerns about potential biases and poses challenges in achieving statistically robust analyses within these projects. To address this issue, we consistently incorporated control samples obtained from the Genotype-Tissue Expression (GTEx) repository [238] in all of the included TCGA projects. The GTEx database comprises a diverse collection of human tissues, including the brain, heart, and whole blood, constituting a comprehensive dataset with over 30,000 samples and 50 tissues. We acquired the gene read counts data for each tissue from the RNASeq datasets available under the *Bulk Tissue Expression* tab on the GTEx portal. The downloaded gene read count files were featured with Ensemble IDs and corresponding gene names.

TCGA data processing

For each TCGA dataset, we first filtered the samples by retaining those originating from fresh frozen tissue while excluding those obtained from FFPE (Formalin-Fixed Paraffin Embedded) sources. Next, we identified samples as either tumor or normal based on their Sample Type Code in the TCGA barcode (https://gdc.cancer.gov/resources-tcga-users/tcga-code-tables/sample-type-codes). Specifically, we retained samples with the Sample Type Code 01 (Primary solid Tumor) and 03 (Primary Blood Derived Cancer - Peripheral Blood) and 09 (Primary Blood Derived Cancer - Bone Marrow) as tumor samples, and samples with the Sample Type Code larger than 09 as normal samples (e.g., Blood Derived Normal, Solid Tissue Normal, etc.). Other sample types, such as metastatic samples, were excluded from the analysis. We then further filtered the samples in each group (tumor or normal) to retain only one sample per patient to ensure that each sample was independent. We prioritized the sample with the highest plate value, followed by the analyte order (H, R, T, D, G, W, and X) from the TCGA Code Tables (https://gdc.cancer.gov/resources-tcga-users/tcga-code-tables/portion-analyte-codes), and the portion value.

We then aligned the TCGA datasets with the GTEx datasets based on the corresponding tissues of origin. In the case of the tissue of the TCGA dataset matched with multiple tissues in the GTEx dataset, we duplicated the TCGA dataset and matched each duplicated dataset with one of the matched tissues in the GTEx dataset. This resulted in a total of 35 TCGA-GTEx pairs. The table 9.3 provides details about the alignment of the TCGA datasets with the GTEx datasets. Next, for each TCGA dataset, we removed genes with zero expression values across all samples. We performed the same procedure for the corresponding GTEx datasets (per tissue). We merged the matrices of TCGA and GTEx counts data using only the genes that are present in both datasets. We then use the ComBatSeq algorithm [239] to integrate the TCGA and GTEx datasets, remove batch effects, and obtain the adjusted counts matrix. We provided the batch and group information to the ComBatSeq algorithm, where the batch was defined between TCGA and GTEx samples, and the group was defined between tumor and control samples.

Table 9.3: Summary of TCGA and GTEx Data

Project ID	**Tissue**	**GTEx Tissue**	**#Tumor**	**#Normal**	**#GTEx**
BLCA	Bladder	Bladder	406	19	21
BRCA	Breast	Breast mammary	1094	113	459
CESC	Cervix uteri	Uterus	304	3	142
COAD	Colon (whole)	Colon sigmoid	458	41	373
		Colon transverse	458	41	406
ESCA	Esophagus	Esophagus gastroesophageal junction	184	13	375
		Esophagus mucosa	184	13	555
		Esophagus muscularis	184	13	515
GBM	Brain	Brain amygdala	155	5	152
		Brain anterior cingulate cortex	155	5	176
		Brain caudate basal ganglia	155	5	246
		Brain cerebellar hemisphere	155	5	215
		Brain cerebellum	155	5	241
		Brain cortex	155	5	255
		Brain frontal cortex	155	5	209
		Brain hippocampus	155	5	197
		Brain hypothalamus	155	5	202
		Brain nucleus accumbens basal ganglia	155	5	246
		Brain putamen basal ganglia	155	5	205
		Brain spinal cord cervical	155	5	159
		Brain substantia nigra	155	5	139
KICH	Kidney	Kidney cortex	66	25	85
		Kidney medulla	66	25	4
KIRC	Kidney	Kidney cortex	533	72	85
		Kidney medulla	533	72	4
KIRP	Kidney	Kidney cortex	290	32	85
		Kidney medulla	290	32	4
LIHC	Liver	Liver	371	50	226
LUAD	Lung	Lung	516	59	578
LUSC	Lung	Lung	501	51	578
PAAD	Pancreas	Pancreas	178	4	328
PCPG	Adrenal gland	Adrenal gland	179	3	258
PRAD	Prostate gland	Prostate	497	52	245
STAD	Stomach	Stomach	412	36	359
THCA	Thyroid gland	Thyroid	505	59	653

We then applied the ID mapping procedure similar to the GEO RNA-Seq counts data processing to convert Ensemble gene IDs into Entrez gene IDs for each integrated TCGA and GTEx project using the obtained adjusted counts matrix. This resulted in a counts matrix with Entrez gene IDs. Next, we use Formula 9.2 to obtain the TPM-normalized expression values for each integrated TCGA and GTEx project.

$$\text{rate}_{ij} = \frac{\text{counts}_{ij}}{\text{lengths}_i} \times \frac{1}{\sum_{k=1}^{m} \frac{\text{counts}_{kj}}{\text{lengths}_k}} \times 1e6 \tag{9.2}$$

Other GDC projects data processing

For each GDC project, we first excluded samples without reported tissues. We then divided the data into specific sub-projects based on tissue type. This results in a total of 15 sub-projects. The Table 9.4 provides details about the sub-projects obtained from the GDC portal. We then assigned the sample labels (tumor or normal) based on the tissue type specified for each sample. Finally, we applied the same ID mapping and TPM normalization procedures as described for the TCGA datasets to the obtained counts matrix for each project.

9.3 Results

9.3.1 PGSA improves the significance ranking of targeted gene sets

The main goal of gene set enrichment analysis is to identify what gene sets are significantly enriched among a large number of gene sets tested. The top most significant gene sets (i.e., those with the smallest enrichment p-values) are considered to be the most relevant gene sets to the biological condition of interest and will be further investigated. In this analysis, we select one gene set for each condition that is known to be associated with the disease of interest, and assess the performance of PGSA and other methods in ranking the significance of these gene sets when analyzing expression data from a wide range of human diseases. We refer to these gene sets as targeted gene sets. For example, with the KEGG database, the targeted gene set of a dataset obtained from an Alzheimer's disease study is the *Alzheimer disease - Homosapiens (human)* gene set with accession id of hsa05010. While the targeted gene sets might not always

Table 9.4: Summary of GDC RNA-Seq projects.

Project ID	Tissue	Sub-project	#Tumor	#Normal
BEATAML1.0-COHORT	Acute Myeloid Leukemia	Blood	131	21
OHSU-CNL	Acute Myeloid Leukemia	Blood	66	4
TARGET-AML	Acute Myeloid Leukemia	Blood	124	31
		Bone marrow	953	324
CPTAC-3	Lung Cancer	Lower part of lung	94	85
		Whole lung	87	80
		Middle part of lung	15	14
		Upper part of lung	142	132
CPTAC-3	Endometrial Carcinoma	Endometrium	174	20
		Uteri	273	25
CPTAC-3	Pancreatic Cancer	Body of pancreas	18	6
		Head of pancreas	137	50
CPTAC-3	Renal Carcinoma	Kidney	312	149
ORGANOID-PANCREATIC	Pancreatic Cancer	Pancreas	44	11
REBC-THYR	Thyroid Cancer	Thyroid gland	428	400

be the most impacted gene sets in the analysis, a good enrichment method should be able to identify the targeted gene sets as significant and are more significant than other non-targeted gene sets, i.e., the targeted gene sets should have better ranks than other gene sets.

To perform a comprehensive assessment of the performance of PGSA, we collected a large number of datasets from the Gene Expression Omnibus (GEO) database and the Genomic Data Commons (GDC) portal. In total, we collected 421 datasets, including 371 datasets from the GEO database and 50 datasets from the GDC portal. The GEO datasets include 188 microarray datasets and 183 RNA-Seq datasets that come from 29 human diseases, including: Alzheimer's disease, Atopic Dermatitis, Dilated Cardiomyopathy (DCM), Huntington's disease, Inflammatory Bowel Disease (IBD), Influenza,COVID-19, Non-alcoholic fatty liver disease (NAFLD), Parkinson's disease, Systemic lupus erythematosus (SLE), Type 1 Diabetes, Type 2 Diabetes, Acute Myeloid Leukemia (AML), Brain Cancer, Breast Cancer, Cervical Cancer, Chronic Myeloid Leukemia (CML), Colorectal Cancer, Endometrial Cancer, Gastric Cancer, Glioma, Liver Cancer, Lung Cancer/Small Cell Lung Cancer (LC/SCLC), Non-small cell lung cancer (NSCLC), Pancreatic Cancer, Prostate Cancer, Renal Carcinoma, and Thyroid Cancer. The GEO datasets were selected based on the availability of the targeted gene sets in the KEGG database or the WikiPathways database. The GDC datasets include 35 datasets

from The Cancer Genome Atlas (TCGA), 9 datasets from the National Cancer Institute's Clinical Proteomic Tumor Analysis Consortium (CPTAC), and 6 datasets from other projects. All GDC datasets are RNA-Seq datasets from cancer studies. For RNA-Seq datasets, we use both counts data and transcripts per million (TPM) normalized data in our analysis.

We compare the performance of PGSA with 10 other methods, including 6 non-topology-based methods (ORA, FGSEA, GSEA, GSA, PADOG, and GAGE) and 4 topology-based methods (SPIA, CePaORA, CePaGSA, and PathNet). We use two pathway databases, KEGG and WikiPathways, as the sources of gene sets and run each method on each database separately. The KEGG database contains 336 gene sets and the WikiPathways database contains 798 gene sets. We run all 11 methods on the KEGG database and only run the non-topology-based methods on the WikiPathways database since the topology-based methods do not support the WikiPathways database (see Methods section for details of software packages and settings). For all analyses, the targeted gene sets are only used a posteriori to assess the results. The list of the targeted gene sets for all diseases is presented in Table 9.5.

Table 9.5: The list of the targeted gene sets from the two datbases: KEGG and WikiPathways

Disease	**Database**	**Pathway ID**	**Pathway Name**
Liver Cancer	KEGG	hsa05225	Hepatocellular carcinoma
DCM	KEGG	hsa05414	Dilated cardiomyopathy
CML	KEGG	hsa05220	Chronic myeloid leukemia
Type 1 Diabetes	KEGG	hsa04940	Type I diabetes mellitus
Type 2 Diabetes	KEGG	hsa04930	Type II diabetes mellitus
IDB	KEGG	hsa05321	Inflammatory bowel disease
Breast Cancer	KEGG	hsa05224	Breast cancer
Colorectal Cancer	KEGG	hsa05210	Colorectal cancer
Gastric Cancer	KEGG	hsa05226	Gastric cancer
Parkinson's	KEGG	hsa05012	Parkinson disease
Tuberculosis Infection	KEGG	hsa05152	Tuberculosis
Alzheimer's	KEGG	hsa05010	Alzheimer disease
Cervical Cancer	KEGG	hsa05165	Human papillomavirus infection
Pancreatic Cancer	KEGG	hsa05212	Pancreatic cancer
SLE	KEGG	hsa05322	Systemic lupus erythematosus
NAFLD	KEGG	hsa04932	Non-alcoholic fatty liver disease
Huntington's	KEGG	hsa05016	Huntington disease
NSCLC	KEGG	hsa05223	Non-small cell lung cancer
Glioma	KEGG	hsa05214	Glioma
Endometrial Cancer	KEGG	hsa05213	Endometrial cancer
COVID-19	KEGG	hsa05171	Coronavirus disease - COVID-19
Influenza	KEGG	hsa05164	Influenza A
Renal Carcinoma	KEGG	hsa05211	Renal cell carcinoma
Atopic Dermatitis	KEGG	hsa04621	NOD-like receptor signaling pathway
Thyroid Cancer	KEGG	hsa05216	Thyroid cancer
LC/SCLC	KEGG	hsa05222	Small cell lung cancer
Prostate Cancer	KEGG	hsa05215	Prostate cancer
AML	KEGG	hsa05221	Acute myeloid leukemia
Brain Cancer	KEGG	hsa05214	Glioma
Liver Cancer	WikiPathways	WP3646	Hepatitis C and hepatocellular carcinoma
DCM	WikiPathways	WP3668	Hypothesized pathways in pathogenesis of cardiovascular disease
CML	WikiPathways	WP3640	Imatinib and chronic myeloid leukemia

Table 9.5: The list of the targeted gene sets from the two datbases: KEGG and WikiPathways

Disease	Database	Pathway ID	Pathway Name
Type 1 Diabetes	WikiPathways	WP1584	Type II diabetes mellitus
Type 2 Diabetes	WikiPathways	WP1584	Type II diabetes mellitus
IDB	WikiPathways	WP5198	Inflammatory bowel disease signaling
Breast Cancer	WikiPathways	WP4262	Breast cancer pathway
Colorectal Cancer	WikiPathways	WP4290	Metabolic reprogramming in colon cancer
Gastric Cancer	WikiPathways	WP2361	Gastric cancer network 1
Parkinson's	WikiPathways	WP2371	Parkinson 39 s disease pathway
Tuberculosis Infection	WikiPathways	WP4197	Immune response to tuberculosis
Alzheimer's	WikiPathways	WP5124	Alzheimer 39 s disease
Pancreatic Cancer	WikiPathways	WP4263	Pancreatic adenocarcinoma pathway
NAFLD	WikiPathways	WP4396	Nonalcoholic fatty liver disease
Huntington's	WikiPathways	WP3853	ERK pathway in Huntington 39 s disease
NSCLC	WikiPathways	WP4255	Non small cell lung cancer
Glioma	WikiPathways	WP2261	Glioblastoma signaling pathways
Endometrial Cancer	WikiPathways	WP4155	Endometrial cancer
COVID-19	WikiPathways	WP4846	SARS CoV 2 and COVID 19 pathway
Renal Carcinoma	WikiPathways	WP4241	Type 2 papillary renal cell carcinoma
Thyroid Cancer	WikiPathways	WP4928	MAPK pathway in congenital thyroid cancer
LC/SCLC	WikiPathways	WP4255	Non small cell lung cancer
Brain Cancer	WikiPathways	WP2261	Glioblastoma signaling pathways
BEATAML1.0-COHORT	KEGG	hsa05221	Acute myeloid leukemia
CPTAC-3	KEGG	hsa05212	Pancreatic cancer
CPTAC-3	KEGG	hsa05213	Endometrial cancer
CPTAC-3	KEGG	hsa05211	Renal cell carcinoma
CPTAC-3	KEGG	hsa05222	Small cell lung cancer
OHSU-CNL	KEGG	hsa05221	Acute myeloid leukemia
ORGANOID-PANCREATIC	KEGG	hsa05212	Pancreatic cancer
REBC-THYR	KEGG	hsa05216	Thyroid cancer
TARGET-AML	KEGG	hsa05221	Acute myeloid leukemia
TCGA-BLCA	KEGG	hsa05219	Bladder cancer
TCGA-BRCA	KEGG	hsa05224	Breast cancer
TCGA-CESC	KEGG	hsa05165	Human papillomavirus infection
TCGA-COAD	KEGG	hsa05210	Colorectal cancer
TCGA-ESCA	KEGG	hsa04110	Cell cycle
TCGA-GBM	KEGG	hsa05214	Glioma
TCGA-KICH	KEGG	hsa05211	Renal cell carcinoma
TCGA-KIRC	KEGG	hsa05211	Renal cell carcinoma
TCGA-KIRP	KEGG	hsa05211	Renal cell carcinoma
TCGA-LIHC	KEGG	hsa05225	Hepatocellular carcinoma
TCGA-LUAD	KEGG	hsa05223	Non-small cell lung cancer
TCGA-LUSC	KEGG	hsa05223	Non-small cell lung cancer
TCGA-PAAD	KEGG	hsa05212	Pancreatic cancer
TCGA-PCPG	KEGG	hsa00020	Citrate cycle (TCA cycle)
TCGA-PRAD	KEGG	hsa05215	Prostate cancer
TCGA-STAD	KEGG	hsa05226	Gastric cancer
TCGA-THCA	KEGG	hsa05216	Thyroid cancer
CPTAC-3	WikiPathways	WP4263	Pancreatic adenocarcinoma pathway
CPTAC-3	WikiPathways	WP4155	Endometrial cancer
CPTAC-3	WikiPathways	WP4241	Type 2 papillary renal cell carcinoma
CPTAC-3	WikiPathways	WP4255	Non small cell lung cancer
ORGANOID-PANCREATIC	WikiPathways	WP4263	Pancreatic adenocarcinoma pathway
REBC-THYR	WikiPathways	WP4928	MAPK pathway in congenital thyroid cancer
TCGA-BLCA	WikiPathways	WP2828	Bladder cancer
TCGA-BRCA	WikiPathways	WP4262	Breast cancer pathway
TCGA-COAD	WikiPathways	WP4290	Metabolic reprogramming in colon cancer
TCGA-GBM	WikiPathways	WP2261	Glioblastoma signaling pathways
TCGA-KICH	WikiPathways	WP4241	Type 2 papillary renal cell carcinoma
TCGA-KIRC	WikiPathways	WP4241	Type 2 papillary renal cell carcinoma
TCGA-KIRP	WikiPathways	WP4241	Type 2 papillary renal cell carcinoma
TCGA-LIHC	WikiPathways	WP3646	Hepatitis C and hepatocellular carcinoma
TCGA-LUAD	WikiPathways	WP4255	Non small cell lung cancer
TCGA-LUSC	WikiPathways	WP4255	Non small cell lung cancer
TCGA-PAAD	WikiPathways	WP4263	Pancreatic adenocarcinoma pathway
TCGA-STAD	WikiPathways	WP2361	Gastric cancer network 1

We use the ranking of the targeted gene sets based on their enrichment p-values as the main metric to assess the performance of the methods. Here, we use the unadjusted p-values to rank the gene sets, as the adjusted p-values might cause many gene sets to have the same p-value, especially with a large number of gene sets in the database. The smaller the enrichment p-value of a gene set, the higher the rank of the gene set. If two gene sets have the same enrichment p-value, the gene set with a smaller number of genes will have a higher rank. We then scale the ranks of the gene sets to the range of 0 to 1 to be able to compare the ranks across different gene set databases.

Performance on microarray datasets and TPM-normalized RNA-Seq datasets downloaded from the GEO database.

Figures 9.2 and 9.3 show the scaled ranks of the targeted gene sets for each method from the KEGG database and the WikiPathways database, respectively, using microarray datasets and TPM-normalized RNA-Seq datasets downloaded from the GEO database. From this point, unless otherwise stated, we refer to the results from the TPM-normalized RNA-Seq datasets as the results from the RNA-Seq datasets. In figures 9.2 and 9.3, we group the datasets by the condition they belong to and present the results for each condition in a subfigure. We also present the overall performance of the methods for only non-cancer datasets, only cancer datasets, only microarray datasets, only RNA-Seq datasets, and all datasets, respectively, in the last 5 subfigures. Using KEGG gene sets, PGSA achieves the highest median rank in 15 (out of 29) conditions. Across all analyses, PGSA achieves the highest median rank in 141 (out of 371) analyses, with a median rank of 0.90. This indicates that in most analyses, the targeted gene sets are ranked among the top 10% most significant gene sets by PGSA. With 336 gene sets analyzed from the KEGG database, the targeted gene sets are ranked among the top 34 most significant gene sets by PGSA in most analyses. The second-best method is Pathnet, with the highest median rank in 6 conditions and 39 analyses. The median rank of targeted gene sets by Pathnet is 0.81, i.e., the targeted gene sets are ranked among the top 19% most significant gene sets by Pathnet, or among the top 64 most significant gene sets. Overall, the ranking of targeted gene sets by PGSA is significantly better than the ranking by any of the other methods

(maximum p-value of 1.8×10^{-19} using one-sided Wilcoxon signed-rank test when comparing PGSA and any of the other methods). PGSA has consistent performance in non-cancer datasets and cancer datasets, with median ranks of 0.90 and 0.89, respectively. The method performs slightly better on microarray datasets than on RNA-Seq datasets, with median ranks of 0.94 and 0.89, respectively.

Using WikiPathways gene sets, PGSA achieves the highest median rank in 12 (out of 23) conditions. Note that the number of conditions analyzed for WikiPathways gene sets is smaller than that number for KEGG gene sets because not all conditions have a corresponding targeted gene set in the WikiPathways database. Across all analyses, PGSA achieves the highest median rank in 147 (out of 334) analyses, with a median rank of 0.91. With 798 gene sets analyzed from the WikiPathways database, the targeted gene sets are ranked among the top 72 most significant gene sets by PGSA in most analyses. The second best method is PADOG, which has a median rank of 0.80, i.e., the targeted gene sets are ranked among the top 160 most significant gene sets in most analyses. PADOG achieves the highest median rank in 4 conditions and 36 analyses. Overall, the ranking of targeted gene sets by PGSA is significantly better than the ranking by any of the other methods (maximum p-value of 5.6×10^{-6} using one-sided Wilcoxon signed-rank test when comparing PGSA and any of the other methods). PGSA has consistent performance in non-cancer datasets and cancer datasets, with median ranks of 0.89 and 0.91, respectively, as well as on microarray datasets and RNA-Seq datasets, with median ranks of 0.89 and 0.91, respectively.

We present the adjusted p-values of the targeted gene sets from the KEGG database and the WikiPathways database for each method in Figure 9.4 and 9.5, respectively. To adjust the p-values, we use the harmonic mean p-value (HMP) method [240]. The harmonic mean p-value method is most influenced by the smallest p-value, and is insensitive to the number of gene sets in the database, hence partially allowing us to compare the p-values across different databases (KEGG and WikiPathway). We also present the FDR-adjusted p-values of the targeted gene sets in Figure 9.6 and 9.7, respectively. Overall, PGSA can identify the targeted gene sets as significant (with a significance threshold of 0.05) in most analyses, regardless of the gene set database and the technology used to generate the data. In contrast, except for PathNet and

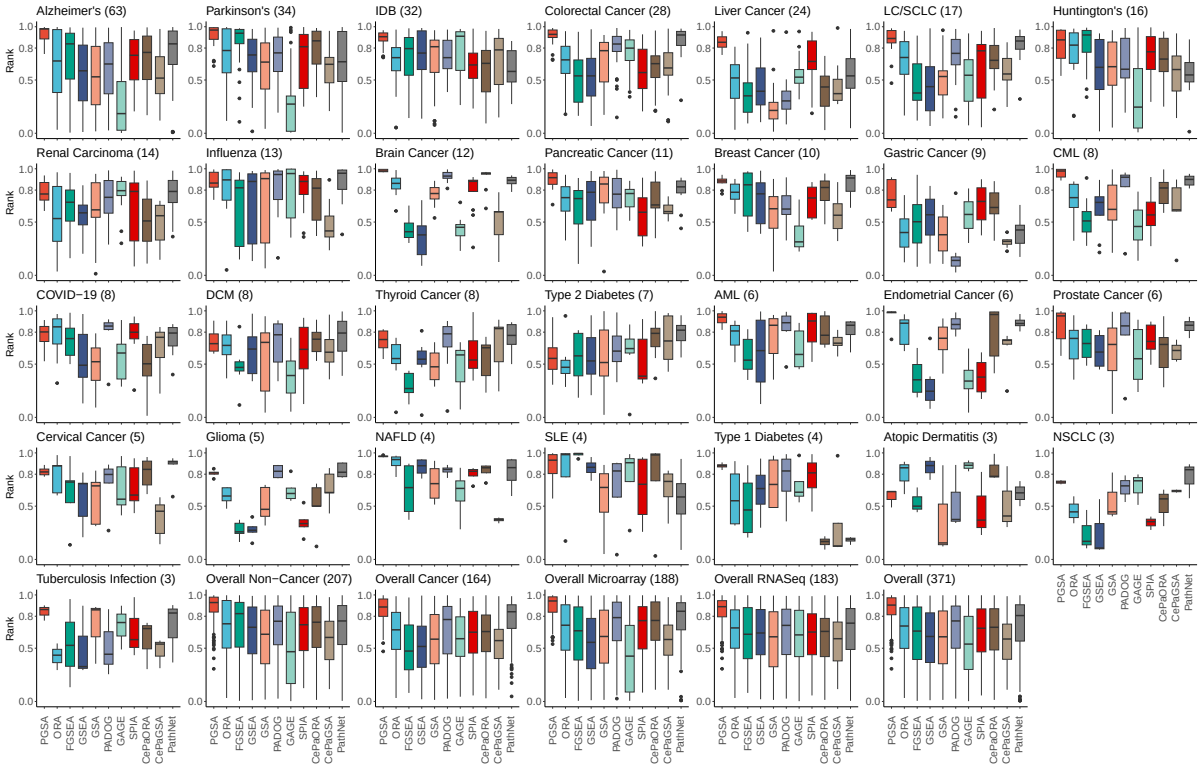

Figure 9.2: The scaled ranks of the targeted gene set from the KEGG database for the 11 methods using microarray datasets and TPM-normalized RNA-Seq datasets from the GEO database. Each subfigure shows the scaled ranks of the gene sets for a condition. The title of each subfigure shows the name of the condition and the number of datasets used for the condition. The x-axis represents the methods and the y-axis represents the scaled ranks of the gene sets. The last 5 subfigures show the overall performance of the methods for only non-cancer datasets, only cancer datasets, only microarray datasets, only RNA-Seq datasets, and all datasets, respectively.

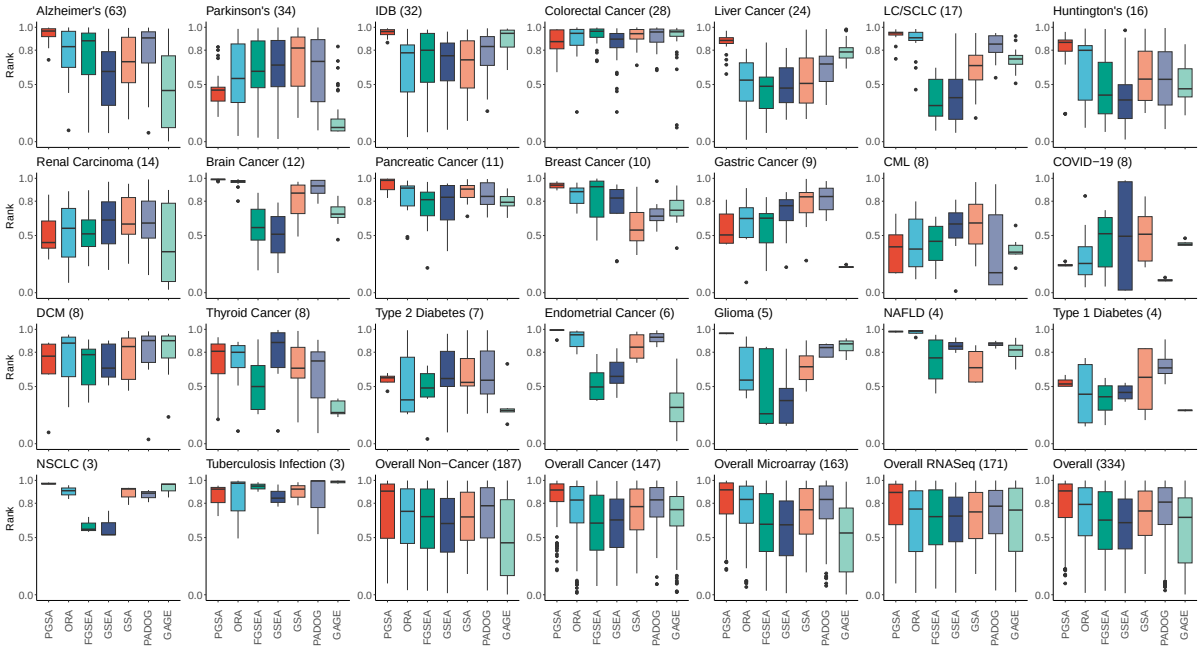

Figure 9.3: The scaled ranks of the targeted gene set from the WikiPathways database for 7 methods using microarray datasets and TPM-normalized RNA-Seq datasets from the GEO database. Each subfigure shows the scaled ranks of the gene sets for a condition. The title of each subfigure shows the name of the condition and the number of datasets used for the condition. The x-axis represents the methods and the y-axis represents the scaled ranks of the gene sets. The last 5 subfigures show the overall performance of the methods for only non-cancer datasets, only cancer datasets, only microarray datasets, only RNA-Seq datasets, and all datasets, respectively.

CePaGSA on KEGG databases, most of the other methods fail to identify the targeted gene sets as significant in most analyses (using both KEGG and WikiPathways databases). With these methods, using p-value as the main threshold for selecting significant gene sets will remove the targeted gene sets from downstream analysis. At the same time, since the ranking of the targeted gene sets by these methods is low (0.8 or lower), a threshold based on the ranking will be at least 20% of the gene sets for the targeted gene sets to be included in the downstream analysis. This is not practical in most cases, especially when the gene set database contains a large number of gene sets, such as the WikiPathways database with 798 gene sets. Table 9.6 shows the percentage of analyses in which the targeted gene sets are included in the list of significant gene sets for further analysis using a significance threshold of 0.05 (HMP adjusted) and different top percentages of gene sets used as the cutoff for significance. With the top 10% gene sets used as the cutoff for significance, PGSA identifies the targeted gene sets as

significant in 51% of the analyses using the KEGG database and 52% of the analyses using the WikiPathways database. The second best method for KEGG gene sets is PathNet with 27% of the analyses, and for WikiPathways gene sets is ORA with 31% of the analyses. Increasing the top percentage of gene sets also increases the percentage of analyses in which the targeted gene sets are identified as significant for most methods, except for GSEA, GSA, and PADOG, which identify the targeted gene sets as non-significant in most analyses. Among the other methods, CePaGSA and PathNet identify the targeted gene sets as significant in most analyses (317 and 224 out of 371 analyses for CePaGSA and PathNet, respectively, using the KEGG database). However, for CePaGSA, the median ranking of the targeted gene sets is 0.59 (only available for KEGG gene sets), indicating that more than 40% of the gene sets are identified as statistically significant and are more significant than the targeted gene sets. Consequently, with a cutoff of 10%, only 6% of the analyses will include the targeted gene sets in the list of significant gene sets for further analysis. PathNet performs relatively well in terms of ranking (median ranking of 0.81) and identifying the targeted gene sets as significant individually. However, when combining the two metrics, the targeted gene sets are only included in the list of significant gene sets for further analysis in 27% of the analyses with a cutoff of 10% using the KEGG database. This suggests that in some analyses, the targeted gene sets are either ranked low or are not identified as significant by PathNet.

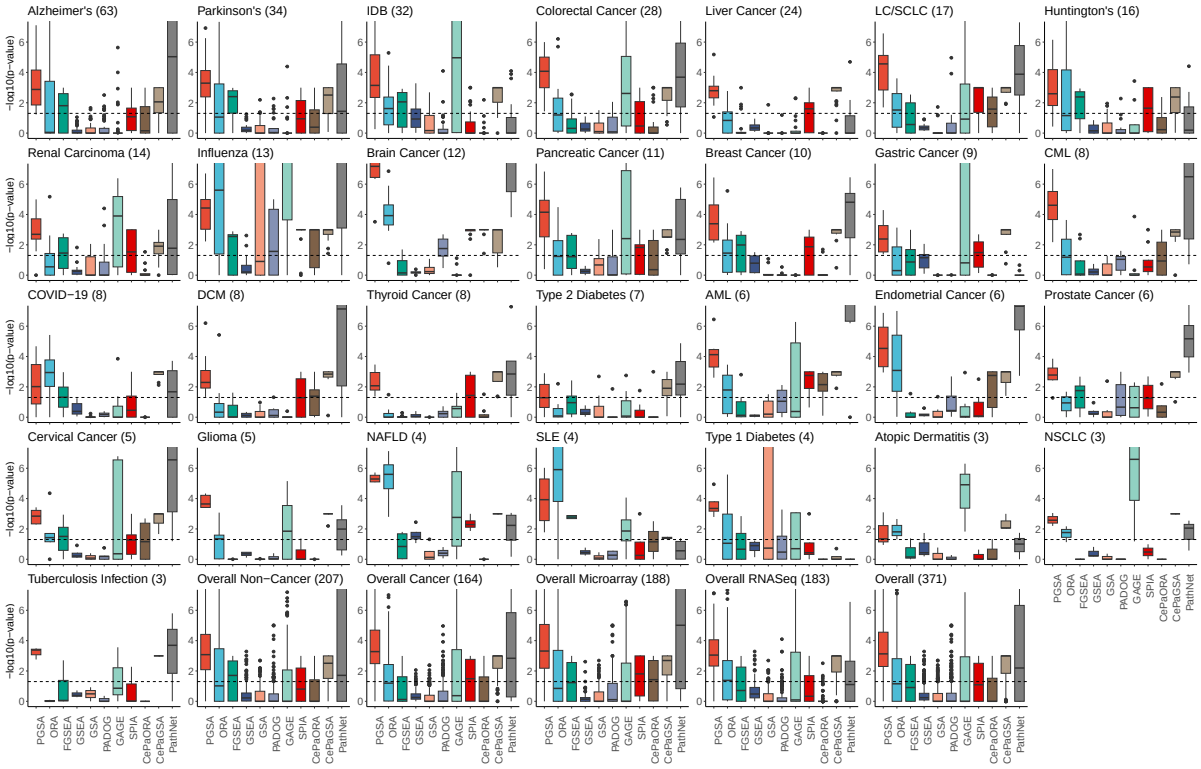

Figure 9.4: The HMP p-values of the targeted gene set from the KEGG database for the 11 methods using microarray datasets and TPM-normalized RNA-Seq datasets from the GEO database. Each subfigure shows the p-values of the gene sets for a condition. The title of each subfigure shows the name of the condition and the number of datasets used for the condition. The x-axis represents the methods and the y-axis represents the p-values of the gene sets. The horizontal dashed line represents the significance threshold of 0.05. The last 5 subfigures show the overall performance of the methods for only non-cancer datasets, only cancer datasets, only microarray datasets, only RNA-Seq datasets, and all datasets, respectively.

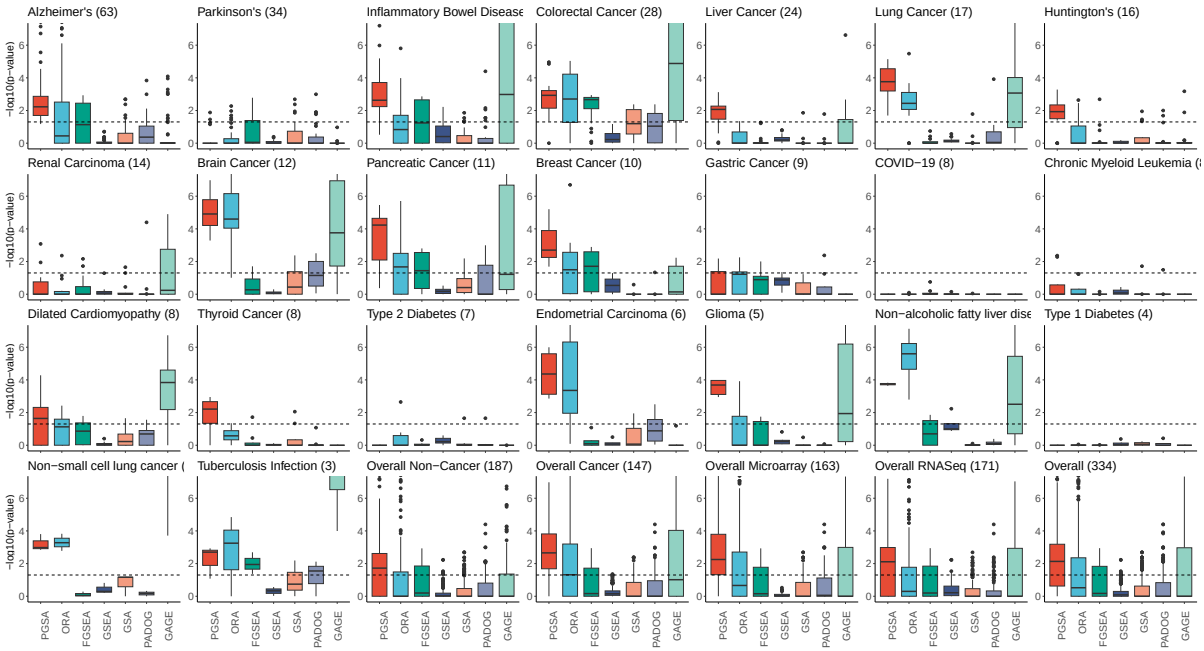

Figure 9.5: The HMP p-values of the targeted gene set from the WikiPathways database for 7 methods using microarray datasets and TPM-normalized RNA-Seq datasets from the GEO database.

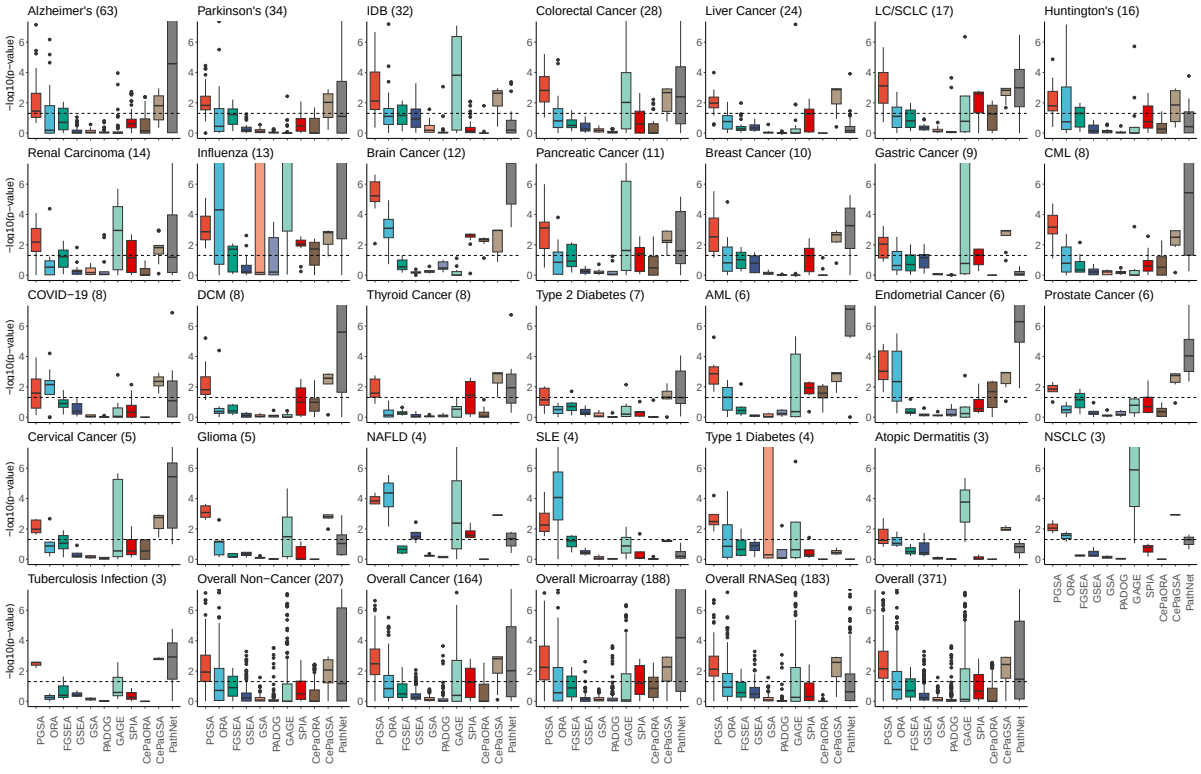

Figure 9.6: The FDR-adjusted p-values of the targeted gene sets from the KEGG database for the 11 methods using microarray datasets and TPM-normalized RNA-seq datasets from the GEO database. Each subfigure shows the p-values of the gene sets for a condition. The title of each subfigure shows the name of the condition and the number of datasets used for the condition. The x-axis represents the methods and the y-axis represents the p-values of the gene sets. The horizontal dashed line represents the significance threshold of 0.05. The last 5 subfigures show the overall performance of the methods for only non-cancer datasets, only cancer datasets, only microarray datasets, only RNA-seq datasets, and all datasets, respectively.

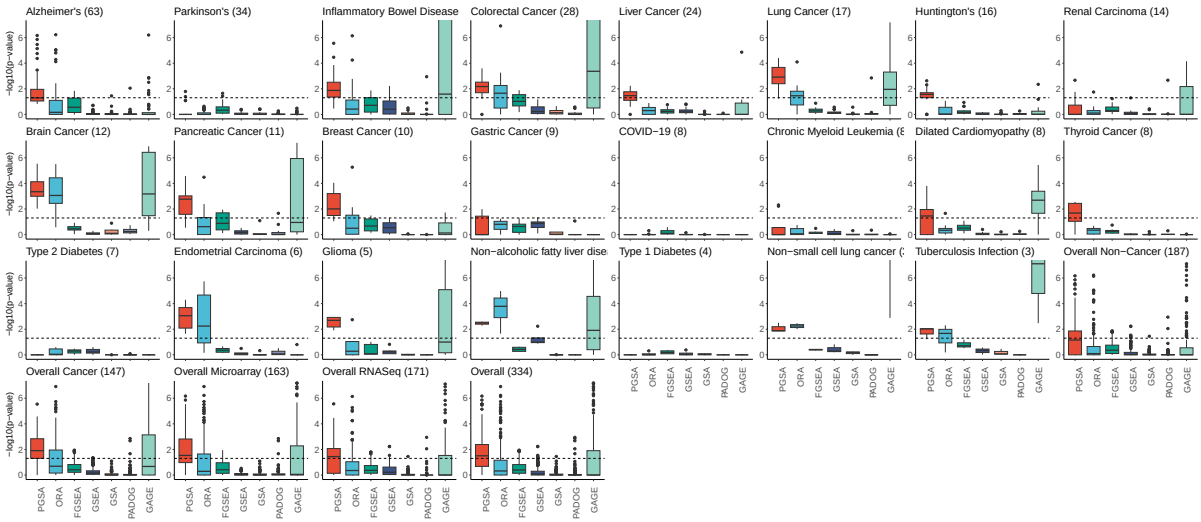

Figure 9.7: The FDR adjusted p-values of the targeted gene sets from the WikiPathways database for 7 methods using microarray datasets and TPM-normalized RNA-seq datasets from the GEO database.

Table 9.6: The percentage of analyses using GEO datasets in which the targeted gene sets are identified as significant by each method using a significance threshold of 0.05 (HMP adjusted). The first column shows the gene set database used in the analysis. The second column shows the top percentage of gene sets used as the cutoff for significance. The third to the last columns show the percentage of analyses in which the targeted gene sets are identified as significant by each method. All percentages are rounded to the nearest integer. The total number of analyses is 371 for KEGG gene sets and 334 for WikiPathways gene sets.

Database	Threshold	Method										
		PGSA	ORA	FGSEA	GSEA	GSA	PADOG	GAGE	SPIA	CePaORA	CePaGSA	PathNet
KEGG	10	51	23	22	4	11	14	15	16	16	6	27
	20	76	33	35	4	12	14	23	29	20	16	47
	30	86	40	42	5	12	14	28	35	24	26	54
WikiP.	10	52	31	25	1	16	15	20				
	20	66	36	31	2	16	15	27				
	30	69	37	32	3	16	15	33				

Performance on counts data of RNA-Seq datasets downloaded from the GEO database.

Figure 9.8, 9.9, 9.10, 9.11, 9.12, and 9.13 show the scaled ranks of the targeted gene sets and the adjusted p-values of the targeted gene sets for each method using counts data of the RNA-seq datasets from the GEO database. Overall, it is clear that PGSA still consistently outperforms other methods even when using the counts data instead of the TPM-normalized data.

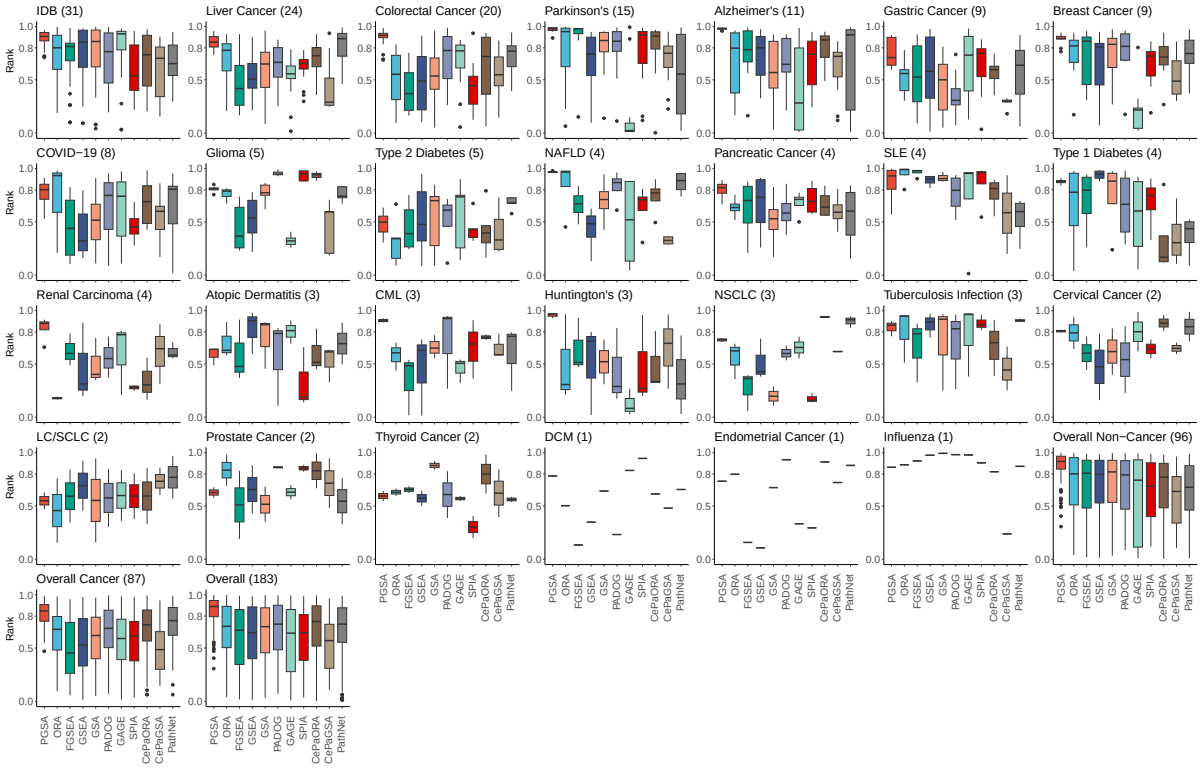

Figure 9.8: The scaled ranks of the targeted gene set from the KEGG database for the 11 methods using counts data of the RNA-seq datasets from the GEO database. Each subfigure shows the scaled ranks of the gene sets for a condition. The title of each subfigure shows the name of the condition and the number of datasets used for the condition. The x-axis represents the methods and the y-axis represents the scaled ranks of the gene sets. The last 3 subfigures show the overall performance of the methods for only non-cancer datasets, only cancer datasets, and all datasets, respectively.

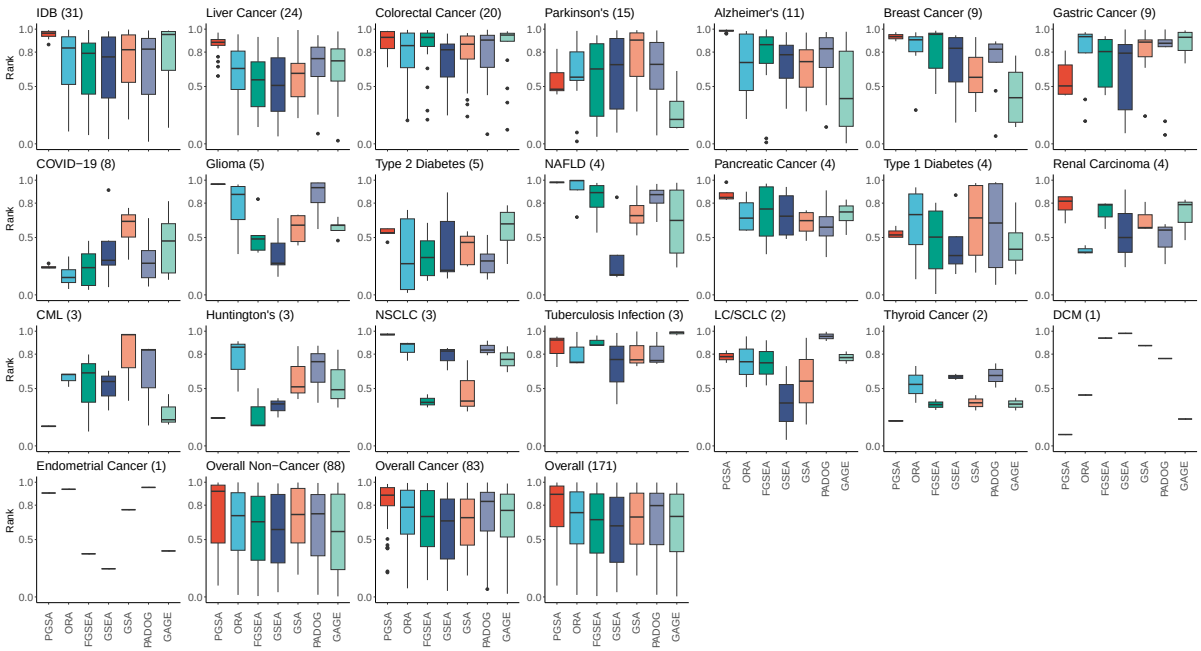

Figure 9.9: The scaled ranks of the targeted gene set from the WikiPathways database for 7 methods using counts data of the RNA-seq datasets from the GEO database.

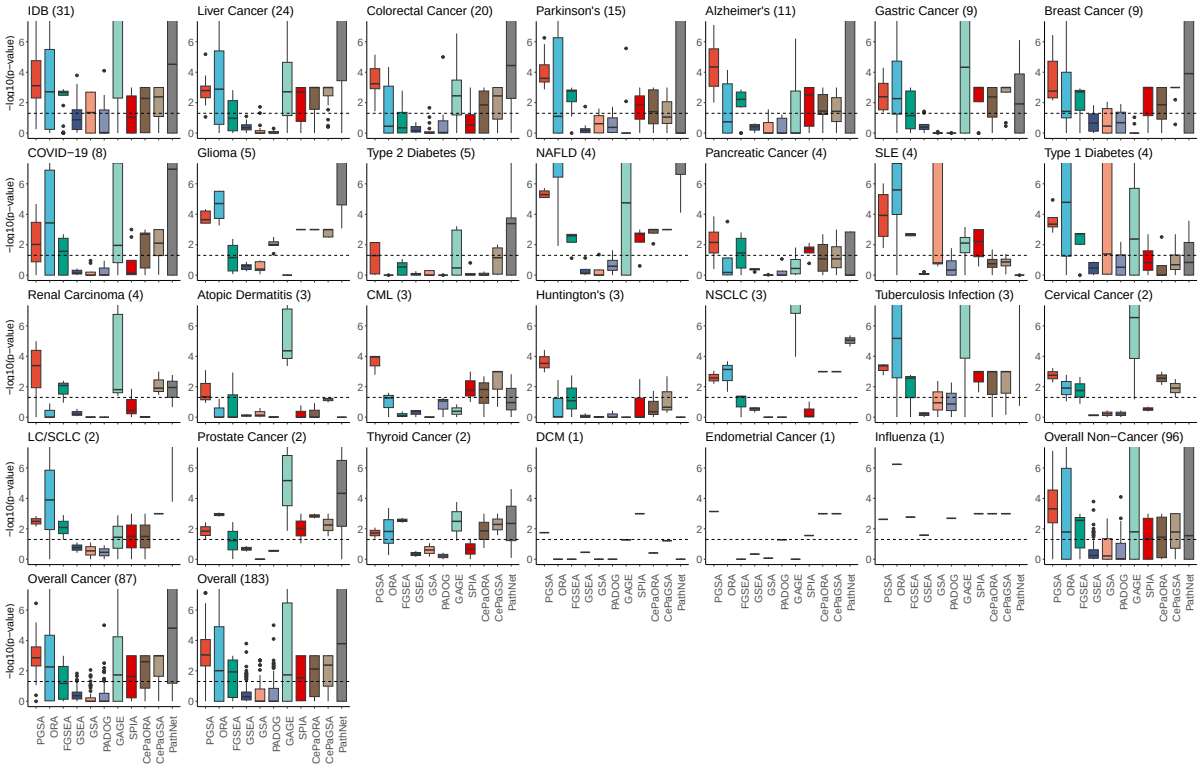

Figure 9.10: The HMP p-values of the targeted gene sets from the KEGG database for the 11 methods using counts data of the RNA-seq datasets from the GEO database. Each subfigure shows the p-values of the gene sets for a condition. The title of each subfigure shows the name of the condition and the number of datasets used for the condition. The x-axis represents the methods and the y-axis represents the p-values of the gene sets. The horizontal dashed line represents the significance threshold of 0.05. The last 3 subfigures show the overall performance of the methods for only non-cancer datasets, only cancer datasets, and all datasets, respectively.

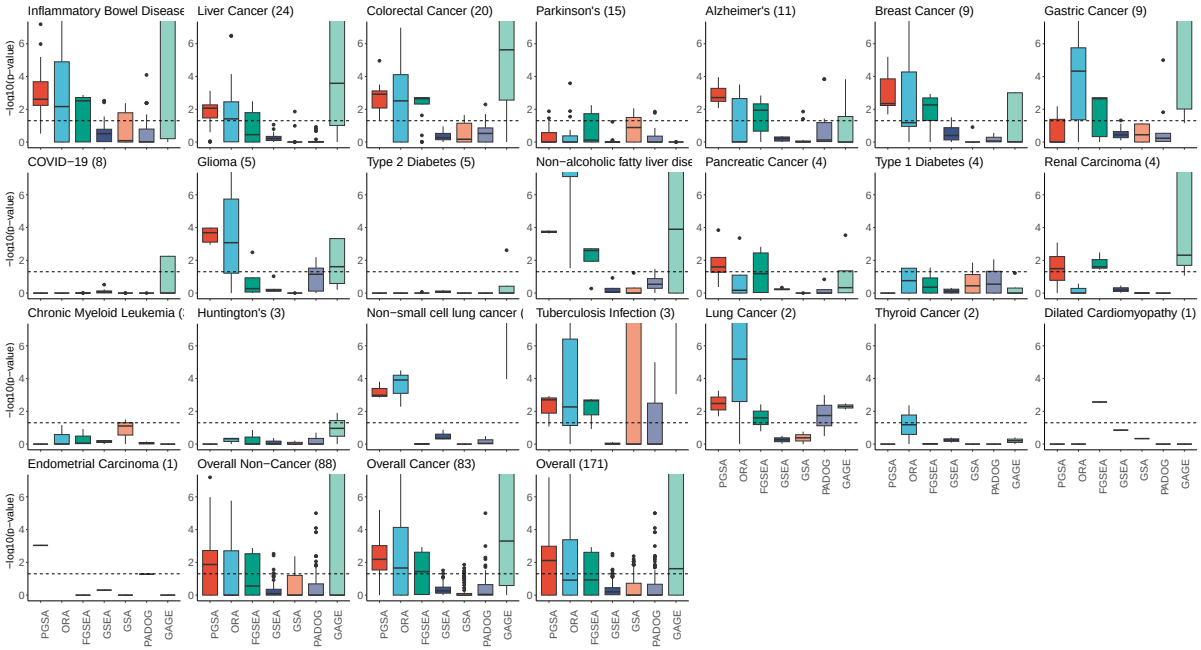

Figure 9.11: The HMP p-values of the targeted gene set from the WikiPathways database for 7 methods using counts data of the RNA-seq datasets from the GEO database.

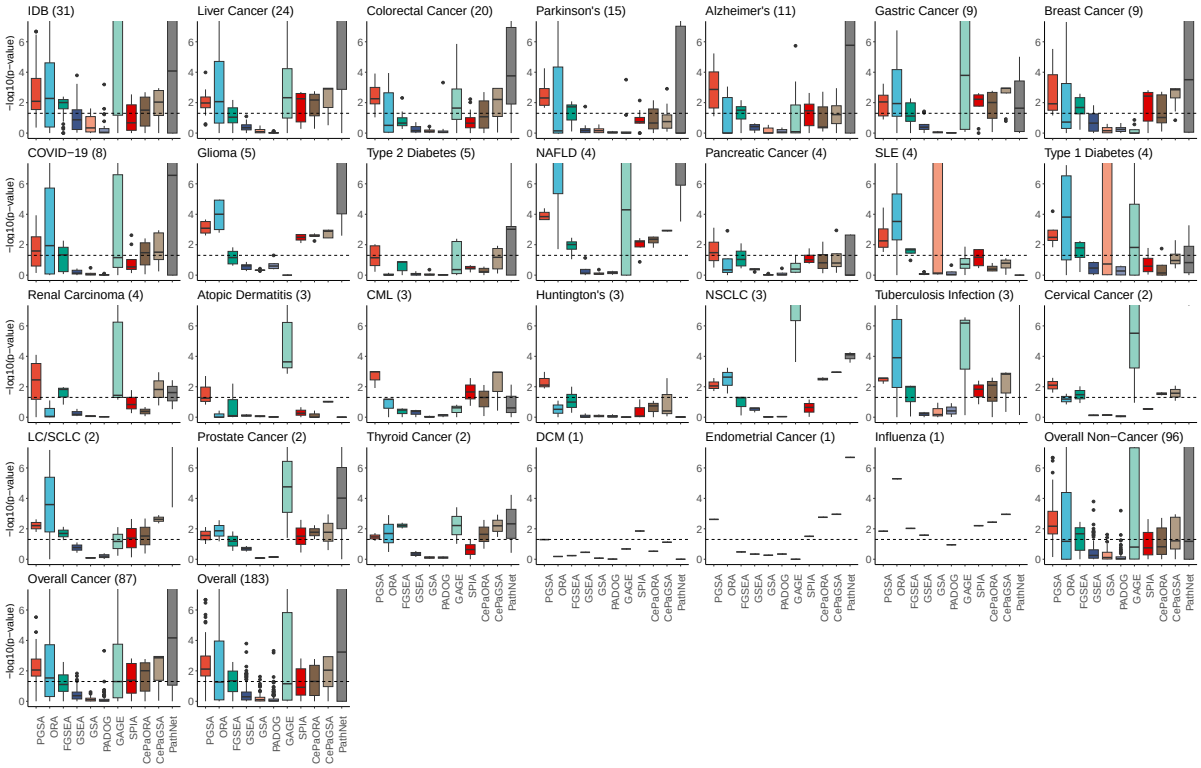

Figure 9.12: The FDR adjusted p-values of the targeted gene sets from the KEGG database for the 11 methods using counts data of the RNA-seq datasets from the GEO database. Each subfigure shows the p-values of the gene sets for a condition. The title of each subfigure shows the name of the condition and the number of datasets used for the condition. The x-axis represents the methods and the y-axis represents the p-values of the gene sets. The horizontal dashed line represents the significance threshold of 0.05. The last 3 subfigures show the overall performance of the methods for only non-cancer datasets, only cancer datasets, and all datasets, respectively.

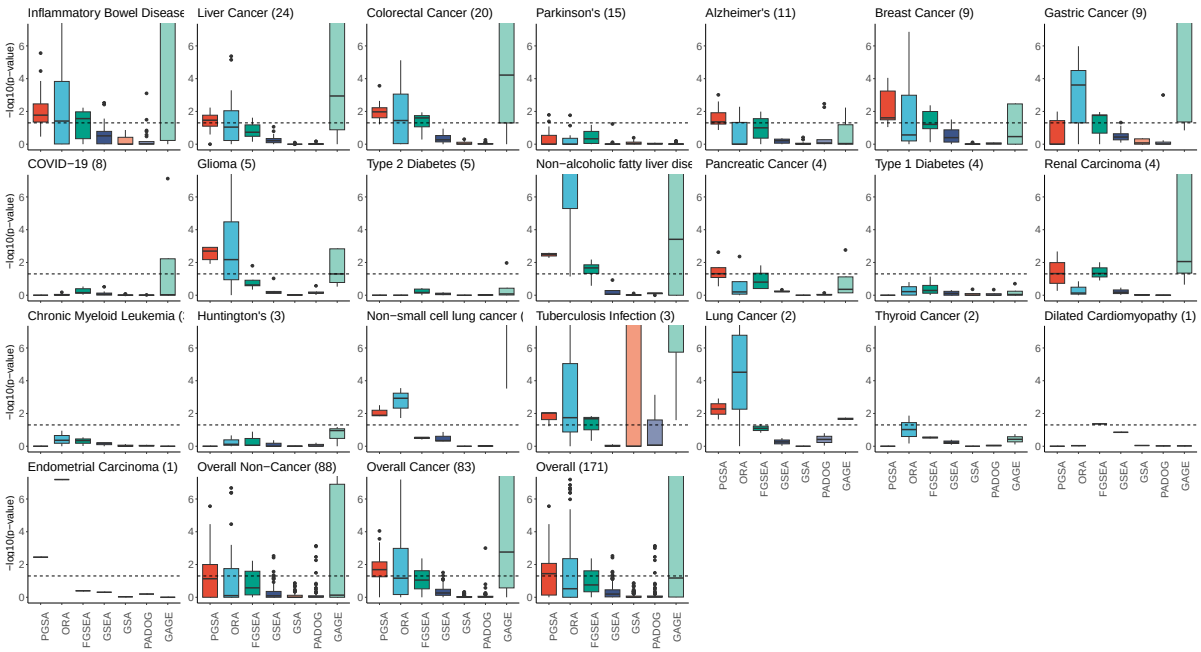

Figure 9.13: The FDR adjusted p-values of the targeted gene sets from the WikiPathways database for 7 methods using counts data of the RNA-seq datasets from the GEO database.

Performance on TPM-normalized RNA-seq datasets downloaded from the GDC portal.

Similar to the GEO datasets, we run all 11 methods on the KEGG database and only run the non-topology-based methods on the WikiPathways database using the cancer datasets downloaded from the GDC portal. The results, including the scaled ranks of the targeted gene sets and the HMP p-values of the targeted gene sets, are presented in Figure 9.14. For results using FDR-adjusted p-values, see Figure 9.15. We group the datasets by the project they belong to and present the results for each cohort in a subfigure. In terms of ranking, using KEGG database (Figure 9.14a), PGSA achieves the highest median rank in 32 (out of 50) analyses, with a median rank of 0.96. The second best method is PADOG, with a median rank of 0.88, and is the best method in 2 analyses. Using WikiPathways database (Figure 9.14b), PGSA achieves the highest median rank in 24 (out of 39) analyses, with a similar median rank of 0.96. The second best method is PADOG with a median rank of 0.90, and is the best method in 3 analyses. We note that all datasets are RNA-Seq datasets of cancer studies. While PGSA shows a consistent ranking performance across different cohorts, with median ranks ranging from 0.93 to 0.97, some methods show a large variation in ranking performance across different cohorts. For example, GSEA has a median rank of 0.72 in the TCGA project, but has a median rank of 0.24 in the CPTAC project. Those numbers for FGSEA and GSA are 0.71 and 0.41, and 0.89 and 0.62, respectively.

In terms of p-values, PGSA identifies the targeted gene sets as significant in most analyses, regardless of the gene set database. The second best method in terms of ranking for both KEGG and WikiPathways databases, PADOG, however, fails to identify the targeted gene sets as significant in most analyses, for both gene set databases. When using a cutoff of 0.05 for the HMP p-values and a maximum of 10% for the top percentage of gene sets ranked by their significance, for PGSA the targeted gene sets are in the list of significant gene sets for further analysis in 76% of the analyses using the KEGG database and 72% of the analyses using the WikiPathways database (Table 9.7). Those numbers for PADOG are 42% for KEGG gene sets, and only 5% for WikiPathways gene sets.

In summary, PGSA consistently outperforms other methods in ranking the targeted gene sets as more significant than other gene sets. When using both p-values and the top percentage

Table 9.7: The percentage of analyses using GDC datasets in which the targeted gene sets are identified as significant by each method using a significance threshold of 0.05 (HMP adjusted). The first column shows the gene set database used in the analysis. The second column shows the top percentage of gene sets used as the cutoff for significance. The third to the last columns show the percentage of analyses in which the targeted gene sets are identified as significant by each method. All percentages are rounded to the nearest integer. The total number of analyses is 50 for KEGG gene sets and 39 for WikiPathway gene sets.

Database	Threshold	Method										
		PGSA	ORA	FGSEA	GSEA	GSA	PADOG	GAGE	SPIA	CePaORA	CePaGSA	PathNet
KEGG	10	76	22	8	0	36	42	8	34	34	2	22
	20	84	34	20	2	36	42	10	52	44	8	40
	30	92	52	44	4	36	42	24	56	68	14	50
WikiPathways	10	72	26	13	8	38	5					
	20	74	46	26	8	38	23					
	30	79	46	51	8	38	41					

of gene sets as the cutoff for significance, PGSA demonstrates an even bigger margin in performance when identifying the targeted gene sets as one of the potential gene sets for further analysis.

Performance on counts data of RNA-seq datasets downloaded from the GDC portal.

Figures 9.16 and 9.17 show the scaled ranks and p-values of the targeted gene sets for each method using counts data of the RNA-seq datasets from the GDC portal. Similar to the results from the TPM-normalized data, PGSA consistently outperforms other methods when using the counts data.

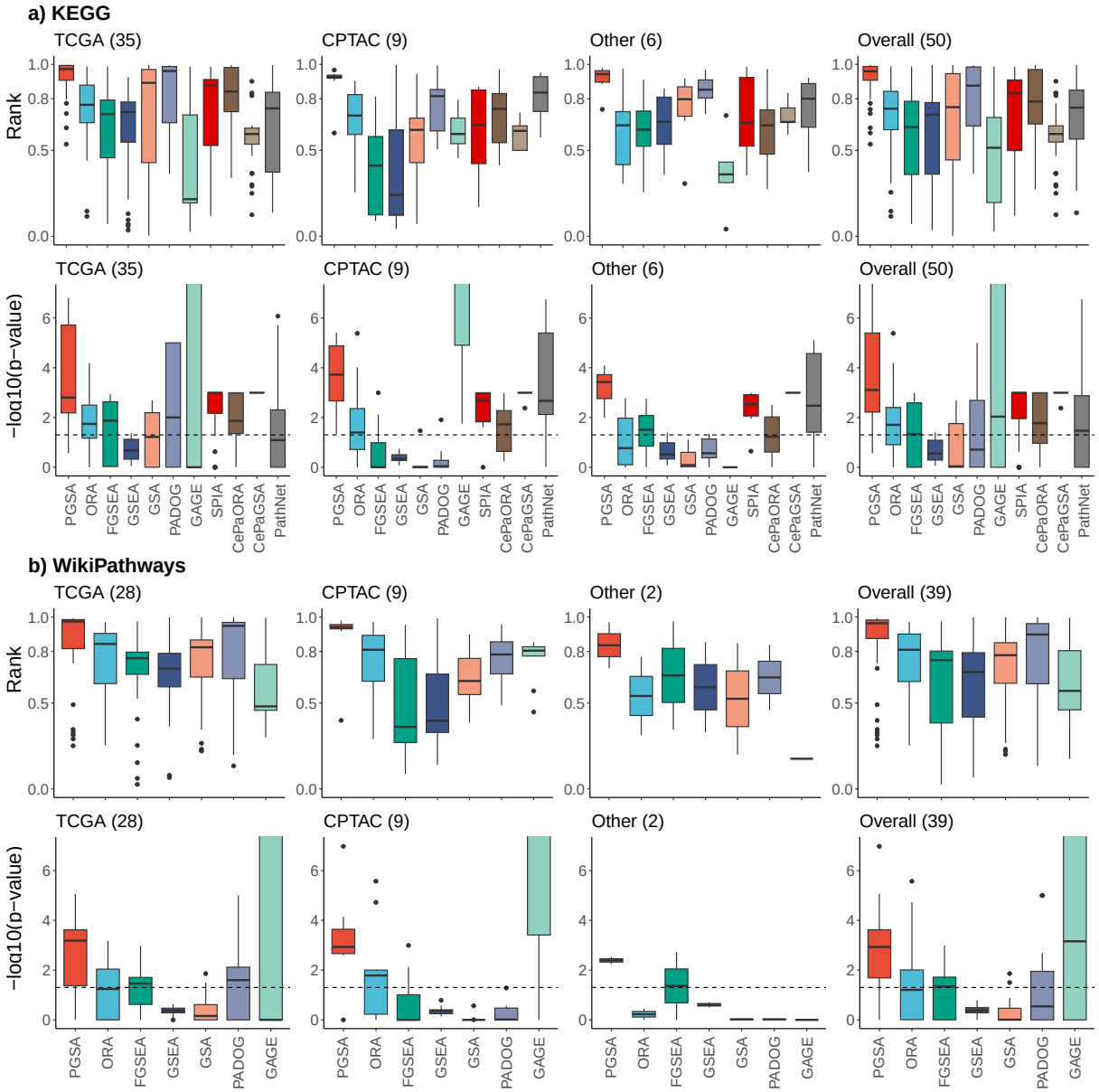

Figure 9.14: The scaled ranks and HMP p-values of the targeted gene sets using the TPM-normalized RNA-Seq datasets from the GDC portal. a) The scaled ranks (top) and p-values (bottom) of the targeted gene sets from the KEGG database for the 11 methods. b) The scaled ranks (top) and p-values (bottom) of the targeted gene sets from the WikiPathways database for 7 methods. In each subfigure, the title shows the name of the cohort and the number of datasets used from the cohort. The x-axis represents the methods and the y-axis represents the scaled ranks or p-values of the gene sets. The horizontal dashed line represents the significance threshold of 0.05.

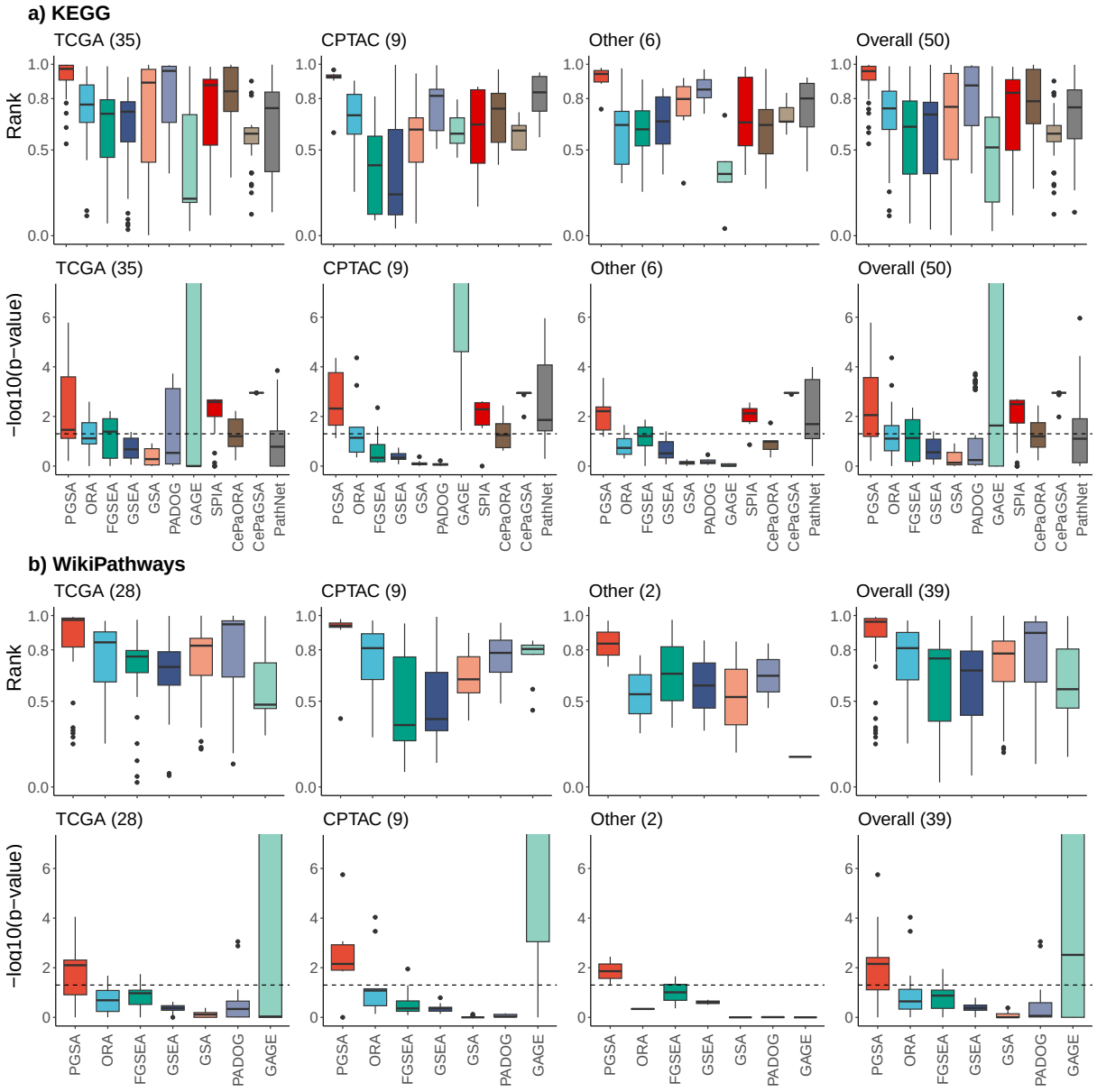

Figure 9.15: The scaled ranks and FDR-adjusted p-values of the targeted gene sets using the TPM-normalized RNA-seq datasets from the GDC portal. a) The scaled ranks (top) and p-values (bottom) of the targeted gene sets from the KEGG database for the 11 methods. b) The scaled ranks (top) and p-values (bottom) of the targeted gene sets from the WikiPathways database for 7 methods. In each subfigure, the title shows the name of the cohort and the number of datasets used from the cohort. The x-axis represents the methods and the y-axis represents the scaled ranks or p-values of the gene sets. The horizontal dashed line represents the significance threshold of 0.05.

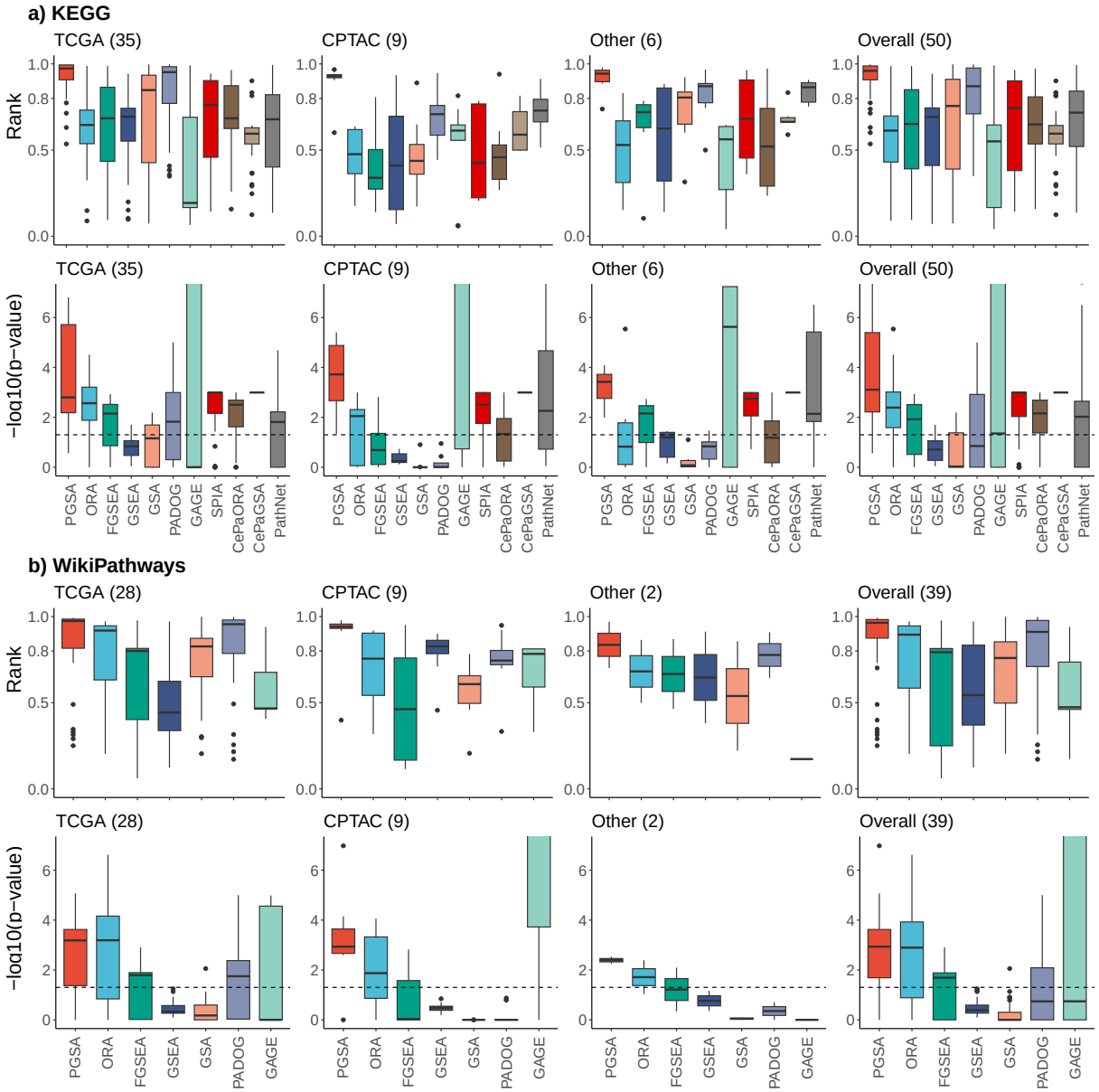

Figure 9.16: The scaled ranks and HMP p-values of the targeted gene sets using the counts data of RNA-seq datasets from the GDC portal. a) The scaled ranks (top) and p-values (bottom) of the targeted gene sets from the KEGG database for the 11 methods. b) The scaled ranks (top) and p-values (bottom) of the targeted gene sets from the WikiPathways database for 7 methods. In each subfigure, the title shows the name of the cohort and the number of datasets used from the cohort. The x-axis represents the methods and the y-axis represents the scaled ranks or p-values of the gene sets. The horizontal dashed line represents the significance threshold of 0.05.

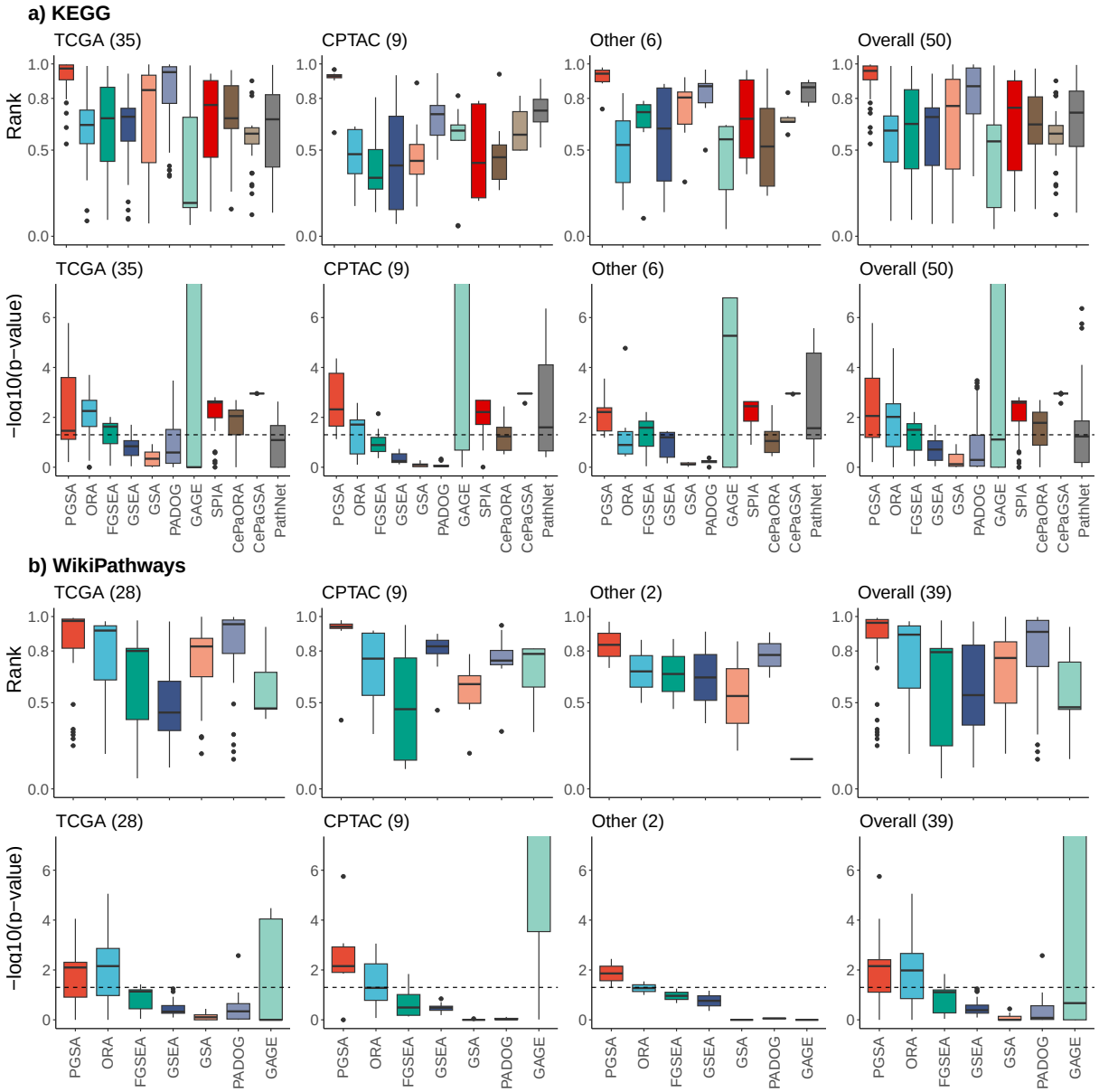

Figure 9.17: The scaled ranks and FDR-adjusted p-values of the targeted gene sets using the counts data of RNA-seq datasets from the GDC portal. a) The scaled ranks (top) and p-values (bottom) of the targeted gene sets from the KEGG database for the 11 methods. b) The scaled ranks (top) and p-values (bottom) of the targeted gene sets from the WikiPathways database for 7 methods. In each subfigure, the title shows the name of the cohort and the number of datasets used from the cohort. The x-axis represents the methods and the y-axis represents the scaled ranks or p-values of the gene sets. The horizontal dashed line represents the significance threshold of 0.05.

9.3.2 PGSA improves ranking of disease-related gene sets

In the previous analysis, we assessed the performance of PGSA and other methods in ranking the significance of a single gene set that is known to be associated with the disease of interest. This analysis solely relies on the availability of the targeted gene sets in the gene set database, and disregards other gene sets that might be relevant to the disease of interest. To extend the assessment to other gene sets that might be relevant to the disease of interest, in this analysis, we compute a disease-relevant score for each pair of gene sets and conditions. We then assess the performance of the methods in ranking disease-related gene sets. A good pathway analysis method should be able to identify gene sets that have a high disease-relevant score as more significant than gene sets that have a low disease-relevant score. For each gene set, we compute the disease-relevant score as

$$\text{drs} = \frac{1}{n} \sum_{i=1}^{n} \text{gda}_i \times \text{dsi}_i \tag{9.3}$$

where n is the number of genes in the gene set, gda_i is the disease association score of the i-th gene, and dsi_i is the disease specificity score of the i-th gene. The disease association score reflects the abundance of evidence supporting the association between the gene and the disease, while the disease specificity score reflects the specificity of the association between the gene and the disease, i.e, a gene that is associated with only one disease has a higher disease specificity score for that disease than a gene that is associated with multiple diseases. We use the disease association scores and the disease specificity scores from the DisGeNET database [241] as the source of the scores.

Note that the gene-disease association scores and the disease specificity scores are based on the amount of evidence supporting the association between the gene and the disease in the literature [241]. This score is by no means representing the causal relationship between the gene and the disease. As a result, our disease-relevant score represents the average amount of evidence supporting the association between the genes in the gene set and the disease of interest. The higher the disease-relevant score of a gene set, the more likely the gene set is relevant to the disease of interest.

In this analysis, we use three gene set databases, KEGG, WikiPathway, and Gene Ontology (GO) Biological Process, as the sources of gene sets. The number of gene sets analyzed from each database is 371, 798, and 7,096, respectively. Similar to the previous analysis, we run all 11 methods on the KEGG database. For the WikiPathways database and the GO database, we only run the non-topology-based methods. For each analysis result, we compute the Spearman correlation coefficient between the negative logarithm of the p-values and the disease-relevant scores of the gene sets. We remove extreme p-values (p-value is 0 or 1) from the analysis as methods that rely on permutation tests might have a large number of gene sets with p-value of 0 or 1 and will cause the correlation to be biased.

Performance on microarray datasets and TPM-normalized RNA-Seq datasets downloaded from the GEO database.

Figure 9.18 shows the overall correlation between the negative logarithm of the p-values and the disease-relevant scores for each method and each gene set database using microarray datasets and RNA-Seq datasets from the GEO database. We group the datasets by the disease type (cancer or non-cancer), the technology used to generate the data (microarray or RNA-Seq), and overall. The details of the correlation for each condition and each gene set database are presented in Figures 9.23, 9.24, and 9.25. Overall, PGSA outperforms other methods in ranking more relevant gene sets as more significant. For KEGG gene sets, PGSA achieves the highest median correlation in 28 (out of 29) conditions, and 332 (out of 371) analyses, with a median correlation of 0.54. For WikiPathways and GO Biological Process gene sets, PGSA achieves the highest median correlation in all conditions, and in 355 and 354 analyses, respectively. The median correlation for PGSA is 0.41 and 0.37 for WikiPathways and GO Biological Process gene sets, respectively. The median correlation for any of the three gene set databases is significantly higher than the median correlation of the other methods (p-value $< \times 10^{-16}$ in all cases using the one-sided Wilcoxon signed-rank test) with a large margin. For example, the median correlation of the second-best method with the KEGG database (PathNet) is 0.30 (compared to 0.54 for PGSA), with the WikiPathways (PADOG) database is 0.11 (compared to 0.41 for

PGSA), and with the GO Biological Process (PADOG) database is 0.12 (compared to 0.37 for PGSA).

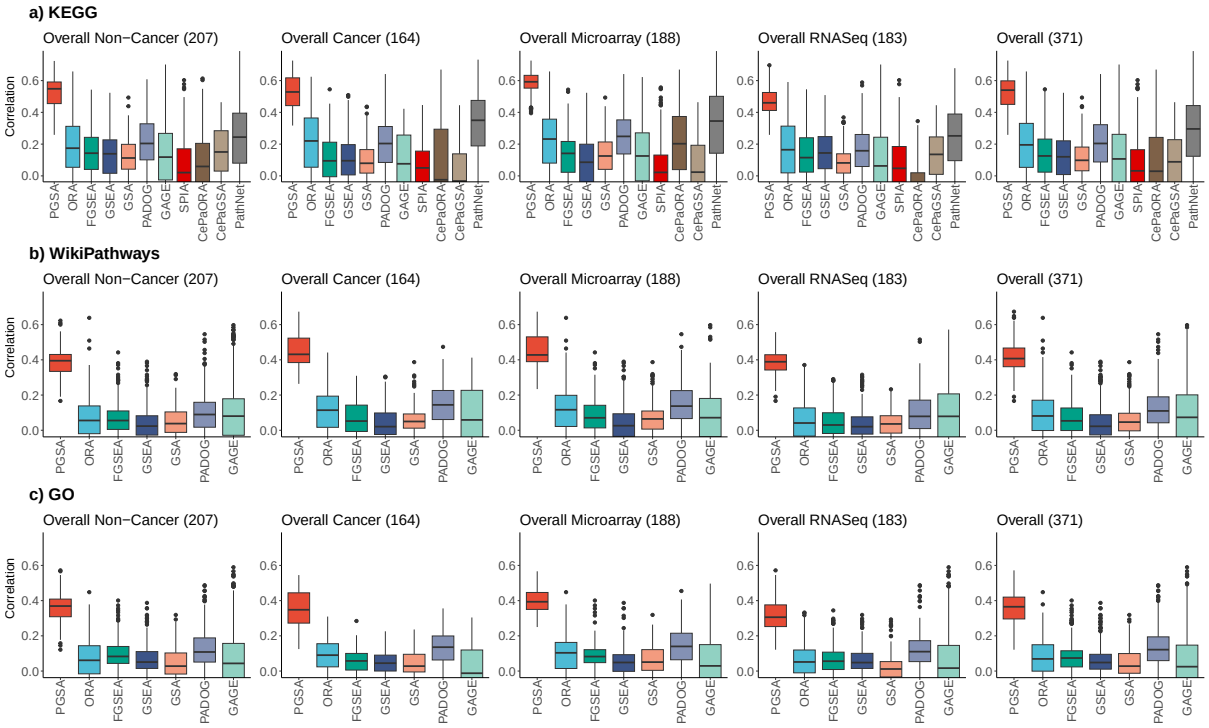

Figure 9.18: The correlation of the disease-relevant scores and the p-values of the gene sets for datasets from the GEO database (microarray and TPM-normalized RNA-Seq) from the 11 methods and 3 gene set databases: a) KEGG, b) WikiPathways, and c) GO Biological Process. In each subfigure, the title shows the grouping of the datasets based on the disease types (cancer or non-cancer), the technology used to generate the data (microarray or RNA-Seq), and overall, along with the number of datasets used for the grouping. The x-axis represents the methods and the y-axis represents the correlation of the disease-relevant scores and the p-values of the gene sets.

We also present the disease-relevant scores of the top 10% most significant gene sets for each method and each gene set database in Figure 9.19. Since the disease-relevant scores are not directly comparable across different conditions and gene set databases, we rank the disease-relevant scores of the gene sets per condition and gene set database. The ranks are from 1 to the number of gene sets in the database, with the lowest score having a rank of 1 and the highest score having a rank of the number of gene sets in the database. We then scale the ranks to the range of 0 to 1, with the highest score having a rank of 1 and the lowest score having a rank of 0. From this point, we refer to this scaled ranks as the disease-relevant ranks. Figure 9.19 shows the disease-relevant scores grouped by the disease type (cancer or non-cancer), the technology used to generate the data (microarray or RNA-Seq), and overall.

We also present the results for each condition and each gene set database in Figures 9.20, 9.21, and 9.22. In this analysis, an ideal method will have a median rank of 0.95 and a random result will have a median rank of 0.5. Overall, the top 10% most significant gene sets identified by PGSA are more relevant to the disease of interest than those identified by other methods. For KEGG gene sets, PGSA achieves the highest median disease-relevant rank in 22 (out of 29) conditions. Among 371 analyses, PGSA achieves the highest median disease-relevant rank in 190 analyses, with a median score of 0.76. In other words, in most analyses, the top 10% most significant gene sets identified by PGSA are those among the top 24% most relevant gene sets in the KEGG database. For WikiPathways gene sets, PGSA achieves the highest median disease-relevant score in 27 conditions, and in 303 analyses, with a median score of 0.70. For GO Biological Process gene sets, PGSA also achieves the highest median disease-relevant score in 28 conditions and in 320 analyses, with a median score of 0.67. These median scores for any of the three gene set databases are significantly higher than the median scores of the other methods (p-value $< \times 10^{-16}$ in all cases using the one-sided Wilcoxon signed-rank test) with a substantial margin. For example, the second-best method with the KEGG database (PathNet) has a median score of 0.68 (compared to 0.76 for PGSA), with the WikiPathways (GAGE) database has a median score of 0.57 (compared to 0.70 for PGSA), and the GO Biological Process (GAGE) database has a median score of 0.58 (compared to 0.67 for PGSA).

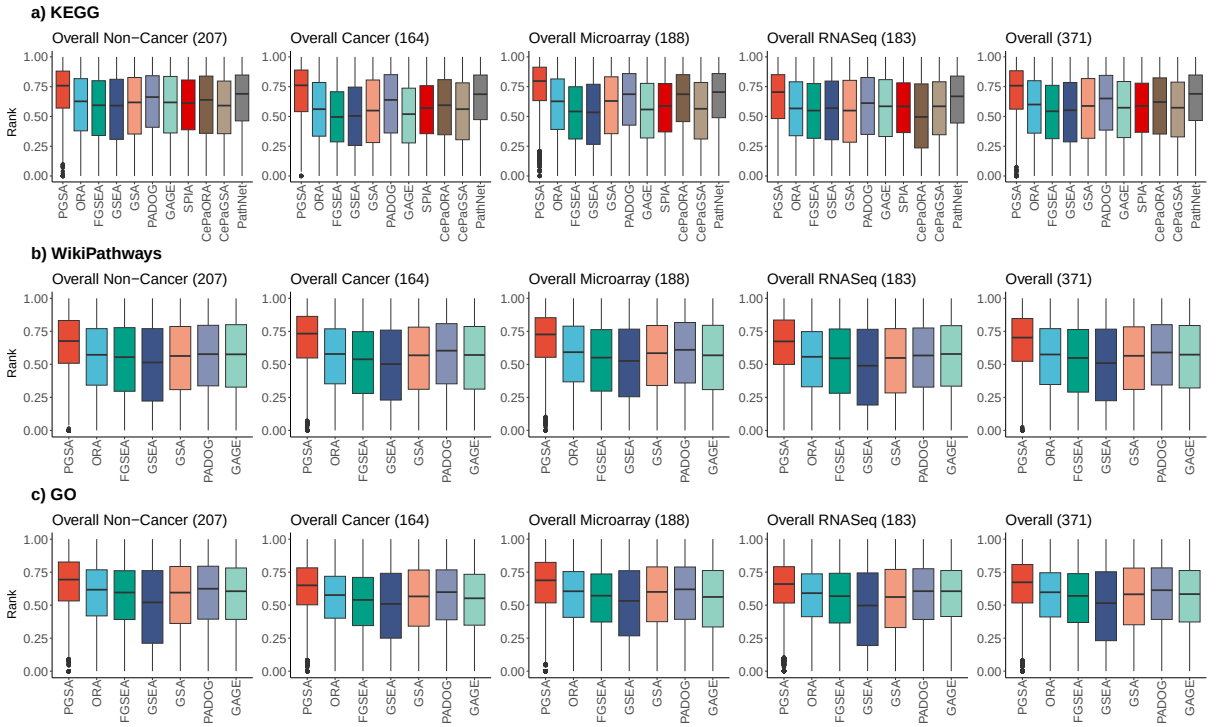

Figure 9.19: The disease-relevant rank of top 10% most significant gene sets for datasets from the GEO database (microarray and TPM-normalized RNA-Seq) for the 11 methods and 3 gene set databases: a) KEGG, b) WikiPathways, and c) GO Biological Process. In each subfigure, the title shows the group of the datasets based on the disease types (cancer or non-cancer), the technology used to generate the data (microarray or RNA-Seq), and overall, along with the number of datasets used for the grouping. The x-axis represents the methods and the y-axis represents the disease-relevant scores of the gene sets.

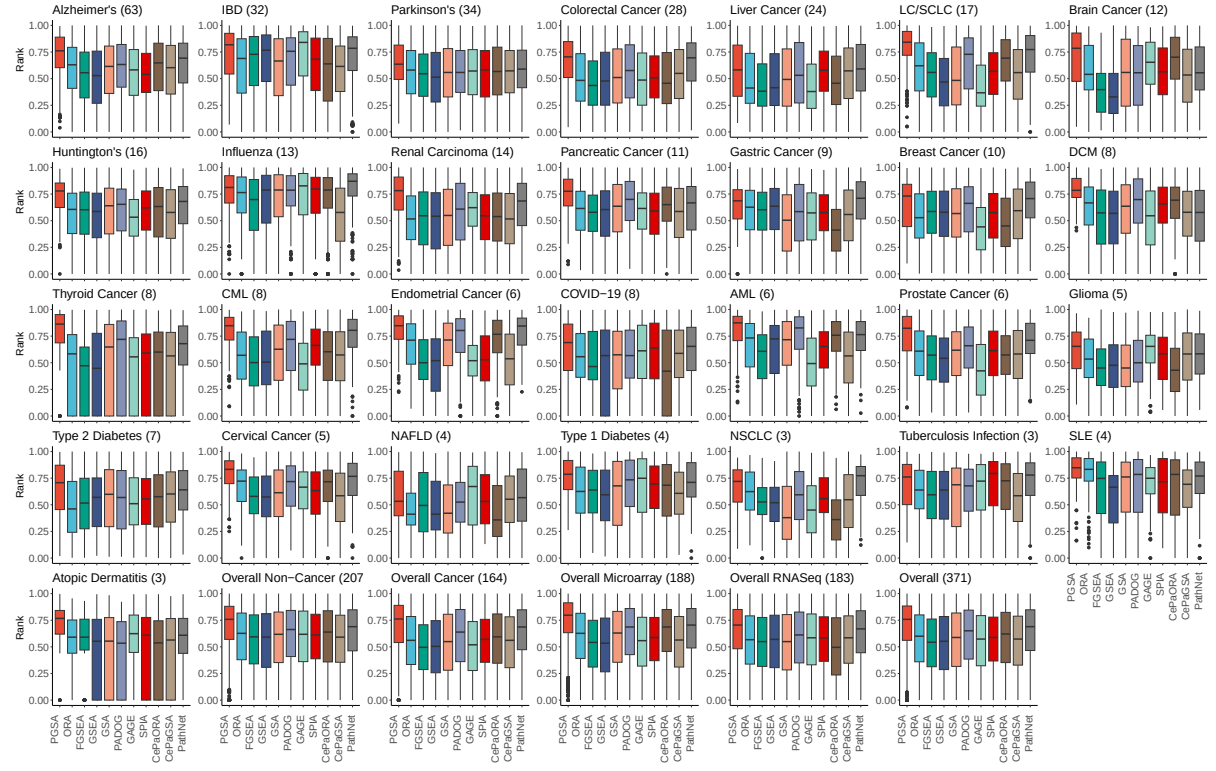

Figure 9.20: The disease-relevant scores of the top 20 gene sets from the KEGG database and GEO datasets (microarray and TPM-normalized RNA-Seq) for the 11 methods.

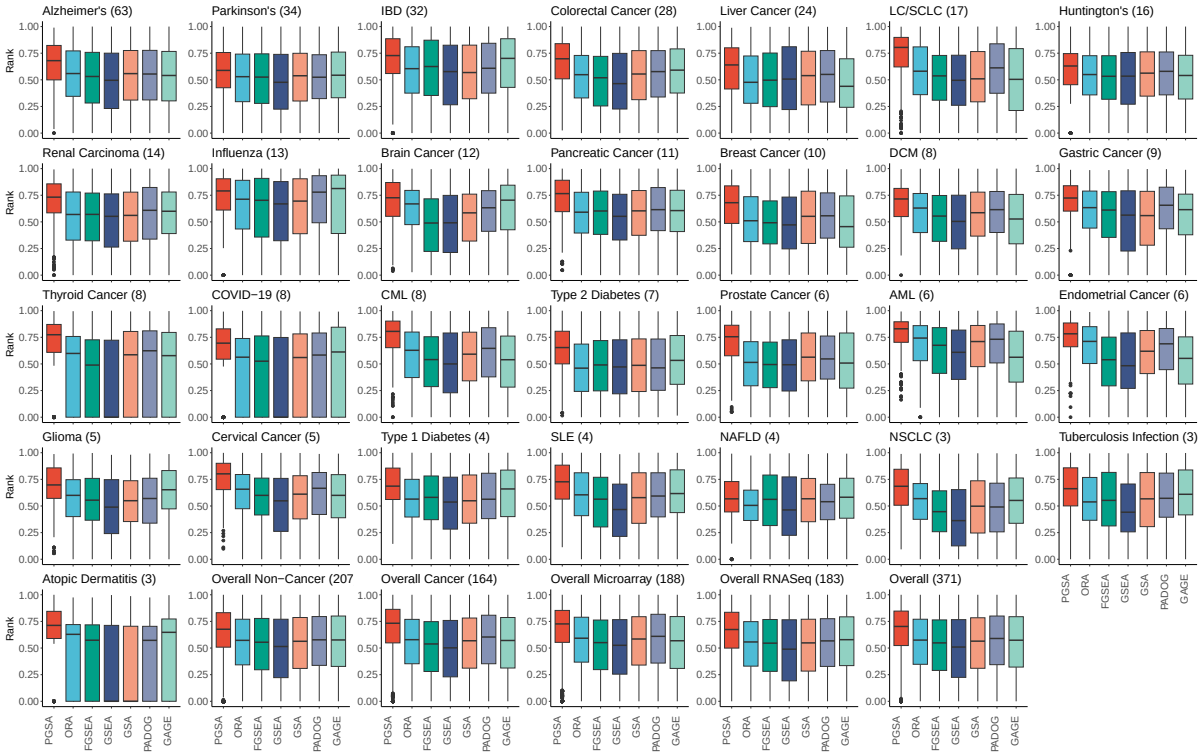

Figure 9.21: The disease-relevant scores of the top 20 gene sets from the WikiPathways database and GEO datasets (microarray and TPM-normalized RNA-Seq) for 7 methods.

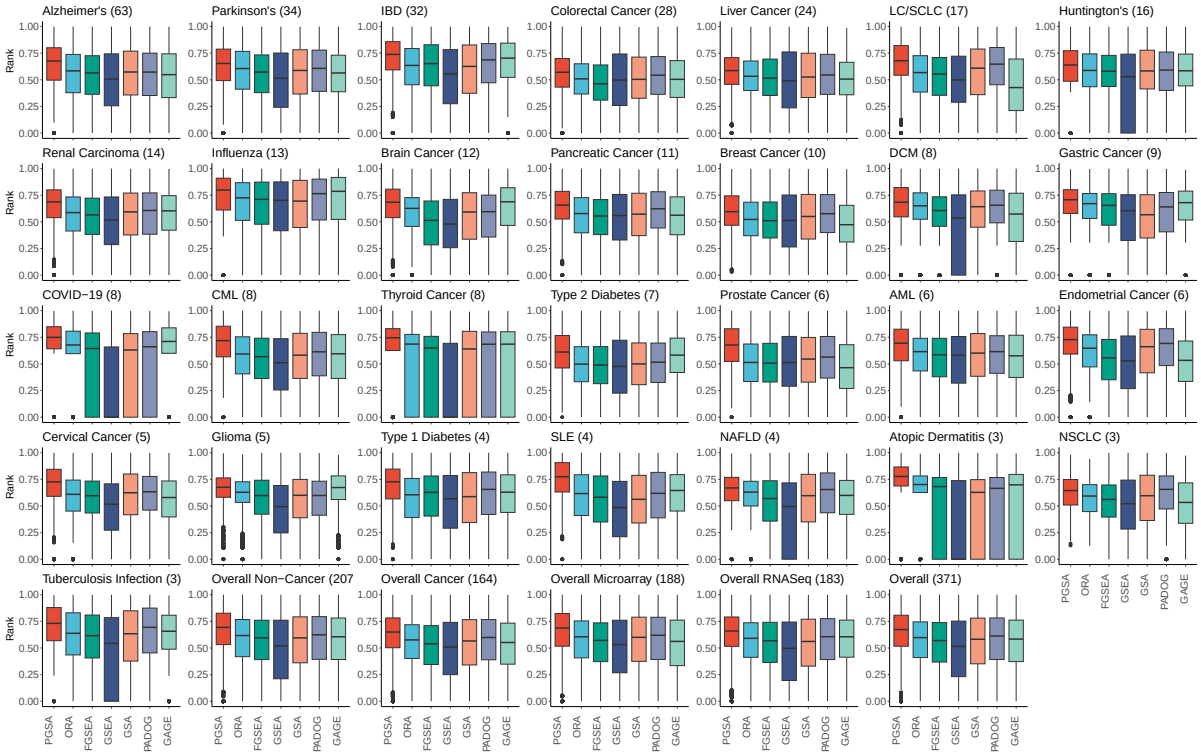

Figure 9.22: The disease-relevant scores of the top 20 gene sets from the GO database and GEO datasets (microarray and TPM-normalized RNA-Seq) for the 11 methods.

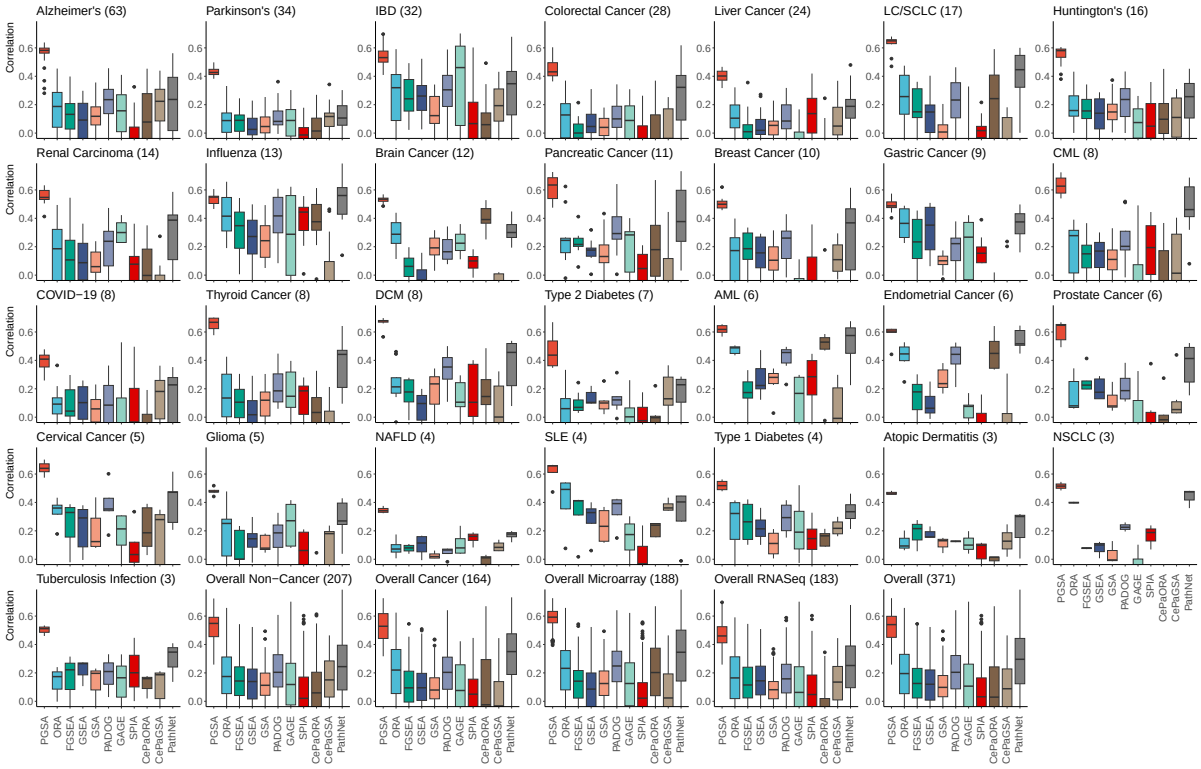

Figure 9.23: The correlation of the disease-relevant scores and the p-values of the gene sets from the KEGG database and GEO datasets (microarray and TPM-normalized RNA-Seq) for the 11 methods.

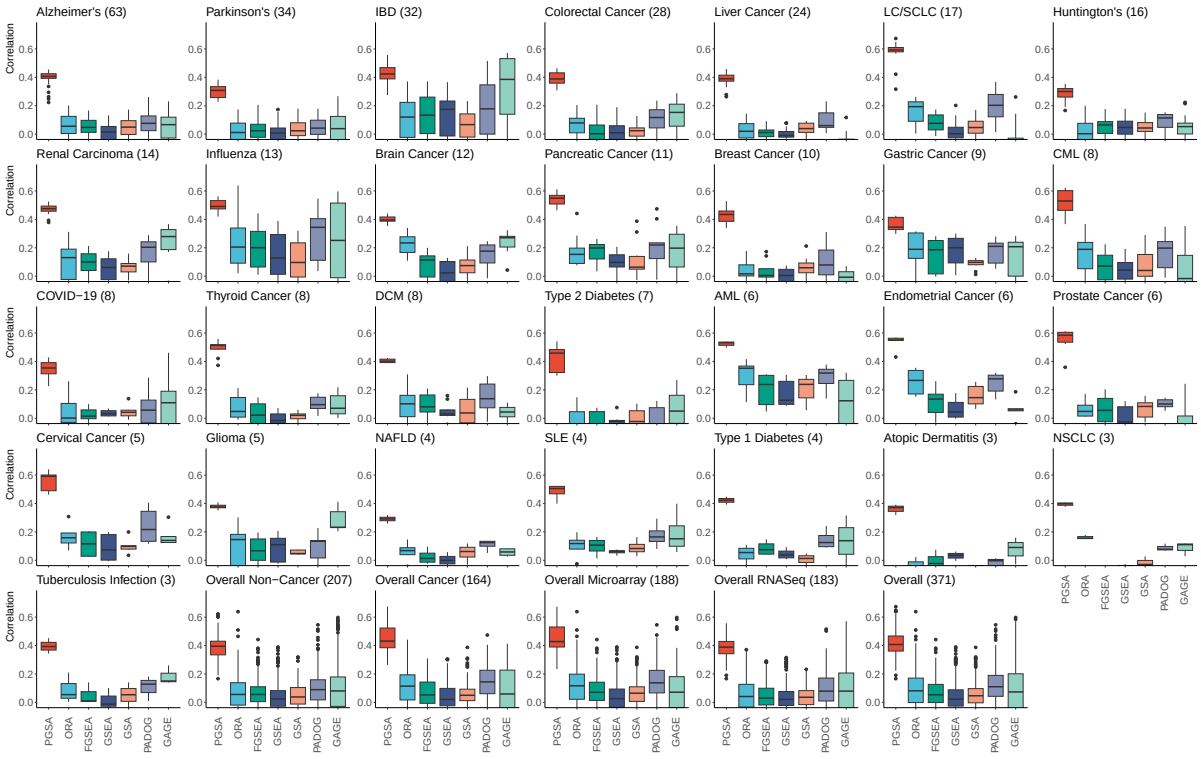

Figure 9.24: The correlation of the disease-relevant scores and the p-values of the gene sets from the WikiPathways database and GEO datasets (microarray and TPM-normalized RNA-Seq) for 7 methods.

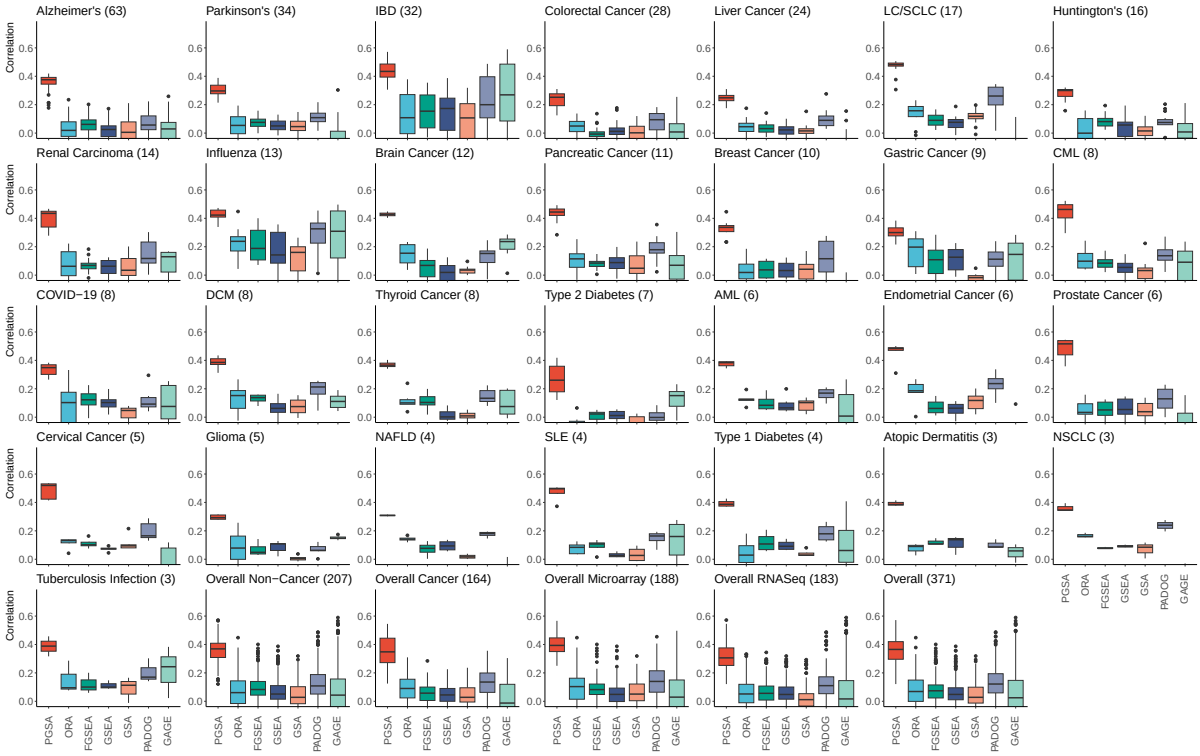

Figure 9.25: The correlation of the disease-relevant scores and the p-values of the gene sets from the GO database and GEO datasets (microarray and TPM-normalized RNA-Seq) for the 11 methods.

Performance on counts data of RNA-Seq datasets downloaded from the GEO database.

Figures 9.26, 9.27, 9.28, 9.29, 9.30, and 9.31 shows the disease-relevant scores and the correlation of the disease-relevant scores and the p-values of the gene sets for each method and each gene set database using counts data of the RNA-seq datasets from the GEO database. Overall, regardless of data normalization methods, PGSA consistently outperforms other methods in ranking the more disease-relevant gene sets as more significant than other gene sets.

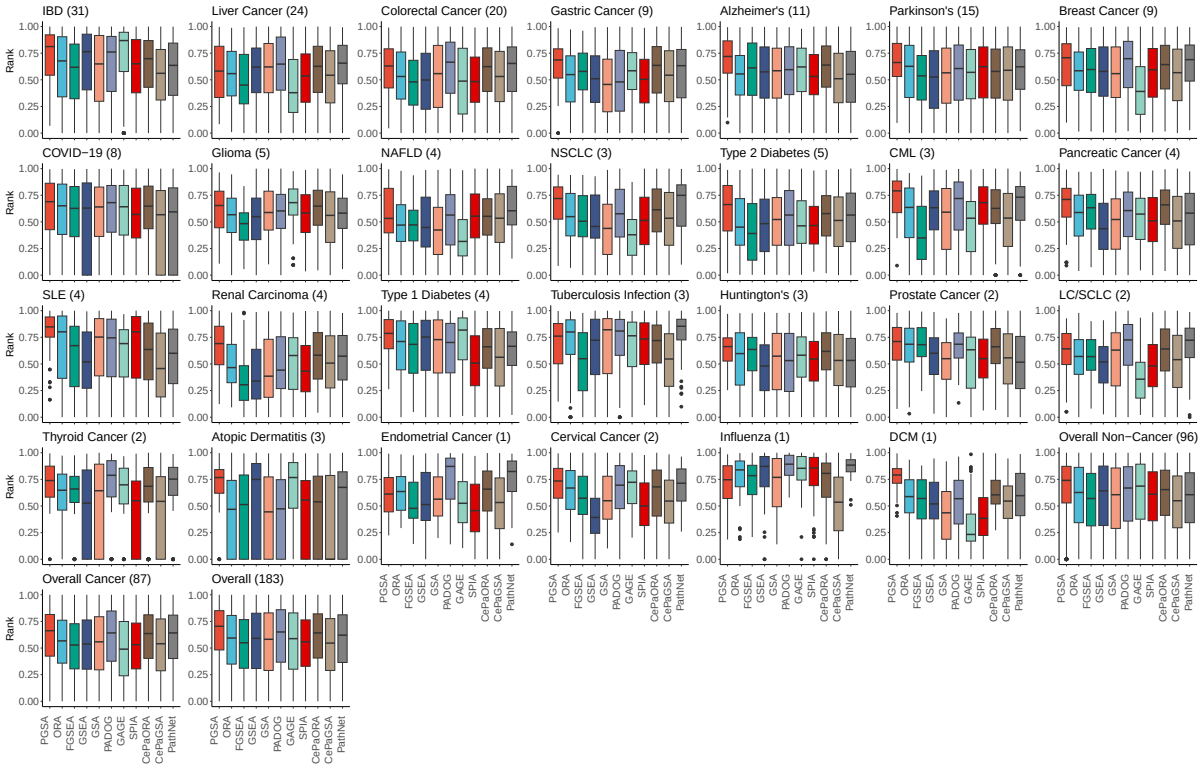

Figure 9.26: The disease-relevant scores of top 20 gene sets from the KEGG database and GEO RNA-Seq datasets using counts data for the 11 methods.

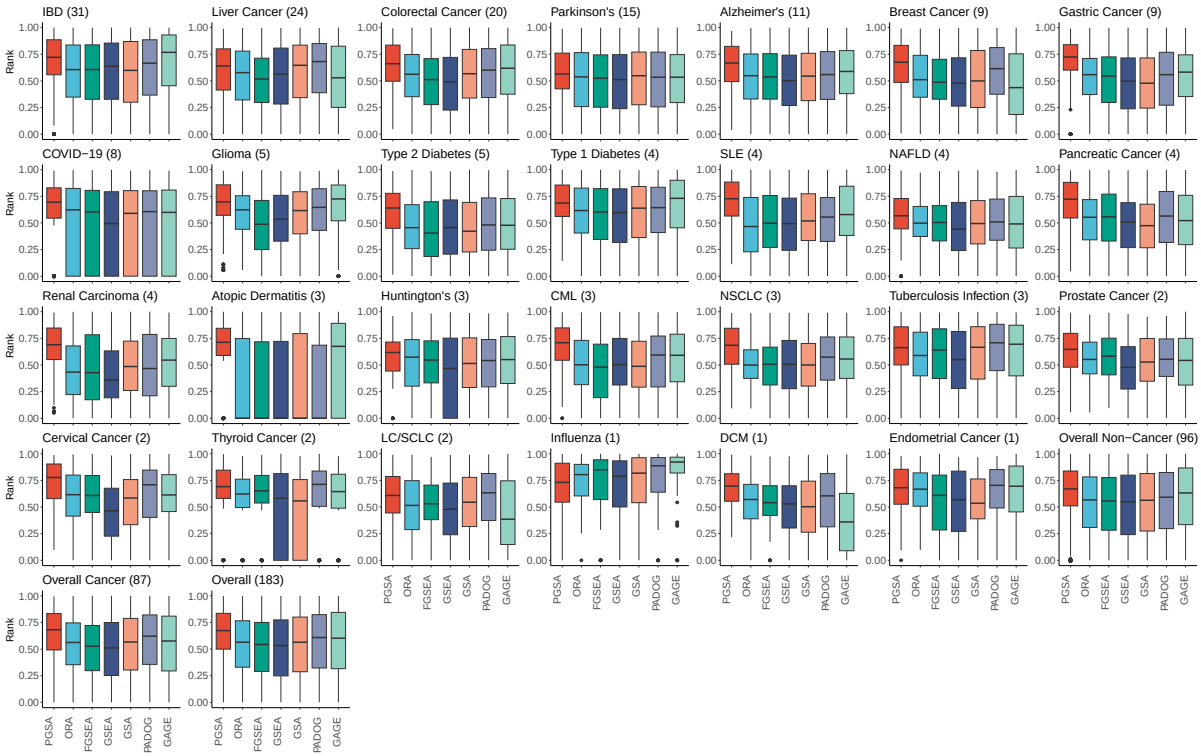

Figure 9.27: The disease-relevant scores of the top 20 gene sets from the WikiPathways database and GEO RNA-Seq datasets using counts data for 7 methods.

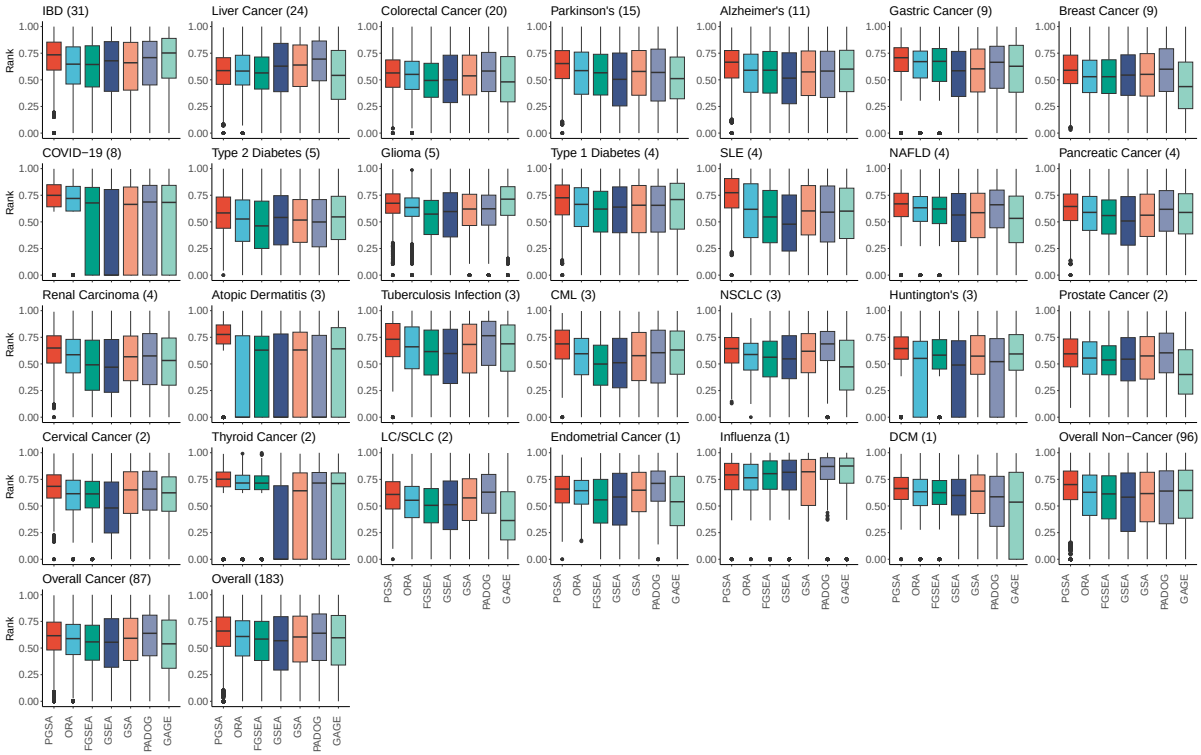

Figure 9.28: The disease-relevant scores of top 20 gene sets from the GO database and GEO RNA-Seq datasets using counts data for 7 methods.

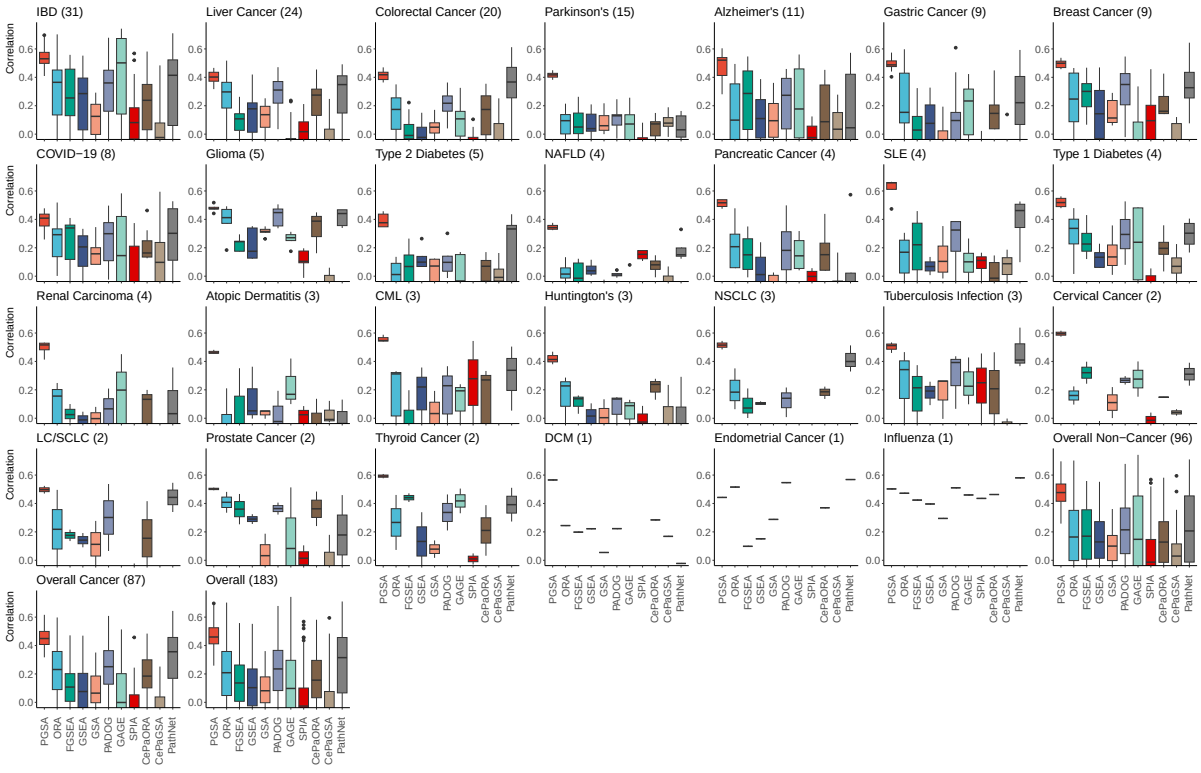

Figure 9.29: The correlation of the disease-relevant scores and the p-values of the gene sets from the KEGG database and GEO RNA-Seq datasets using counts data for the 11 methods.

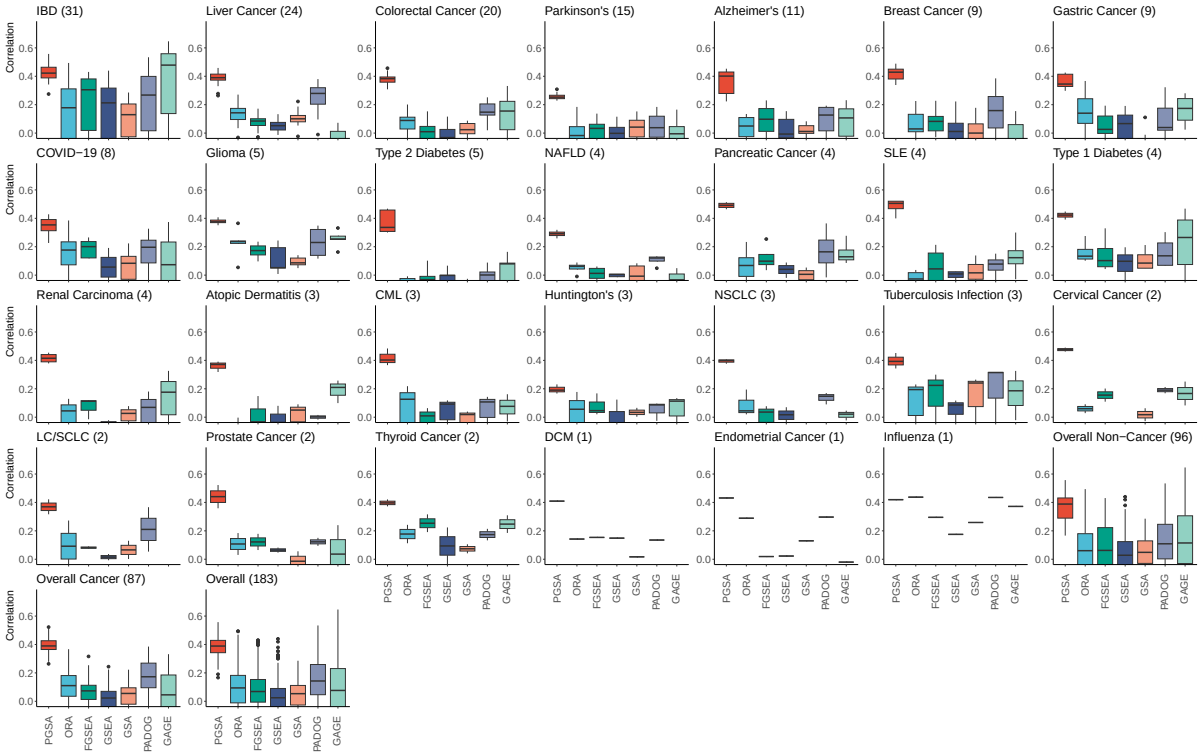

Figure 9.30: The correlation of the disease-relevant scores and the p-values of the gene sets from the WikiPathways database and GEO RNA-Seq datasets using counts data for 7 methods.

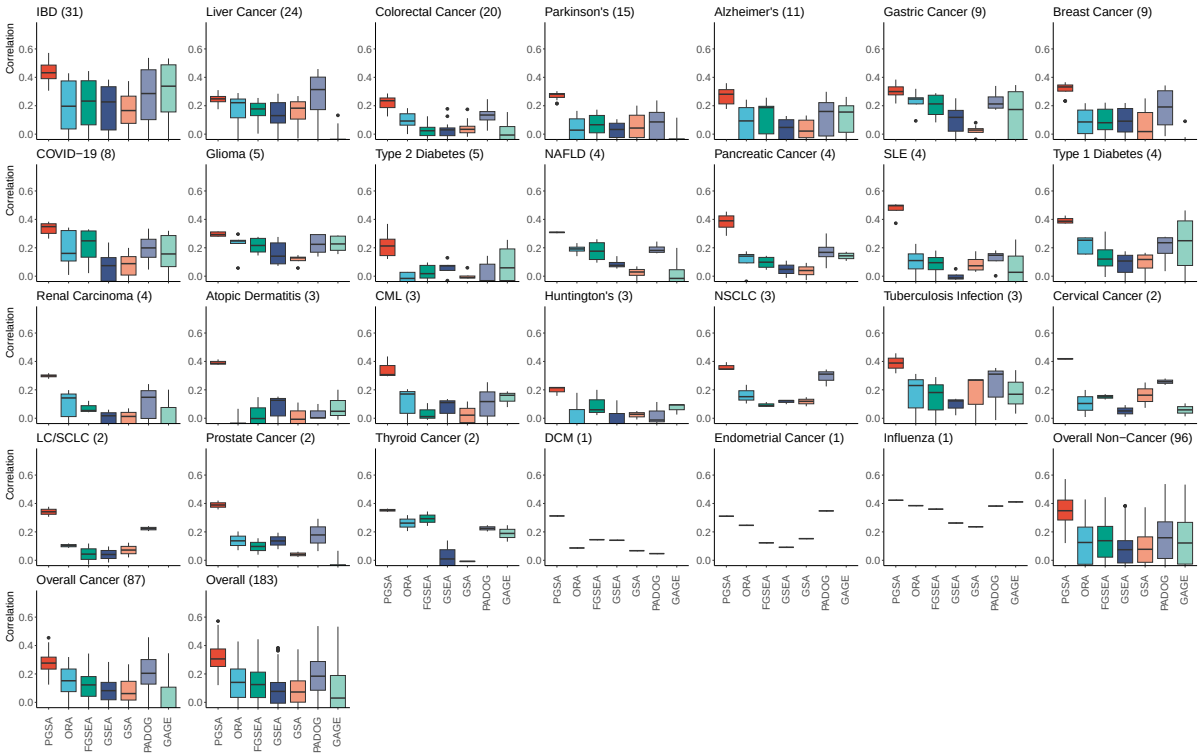

Figure 9.31: The correlation of the disease-relevant scores and the p-values of the gene sets from the GO database and GEO RNA-Seq datasets using counts data for 7 methods.

Performance on TPM-normalized RNA-seq datasets downloaded from the GDC portal.

Figure 9.32 shows the overall correlation between the negative logarithm of the p-values and the disease-relevant scores for each method and each gene set database using the cancer datasets from the GDC portal. Similar to the targeted gene set analysis, we group the datasets by the project they belong to and present the results for each cohort in a subfigure. Similar to the results from the GEO datasets, PGSA outperforms other methods in ranking gene sets that are more relevant to the disease of interest as more significant, with a large margin. For KEGG gene sets, PGSA achieves the highest median correlation in 48 (out of 50) analyses, with a median correlation of 0.56. The second best method is CePaORA, with a median correlation of 0.39. For WikiPathways gene sets, PGSA has the highest median correlation in all 50 analyses, with a median correlation of 0.46. The second best method is PADOG, with a median correlation of 0.22. Similarly, for GO Biological Process gene sets, PGSA achieves the highest median correlation in all analyses, with a median correlation of 0.40. The second best method is PADOG, with a median correlation of 0.23.

Figure 9.33 shows the disease-relevant ranks of the top 10% most significant gene sets for each method and each gene set database using the cancer datasets from the GDC portal. Similar to the results from the GEO datasets, the top 10% most significant gene sets identified by PGSA are more relevant to the disease of interest than those identified by other methods. For KEGG gene sets, PGSA achieves the highest median disease-relevant rank in 35 (out of 50) analyses, with a median score of 0.79. For WikiPathways gene sets, those numbers are 35 and 0.76, respectively, and for GO Biological Process gene sets, those numbers are 25 and 0.69, respectively. The median scores for any of the three gene set databases are significantly higher than the median scores of the other methods (p-value $< \times 10^{-16}$ in all cases using the one-sided Wilcoxon signed-rank test). The second best method for KEGG gene sets is CePaORA, with a median score of 0.69, and for WikiPathways and GO Biological Process gene sets, PADOG, with a median score of 0.64 and 0.65, respectively.

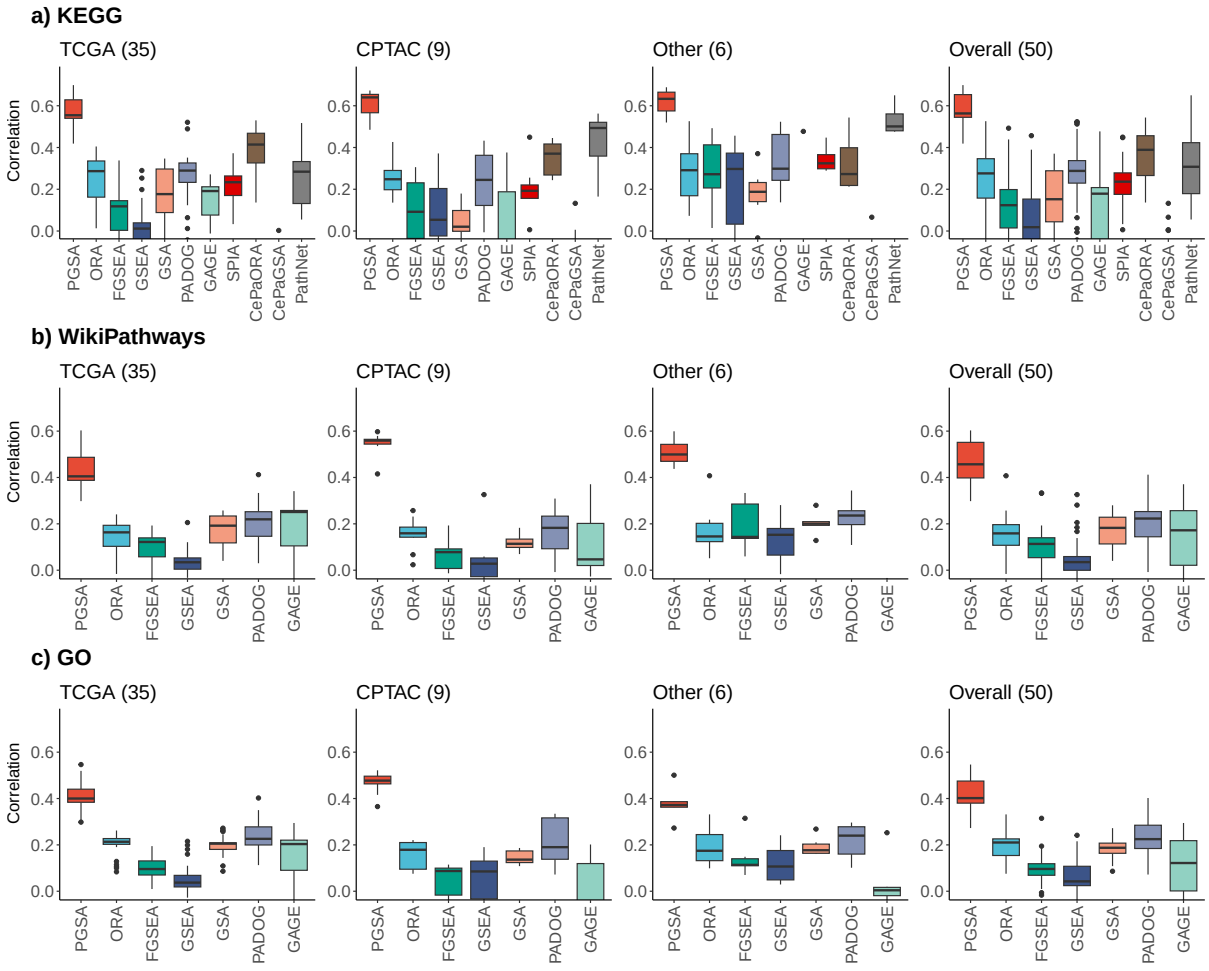

Figure 9.32: The correlation of the disease-relevant scores and the p-values of the gene sets for datasets from the GDC portal (TPM-normalized RNA-Seq) for the 11 methods and 3 gene set databases: a) KEGG, b) WikiPathways, and c) GO Biological Process. In each subfigure, the title shows the name of the cohort and the number of datasets used from the cohort. The x-axis represents the methods and the y-axis represents the disease-relevant scores of the gene sets.

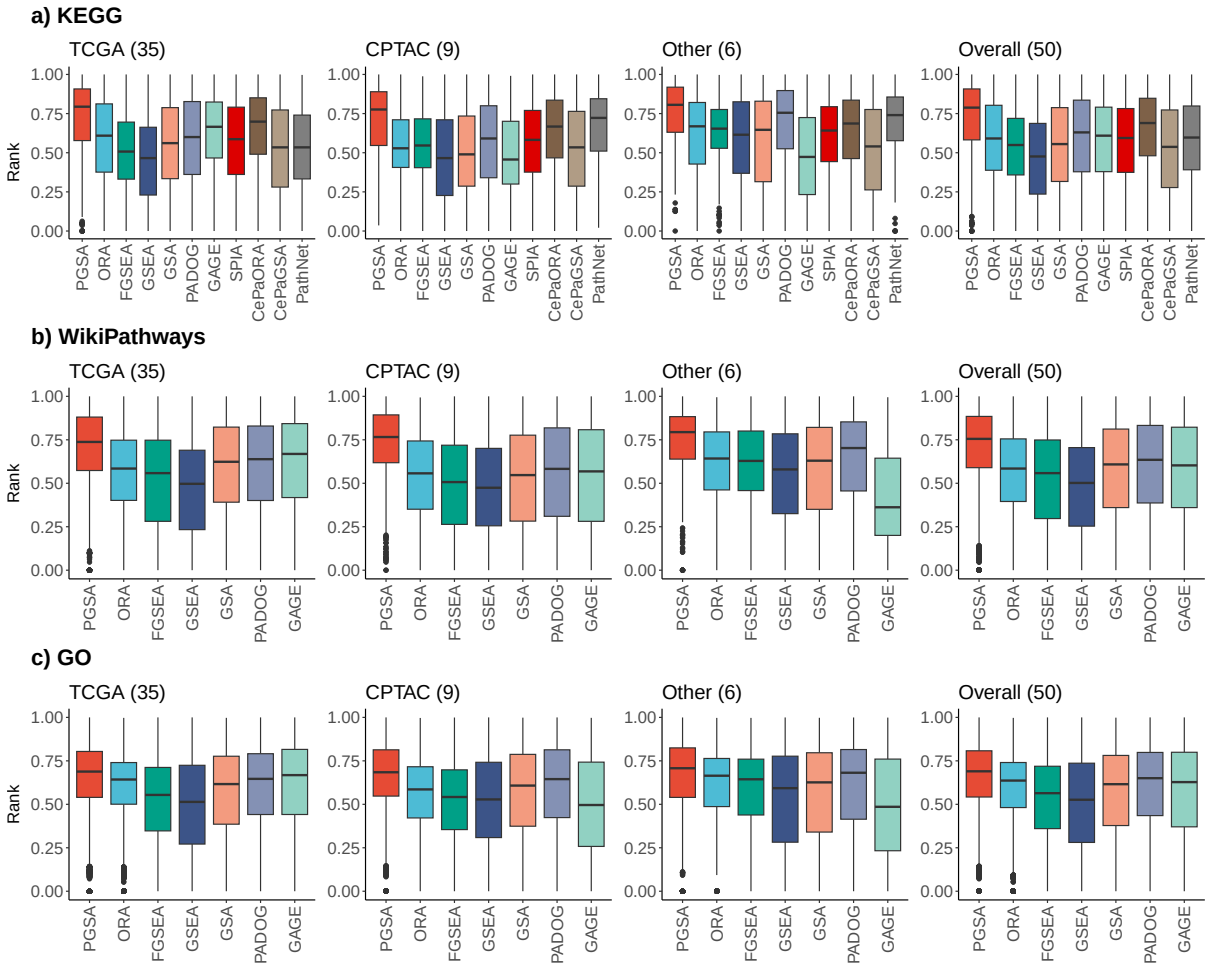

Figure 9.33: The disease-relevant rank of top 10% most significant gene sets for datasets from the GDC portal (TPM-normalized RNA-Seq) for the 11 methods and 3 gene set databases: a) KEGG, b) WikiPathways, and c) GO Biological Process. In each subfigure, the title shows the name of the cohort and the number of datasets used from the cohort. The x-axis represents the methods and the y-axis represents the disease-relevant scores of the gene sets.

Performance on counts data of RNA-Seq datasets downloaded from the GDC portal.

Figures 9.34 and 9.35 show the overall correlation between the negative logarithm of the p-values and the disease-relevant scores for each method and each gene set database using counts data of the RNA-seq datasets from the GDC portal. Similar to the results from the TPM-normalized data, PGSA consistently outperforms other methods when using the counts data.

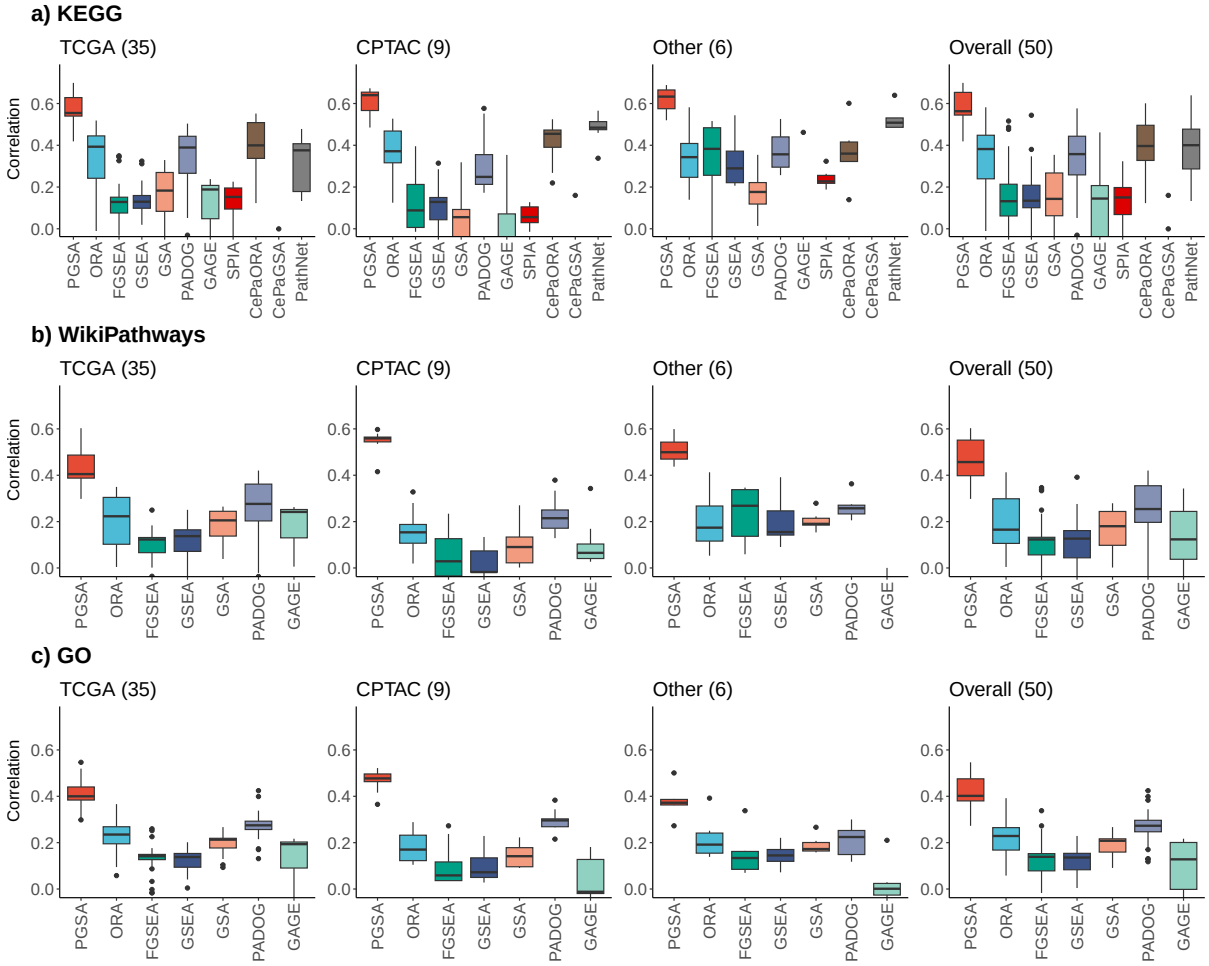

Figure 9.34: The correlation of the disease-relevant scores and the p-values of the gene sets for datasets from the GDC portal (counts data) for the 11 methods and 3 gene set databases: a) KEGG, b) WikiPathways, and c) GO Biological Process. In each subfigure, the title shows the name of the cohort and the number of datasets used from the cohort. The x-axis represents the methods and the y-axis represents the disease-relevant scores of the gene sets.

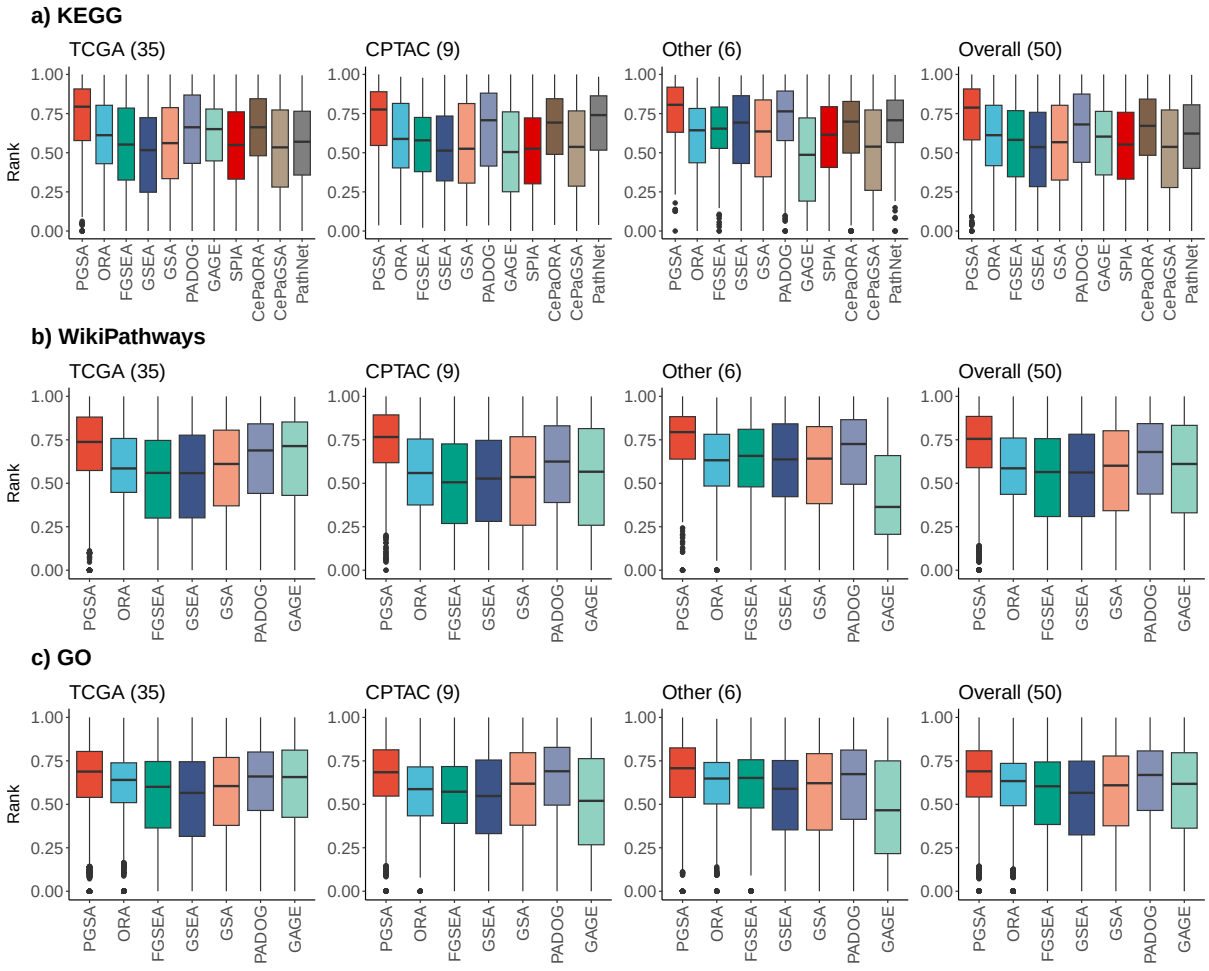

Figure 9.35: The disease-relevant rank of the top 10% most significant gene sets for datasets from the GDC portal (counts data) for the 11 methods and 3 gene set databases: a) KEGG, b) WikiPathways, and c) GO Biological Process. In each subfigure, the title shows the name of the cohort and the number of datasets used from the cohort. The x-axis represents the methods and the y-axis represents the disease-relevant scores of the gene sets.

Chapter 10

Conclusion (CPA and PGSA)

The fact that over 70 pathway analysis methods have been developed and that number is still increasing, shows that pathway analysis is a challenging problem and the current methods are not generally applicable to a wide range of datasets and conditions. In an attempt to address this issue, we have first developed a web-based platform, Consensus Pathway Analysis (CPA), that allows researchers to (1) analyze gene/protein expression data using eight popular methods (GSEA, GSA, FGSEA, PADOG, Impact Analysis, Webgestalt, KS-test, Wilcox-test), (2) perform meta-analysis of multiple datasets, (3) combine methods and datasets to find consensus results, and (4) interactively explore significantly impacted pathways across multiple analyses, and browsing relationships between pathways and genes. Our main objective is to help life scientists who are trying to understand the underlying biological mechanisms when comparing two phenotypes. The platform is user-friendly, with rich features to explore and visualize pathway analysis results. More importantly, it allows users to see the differences, as well as the consensus results across many methods and experiments. At the same time, we also aim to help bioinformaticians who are developing new pathway analysis methods.

We have also developed a novel pathway analysis method, PGSA, that is robust against noise and can detect pathways that are significantly impacted in many conditions. In an unprecedented benchmarking study using more than 400 datasets, we have shown that PGSA outperforms other popular methods in ranking targeted and disease-related pathways. This shows that PGSA potentially can be used as a general-purpose pathway analysis method and as one of the standard methods for pathway analysis.

Part III

Summary and Future Research

Chapter 11

Summary

The rapid development of high-throughput technologies has greatly impacted the field of molecular biology, allowing for the generation of vast amounts of molecular data. These platforms have enabled the comprehensive profiling of various molecular features, including gene expression, DNA methylation, and microRNA expression. By integrating these diverse data types, we gain a more holistic view of the molecular landscape underlying complex diseases, particularly cancer, where multiple molecular processes work together to influence disease progression and treatment response.

In this dissertation, we have introduced novel contributions to the areas of integrative cancer subtyping and gene set analysis. We have developed new subtyping frameworks, including MGKA, DSCC, PINSPlus, and SMRT, which effectively integrate multiple omics data types to classify cancer patients into clinically relevant subtypes. MGKA utilizes a multi-objective genetic algorithm to refine the k-means clustering algorithm and automatically determine the optimal number of subtypes. DSCC employs a consensus network approach, building patient similarity networks from individual data types and using community detection to identify robust subtypes. PINSPlus is an extension of the original PINS method, integrating multiple data types and providing a more accurate and efficient subtyping analysis. SMRT is a comprehensive framework that integrates a large number of omics data types, utilizing a randomized data transformation approach to handle noise and missing data effectively. Our approaches demonstrate robustness against noise and missing data, and their performance improves as more data types are incorporated, providing advantages over existing methods that often focus on a single

data type and may have limitations in scalability and robustness. Through extensive analyses on a wide range of cancer datasets, we have shown that MGKA, DSCC, PINSPlus, and SMRT outperform state-of-the-art methods in identifying subtypes with significant differences in patient survival and clinical characteristics. These frameworks offer a powerful and flexible approach to integrative cancer subtyping, enabling researchers to uncover novel insights into cancer heterogeneity and develop more personalized treatment strategies.

More importantly, we have developed a web-based platform, Consensus Pathway Analysis (CPA), to facilitate the application of pathway analysis methods. CPA provides a user-friendly interface for researchers to analyze their data using multiple pathway analysis methods and visualize the results interactively. The platform allows users to perform consensus pathway analysis, combining the results from different methods and datasets to identify the most robustly impacted pathways. CPA offers a rich set of visualization features, enabling researchers to explore the relationships between pathways, genes, and datasets. By integrating multiple methods and datasets, CPA helps researchers gain a more comprehensive understanding of the biological processes underlying their data and increases confidence in the identified pathways.

We have also introduced a new consensus approach for pathway analysis called Perturbation-based Gene Set Analysis (PGSA). PGSA has shown remarkable efficiency in identifying significantly impacted pathways across a wide range of diseases. It addresses several challenges faced by existing methods, such as bias towards well-studied diseases, sensitivity to noise, and limited validation. PGSA employs a novel perturbation-based approach, where input gene expression data is perturbed multiple times, and gene set analysis is performed on each perturbed dataset. This allows PGSA to capture the inherent variability and noise in the data, providing more robust and reliable results. By integrating the results from multiple perturbations, PGSA identifies pathways that are consistently impacted across different conditions, reducing the influence of dataset-specific noise and biases.

We have conducted a comprehensive benchmarking study using an unprecedented number of datasets, spanning over 30 diseases and including more than 400 individual datasets. This extensive validation establishes the robustness and reliability of PGSA across a diverse range of

biological conditions and data types. Our results demonstrate that PGSA consistently outperforms existing pathway analysis methods in identifying disease-relevant pathways, even in the presence of noise and heterogeneity in the data. Moreover, PGSA's performance remains stable across different disease types and data sources, highlighting its versatility and broad applicability. By testing PGSA on such a large number of datasets and diseases, we have ensured that its performance is not biased towards well-studied diseases or overfitted to specific datasets. This comprehensive validation sets a new standard for the evaluation of pathway analysis methods and provides confidence in PGSA's ability to uncover meaningful biological insights from diverse datasets.

In summary, this dissertation represents a significant step forward in the fields of integrative cancer subtyping and gene set analysis. The frameworks and techniques introduced here, including MGKA, DSCC, PINSPlus, SMRT, CPA, and PGSA, can be adapted and applied to various complex diseases, offering new opportunities for understanding their molecular basis and developing targeted interventions. By harnessing multi-omics data integration and advanced computational methods, we can accelerate biomedical research and translate these findings into benefits for patients worldwide.

Chapter 12

Future Research

Although the work presented in this dissertation has addressed many challenges in the field of cancer subtyping and pathway analysis, there are still many opportunities for future research. Here, we outline some potential directions for future research.

For future work on multi-omics subtyping, there are several key directions to explore. First, the subtyping framework can be extended to integrate additional types of omics data beyond mRNA expression, miRNA expression, and DNA methylation. This includes data types such as copy number variation, somatic mutation, proteomics, and metabolomics. Evaluating how these additional levels of molecular information impact the ability to define robust and clinically relevant subtypes will be an important area of investigation. Second, methods can be developed to meaningfully integrate key clinical variables with the molecular subtyping. Important clinical factors to consider include cancer stage, tumor grade, and patient demographics like age and gender. Assessing if incorporating clinical data alongside the multi-omics data provides more prognostic subtypes compared to using molecular data alone will be a valuable line of inquiry.

Third, the multi-omics subtyping approach can be adapted and applied to other complex, heterogeneous diseases beyond cancer, such as neurodegenerative diseases like Alzheimer's and Parkinson's, autoimmune disorders like rheumatoid arthritis and multiple sclerosis, and cardiovascular diseases. Evaluating the performance and utility of multi-omics subtyping in these other disease contexts will be important for extending the impact of this work. Fourth,

the subtyping framework can be extended to handle data from emerging technologies like spatial transcriptomics and single-cell sequencing. These data types can provide intratumor heterogeneity information that is lost in bulk tumor profiling, so developing methods to define subtypes that account for spatial heterogeneity and cellular composition of the tumor microenvironment will be a cutting-edge area of research.

For future work on pathway analysis, there are also several exciting directions to pursue. First, methods can be developed to identify pathways that are significantly impacted in specific biological contexts, such as specific tissue types, cell types, disease subtypes, disease stages, treatment conditions, or time points. This context-specific approach can provide a more granular understanding of when and where certain pathways are dysregulated rather than assuming a pathway is uniformly impacted across all contexts. Second, pathway analysis methods can be extended to integrate data from multiple omics levels, such as combining transcriptomics with proteomics, metabolomics, or epigenomics data. Assessing if integrating multiple data types provides more robust and biologically meaningful pathway results compared to using a single data type will be an important question to address.

Third, network-based approaches that incorporate protein-protein interaction networks or gene regulatory networks into the pathway analysis framework can be developed. This can help identify key driver genes or network modules that are significantly perturbed, rather than just individual genes, and evaluating if network-based approaches provide more interpretable and actionable results compared to traditional pathway analysis methods will be a valuable line of inquiry. Fourth, the clinical utility of pathway-based biomarkers can be evaluated, assessing if significantly impacted pathways can serve as robust biomarkers for clinical outcomes like disease prognosis, treatment response, or risk of recurrence and if pathway-based biomarkers outperform traditional single-gene biomarkers in terms of accuracy and reproducibility. Promising pathway biomarkers can then be validated in independent patient cohorts and potentially translated into clinical tests. Finally, methods can be developed to assess the impact of rare genetic variants on pathway dysregulation, which may be missed by traditional pathway analysis methods that focus on common variants, and to extend pathway analysis to single-cell

data to evaluate if certain pathways are specifically dysregulated in certain cell types or cell states.

Chapter 13

Publication list

13.1 Peer-reviewed Journal Articles

1. **Hung Nguyen**, Duc Tran, Afshin Beheshti, Jonathan M. Galazka, Sylvain V. Costes, Sorin Draghici, and Tin Nguyen. CPA: Consensus Pathway Analysis and Interactive Visualization. *Nucleic Acids Research*, 49(W1):W114-W124, 2021.

2. **Hung Nguyen**, Duc Tran, Bang Tran, Monikrishna Roy, Adam Cassell, Sergiu Dascalu, Sorin Draghici, and Tin Nguyen. SMRT: Randomized Data Transformation for Cancer Subtyping and Big Data Analysis. *Frontier in Oncology*, 2021.

3. **Hung Nguyen**, Duc Tran, Bang Tran, Bahadir Pehlivan, and Tin Nguyen. A comprehensive survey of regulatory network inference methods using single-cell RNA sequencing data. *Briefings in Bioinformatics*, 22(3):bbaa190, 2020. DOI: 10.1093/bib/bbaa190.

4. **Hung Nguyen**, Sangam Shrestha, Sorin Draghici, and Tin Nguyen. PINSPlus: A tool for tumor subtype discovery in integrated genomic data. *Bioinformatics*, 36(16): 2843-2846, 2019.

5. **Hung Nguyen**, Sangam Shrestha, Duc Tran, and Tin Nguyen. A comprehensive survey for active subnetwork identification. *Frontiers in Genetics*, DOI: 10.3389/fgene.2019.00155, 2019.

6. Caio de Carvalho, Ian Murray, **Hung Nguyen**, Tin Nguyen, David Cantu. Acyltransferase families that act on thioesters: Sequences, structures, and mechanisms. *PROTEINS: Structure, Function, and Bioinformatics*, 2023.

7. Egle Cekanaviciute, Duc Tran, **Hung Nguyen**, Alejandra Lopez Macha, Eloise Pariset, Sasha Langley, Giulia Babbi, Sherina Malkani, Sébastien Penninckx, Jonathan C. Schisler, Tin Nguyen, Gary H.Karpen, and Sylvain.V. Costesa. Mouse genomic associations with in vitro sensitivity to simulated space radiation. *Life Sciences in Space Research*, 2022.

8. Evagelia C. Laiakis, Maisa Pinheiro, Tin Nguyen, **Hung Nguyen**, Afshin Beheshti, Sucharita M. Dutta, William K. Russell, Mark R. Emmett, and Richard A. Britten. Quantitative proteomic analytic approaches to identify metabolic changes in the medial prefrontal cortex of rats exposed to space radiation. *Frontiers in Physiology*, 1643, 2022.

9. Bang Tran, **Hung Nguyen**, Duc Tran, Seungil Ro, and Tin Nguyen. scCAN: single-cell clustering using autoencoder and network fusion. *Scientific reports*, 12(1), 1-10, 2022.

10. Duc Tran, Bang Tran, **Hung Nguyen**, and Tin Nguyen. A novel method for single-cell data imputation using subspace regression. *Scientific Reports*, 12(1):1-13, 2022.

11. Benjamin T. Caswell, Caio C. de Carvalho, **Hung Nguyen**, Monikrishna Roy, Tin Nguyen, David C. Cantu. Thioesterase enzyme families: Functions, structures, and mechanisms. *Protein Science*, 2021.

12. Attila Gabor, Marco Tognetti, Alice Driessen, Jovan Tanevski, Baosen Guo, Wencai Cao, He Shen, Thomas Yu, Verena Chung, Single Cell Signaling in Breast Cancer DREAM Consortium members, Bernd Bodenmiller, and Julio Saez-Rodriguez. Cell-to-cell and type-to-type heterogeneity of signaling networks: insights from the crowd. *Molecular Systems Biology*, 17(10):e10402, 2021.

13. Duc Tran, **Hung Nguyen**, Bang Tran, Carlo La Vecchia, Hung N. Luu, and Tin Nguyen. Fast and precise single-cell data analysis using hierarchical autoencoder. *Nature Communications*, 12(1):1-10, 2021.

14. Duc Tran, **Hung Nguyen**, Uyen Le, Hung N. Luu, and Tin Nguyen. A novel method for cancer subtyping and risk prediction using consensus factor analysis. *Frontiers in Oncology*, 10:1052, 2020. DOI: 10.3389/fonc.2020.01052.

15. Nguyen Quang-Huy, **Nguyen Hung**, Nguyen Tin, Le Duc-Hau. Multi-Omics Analysis Detects Novel Prognostic Subgroups of Breast Cancer. *Frontiers in Genetics*. 2020. DOI: 10.3389/fgene.2020.574661.

16. Edward R. Cruz, **Hung Nguyen**, Tin Nguyen, and Ian S. Wallace. FAT-PTM: A post-translational modification database for analysis of proteins and metabolic pathways. *The Plant Journal*, DOI: 10.1111/tpj.14372, 2019.

17. Adib Shafi, Tin Nguyen, Azam Peyvandipour, **Hung Nguyen**, and Sorin Draghici. A multi-cohort and multi-omics meta-analysis framework to identify network-based gene signatures. *Frontiers in Genetics*, DOI: 10.3389/fgene.2019.00159, 2019.

18. Grant Schissler, **Hung Nguyen**, Tin Nguyen, Juli Petereit, and Vincent Gardeux. Statistical Software. *Wiley StatsRef: Statistics Reference Online*,
DOI: 10.1002/9781118445112.stat00527.pub2, 2019.

13.2 Peer-reviewed Conference Papers

1. **Hung Nguyen**, Bang Tran, Duc Tran, Quang-Huy Nguyen, Duc-Hau Le, Tin Nguyen. Disease subtyping using community detection from consensus networks. In *2020 12th International Conference on Knowledge and Systems Engineering (KSE)*, pages 318-323. IEEE, 2020.

2. **Hung Nguyen**, Louis J. Sushil, and Tin Nguyen. MGKA: A genetic algorithm-based clustering technique for genomic data. In *2019 IEEE Congress on Evolutionary Computation (CEC)*, pages 103-110. IEEE, 2019.

3. Duc Tran, Ha Nguyen, **Hung Nguyen**, and Tin Nguyen. DWEN: A novel method for accurate estimation of cell type compositions from bulk data samples. In *2022 14th International Conference on Knowledge and Systems Engineering (KSE)*, pp. 1-6. IEEE, 2022.

4. Nathan Thom, **Hung Nguyen**, Emily M Hand. Consensus Subspace Clustering. In *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 391-395. IEEE.

5. Duc Tran, **Hung Nguyen**, Frederick C. Harris, Jr., Tin Nguyen. Single-cell RNA sequencing data imputation using similarity preserving network. In *2021 13th International Conference on Knowledge and Systems Engineering (KSE)*, pages 1-6. IEEE.

6. Duc Tran, Frederick C. Harris, Bang Tran, Nam Sy Vo, **Hung Nguyen**, and Tin Nguyen. Single-Cell RNA Sequencing Data Imputation Using Deep Neural Network. In *ITNG 2021 18th International Conference on Information Technology-New Generations*, pages 403-410. Springer, 2021.

7. Bang Tran, Duc Tran, **Hung Nguyen**, Nam Sy Vo, and Tin Nguyen. RIA: a novel Regression-based Imputation Approach for single-cell RNA sequencing. In *2019 11th International Conference on Knowledge and Systems Engineering (KSE)*, pages 1-9. IEEE, 2019.

8. Brian Marks, Nina Hees, **Hung Nguyen**, and Tin Nguyen. MIA: A Multi-cohort Integrated Analysis for biomarker identification. In *Proceedings of the 9th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pages 360-365. ACM, 2018.

9. Phung Do, **Hung Nguyen**, Nguyen Vu, and Dung Tran. A context-aware recommendation framework in e-learning environment. In *International Conference on Future Data and Security Engineering* pages 272-284. Springer, Cham, 2015.

References

[1] Hung Nguyen, Duc Tran, Jonathan M. Galazka, Sylvain V. Costes, Afshin Beheshti, Sorin Draghici, and Tin Nguyen. CPA: A web-based platform for consensus pathway analysis and interactive visualization. *Nucleic Acids Research*, 49(W1):W114–W124, 2021.

[2] Hung Nguyen, Duc Tran, Bang Tran, Monikrishna Roy, Adam Cassell, Sergiu Dascalu, Sorin Draghici, and Tin Nguyen. SMRT: Randomized data transformation for cancer subtyping and big data analysis. *Frontiers in Oncology*, 11:725133, 2021.

[3] Hung Nguyen, Duc Tran, Bang Tran, Bahadir Pehlivan, and Tin Nguyen. A comprehensive survey of regulatory network inference methods using single-cell RNA sequencing data. *Briefings in Bioinformatics*, 22(3):1–15, 2021.

[4] Hung Nguyen, Sangam Shrestha, Sorin Draghici, and Tin Nguyen. PINSPlus: A tool for tumor subtype discovery in integrated genomic data. *Bioinformatics*, 35(16):2843–2846, 2019.

[5] Hung Nguyen, Sangam Shrestha, Duc Tran, Adib Shafi, Sorin Draghici, and Tin Nguyen. A comprehensive survey of tools and software for active subnetwork identification. *Frontiers in Genetics*, 10:155, 2019.

[6] Caio C de Carvalho, Ian P Murray, Hung Nguyen, Tin Nguyen, and David C Cantu. Acyltransferase families that act on thioesters: Sequences, structures, and mechanisms. *Proteins: Structure, Function, and Bioinformatics*, 92(2):157–169, 2024.

[7] Egle Cekanaviciute, Duc Tran, Hung Nguyen, Alejandra Lopez Macha, Eloise Pariset, Sasha Langley, Giulia Babbi, Sherina Malkani, Sébastien Penninckx, Jonathan C.

Schisler, Tin Nguyen, Gary H. Karpen, and Sylvain V. Costes. Mouse genomic associations with in vitro sensitivity to simulated space radiation. *Life Sciences in Space Research*, DOI: 10.1016/j.lssr.2022.07.006, 2022.

[8] Evagelia C. Laiakis, Maisa Pinheiro, Tin Nguyen, Hung Nguyen, Afshin Beheshti, Sucharita M. Dutta, William K. Russell, Mark R. Emmett, and Richard Britten. Quantitative proteomic analytic approaches to identify metabolic changes in the medial prefrontal cortex of rats exposed to space radiation. *Frontiers in Physiology*, DOI: 10.3389/fphys.2022.971282, 2022.

[9] Bang Tran, Duc Tran, Hung Nguyen, Seungil Ro, and Tin Nguyen. scCAN: single-cell clustering using autoencoder and network fusion. *Scientific Reports*, 12:10267, 2022.

[10] Duc Tran, Bang Tran, Hung Nguyen, and Tin Nguyen. A novel method for single-cell data imputation using subspace regression. *Scientific Reports*, 12:2697, 2022.

[11] Benjamin T. Caswell, Caio C. de Carvalho, Hung Nguyen, Monikrishna Roy, Tin Nguyen, and David C. Cantu. Thioesterase enzyme families: Functions, structures, and mechanisms. *Protein Science*, 31(3):652–676, 2022.

[12] Attila Gabor, Marco Tognetti, Alice Driessen, Jovan Tanevski, Baosen Guo, Wencai Cao, He Shen, Thomas Yu, Verena Chung, Bernd Bodenmiller, Julio Saez-Rodriguez, Augustinas Prusokas, Alidivinas Prusokas, Renata Retkute, Anand Rajasekar, Karthik Raman, Malvika Sudhakar, Raghunathan Rengaswamy, Edward S.C. Shih, Min jeong Kim, Changje Cho, Dohyang Kim, Hyeju Oh, Jinseub Hwang, Kim Jongtae, Yeongeun Nam, Sanghoo Yoon, Taeyong Kwon, Kyeongjun Lee, Sarika Chaudhary, Nehal Sharma, Shreya Bande, Gao Gao fan zhu Cankut Cubuk, Pelin Gundogdu, Joaquin Dopazo, Kinza Rian, Carlos Loucera, Matias M Falco, Martin Garrido-Rodriguez, Maria Peña-Chilet, Huiyuan Chen, Gabor Turu, Laszlo Hunyadi, Adam Misak, Baosen Guo, Wencai Cao, He Shen, Lisheng Zhou, Xiaoqing Jiang, Pieta Zhang, Aakansha Rai, Rintu Kutum, Sadhna Rana, Rajgopal Srinivasan, Swatantra Pradhan, James Li, Vladimir Bajic, Christophe Van Neste, Didier Barradas-bautista, Somayah Abdullah Albarade, Igor

Nikolskiy, Musalula Sinkala, Duc Tran, Hung Nguyen, Tin Nguyen, Alexander Wu, Benjamin DeMeo, Brian Hie, Rohit Singh, Jiwei Liu, Xueer Chen, Leonor Saiz, Jose M. G Vilar, Peng Qiu, Akash Gosain, Anjali Dhall, Dinesh Bajaj, Harpreet Kaur, Krishna Bagaria, Mayank Chauhan, Neelam Sharma, Gajendra Raghava, Sumeet Patiyal, Jianye Hao, Jiajie Peng, Shangyi Ning, Yi Ma, Zhongyu Wei, Atte Aalto, Jorge Goncalves, Laurent Mombaerts, Xinnan Dai, Jie Zheng, Piyushkumar Mundra, Fan Xu, Jie Wang, Krishna Kant Singh, and Mingyu Lee. Cell-to-cell and type-to-type heterogeneity of signaling networks: insights from the crowd. *Molecular Systems Biology*, 17:e10402, 2021.

[13] Duc Tran, Hung Nguyen, Bang Tran, Carlo La Vecchia, Hung N. Luu, and Tin Nguyen. Fast and precise single-cell data analysis using hierarchical autoencoder. *Nature Communications*, 12:1029, 2021.

[14] Duc Tran, Hung Nguyen, Uyen Le, George Bebis, Hung N. Luu, and Tin Nguyen. A novel method for cancer subtyping and risk prediction using consensus factor analysis. *Frontiers in Oncology*, 10:1052, 2020.

[15] Quang-Huy Nguyen, Hung Nguyen, Tin Nguyen, and Duc-Hau Le. Multi-omics analysis detects novel prognostic subgroups of breast cancer. *Frontiers in Genetics*, 11:1265, 2020.

[16] Edward Cruz, Hung Nguyen, Tin Nguyen, and Ian Wallace. Functional analysis tools for post-translational modification: a post-translational modification database for analysis of proteins and metabolic pathways. *The Plant Journal*, 99(5):1003–1013, 2019.

[17] Adib Shafi, Tin Nguyen, Azam Peyvandipour, Hung Nguyen, and Sorin Draghici. A multi-cohort and multi-omics meta-analysis framework to identify network-based gene signatures. *Frontiers in Genetics*, 10:159, 2019.

[18] Alfred G. Schissler, Hung Nguyen, Tin Nguyen, Juli Petereit, and Vincent Gardeux. *Statistical Software*, volume 10.1002/9781118445112.stat00527.pub2, pages 1–11. American Cancer Society, 2019.

[19] Hung Nguyen, Bang Tran, Duc Tran, Quang-Huy Nguyen, Duc-Hau Le, and Tin Nguyen. Disease subtyping using community detection from consensus networks. In *2020 12th International Conference on Knowledge and Systems Engineering (KSE)*, pages 318–323. IEEE, 2020.

[20] Hung Nguyen, Sushil J Louis, and Tin Nguyen. MGKA: A genetic algorithm-based clustering technique for genomic data. In *2019 IEEE Congress on Evolutionary Computation (CEC)*, pages 103–110. IEEE, 2019.

[21] Duc Tran, Ha Nguyen, Hung Nguyen, and Tin Nguyen. Dwen: A novel method for accurate estimation of cell type compositions from bulk data samples. In *2022 14th International Conference on Knowledge and Systems Engineering (KSE)*, pages 1–6. IEEE, 2022.

[22] Nathan Thom, Hung Nguyen, and Emily M Hand. Consensus Subspace Clustering. In *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 391–395. IEEE, 2021.

[23] Duc Tran, Frederick C Harris, Bang Tran, Nam Sy Vo, Hung Nguyen, and Tin Nguyen. Single-cell RNA sequencing data imputation using deep neural network. In *ITNG 2021 18th International Conference on Information Technology-New Generations*, pages 403–410. Springer, 2021.

[24] Duc Tran, Hung Nguyen, Frederick C Harris, and Tin Nguyen. Single-cell RNA sequencing data imputation using similarity preserving network. In *2021 13th International Conference on Knowledge and Systems Engineering (KSE)*, pages 1–6. IEEE, 2021.

[25] Bang Tran, Duc Tran, Hung Nguyen, Nam Sy Vo, and Tin Nguyen. RIA: a novel Regression-based Imputation Approach for single-cell RNA sequencing. In *2019 11th International Conference on Knowledge and Systems Engineering (KSE)*, pages 1–9. IEEE, 2019.

[26] Brian Marks, Nina Hees, Hung Nguyen, and Tin Nguyen. MIA: A Multi-cohort Integrated Analysis for biomarker identification. In *Proceedings of the 9th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*. ACM, 2018.

[27] Antonija Kreso and John E Dick. Evolution of the cancer stem cell model. *Cell Stem Cell*, 14(3):275–291, 2014.

[28] Rebecca A Burrell, Nicholas McGranahan, Jiri Bartek, and Charles Swanton. The causes and consequences of genetic heterogeneity in cancer evolution. *Nature*, 501(7467):338–345, 2013.

[29] Mel Greaves. Darwin and evolutionary tales in leukemia. *ASH Education Program Book*, 2009(1):3–12, 2009.

[30] Mel Greaves and Carlo C Maley. Clonal evolution in cancer. *Nature*, 481(7381):306–313, 2012.

[31] Daniele Ramazzotti, Avantika Lal, Bo Wang, Serafim Batzoglou, and Arend Sidow. Multi-omic tumor data reveal diversity of molecular mechanisms that correlate with survival. *Nature Communications*, 9:4453, 2018.

[32] Laura J Esserman, Ian M Thompson, Brian Reid, Peter Nelson, David F Ransohoff, H Gilbert Welch, Shelley Hwang, Donald A Berry, Kenneth W Kinzler, William C Black, Mina Bissell, Howard Parnes, and Sudhir Srivastava. Addressing overdiagnosis and overtreatment in cancer: a prescription for change. *The Lancet Oncology*, 15(6):e234–e242, 2014.

[33] Laura Esserman, Yiwey Shieh, and Ian Thompson. Rethinking screening for breast cancer and prostate cancer. *Journal of the American Medical Association*, 302(15):1685–1692, 2009.

[34] Hidetaka Uramoto and Fumihiro Tanaka. Recurrence after surgery in patients with NSCLC. *Translational Lung Cancer Research*, 3(4):242–249, 2014.

[35] Lisa A Carey, Charles M Perou, Chad A Livasy, Lynn G Dressler, David Cowan, Kathleen Conway, Gamze Karaca, Melissa A Troester, Chiu Kit Tse, Sharon Edmiston, et al. Race, breast cancer subtypes, and survival in the carolina breast cancer study. *Jama*, 295(21):2492–2502, 2006.

[36] Carol A Parise, Katrina R Bauer, Monica M Brown, and Vincent Caggiano. Breast cancer subtypes as defined by the estrogen receptor (er), progesterone receptor (pr), and the human epidermal growth factor receptor 2 (her2) among women with invasive breast cancer in california, 1999–2004. *The breast journal*, 15(6):593–602, 2009.

[37] Li Yin, Jiang-Jie Duan, Xiu-Wu Bian, and Shi-cang Yu. Triple-negative breast cancer molecular subtyping and treatment progress. *Breast Cancer Research*, 22(1):1–13, 2020.

[38] Tin Nguyen, Rebecca Tagett, Diana Diaz, and Sorin Draghici. A novel approach for data integration and disease subtyping. *Genome Research*, 27:2025–2039, 2017.

[39] Eric A Collisson, Peter Bailey, David K Chang, and Andrew V Biankin. Molecular subtypes of pancreatic cancer. *Nature Reviews Gastroenterology & Hepatology*, 16(4):207–220, 2019.

[40] Rodrigo Dienstmann, Louis Vermeulen, Justin Guinney, Scott Kopetz, Sabine Tejpar, and Josep Tabernero. Consensus molecular subtypes and the evolution of precision medicine in colorectal cancer. *Nature Reviews Cancer*, 17:79–92, 2017.

[41] The Cancer Genome Atlas Research Network. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61, 2012.

[42] Robert L. Grossman, Allison P. Heath, Vincent Ferretti, Harold E. Varmus, Douglas R. Lowy, Warren A. Kibbe, and Louis M. Staudt. Toward a shared vision for cancer genomic data. *New England Journal of Medicine*, 375(12):1109–1112, 2016.

[43] Christina Curtis, Sohrab P. Shah, Suet-Feung Chin, Gulisa Turashvili, Oscar M. Rueda, Mark J. Dunning, Doug Speed, Andy G. Lynch, Shamith Samarajiwa, Yinyin Yuan, Stefan Graf, Gavin Ha, Gholamreza Haffari, Ali Bashashati, Roslin Russell, Steven McKinney, METABRIC Group, Anita Langerød, Andrew Green, Elena Provenzano, Gordon

Wishart, Sarah Pinder, Peter Watson, Florian Markowetz, Leigh Murphy, Ian Ellis, Arnie Purushotham, Anne-Lise Børresen-Dale, James D. Brenton, Simon Tavaré, Carlos Caldas, and Samuel Aparicio. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486(7403):346–352, 2012.

[44] Cathie Sudlow, John Gallacher, Naomi Allen, Valerie Beral, Paul Burton, John Danesh, Paul Downey, Paul Elliott, Jane Green, Martin Landray, Bette Liu, Paul Matthews, Giok Ong, Jill Pell, Alan Silman, Alan Young, Tim Sprosen, Tim Peakman, and Rory Collins. UK biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS Medicine*, 12(3):e1001779, 2015.

[45] James MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 4, pages 281–297. California, USA, 1967.

[46] Leonard Kaufman and Peter J Rousseeuw. Partitioning around medoids (program pam). *Finding groups in data: an introduction to cluster analysis*, pages 68–125, 1990.

[47] Jelili Oyelade, Itunuoluwa Isewon, Funke Oladipupo, Olufemi Aromolaran, Efosa Uwoghiren, Faridah Ameh, Moses Achas, and Ezekiel Adebiyi. Clustering algorithms: Their application to gene expression data. *Bioinformatics and Biology insights*, 10:BBI–S38316, 2016.

[48] Paul S Bradley and Usama M Fayyad. Refining initial points for k-means clustering. In *ICML*, volume 98, pages 91–99. Citeseer, 1998.

[49] Stephen J Redmond and Conor Heneghan. A method for initialising the k-means clustering algorithm using kd-trees. *Pattern Recognition Letters*, 28(8):965–973, 2007.

[50] Michael Laszlo and Sumitra Mukherjee. A genetic algorithm that exchanges neighboring centers for k-means clustering. *Pattern Recognition Letters*, 28(16):2359–2366, 2007.

[51] Jian-Feng Lu, JB Tang, Zhen-Min Tang, and Jing-Yu Yang. Hierarchical initialization approach for k-means clustering. *Pattern Recognition Letters*, 29(6):787–795, 2008.

[52] Xiaoping Qin and Shijue Zheng. A new method for initialising the k-means clustering algorithm. In *2009 2nd International Symposium on Knowledge Acquisition and Modeling, KAM 2009*, volume 2, pages 41–44. IEEE, 2009.

[53] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[54] Sanghamitra Bandyopadhyay, Ujjwal Maulik, and Malay Kumar Pakhira. Clustering using simulated annealing with probabilistic redistribution. *International Journal of Pattern Recognition and Artificial Intelligence*, 15(02):269–285, 2001.

[55] John Holland. Adaptation in natural and artificial systems: an introductory analysis with application to biology. *Control and artificial intelligence*, 1975.

[56] K Krishna and M Narasimha Murty. Genetic k-means algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 29(3):433–439, 1999.

[57] M Anusha and JGR Sathiaseelan. An enhanced k-means genetic algorithms for optimal clustering. In *2014 IEEE International Conference on Computational Intelligence and Computing Research, IEEE ICCIC 2014*, pages 1–5. IEEE, 2014.

[58] Yi Lu, Shiyong Lu, Farshad Fotouhi, Youping Deng, and Susan J Brown. Fgka: A fast genetic k-means clustering algorithm. In *Proceedings of the 2004 ACM symposium on Applied computing*, pages 622–623. ACM, 2004.

[59] Dharmendra K Roy and Lokesh K Sharma. Genetic k-means clustering algorithm for mixed numeric and categorical data sets. *International Journal of Artificial Intelligence & Applications*, 1(2):23–28, 2010.

[60] Zheyun Feng. Data clustering using genetic algorithms. *Evolutionary Computation: Project Report, CSE484*, 2012.

[61] Yi Lu, Shiyong Lu, Farshad Fotouhi, Youping Deng, and Susan J Brown. Incremental genetic k-means algorithm and its application in gene expression data analysis. *BMC bioinformatics*, 5(1):172, 2004.

[62] Stefano Monti, Pablo Tamayo, Jill Mesirov, and Todd Golub. Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. *Machine Learning*, 52(1-2):91–118, 2003.

[63] Matthew D. Wilkerson and D. Neil Hayes. ConsensusClusterPlus: a class discovery tool with confidence assessments and item tracking. *Bioinformatics*, 26(12):1572–1573, 2010.

[64] Alberto Pascual-Montano, Pedro Carmona-Saez, Monica Chagoyen, Francisco Tirado, Jose M Carazo, and Roberto D Pascual-Marqui. bionmf: a versatile tool for non-negative matrix factorization in biology. *BMC bioinformatics*, 7(1):1–9, 2006.

[65] Zhong-Yuan Zhang, Tao Li, Chris Ding, Xian-Wen Ren, and Xiang-Sun Zhang. Binary matrix factorization for analyzing gene expression data. *Data Mining and Knowledge Discovery*, 20(1):28–52, 2010.

[66] Kentaro Inamura, Takeshi Fujiwara, Yujin Hoshida, Takayuki Isagawa, Michael H Jones, Carl Virtanen, Miyuki Shimane, Yukitoshi Satoh, Sakae Okumura, Ken Nakagawa, et al. Two subclasses of lung squamous cell carcinoma with different gene expression profiles and prognosis identified by hierarchical clustering and non-negative matrix factorization. *Oncogene*, 24(47):7105–7113, 2005.

[67] Chun-Hou Zheng, De-Shuang Huang, Lei Zhang, and Xiang-Zhen Kong. Tumor clustering using nonnegative matrix factorization with gene selection. *IEEE Transactions on Information Technology in Biomedicine*, 13(4):599–607, 2009.

[68] Jim Jing-Yan Wang, Xiaolei Wang, and Xin Gao. Non-negative matrix factorization by maximizing correntropy for cancer clustering. *BMC bioinformatics*, 14(1):1–11, 2013.

[69] Ran He, Wei-Shi Zheng, and Bao-Gang Hu. Maximum correntropy criterion for robust face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1561–1576, 2010.

[70] Eduardo R Hruschka, Leandro Nunes de Castro, and Ricardo JGB Campello. Evolutionary algorithms for clustering gene-expression data. In *Fourth IEEE International Conference on Data Mining (ICDM'04)*, pages 403–406. IEEE, 2004.

[71] Vito Di Gesú, Raffaele Giancarlo, Giosué Lo Bosco, Alessandra Raimondi, and Davide Scaturro. Genclust: A genetic algorithm for clustering gene expression data. *BMC bioinformatics*, 6(1):1–11, 2005.

[72] Anupam Chakraborty and Hitashyam Maka. Biclustering of gene expression data using genetic algorithm. In *2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology*, pages 1–8. IEEE, 2005.

[73] David Gutiérrez-Avilés, Cristina Rubio-Escudero, Francisco Martínez-Álvarez, and José C Riquelme. Trigen: A genetic algorithm to mine triclusters in temporal gene expression data. *Neurocomputing*, 132:42–53, 2014.

[74] Anirban Mukhopadhyay, Ujjwal Maulik, Sanghamitra Bandyopadhyay, and Benedikt Brors. Goga: Go-driven genetic algorithm-based fuzzy clustering of gene expression data. In *2010 International Conference on Systems in Medicine and Biology*, pages 221–226. IEEE, 2010.

[75] Ravindra Krovi. Genetic algorithms for clustering: a preliminary investigation. In *System Sciences, 1992. Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences*, volume 4, pages 540–544. IEEE, 1992.

[76] Paul Scheunders. A genetic c-means clustering algorithm applied to color image quantization. *Pattern recognition*, 30(6):859–866, 1997.

[77] Ujjwal Maulik and Sanghamitra Bandyopadhyay. Genetic algorithm-based clustering technique. *Pattern recognition*, 33(9):1455–1465, 2000.

[78] Sanghamitra Bandyopadhyay and Ujjwal Maulik. An evolutionary technique based on k-means algorithm for optimal clustering in rn. *Information Sciences*, 146(1-4):221–237, 2002.

[79] Dingming Wu, Dongfang Wang, Michael Q. Zhang, and Jin Gu. Fast dimension reduction and integrative clustering of multi-omics data using low-rank approximation: application to cancer molecular classification. *BMC Genomics*, 16(1):1022, 2015.

[80] Nora K. Speicher and Nico Pfeifer. Integrating different data types by regularized unsupervised multiple kernel learning with application to cancer subtype discovery. *Bioinformatics*, 31(12):i268–i275, 2015.

[81] Qianxing Mo, Sijian Wang, Venkatraman E. Seshan, Adam B. Olshen, Nikolaus Schultz, Chris Sander, R. Scott Powers, Marc Ladanyi, and Ronglai Shen. Pattern discovery and cancer gene identification in integrated cancer genomic data. *Proceedings of the National Academy of Sciences*, 110(11):4245–4250, 2013.

[82] Qianxing Mo, Ronglai Shen, Cui Guo, Marina Vannucci, Keith S. Chan, and Susan G. Hilsenbeck. A fully Bayesian latent variable model for integrative clustering analysis of multi-type omics data. *Biostatistics*, 19(1):71–86, 2018.

[83] Pietro Coretto, Angela Serra, and Roberto Tagliaferri. Robust clustering of noisy high-dimensional gene expression data for patients subtyping. *Bioinformatics*, 34(23):4064–4072, 2018.

[84] Ashar Ahmad and Holger Fröhlich. Towards clinically more relevant dissection of patient heterogeneity via survival-based bayesian clustering. *Bioinformatics*, 33(22):3558–3566, 2017.

[85] Eric F. Lock and David B. Dunson. Bayesian consensus clustering. *Bioinformatics*, 29(20):2610–2616, 2013.

[86] Paul Kirk, Jim E. Griffin, Richard S. Savage, Zoubin Ghahramani, and David L. Wild. Bayesian correlated clustering to integrate multiple datasets. *Bioinformatics*, 28(24):3290–3297, 2012.

[87] Eric F Lock, Katherine A Hoadley, James Stephen Marron, and Andrew B Nobel. Joint and individual variation explained (jive) for integrated analysis of multiple data types. *The Annals of Applied Statistics*, 7(1):523, 2013.

[88] Chen Meng, Bernhard Kuster, Aedín C Culhane, and Amin Moghaddas Gholami. A multivariate approach to the integration of multi-omics datasets. *BMC Bioinformatics*, 15:162, 2014.

[89] Chen Meng, Dominic Helm, Martin Frejno, and Bernhard Kuster. moCluster: Identifying joint patterns across multiple omics data sets. *Journal of Proteome Research*, 15(3):755–765, 2016.

[90] Wenyuan Li, Shihua Zhang, Chun-Chi Liu, and Xianghong Jasmine Zhou. Identifying multi-layer gene regulatory modules from multi-dimensional genomic data. *Bioinformatics*, 28(19):2458–2466, 2012.

[91] Na Yu, Ying-Lian Gao, Jin-Xing Liu, Junliang Shang, Rong Zhu, and Ling-Yun Dai. Co-differential gene selection and clustering based on graph regularized multi-view NMF in cancer genomic data. *Genes*, 9(12):586, 2018.

[92] Jialu Liu, Chi Wang, Jing Gao, and Jiawei Han. Multi-view clustering via joint nonnegative matrix factorization. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 252–260. SIAM, 2013.

[93] Prabhakar Chalise and Brooke L. Fridley. Integrative clustering of multi-level 'omic data based on non-negative matrix factorization algorithm. *PLOS ONE*, 12(5):e0176278, 2017.

[94] Zi Yang and George Michailidis. A non-negative matrix factorization method for detecting modules in heterogeneous omics multi-modal data. *Bioinformatics*, 32(1):1–8, 2016.

[95] Shihua Zhang, Chun-Chi Liu, Wenyuan Li, Hui Shen, Peter W. Laird, and Xianghong Jasmine Zhou. Discovery of multi-dimensional modules by integrative analysis of cancer genomic data. *Nucleic Acids Research*, 40(19):9379–9391, 2012.

[96] Daniela M Witten and Robert J Tibshirani. Extensions of sparse canonical correlation analysis with applications to genomic data. *Statistical Applications in Genetics and Molecular Biology*, 8(1):28, 2009.

[97] Bo Wang, Aziz M. Mezlini, Feyyaz Demir, Marc Fiume, Zhuowen Tu, Michael Brudno, Benjamin Haibe-Kains, and Anna Goldenberg. Similarity network fusion for aggregating data types on a genomic scale. *Nature Methods*, 11(3):333–337, 2014.

[98] Yinyin Yuan, Richard S Savage, and Florian Markowetz. Patient-specific data fusion defines prognostic cancer subtypes. *PLoS Computational Biology*, 7(10):e1002227, 2011.

[99] Qianqian Shi, Chuanchao Zhang, Minrui Peng, Xiangtian Yu, Tao Zeng, Juan Liu, and Luonan Chen. Pattern fusion analysis by adaptive alignment of multiple heterogeneous omics data. *Bioinformatics*, 33(17):2706–2714, 2017.

[100] Zhiguang Huo and George Tseng. Integrative sparse k-means with overlapping group lasso in genomic applications for disease subtype discovery. *The Annals of Applied Statistics*, 11(2):1011, 2017.

[101] Nimrod Rappoport and Ron Shamir. NEMO: Cancer subtyping by integration of partial multi-omic data. *Bioinformatics*, 35(18):3348–3356, 2019.

[102] Anil K. Jain and Richard C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

[103] Carlos B Lucasius, Adrie D Dane, and Gerrit Kateman. On k-medoid clustering of large data sets with the aid of a genetic algorithm: background, feasiblity and comparison. *Analytica Chimica Acta*, 282(3):647–669, 1993.

[104] Weiguo Sheng and Xiaohui Liu. A hybrid algorithm for k-medoid clustering of large data sets. In *Proceedings of the 2004 Congress on Evolutionary Computation (IEEE Cat. No.04TH8753)*, volume 1, pages 77–82. IEEE, 2004.

[105] Juha Kivijärvi, Pasi Fränti, and Olli Nevalainen. Self-adaptive genetic algorithm for clustering. *Journal of Heuristics*, 9(2):113–129, 2003.

[106] Ram Bhushan Agrawal, K Deb, and RB Agrawal. Simulated binary crossover for continuous search space. *Complex Systems*, 9(2):115–148, 1995.

[107] David L Davies and Donald W Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.

[108] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[109] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[110] Ruprecht Kuner, Thomas Muley, Michael Meister, Markus Ruschhaupt, Andreas Buness, Elizabeth C Xu, Phillipp Schnabel, Arne Warth, Annemarie Poustka, Holger Sultmann, and Hans Hoffmann. Global gene expression analysis reveals specific patterns of cell junctions in non-small cell lung cancer subtypes. *Lung Cancer*, 63(1):32–38, 2009.

[111] Jun Hou, Joachim Aerts, Bianca Den Hamer, Wilfred Van Ijcken, Michael Den Bakker, Peter Riegman, Cor van der Leest, Peter van der Spek, John A Foekens, Henk C Hoogsteden, Frank Grosveld, and Sjaak Philipsen. Gene expression-based classification of non-small cell lung carcinomas and survival prediction. *PLoS ONE*, 5(4):e10312, 2010.

[112] Adi L Tarca, Mario Lauria, Michael Unger, Erhan Bilal, Stephanie Boue, Kushal Kumar Dey, Julia Hoeng, Heinz Koeppl, Florian Martin, Pablo Meyer, Preetam Nandy, Raquel Norel, Manuel Peitsch, Jeremy J Rice, Roberto Romero, Gustavo Stolovitzky, Marja Talikka, Yang Xiang, Christoph Zechner, and IMPROVER DSC Collaborators. Strengths and limitations of microarray-based phenotype prediction: lessons learned from the IMPROVER diagnostic signature challenge. *Bioinformatics*, 29(22):2892–2899, 2013.

[113] Rifca Le Dieu, David C. Taussig, Alan G. Ramsay, Richard Mitter, Faridah Miraki-Moud, Rewas Fatah, Abigail M. Lee, T. Andrew Lister, and John G. Gribben. Peripheral blood T cells in acute myeloid leukemia (AML) patients at diagnosis have abnormal

phenotype and genotype and form defective immune synapses with AML blasts. *Blood*, 114(18):3909–3916, October 2009.

[114] Ken I Mills, Alexander Kohlmann, P Mickey Williams, Lothar Wieczorek, Wei-min Liu, Rachel Li, Wen Wei, David T Bowen, Helmut Loeffler, Jesus M Hernandez, Wolf-Karsten Hofmann, and Torsten Haferlach. Microarray-based classifiers and prognosis models identify subgroups with distinct clinical outcomes and high risk of AML transformation of myelodysplastic syndrome. *Blood*, 114(5):1063–1072, 2009.

[115] A Bhattacharjee, WG Richards, J Staunton, C Li, S Monti, P Vasa, C Ladd, J Beheshti, R Bueno, M Gillette, M Loda, G Weber, EJ Mark, ES Lander, W Wong, BE Johnson, TR Golub, DJ Sugarbaker, and M Meyerson. Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. *Proceedings of the National Academy of Sciences*, 98(24):13790–13795, 2001.

[116] Todd R Golub, Donna K Slonim, Pablo Tamayo, Christine Huard, Michelle Gaasenbeek, Jill P Mesirov, Hilary Coller, Mignon L Loh, James R Downing, Mark A Caligiuri, Clara D Bloomfield, and Eric S Lander. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999.

[117] Jean-Philippe Brunet, Pablo Tamayo, Todd R. Golub, and Jill P. Mesirov. Metagenes and molecular pattern discovery using matrix factorization. *Proceedings of the National Academy of Sciences*, 101(12):4164–4169, 2004.

[118] SL Pomeroy, P Tamayo, M Gaasenbeek, LM Sturla, M Angelo, ME McLaughlin, JY Kim, LC Goumnerova, PM Black, C Lau, JC Allen, D Zagzag, JM Olson, T Curran, C Wetmore, JA Biegel, T Poggio, S Mukherjee, R Rifkin, A Califano, G Stolovitzky, DN Louis, JP Mesirov, ES Lander, and TR Golub. Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, 415(6870):436–442, January 2002.

[119] Vladimir Yu Kiselev, Kristina Kirschner, Michael T Schaub, Tallulah Andrews, Andrew Yiu, Tamir Chandra, Kedar N Natarajan, Wolf Reik, Mauricio Barahona, Anthony R Green, and Martin Hamberg. SC3: consensus clustering of single-cell RNA-seq data. *Nature Methods*, 14:483–486, 2017.

[120] Andrew Butler, Paul Hoffman, Peter Smibert, Efthymia Papalexi, and Rahul Satija. Integrating single-cell transcriptomic data across different conditions, technologies, and species. *Nature Biotechnology*, 36:411–420, 2018.

[121] Liying Yan, Mingyu Yang, Hongshan Guo, Lu Yang, Jun Wu, Rong Li, Ping Liu, Ying Lian, Xiaoying Zheng, Jie Yan, Jin Huang, Ming Li, Xinglong Wu, Lu Wen, Kaiqin Lao, Ruiqiang Li, Jie Qiao, and Fuchou Tang. Single-cell RNA-seq profiling of human preimplantation embryos and embryonic stem cells. *Nature Structural & Molecular Biology*, 20:1131–1139, 2013.

[122] Mubeen Goolam, Antonio Scialdone, Sarah JL Graham, Iain C Macaulay, Agnieszka Jedrusik, Anna Hupalowska, Thierry Voet, John C Marioni, and Magdalena Zernicka-Goetz. Heterogeneity in Oct4 and Sox2 targets biases cell fate in 4-cell mouse embryos. *Cell*, 165(1):61–74, 2016.

[123] Qiaolin Deng, Daniel Ramsköld, Björn Reinius, and Rickard Sandberg. Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells. *Science*, 343(6167):193–196, 2014.

[124] Alex A. Pollen, Tomasz J. Nowakowski, Joe Shuga, Xiaohui Wang, Anne A. Leyrat, Jan H. Lui, Nianzhen Li, Lukasz Szpankowski, Brian Fowler, Peilin Chen, Naveen Ramalingam, Gang Sun, Myo Thu, Michael Norris, Ronald Lebofsky, Dominique Toppani, Darnell W. Kemp Ii, Michael Wong, Barry Clerkson, Brittnee N. Jones, Shiquan Wu, Lawrence Knutsson, Beatriz Alvarado, Jing Wang, Lesley S. Weaver, Andrew P. May, Robert C. Jones, Marc A. Unger, Arnold R. Kriegstein, and Jay A. A. West. Low-coverage single-cell mRNA sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex. *Nature Biotechnology*, 32:1053–1058, 2014.

[125] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.

[126] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[127] Morgane Pierre-Jean, Jean-François Deleuze, Edith Le Floch, and Florence Mauger. Clustering and variable selection evaluation of 13 unsupervised methods for multi-omics data integration. *Briefings in Bioinformatics*, 2019. bbz138.

[128] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

[129] David R. Cox. Regression Models and Life-Tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(2):187–220, 1972.

[130] Tin Nguyen. *Horizontal and vertical integration of bio-molecular data*. PhD thesis, Wayne State University, July 2017.

[131] Leonard Kaufman and Peter Rousseeuw. Clustering by Means of Medoids. In Yadolah Dodge, editor, *Statistical Data Analysis Based on the L1–Norm and Related Methods*, pages 405–416. North-Holland, Amsterdam, 1987.

[132] Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617, 2003.

[133] Christina Curtis, Sohrab P. Shah, Suet-Feung Chin, Gulisa Turashvili, Oscar M. Rueda, Mark J. Dunning, Doug Speed, Andy G. Lynch, Shamith Samarajiwa, Yinyin Yuan, Stefan Graf, Gavin Ha, Gholamreza Haffari, Ali Bashashati, Roslin Russell, Steven McKinney, METABRIC Group, Anita Langerød, Andrew Green, Elena Provenzano, Gordon Wishart, Sarah Pinder, Peter Watson, Florian Markowetz, Leigh Murphy, Ian Ellis, Arnie

Purushotham, Anne-Lise Børresen-Dale, James D. Brenton, Simon Tavaré, Carlos Caldas, and Samuel Aparicio. The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486(7403):346–352, 2012.

[134] Jianhua Zhang. *CNTools: Convert segment data into a region by sample matrix to allow for other high level computational analyses*, 2014.

[135] Gene Golub and William Kahan. Calculating the singular values and pseudo-inverse of a matrix. *Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis*, 2(2):205–224, 1965.

[136] Alina Beygelzimer, Sham Kakadet, John Langford, Sunil Arya, David Mount, and Shengqiao Li. *FNN: Fast Nearest Neighbor Search Algorithms and Applications*, 2019. R package version 1.1.3.

[137] Brian D Ripley. *Modern applied statistics with S*. Springer, 2002.

[138] Peter Langfelder, Bin Zhang, and Steve Horvath. Defining clusters from a hierarchical cluster tree: the Dynamic Tree Cut package for R. *Bioinformatics*, 24(5):719–720, 2008.

[139] Terry M. Therneau and Patricia M. Grambsch. *Modeling Survival Data: Extending the Cox Model*. Springer, 2000.

[140] Deena M.A. Gendoo, Natchar Ratanasirigulchai, Markus S. Schroeder, Laia Pare, Joel S. Parker, Aleix Prat, and Benjamin Haibe-Kains. *genefu: Computation of Gene Expression-Based Signatures in Breast Cancer*, 2020. R package version 2.18.1.

[141] David N Louis, Arie Perry, Guido Reifenberger, Andreas Von Deimling, Dominique Figarella-Branger, Webster K Cavenee, Hiroko Ohgaki, Otmar D Wiestler, Paul Kleihues, and David W Ellison. The 2016 world health organization classification of tumors of the central nervous system: a summary. *Acta Neuropathologica*, 131(6):803–820, 2016.

[142] Zhaonian Hao and Dongsheng Guo. Egfr mutation: novel prognostic factor associated with immune infiltration in lower-grade glioma; an exploratory study. *BMC Cancer*, 19(1):1–13, 2019.

[143] Roger Stupp, Monika E Hegi, Warren P Mason, Martin J van den Bent, Martin JB Taphoorn, Robert C Janzer, Samuel K Ludwin, Anouk Allgeier, Barbara Fisher, Karl Belanger, Peter Hau, Alba A Brandes, Johanna Gijtenbeek, Christine Marosi, Charles J Vecht, Karima Mokhtari, Pieter Wesseling, Salvador Villa, Elizabeth Eisenhauer, Thierry Gorlia, Michael Weller, Denis Lacombe, J Gregory Cairncross, and René-Olivier Mirimanoff. Effects of radiotherapy with concomitant and adjuvant temozolomide versus radiotherapy alone on survival in glioblastoma in a randomised phase iii study: 5-year analysis of the eortc-ncic trial. *The Lancet Oncology*, 10(5):459–466, 2009.

[144] Hiroko Ohgaki and Paul Kleihues. Genetic pathways to primary and secondary glioblastoma. *The American Journal of Pathology*, 170(5):1445–1453, 2007.

[145] Iqbal Unnisa Ali, Lynn M Schriml, and Michael Dean. Mutational spectra of pten/mmac1 gene: a tumor suppressor with lipid phosphatase activity. *Journal of the National Cancer Institute*, 91(22):1922–1932, 1999.

[146] Gennady Korotkevich, Vladimir Sukhov, Nikolay Budin, Boris Shpak, Maxim N. Artyomov, and Alexey Sergushichev. Fast gene set enrichment analysis. *BioRxiv*, page 060012, 2021.

[147] Nicholas J. Szerlip, Alicia Pedraza, Debyani Chakravarty, Mohammad Azim, Jeremy McGuire, Yuqiang Fang, Tatsuya Ozawa, Eric C. Holland, Jason T. Huse, Suresh Jhanwar, Margaret A. Leversha, Tom Mikkelsen, and Cameron W. Brennan. Intratumoral heterogeneity of receptor tyrosine kinases egfr and pdgfra amplification in glioblastoma defines subpopulations with distinct growth factor response. *Proceedings of the National Academy of Sciences*, 109(8):3041–3046, 2012.

[148] Isabella Gomes Cantanhede and João Ricardo Mendes de Oliveira. Pdgf family expression in glioblastoma multiforme: data compilation from ivy glioblastoma atlas project database. *Scientific Reports*, 7(1):1–9, 2017.

[149] Guiyan Xu and Jian Yi Li. Differential expression of pdgfrb and egfr in microvascular proliferation in glioblastoma. *Tumor Biology*, 37(8):10577–10586, 2016.

[150] Estefanía Carrasco-García, Miguel Saceda, and Isabel Martínez-Lacaci. Role of receptor tyrosine kinases and their ligands in glioblastoma. *Cells*, 3(2):199–235, 2014.

[151] Carlo Cenciarelli, Hany ES Marei, Manuela Zonfrillo, Pasquale Pierimarchi, Emanuela Paldino, Patrizia Casalbore, Armando Felsani, Angelo Luigi Vescovi, Giulio Maira, and Annunziato Mangiola. Pdgf receptor alpha inhibition induces apoptosis in glioblastoma cancer stem cells refractory to anti-notch and anti-egfr treatment. *Molecular Cancer*, 13(1):1–15, 2014.

[152] Roel G. W. Verhaak, Katherine A. Hoadley, Elizabeth Purdom, Victoria Wang, Yuan Qi, Matthew D. Wilkerson, C. Ryan Miller, Li Ding, Todd Golub, Jill P. Mesirov, Gabriele Alexe, Michael Lawrence, Michael O'Kelly, Pablo Tamayo, Barbara A. Weir, Stacey Gabriel, Wendy Winckler, Supriya Gupta, Lakshmi Jakkula, Heidi S. Feiler, J. Graeme Hodgson, C. David James, Jann N. Sarkaria, Cameron Brennan, AriKahn, Paul T. Spellman, Richard K. Wilson, Terence P. Speed, Joe W. Gray, Matthew Meyerson, Gad Getz, Charles M. Perou, D. Neil Hayes, and The Cancer Genome Atlas Research Network. Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in PDGFRA, IDH1, EGFR, and NF1. *Cancer Cell*, 17(1):98–110, 2010.

[153] Alan T Yeo, Hyun Jung Jun, Vicky A Appleman, Piyan Zhang, Hemant Varma, Jann N Sarkaria, and Al Charest. Egfrviii tumorigenicity requires pdgfra co-signaling and reveals therapeutic vulnerabilities in glioblastoma. *Oncogene*, 40(15):2682–2696, 2021.

[154] J Craig Venter, Mark D Adams, Eugene W Myers, Peter W Li, Richard J Mural, Granger G Sutton, Hamilton O Smith, Mark Yandell, Cheryl A Evans, Robert A Holt, et al. The sequence of the human genome. *Science*, 291(5507):1304–1351, 2001.

[155] Eric S Lander. Initial impact of the sequencing of the human genome. *Nature*, 470(7333):187–197, 2011.

[156] International Human Genome Sequencing Consortium et al. Finishing the euchromatic sequence of the human genome. *Nature*, 431(7011):931–945, 2004.

[157] Gregory A Wray. The evolutionary significance of cis-regulatory mutations. *Nature Reviews Genetics*, 8(3):206–216, 2007.

[158] David B Allison, Xiangqin Cui, Grier P Page, and Mahyar Sabripour. Microarray data analysis: from disarray to consolidation and consensus. *Nature reviews genetics*, 7(1):55–65, 2006.

[159] Charlotte Soneson and Mauro Delorenzi. A comparison of methods for differential expression analysis of RNA-seq data. *BMC Bioinformatics*, 14:91, 2013.

[160] Albert-László Barabási and Zoltán N Oltvai. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.

[161] Minoru Kanehisa, Miho Furumichi, Mao Tanabe, Yoko Sato, and Kanae Morishima. KEGG: new perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Research*, 45(D1):D353–D361, 2017.

[162] Bert Vogelstein and Kenneth W Kinzler. Cancer genes and the pathways they control. *Nature Medicine*, 10(8):789–799, 2004.

[163] Michael Stumvoll, Barry J Goldstein, and Timon W van Haeften. Type 2 diabetes: principles of pathogenesis and therapy. *The Lancet*, 365(9467):1333–1346, 2005.

[164] Peter Libby, Paul M Ridker, and Attilio Maseri. Inflammation and atherosclerosis. *Circulation*, 105(9):1135–1143, 2002.

[165] Purvesh Khatri, Marina Sirota, and Atul J Butte. Ten years of pathway analysis: current approaches and outstanding challenges. *PLoS computational biology*, 8(2):e1002375, 2012.

[166] Aravind Subramanian, Pablo Tamayo, Vamsi K. Mootha, Sayan Mukherjee, Benjamin L. Ebert, Michael A. Gillette, Amanda Paulovich, Scott L. Pomeroy, Todd R. Golub, Eric S. Lander, and Jill P. Mesirov. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences*, 102(43):15545–15550, 2005.

[167] Justin Lamb, Emily D Crawford, David Peck, Joshua W Modell, Irene C Blat, Matthew J Wrobel, Jim Lerner, Jean-Philippe Brunet, Aravind Subramanian, Kenneth N Ross, Michael Reich, Haley Hieronymus, Gue Wie, Scott A Armstrong, Stephen Haggarty, Paul Clemons, Ru Wie, Steven Carr, Eric Lander, and Todd Golub. The Connectivity Map: using gene-expression signatures to connect small molecules, genes, and disease. *Science*, 313(5795):1929–1935, 2006.

[168] The Gene Ontology Consortium. The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Research*, 49(D1):D325–D334, 2021.

[169] Tim Beißbarth and Terence P. Speed. GOstat: find statistically overrepresented Gene Ontologies within a group of genes. *Bioinformatics*, 20(9):1464–1465, June 2004.

[170] Douglas A. Hosack, Glynn Dennis Jr, Brad T. Sherman, H. Clifford Lane, and Richard A. Lempicki. Identifying biological themes within lists of genes with EASE. *Genome Biology*, 4:R70, 2003.

[171] Purvesh Khatri, Sorin Draghici, G. Charles Ostermeier, and Stephen A. Krawetz. Profiling Gene Expression Using Onto-Express. *Genomics*, 79(2):266–270, 2002.

[172] Bradley Efron and Robert Tibshirani. On testing the significance of sets of genes. *The Annals of Applied Statistics*, 1(1):107–129, 2007.

[173] Vamsi K. Mootha, Cecilia M. Lindgren, Karl-Fredrick Eriksson, Aravind Subramanian, Smita Sihag, Joseph Lehar, Pere Puigserver, Emma Carlsson, Martin Ridderstråle, Esa

Laurila, Nicholas Houstis, Mark J Daly, Nick Patterson, Jill P Mesirov, Todd R Golub, Pablo Tamayo, Bruce Spiegelman, Eric S Lander, Joel N Hirschhorn, David Altshuler, and Leif C Groop. PGC-11α-responsive genes involved in oxidative phosphorylation are coordinately downregulated in human diabetes. *Nature Genetics*, 34:267–273, 2003.

[174] Aravind Subramanian, Pablo Tamayo, Vamsi K. Mootha, Sayan Mukherjee, Benjamin L. Ebert, Michael A. Gillette, Amanda Paulovich, Scott L. Pomeroy, Todd R. Golub, Eric S. Lander, and Jill P. Mesirov. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceeding of The National Academy of Sciences*, 102(43):15545–15550, 2005.

[175] David Croft, Antonio Fabregat Mundo, Robin Haw, Marija Milacic, Joel Weiser, Guanming Wu, Michael Caudy, Phani Garapati, Marc Gillespie, Maulik R. Kamdar, Bijay Jassal, Steven Jupe, Lisa Matthews, Bruce May, Stanislav Palatnik, Karen Rothfels, Veronica Shamovsky, Heeyeon Song, Mark Williams, Ewan Birney, Henning Hermjakob, Lincoln Stein, and Peter D'Eustachio. The Reactome pathway knowledgebase. *Nucleic Acids Research*, 42(D1):D472–D477, 2014.

[176] Sorin Draghici, Purvesh Khatri, Adi L. Tarca, Kashyap Amin, Arina Done, Calin Voichita, Constantin Georgescu, and Roberto Romero. A systems biology approach for pathway level analysis. *Genome Research*, 17:1537–1545, 2007.

[177] Adi L. Tarca, Sorin Draghici, Purvesh Khatri, Sonia S. Hassan, Pooja Mittal, Jung-sun Kim, Chong J. Kim, Juan P. Kusanovic, and Roberto Romero. A novel signaling pathway impact analysis. *Bioinformatics*, 25(1):75–82, 2009.

[178] E. Glaab, A. Baudot, N. Krasnogor, and A. Valencia. TopoGSA: network topological gene set analysis. *Bioinformatics*, 26(9):1271–1272, 2010.

[179] Zuguang Gu, Jialin Liu, Kunming Cao, Junfeng Zhang, and Jin Wang. Centrality-based pathway enrichment: a systematic approach for finding significant pathways dominated by key genes. *BMC Systems Biology*, 6:56, 2012.

[180] Zuguang Gu and Jin Wang. CePa: an R package for finding significant pathways weighted by multiple network centralities. *Bioinformatics*, 29(5):658–660, 2013.

[181] Cristina Mitrea, Zeinab Taghavi, Behzad Bokanizad, Samer Hanoudi, Rebecca Tagett, Michele Donato, Călin Voichiţa, and Sorin Drăghici. Methods and approaches in the topology-based analysis of biological pathways. *Frontiers in Physiology*, 4:278, 2013.

[182] Tin Nguyen, Diana Diaz, Rebecca Tagett, and Sorin Draghici. Overcoming the matched-sample bottleneck: an orthogonal approach to integrate omic data. *Scientific Reports*, 6:29251, 2016.

[183] Tin Nguyen, Adib Shafi, Tuan-Minh Nguyen, A. Grant Schissler, and Sorin Draghici. NBIA: a network-based integrative analysis framework–applied to pathway analysis. *Scientific Reports*, 10:4188, 2020.

[184] Tuan-Minh Nguyen, Adib Shafi, Tin Nguyen, and Sorin Draghici. Identifying significantly impacted pathways: a comprehensive review and assessment. *Genome Biology*, 20:203, 2019.

[185] Minoru Kanehisa and Susumu Goto. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 28(1):27–30, 2000.

[186] Bijay Jassal, Lisa Matthews, Guilherme Viteri, Chuqiao Gong, Pascual Lorente, Antonio Fabregat, Konstantinos Sidiropoulos, Justin Cook, Marc Gillespie, Robin Haw, Fred Loney, Bruce May, Marija Milacic, Karen Rothfels, Cristoffer Sevilla, Veronica Shamovsky, Solomon Shorser, Thawfeek Varusai, Joel Weiser, Guanming Wu, Lincoln Stein, Henning Hermjakob, and Peter D'Eustachio. The Reactome pathway knowledgebase. *Nucleic Acids Research*, 48(D1):D498–D503, 2020.

[187] Thomas Kelder, Alexander R. Pico, Kristina Hanspers, Martijn P. Van Iersel, Chris Evelo, and Bruce R. Conklin. Mining biological pathways using wikipathways web services. *PloS ONE*, 4(7):e6447, 2009.

[188] Tin Nguyen, Cristina Mitrea, and Sorin Draghici. Network-based approaches for pathway level analysis. *Current Protocols in Bioinformatics*, 61(1):8–25, 2018.

[189] Douglas A. Hosack, Glynn Dennis Jr., Brad T. Sherman, H. Clifford Lane, and Richard A. Lempicki. Identifying Biological Themes within Lists of Genes with EASE. *Genome Biology*, 4(6):P4, 2003.

[190] Fátima Al-Shahrour, Ramón Díaz-Uriarte, and Joaquín Dopazo. FatiGO: a web tool for finding significant associations of Gene Ontology terms with groups of genes. *Bioinformatics*, 20(4):578–580, 2004.

[191] Da W. Huang, Brad T. Sherman, and Richard A. Lempicki. Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nature Protocols*, 4:44–57, 2009.

[192] Jing Wang, Dexter Duncan, Zhiao Shi, and Bing Zhang. WEB-based GEne SeT AnaLysis Toolkit (WebGestalt): update 2013. *Nucleic Acids Research*, 41(W1):W77–W83, 2013.

[193] Bradley Efron and Robert Tibshirani. On testing the significance of sets of genes. *The Annals of Applied Statistics*, 1(1):107–129, 2007.

[194] Zhen Jiang and Robert Gentleman. Extensions to gene set enrichment. *Bioinformatics*, 23(3):306–313, 2007.

[195] Adi L Tarca, Sorin Drăghici, Gaurav Bhatti, and Roberto Romero. Down-weighting overlapping genes improves gene set analysis. *BMC Bioinformatics*, 13:136, 2012.

[196] Sek W. Kong, William T. Pu, and Peter J. Park. A multivariate approach for integrating genome-wide expression data and biological knowledge. *Bioinformatics*, 22(19):2373–2380, 2006.

[197] Jörg Rahnenführer, Francisco S. Domingues, Jochen Maydt, and Thomas Lengauer. Calculating the Statistical Significance of Changes in Pathway Activity From Gene Expression Data. *Statistical Applications in Genetics and Molecular Biology*, 3(1), 2004.

[198] Ali Shojaie and George Michailidis. Analysis of Gene Sets Based on the Underlying Regulatory Network. *Journal of Computational Biology*, 16(3):407–426, 2010.

[199] Sharon I. Greenblum, Sol Efroni, Carl F. Schaefer, and Ken H. Buetow. The PathOlogist: an automated tool for pathway-centric analysis. *BMC Bioinformatics*, 12:133, 2011.

[200] Enrico Glaab, Anaïs Baudot, Natalio Krasnogor, Reinhard Schneider, and Alfonso Valencia. EnrichNet: network-based gene set enrichment analysis. *Bioinformatics*, 28(18):i451–i457, 2012.

[201] Alberto Luiz P. Reyes, Tiago C. Silva, Simon G. Coetzee, Jasmine T. Plummer, Brian D. Davis, Stephanie Chen, Dennis J. Hazelett, Kate Lawrenson, Benjamin P. Berman, and Michelle R. Gayther, Simon A.and Jones. GENAVi: a shiny web application for gene expression normalization, analysis and visualization. *BMC Genomics*, 20(1):1–9, 2019.

[202] Yuxing Liao, Jing Wang, Eric J. Jaehnig, Zhiao Shi, and Bing Zhang. WebGestalt 2019: gene set analysis toolkit with revamped UIs and APIs. *Nucleic Acids Research*, 47(W1):W199–W205, 2019.

[203] Liang Sun, Yongnan Zhu, ASM Ashique Mahmood, Catalina O. Tudor, Jia Ren, K. Vijay-Shanker, Jian Chen, and Carl J. Schmidt. WebGIVI: a web-based gene enrichment analysis and visualization tool. *BMC Bioinformatics*, 18(1):1–10, 2017.

[204] Glynn Dennis Jr., Brad T. Sherman, Douglas A. Hosack, Jun Yang, Wei Gao, H. Clifford Lane, and Richard A. Lempicki. DAVID: Database for annotation, visualization, and integrated discovery. *Genome Biology*, 4:R60, 2003.

[205] Jianguo Xia, Christopher D Fjell, Matthew L Mayer, Olga M Pena, David S Wishart, and Robert EW Hancock. INMEX—a web-based tool for integrative meta-analysis of expression data. *Nucleic Acids Research*, 41(W1):W63–W70, 2013.

[206] Uku Raudvere, Liis Kolberg, Ivan Kuzmin, Tambet Arak, Priit Adler, Hedi Peterson, and Jaak Vilo. g:Profiler: a web server for functional enrichment analysis and conversions of gene lists (2019 update). *Nucleic Acids Research*, 47(W1):W191–W198, 2019.

[207] Maxim V Kuleshov, Matthew R Jones, Andrew D Rouillard, Nicolas F Fernandez, Qiaonan Duan, Zichen Wang, Simon Koplev, Sherry L Jenkins, Kathleen M Jagodnik,

Alexander Lachmann, Michael G McDermott, Caroline D Monteiro, Gregory W Gundersen, and Avi Ma'ayan. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic Acids Research*, 44(W1):W90–W97, 2016.

[208] Toshiaki Tokimatsu, Nozomu Sakurai, Hideyuki Suzuki, Hiroyuki Ohta, Kazuhiko Nishitani, Tanetoshi Koyama, Toshiaki Umezawa, Norihiko Misawa, Kazuki Saito, and Daisuke Shibata. KaPPA-View. A Web-Based Analysis Tool for Integration of Transcript and Metabolite Data on Plant Metabolic Pathway Maps. *Plant Physiology*, 138(3):1289–1300, 2005.

[209] Tien-Chueh Kuo, Tze-Feng Tian, and Yufeng Jane Tseng. 3Omics: a web-based systems biology tool for analysis, integration and visualization of human transcriptomic, proteomic and metabolomic data. *BMC Systems Biology*, 7:64, 2013.

[210] Rafael Hernández-de Diego, Sonia Tarazona, Carlos Martínez-Mira, Leandro Balzano-Nogueira, Pedro Furió-Tarí, Georgios J Pappas Jr, and Ana Conesa. PaintOmics 3: a web resource for the pathway analysis and visualization of multi-omics data. *Nucleic Acids Research*, 46(W1):W503–W509, 2018.

[211] Atanas Kamburov, Rachel Cavill, Timothy MD Ebbels, Ralf Herwig, and Hector C Keun. Integrated pathway-level analysis of transcriptomics and metabolomics data with IMPaLA. *Bioinformatics*, 27(20):2917–2918, 2011.

[212] Daniel Stöckel, Tim Kehl, Patrick Trampert, Lara Schneider, Christina Backes, Nicole Ludwig, Andreas Gerasch, Michael Kaufmann, Manfred Gessler, Norbert Graf, Eckart Meese, Andreas Keller, and Hans-Peter Lenhof. Multi-omics enrichment analysis using the GeneTrail2 web service. *Bioinformatics*, 32(10):1502–1508, 01 2016.

[213] Alexey A. Sergushichev. An algorithm for fast preranked gene set enrichment analysis using cumulative statistic calculation. *BioRxiv*, page 060012, 2016.

[214] Sorin Draghici, Purvesh Khatri, Rui P. Martins, G. Charles Ostermeier, and Stephen A. Krawetz. Global functional profiling of gene expression. *Genomics*, 81(2):98–104, 2003.

[215] Alan Stuart, Steven Arnold, J Keith Ord, Anthony O'Hagan, and Jonathan Forster. *Kendall's advanced theory of statistics*, volume 1. Wiley, London, 6th edition, 1994.

[216] Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics*, 1(6):80–83, 1945.

[217] Tanya Barrett, Stephen E. Wilhite, Pierre Ledoux, Carlos Evangelista, Irene F. Kim, Maxim Tomashevsky, Kimberly A. Marshall, Katherine H. Phillippy, Patti M. Sherman, Michelle Holko, Andrey Yefanov, Hyeseung Lee, Naigong Zhang, Cynthia L. Robertson, Nadezhda Serova, Sean Davis, and Alexandra Soboleva. NCBI GEO: archive for functional genomics data sets–update. *Nucleic Acids Research*, 41(D1):D991–D995, 2013.

[218] Sean Davis and Paul S Meltzer. GEOquery: a bridge between the Gene Expression Omnibus (GEO) and BioConductor. *Bioinformatics*, 23(14):1846–1847, 2007.

[219] Michael Ashburner, Catherine A. Ball, Judith A. Blake, David Botstein, Heather Butler, J. Michael Cherry, Allan P. Davis, Kara Dolinski, Selina S. Dwight, Janan T. Eppig, Midori A. Harris, David P. Hill, Laurie Issel-Tarver, Andrew Kasarskis, Suzanna Lewis, John C. Matese, Joel E. Richardson, Martin Ringwald, Gerald M. Rubin, and Gavin Sherlock. Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25:25–29, 2000.

[220] Ronald A. Fisher. *Statistical methods for research workers*. Oliver & Boyd, Edinburgh, 1925.

[221] Samuel A. Stouffer, Edward A. Suchman, Leland C. DeVinney, Shirley A. Star, and Robin M. Williams Jr. *The American Soldier: Adjustment during army life*, volume 1. Princeton University Press, Princeton, 1949.

[222] Tin Nguyen, Rebecca Tagett, Michele Donato, Cristina Mitrea, and Sorin Draghici. A novel bi-level meta-analysis approach-applied to biological pathway analysis. *Bioinformatics*, 32(3):409–416, 2016.

[223] Leonard H. C. Tippett. *The Methods of Statistics. An Introduction mainly for Workers in the Biological Sciences*. Williams & Norgate, London, 1931.

[224] Winnie S. Liang, Travis Dunckley, Thomas G. Beach, Andrew Grover, Diego Mastroeni, Douglas G. Walker, Richard J. Caselli, Walter A. Kukull, Daniel McKeel, John C. Morris, Christine Hulette, Donald Schmechel, Gene E. Alexander, Eric M. Reiman, Joseph Rogers, and Dietrich A. Stephan. Gene expression profiles in anatomically and functionally distinct regions of the normal aged human brain. *Physiological Genomics*, 28(3):311–322, 2007.

[225] Minghui Wang, Panos Roussos, Andrew McKenzie, Xianxiao Zhou, Yuji Kajiwara, Kristen J. Brennand, Gabriele C. De Luca, John F. Crary, Patrizia Casaccia, Joseph D. Buxbaum, Michelle Ehrlich, Sam Gandy, Alison Goate, Pavel Katsel, Eric Schadt, Vahram Haroutunian, and Bin Zhang. Integrative network analysis of nineteen brain regions identifies molecular signatures and networks underlying selective regional vulnerability to alzheimer's disease. *Genome Medicine*, 8(1):1–21, 2016.

[226] Nicole C. Berchtold, David H. Cribbs, Paul D. Coleman, Joseph Rogers, Elizabeth Head, Ronald Kim, Tom Beach, Carol Miller, Juan Troncoso, John Q. Trojanowski, H. Ronald Zielke, and Carl W. Cotman. Gene expression changes in the course of normal brain aging are sexually dimorphic. *Proceedings of the National Academy of Sciences*, 105(40):15605–15610, 2008.

[227] Russell H. Swerdlow. Brain aging, Alzheimer's disease, and mitochondria. *Biochimica et Biophysica Acta (BBA)-Molecular Basis of Disease*, 1812(12):1630–1639, 2011.

[228] Aleksandra Maruszak and Cezary Żekanowski. Mitochondrial dysfunction and Alzheimer's disease. *Progress in Neuro-Psychopharmacology and Biological Psychiatry*, 35(2):320–330, 2011.

[229] Xiongwei Zhu, George Perry, Mark A Smith, and Xinglong Wang. Abnormal mitochondrial dynamics in the pathogenesis of Alzheimer's disease. *Journal of Alzheimer's Disease*, 33(S1):S253–S262, 2013.

[230] Henry W Querfurth and Frank M LaFerla. Mechanisms of disease. *New England Journal of Medicine*, 362(4):329–344, 2010.

[231] Y. Benjamini and Y. Hochberg. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of The Royal Statistical Society B*, 57(1):289–300, 1995.

[232] Matthew E. Ritchie, Belinda Phipson, Di Wu, Yifang Hu, Charity W. Law, Wei Shi, and Gordon K. Smyth. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Research*, 43(7):e47–e47, 2015.

[233] T. Barrett, T. O. Suzek, D. B. Troup, S. E. Wilhite, W. C. Ngau, P. Ledoux, D. Rudnev, A. E. Lash, W. Fujibuchi, and R. Edgar. NCBI GEO: mining millions of expression profiles–database and tools. *Nucleic Acids Research*, 33(Database Issue):D562–D566, 2005.

[234] Alexander Lachmann, Denis Torre, Alexandra B Keenan, Kathleen M Jagodnik, Hoyjin J Lee, Lily Wang, Moshe C Silverstein, and Avi Ma'ayan. Massive mining of publicly available RNA-seq data from human and mouse. *Nature Communications*, 9(1):1366, 2018.

[235] Benilton S Carvalho and Rafael A Irizarry. A framework for oligonucleotide microarray preprocessing. *Bioinformatics*, 26(19):2363–2367, 2010.

[236] Rafael A Irizarry, Benjamin M Bolstad, Francois Collin, Leslie M Cope, Bridget Hobbs, and Terence P Speed. Summaries of Affymetrix GeneChip probe level data. *Nucleic Acids Research*, 31(4):e15–e15, 2003.

[237] Michael I. Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biology*, 15:550, 2014.

[238] GTEx Consortium. The gtex consortium atlas of genetic regulatory effects across human tissues. *Science*, 369(6509):1318–1330, 2020.

[239] Yuqing Zhang, Giovanni Parmigiani, and W Evan Johnson. Combat-seq: batch effect adjustment for rna-seq count data. *NAR genomics and bioinformatics*, 2(3):lqaa078, 2020.

[240] Daniel J. Wilson. The harmonic mean p-value for combining dependent tests. *Proceedings of the National Academy of Sciences*, 116(4):1195–1200, 2019.

[241] Janet Piñero, Juan Manuel Ramírez-Anguita, Josep Saüch-Pitarch, Francesco Ronzano, Emilio Centeno, Ferran Sanz, and Laura I Furlong. The DisGeNET knowledge platform for disease genomics: 2019 update. *Nucleic Acids Research*, 48(D1):D845–D855, 2020.