AI-POWERED SYSTEMS BIOLOGY MODELS TO STUDY HUMAN DISEASE

by

Wennan Chang

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

School of Electrical and Computer Engineering

West Lafayette, Indiana

May 2022

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Chi Zhang, Co-chair

Department of Electrical and Computer Engineering

Dr. Mireille Boutin, Co-chair

Department of Electrical and Computer Engineering

Dr. Zina Ben Miled

Department of Electrical and Computer Engineering

Dr. Edward J. Delp

Department of Electrical and Computer Engineering

Approved by:

Dr. Dimitrios Peroulis

To my parents,

for their unconditional support, inspiration, and encouragement;

To my wife, Yue Fang,

for her endless trust and love.

ACKNOWLEDGMENTS

There are many people that have earned my gratitude for their help, advice, and devotion to my PhD study. Firstly, I would like to express my deepest gratitude to my advisor, Prof. Chi Zhang for his full support, expert guidance and encouragement through my research. He is much more than an academic advisor, a mentor and a colleague to me. Over the years, he has not only provided me with the best learning and research environment, enlightening advises, and numerous great opportunities, but also helped me open my eyes to different research areas, build up my confidence and self-esteem and lay down my career path. This thesis would not have been possible without his support and encouragement. I feel proud and privileged to have worked with him.

My special thanks go to my committee members Prof. Mireille Boutin (Mimi), Prof. Zina Ben Miled, and Prof. Edward J. Delp for their invaluable assistance, feedback, criticisms, and advises at different stages of my graduate study. I am grateful for their patience and insightful suggestions.

I wish to thank Sherrie Tucker, Matt Golden, and Elisheba Van Winkle from ECE department who helps me a lot during my graduate study.

I also own special thanks to Prof. Sha Cao who supported me in various ways. I appreciate her engagement with all details of my research project. She has contributed for my research meeting, proposed thinking from different perspectives that I did not come up with, and performed paper revisions using countless hours. I would like to thank my friends who supported me: Changlin Wan, Chengguang Xu, Zhi Huang, Juan Antonic Chone Chie, Xiaoyu Xiang, Shijian Gao, Shili Sheng, Jiangpeng He, Runyu Mao, Sige Hu, Qingyu Yang, Zhuo Yang, Cho-Hsin Tsai; my fellow colleagues in the CCBB center: Siyuan Qi, Pengtao Dang, Tingbo Guo, Kamon So, Haiqi Zhu, Szu-Wei Tu, Dr. Norah Alghamdi, Xiaoyu Lu, Xingyu Zhou, Dr. Steven X Chen.

I would like to express my deep gratitude to my parents and my older sister. During the time of my graduate study, they had suffered the biggest pains physically and emotionally, but they freed me from all the responsibilities and instead held all the weights on their own shoulders. I hope all my years of concentration and hard work could make you a little bit

satisfied and proud for me. I also want to thank my father-in-law and mother-in-law for their belief in me.

Finally, I want to give my heartfelt and deepest gratitude to my wife, Yue Fang. She supports me, encourages me and backs me up. She always helps clear up the obstacles to enable me can focus on my research. I am grateful for her love and supports.

This work was partially supported by National Science Foundation Div of Information & Intelligent System (No. 1850360); National Institute of General Medical Sciences (R01 award #1R01GM131399-01); Indiana Clinical; Transnational Sciences Institute Showalter Young Investigator Award; and American Cancer Society Research Scholar Award RSG-22-062-01-MM.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

μ	mean value
σ^2	variance
E	expectation
$\mathcal{N}$	Gaussian distribution
$\boldsymbol{\pi}_k$	the mixing parameter
K	the cluster number

ABBREVIATIONS

ACC	accuracy rate
ARI	adjust rand index
BIC	Bayesian information criterion
CAT	Component-wise Adaptive Trimming method
CCLE	Cancer Cell Line Encyclopedia
CEM	Classification-Expectation-Maximization
CPM	counts per million mapped reads
CSMR	Component-wise Sparse Mixture Regression
EM algorithm	Expectation Maximization algorithm
FBA	Flux Balance Analysis
FMGR	finite mixture Gaussian regression
FPKM	Fragments per kilo base of transcript per million mapped fragments
GDC	the Genomics Data Commons data portal
GEO	Gene Expression Omnibus
KEGG	the Kyoto Encyclopedia of Genes and Genomes database
LASSO	least absolute shrinkage and selection operator
LTMG	left truncated mixture Gaussian
MSE	mean squared error
NGS	Next-generation sequencing
OLS	ordinary least square
PCC	Pearson correlation coefficient
PCE	error of predicted coefficient
RI	rand index
RMs	reprogrammed metabolisms
RMSE	root mean square error
scFEA	single-cell Flux Estimation Analysis
scRNA-seq	single cell RNA sequencing
spRNA-seq	spatial RNA sequencing

SRMR	Spatial Robust Mixture Regression
ST	spatial transcriptomics
TCGA	The Cancer Genome Atlas
TPM	transcripts per million
UMAP	Uniform Manifold Approximation and Projection
UMIs	unique molecular identifiers

ABSTRACT

The fast advancing of high-throughput technology has reinforced the biomedical research ecosystem with highly scaled and commercialized data acquisition standards, which provide us with unprecedented opportunity to interrogate biology in novel and creative ways. However, unraveling the high dimensional data in practice is difficult due to the following challenges: 1) how to handle outlier and data contaminations; 2) how to address the curse of dimensionality; 3) how to utilize occasionally provided auxiliary information such as an external phenotype observation or spatial coordinate; 4) how to derive the unknown non-linear relationship between observed data and underlying mechanisms in complex biological system such as human metabolic network.

In sight of the above challenges, this thesis majorly focused on two research directions, for which we have proposed a series of statistical learning and AI-empowered systems biology models. This thesis separates into two parts. The first part focuses on identifying latent low dimensional subspace in high dimensional biomedical data. Firstly, we proposed CAT method which is a robust mixture regression method to detect outliers and estimate parameter simultaneously. Then, we proposed CSMR method in studying the heterogeneous relationship between high dimensional genetic features and a phenotype with penalized mixture regression. At last, we proposed SRMR which investigate mixture linear relationship over spatial domain. The second part focuses on studying the non-linear relationship for human metabolic flux estimation in complex biological system. We proposed the first method in this domain that can robustly estimate flux distribution of a metabolic network at the resolution of individual cells.

1. INTRODUCTION

Artificial Intelligence (AI) has been facilitating revolutions in various fields such as economics, agriculture sciences, engineering, entertainment, and biomedical sciences. It is no exaggeration to say that human being could not live without AI now. Without exception, AI has enabled many new capabilities in biomedical sciences. For example, AlphaFold, as a deep learning based program developed by Google's DeepMind, got champion twice in protein structure prediction Olympics [1]. In the light of new technological developments, the use of AI in biomedical research is paving the road to precision medicine to cure human disease such as Alzheimer's disease and cancer [2].

We now live in the age of "big data", an era in which we have the capacity to collect enormous bank of information. On one hand, the huge amount of data sets accumulated in the past years provide good training and testing resources for AI development. On the other hand, dramatically increased computational power boosts the development of new AI frameworks. In biomedical research, it is possible to sequence an entire genome in fraction of that time - just one day- due to the advent of next generation sequencing (NGS). Based on the complete genetic picture including gene expression, gene fusions, splicing variants, mutations/indels, the doctor can make personal plan of precision medicine. treatment plan for every patient. What's more, NGS still evolving rapidly and recent advancements such as single cell RNA sequencing (scRNA-seq) and spatial transcriptomic (ST) have been achieving significant progress in biomedical research in the last decade [3].

In this thesis, to solve two major challenges presented in Section 1.2, we designed several systems biology models. This thesis contains two major parts. The first part focuses on identifying low dimensional latent subspace which represents disease heterogeneity and disease subtypes. The second part focuses on establishing a systems biology model for human metabolic flux estimation based on deep learning. In this introduction, we will first briefly introduce the NGS technologies and the biomedical datasets utilized in this thesis. We will further present the current challenges and key contributions of this thesis, and then summarize the relationship between two parts.

1.1 Next Generation Sequencing Data

A Brief History of NGS

The sequencing field time point could back to 1953 when Watson and Crick discovered double-helix DNA structure [4]. After that, scientists have devoted a huge amount of effort to propose the first DNA sequencing method, called Sanger sequencing [5]. At that time, scientists could sequence only a few base pairs per year with this fragment-cloning method. With the unremitting effort of scientists and industrial researchers, Next-generation sequencing (NGS) merged in late 1970 [6] and dominated for four decades. NGS represents deep, high-throughput, in-parallel DNA sequencing technologies. The major difference of NGS and Sanger sequencing is millions to billions of DNA nucleotide can be sequenced in parallel, yielding substantially more throughput and minimizing the time and cost [7].

In the golden age of NGS, there are several milestones we should memory. In 1990, The Human Genome Project formally began, involving the US, UK, France, Germany, Japan, China and India [8]. The estimated finish time is 15 years. In 2000, a complete human genome was finished by the Human Genome Project, thanks to advances of sequencing analysis in the genomics field. In 2000, the National Center for Biotechnology Information (NCBI) created a worldwide resource for gene expression studies, named the Gene Expression Omnibus (GEO) [9]. In 2003, ENCODE Project targeted to identify all functional elements in the human genome was released [10]. In 2005, The Cancer Genome Atlas (TCGA) project begun, which applies high-throughput genome analysis technique to improve the ability to diagnose, treat, and prevent cancer. It was supervised by the National Cancer Institues's Center for Cancer Genomics and National Human Genome Research Institute (NHGRI) [11]. In 2010, the National Institutes of Health (NIH) launched the GTEx project in that the objective is strengthening understanding the role of non-coding variants in tissue-specific contexts [12]. In 2019, NIH reported that the prices of sequencing a complete human genome was 942, beating Moore's Law prediction [13] (Figure 1.1). Nowadays, "Omics" is a widely used term for describing high throughput cataloging and/or analysis of cellular molecules [14].

Rapid progress in NGS technology also boost other fields such as bioinformatics and transcriptome. Gene expression microarrays [15] dominated genomewide profiling during late 2000 before RNA-seq emerged. Microarrays is a hyridzation-based approach which profiles predefined transcripts/genes through hybridization. Its principle brings several natural issues which are not easily to overcome. These include varying background noise, requirements for high RNA amounts, dependence of annotated probe sets included on the array, and lack of precise quantification. Conversely, RNA-seq [16] allows comprehensive qualitative and full sequencing of the whole transcriptome. RNA-seq's advantages for examining trasncriptome fine structure such as the detection of novel transcript, allele-specific expression and splice junctions [17], which makes RNA-seq is a replacement of microarrays in the past long period of time.

Single cell RNA sequencing (scRNA-seq)

In recent years, single cell RNA sequencing (scRNA-seq) technology has been achieving significant influence ranging from cancer biology, stem cell biology to immunology [18]–[21]. Traditionally, to measure molecular states, bulk RNA-seq methods take average of signal values from millions of cells in specific tissue. Although these bulk methods enable large sample number due to low cost of sequencing, they have a low resolution to reveal the inside of tissue. Thus they overlook the differences in cell population and treat cell homogeneous.

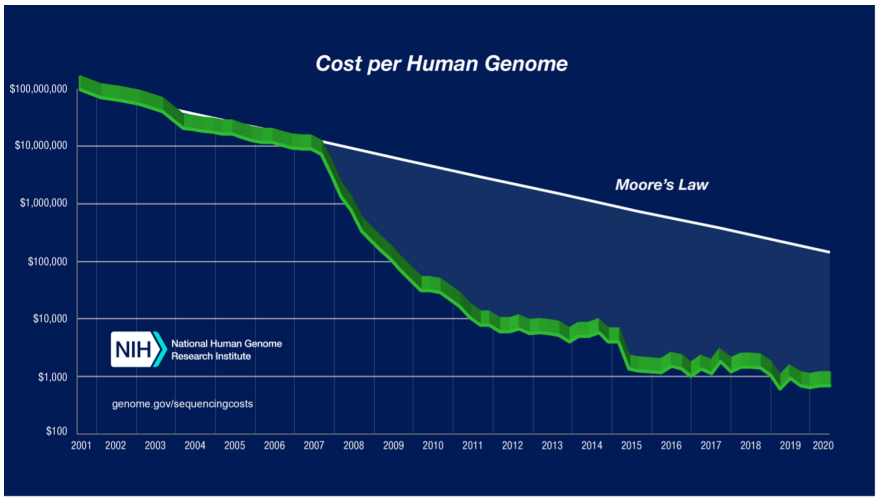


Figure 1.1. Morre's Law Prediction.

However, this could misrepresent signals of interest [22], [23]. To solve this problem, scRNA-seq technology are developed [24] and provides an opportunity to analysis the composition of tissues/organs and the diversity of cellular states, as well as to detect rare cell types [25]. With the clustering of cell types and measurement of gene expression distribution, we now have the capability to characterize subpopulation structure, understand disease progression and mechanisms of transcription regulation [26]–[28].

Smart-seq2 [29] is the first frequently-used scRNA-seq platform. Smart-seq2 is a plate-based full-length method which detected more genes in a cell, especially low abundance transcripts as well as alternatively spliced transcripts [30]. Until 10X Genomics Chromium (10X Genomics) emerged and triggered a rapid adoption of this revolutionized technology, it has been broadly applied in translational or clinical research [31]. 10X Genomics is a droplet-based scRNA-seq technology, allowing genome-wide expression profiling for thousands of cells at once. The number of unique molecular identifiers (UMIs) is considered as a direct presentation of gene expression level. However, droplet-based scRNA-seq technology has the distinctive feature of data is the increase sparsity, where data has a fraction of observed "zeros" [32]. These observed zeros can be due to biological fluctuations in the measured trait or technical limitation related to challenges in quantifying small numbers of molecules. The word *dropout* has been previously used to describe both biological and technical observed zeros, and thus led to a series of methods called scRNA-seq imputation to mitigate this issue [33]–[37].

ScRNA-seq provides us the capability to depict the heterogeneity of single cell population. However, scRNA-seq data does not provide insight into upstream regulatory networks or downstream functional consequences. Recently, integrated single cell multimodal omics is emerging as a hot topic for providing a comprehensive characterization in different aspect at different genome level.

Spatial RNA sequencing (spRNA-seq)

One major issue of scRNA-seq is that it loses critical spatial information, which negatively impacts the understanding of cell functionality and pathological changes [38]. The reason is the step of isolation of single cells during the necessary tissue dissociation step of scRNA-seq destroys information on their spatial localization within native tissue and their proximities to

each other. To circumvent this situation and study gene expression spatially, spRNA-seq was developed recently [39]. Specially, Nature Methods named spatially resolved transcriptomics as the method of the Year in 2021 [40]. Currently, 10X Genomics is the most widely used spatial transcriptomics technology in biomedical research due to its affordable cost and standard procedure [41]. Further, spRNA-seq enable the traditional research field of cell-cell interactions/communications elevated from a simple ligand-receptor level to a higher resolution level [42]–[45].

1.2 Challenges and Objectives

High throughput omics technologies are revolutionizing the fields of biological and biomedical sciences, providing new capabilities to tackle unsolved biological problems. Recently, a simultaneous profiling of multi-omics data of genome, transcriptome, epigenome, proteome, and metabolome from same samples through different conditions enable the inferences of biological functions and relations. Such data enables the study of the mathematical representation form of the biological mechanism or phenotypic features in multi-omics data. As the basic unit in transcriptional regulation, a gene is regulated by a wide range of mechanisms that are used by cells to react to environmental stimuli or phenotypic demands. Thus, functionally dependent genes are regulated together to meet the functional demand. This mechanism has been well captured by gene co-expression analysis, by which strong linear correlations of the expression of functionally dependent genes have been common observed. Gene co-expression does not only exist in human normal development [46] but also plays an important role in disease conditions such as cancer [47]. In addition, identifying linear dependence in the high dimensional biomedical data is crucial for biomarker identification, biological mechanism discovery, and guiding patient grouping to optimize clinical treatment [48]–[50].

In short, our first part is subspace learning in biomedical data with existing of (1) outliers, (2) curse of dimensionality, (3) supervised external variable, (4) extra spatial information. Unraveling the high dimensional data into contextual explainable subspaces is an important problem in multiple fields. In biomedical research, identifying intrinsic subspace

enable us the capability to explorer the hidden disease subtypes and understand the inherent heterogeneity of pathogenesis. To enable a comprehensive characterization of the information conceived in high dimensional biomedical data in studying disease subtypes, we develop a series of new algorithms to solve **four challenges: 1) eliminating the influence of outliers and noise; 2) discovering multiple views of the biomedical data.** Biological omics data collect thousands of molecular variables from a set of samples (biospecimens). One property of biological omics data is that the sample group and the underlying covariance structure could be drastically changed based on different sets of molecular features, and they tend to be aggregated differently depending on the different views of the molecular features; **3) detecting the sample subgroups if the optional external variables provided.** Thus, sample group derived by using an unsupervised approach is unnecessarily biologically or clinically meaningful. An interesting problem is raised here, which is to identify sample subgroups under the guidance of external information to maximize the contextual meaningfulness; **4) identify subgroups of correlated molecular features with spatial constraints, to simultaneously handle the spatial nonstationarity, local homogeneity, and outlier contaminations.**

The second part of this theses focus on solving a non-linear system, namely metabolism. Human metabolism is the process by which cells and organisms obtain nutrients and energy to perform their functions. Many human diseases, including cancer, diabetes, obesity, and metabolic disorder have been associated with altered metabolism. Understanding these metabolic alterations at a systems level will help us to design a more efficient therapies and treatments plan [51]. In human metabolism study, Michaelis-Menten kinetics formula is a core mathematical model [52]. This model named after German biochemist Leonor Michaelis and Canadian physician Maud Menten [53]. As one of the best-known models of enzyme kinetics, this model describes the rate of enzymatic reactions, by relating reaction rate of formation of product to the concentration of a substrate. Due to the non-linear dependence between rate of product formation and substrate concentration, traditional linear model cannot handle related estimation. However, there is a lack of computational model for human metabolic flux estimation due to several difficulties. Thus, our second part is that proposing an efficient computational model for human metabolic flux estimation.

In summary, **computational challenges to estimate human metabolic flux arise from the following aspects: 1) multiple key factors determine cell's metabolic states, including exogeneous nutrient availability, leading to the discrepancy of cell type specific markers and metabolic phenotypes and states; 2) the whole metabolic network is of high complexity, hence a proper computational reduction and reconstruction of the network is needed to reach a balance between resolution of metabolic state characterization and computational feasibility; 3) the intricate non-linear dependency between transcriptomic expressions and metabolic reaction rates calls for a more sophisticated model to fully capitulate the relationships; and 4) alternative enzymes share such common effect to the metabolic flux change remains largely unknown.**

1.3 Outline

The rest of the thesis is divided into two parts.

The first part includes Chapters 2, 3, 4, and 5, and this part focuses on subspace learning with mixture model. For this part, first in Chapter 2 a comprehensive literature review on mixture model is presented, and the preliminaries of mixture model and related derivation is provided. In addition, the notations and evaluation metrics are discussed.

- In Chapter 3, we describe CAT algorithm which consider simultaneous outlier detection and robust parameter estimation to minimize the effect of outlier contamination.
- In Chapter 4, we describe CSMR algorithm which using an external variable to deal with the challenges in studying the heterogeneous relationships between high dimensional genetic features under supervised scheme.
- In Chapter 5, we describe SRMR algorithm which investigate mixture linear relationship over spatial domain.

The second part focuses on human metabolism. This part includes Chapter 6. In Chapter 6, we describe scFEA method which is the first capability to estimate human metabolic flux at single cell resolution.

Finally, in Chapter 7, the conclusions are presented and several future research directions are proposed.

1.4 Key Contributions

1.4.1 Subspace Learning

In part I, to identify the latent space in high dimensional biomedical data, we proposed a set of mixture model based methods to handle three challenges in inferring subspace from high dimensional biological omics data: (1) sample-wise outliers, (2) integration of external demographic or phenotypic information of samples, (3) spatial information associated with each sample.

First, we focus on solving the sample-wise outliers in subspace learning. Noted, parameter estimation of mixture regression model using the expectation maximization (EM) algorithm is highly sensitive to outliers. We proposed a fast and efficient robust mixture regression algorithm, called Component-wise Adaptive Trimming (CAT) method. In this method, we considered simultaneous outlier detection and robust parameter estimation to minimize the effect of outlier contamination. Based on the framework of classification expectation maximization (CEM), we implemented CAT algorithm. Under the framework of CEM, we derived a novel definition of outliers which has been derived in a natural way. This method has 3 key contributions:

1) We proposed a novel method which simultaneously identify outliers and estimate parameters of mixture regression model, in a component-wise and adaptive way.

2) By introducing classification expectation maximization (CEM) framework to Finite Mixture Gaussian Regression (FMGR), we provided a platform that migrates the robustness issue from mixture regression to (single component) linear regression, for which LTS estimators have been extensively studied.

3) We provided a natural definition of outlier which transforms the robustness issue from a mixture model to its K linear regression components, thus the robustness issue could be much easily handled give the tremendous amount of research conducted for robust linear regression.

Then, our interest moves to supervised clustering of high dimensional data because the availability of external variable as auxiliary information. We proposed a novel supervised clustering algorithm using penalized mixture regression, called Component-wise Sparse Mixture Regression (CSMR), to deal with the challenges in studying the heterogeneous relationships between high dimensional genetic features and a phenotype. The algorithm was adapted from the classification expectation maximization algorithm, which offers a novel supervised solution to the clustering problem, with substantial improvement on both the computational efficiency and biological interpretability. This method has 4 key contributions:

1) Detecting genetic markers associated with phenotype is crucial; however, existing predictive models have been challenged by disease heterogeneity. While unsupervised learning can deal with heterogeneity, the defined clusters may not necessarily relate to the phenotype of interest.

2) We proposed a supervised clustering algorithm based on the regularized mixture regression model, which handles the high dimensional genetic features, and greatly improved the computational efficiency over others. Specifically, it efficiently performs clustering, feature selection and hyperparameter tuning in the same process.

3) Evaluation on both simulated datasets and a real-world dataset for 500 cell lines and 24 drugs demonstrated the superior performance of our algorithm over the others. Particularly, our algorithm is powerful in recapitulating the distinct subgroups hidden in the pool of cell lines with regards to their comping mechanisms to different drugs.

4) Our algorithm represents a big data analysis tool with the potential to resolve the complexity of translating the clinical representations of the disease to the real causes underpinning it. It has special relevance in the growing field of personalized medicine.

Finally, as emergence of spatial transcriptomic data, we try to extend our novel mixture model by combining spatial domain. We propose a Spatial Robust Mixture Regression model (SRMR) to investigate the relationship between a a response variable and a set of explanatory variables over the spatial domain, assuming that the relationships may exhibit complex spatially dynamic patterns that cannot be captured by constant regression coefficients. This method has 3 key contributions:

(1) We developed the very first computational concept of spatially dependent mixture regression analysis.

(2) We provided the SRMR model that efficiently solves the spatially dependent mixture regression problem, which is also empowered by a statistical inference approach to assess regression significance.

(3) SRMR enables a new type of spatial segmentation analysis to detect overlapped spatial regions of varied dependencies among subset of features, which have high contextual meaningfulness.

1.4.2 Computational Modeling of Metabolic Flux

In part II, we developed a novel computational model, namely **s**ingle-**c**ell **F**lux **E**stimation **A**nalysis (scFEA) to estimate the relative rate of metabolic flux at single cell resolution from scRNA-seq data. To the best of our knowledge, scFEA is the first capability to predict whole human metabolic flux at the single cell resolution.

Specifically, scFEA can effectively solve the above challenges with the following computational innovations: (i) an optimization function derived based upon a probabilistic model to consider the flux balance constraints among a large number of single cells with varied metabolic fluxomes, (ii) a metabolic map reduction approach based on network topology and gene expression status, (iii) a multi-layer neural network model to capture the non-linear dependency of metabolic flux on the enzymatic gene expressions, and (iv) a novel neural network architecture and solution to maximize the overall flux balance of intermediate substrates throughout all cells. In addition, both user friendly python package and online server are available for either bioinformatics researcher or biology background researcher without programming experience.

1.4.3 Biologically Explanation Representational Learning of Biomedical Omics Data

Subspace learning and human metabolic flux estimation represent two different research topics and have different problem definitions and application. The major difference appears in below respects:

1. The two topics have different assumption where subspace learning focuses on linear latent space identification while human metabolic flux estimation focuses on non-linear relationship between gene expression and involved metabolic modules.

2. Different data representation form such as low rank decomposition, linear/nonlinear dependency between certain features may suggests different biological mechanisms. For subspace learning topic, we proposed several statistical models to identify the latent space in biomedical data. For human metabolic flux topic, we proposed a machine learning method to estimate the cell-wise human metabolic flux rate where the genes involved the metabolic reaction and metabolic activity level have a nonlinear relationship.

3. Although two topics both interested in discovering mechanism of disease, human metabolic flux estimation could provide us more insight from a metabolism aspect while subspace learning emphasizes phenotype and gene biomarker.

However, the underlying reasons that drove the formation and design of these two topics are same, which form the main theme of this thesis. In general, both topics focus on systems biology models and utilize AI empowered data mining approaches to directly study biological mechanisms from omics data. For both topics, we have 1) proposed a mathematical model, 2) given a rational solution for proposed mathematical model, 3) provided biological explanation or biological discovery based on the identified patterns in biomedical data.

In particular, there is a significant analogy between two topics as we have proposed system biology models of real biological processes in both topics. Moreover, methods developed in the two topics focused on providing biologically meaningful explanations to different features collected in same omics data types. For example, the subspace learning approaches could take bulk data, single cell transcriptomics data and spatial transcriptomics data as source. And the metabolic flux estimation also uses single cell transcriptomics data or bulk data.

With additional spatial transcriptomics data, metabolic flux analysis could provide more understanding considering cell-cell interaction. In addition, both two topics significantly boost the development of computational biology, provide a capability to analysis huge accumulated biomedical data sets. Till 03/2022, our developed R package "RobMixReg" under subspace learning topic and python package "scFEA" under human metabolic flux estimation topic have been downloaded around 10,000 times totally. In summary, these two topics contribute to the community and pave the way for further exploring in human disease.

1.5 Related Publications

1.5.1 Major Publications

1. **W. Chang**, X. Zhou, Y. Zang, C. Zhang, S. Cao, "Component-wise Adaptive Trimming For Robust Mixture Regression", *arXiv preprint arXiv:2005.11599*, 2020. DOI: https://doi.org/10.48550/arXiv.2005.11599.

2. **W. Chang**, C. Wan, Y. Zang, C. Zhang, S. Cao, "Supervised clustering of high-dimensional data using regularized mixture modeling", *Briefings in bioinformatics*, 22(4), p. 291, 2021. DOI: https://doi.org/10.1093/bib/bbaa291.

3. **W. Chang**, P. Dang, C. Wan, X. Lu, Y. Fang, T. Zhao, Y. Zang, B. Li, C. Zhang, S. Cao, " Spatially and Robustly Hybrid Mixture Regression Model for Inference of Spatial Dependence", *IEEE International Conference on Data Mining (ICDM)*, pp. 31-40, 2021. DOI: https://doi.org/10.1109/ICDM51629.2021.00013.

4. **W. Chang**, C. Wan, C. Yu, W. Yao, C. Zhang, S. Cao, "RobMixReg: an R package for robust, flexible and high dimensional mixture regression", *bioRxiv*, 2020. DOI: https://doi.org/10.1101/2020.08.02.233460.

5. N. Alghamdi, **W. Chang**$^{+}$, P. Dang, X. Lu, C. Wan, S. Gampala, Z. Huang, J. Wang, Q. Ma, Y. Zang, M. Fishel, S. Cao, C. Zhang, "A graph neural network model to estimate cell-wise metabolic flux using single-cell RNA-seq data", *Genome research*, 31(10), pp.1867-1884, 2021. DOI: https://doi.org/10.1101/gr.271205.120.

6. Z. Zhang, **W. Chang**[+], N. Alghamdi, M. Fei, C. Wan, A. Lu, Y. Zang, Y.Xu, W. Wu, S. Cao, Y. Zhang, C. Zhang, "scFLUX: a web server for metabolic flux and variation prediction using transcriptomics data", under revision at *Nucleic Acids Research*, 2022.

7. Y. Zhou, **W. Chang**[+], X. Lu, J. Wang, C. Zhang, Y. Xu, "Acid-Base Homeostasis and Implications to the Phenotypic Behaviors of Cancer", accepted by *Genomics, Proteomics and Bioinformatics*, 2022. DOI: https://doi.org/10.1101/2022.03.04.482927.

8. Z. An, **W. Chang**[+], H. Zhu, C. Zhang, Y. Xu, "Subcutaneous mouse tumors do not resemble human cancers in their immune response and nutrient consumption", under revision at *National Science Review*, 2021.

1.5.2 Collaborative Publications

1. **W. Chang**, C. Wan, X. Lu, S. Tu, Y. Sun, X. Zhang, Y. Zang, A. Zhang, K. Huang, Y. Liu, X. Lu, "ICTD: A semi-supervised cell type identification and deconvolution method for multi-omics data", *bioRxiv*, 2019. DOI: https://doi.org/10.1101/426593.

2. C. Wan, **W. Chang**, Y. Zhang, F. Shah, S. Cao, X. Chen, M. Fishel, Q. Ma, C. Zhang, "LTMG: a novel statistical modeling of transcriptional expression states in single-cell RNA-Seq data", *Nucleic acids research* 47(18), e111-e111, 2019. DOI: https://doi.org/10.1093/nar/gkz655.

3. X. Lu, S. Tu, **W. Chang**, C. Wan, J. Wang, Y. Zang, B.Ramdas, R.Kapur, Lu, X., Cao, S. and Zhang, C., 2021. SSMD: a semi-supervised approach for a robust cell type identification and deconvolution of mouse transcriptomics data. *Briefings in bioinformatics*, 22(4), p.bbaa307. DOI: https://doi.org/10.1093/bib/bbaa307.

4. C. Wan, **W. Chang**, T. Zhao, M. Li, S. Cao, C. Zhang, "Fast and efficient boolean matrix factorization by geometric segmentation", *In Proceedings of the AAAI Conference on Artificial Intelligence*, 34(4), pp. 6086-6093, 2020. DOI: https://doi.org/10.1609/aaai.v34i04.6072.

↑+ is co-first author.

5. C. Wan, **W. Chang**, T. Zhao, S. Cao, C. Zhang, "Geometric All-Way Boolean Tensor Decomposition", *Advances in Neural Information Processing Systems*, 33, pp.2848-2857, 2020. DOI: https://doi.org/10.48550/arXiv.2007.15821.

6. C. Wan, **W. Chang**, T. Zhao, S. Cao, C. Zhang, "Denoising Individual Bias for Fairer Binary Submatrix Detection", *In Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, pp. 2245-2248, 2020. DOI: https://doi.org/10.1145/3340531.3412156.

7. C. Wan, D. Jia, Y. Zhao, **W. Chang**, S. Cao, X. Wang, C. Zhang,"A data denoising approach to optimize functional clustering of single cell RNA-sequencing data", *In 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 217-222, 2020. DOI: https://doi.org/10.1109/BIBM49941.2020.9313483.

8. Y. Zhang, C. Wan, P. Wang, **W. Chang**, Y. Huo, J. Chen, Q. Ma, S. Cao, C.Zhang, "M3S: a comprehensive model selection for multi-modal single-cell RNA sequencing data", *BMC bioinformatics*, 20(24), pp.1-5, 2019. DOI: https://doi.org/10.1186/s12859-019-3243-1.

9. J. Wang, F. Cheng, J. Tedrow, **W. Chang**, C. Zhang, A. Mitra, "Modulation of immune infiltration of ovarian cancer tumor microenvironment by specific subpopulations of fibroblasts", *Cancers*, 12(11), p.3184, 2020. DOI: https://doi.org/10.3390/cancers12113184.

Part I

Mixture Model based Subspace Learning of in Biomedical Data

2. PRELIMINARIES OF SUBSPACE LEARNING

2.1 Literature Reviews

2.1.1 Robust Mixture Regression

Finite Mixture Gaussian Regression (FMGR) model was first introduced by Goldfeld et al [54], and has been widely used to explore the latent relationship between a response and independent variables in many fields [55]–[60]. Parameter estimation in FMGR is usually conducted through maximum likelihood using expectation maximization (EM) algorithm assuming normally distributed component errors, which is vulnerable to outliers or heavy-tailed noises. Many algorithms have been developed to estimate the FMGR parameters robustly [61]. Using the idea of weighted regression, Markatou [62] and Shen et al. [63] proposed using a weight factor for each data point to robustify the estimation procedure. By modifying the M-step in the EM algorithm, Bai et al. [64] replaced the least squares criterion in M step with a robust bi-square criterion (**MIXBI**); Bashir and Carter [65] adopted the idea of the S-estimator to mixture of linear regression; Song et al. [66] proposed using Laplace distribution to model the error distribution (**MIXL**); Yao et al. [67] extended the idea of mixture of t-distributions proposed by Peel and McLachlan [68] from clustering to the regression setting (**MIXT**). These methods seek for robust parameter estimation in the presence of outliers, however, the outliers may still corrupt the robust algorithms, and the identities of the outliers still remain unknown unless further screening steps are taken. The identities of the outliers are often interesting for two reasons: firstly, removal of the outliers could reduce their effect on the estimators and improve the estimation accuracy; secondly, for practical reasons, outlying samples could be caused by measurement errors, or they may represent a novel mechanism not representative by the majority of the current observations, both of which are worthy of further investigation.

Currently, to enable outlier detection, usually a hyperparameter regarding the proportion of outlying samples needs to be specified. Yu et al. [69] proposed a penalized mean-shift mixture model, RM^2, for simultaneous outlier detection and robust parameter estimation. Neykov et al. [70] proposed the trimmed likelihood estimator (**TLE**), where given a trimming parameter α, $0 \leq \alpha \leq 1$, the outliers are defined as the $N\alpha$ observations with the

smallest sample likelihood, and they presented a Trimmed Likelihood Estimator (TLE) algorithm based on EM algorithm, and a FAST-TLE algorithm using classification EM algorithm [71]. Similar to fast-TLE, Dogru and Arslan [72] proposed the adapted complete data log-likelihood function using the least trimmed squares (LTS) [73] criterion, where the sum of log-likelihood for portions of the data points were optimized. Following very similar steps as TLE algorithm, García-Escudero et al. [74], [75] proposed an algorithm (**CWM**) with further control over the scattering parameters, as well as a second trimming strategy on the explanatory variables. The challenge with the trimming based algorithms are the involvement of hyperparameters, namely, penalty parameter in RM^2 and the trimmning parameter α in the other algorithms, which could heavily impact the performance of these trimming based algorithms. Yu et al. [69] proposed using BIC procedure for hyperparameter tuning, however, BIC criterion becomes highly unstable when the total number of parameters, which equals to the total number of outliers, becomes large. As discussed in a recent review article [61], the involved parameters are interrelated with the number of components, where a high trimming level will lead to the removal of components with fewer observations. In summary, there is a lack of an algorithm that could adaptively trim the outlying samples to minimize its impact on the parameter estimation, while avoiding the pre-specification of the level of trimming.

2.1.2 Supervised High-dimensional Mixture Regression

Unsupervised learning algorithms that are typically employed to deal with heterogeneity in subpopulations include finite mixture models by assuming a separate distribution for each subpopulation [58], [76]–[78], and bi-clustering based discrete algorithms that performs feature selection and sample clustering simultaneously [79]–[81]. Based on the resultant genetic subtypes, deeper investigation into the genetic and phenotypic distinctions within each subtype could be carried out. Although the clustering methods may produce satisfactory classification of subtypes, many methods do not select genetic markers distinctive for each subtype, which however is essential in precision medicine. In addition, in the unsupervised clustering of high dimensional omics data, the high dimensional genetic feature space may

give rise to many different ways of clustering the samples, which may or may not be biologically/clinically meaningful [82]. Usually, the relevance of the subtypes to external biological or clinical presentations is analyzed in a *post hoc* fashion. As a result, without any supervision, the defined clusters based on a sea of genetic features may not necessarily relate to the phenotype of interest. Existing supervised clustering methods apply an *ad hoc* two-stage approach that consists of feature selection based on association with an external biological or clinical response variable, and clustering of samples using the selected features. However, due to the heterogeneity in sample population, the relationship between the external variable and the individual features could be highly non-linear, and a pre-selection of the features is not optimal.

As an external biological or clinical response variable available, mixture regression model intuitively become a feasible solution to identify intrinsic sample subgroups guided by the external information. While the mixture regression model is capable of handling the heterogeneous relationships, it doesn't work in the case of high dimensional genetic features, where the total number of parameters to be estimated is far more than the total number of observations. In addition, with the dense linear coefficients given by the ordinary EM algorithm, it is hard to deduce the disease subtype-specific genetic markers and make meaningful interpretations.

Penalized mixture regression has been explored in different settings [83]–[87] to handle the high dimensional mixture regression problem. The variable selection problem in the finite mixture of regression model was first studied using regularization methods such as LASSO [88] and SCAD [89] in [83]. They considered the traditional cases when the number of candidate covariates is much smaller than the sample size, and proposed a modified expectation–maximization (EM) algorithm to perform both estimation and variable selection simultaneously. The following methods consider the cases where the number of covariates may be much larger than the sample size. In [84], the authors proposed a reparameterized mixture regression model, and showed evidence for the benefit of the reparameterized model with numerically better behaviors. A block-wise Minorization Maximization (MM) algorithm was proposed in [86], where at each iteration, the likelihood function is maximized with respect to a block of variables while the rest of the blocks are held fixed. The work proposed

by Devijver [90] mainly considers the parameter estimation, after the variables have been selected by $\mathcal{L}_1$-penalized maximum likelihood estimator, as well as model selection among a set of pre-given ones. The imputation-conditional consistency (ICC) algorithm proposed by [87] adopted a two-stage approach: variable selection by aggregating the selection results through multiple Expectation–maximization iterations, and the parameter estimation stage for which the problem could be cast as a low dimensional one.

While some of the methods may produce consistent estimates of parameters under proper conditions, they tend to suffer from slow convergence rate in high dimensional setting, especially with smaller N or larger K, and the number of hyper-parameters for regularization further drags down the computational efficiency caused by the need of cross validation.

2.1.3 Inference of Spatial Dependency

Many problems in the environmental, economic, and biological sciences involve spatially collected data, and a main problem of interest is investigation of the relationship between a response variable and a set of explanatory variables over the spatial domain using regression modeling. Notably, the relationships between response variables and covariates may exhibit complex spatially dynamic patterns that cannot be captured by constant regression coefficients. Instead, such relationships may abruptly change at a certain boundary of two neighboring spatial clusters, but stay relatively homogeneous within clusters. Detecting clusters of observations that display similarity in both regression relationships and spatial proximity allows straightforward interpretations of local associations between response variables and covariates. For example, the residential real estate pricing could be quite similar in a local community, but drastically differ for two houses across the street [91]; a major goal of analyzing functional magnetic resonance imaging (fMRI) data is to detect spatially distributed and functionally linked regions that continuously share information with each other in reaction to different stimuli [92]. For all these real-world application settings, the collected data may often contain outliers, which may severely corrupt the analysis results if not properly handled. Overall, the spatial nonstationarity, local homogeneity, and model robustness are three main challenges in spatial regression modeling.

In the nonspatial setting, finite mixture regression models have been used in many areas as an effective exploratory approach to identify heterogeneity in response–predictor relationships. For an overview, see [58], [60]. To account for outliers or heavy-tailed noises, many algorithms have been developed to estimate the parameters robustly [61]. To seek for robust parameter estimation in the presence of outliers, methods have been developed that replaced the least-square criterion in the M-step of the expectation maximization (EM) algorithm by more robust criterion [62]–[68]. To enable simultaneous model estimation and outlier removal, penalized mean-shift mixture model [69], and the least trimmed likelihood estimator [70], [72], [74], [75] were proposed. While these methods could robustly capture the heterogeneous relationship between response and predictor variables, they are not designed to model the spatial dependency.

In modeling the spatial dependency, conventional nonstationary spatial regression models such as geographically weighted regression (GWR) [93]–[95] and Bayesian spatially varying coefficient (SVC) [96], [97] models fit as many regression models to the data as there are observations, at the cost of a large computational burden for large spatial datasets, and sometimes may lead to overfitting. In addition, interpretation of the GWR and SVC models require visual inspection of the coefficient maps to pursue local homogeneity, and can not automatically capture the spatially clustered patterns. In order to automatically detect spatially homogeneity cluster, a penalized spatial regression model has been proposed [98], where a fused-lasso [99] type of penalty has been developed to account for the spatial homogeneity in the linear regression setting. Nevertheless, the spatial smoothness assumption in the above spatial regression models could be problematic and violated due to natural or man-made discontinuities in the spatial domain. In addition, none of these methods is designed to handle outliers.

Model-based spatial segmentation is another type of methods to deal with spatial data using spatially constrained Gaussian mixture model [100], [101]. Spatial segmentation incorporates spatial information between neighboring pixels into the Gaussian mixture model based on Markov random field (MRF), with a goal to cluster all variables (e.g. pixels in image), where the distance of two instances is dependent on both their feature expressions and spatial proximity. This comes at a high computational cost. While robust spatial seg-

mentation algorithms are available [100], [101], they fail to intentionally model the linear relationship between the response and predictors, but instead simply treat the response and predictors as different features.

2.2 Preliminaries for Mixture Regression

2.2.1 Basis

A continuous L-dimensional random variable will be denoted as $X = (X_1, ..., X_p, ..., X_P)$, where X_p corresponds to the pth variable. Lower case letters will be used for a particular observation (or realization) $x = (x_1, ..., x_p, ..., x_P)$ of a variable X. Bold face letters, such as X, will denote a data of N observations of variable X or equivalently, a $N \times P$ matrix, where x_{ip} is the value of the ith observation for the pth variable in X.

A probability density function (pdf) $p(x)$ is any function defining the probability density of a variable X such that $p(x) \geq 0$ and $\int_{-\infty}^{\infty} p(x) = 1$. For a given pdf $p(x)$, the expectation of X is defined as,

$$E[X] = \int_{-\infty}^{\infty} xp(x)dx. \tag{2.1}$$

Suppose we have N observation $X_1, ..., X_N$ from a Gaussian distribution with unknown mean μ and known variance σ^2. To find the maximum likelihood estimate for μ, we find the log-likelihood $L(\mu)$, take the derivative with respect to μ, set it equal zero, and solve for μ:

$$\begin{aligned} L(\mu) &= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp -\frac{(x_i - \mu)^2}{2\sigma^2} \\ \Rightarrow L(\mu) &= \sum_{i=1}^{N} [\log(\frac{1}{\sqrt{2\pi\sigma^2}}) - \frac{(x_i - \mu)^2}{2\sigma^2} \\ &\Rightarrow \frac{d}{d\mu} l(\mu) = \sum_{i=1}^{N} \frac{x_i - \mu}{\sigma^2} \end{aligned} \tag{2.2}$$

Setting this equal to zero and solving for μ, we get that $\mu_{MLE} = \frac{1}{N} \sum_{i=1}^{N} x_i$. Note that applying the log function to the likelihood helpes us decompose the product and removed the exponential function so that we could easily sovle for the MLE.

2.2.2 Finite Mixture Model

Let $X = (X1, ..., X_P)$ be a P-dimensional continuous random variable and $x = (x_1, ..., x_p)$ be an observation of X. A probability density function (pdf) of a mixture model is defined by a convex combination of K components pdfs [102],

$$p(x \mid \Theta) = \sum_{k=1}^{K} \pi_k p_k(x \mid \theta_k) \tag{2.3}$$

where $p_k(x \mid \theta_k)$ is the pdf of the kth component, π_k are the mixing proportions (or component priors) and $\Theta = (\pi_1, ..., \pi_K, \theta_1, ..., \theta_K)$ is the set of parameters. We assume that

$$\begin{gathered} \pi_k \geq 0, for\ k \in \{1, ..., K\} and \\ \sum_{k=1}^{K} \pi_k = 1 \end{gathered} \tag{2.4}$$

By the property of convexity, given that each $p_k(x \mid \theta_k)$ defines a probability density function, $p(x \mid \Theta)$ will also be a probability density function.

The most straightforward interpretation of mixture models is that the random variable X is generated from K distinct random processes. Each of these processes is modeled by the density $p_k(x \mid \theta)$, and π_k represents the proportion of observations from this particular process.

2.2.3 Gaussian Mixture Model

A Gaussian mixture model [103] represents a distribution as

$$p(x \mid \Theta) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x \mid \mu_k, \sigma_k^2) \tag{2.5}$$

This is easy to deduce Gaussian mixture model by replacing the probability density function of each component to a Gaussian distribution. In addition, the parameters here contains mean value μ and variance σ. The mixing parameter π_k still following previous rules.

2.2.4 Gaussian Mixture Regression

Let $Y = (y_1, ..., y_N)^T \in \mathcal{R}^N$, $X = [\boldsymbol{x}_1, ..., \boldsymbol{x}_N]^T \in \mathcal{R}^{N\times(P+1)}$ be a finite set of observations, and X the design matrix, and Y the response vector. Consider an FMGR model parameterized by $\theta = \{(\boldsymbol{\pi_k}, \beta_k, \sigma_k^2)\}_{k=1}^K$, it is assumed that when $(\boldsymbol{x}, y)$ belongs to the k-th component, $k = 1, ..., K$, then $y = \boldsymbol{x}^T\beta_k + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma_k^2)$. Then, the condition density of y given $\boldsymbol{x}$ is $p(y \mid \boldsymbol{x}, \theta) = \sum_{k=1}^K \boldsymbol{\pi_k}\mathcal{N}(y; \boldsymbol{x}^T\beta_k, \sigma_k^2)$, where $\mathcal{N}(y; \mu, \sigma^2)$ is the normal density function with mean μ and variance σ^2.

It is easy to deduce Gaussian mixture regression from Gaussian mixture model by little modification [104]. In detail, a random variable x was replaced by a residual of linear regression fitting. The previous mean center μ is replaced as zero and the previous variance still keep the same notation.

2.2.5 MLE of Gaussian Mixture Regression

Based on unknown parameters, so from the first section, our likelihood is:

$$L_{X,Y}(\theta) := \prod_{\mathrm{i}=1}^{N}\sum_{k=1}^{K}\boldsymbol{\pi_k}\mathcal{N}(y_\mathrm{i}; \boldsymbol{x}_\mathrm{i}^{\boldsymbol{T}}\boldsymbol{\beta_k}, \sigma_k^2) \tag{2.6}$$

The maximum likelihood estimate for θ is through maximizing the following log likelihood:

$$L_{X,Y}(\theta) := \sum_{\mathrm{i}=1}^{N}\log(\sum_{k=1}^{K}\boldsymbol{\pi_k}\mathcal{N}(y_\mathrm{i}; \boldsymbol{x}_\mathrm{i}^{\boldsymbol{T}}\boldsymbol{\beta_k}, \sigma_k^2)) \tag{2.7}$$

Taking a look at the expression above, we already see a difference between this scenario and the simple setup in the *Basics* section. We see that the summation over the K components "blocks" our log function from being applied to the normal densities. If we were to follow the same steps as above and differentiate with respect to μ_k and set the expression equal to zero, we would get stuck because we cannot analytically solve for μ_k.

The trick here we use is to introduce a hidden variable such that its knowledge would simplify the maximization. This hidden (latent) variable z which would represent which Gaussian generated our observation x, with some probability. These are variables which are

never observed, and where we don't know the correct values in advance. They are roughly analogous to hidden units,in that the learning algorithm needs to figure out what they should represent, without a human specifying it by hand. In mixture models, the latent variable corresponds to the mixture component.

Let z be the membership indicator for observation $(\boldsymbol{x}, y)$, then $\boldsymbol{\pi}_k = p(z = k)$. In machine learning, the latent variable z is considered as a latent pattern lying under the data, which the observer is not able to see very directly. The wide application of this circumstance in machine learning is what makes EM algorithm so important.

2.2.6 EM algorithm

Intuitively, the latent variable z_n should help us find the MLEs. We first attempt to compute the posterior distribution of z_n given the observations:

$$p(z_n = k \mid x_n) = \frac{p(x_n \mid z_n = k)p(z_n = k)}{p(x_n)} = \frac{\boldsymbol{\pi}_k \mathcal{N}(\mu_k, \sigma_k^2)}{\sum_{k=1}^{K} \boldsymbol{\pi}_k \mathcal{N}(\mu_k, \sigma_k^2)} = \hat{p}_{nk} \tag{2.8}$$

Then, we can solve for μ_k to get:

$$\hat{\mu}_k = \frac{\sum_{n=1}^{N} \hat{p}_{nk} x_n}{\sum_{n=1}^{N} \hat{p}_{nk}} \tag{2.9}$$

We see that $\hat{\mu}_k$ is therefore a weighted average of the data with weights $\hat{p}_{nk}$. Similarly, if we apply a similar method to finding $\hat{\sigma_k^2}$ and $\hat{\boldsymbol{\pi}}_k$, we find that:

$$\hat{\sigma_k^2} = \frac{\sum_{n=1}^{N} \hat{p}_{nk}(x_n - \mu_k)^2}{\sum_{n=1}^{N} \hat{p}_{nk}} \tag{2.10}$$

We can derive the prior class probabilities as

$$\hat{\boldsymbol{\pi}}_k = \frac{1}{N} \sum_{n=1}^{N} \hat{p}_{nk} \tag{2.11}$$

Again, remember that $\hat{p}_{nk}$ depends on the unknown parameters, so these equations are not closed-form expression. This looks like a vicious circle. EM algorithm is an alternative solution which provides a general framework for fitting models on incomplete data.

The EM algorithm [105], proceeds as follows:

1. Initialize the parameter and evaluate the log-likelihood with these parameters.

2. **E-step**: Estimate the posterior class probabilities for each observation $\hat{p}_{nk}$ using the current values of the μ_k and σ_k^2,

3. **M-step**: Estimate new parameters $\theta = \{(\boldsymbol{\pi}_k, \mu_k, \sigma_k^2)\}_{k=1}^K$ with above equations. The goal is maximizing the log-likelihood for each component separately using the posterior probabilities as weights $\underset{\theta_k}{max} \sum_{n=1}^N \hat{p}_{nk} \log f(y_n \mid x_n, \theta_k)$.

4. Evaluate the log-likelihood with new parameters. If the log-likelihood has converged by measured with some small ϵ, stop. Otherwise, go back to step 2.

3. COMPONENT-WISE ADAPTIVE TRIMMING FOR ROBUST MIXTURE REGRESSION

This chapter focuses on developing effective and efficient algorithm for robust mixture regression. Robust mixture regression has many important applications including in human cancer genomics data, where the population often displays strong heterogeneity added by unwanted technological perturbations. A novel Component-wise Adaptive Trimming (CAT) method was proposed, which simultaneously detect outliers and estimate parameter of mixture model.

In this chapter, we have three major contributions by using robust mixture regression model to handle outlier.

1. We proposed a component-wise trimming robust mixture regression model to detect outliers and estimate parameter of mixture model simultaneously.

2. We had a new definition of outlier under Classification-Expectation-Maximization (CEM) algorithm.

3. We provided a platform that migrates the robustness issue from mixture regression to (single component) linear regression, where the latter have been extensively studied.

EM algorithm is the traditional method to solve mixture regression problem in last decade [105]. However, EM algorithm is vulnerable to outliers or heavy-tailed noise. In other words, the convergence and each iteration will be impacted if outliers were added or heavy-tailed noise was measured in data. Identifying outliers not only could improve the parameter estimation for the mixture model, but also enable us a way to find novel mechanism. The reason is that outlying samples may represent a novel or different regression models which not represented by the current observations. To identify the outlier, several robust mixture regression models (Section 2.1.1) were proposed. But available methods face the challenge that highly specific mathematical consideration prohibit their application to the biomedical data with unknown error distribution. In this chapter, we first introduced the CEM algorithm which is a variant of the EM algorithm in Section 3.2.2. Under the CEM algorithm, we had a new definition of outlier. The core idea is only considering an observation as an outlier if it is an outlier to the component it belongs to. This new definition shifts the robustness issue

from a mixture model to several independent robust linear regression models. The latter has been well defined and studied. By updating the clustering membership in each iteration, we can disentangle a complex problem into several simple problem. Thus, this method not only solve this specific issue but also provide a platform and framework to solve similar problem.

3.1 Introduction

In biomedical science, the patient population often consists of different molecular subtypes [106], and the outlying samples introduced by technological errors makes it more challenging to tease out the latent relations among the genomic markers using FMGR. To address the challenges in simultaneous outlier detection and robust parameter estimation in FMGR, we adopted the idea of Classification-Expectation-Maximization (CEM) algorithm, where individual observations are assigned to a definite cluster as part of the maximization process, different from traditional EM algorithm [71]. Essentially, CEM maximizes the complete data likelihood, instead of the observed data likelihood as in traditional EM, and has been shown to outperform traditional EM with faster convergence rate and better or comparable estimates [107]. Under CEM, each component has its exclusive members, which makes it possible to apply a trimmed likelihood approach designed for (single component) linear regression on its member, and hence enables both robust parameter estimation as well as outlier detection for the component. Our major contribution in this method is, by introducing CEM to FMGR, we provided a platform that migrates the robustness issue from mixture regression to (single component) linear regression, for which LTS estimators have been extensively studied [73], [108]. Therefore, since LTS-based robust regression has a high breakdown point, we could avoid the pre-specification of trimming parameter, by simply opt to maximize the sum of decreasing ranked likelihood of a sufficiently small portion of the samples within each component, namely, 0.5. In addition, since the task of outlier detection in the mixture model was converted to that of linear regression in each component, it is possible to formally define outliers in FMGR. Overall, our algorithm detects outlier in a data-driven fashion free of hyperparameters, and is hence computationally efficient and user-friendly.

The remainder of this chapter is organized as follows. In Section 3.2, we will introduce the complete data maximum likelihood, and the CEM algorithm, based on which, our component-wise adaptive trimming method is developed. In Section 3.3, we show the performance comparison of our method with other six state of the art methods on synthetic datasets. In Section 3.4, we will apply all methods to a real world dataset studying the heterogeneous DNA methylation regulatory effects on gene expression in colon cancer.

3.2 Methods

3.2.1 The Complete Data Maximum Likelihood Estimation

We are given a set of observations $(\boldsymbol{x}_{\mathrm{i}}, y_{\mathrm{i}})_{\mathrm{i}=1}^{N}$ and assignments $(z_{\mathrm{i}})_{\mathrm{i}=1}^{N}$. Then, the likelihood that all observations have been drawn according to a FMGR $\boldsymbol{\theta}$ and that each observation $(\boldsymbol{x}_{\mathrm{i}}, y_{\mathrm{i}})$ has been generated by the z_{i}-th component, is given by

$$\prod_{\mathrm{i}=1}^{N} p(y_{\mathrm{i}}, z_{\mathrm{i}} \mid \boldsymbol{x}_{\mathrm{i}}, \boldsymbol{\theta}) = \prod_{\mathrm{i}=1}^{N} \boldsymbol{\pi}_{z_{\mathrm{i}}} \mathcal{N}(y_{\mathrm{i}}; \boldsymbol{x}_{\mathrm{i}}^{\boldsymbol{T}} \boldsymbol{\beta}_{z_{\mathrm{i}}}, \sigma_{z_{\mathrm{i}}}^2) \tag{3.1}$$

This is called the complete-data likelihood. Note that the assignments $\{z_{\mathrm{i}}\}_{\mathrm{i}=1}^{N}$ define a partition of the N observations, $\mathcal{C} = \bigcup_{k=1}^{K} C_k$, such that $\mathrm{i} \in C_k$ iff $z_{\mathrm{i}} = k$. Denote n_k as the total number of elements in C_k, we can then rewrite the Equation 3.1 in its logarithm form as

$$\mathcal{L}_{X,Y}^{f}(\boldsymbol{\theta}, \boldsymbol{\mathcal{C}}) := \sum_{k=1}^{K} \{ \sum_{\mathrm{i} \in C_k} \log \mathcal{N}(y_{\mathrm{i}}; \boldsymbol{x}_{\mathrm{i}}^{T} \boldsymbol{\beta}_k, \sigma_k^2) + \log \boldsymbol{\pi}_k n_k \} \tag{3.2}$$

We introduce the complete data maximum likelihood estimates (CMLE) as follows.

Definition 3.2.1. ***(Complete-data Maximum Likelihood Estimates, CMLE)*** *Let X be the design matrix, and Y be the response vector. Given an integer K, find a partition $\mathcal{C} = \{C_1, ..., C_K\}$ of the N observations and FMGR parameters $\boldsymbol{\theta} = \{(\boldsymbol{\pi}_k, \boldsymbol{\beta}_k, \sigma_k)\}_{k=1}^{K}$ that maximizes $\mathcal{L}_{X,Y}^{f}(\boldsymbol{\theta}, \mathcal{C})$ defined in Equation 3.2.*

Note that, CMLE is not well defined in this form. For example, for an observation $(\boldsymbol{x}_{\mathrm{i}}, y_{\mathrm{i}})$, if $\boldsymbol{\beta_k}$ is chosen such that $y_{\mathrm{i}} = \boldsymbol{x}_{\mathrm{i}}^{T} \boldsymbol{\beta}_k$ and we let $\sigma_k \to 0$, then $f(y_{\mathrm{i}}; \boldsymbol{x}_{\mathrm{i}}, \boldsymbol{\theta}) \to \infty$, which results in infinite likelihood. A common practice is to put some mild restrictions on the cluster

size or the variance parameter, then we can lower bound the variance associated with each regression line, and the CMLE will be well defined [75], [109].

3.2.2 Alternating Optimization Scheme using the CEM algorithm

We introduce the alternating optimization algorithm to solve the CMLE problem [109]. Clearly, fixing the partition $\mathcal{C} = \{C_1, ..., C_K\}$, the optimal mixture parameter is given by $\boldsymbol{\theta} = \{(\boldsymbol{\pi}_k, \boldsymbol{\beta}_k, \sigma_k)\}_{k=1}^K$ with

$$\boldsymbol{\pi}_k = \frac{n_k}{\sum_{l=1}^K n_l} \tag{3.3}$$

$$(\boldsymbol{\beta}_k, \sigma_k^2) = \mathbf{OLS}(Y_{C_k}, X_{C_k,;}) \tag{3.4}$$

Here, $\mathbf{OLS}(Y_{C_k}, X_{C_k,;})$ means the ordinary least squares solution to regressing Y on X using only observations from C_k.

Fixing the FMGR parameters $\boldsymbol{\theta} = \{(\boldsymbol{\pi}_k, \boldsymbol{\beta}_k, \sigma_k)\}_{k=1}^K$, the optimal partition is given by assigning each point to its most likely component, i.e.

$$\mathrm{i} \in C_k \Longleftrightarrow k = \underset{l \in \{1,...,K\}}{\mathbf{argmax}} p(z_\mathrm{i} = l \mid \boldsymbol{x}_\mathrm{i}, y_\mathrm{i}, \boldsymbol{\theta}) \tag{3.5}$$

where

$$p(z_\mathrm{i} = k \mid \boldsymbol{x}_\mathrm{i}, y_\mathrm{i}, \boldsymbol{\theta}) = \frac{\boldsymbol{\pi}_k \mathcal{N}(y_\mathrm{i}; \boldsymbol{x}_\mathrm{i}^T \boldsymbol{\beta}_k, \sigma_k^2)}{\sum_{l=1}^K \boldsymbol{\pi}_l \mathcal{N}(y_\mathrm{i}; \boldsymbol{x}_\mathrm{i}^T \boldsymbol{\beta}_l, \sigma_l^2)} \tag{3.6}$$

which is the posterior probability that $(\boldsymbol{x}_\mathrm{i}, y_\mathrm{i})$ lies on the k-th regression line of the mixtures. By repeatedly updating between $\boldsymbol{\theta}$ and $\mathcal{C}$, we will show in Theorem 3.2.1 that the solution converges to a stationary point of the full likelihood function. We call this alternating scheme the CEM algorithm, and Algorithm 1 outlined the major steps including initialization, estimation, classification and maximization steps.

Theorem 3.2.1. *The complete data log likelihood, $\mathcal{L}_{X,Y}^f(\boldsymbol{\theta}^{(m)}, \mathcal{C}^{(m)})$, is non-decreasing for any sequence $\mathcal{C}^{(m)}, \boldsymbol{\theta}^{(m)}$ defined as in Algorithm 1, and it converges to a stationary value. Moreover, if the maximum likelihood estimates of the parameters are well-defined, the se-*

quence of $\mathcal{C}^{(m)}, \boldsymbol{\theta}^{(m)}$ *converges to a stationary position.*

The CEM algorithm has been popularly used in both the clustering and regression-based clustering settings. Obviously, it is vulnerable to outliers, and in the next sections, we introduce our robust procedure on top of the CEM algorithm.

Input: Response vector Y; independent variables in matrix $X_{N\times(P+1)}$; the number of mixing component, K; size of initialization random sample, n_0; the maximum number of iteration L_0
Output: $\boldsymbol{\theta} = \{\boldsymbol{\pi}_k, \boldsymbol{\beta}_k\}_{k=1}^{K}; \mathcal{C} = \bigcup_{k=1}^{K} \mathcal{C}_k$
for $k = 1, ..., K$ **do**
  Draw a random sample of size n_0 from set $\{1, ..., N\}$, indexed by I_k
  Run ordinary linear regression to get initial regression parameter estimates: $(\boldsymbol{\beta}_k^{(0)}, \sigma_k^{(0)}) =: \mathbf{OLS}(Y_{I_k} \sim X_{I_k,:})$
end
for $m = 0, ..., L_0$ *or until convergence* **do**
  E-step: Compute for $\mathrm{i} = 1, ..., N$ and $k = 1, ..., K$, the current posterior probabilities $p_{\mathrm{i}k}^{(m)}$ by
  $p_{\mathrm{i}k}^{(m)} = p(z_{\mathrm{i}} = k \mid \boldsymbol{x}_{\mathrm{i}}, y_{\mathrm{i}}, \boldsymbol{\theta}^{(m)})$
  C-step: For $k = 1, ..., K$, assign $C_k^{(m)} = \{\mathrm{i} \mid \underset{l\in\{1,...,K\}}{\operatorname{argmax}}\ p_{\mathrm{i}l}^{(m)} = k, \mathrm{i} = 1, ..., N\}$, and let $n_k^{(m)}$ be the size of $C_k^{(m)}$
  M-step: For $k = 1, ..., K$, the parameters are then updated by
  $\boldsymbol{\pi}_k^{(m+1)} = \frac{n_k^{(m)}}{\sum_{l=1}^{K} n_l^{(m)}}$,
  $(\boldsymbol{\beta}_k^{(m+1)}, \sigma_k^{2(m+1)}) = \mathbf{OLS}(Y_{C_k^{(m)}}, X_{C_k^{(m)},:})$
end

Algorithm 1: CEM

3.2.3 A new definition of outlier within the CEM framework

In linear regression, outliers are understood as observations that deviate from the model assumptions, and obviously, samples with lower likelihood are more likely to be outliers. If the ratio of outliers, α, is known, the outliers are identified as the ratio α of the total observations with the lowest likelihood.

Unfortunately, such a definition for outliers becomes less applicable in the case of mixture regression. Given a robust mixture regression model and a trimming ratio α, if we follow the same logic as in linear regression, then the $\lceil n * \alpha \rceil$ observations with the smallest overall likelihood will be detected as outliers, as in [70]. This trimmed likelihood approach implies that an observation with lower overall likelihood is more likely to be an outlier than an observation with higher overall likelihood. However, the overall likelihood depends on not only the likelihood of the observation with respect to each component, but also the proportion of each component, and such a criteria for outlier becomes problematic if the mixing components are unbalanced. In other words, a low $\boldsymbol{\pi}_k$ will down-weigh the "outlierness" of an observation from the k-th component. In addition, if we argue that, given a set of observations, we could always find certain mixture model to well explain it, there is no basis for us to call any observation an outlier.

The complete data likelihood approach based CEM algorithm disentangles the mixture distribution into exclusive clusters, within which, the robustness issue could be much easily handled give the tremendous amount of research conducted for robust linear regression. More importantly, we could introduce a more natural definition for outliers.

Definition 3.2.2. ***(Outliers of FMGR)***
Given an FMGR model parameterized by $\boldsymbol{\theta} = \{(\boldsymbol{\pi}_k, \boldsymbol{\beta}_k, \sigma_k)\}_{k=1}^K$, *under CMLE, an observation* $(\boldsymbol{x}_{\mathrm{i}}, y_{\mathrm{i}})$ *is considered as an outlier, if* $\mathrm{i} \in \mathcal{C}_k$ *and* $\mid y_{\mathrm{i}} - \boldsymbol{x}_{\mathrm{i}}^T \boldsymbol{\beta}_k \mid \geq \eta_k(\sigma_k)$. *In other words, an observation is considered as an outlier if it is an outlier to the component it belongs to.*

Here $\eta_k(\cdot)$ is a criteria for outlier-ness in linear regression, which usually depends on the variance level of the component. This new definition transforms the robustness issue from a mixture model to its K linear regression components, the latter of which has been well defined and studied. Different from the overall likelihood-based outlier definition adopted by TLE and CWM, our definition of outlier does not involve the cluster prior, and is hence more fair to clusters with relatively smaller sizes. In fact, in tables 3.5 - 3.8, we demonstrated using simulation data that, under unbalanced cluster sizes, TLE and CWM, which defines outliers based on the overall likelihood, perform much worse than our proposed method, which adopted our new definition of outliers.

Naturally, to confer a robust parameter estimation for FMGR under CMLE, we could replace the least square criterion for parameter estimation in the M-step by a robust criterion; and further to enable simultaneous outlier detection, we could choose to use any trimmed likelihood approach with high break-down point [110].

3.2.4 Robust CEM algorithm

Under Definition 3.2.2, detecting outliers of the FMGR model could be accomplished through detecting the component-wise outliers. Many robust estimators have been proposed for linear regression to achieve high breakdown point or high efficiency or both [111], where the objective of minimizing the sum of the squared residuals has been replaced by more robust measures. Among them, the Least Median of Squares (LMS) estimates [73], [108] which minimize the median of squared residuals, Least Trimmed Squares (LTS) estimates [73], [112] which minimize the trimmed sum of squared residuals, and S-estimates [113] which minimize the variance of the residuals, remain to be powerful robust algorithms with a breakdown point as high as 0.5, the best that can be expected. This means that the resulting estimators from these algorithms can resist the effect of nearly 50% of contamination in the data.

To achieve simultaneous outlier detection and robust parameter estimation, instead of maximizing the component-wise sum of likelihood in CEM, our **C**omponent-wise **A**daptive **T**rimming method, namely **CAT**, maximizes the component-wise sum of trimmed likelihood, i.e.,

$$\mathcal{L}_{X,Y}^{f,trim}(\boldsymbol{\theta}, \boldsymbol{\mathcal{C}}) := \sum_{k=1}^{K} \sum_{\mathrm{i}=g_k}^{h_k} \{(l_k)_{\mathrm{i}:n} + \log \boldsymbol{\pi}_k\} \tag{3.7}$$

where $l_{k\mathrm{i}} = \log\{\mathcal{N}(y_{\mathrm{i}}; \boldsymbol{x}_{\mathrm{i}}^T \boldsymbol{\beta}_k, \sigma_k^2)\}$, and $(l_k)_{r:n_k}$ denotes the r-th largest value of the sequence of $l_{k\mathrm{i}}$, namely, $(l_k)_{1:n_k} \geq \cdots \geq (l_k)_{n_k:n_k}$. For component-wise LMS estimates, one could let $g_k = h_k = [n_k/2] + 1$; for LTS estimate, one could let $g_k = 1, h_k = [n_k/2] + 1$, which obtains the highest break-down point of LTS. Our CAT algorithm adopted the latter robust procedure in order to avoid the selection of a trimming parameter. The high breakdown point of the LTS algorithm makes it possible to minimize the effect of outliers in parameter

estimation even if only half of the likelihood were optimized, and we could thus develop a data-driven algorithm for simultaneous outlier detection and robust parameter estimation in FMGR.

In general, as outlined in Algorithm 2, our **CAT** algorithm implements very similar steps to the CEM algorithm outlined in Algorithm 1, except that the OLS estimates in the M-step is replaced by the LTS estimate. CAT starts by initializing the posterior probability matrix, W. For $k = 1, ..., K$, we randomly draw n_0 samples to build a robust linear regression model, and the posterior probability of sample i for component k will be initialized as the density of sample i fitting the k-th robust regression line. For robust linear regression with trimmed likelihood approach, we used the "ltsReg" function in the "robustbase" library in R [73], [114]–[116], where the parameters were estimated to maximize the sum of the likelihood of the largest half, and outliers were detected as those with relatively large residuals. With initialized W, CAT then runs a robust CEM algorithm where the OLS estimates in Algorithm 1 in the M-step was replaced by robust estimates using trimmed likelihood method.

Input: Response vector Y; independent variables in matrix $X_{N\times(P+1)}$; the number of mixing component, K; size of initialization random sample, n_0; the maximum number of iteration L_0
Output: Robust FMGR parameter estimate $\boldsymbol{\theta}^* = \boldsymbol{\theta}$; outlier set $U^* = U$
Initialization: Same as Algorithm 1
for $m = 0, ..., L_0$ *or until convergence* **do**
  E-step: Compute for $\mathrm{i} = 1, ..., N$ and $k = 1, ..., K$, the current posterior probabilities $p_{\mathrm{i}k}^{(m)}$ by
  $p_{\mathrm{i}k}^{(m)} = p(z_{\mathrm{i}} = k \mid \boldsymbol{x}_{\mathrm{i}}, y_{\mathrm{i}}, \boldsymbol{\theta}^{(m)})$
  C-step: For $k = 1, ..., K$, assign $C_k^{(m)} = \{\mathrm{i} \mid \underset{l\in\{1,...,K\}}{\operatorname{argmax}}\ p_{\mathrm{i}l}^{(m)} = k, \mathrm{i} = 1, ..., N\}$, and let $n_k^{(m)}$ be the size of $C_k^{(m)}$
  M-step: For $k = 1, ..., K$, run robust linear regression using samples in $C_k^{(m)}$, i.e., $(\boldsymbol{\beta}_k^{(m+1)}, \sigma_k^{2(m+1)}) = \mathbf{RLM}(Y_{C_k^{(m)}}, X_{C_k^{(m)},:})$; $\boldsymbol{\pi}_k^{(m+1)} = \frac{n_k^{(m)}}{\sum_{l=1}^{K} n_l^{(m)}}$;and let U_k be outliers of component k
end

Algorithm 2: CAT: Component-wise Adaptive Trimming

Theorem 3.2.2. *The component-wise trimmed complete data log likelihood,* $\mathcal{L}_{X,Y}^{f,trim}(\boldsymbol{\theta}^{(m)}, \mathcal{C}^{(m)})$, *is non-decreasing for any sequence* $\mathcal{C}^{(m)}, \boldsymbol{\theta}^{(m)}$ *defined as in Algorithm 2, and it converges to a*

stationary value. Moreover, if the component-wise trimmed maximum likelihood estimates of the parameters are well-defined, the sequence of $\mathcal{C}^{(m)}, \boldsymbol{\theta}^{(m)}$ *converges to a stationary position.*

In the M-step, CAT updates the parameter $\boldsymbol{\theta}$ using the robust estimates of each component. We also propose a fast implementation, fast-CAT, to achieve faster convergence. As outlined in Algorithm 3, CAT adopted a model refit step, where an MLE estimate of $\boldsymbol{\theta}$ is obtained using the non-outlying samples only, as an update of the $\boldsymbol{\theta}$ at the current step, instead of the component-wise LTS estimates used in Algorithm 2. Note that standard regression analysis tools can be applied to recover the observations that should not have been regarded as outliers. The MLE estimates were conducted using function "flexmix" from the "flexmix" R package [117]. Similar to other algorithms, in our package implementation, we used multiple random starts to stabilize the results.

Input: Response vector Y; independent variables in matrix $X_{N\times(P+1)}$; the number of mixing component, K; size of initialization random sample, n_0; the maximum number of iteration L_0
Output: Robust FMGR parameter estimate $\boldsymbol{\theta}^* = \boldsymbol{\theta}$; outlier set $U^* = U$
Initialization: Same as Algorithm 1
for $m = 0, ..., L_0$ *or until convergence)* **do**
  E-step: Same as Algorithm 2
  C-step: Same as Algorithm 2
  M-step: For $k = 1, ..., K$, run robust linear regression using samples in $C_k^{(m)}$, i.e., $\mathbf{RLM}(Y_{C_k^{(m)}}, X_{C_k^{(m)},:})$; and let U_k be outliers of component k
  Refit-step: Let $U = \bigcup_k U_k$; $S = \{1, ..., N\} - U$; let $\boldsymbol{\theta}_S$ be the MLE estimates of fitting K mixture regression lines using samples in S only, and update the parameter $\boldsymbol{\theta}^{(m+1)}) = \boldsymbol{\theta}_S$
end

Algorithm 3: fast-CAT: Component-wise Adaptive Trimming

3.3 Experiments by using Simulated Data

3.3.1 Selection of Baseline Methods

We evaluated the performance of CAT, using the fast-CAT implementation in Algorithm 3, on synthetic datasets, and compare it with several existing method, including MLE, TLE,

CWM, MIXBI, MIXL, and MIXT. They stand for the maximum likelihood estimates using traditional EM algorithm [117], the two trimmed likelihood approaches [70], [75], the mixture bisquare [64], mixture Laplacian[66], the mixture t [67] approaches respectively.

Note that for TLE and CWM, the choice of the trimming proportion, α, needs to be pre-specified, and a large α will result in reduced efficiency, while a small α corrupt the parameter estimates. We always give the true ratio of outliers to TLE and CWM. In addition, a 5% increase of the true outlier ratio is also given to CWM, as the authors recommended the use of a relatively larger trimming ratio than needed [74] as a preventive procedure. As a result, we gave true outlier ratio to CWM method and the result is shown as CWM1. In addition, we also gave a wrong outlier ratio (5% larger than true ratio) to CWM method and the result is shown as CWM2.

3.3.2 Simulation Settings

We simulated data using two models with different number of covariates P, and number of components K. To simulate outliers, a mean-shift parameter, γ_{ij}, is added to the mean structure for its observations in each mixture component. For each model, we considered scenarios with different error distributions and different levels of outlier contamination.

Model 1: For each $i = 1, ..., N$, y_i is independently generated with

$$y_i = \begin{cases} 1 - x_{i1} + x_{i2} + \gamma_{i1} + \epsilon_{i1} & \text{if} \quad z_i = 1 \\ 1 + 3x_{i1} + x_{i2} + \gamma_{i2} + \epsilon_{i2} & \text{if} \quad z_i = 2 \end{cases}$$

where z_i is a component indicator generated from a Bernoulli distribution with $P(z_i = 1) = 0.43, P(z_i = 1) = 0.57$; x_{i1} and x_{i2} are independently generated from $N(0, 1)$; and the error terms ϵ_{i1} and ϵ_{i2} have the same distribution as ϵ. We consider the following five scenarios:

Scenario 1: $\epsilon \sim N(0, 1), \gamma_{i1} = \gamma_{i2} = 0$, standard normal distribution.
Scenario 2: $\epsilon \sim t_1, \gamma_{i1} = \gamma_{i2} = 0$, t-distribution with degree of freedom of 1.
Scenario 3: $\epsilon \sim t_3, \gamma_{i1} = \gamma_{i2} = 0$, t-distribution with degree of freedom of 3.

Scenario 4: $\epsilon \sim N(0,1), P(\gamma_{i1} \in (4,6)) = P(\gamma_{i2} \in (4,6)) = 0.05$, standard normal distribution with 5% outlier contamination.
Scenario 5: $\epsilon \sim N(0,1), P(\gamma_{i1} \in (4,6)) = P(\gamma_{i2} \in (4,6)) = 0.1$, standard normal distribution with 10% outlier contamination.

Model 2: For each $i = 1, ..., N$, y_i is independently generated with

$$y_i = \begin{cases} 1 - x_{i1} + \gamma_{i1} + \epsilon_{i1} & \text{if} \quad z_i = 1 \\ 1 + 3x_{i1} + \gamma_{i2} + \epsilon_{i2} & \text{if} \quad z_i = 2 \\ -1 + 0.1x_{i1} + \gamma_{i3} + \epsilon_{i3} & \text{if} \quad z_i = 3 \end{cases}$$

where z_i is a component indicator generated from a Multinomial distribution with $P(z_i = 1) = 0.3, P(z_i = 2) = 0.4, P(z_i = 3) = 0.3$. x_{i1} is independently generated from $N(0,1)$; and the error terms ϵ_{i1}, ϵ_{i2}, ϵ_{i3} have the same distribution with ϵ.

Scenario 1: $\epsilon \sim N(0,1), \gamma_{i1} = \gamma_{i2} = \gamma_{i3} = 0$, standard normal distribution.
Scenario 2: $\epsilon \sim t_1, \gamma_{i1} = \gamma_{i2} = \gamma_{i3} = 0$, t-distribution with degree of freedom of 1.
Scenario 3: $\epsilon \sim t_3, \gamma_{i1} = \gamma_{i2} = \gamma_{i3} = 0$, t-distribution with degree of freedom of 3.
Scenario 4: $\epsilon \sim N(0,1), P(\gamma_{i1} \in (4,6)) = P(\gamma_{i2} \in (4,6)) = P(\gamma_{i3} \in (4,6)) = 0.05$, standard normal distribution with 5% outlier contamination.
Scenario 5: $\epsilon \sim N(0,1), P(\gamma_{i1} \in (4,6)) = P(\gamma_{i2} \in (4,6)) = P(\gamma_{i3} \in (4,6)) = 0.1$, standard normal distribution with 10% outlier contamination.

For scenarios in both models 1 and 2, we simulated data of sample sizes 200 and 400. The bias and mean square error (MSE) of the regression coefficients and mixing proportions are calculated for each competing methods over 100 repetitions, i.e., $\widehat{\text{bias}}(\hat{\theta}) = \frac{1}{N}\sum_{j=1}^{N}\hat{\theta}_j - \theta$; $\widehat{\text{MSE}}(\hat{\theta}) = \frac{1}{N}\sum_{j=1}^{N}\left(\hat{\theta}_j - \theta\right)^2$. When number of feature for each component larger than 1, the order of estimated coefficient also matters. This is called the label switching issue [118]–[120]. The label switching issue creates some trouble on how to align the parameters of one component from predicted model to that of the true model. Different

component orders in the predicted and true model might give totally different results and there are no widely accepted methods to adjust for that. In our simulation study, we simply choose to order the components in the estimated parameter matrix by minimizing the Euclidean distance to the true parameter matrix.

3.3.3 Methods Comparisons and Performance Evaluations

Table 3.1 reports the bias (MSE) of parameter estimates for the seven methods under model 1 with 200 simulated samples. It is a two component model with two independent variables. Note that in scenarios 1,2,3, there are no added outliers, so the TLE estimates are the same as MLE estimates. For scenarios 4 and 5, the true outlier proportions were given to TLE and CWM1, and 5% increase of outlier proportion is given to CWM2. Similarly for tables 3.2, 3.3 and 3.4. When the component error terms are all normally distributed without outlier contamination (scenario 1), all methods have comparable performances, except that the CWM methods perform much worse. In scenario 2, where the error terms are t-distributed with degree of freedom of 1 (or Cauchy distribution), CAT resulted in the most favorable performances, followed by MIXBI and then CWM2. MLE, TLE and CWM1 estimates are severely corrupted, and MIXL and MIXT are also far off from the true values. For scenario 3, where the error terms are t-distributed with degree of freedom of 3, CAT, MIXBI and MIXL all worked equally well in parameter estimation. The MLE, TLE, CWM1, CWM2 and MIXT estimates are slightly off from the true regression coefficients. For scenarios 4 and 5, where the error terms are normally distributed, with 5% and 10% outlier contamination, CAT significantly outperformed all the other methods in terms of all the parameter estimations. Overall, CAT represents the most competitive one among the seven methods; TLE is very sensitive to heavy-tailed noise, and its performance is not ideal even given the true trimming proportion; for CWM, a larger than needed trimming proportion indeed leads to more accurate estimates.

Table 3.2 reports the bias (MSE) of the seven methods of parameter estimates under model 1 with 400 simulated samples. It is a two component model with two independent

variables. When the component error terms are all normally distributed without outlier contamination (scenario 1), CAT performed comparatively good with all methods, except that the CWM methods performed much worse. In Scenario 2 and 3, where the error terms are t-distributed with degrees of freedom of 1 and 3, CAT performed the best among all, followed by MIXBI and then MIXL, while the rest of the methods seem to be severely corrupted by the heavy tail noise. Such corruption is much alleviated with the increase of degree of freedom, as t-distribution with higher degree of freedom is more like normal distribution. For scenarios 4 and 5, where the error terms are normally distributed, with 5% and 10% outlier contamination, CAT significantly outperformed all the other methods in terms of parameter estimation. Again, CAT remains the best-performing one among the six methods.

Table 3.3 reports the bias (MSE) of the seven methods of parameter estimates under model 2 with 200 simulated samples. It is a three component model with one independent variable. When the component error terms are all normally distributed without outlier contamination (scenario 1), all six methods have comparable performances. In scenario 2, where the error terms are t-distributed with degree of freedom of 1 (or Cauchy distribution), CAT and MIXBI resulted in comparable performances, with CAT being slightly more accurate and stable. MLE, TLE, CWM1, and MIXL estimates are severely biased, while CWM2 is highly unstable with large variance, and MIXT is also far off from the true values. For scenario 3, where the error terms are t-distributed with degree of freedom of 3, CAT, MIXBI and MIXL all worked equally well in parameter estimation. The MLE, TLE, CWM and MIXT estimates are slightly off from the true regression coefficients. For scenarios 4 and 5, where the error terms are normally distributed, with 5% and 10% outlier contamination, CAT significantly outperformed all the other methods in terms of parameter estimation. We show again the more competitive performance of CAT over others.

Table 3.4 reports the bias (MSE) of the six methods of parameter estimates under model 2 with 400 simulated samples. It is a three component model with one independent variable. When the component error terms are all normally distributed without outlier contamination (scenario 1), CAT and MIXBI both performed well, followed by MIXL; and they all outperformed the rest of the algorithms, even including the MLE estimates. We argue that

even though no outliers or heavy tailed noise is added, there is still likely "outlying" samples caused purely by chance; robust procedures like CAT and MIXBI work in preventive fashion to account for these random outliers, and are hence more robust even compared with the MLE. In scenario 2, where the error terms are t-distributed with degree of freedom of 1 (or Cauchy distribution), CAT, CWM2, and MIXBI resulted in comparable performances, while the rest of the methods did much worse. For scenario 3, where the error terms are t-distributed with degree of freedom of 3, CAT and MIXBI both worked equally well in parameter estimation, with CAT slightly better. The MLE, TLE, CWM, MIXL, and MIXT estimates are slightly off from the true regression coefficients. For scenarios 4 and 5, where the error terms are normally distributed, with 5% and 10% outlier contamination, CAT significantly outperformed all the other methods in terms of parameter estimation. We demonstrated here that CAT is a more robust method compared with others.

Overall, CAT demonstrated its advantage over others with its strong capacity of adaptive trimming and robustness to both outliers and heavy-tailed noises. Of note, MIXBI is a robust algorithm whose performance is next to CAT, however, it conducts parameter estimations in presence of outliers, which may tend to bring down the accuracy in parameter estimation. For CWM, indeed a slightly large trimming ratio may lead to better estimates, but it seems to over-trim the data with its "second" trimming step, which may be the reason for its low estimation efficiency. We also show that even when given the right outlier prevalence, TLE still can't produce results as robust as CAT, and it did worse when P increases. In the case of no deliberate outlier contamination (scenario 1), CAT performs better than or comparable to MLE, which is because CAT can automatically trim off observations that are highly noisy as a preventive procedure. When the error terms are t-distributed, CAT still remains the most robust method among all.

We also simulated data scenarios where the clusters are highly unbalanced, and basically, for $K = 2$, we simulated data with cluster prior probability $P(z_{\mathrm{i}} = 1) = 0.38, P(z_{\mathrm{i}} = 2) = 0.62$; for $K = 3$, we simulated data with cluster prior probability $P(z_{\mathrm{i}} = 1) = 0.2, P(z_{\mathrm{i}} = 2) = 0.32, P(z_{\mathrm{i}} = 3) = 0.48$. We repeated the same experiments as shown in tables 3.1-3.4 for the unbalanced cluster priors, and reported the findings in tables 3.5-3.8. Similar to the bal-

anced cases presented in tables 3.1-3.4, CAT demonstrated the most desirable performances compared with all other methods.

3.4 Experiments by using Real-world Data

3.4.1 Description of Real-world Data

Colon adenocarcinoma is known as a heterogeneous disease with different molecular subtypes [106]. CREB3L1, or cyclic AMP responsive element-binding protein 3-like protein 1, is an important transcription factor that can suppress cell cycle [121], [122]. The regulation of CREB3L1 is largely accomplished through epigenetic mechanisms in cancer and other disease [122]. We checked the latent relationship between CREB3L1 and one of its epigenetic regulators, cg16012690, in colon cancer. We collected the gene expression profile of CREB3L1 and the methylation profile of cg16012690 on 299 colon adenocarcinoma patients from the Cancer Genome Atlas (TCGA) cohort [123].

3.4.2 Results

We fitted the data using CAT, MLE, CWM, TLE, MIXBI, MIXL and MIXT with $K = 2$, and the two regression lines are colored in red and shown in the top panel of Figure 3.1. We could see that even though the regression lines fitted by the methods are slightly different, they all seem to fit well the data points. In order to compare the robustness of the six methods, we added 10 high leverage points at $x = 0$ (middle panel of Figure 3.1), and 10 high leverage points at $x = 0.7$ (bottom panel of Figure 3.1), in both cases of which, y is a random draw from uniform distribution $U(18, 20)$. We then refitted the contaminated data using the seven methods for the two scenarios. For CWM and TLE, we give them both the true outlier proportion (CWM1 and TLE1), as well as 5% increase of the true outlier proportion (CWM2 and TLE2). In the middle and bottom panels of Figure 3.1, the regression lines fitted using the intact data were shown as red lines, and those refitted using the contaminated data were shown as dashed blue lines. Clearly, CAT was robust to the added high leverage outliers in both scenarios, as the lines fitted before and after contamination overlap. For contaminated data at $x = 0$ (middle panel), all methods except

Table 3.1. Experiment Setting: K=2, P=2, N=200, Balanced

Scenario	CAT	MLE	TLE	CWM1	CWM2	MIXBI	MIXL	MIXT
				Scenario 1: $\epsilon \sim N(0,1)$				
1	-0.030(0.130)	-0.030(0.120)	-0.030(0.120)	0.060(0.450)	0.070(0.360)	-0.030(0.130)	-0.080(0.210)	-0.030(0.130)
	-0.010(0.130)	-0.010(0.120)	-0.010(0.120)	0.010(0.310)	-0.020(0.270)	-0.010(0.120)	-0.020(0.130)	-0.010(0.130)
	0.010(0.130)	0.010(0.120)	0.010(0.120)	0.350(0.720)	0.370(0.870)	0.010(0.120)	0.040(0.160)	0.020(0.120)
	-0.010(0.110)	0.000(0.100)	0.000(0.100)	-0.140(0.450)	-0.190(0.450)	0.000(0.100)	-0.030(0.130)	0.010(0.100)
	0.010(0.140)	0.010(0.130)	0.010(0.130)	0.030(0.250)	-0.010(0.300)	0.010(0.130)	-0.010(0.170)	0.020(0.140)
	-0.020(0.110)	-0.030(0.110)	-0.030(0.110)	0.010(0.230)	0.040(0.230)	-0.020(0.110)	-0.020(0.140)	-0.030(0.110)
	-0.000(0.050)	-0.000(0.040)	-0.000(0.040)	0.040(0.140)	0.020(0.140)	-0.000(0.040)	-0.010(0.040)	-0.010(0.040)
	0.000(0.050)	0.000(0.040)	0.000(0.040)	-0.040(0.140)	-0.020(0.140)	0.000(0.040)	0.010(0.040)	0.010(0.040)
				Scenario 2: $\epsilon \sim t_1$				
2	-0.030(0.820)	-30.780(283.320)	-30.780(283.320)	-39.680(438.250)	-0.260(2.480)	0.030(0.280)	0.290(5.070)	8.300(52.430)
	0.050(0.680)	36.360(302.980)	36.360(302.980)	-1.130(30.160)	-0.130(2.880)	-0.020(0.230)	1.780(13.920)	0.290(16.510)
	0.290(0.600)	-32.720(152.040)	-32.720(152.040)	-39.430(262.650)	1.200(1.650)	0.780(0.860)	-1.490(11.580)	-9.540(35.410)
	-0.030(0.290)	73.300(323.140)	73.300(323.140)	9.680(30.910)	-0.100(4.540)	0.030(0.330)	3.110(14.440)	5.070(16.660)
	0.140(1.480)	-8.530(461.870)	-8.530(461.870)	-95.450(905.720)	-0.170(1.810)	0.040(0.320)	0.280(10.860)	4.280(30.950)
	0.010(0.310)	-47.240(476.990)	-47.240(476.990)	-3.780(40.800)	-0.700(5.450)	0.010(0.260)	-0.380(7.110)	3.450(18.460)
	0.060(0.120)	0.100(0.420)	0.100(0.420)	0.010(0.410)	0.070(0.220)	0.170(0.180)	0.050(0.350)	0.020(0.410)
	-0.060(0.120)	-0.100(0.420)	-0.100(0.420)	-0.010(0.410)	-0.070(0.220)	-0.170(0.180)	-0.050(0.350)	-0.020(0.410)
				Scenario 3: $\epsilon \sim t_3$				
3	-0.020(0.170)	-0.040(0.290)	-0.040(0.290)	0.060(0.370)	-0.070(0.430)	-0.020(0.170)	-0.040(0.180)	-0.030(0.270)
	-0.020(0.140)	-0.030(0.180)	-0.030(0.180)	0.010(0.260)	0.030(0.440)	-0.020(0.150)	-0.030(0.160)	-0.030(0.180)
	0.020(0.160)	0.280(0.450)	0.280(0.450)	0.640(0.870)	0.560(0.870)	0.010(0.180)	0.040(0.180)	0.370(0.540)
	0.000(0.150)	0.010(0.190)	0.010(0.190)	-0.320(0.650)	-0.220(0.550)	0.030(0.140)	-0.010(0.160)	0.020(0.190)
	-0.010(0.210)	-0.000(0.270)	-0.000(0.270)	0.020(0.400)	0.020(0.490)	-0.000(0.190)	-0.000(0.200)	-0.010(0.260)
	-0.010(0.150)	0.010(0.190)	0.010(0.190)	0.020(0.280)	-0.030(0.290)	-0.000(0.150)	-0.030(0.170)	0.010(0.190)
	0.020(0.060)	0.030(0.070)	0.030(0.070)	0.050(0.140)	0.040(0.150)	0.010(0.070)	0.010(0.060)	0.050(0.080)
	-0.020(0.060)	-0.030(0.070)	-0.030(0.070)	-0.050(0.140)	-0.040(0.150)	-0.010(0.070)	-0.010(0.060)	-0.050(0.080)
				Scenario 4: 5% added outliers				
4	-0.100(0.180)	-0.250(0.510)	0.220(1.500)	0.560(6.030)	0.480(3.520)	-0.190(0.160)	-0.290(0.290)	0.200(0.440)
	-0.000(0.130)	0.330(0.340)	0.270(1.010)	0.090(0.500)	-0.160(1.540)	0.060(0.130)	0.140(0.190)	-0.010(0.210)
	-0.080(0.190)	0.280(0.880)	1.780(2.400)	0.230(2.530)	0.530(1.190)	-0.200(0.180)	-0.220(0.380)	0.940(0.630)
	0.020(0.150)	0.040(0.770)	-1.160(2.080)	-0.190(0.820)	-0.000(1.150)	0.090(0.160)	0.180(0.180)	-0.500(0.590)
	-0.110(0.240)	-0.440(1.020)	-0.340(1.840)	-0.140(1.300)	-0.330(2.110)	-0.310(0.240)	-0.560(0.420)	0.520(0.710)
	0.040(0.170)	0.710(0.640)	0.380(1.150)	0.120(0.770)	0.080(0.860)	0.140(0.170)	0.350(0.290)	0.060(0.400)
	0.010(0.050)	0.000(0.150)	0.060(0.170)	-0.010(0.160)	0.040(0.200)	0.010(0.050)	-0.000(0.070)	-0.060(0.150)
	-0.010(0.050)	-0.000(0.150)	-0.060(0.170)	0.010(0.160)	-0.040(0.200)	-0.010(0.050)	0.000(0.070)	0.060(0.150)
				Scenario 5: 10% added outliers				
5	-0.000(0.140)	-0.120(0.450)	-0.030(1.120)	-0.030(0.210)	0.080(0.570)	-0.030(0.140)	-0.100(0.170)	0.080(0.380)
	0.020(0.120)	0.220(0.310)	0.020(1.140)	-0.110(1.270)	-0.140(0.590)	0.040(0.110)	0.080(0.160)	0.050(0.150)
	-0.020(0.160)	0.340(0.780)	1.990(2.660)	0.340(0.760)	0.570(0.910)	-0.070(0.160)	-0.080(0.190)	0.500(0.630)
	0.000(0.120)	-0.050(0.580)	-1.390(2.430)	-0.110(0.430)	-0.230(0.530)	0.030(0.120)	0.060(0.120)	-0.190(0.460)
	-0.040(0.150)	-0.240(0.740)	-0.220(2.000)	-0.010(0.340)	-0.090(0.590)	-0.120(0.150)	-0.260(0.200)	0.130(0.670)
	0.020(0.120)	0.410(0.670)	-0.580(1.910)	0.150(1.130)	0.060(0.600)	0.040(0.120)	0.150(0.160)	0.040(0.200)
	0.000(0.050)	-0.030(0.160)	0.070(0.080)	0.030(0.140)	0.030(0.160)	0.010(0.050)	0.000(0.050)	-0.040(0.120)
	-0.000(0.050)	0.030(0.160)	-0.070(0.080)	-0.030(0.140)	-0.030(0.160)	-0.010(0.050)	-0.000(0.050)	0.040(0.120)

Table 3.2. Experiment Setting: K=2, P=2,N=400, Balanced

Scenario	CAT	MLE	TLE	CWM1	CWM2	MIXBI	MIXL	MIXT
				Scenario 1: $\epsilon \sim N(0,1)$				
1	-0.000(0.110)	0.000(0.100)	0.000(0.100)	0.020(0.110)	-0.000(0.380)	-0.000(0.100)	-0.000(0.130)	0.000(0.100)
	0.000(0.090)	0.010(0.080)	0.010(0.080)	0.020(0.100)	0.010(0.110)	0.010(0.080)	0.010(0.100)	0.010(0.080)
	0.000(0.080)	0.000(0.080)	0.000(0.080)	0.250(0.610)	0.290(0.570)	0.000(0.080)	0.020(0.100)	0.010(0.080)
	-0.020(0.080)	-0.010(0.070)	-0.010(0.070)	-0.080(0.310)	-0.070(0.260)	-0.010(0.080)	-0.030(0.100)	-0.000(0.090)
	0.010(0.100)	0.000(0.080)	0.000(0.080)	0.020(0.110)	0.010(0.150)	0.000(0.090)	-0.000(0.120)	0.000(0.080)
	-0.010(0.080)	-0.000(0.080)	-0.000(0.080)	0.000(0.100)	-0.000(0.110)	-0.010(0.080)	-0.010(0.090)	-0.000(0.080)
	0.000(0.030)	0.000(0.030)	0.000(0.030)	0.020(0.080)	0.020(0.100)	0.000(0.030)	-0.000(0.030)	0.000(0.030)
	-0.000(0.030)	-0.000(0.030)	-0.000(0.030)	-0.020(0.080)	-0.020(0.100)	-0.000(0.030)	0.000(0.030)	-0.000(0.030)
				Scenario 2: $\epsilon \sim t_1$				
2	-0.020(0.220)	-0.710(43.750)	-0.710(43.750)	1.630(44.370)	-0.010(0.720)	0.000(0.230)	-7.310(84.040)	1.000(12.990)
	-0.010(0.130)	-4.470(73.010)	-4.470(73.010)	-9.330(92.790)	-0.110(1.090)	0.010(0.130)	-20.000(207.170)	-0.190(15.170)
	0.330(0.450)	-16.440(92.360)	-16.440(92.360)	-28.470(84.080)	1.240(0.950)	0.930(0.650)	-12.320(67.180)	-2.880(13.170)
	0.000(0.160)	9.220(33.280)	9.220(33.280)	13.480(41.630)	-0.620(1.330)	0.060(0.160)	31.680(296.340)	6.010(13.120)
	0.010(0.230)	3.840(67.430)	3.840(67.430)	0.260(45.000)	0.160(1.150)	-0.010(0.210)	-10.200(83.010)	2.250(17.710)
	-0.020(0.130)	-2.420(63.500)	-2.420(63.500)	-4.210(61.080)	-0.150(1.200)	-0.010(0.130)	16.700(183.670)	0.290(18.170)
	0.100(0.110)	0.050(0.430)	0.050(0.430)	0.070(0.440)	0.060(0.190)	0.220(0.120)	0.030(0.390)	0.090(0.420)
	-0.100(0.110)	-0.050(0.430)	-0.050(0.430)	-0.070(0.440)	-0.060(0.190)	-0.220(0.120)	-0.030(0.390)	-0.090(0.420)
				Scenario 3: $\epsilon \sim t_3$				
3	-0.010(0.130)	0.010(0.220)	0.010(0.220)	0.050(0.570)	-0.000(0.270)	-0.000(0.130)	-0.010(0.140)	0.020(0.220)
	0.010(0.110)	0.010(0.140)	0.010(0.140)	-0.030(0.270)	0.080(0.820)	0.010(0.120)	0.020(0.120)	0.010(0.130)
	0.010(0.120)	0.200(0.350)	0.200(0.350)	0.800(0.760)	0.570(0.850)	0.020(0.160)	0.020(0.130)	0.300(0.400)
	0.010(0.100)	0.010(0.150)	0.010(0.150)	-0.250(0.570)	-0.110(0.770)	0.030(0.090)	0.010(0.100)	0.030(0.130)
	-0.010(0.130)	-0.010(0.190)	-0.010(0.190)	0.010(0.740)	-0.030(0.210)	-0.010(0.120)	-0.010(0.120)	-0.010(0.210)
	0.000(0.110)	-0.000(0.110)	-0.000(0.110)	-0.090(0.530)	-0.110(0.660)	0.000(0.100)	0.010(0.110)	-0.000(0.120)
	0.020(0.040)	0.020(0.050)	0.020(0.050)	0.060(0.160)	0.070(0.160)	0.020(0.050)	0.000(0.040)	0.040(0.060)
	-0.020(0.040)	-0.020(0.050)	-0.020(0.050)	-0.060(0.160)	-0.070(0.160)	-0.020(0.050)	-0.000(0.040)	-0.040(0.060)
				Scenario 4: 5% added outliers				
4	-0.050(0.120)	-0.100(0.480)	-0.010(0.740)	0.390(2.590)	0.170(1.220)	-0.170(0.120)	-0.280(0.150)	0.320(0.280)
	0.010(0.090)	0.270(0.320)	-0.010(0.560)	-1.040(5.440)	-0.290(2.220)	0.070(0.090)	0.170(0.120)	0.030(0.120)
	-0.080(0.130)	0.330(0.790)	2.160(1.730)	0.330(1.710)	0.100(1.720)	-0.220(0.130)	-0.290(0.180)	1.010(0.400)
	0.020(0.090)	0.040(0.650)	-1.370(1.540)	0.300(2.290)	0.480(5.860)	0.090(0.100)	0.180(0.140)	-0.500(0.550)
	-0.130(0.180)	-0.290(0.920)	-0.130(1.140)	-0.230(0.940)	-0.000(0.720)	-0.350(0.170)	-0.590(0.200)	0.580(0.490)
	0.010(0.100)	0.570(0.610)	0.030(1.130)	0.210(1.540)	-0.400(4.380)	0.130(0.120)	0.330(0.150)	0.030(0.180)
	0.010(0.040)	-0.010(0.120)	0.060(0.160)	0.050(0.200)	0.000(0.160)	0.000(0.040)	-0.010(0.050)	-0.080(0.130)
	-0.010(0.040)	0.010(0.120)	-0.060(0.160)	-0.050(0.200)	-0.000(0.160)	-0.000(0.040)	0.010(0.050)	0.080(0.130)
				Scenario 5: 10% added outliers				
5	-0.010(0.110)	-0.040(0.380)	-0.160(1.290)	-0.050(0.360)	-0.030(0.250)	-0.050(0.110)	-0.110(0.130)	0.180(0.250)
	0.010(0.090)	0.210(0.260)	0.010(1.220)	-0.020(0.600)	-0.010(0.350)	0.030(0.090)	0.090(0.110)	0.040(0.160)
	-0.010(0.110)	0.440(0.760)	1.480(2.270)	0.330(0.770)	0.280(0.710)	-0.060(0.120)	-0.090(0.120)	0.730(0.590)
	0.010(0.080)	-0.070(0.610)	-1.400(2.200)	-0.090(0.440)	-0.100(0.380)	0.030(0.080)	0.070(0.100)	-0.210(0.400)
	-0.010(0.120)	-0.130(0.700)	0.100(1.660)	-0.030(0.310)	-0.010(0.180)	-0.100(0.130)	-0.200(0.130)	0.370(0.430)
	0.010(0.080)	0.360(0.450)	0.250(1.430)	0.050(0.500)	0.020(0.180)	0.050(0.090)	0.140(0.110)	0.040(0.310)
	0.000(0.030)	-0.020(0.130)	0.050(0.160)	0.030(0.120)	0.010(0.110)	0.000(0.030)	-0.000(0.040)	-0.020(0.120)
	-0.000(0.030)	0.020(0.130)	-0.050(0.160)	-0.030(0.120)	-0.010(0.110)	-0.000(0.030)	0.000(0.040)	0.020(0.120)

Table 3.3. Experiment Setting: K=3, P=1,N=200, Balanced

Scenario	CAT	MLE	TLE	CWM1	CWM2	MIXBI	MIXL	MIXT
				Scenario 1: $\epsilon \sim N(0,1)$				
1	-0.040(0.290)	-0.080(0.430)	-0.080(0.430)	-0.330(0.510)	-0.440(0.470)	-0.060(0.330)	-0.180(0.320)	-0.170(0.430)
	-0.020(0.170)	0.030(0.250)	0.030(0.250)	0.010(0.330)	0.030(0.440)	-0.020(0.170)	0.000(0.210)	0.040(0.220)
	-0.020(0.300)	0.190(0.550)	0.190(0.550)	0.770(0.710)	0.860(0.720)	0.090(0.390)	0.270(0.410)	0.310(0.720)
	0.040(0.330)	0.090(0.530)	0.090(0.530)	0.360(0.720)	0.360(0.880)	-0.010(0.240)	-0.040(0.260)	0.040(0.430)
	0.020(0.160)	0.030(0.160)	0.030(0.160)	-0.100(0.400)	-0.180(0.510)	0.030(0.140)	-0.000(0.170)	0.070(0.280)
	0.030(0.270)	0.330(0.890)	0.330(0.890)	0.600(1.200)	0.740(1.160)	0.120(0.560)	0.170(0.380)	0.520(1.010)
	0.010(0.080)	0.030(0.100)	0.030(0.100)	0.060(0.140)	0.050(0.140)	0.010(0.100)	0.010(0.070)	0.050(0.120)
	-0.010(0.060)	-0.030(0.080)	-0.030(0.080)	-0.040(0.110)	-0.060(0.120)	-0.010(0.070)	-0.000(0.060)	-0.030(0.090)
	-0.000(0.090)	-0.000(0.080)	-0.000(0.080)	-0.010(0.130)	0.000(0.130)	-0.000(0.100)	-0.000(0.090)	-0.020(0.110)
				Scenario 2: $\epsilon \sim t_1$				
2	0.080(1.380)	7.190(36.260)	7.190(36.260)	7.140(75.550)	-0.360(1.470)	-0.240(0.740)	14.400(92.190)	-0.230(8.140)
	0.140(0.590)	10.580(127.780)	10.580(127.780)	13.280(66.830)	-0.190(1.530)	-0.070(0.350)	29.740(172.710)	2.110(14.780)
	0.500(1.030)	-22.530(74.610)	-22.530(74.610)	-12.070(49.140)	0.100(3.500)	0.450(2.540)	-41.960(132.220)	-1.460(5.950)
	0.360(0.960)	-9.090(27.760)	-9.090(27.760)	-43.040(188.940)	0.240(2.750)	0.570(0.890)	-39.440(208.580)	-4.210(12.880)
	-0.010(0.460)	48.320(145.340)	48.320(145.340)	21.960(98.690)	-0.190(1.670)	-0.190(0.640)	42.380(142.150)	6.830(18.380)
	0.480(1.280)	-12.550(82.170)	-12.550(82.170)	-2.500(29.920)	0.050(3.620)	0.540(1.080)	-27.790(142.750)	0.690(3.730)
	0.040(0.130)	0.060(0.270)	0.060(0.270)	0.030(0.250)	0.060(0.160)	0.050(0.190)	0.080(0.310)	0.020(0.210)
	-0.090(0.090)	-0.110(0.260)	-0.110(0.260)	-0.120(0.230)	-0.090(0.160)	-0.090(0.170)	-0.130(0.270)	-0.080(0.180)
	0.050(0.130)	0.050(0.260)	0.050(0.260)	0.090(0.270)	0.030(0.170)	0.040(0.190)	0.050(0.290)	0.060(0.200)
				Scenario 3: $\epsilon \sim t_3$				
3	0.040(0.530)	0.130(1.600)	0.130(1.600)	-0.120(0.900)	-0.390(0.640)	-0.170(0.470)	-0.260(0.450)	-0.250(0.590)
	0.070(0.390)	0.310(2.620)	0.310(2.620)	-0.090(0.480)	0.050(0.420)	0.040(0.240)	0.030(0.290)	0.070(0.300)
	0.130(0.600)	0.520(1.580)	0.520(1.580)	0.720(0.890)	0.780(0.650)	0.340(0.670)	0.400(0.660)	0.570(0.760)
	0.110(0.390)	0.120(1.520)	0.120(1.520)	0.480(1.080)	0.330(0.740)	0.020(0.360)	-0.020(0.380)	0.220(0.770)
	0.000(0.190)	0.460(1.650)	0.460(1.650)	-0.230(0.590)	-0.220(0.460)	0.070(0.230)	0.010(0.240)	0.050(0.330)
	0.200(0.780)	0.310(2.030)	0.310(2.030)	0.500(1.190)	0.490(1.010)	0.340(0.880)	0.190(0.810)	0.550(1.310)
	0.010(0.100)	0.010(0.190)	0.010(0.190)	0.030(0.150)	0.020(0.150)	0.010(0.150)	0.010(0.110)	0.030(0.140)
	-0.030(0.080)	-0.070(0.130)	-0.070(0.130)	-0.050(0.130)	-0.050(0.120)	-0.040(0.090)	-0.010(0.080)	-0.060(0.110)
	0.030(0.110)	0.060(0.170)	0.060(0.170)	0.020(0.140)	0.030(0.150)	0.040(0.160)	0.000(0.110)	0.030(0.140)
				Scenario 4: 5% added outliers				
4	-0.420(0.480)	-1.300(0.890)	-0.630(1.590)	0.180(4.760)	-0.470(1.010)	-0.910(0.630)	-1.150(0.840)	-0.810(0.450)
	0.110(0.420)	0.720(0.770)	-0.450(1.530)	-0.270(1.360)	-0.110(0.710)	0.260(0.500)	0.380(0.560)	0.140(0.330)
	0.300(0.590)	0.190(1.690)	0.980(1.370)	0.880(0.760)	0.890(0.690)	0.510(1.070)	0.290(1.370)	1.160(0.650)
	-0.200(0.550)	-0.430(1.730)	1.520(2.690)	-0.440(3.050)	0.250(1.020)	-0.330(1.200)	-0.650(1.360)	0.360(0.730)
	0.180(0.560)	0.890(0.970)	-0.910(2.240)	0.170(1.140)	0.010(0.810)	0.450(0.670)	0.480(0.610)	0.170(0.380)
	0.260(0.900)	-0.780(2.410)	0.900(2.280)	0.390(1.480)	0.630(1.140)	0.340(1.750)	-0.670(2.650)	0.430(1.290)
	0.030(0.120)	0.110(0.210)	0.010(0.160)	0.020(0.170)	0.050(0.130)	0.120(0.140)	0.050(0.170)	0.120(0.170)
	-0.010(0.100)	-0.090(0.120)	-0.020(0.160)	-0.070(0.140)	-0.070(0.140)	-0.040(0.100)	-0.030(0.140)	-0.040(0.090)
	-0.020(0.100)	-0.020(0.250)	0.010(0.150)	0.050(0.150)	0.020(0.160)	-0.080(0.130)	-0.020(0.160)	-0.080(0.180)
				Scenario 5: 10% added outliers				
5	-0.140(0.410)	-0.810(0.950)	-0.620(1.270)	0.720(8.710)	-0.270(1.140)	-0.440(0.370)	-0.770(0.590)	-0.600(0.610)
	0.030(0.170)	0.530(0.830)	-0.870(1.160)	-0.030(0.410)	-0.060(0.560)	0.090(0.270)	0.390(0.580)	0.190(0.530)
	0.090(0.420)	0.390(1.620)	0.790(1.350)	0.700(0.970)	0.670(0.660)	0.340(0.600)	0.590(1.240)	0.890(1.110)
	-0.160(0.390)	0.330(1.570)	0.930(2.040)	-0.420(5.000)	0.400(1.080)	-0.100(0.360)	-0.140(0.930)	0.550(1.050)
	0.030(0.250)	0.620(1.090)	-2.060(2.550)	0.000(0.440)	-0.120(0.530)	0.150(0.460)	0.430(0.680)	0.190(0.540)
	0.090(0.520)	-0.590(2.560)	0.350(2.110)	0.480(1.290)	0.400(1.200)	0.270(0.770)	0.080(2.460)	-0.020(2.020)
	-0.000(0.090)	0.110(0.230)	-0.010(0.180)	0.050(0.130)	0.040(0.150)	0.060(0.130)	0.090(0.150)	0.060(0.220)
	0.000(0.060)	-0.080(0.150)	-0.000(0.200)	-0.060(0.110)	-0.050(0.130)	-0.010(0.090)	-0.070(0.170)	-0.040(0.130)
	-0.000(0.100)	-0.020(0.250)	0.010(0.170)	0.000(0.120)	0.010(0.140)	-0.060(0.120)	-0.020(0.150)	-0.010(0.240)

Table 3.4. Experiment Setting: K=3, P=1,N=400, Balanced

Scenario	CAT	MLE	TLE	CWM1	CWM2	MIXBI	MIXL	MIXT
				Scenario 1: $\epsilon \sim N(0,1)$				
1	-0.000(0.220)	-0.080(0.320)	-0.080(0.320)	-0.310(0.580)	-0.370(0.500)	-0.030(0.200)	-0.140(0.230)	-0.080(0.250)
	0.010(0.170)	0.020(0.180)	0.020(0.180)	-0.040(0.360)	-0.050(0.400)	-0.000(0.110)	-0.000(0.130)	0.000(0.110)
	0.040(0.360)	0.160(0.510)	0.160(0.510)	0.900(0.660)	0.800(0.730)	0.010(0.200)	0.160(0.250)	0.070(0.310)
	0.090(0.180)	0.100(0.450)	0.100(0.450)	0.530(0.780)	0.530(0.870)	0.010(0.150)	-0.030(0.190)	-0.020(0.170)
	0.010(0.110)	0.020(0.120)	0.020(0.120)	-0.040(0.390)	-0.090(0.510)	0.020(0.090)	0.010(0.110)	0.020(0.130)
	-0.020(0.460)	0.210(0.760)	0.210(0.760)	0.630(1.130)	0.700(1.180)	-0.030(0.180)	0.070(0.210)	0.090(0.390)
	0.010(0.070)	0.020(0.100)	0.020(0.100)	0.060(0.150)	0.080(0.150)	0.000(0.060)	0.000(0.050)	0.010(0.070)
	-0.010(0.060)	-0.020(0.060)	-0.020(0.060)	-0.060(0.110)	-0.070(0.130)	0.000(0.040)	0.000(0.040)	-0.000(0.050)
	0.000(0.070)	-0.000(0.070)	-0.000(0.070)	0.000(0.110)	-0.010(0.110)	-0.000(0.070)	-0.000(0.060)	-0.010(0.070)
				Scenario 2: $\epsilon \sim t_1$				
2	-0.060(1.060)	52.300(421.560)	52.300(421.560)	210.880(2052.960)	0.420(3.520)	-0.400(0.500)	215.860(1933.390)	2.760(12.470)
	0.090(0.510)	14.380(125.220)	14.380(125.220)	-1.850(69.520)	0.170(1.950)	-0.030(0.300)	192.110(1939.830)	0.430(6.240)
	0.850(0.750)	-37.930(147.630)	-37.930(147.630)	-21.000(71.890)	0.670(1.300)	0.920(0.670)	-66.660(389.220)	-1.250(5.190)
	0.500(0.920)	-81.630(437.820)	-81.630(437.820)	-435.630(4236.130)	0.150(3.780)	0.520(0.870)	291.150(3140.910)	-4.990(13.140)
	0.020(0.350)	74.650(183.210)	74.650(183.210)	38.330(147.010)	-0.440(0.970)	-0.100(0.410)	390.210(3137.850)	3.780(14.810)
	0.660(1.130)	-13.880(108.340)	-13.880(108.340)	-0.010(30.140)	0.510(1.260)	0.580(0.860)	12.430(223.510)	-0.460(5.600)
	0.030(0.150)	0.070(0.320)	0.070(0.320)	0.030(0.260)	0.040(0.170)	0.030(0.160)	0.020(0.310)	0.060(0.230)
	-0.090(0.100)	-0.140(0.270)	-0.140(0.270)	-0.070(0.300)	-0.080(0.150)	-0.110(0.180)	-0.120(0.300)	-0.100(0.180)
	0.060(0.150)	0.070(0.310)	0.070(0.310)	0.040(0.270)	0.040(0.160)	0.080(0.200)	0.100(0.340)	0.040(0.220)
				Scenario 3: $\epsilon \sim t_3$				
3	0.100(0.320)	-0.450(1.120)	-0.450(1.120)	-0.420(0.740)	-0.340(0.560)	-0.110(0.370)	-0.160(0.310)	-0.600(0.400)
	-0.040(0.160)	0.140(1.180)	0.140(1.180)	-0.160(0.970)	0.020(0.370)	-0.030(0.170)	-0.030(0.170)	-0.020(0.470)
	-0.000(0.250)	-1.620(20.930)	-1.620(20.930)	0.860(1.100)	0.690(0.670)	0.130(0.450)	-1.840(20.780)	1.000(0.790)
	0.150(0.430)	0.200(2.030)	0.200(2.030)	0.570(0.740)	0.600(0.780)	0.030(0.330)	-0.030(0.210)	0.430(0.600)
	-0.010(0.130)	0.250(1.400)	0.250(1.400)	-0.190(1.330)	-0.220(0.420)	0.030(0.130)	0.010(0.170)	0.020(0.500)
	0.010(0.360)	-2.110(28.490)	-2.110(28.490)	0.610(1.250)	0.490(1.060)	0.150(0.490)	-2.680(28.370)	1.240(1.220)
	0.010(0.070)	0.090(0.190)	0.090(0.190)	0.040(0.150)	0.070(0.150)	0.030(0.120)	0.010(0.080)	0.140(0.140)
	-0.020(0.060)	-0.090(0.130)	-0.090(0.130)	-0.060(0.130)	-0.070(0.120)	-0.030(0.080)	-0.010(0.060)	-0.090(0.120)
	0.020(0.070)	-0.010(0.180)	-0.010(0.180)	0.020(0.150)	0.010(0.140)	-0.000(0.140)	-0.000(0.080)	-0.050(0.130)
				Scenario 4: 5% added outliers				
4	-0.400(0.490)	-1.280(0.940)	-0.540(0.920)	0.310(4.980)	-0.510(0.420)	-0.850(0.570)	-1.120(0.670)	-0.770(0.420)
	0.070(0.380)	0.620(0.640)	-0.370(1.010)	-0.380(1.010)	-0.110(0.420)	0.250(0.470)	0.570(0.630)	0.120(0.240)
	0.270(0.470)	0.460(1.670)	1.060(1.240)	0.990(0.590)	0.880(0.780)	0.500(0.820)	0.580(1.500)	1.410(0.600)
	-0.330(0.630)	-0.530(1.960)	1.950(1.760)	-0.430(3.360)	0.460(0.770)	-0.370(1.130)	-0.520(1.200)	0.440(0.730)
	0.110(0.400)	0.770(0.790)	-1.210(1.700)	0.240(1.010)	-0.080(0.460)	0.380(0.570)	0.680(0.690)	0.130(0.210)
	0.140(0.560)	-0.740(2.460)	0.910(1.960)	0.550(1.370)	0.690(1.260)	0.330(1.280)	-0.200(2.920)	0.670(1.120)
	0.020(0.090)	0.100(0.220)	0.030(0.140)	0.050(0.150)	0.080(0.140)	0.110(0.120)	0.080(0.180)	0.150(0.150)
	-0.010(0.060)	-0.070(0.100)	-0.030(0.160)	-0.060(0.130)	-0.070(0.130)	-0.010(0.090)	-0.070(0.160)	-0.030(0.070)
	-0.010(0.080)	-0.030(0.230)	0.000(0.150)	0.010(0.140)	-0.010(0.130)	-0.100(0.110)	-0.010(0.150)	-0.110(0.160)
				Scenario 5: 10% added outliers				
5	-0.060(0.270)	-1.060(0.950)	-0.660(1.410)	0.100(5.540)	-0.350(0.570)	-0.530(0.380)	-0.870(0.600)	-0.670(0.640)
	0.030(0.160)	0.320(0.480)	-0.270(1.340)	-0.070(0.380)	-0.090(0.330)	0.050(0.150)	0.260(0.460)	0.130(0.420)
	0.040(0.260)	0.580(1.480)	1.330(1.260)	0.850(0.600)	0.770(0.630)	0.310(0.530)	0.560(0.900)	1.270(0.550)
	0.010(0.210)	-0.220(1.970)	1.080(2.750)	-0.290(3.430)	0.370(0.790)	-0.140(0.710)	-0.400(0.930)	0.610(0.920)
	0.020(0.120)	0.310(0.420)	-1.390(2.090)	0.000(0.440)	-0.070(0.480)	0.110(0.250)	0.320(0.700)	0.160(0.430)
	-0.030(0.370)	-0.660(2.260)	0.740(2.180)	0.470(1.200)	0.480(1.040)	0.170(0.600)	-0.060(1.820)	0.400(1.220)
	0.000(0.060)	0.090(0.250)	0.040(0.160)	0.050(0.140)	0.040(0.150)	0.060(0.120)	0.030(0.160)	0.100(0.220)
	-0.000(0.040)	-0.040(0.110)	-0.010(0.170)	-0.030(0.120)	-0.040(0.130)	0.010(0.060)	-0.030(0.140)	-0.030(0.100)
	-0.000(0.070)	-0.050(0.250)	-0.030(0.180)	-0.020(0.120)	0.010(0.140)	-0.070(0.120)	-0.000(0.140)	-0.070(0.220)

Table 3.5. Experiment Setting: K=2, P=2, N=200, Unbalanced

Scenario	CAT	MLE	TLE	CWM1	CWM2	MIXBI	MIXL	MIXT
				Scenario 1: $\epsilon \sim N(0,1)$				
1	0.010(0.170)	-0.000(0.160)	-0.000(0.160)	0.050(0.410)	0.000(0.470)	0.000(0.170)	-0.050(0.230)	-0.000(0.170)
	0.030(0.130)	0.020(0.120)	0.020(0.120)	-0.000(0.220)	0.150(1.280)	0.020(0.120)	0.030(0.140)	0.010(0.120)
	0.010(0.160)	0.010(0.130)	0.010(0.130)	0.460(0.780)	0.780(1.020)	0.010(0.140)	0.070(0.230)	0.020(0.140)
	-0.000(0.120)	0.000(0.100)	0.000(0.100)	-0.100(0.370)	-0.060(1.110)	0.010(0.100)	-0.010(0.130)	0.010(0.100)
	0.000(0.210)	-0.000(0.170)	-0.000(0.170)	0.010(0.200)	-0.030(0.530)	0.000(0.180)	-0.030(0.230)	0.010(0.190)
	0.000(0.120)	0.010(0.110)	0.010(0.110)	-0.010(0.250)	0.030(0.900)	0.010(0.110)	0.010(0.140)	0.010(0.120)
	0.000(0.050)	0.000(0.050)	0.000(0.050)	0.040(0.120)	0.050(0.150)	0.000(0.050)	-0.000(0.050)	0.000(0.050)
	-0.000(0.050)	-0.000(0.050)	-0.000(0.050)	-0.040(0.120)	-0.050(0.150)	-0.000(0.050)	0.000(0.050)	-0.000(0.050)
				Scenario 2: $\epsilon \sim t_1$				
2	0.000(0.560)	-202.260(2012.020)	-202.260(2012.020)	-20.150(279.560)	0.270(2.540)	0.010(0.330)	-571.250(5687.290)	1.360(15.460)
	0.030(0.230)	21.480(202.920)	21.480(202.920)	11.350(116.180)	-0.370(2.560)	0.010(0.200)	-0.080(23.720)	-3.410(29.910)
	0.450(0.690)	-1200.700(11913.540)	-1200.700(11913.540)	-49.390(178.770)	1.220(2.440)	1.180(0.800)	-250.420(2486.920)	-5.820(27.590)
	-0.020(0.230)	38.960(160.180)	38.960(160.180)	30.400(109.180)	-0.380(1.410)	0.030(0.220)	5.710(27.100)	11.100(30.450)
	-0.010(0.620)	-152.190(1483.380)	-152.190(1483.380)	-20.880(297.780)	-0.170(3.470)	-0.000(0.330)	196.580(1964.350)	-1.490(17.740)
	-0.010(0.230)	32.330(156.300)	32.330(156.300)	19.390(231.390)	-0.160(2.420)	-0.020(0.220)	-1.690(28.180)	-0.850(25.270)
	0.110(0.110)	0.130(0.420)	0.130(0.420)	0.110(0.420)	0.080(0.200)	0.250(0.130)	0.040(0.350)	0.150(0.410)
	-0.110(0.110)	-0.130(0.420)	-0.130(0.420)	-0.110(0.420)	-0.080(0.200)	-0.250(0.130)	-0.040(0.350)	-0.150(0.410)
				Scenario 3: $\epsilon \sim t_3$				
3	-0.040(0.170)	-0.020(0.260)	-0.020(0.260)	0.020(1.060)	-0.050(0.530)	-0.040(0.170)	-0.100(0.260)	-0.010(0.270)
	-0.020(0.150)	0.090(1.100)	0.090(1.100)	-0.010(0.260)	-0.060(0.430)	-0.020(0.150)	-0.030(0.160)	-0.020(0.160)
	0.050(0.180)	0.370(0.580)	0.370(0.580)	0.920(1.110)	0.800(0.980)	0.050(0.240)	0.080(0.220)	0.500(0.610)
	-0.010(0.140)	0.220(1.990)	0.220(1.990)	-0.250(0.550)	-0.080(0.700)	0.020(0.140)	-0.020(0.170)	0.020(0.150)
	0.030(0.220)	0.000(0.340)	0.000(0.340)	-0.120(0.840)	0.040(0.380)	0.000(0.220)	-0.050(0.260)	0.020(0.350)
	-0.010(0.140)	0.150(1.510)	0.150(1.510)	0.020(0.320)	-0.010(0.540)	-0.010(0.130)	-0.010(0.150)	0.010(0.160)
	0.020(0.070)	0.050(0.090)	0.050(0.090)	0.060(0.160)	0.090(0.160)	0.020(0.080)	0.000(0.060)	0.070(0.080)
	-0.020(0.070)	-0.050(0.090)	-0.050(0.090)	-0.060(0.160)	-0.090(0.160)	-0.020(0.080)	-0.000(0.060)	-0.070(0.080)
				Scenario 4: 5% added outliers				
4	-0.070(0.190)	-0.180(0.520)	-0.120(1.270)	-0.030(1.620)	0.050(0.790)	-0.170(0.200)	-0.270(0.330)	0.250(0.430)
	0.020(0.100)	0.330(0.330)	-0.030(1.310)	-0.130(1.990)	0.320(4.680)	0.050(0.090)	0.120(0.130)	0.030(0.160)
	-0.120(0.210)	0.230(0.890)	2.110(2.250)	0.620(1.170)	0.820(1.180)	-0.250(0.190)	-0.250(0.330)	0.970(0.640)
	0.040(0.120)	0.190(0.640)	-0.930(2.170)	-0.030(1.000)	-0.040(0.610)	0.080(0.120)	0.170(0.170)	-0.240(0.420)
	-0.190(0.260)	-0.540(1.060)	-0.450(2.090)	0.350(2.140)	0.070(1.210)	-0.410(0.270)	-0.560(0.550)	0.540(0.860)
	0.030(0.130)	0.690(0.660)	0.040(1.480)	0.200(0.920)	-0.020(3.580)	0.100(0.150)	0.260(0.190)	0.070(0.200)
	0.020(0.050)	0.010(0.130)	0.090(0.160)	0.040(0.190)	0.090(0.220)	0.020(0.040)	-0.000(0.050)	-0.020(0.120)
	-0.020(0.050)	-0.010(0.130)	-0.090(0.160)	-0.040(0.190)	-0.090(0.220)	-0.020(0.040)	0.000(0.050)	0.020(0.120)
				Scenario 5: 10% added outliers				
5	-0.030(0.170)	0.020(0.680)	-0.120(1.470)	-0.030(0.580)	-0.010(0.990)	-0.050(0.160)	-0.120(0.230)	0.170(0.410)
	0.010(0.110)	0.170(0.360)	-0.230(1.530)	-0.020(0.320)	0.040(0.200)	0.020(0.110)	0.070(0.150)	0.010(0.140)
	-0.020(0.150)	0.520(0.840)	2.520(2.590)	0.250(0.700)	0.230(0.950)	-0.060(0.150)	-0.060(0.200)	0.680(0.640)
	-0.010(0.120)	-0.010(0.490)	-0.970(2.370)	-0.110(0.460)	-0.080(0.360)	0.010(0.120)	0.030(0.140)	-0.130(0.270)
	-0.060(0.190)	0.030(1.170)	-0.070(2.110)	0.010(0.760)	0.010(1.730)	-0.140(0.200)	-0.270(0.230)	0.400(0.650)
	-0.010(0.110)	0.340(0.670)	0.030(2.130)	0.020(0.170)	0.030(0.390)	0.010(0.100)	0.110(0.150)	0.010(0.200)
	-0.010(0.050)	-0.040(0.120)	0.120(0.080)	0.020(0.130)	0.010(0.150)	-0.000(0.040)	-0.010(0.050)	-0.040(0.080)
	0.010(0.050)	0.040(0.120)	-0.120(0.080)	-0.020(0.130)	-0.010(0.150)	0.000(0.040)	0.010(0.050)	0.040(0.080)

Table 3.6. Experiment Setting: K=2, P=2, N=400, Unbalanced

Scenario	CAT	MLE	TLE	CWM1	CWM2	MIXBI	MIXL	MIXT
Scenario 1: $\epsilon \sim N(0,1)$								
1	-0.010(0.120)	-0.010(0.100)	-0.010(0.100)	0.010(0.110)	-0.040(0.190)	-0.010(0.110)	-0.020(0.140)	-0.010(0.100)
	0.000(0.080)	0.010(0.070)	0.010(0.070)	-0.000(0.090)	0.010(0.130)	0.000(0.080)	0.000(0.090)	0.010(0.070)
	0.010(0.110)	0.010(0.100)	0.010(0.100)	0.310(0.660)	0.580(0.880)	0.010(0.100)	0.010(0.120)	0.010(0.110)
	0.000(0.070)	0.010(0.060)	0.010(0.060)	-0.080(0.300)	-0.030(0.260)	0.010(0.060)	-0.010(0.080)	0.020(0.070)
	-0.010(0.110)	-0.010(0.100)	-0.010(0.100)	-0.010(0.210)	0.020(0.160)	-0.010(0.100)	0.000(0.110)	-0.010(0.110)
	0.000(0.070)	0.010(0.070)	0.010(0.070)	-0.010(0.110)	-0.020(0.130)	0.010(0.070)	0.000(0.080)	0.010(0.080)
	0.010(0.040)	0.010(0.030)	0.010(0.030)	0.020(0.090)	0.070(0.140)	0.010(0.030)	0.000(0.030)	0.010(0.030)
	-0.010(0.040)	-0.010(0.030)	-0.010(0.030)	-0.020(0.090)	-0.070(0.140)	-0.010(0.030)	-0.000(0.030)	-0.010(0.030)
Scenario 2: $\epsilon \sim t_1$								
2	0.040(0.430)	4.000(58.690)	4.000(58.690)	-3.580(44.630)	-0.100(1.020)	-0.000(0.270)	-0.170(14.690)	-0.450(12.130)
	-0.010(0.110)	-11.560(89.090)	-11.560(89.090)	-20.980(265.540)	0.110(0.580)	-0.000(0.120)	9.410(77.350)	-2.160(19.060)
	0.440(0.500)	-8.230(34.550)	-8.230(34.550)	-14.000(40.110)	1.430(1.620)	1.240(0.590)	-2.780(14.890)	-2.570(8.740)
	-0.020(0.160)	18.410(65.280)	18.410(65.280)	37.620(213.420)	-0.540(0.580)	0.030(0.140)	16.120(84.360)	3.510(9.710)
	0.020(0.280)	-5.230(62.060)	-5.230(62.060)	-1.150(98.320)	-0.230(1.780)	-0.010(0.220)	-0.930(9.590)	-2.280(14.630)
	-0.000(0.100)	-5.450(77.360)	-5.450(77.360)	-11.880(60.880)	-0.050(1.150)	-0.000(0.120)	-0.650(57.100)	0.310(13.440)
	0.120(0.090)	0.160(0.430)	0.160(0.430)	0.080(0.440)	0.100(0.210)	0.270(0.100)	0.080(0.390)	0.090(0.420)
	-0.120(0.090)	-0.160(0.430)	-0.160(0.430)	-0.080(0.440)	-0.100(0.210)	-0.270(0.100)	-0.080(0.390)	-0.090(0.420)
Scenario 3: $\epsilon \sim t_3$								
3	-0.000(0.130)	0.020(0.190)	0.020(0.190)	-0.030(0.220)	-0.030(0.320)	0.010(0.120)	-0.010(0.140)	0.010(0.200)
	0.000(0.100)	-0.000(0.110)	-0.000(0.110)	0.010(0.170)	0.000(0.320)	-0.000(0.090)	0.010(0.100)	-0.000(0.100)
	0.030(0.120)	0.430(0.440)	0.430(0.440)	0.970(0.800)	0.830(0.980)	0.020(0.120)	0.030(0.120)	0.530(0.450)
	-0.000(0.090)	0.040(0.100)	0.040(0.100)	-0.150(0.410)	-0.140(0.360)	0.020(0.090)	-0.010(0.090)	0.040(0.110)
	-0.020(0.130)	0.010(0.190)	0.010(0.190)	0.020(0.230)	0.020(0.280)	-0.020(0.130)	-0.030(0.140)	0.000(0.220)
	0.010(0.100)	0.010(0.100)	0.010(0.100)	0.010(0.390)	0.000(0.200)	0.010(0.090)	0.010(0.100)	0.010(0.100)
	0.020(0.040)	0.040(0.050)	0.040(0.050)	0.080(0.140)	0.070(0.160)	0.020(0.040)	-0.000(0.040)	0.060(0.060)
	-0.020(0.040)	-0.040(0.050)	-0.040(0.050)	-0.080(0.140)	-0.070(0.160)	-0.020(0.040)	0.000(0.040)	-0.060(0.060)
Scenario 4: 5% added outliers								
4	-0.120(0.130)	-0.140(0.530)	0.040(0.790)	-0.330(3.180)	-0.050(0.550)	-0.230(0.140)	-0.310(0.170)	0.320(0.270)
	0.030(0.090)	0.280(0.280)	0.050(0.690)	-0.130(1.760)	-0.560(4.150)	0.070(0.100)	0.160(0.130)	0.030(0.100)
	-0.150(0.160)	0.300(0.750)	2.440(2.180)	0.400(0.990)	0.770(1.240)	-0.290(0.140)	-0.330(0.190)	1.040(0.360)
	0.010(0.090)	0.230(0.400)	-0.960(1.650)	0.070(0.990)	0.270(2.060)	0.060(0.100)	0.140(0.140)	-0.190(0.200)
	-0.190(0.210)	-0.280(1.020)	-0.040(0.860)	0.220(2.490)	0.010(0.570)	-0.410(0.180)	-0.610(0.270)	0.730(0.450)
	0.030(0.090)	0.590(0.580)	0.090(0.820)	0.220(0.910)	0.080(1.810)	0.110(0.100)	0.280(0.160)	0.050(0.100)
	0.010(0.030)	-0.000(0.060)	0.100(0.170)	0.040(0.170)	0.060(0.200)	0.000(0.030)	-0.010(0.040)	-0.030(0.050)
	-0.010(0.030)	0.000(0.060)	-0.100(0.170)	-0.040(0.170)	-0.060(0.200)	-0.000(0.030)	0.010(0.040)	0.030(0.050)
Scenario 5: 10% added outliers								
5	-0.060(0.110)	0.030(0.410)	-0.210(1.510)	-0.010(0.180)	0.030(0.470)	-0.090(0.120)	-0.140(0.130)	0.180(0.280)
	0.020(0.070)	0.150(0.270)	-0.090(1.350)	-0.070(0.740)	0.010(0.150)	0.030(0.070)	0.080(0.100)	0.020(0.090)
	-0.060(0.110)	0.520(0.640)	2.570(2.000)	0.310(0.800)	0.500(0.840)	-0.110(0.120)	-0.120(0.140)	0.660(0.520)
	-0.000(0.080)	-0.000(0.330)	-0.910(2.110)	-0.060(0.240)	-0.070(0.320)	0.020(0.080)	0.050(0.100)	-0.130(0.160)
	-0.080(0.150)	0.010(0.720)	0.220(1.800)	-0.040(0.220)	-0.050(0.310)	-0.160(0.160)	-0.260(0.170)	0.390(0.540)
	0.020(0.090)	0.290(0.410)	0.320(1.720)	-0.050(0.680)	-0.010(0.150)	0.050(0.090)	0.130(0.100)	0.030(0.110)
	0.000(0.040)	-0.020(0.100)	0.140(0.160)	0.030(0.110)	0.040(0.130)	0.000(0.030)	-0.000(0.030)	-0.040(0.060)
	-0.000(0.040)	0.020(0.100)	-0.140(0.160)	-0.030(0.110)	-0.040(0.130)	-0.000(0.030)	0.000(0.030)	0.040(0.060)

Table 3.7. Experiment Setting: K=3, P=1, N=200, Unbalanced

Scenario	CAT	MLE	TLE	CWM1	CWM2	MIXBI	MIXL	MIXT
				Scenario 1: $\epsilon \sim N(0,1)$				
1	-0.220(0.620)	-0.110(0.550)	-0.110(0.550)	-0.750(0.600)	-0.800(0.640)	-0.150(0.510)	-0.440(0.500)	-0.260(0.580)
	0.020(0.220)	-0.010(0.190)	-0.010(0.190)	0.010(0.420)	-0.090(0.530)	-0.000(0.180)	0.030(0.220)	0.010(0.220)
	-0.070(0.350)	0.070(0.360)	0.070(0.360)	0.250(0.660)	0.130(0.490)	0.030(0.280)	0.060(0.270)	0.100(0.430)
	0.290(0.470)	0.090(0.580)	0.090(0.580)	0.590(0.890)	0.810(1.150)	0.050(0.340)	0.060(0.410)	0.100(0.600)
	-0.030(0.250)	0.010(0.210)	0.010(0.210)	-0.240(0.740)	-0.430(0.810)	0.010(0.190)	-0.040(0.210)	0.020(0.300)
	0.060(0.380)	0.130(0.530)	0.130(0.530)	0.280(0.910)	0.170(0.620)	0.080(0.370)	0.090(0.240)	0.260(0.720)
	0.050(0.110)	0.030(0.120)	0.030(0.120)	0.120(0.160)	0.120(0.130)	0.040(0.130)	0.050(0.100)	0.060(0.140)
	-0.000(0.060)	-0.000(0.060)	-0.000(0.060)	-0.000(0.110)	0.010(0.120)	-0.000(0.060)	0.000(0.050)	-0.010(0.070)
	-0.050(0.110)	-0.030(0.100)	-0.030(0.100)	-0.120(0.150)	-0.140(0.130)	-0.030(0.120)	-0.050(0.100)	-0.060(0.120)
				Scenario 2: $\epsilon \sim t_1$				
2	-0.160(1.280)	8.210(39.220)	8.210(39.220)	13.380(89.170)	0.050(4.260)	-0.550(0.760)	34.580(192.760)	0.630(7.860)
	-0.070(1.190)	48.250(303.090)	48.250(303.090)	-3.880(238.410)	-0.630(2.030)	-0.310(0.620)	33.300(252.260)	0.720(4.580)
	0.170(0.840)	-34.590(162.640)	-34.590(162.640)	-18.070(47.120)	-0.100(2.170)	0.090(0.500)	-21.280(82.890)	-4.690(13.370)
	0.660(1.840)	-45.270(336.280)	-45.270(336.280)	-36.040(141.380)	-0.310(8.410)	0.820(0.990)	-55.330(362.950)	-3.550(13.840)
	0.330(1.480)	73.070(389.780)	73.070(389.780)	70.780(326.440)	-0.500(1.670)	-0.450(0.870)	34.570(179.730)	2.160(5.750)
	0.330(1.020)	-20.300(122.470)	-20.300(122.470)	-4.680(35.450)	0.250(1.880)	0.190(0.580)	-8.940(48.910)	-4.050(15.890)
	0.160(0.160)	0.160(0.250)	0.160(0.250)	0.140(0.280)	0.110(0.160)	0.200(0.210)	0.140(0.300)	0.170(0.230)
	-0.070(0.080)	-0.050(0.210)	-0.050(0.210)	0.000(0.270)	0.020(0.160)	0.000(0.210)	-0.050(0.220)	-0.020(0.200)
	-0.090(0.150)	-0.110(0.270)	-0.110(0.270)	-0.140(0.260)	-0.130(0.160)	-0.210(0.140)	-0.090(0.310)	-0.150(0.220)
				Scenario 3: $\epsilon \sim t_3$				
3	-0.060(0.850)	-0.630(1.160)	-0.630(1.160)	-0.770(0.760)	-0.680(0.700)	-0.680(0.630)	-0.520(0.580)	-0.660(0.620)
	0.010(0.300)	0.100(2.600)	0.100(2.600)	-0.120(0.570)	-0.150(0.520)	-0.030(0.290)	0.010(0.430)	-0.040(0.450)
	-0.090(0.370)	-0.340(2.080)	-0.340(2.080)	0.190(0.700)	0.090(0.400)	-0.030(0.330)	0.000(0.690)	0.150(0.530)
	0.290(0.670)	0.090(1.760)	0.090(1.760)	0.610(1.170)	0.880(0.980)	0.490(0.750)	0.060(0.740)	0.510(0.820)
	-0.040(0.290)	0.080(1.190)	0.080(1.190)	-0.370(0.770)	-0.560(0.840)	0.050(0.300)	-0.030(0.280)	-0.050(0.520)
	0.080(0.300)	0.300(1.580)	0.300(1.580)	0.280(0.960)	0.250(0.700)	0.220(0.470)	0.130(0.800)	0.400(0.850)
	0.060(0.100)	0.130(0.170)	0.130(0.170)	0.130(0.150)	0.140(0.130)	0.210(0.170)	0.070(0.130)	0.150(0.150)
	-0.010(0.070)	-0.010(0.090)	-0.010(0.090)	0.000(0.120)	0.020(0.140)	-0.030(0.090)	-0.000(0.070)	-0.020(0.100)
	-0.050(0.110)	-0.120(0.160)	-0.120(0.160)	-0.130(0.150)	-0.150(0.140)	-0.170(0.140)	-0.070(0.120)	-0.140(0.130)
				Scenario 4: 5% added outliers				
4	-0.470(0.770)	-1.510(1.370)	-0.870(1.240)	0.560(9.140)	-0.620(0.760)	-1.580(0.990)	-2.280(1.110)	-0.670(1.090)
	0.140(0.410)	0.560(0.580)	-0.520(1.360)	-0.300(1.120)	-0.150(0.510)	0.300(0.500)	0.320(0.350)	0.200(0.330)
	0.110(0.300)	-0.000(1.350)	0.880(1.200)	0.470(0.620)	0.070(1.490)	0.160(0.580)	0.090(0.940)	0.400(0.580)
	-0.290(1.160)	-0.750(2.360)	1.170(2.520)	-0.420(5.370)	0.850(1.160)	-0.770(1.760)	-2.080(2.090)	0.620(1.530)
	0.220(0.470)	0.750(0.780)	-0.960(2.240)	-0.050(1.270)	-0.330(0.960)	0.510(0.550)	0.520(0.490)	0.260(0.580)
	0.120(0.420)	-0.640(1.650)	0.810(2.060)	0.420(1.060)	0.280(1.330)	0.130(0.950)	-0.300(1.300)	-0.300(0.790)
	0.020(0.090)	0.020(0.220)	0.080(0.140)	0.110(0.150)	0.090(0.150)	0.140(0.170)	-0.020(0.160)	-0.030(0.130)
	-0.010(0.070)	-0.040(0.080)	0.010(0.150)	-0.010(0.140)	0.010(0.140)	0.010(0.080)	0.020(0.080)	-0.030(0.060)
	-0.010(0.100)	0.010(0.200)	-0.090(0.160)	-0.090(0.140)	-0.100(0.170)	-0.140(0.150)	-0.000(0.160)	0.060(0.140)
				Scenario 5: 10% added outliers				
5	-0.290(0.610)	-0.930(1.530)	-0.790(1.410)	-0.490(1.400)	-0.720(0.710)	-0.930(0.720)	-1.440(1.130)	-0.650(1.310)
	0.050(0.200)	0.350(0.490)	-0.870(1.980)	-0.210(0.730)	-0.060(0.490)	0.070(0.220)	0.340(0.750)	0.160(0.410)
	-0.040(0.280)	0.080(1.400)	1.080(1.380)	0.300(0.640)	0.230(0.570)	0.060(0.260)	0.210(0.880)	0.500(0.620)
	0.120(0.590)	0.050(2.280)	1.270(1.920)	0.630(1.520)	0.740(1.050)	-0.100(1.250)	-0.770(2.080)	0.500(1.790)
	0.030(0.210)	0.340(0.470)	-1.390(2.630)	-0.180(1.050)	-0.430(0.610)	0.120(0.250)	0.330(0.590)	0.150(0.430)
	0.050(0.170)	-0.520(1.700)	0.520(1.910)	0.200(0.950)	0.300(0.730)	0.150(0.350)	-0.020(1.400)	-0.090(1.050)
	0.040(0.100)	0.020(0.250)	0.110(0.190)	0.120(0.150)	0.130(0.150)	0.130(0.140)	0.030(0.190)	-0.050(0.180)
	0.010(0.050)	-0.030(0.100)	0.030(0.140)	-0.010(0.110)	0.000(0.130)	0.020(0.060)	-0.010(0.110)	-0.020(0.080)
	-0.050(0.110)	0.010(0.240)	-0.140(0.200)	-0.110(0.140)	-0.140(0.150)	-0.150(0.130)	-0.030(0.160)	0.070(0.160)

Table 3.8. Experiment Setting: K=3, P=1, N=400, Unbalanced

Scenario	CAT	MLE	TLE	CWM1	CWM2	MIXBI	MIXL	MIXT
				Scenario 1: $\epsilon \sim N(0,1)$				
1	-0.030(0.420)	-0.000(0.360)	-0.000(0.360)	-0.650(0.590)	-0.710(0.690)	0.030(0.280)	-0.220(0.390)	0.010(0.290)
	0.030(0.130)	0.020(0.160)	0.020(0.160)	0.010(0.260)	-0.050(0.410)	0.010(0.110)	0.030(0.150)	0.020(0.120)
	-0.050(0.210)	0.040(0.330)	0.040(0.330)	0.310(0.540)	0.110(0.410)	-0.010(0.140)	0.020(0.190)	0.010(0.150)
	0.220(0.350)	0.000(0.220)	0.000(0.220)	0.610(0.740)	1.050(0.910)	-0.000(0.200)	-0.000(0.290)	-0.080(0.220)
	0.010(0.130)	0.020(0.130)	0.020(0.130)	-0.200(0.430)	-0.360(0.560)	0.020(0.100)	-0.010(0.140)	0.010(0.120)
	-0.010(0.130)	0.080(0.480)	0.080(0.480)	0.350(0.750)	0.170(0.450)	-0.010(0.110)	0.030(0.130)	0.020(0.120)
	0.010(0.080)	0.010(0.090)	0.010(0.090)	0.130(0.150)	0.120(0.150)	0.000(0.050)	0.020(0.070)	-0.010(0.050)
	-0.000(0.040)	-0.010(0.050)	-0.010(0.050)	-0.010(0.090)	-0.000(0.110)	-0.000(0.040)	-0.000(0.040)	-0.000(0.030)
	-0.000(0.090)	-0.000(0.070)	-0.000(0.070)	-0.130(0.130)	-0.120(0.130)	0.000(0.060)	-0.020(0.070)	0.010(0.060)
				Scenario 2: $\epsilon \sim t_1$				
2	-0.530(1.130)	-4.720(240.250)	-4.720(240.250)	12.570(59.800)	-0.420(2.120)	-0.670(0.510)	29.950(107.900)	1.520(7.220)
	0.030(0.280)	-3.670(124.420)	-3.670(124.420)	123.370(1149.950)	-0.450(2.220)	-0.260(0.400)	16.420(219.360)	0.320(3.850)
	0.120(0.800)	-52.110(299.120)	-52.110(299.120)	-55.660(461.180)	-0.010(1.560)	0.080(0.230)	-68.210(227.540)	-4.220(12.490)
	0.640(1.290)	-169.330(1179.540)	-169.330(1179.540)	-21.540(103.290)	0.680(2.290)	0.820(0.830)	-47.840(216.920)	-1.350(7.120)
	0.000(0.290)	28.510(104.000)	28.510(104.000)	81.240(445.560)	-0.590(1.820)	-0.560(0.800)	77.830(353.090)	1.240(4.470)
	0.280(0.800)	-23.250(235.490)	-23.250(235.490)	-21.000(214.110)	0.180(1.120)	0.060(0.220)	-31.740(134.970)	-2.310(9.870)
	0.170(0.170)	0.130(0.290)	0.130(0.290)	0.130(0.270)	0.120(0.170)	0.190(0.230)	0.150(0.360)	0.180(0.230)
	-0.060(0.090)	-0.000(0.270)	-0.000(0.270)	-0.060(0.250)	0.010(0.160)	0.040(0.240)	-0.070(0.290)	-0.030(0.190)
	-0.110(0.160)	-0.120(0.310)	-0.120(0.310)	-0.070(0.280)	-0.130(0.170)	-0.230(0.110)	-0.090(0.360)	-0.150(0.220)
				Scenario 3: $\epsilon \sim t_3$				
3	0.020(0.470)	-0.340(2.120)	-0.340(2.120)	-0.670(0.780)	-0.800(0.580)	-0.740(0.540)	-0.330(0.410)	-0.620(0.610)
	0.020(0.250)	-0.050(1.360)	-0.050(1.360)	-0.250(0.600)	-0.100(0.440)	-0.000(0.200)	0.010(0.190)	-0.040(0.290)
	-0.060(0.240)	0.260(0.580)	0.260(0.580)	0.360(0.560)	0.250(0.500)	-0.010(0.310)	-0.010(0.320)	0.280(0.450)
	0.290(0.360)	0.490(0.760)	0.490(0.760)	1.030(0.940)	0.790(0.830)	0.440(0.490)	-0.000(0.360)	0.750(0.780)
	-0.020(0.190)	0.060(1.400)	0.060(1.400)	-0.540(0.680)	-0.460(0.620)	0.030(0.150)	-0.040(0.170)	-0.070(0.300)
	0.040(0.370)	0.230(0.820)	0.230(0.820)	0.270(0.840)	0.250(0.610)	0.150(0.480)	0.100(0.380)	0.210(0.660)
	0.050(0.090)	0.100(0.170)	0.100(0.170)	0.140(0.150)	0.120(0.140)	0.230(0.170)	0.040(0.080)	0.120(0.150)
	-0.010(0.050)	-0.010(0.090)	-0.010(0.090)	-0.010(0.110)	0.010(0.120)	-0.030(0.060)	0.010(0.050)	-0.030(0.070)
	-0.040(0.090)	-0.090(0.140)	-0.090(0.140)	-0.130(0.140)	-0.140(0.130)	-0.200(0.130)	-0.040(0.090)	-0.090(0.150)
				Scenario 4: 5% added outliers				
4	-0.720(0.740)	-1.700(1.340)	-1.060(0.960)	0.340(6.670)	-0.550(0.930)	-1.420(0.900)	-2.520(1.150)	-0.630(0.770)
	0.090(0.200)	0.530(0.390)	-0.760(1.100)	-0.470(1.060)	-0.120(0.430)	0.250(0.200)	0.380(0.340)	0.170(0.260)
	0.090(0.210)	0.080(1.080)	0.600(1.040)	0.510(0.640)	0.210(0.780)	0.090(0.320)	0.030(1.040)	0.520(0.210)
	-0.500(0.750)	-0.810(2.120)	1.600(1.830)	-0.150(4.160)	0.980(1.160)	-0.600(1.260)	-2.210(2.100)	0.650(1.190)
	0.110(0.240)	0.690(0.590)	-1.340(2.040)	0.210(1.300)	-0.360(0.680)	0.410(0.240)	0.450(0.370)	0.240(0.450)
	0.070(0.150)	-0.830(1.600)	0.720(1.820)	0.380(0.980)	0.020(0.770)	0.070(0.540)	-0.470(1.680)	-0.330(0.250)
	0.040(0.070)	-0.010(0.190)	0.100(0.170)	0.140(0.170)	0.080(0.150)	0.130(0.120)	0.000(0.190)	-0.050(0.120)
	0.010(0.040)	-0.030(0.050)	0.010(0.180)	-0.040(0.120)	0.010(0.130)	0.030(0.040)	0.020(0.070)	-0.030(0.040)
	-0.050(0.080)	0.040(0.200)	-0.110(0.160)	-0.100(0.140)	-0.090(0.150)	-0.160(0.110)	-0.020(0.170)	0.080(0.120)
				Scenario 5: 10% added outliers				
5	-0.200(0.500)	-1.260(1.680)	-1.040(1.320)	-0.280(2.780)	-0.720(0.580)	-0.870(0.290)	-1.420(1.050)	-0.390(1.200)
	0.010(0.130)	0.250(0.310)	-0.700(1.860)	-0.200(0.570)	-0.130(0.530)	0.040(0.120)	0.260(0.440)	0.120(0.280)
	0.020(0.250)	0.360(0.830)	0.320(1.360)	0.300(1.400)	0.290(0.540)	0.030(0.150)	0.330(0.730)	0.530(0.380)
	0.040(0.440)	-0.480(2.180)	0.900(2.490)	0.380(2.170)	1.010(1.070)	0.100(0.390)	-0.700(1.880)	0.730(1.520)
	0.030(0.160)	0.290(0.350)	-1.420(2.550)	-0.070(0.860)	-0.310(0.630)	0.130(0.150)	0.270(0.400)	0.110(0.240)
	0.040(0.350)	-0.380(0.990)	0.300(2.260)	0.430(1.410)	0.290(0.720)	0.110(0.130)	0.130(1.210)	-0.240(0.710)
	0.010(0.080)	-0.090(0.150)	0.050(0.150)	0.110(0.160)	0.120(0.140)	0.150(0.100)	0.010(0.180)	-0.090(0.140)
	0.010(0.040)	-0.010(0.050)	0.010(0.160)	-0.010(0.120)	-0.040(0.120)	0.020(0.040)	-0.010(0.110)	-0.010(0.050)
	-0.030(0.070)	0.100(0.150)	-0.070(0.180)	-0.100(0.130)	-0.090(0.150)	-0.170(0.090)	-0.000(0.140)	0.100(0.130)

for CAT and MIXT were affected by the outliers as the regression lines after contamination tend to go through the outliers and were quite different from those before contamination. For contaminated data at $x = 0.7$ (bottom panel), all other methods except for CAT, TLE and MIXBI were severely affected by the outliers, seen from a dramatic deviation of the regression lines before and after contamination. This shows that CAT achieves better performances over others in robustly revealing the latent relationship among genomic markers.

3.5 Conclusions

We proposed a novel robust algorithm for solving FMGR using the Classification Expectation Maximization (CEM) algorithm, based on which, the outliers are more naturally defined, and robustness issue better and more conveniently handled. Our method, CAT, enables the automatic detection of outliers and robust estimation of parameters simultaneously, which is not capable of by existing methods. The removal of outliers in final parameter estimation significantly increased the estimation efficiency than other algorithms; and the adoption of the highly robust least trimmed likelihood estimator within each component makes it possible to avoid the pre-specification of trimming parameter. Under the CEM framework, the adaptive trimming of a mixture model boils down into that of simple linear regression models corresponding to each component. Owing to its high breakdown point property, we assumed a high portion of outlying samples and the trimmed likelihood approach is optimized on only half of the samples within each component with the highest valued likelihood. This is the reason why CAT could be robust to outliers and heavy tailed noise. In summary, CAT is an robust mixture regression algorithm of high potential and practical utility in robustly mining the heterogeneous relations among noisy variables in biomedical data.

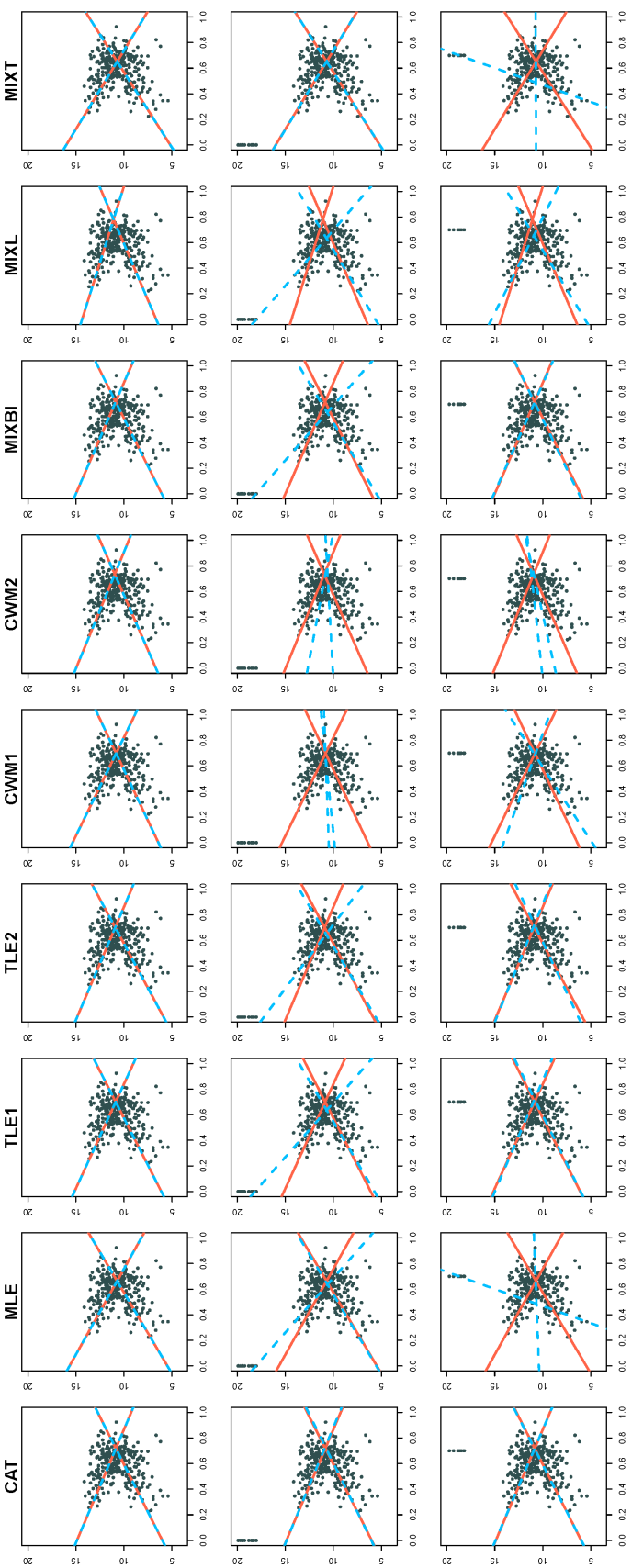


Figure 3.1. Mixture regression of CREB3L1 expression (y-axis) on cg16012690 methylation (x-axis) using different methods

4. SUPERVISED CLUSTERING OF HIGH DIMENSIONAL DATA USING REGULARIZED MIXTURE MODEL

This chapter focuses on the challenges in studying the heterogeneous relationship between high-dimensional genetic features and a phenotype. The goal of the proposed method in this chapter consists of two parts. The first is that identifying the sample level heterogeneity by using an external supervised variable. The second is that exploring the linear dependency between identified features within high dimensional features and the external supervised variable. This linear dependency owns clinical or biological interpretability of clustering.

In this chapter, we have two contributions on supervised clustering of high dimensional data by the regularized mixture model.

1. We proposed a novel supervised clustering algorithm using penalized mixture regression model, called Component-wise Sparse Mixture Regression (CSMR) which substantial improvement on both the computational efficiency and biological interpretability.

2. Our method could perform clustering and regression of the response on the features at the same time. In addition, we use a penalized term to handle high dimensional features.

In robust mixture regression problem, there are usually enough observation in the collected data. However, it is challenging to analysis biomedical data because there are more than 20,000 genetic features (e.g. genes) with limited observations. As a result, it is hard to deduce the disease subtype (e.g. clusters of sample) related genetic features and make good interpretations. Firstly, we proposed a penalized mixture regression model by adding a constraint on the coefficient matrix. This constraint forces the sparsity of the coefficient matrix so that selecting the useful features and removing redundant features. In other words, this regularized term enables a feature selection step while estimating the parameter of the mixture model. However, there is a hyper-parameter to balance the two parts in the log likelihood during the optimization. The choice of component specific penalty parameter is difficult. Secondly, to handle this issue, our method conducted the tuning process of hyper-parameter inside each CEM iteration. This makes overall grid-search not that hard anymore. Then, we also added a model refit step following each CEM step which significantly increase the numerical stability and the rate of convergence.

4.1 Introduction

Detection and estimation of the genetic markers associated with phenotypic features is one of the most important problems in biomedical research. Predictive models have been extensively used to link genetic markers to a phenotypic trait, however, the unobserved patient heterogeneity obfuscates the effort to build a unified model that works for all hidden disease subtypes. It has been well understood that various subtypes exist for many common diseases, which vary in etiology, pathogenesis, and prognosis [106], [124], [125]. For example, the cancer cells are constantly evolving in the tumor microenvironment, and they may acquire variations on alternative pathways in response to treatment, which explains why certain patients have better prognoses than others in response to the same treatment [82], [126]. This implies that the same predictive model that links genetic markers to a phenotypic trait may not be valid for every patient, and further it is unclear to what extent the patients should be considered together [127]. Therefore, it is judicious to construct a set of heterogeneous models, each of which corresponds to one subtype.

In order to find sample subgroups guided by an external response variable, which carries important biological/clinical information, we need to perform clustering of the samples and regression of the response on the features at the same time. Clearly, our challenges are distinct in two ways: the variables of interest to each subgroup may be a distinct and sparse set of the high dimensional genetic features, and the set of patients in each subgroup is not known *a priori*. Essentially, we assume that observations belong to unlabeled classes with class-specific regression models relating their unique and selective genetic markers to the phenotypic outcome. The ultimate goal is to group the subjects into clusters such that the observed response variable conditional on the feature variables in the same cluster are more similar to each other than those from different clusters. In other words, we are detecting sample clusters such that in the same cluster, the relationship between the features and the response could be described by one unified model, which differs from another cluster.

The rest of this chapter is organized as follows: in Section 4.2, we introduce our algorithm, Component-wise Sparse Mixture Regression (CSMR); in Section 4.3, we compare CSMR with five state-of-the-art algorithms on simulation datasets, namely, LASSO, Ridge regression,

random forest (RF), ICC [87], and FMRS [83]; in Section 4.4, we applied all the algorithms on 24 drug sensitivity data in CCLE, to screen for genes that underlie the heterogeneous drug resistance mechanisms.

4.2 Methods

4.2.1 Motivation

We assume that the samples belong to different sub-populations, each of which is defined by a distinct relationship between the genetic biomarkers and the phenotype of interest, and the genetic markers are sparse subsets of the high dimensional genetic profiles specific to each sub-population. Figure 4.2 illustrated an example where the patients fall under two distinct subgroups: blue for patients acquiring one mechanism to the treatment that resulted in responsiveness, while pink for patients acquiring another mechanism to the same drug that resulted in non-responsiveness. The goal of our method is to cluster the samples (blue and pink) supervised by the patients drug sensitivity measure, and find the defining genetic features (yellow) associated with each cluster. The identified genetic features could be further studied to guide targeted therapeutic designs.

4.2.2 The Penalized Likelihood of Mixture Regression

Knowing that $\boldsymbol{\beta}_k$ is sparse means many elements in $\boldsymbol{\beta}_k$ will tend to be close to zero, but not exactly zero without proper regularization in the model. To simultaneously shrink the insignificant regression coefficients in $\boldsymbol{\beta}_k$ and estimate $\boldsymbol{\theta}$, we could introduce a penalty term to equation 2.7 and optimize the following penalized log likelihood function:

$$\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) - P_{\boldsymbol{\lambda}}(\boldsymbol{\theta}) \tag{4.1}$$

where $\mathcal{L}(\boldsymbol{\theta})$ denotes the observed log likelihood, and $P_{\boldsymbol{\lambda}}(\boldsymbol{\theta}) : \mathcal{R}^P \to \mathcal{R}$ is a regularizer of the regression coefficients, the penalty for each component is dependent on a component specific hyperparameter $\lambda_k > 0, k = 1, ..., K$. Various types of penalty forms were used in

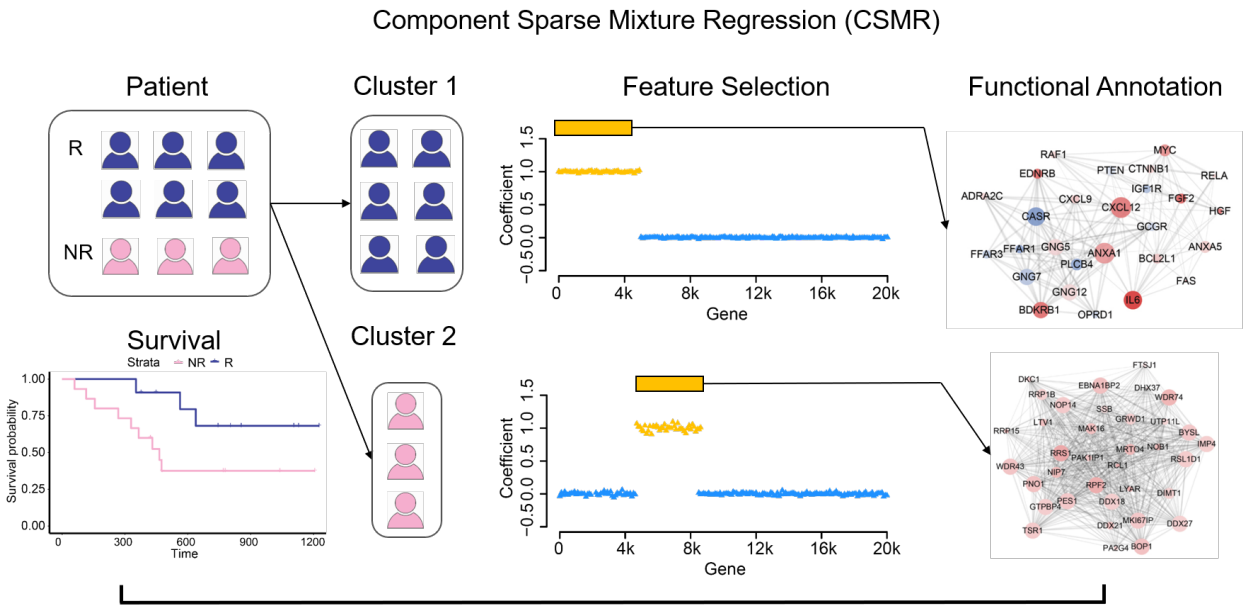


Figure 4.1. The motivation of CSMR. Under the same treatment, some patients acquired one mechanism to deal with the drug, (blue), while others picked up another (pink), resulting in different prognoses for the same treatment. The motivation of CSMR is to cluster the patients in a supervised fashion and examine what are the genes (yellow) that are selected in tumor progression that led to the different drug resistance subtypes of patients, and their functions (network).

the mixture regression model, and we could consider the LASSO penalty form as it is convex and thus advantageous for numerical computation [83], [84], [87], i.e.,

$$P_{\boldsymbol{\lambda}}(\boldsymbol{\theta}) = \Sigma_{k=1}^{K} \lambda_k \boldsymbol{\pi}_k^{\gamma} \Sigma_{\mathrm{j}=1}^{P} \mid \beta_{\mathrm{j}k} \mid \tag{4.2}$$

where γ is usually chosen among 0, 0.5, 1 as in [84]. A non-zero γ would involve $\boldsymbol{\pi}_k, k = 1, ..., K$ in the penalty term $P_{\boldsymbol{\lambda}}(\boldsymbol{\theta})$, that could largely increase the computational complexity in the maximization step in using EM algorithm [83], [84].

Similar to the case of low dimensional mixture regression, the EM algorithm could be adopted by maximizing the penalized complete log likelihood function defined as $\mathcal{L}^{pc}(\boldsymbol{\theta}) = \mathcal{L}^{c}(\boldsymbol{\theta}) - P_{\boldsymbol{\lambda}}(\boldsymbol{\theta})$, as in existing methods. The conditional expectation corresponding to the penalized likelihood function is given by

$$Q^{p}(\boldsymbol{\theta}; \boldsymbol{\theta}^{(m)}) = \Sigma_{\mathrm{i}=1}^{N} \Sigma_{k=1}^{K} p_{\mathrm{i}k}^{(m)} [\log \boldsymbol{\pi}_k + \log \mathcal{N}(y_{\mathrm{i}}; \boldsymbol{x}_{\mathrm{i}}^{T} \boldsymbol{\beta}_k, \sigma_k^2)] - P_{\boldsymbol{\lambda}}(\boldsymbol{\theta}) \tag{4.3}$$

In the E step, the conditional expectation of $z_{\mathrm{i}k}$ is similar to the low-dimensional case. However, in the M step, maximizing $Q^{p}(\boldsymbol{\theta}; \boldsymbol{\theta}^{(m)})$ with respect to $\boldsymbol{\theta}$, is more complicated than the low-dimensional case. The involvement of $\boldsymbol{\pi}_k, \boldsymbol{\beta}_k$ in $P_{\boldsymbol{\lambda}}(\boldsymbol{\theta})$ makes it impossible to obtain nice closed form solutions for neither of the two. We next review the use of the CEM algorithm, as we introduced in Section 3.2.2 and described as algorithm 1, that could largely increase the computational efficiency.

The biggest advantage of the CEM algorithm is that it disentangles the mixture into individual non-overlapping components, such that flexible sparsity control could be easily achievable within each component. Hence for the high dimensional mixture regression problem, we could simply replace the OLS estimator in the M step of the CEM algorithm by a sparse estimator, i.e.,

$$\underset{\boldsymbol{\beta}_k, \sigma_k^2}{\operatorname{argmax}} \sum_{\mathrm{i} \in \mathcal{C}_k^{(m+1)}} \log \mathcal{N}(y_{\mathrm{i}}; \boldsymbol{x}_{\mathrm{i}}^{T} \boldsymbol{\beta}_k, \sigma_k^2) - \lambda_k \boldsymbol{\pi}_k^{\gamma} \Sigma_{\mathrm{j}=1}^{P} \mid \beta_{\mathrm{j}k} \mid \tag{4.4}$$

This is simply L_1 regularized linear regression, for which many efficient algorithms exist [128]. Note that when $\gamma \neq 0$, the involvement of $\boldsymbol{\pi}_k^{\gamma}$ in the penalty term makes it challenging for the maximization with regards to both $\boldsymbol{\pi}_k$ and $\boldsymbol{\beta}_k$. Maximizing the function $Q^p(\boldsymbol{\theta}; \boldsymbol{\theta}^{(m)})$ with regards to the mixing proportions is much more complex than maximizing its leading term, i.e., $Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(m)})$, and in our CSMR algorithm, for simplicity, we ignored the penalty term involving $\boldsymbol{\pi}_k$'s, i.e., $P_{\boldsymbol{\lambda}}(\boldsymbol{\theta})$, when solving for $\boldsymbol{\pi}_k$, such that it will have a nice closed form solution. And it has been shown to work well in [83] and our own simulation data. As for the solution of $\boldsymbol{\beta}_k$, we show in the next section that under the proposed CEM updates, the involvement of $\boldsymbol{\pi}_k$'s only affects the scale of λ_k's, which are selected with cross validation within each iteration, and hence does not impact the estimation of $\boldsymbol{\beta}_k$.

4.2.3 The CSMR algorithm

Here we proposed the CSMR algorithm to solve the high dimensional mixture regression problem based on the CEM algorithm. In CSMR, the mixture regression setting could handle the hidden cluster problem, and the disentangled clusters under CEM could efficiently solve the feature selection problem in a high dimensional setting. At the E-step, we calculate the posterior probability p_{ik} similar to traditional EM algorithm; at the C-step, we assign each observation to a cluster that it most likely belongs to, similar to traditional CEM; at the M-step, for each component, we perform regularized linear regression to obtain a sparse set of non-zero coefficients.

A big challenge with the penalized mixture regression problem is the choice of component specific penalty parameters λ_k. The λ_k's are related to the amount of regularization, and their selection is a critical issue in a penalized likelihood approach. It is usually based on a trade-off between bias and variance: large values of tuning parameters tend to select a simple model whose parameter estimates have smaller variance, whereas small values of the tuning parameters lead to complex models, with smaller bias. Cross-validation over a grid search is the commonly adopted method to select the optimal combination of λ_k, but this becomes increasingly prohibitive with the increase of K, especially when we don't have a good knowledge of the theoretical range of the λ_k.

Hence, instead of first performing penalized linear regression for given λ_k and then searching for the optimal combination of λ_k [83], we propose to conduct the tuning of λ_k with cross validation inside the CEM iterations. Specifically, under the CEM algorithm, all the components are disentangled, we could hence perform hyperparameter tuning inside each iteration within each component. This is to say, at the M-step, we not only estimate the regression coefficients, but also find the best tuning parameter λ_k for the component. Hence, at the end of the algorithm, we avoid the hyperparameter tuning, as they have already been selected within the iteration. We adopted the efficient cross validation algorithm for selecting the optimal regularization parameter under L_1 regularized linear regression [128]. Since we no longer need to run the algorithm multiple times on a K-dimensional grid space of the penalty parameters, we could hence largely reduce the computational cost. We have shown in simulation studies that penalty parameters selected this way empirically worked very well.

Another adaptation on the traditional CEM algorithm of CSMR is a model refit step following the CEM steps. To increase the numerical stability and achieve faster convergence, at the end of each iteration, we refit the mixture regression model using flexible EM algorithm with only the selected variables of each component. Basically, for each component, the coefficients of the variables not selected at the M-step will be forced to be zero. This could be easily achievable by allowing only the selected variables of component k to enter into the model fitting of the k-th regression parameters.

The CSMR algorithm requires the initialized values $\boldsymbol{\theta}$. Here, we order the features based on its individual Pearson correlation with the response variable, and then fit a low-dimensional mixture regression model solved by the traditional EM algorithm using the top correlated genes. CSMR is implemented in R, and was made available in Github[1].

[1]↑https://github.com/zcslab/CSMR

Input: $X_{N\times P}, Y_{N\times 1}, K$
Output: $\boldsymbol{\theta}, \mathcal{C} = \bigcup_{k=1}^{K} \mathcal{C}_k, \{\beta_{0k}, \boldsymbol{\beta}_k\}_{k=1}^{K}$
Initialization: $\boldsymbol{\theta}^{(0)} = \{\boldsymbol{\pi}_k^0, \boldsymbol{\beta}_k^{(0)}, \sigma_k^{2^{(0)}}\}_{k=1}^{K}$
for *m=0,...,Max Iteration* **do**
E-step: compute the conditional expectation of z_{ik} similar to traditional EM algorithm.
C-step: for $k = 1, ..., K$, assign $C_k^{(m+1)}$ as the set of observations that are mostly likely in component k.
M-step: for $k = 1, ..., K$, the relative cluster size is updated by $\hat{\boldsymbol{\pi}}_k^{(m+1)} = \frac{n_k^{(m+1)}}{N}$, and the tuning parameter $\lambda_k^{(m+1)}$, and regression parameters $(\hat{\boldsymbol{\beta}}_k^{(m+1)}, \hat{\sigma}_k^{(m+1)})$ are selected and estimated using cross validation, such as the cv.glmnet function in glmnet package.
Model refit: refit the FMGR model by allowing only the selected variables in each component and to obtain $\{\boldsymbol{\pi}_k^{(m+1)}, \boldsymbol{\beta}_k^{(m+1)}, \sigma_k^{(m+1)}\}_{k=1}^{K}$ given by this flexible modeling
Stop if converged.
end

Algorithm 4: CSMR

4.2.4 Selection of Component Number K

The number of clusters K is a sensible parameter because it describes the heterogeneity of the population. For selection of K, we could use a modified BIC criterior that minimizes

$$BIC(K) = -2\mathcal{L}^{pc}(\boldsymbol{\theta}_K^*) + log(N)d_K \tag{4.5}$$

where $\boldsymbol{\theta}_K^*$ represents the parameter estimates for K, and $d_K = K+(K-1)+\Sigma_{k=1}^{K}\Sigma_{\mathrm{j}=1}^{P}1_{\{\beta_{\mathrm{j}k}\neq 0\}}$ is the effective number of parameters to be estimated, similar to [129]. Specifically, there are K standard deviations, σ_k, associated with the K regression lines; $K-1$ component proportions, $\boldsymbol{\pi}_k$, since $\Sigma_k\boldsymbol{\pi}_k = 1$; and all the non-zero linear regression coefficients for all the K components.

In addition to the BIC criteria, we also offer a cross validation algorithm for the selection of K. Take a 5-fold cross validation as an example. For given K, at each repetition, 80% samples are used for training to obtain the regularized parameter $\boldsymbol{\theta}_K^*$. Then, for a sample $(\boldsymbol{x}_{\mathrm{i}}, y_{\mathrm{i}})$ drawn from the 20% testing samples, its cluster membership, k_0, is first predicted as

$$k_0 = \max_k \boldsymbol{\pi}_{k,K}^* \mathcal{N}(y_{\mathrm{i}}; \boldsymbol{x}_{\mathrm{i}}^T \boldsymbol{\beta}_{k,K}^*, \sigma_{k,K}^{2*}) \tag{4.6}$$

Here, $\{\boldsymbol{\pi}_{k,K}^*, \boldsymbol{\beta}_{k,K}^*, \sigma_{k,K}^{2*}\}_{k=1}^{K}$ denotes the CSMR estimated parameters when the number of components is K. After assigning the observation to component k_0, we could make prediction of the response based on linear regression, i.e. $\hat{y}_{\mathrm{i}} = \boldsymbol{x}_{\mathrm{i}}^T \boldsymbol{\beta}_{k_0,K}^*$, as well as the associated residual, $y_{\mathrm{i}} - \hat{y}_{\mathrm{i}}$. Notably, such a prediction of the response is different from simple linear regression, as the prediction process requires knowing the value of the response, in order to assign it to the right cluster. After knowing its cluster membership, a prediction of the response could be made.

A large K will tend to overfit the data with a more complex model of higher variance, while smaller K might select a simpler model with larger bias. Using the independent testing data, we could decide how to balance the trade-off between bias and variance. To evaluate how the estimated model under K explains the testing data, we could calculate the root-mean-square-error between y_{i} and $\hat{y}_{\mathrm{i}}$, or Pearson correlation between the two. By repeating

this procedure multiple times, a more robust and stable evaluation of the choice of K should be derived based on the summarized RMSE or Pearson correlations.

4.3 Experiments by using Simulated Data

4.3.1 Simulation Settings

To simulate N observations with P independent variables, we first simulated the design matrix X, such that the first column of X is all 1, corresponding to the intercept, i.e., $x_{\mathrm{i}1} = 1, \mathrm{i} = 1, ..., N$; and for the rest of the P columns, all the elements follow i.i.d standard normal distribution, i.e., $x_{\mathrm{ij}} \sim N(0,1), \mathrm{i} = 1, ..., N, \mathrm{j} = 2, ..., P+1$. The component proportions were made to be equal, i.e., $\boldsymbol{\pi}_k = \frac{1}{K}$. For component k, a random sample of size M_0 were taken from the set $\{2, ..., P+1\}$, denoted as J_k, to mimic the sparse component specific variables predictive of the response. And we simulated the component specific coefficients, $\beta_{\mathrm{j}k}$, to be a random draw from $Unif((-b,-a)\bigcup(a,b))$, if $\beta_{\mathrm{j}k} \in J_k$; and let $\beta_{\mathrm{j}k} = 0$, if $\beta_{\mathrm{j}k} \notin J_k$.

The response variable Y was generated by the following two-step process:

1. Draw the component indicator variable $z_{\mathrm{i}k}$ following the multinomial distribution $Multi(1; \boldsymbol{\pi}_1, ..., \boldsymbol{\pi}_K)$, in other words, $p(z_{\mathrm{i}k} = 1) = \boldsymbol{\pi}_k$.
2. Draw an observation y_{i} according to a normal distribution $N(\beta_{0k} + \boldsymbol{x}_{\mathrm{i}}^{\boldsymbol{T}}\boldsymbol{\beta_k}, \sigma_k^2)$, if $z_{\mathrm{i}k} = 1$.

Here, we fix $a = 2,\ b = 5$. We explored the performances of existing methods under 15 different simulation scenarios, for each of which, 100 repetitions were conducted:

Cases 1-3. $N = 200, 300, 400, P = 100, K = 2, \sigma = 1, M_0 = 5$

Cases 4-6. $N = 400, P = 100, K = 2, 3, 4, \sigma = 1, M_0 = 5$

Cases 7-9. $N = 400, P = 100, K = 2, \sigma = 0.5, 1, 2, M_0 = 5$

Cases 10-12. $N = 400, P = 100, K = 2, \sigma = 1, M_0 = 5, 8, 20$

Cases 13-15. $N = 400, K = 2, \sigma = 1, M_0 = 5, P = 100, 500, 2000$

Table 4.1. Baseline methods

	Prediction	Clustering	Variable selection
CSMR	×	×	×
LASSO	×		×
RIDGE	×		
RF	×		×
FMRS	×	×	×
ICC	×	×	×

4.3.2 Selection of Baseline Methods

We compared CSMR with five different methods, including $\mathcal{L}_1$ penalized regression, or LASSO; $\mathcal{L}_2$ penalized regression, or Ridge regression (RIDGE); random forest based regression (RF), sparse mixture regression, FMRS [130], and the imputation-conditional consistency algorithm, ICC [87]. They differ in their ability to perform prediction, clustering and variable selection, as shown in Table 4.1.

Among them, CSMR, ICC and FMRS are all capable of doing variable selection at the same time as sample clustering. However, FMRS can only deal with relatively lower dimensional features, while ICC represents the method that is powerful in dealing with much higher dimensional features.

4.3.3 Methods Comparisons and Performance Evaluations

We focused on four metrics for method comparisons: 1) the average correlation between predicted and observed response; 2) the true positive rate (TPR) and 3) true negative rate (TNR) of variable selection; 4) the rand index of sample clustering (RI). Note that for observation i, its predicted response is given by $\Sigma_{k=1}^{K} z_{ik}(\boldsymbol{x}_{\mathrm{i}}^T \boldsymbol{\beta}_k)$, where $z_{\mathrm{i}k}$ is its cluster membership indicator. The average of the four metrics over 100 simulations in each scenario was calculated and shown in Table 4.2. Here, we assume that the true K is known.

In general, CSMR performs the best in terms of the four evaluation metrics in the majority of the scenarios. For prediction accuracy of the response using correlation, CSMR and ICC perform comparably well, and CSMR slightly better in most of the cases. This is expected as LASSO, RIDGE and RF can not deal with the sample heterogeneity, and FMRS does

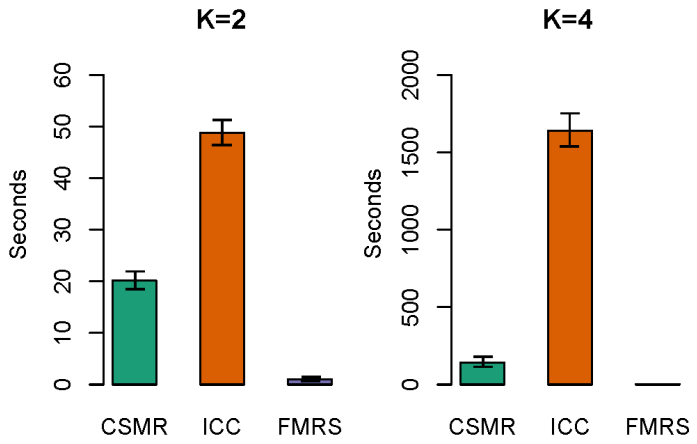


Figure 4.2. Time consumption of CSMR, and ICC on simulation data for $K = 2$ (left) and $K = 4$ (right), and $N = 400, \sigma = 1, M_0 = 20, P = 100$ over 100 repetitions, error bars indicate standard deviations.

not work well when the feature dimension is high. In particular, FMRS failed to converge for $P = 2,000$. For sensitivity and specificity of the variable selection, CSMR performs significantly better than ICC and FMRS. Selection of the correct variables is very important as it characterizes the unique features of each component, based on which, we could further deduce the biological interpretation of each unique component. ICC and FMRS suffer from very low sensitivity of variable selection in almost all cases, and their specificity metrics are not desirable either. For clustering, CSMR again has the best or close to the best performance compared with ICC and FMRS. ICC achieved similar performance with CSMR in some cases, but it clearly suffers when K or the number of true predictors M_0 becomes large. We also compared the computational efficiency of CSMR and ICC under the parameter setting: $N = 400, P = 100, \sigma = 1, M_0 = 20, K = 2$ or 4. Figure 4.2 shows the computational cost and its standard deviation for two algorithms over 100 repetitions. Clearly, the computational efficiency of ICC drops significantly when K increases from 2 to 4, while the time consumption for CSMR stays approximately the same.

Hence from simulation data, we could see that CSMR achieved the most desirable performance in terms of prediction accuracy, variable selection and clustering, compared with three

non-mixture regularized models, and two regularized mixture regression models. While ICC is competitive in some cases, it severely suffers from poor variable selection, and its computational cost is too prohibitive compared with CSMR.

In ICC, the clustering membership was treated as missing data in Markov chain. In each iteration, the ICC method automatically selects a distinct set of variables. In detail, ICC method treated the clustering membership and parameters need to be estimated as two sequences of Markov chain. The parameter sequence has a stationary distribution after iteration converged. In contrast, FMRS does variable selection for fixed hyperparameter set, and by searching among a grid of the hyperparameter sets, the variables are selected that corresponds to the hyperparameter set with the best variance-bias trade-off. CSMR inherited the advantages of both ICC and FMRS in that it conducts variable selection within each iteration step by automatically tuning the hyperparameters $\lambda_k, k = 1, ..., K$, saving the computational cost of large scale selection of λ_k, which increases exponentially with K. In addition, the highly efficient built-in cross validation step within the CEM iterations could largely increase the sensitivity and specificity of the variable selection procedure, and the flexible model refitting step following the CEM steps guarantees that the algorithm could achieve faster convergence and more stable results.

4.4 Experiments by using Real-world Data

4.4.1 Description of Real-world Data

Over the past three decades, the use of genetic data to inform drug discovery and development pipeline has generated huge excitement. Predicting the drug sensitivity becomes an integral part of the precision health initiative. Although earlier efforts successfully identified many new drug targets, the overall clinical efficacy of the developed drugs has remained unimpressive, owing in large part to the population heterogeneity, that is, different patients may have different disease causing factors, and hence drug targets. Here, we apply CSMR to study the patient heterogeneity in their response to different drug treatments, and select the most key genetic features that underlie the heterogeneous disease causes.

Table 4.2. Comparisons of CSMR with other five methods in various simulation settings

Metrics	Experiment	$\sigma=1, N=400,$ $M_0=5, P=100$ K			$K=2, N=400,$ $M_0=5, P=100$ σ			$K=2, \sigma=1,$ $M_0=5, P=100$ N			$K=2, \sigma=1,$ $N=400, P=100$ M_0			$K=2, \sigma=1,$ $M_0=5, N=400$ P		
	Parameter	2	3	4	0.5	1	2	200	300	400	5	8	20	100	500	2000
$Cor(y, \hat{y})$	**CSMR**	**0.994**	**0.988**	**0.999**	**0.998**	**0.994**	0.977	0.987	**0.993**	**0.994**	**0.994**	**0.995**	**0.992**	**0.994**	**0.994**	0.951
	ICC	0.984	0.985	0.909	**0.998**	0.984	**0.982**	**0.992**	0.992	0.984	0.984	0.994	**0.992**	0.984	**0.994**	0.993
	LASSO	0.778	0.654	0.585	0.745	0.778	0.729	0.776	0.756	0.778	0.778	0.754	0.743	0.778	0.799	0.779
	RIDGE	0.789	0.697	0.639	0.783	0.789	0.772	0.834	0.802	0.789	0.789	0.782	0.784	0.789	0.885	0.955
	RF	0.605	0.583	0.487	0.719	0.605	0.700	0.717	0.720	0.605	0.605	0.691	0.716	0.605	0.804	**1**
	FMRS	0.706	0.676	0.568	0.780	0.706	0.769	0.727	0.797	0.706	0.706	0.780	0.780	0.706	0.143	-
Variable Selection (TPR)	**CSMR**	**0.980**	**0.950**	**0.538**	**1**	**0.980**	**1**	**0.956**	**1**	**0.980**	**0.980**	**0.998**	**0.999**	**0.980**	**1**	**0.900**
	ICC	0.461	0.332	0.339	0.500	0.461	0.500	0.500	0.500	0.461	0.461	0.496	0.500	0.461	0.500	0.500
	FMRS	0.579	0.552	0.487	0.681	0.579	0.674	0.672	0.706	0.579	0.579	0.635	0.679	0.579	0.700	-
Variable Selection (TNR)	**CSMR**	**0.992**	**0.976**	**0.785**	**0.994**	**0.992**	**0.968**	**0.966**	**0.990**	**0.992**	**0.992**	**0.992**	**0.993**	**0.992**	**0.998**	**1**
	ICC	0.870	0.957	0.669	0.973	0.870	0.735	0.966	0.972	0.870	0.870	0.953	0.972	0.870	0.994	0.998
	FMRS	0.512	0.680	0.758	0.504	0.512	0.500	0.502	0.506	0.512	0.512	0.515	0.499	0.512	0.502	-
Sample Clustering (RI)	**CSMR**	**0.917**	0.833	**0.624**	**0.943**	**0.917**	**0.787**	0.852	**0.886**	**0.917**	**0.917**	**0.908**	**0.893**	**0.917**	**0.865**	0.869
	ICC	0.879	**0.838**	0.549	0.941	0.879	0.787	**0.878**	0.881	0.879	0.879	0.903	0.887	0.879	0.856	**0.886**
	FMRS	0.513	0.546	0.624	0.502	0.513	0.502	0.501	0.501	0.513	0.513	0.502	0.501	.513	0.502	-

We collected gene expression data of 470 cell lines on 7902 genes, as well as the cell lines' sensitivity score to all 24 drugs, from the Cancer Cell Line Encyclopedia (CCLE) dataset [131]. The sensitivity score, also called the AUCC score, is defined as the area above the fitted dose response curve, and it has been shown to have better predictive accuracy of sensitivity to targeted therapeutic strategies than other measures, such as IC50 or EC50 [132]. We applied all five methods on the dataset, where the drug sensitivity score was treated as the response variable and the gene expressions as independent variables. Here, FMRS is not applicable as the feature dimension is too high while the sample size is too small, hence it is omitted from further analysis. Our goal is to study the biological mechanism of possible heterogeneity in drug sensitivity, under the hypothesis that cells exhibit subgroup characteristics by selecting different genes that confer their different levels of drug sensitivity.

4.4.2 Results

We compare the performances of the five methods using cross validation. Basically, for each drug, we conduct a 5-fold cross validation by holding 80% of the data as training, and 20% as testing data, for each of the 100 repetitions. At each repetition, the 20% testing data is used to independently evaluate the performance of each method. At the training phase, we start by fixing the hyper parameters involved in all methods. The penalty parameters for LASSO and RIDGE were selected by cross validation within the training samples. For RF, the default parameters were used in the function 'randomForest' of the package with the same name. The default parameters were described as follow. The number of trees to grow is 500. For regression problem, the minimum size of terminal nodes is 5. For ICC, we used the selected component number as in its original paper [87]. For CSMR, to select the best K, we performed both cross validation and the traditional BIC criteria, over a grid of $K = 1, 2, 3, 4, 5, 6$. However, the traditional BIC criteria is proposed for low dimensional setting instead of high dimensional setting. Thus, there is a lack of rigorous theoretical foundation for the validity of the traditional BIC under this high dimensional setting, and the data driven selection of cross validation seems more reasonable. With the

hyper parameters fixed, we then conduct parameter estimations for each of the five methods using the training samples, and concludes the training phase.

At the testing phase, the predicted and true drug sensitivity scores were examined in terms of their correlation, and residual mean squared error (RMSE). The distributions of RMSE and correlations over 100 repetitions for all the 24 drugs for all the five methods were shown in Figure 4.3 and Figure 4.4, respectively. For 22 drugs, CSMR had the significantly smaller average RMSE, and was very close to the smallest RMSE for the rest of the two drugs; and we could make the same conclusions based on the correlation results as well. This demonstrated the consistent and robust performance of CSMR over the others.

Among the five methods, RF had the poorest performance on the testing data, probably caused by model overfitting. LASSO and RIDGE worked much better than RF, probably due to its power in model selection. However, they performed significantly worse than ICC and CSMR in the majority of the cases, which indicates the existence of population heterogeneity and necessity of using mixture modeling. The performance of ICC is much worse than CSMR in most of the cases, which we believe is caused by the under-estimation of the population heterogeneity by ICC. In other words, the selection of K in ICC is too conservative. In fact, according to cross validation, the number of distinct clusters given by CSMR for the drugs is either 3 or 4, while for ICC, the number of distinct clusters are determined to be less than 3 for half of the drugs. We believe that cross validation is a data driven approach for selection of K, and should be more reasonable than theoretically derived criteria. In the case of CCLE data, the samples are different types of cells from very different experimental and genetic backgrounds, and it is expected that they would pick up different molecular mechanisms to deal with the attacks of the drugs. Hence, the cluster number given by CSMR is more realistic than ICC. It is note that for those drugs that CSMR and ICC gave the same number of distinct clusters, namely Irinotecan, L-685458, Lapatinib, Paclitaxel, PD-0332991, PHA-665752 and TKI258, CSMR exhibited much smaller RMSE than ICC.

Figure 4.5 demonstrated the Venn diagram of the selected genes for different components for each drug. It could be seen that for the same drug, different clusters of cells indeed acquire different coping mechanisms, as seen by the different set of genes selected. This again confirms the high heterogeneous populations within the CCLE cohort. For each drug,

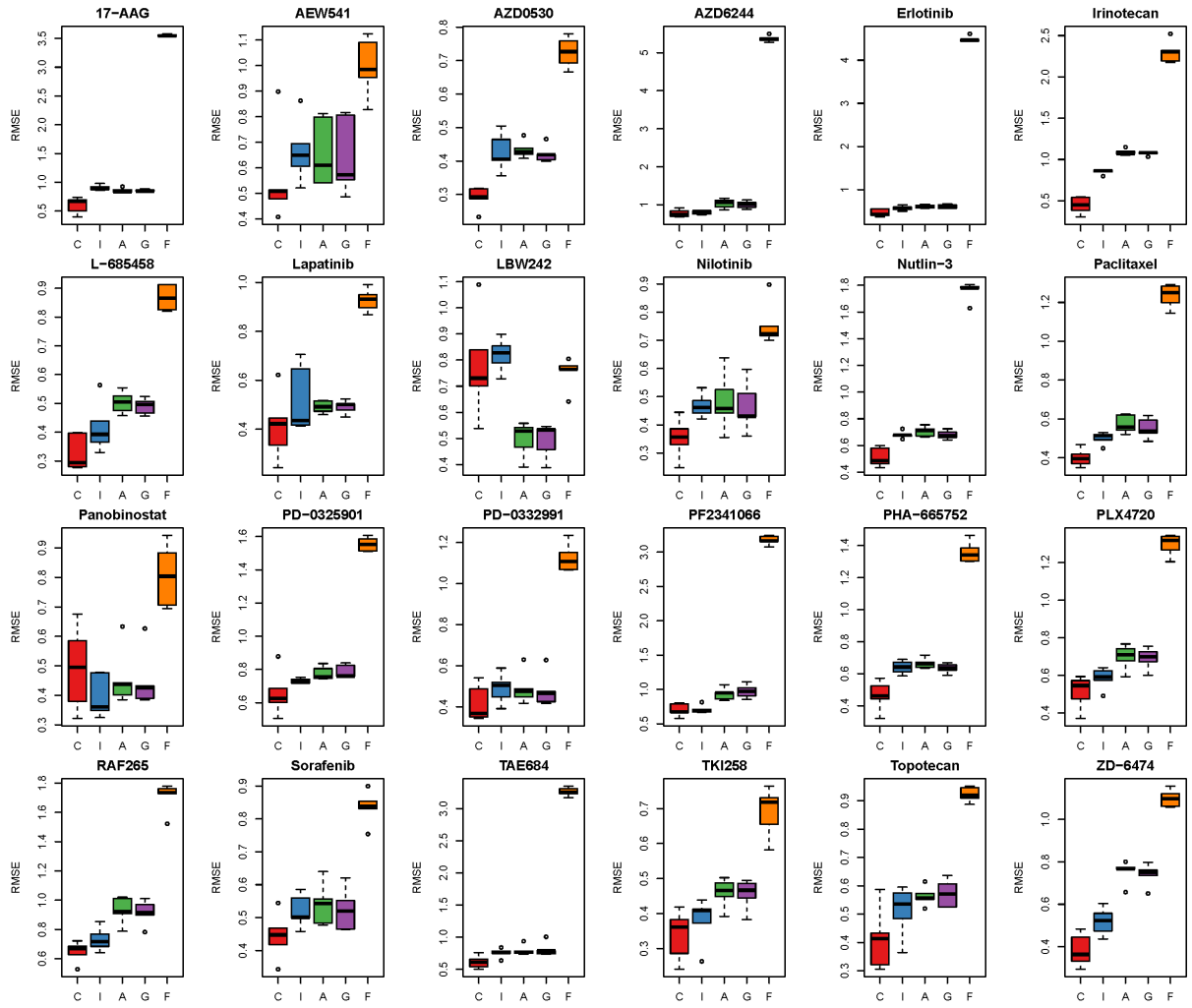


Figure 4.3. The distributions of the RMSE over 100 repetitions for the five methods, for the 24 drugs. The lower the RMSE value, the better the performance. 'C','I','A','G','F' stand for 'CSMR','ICC','LASSO','RIDGE','Random Forest'.

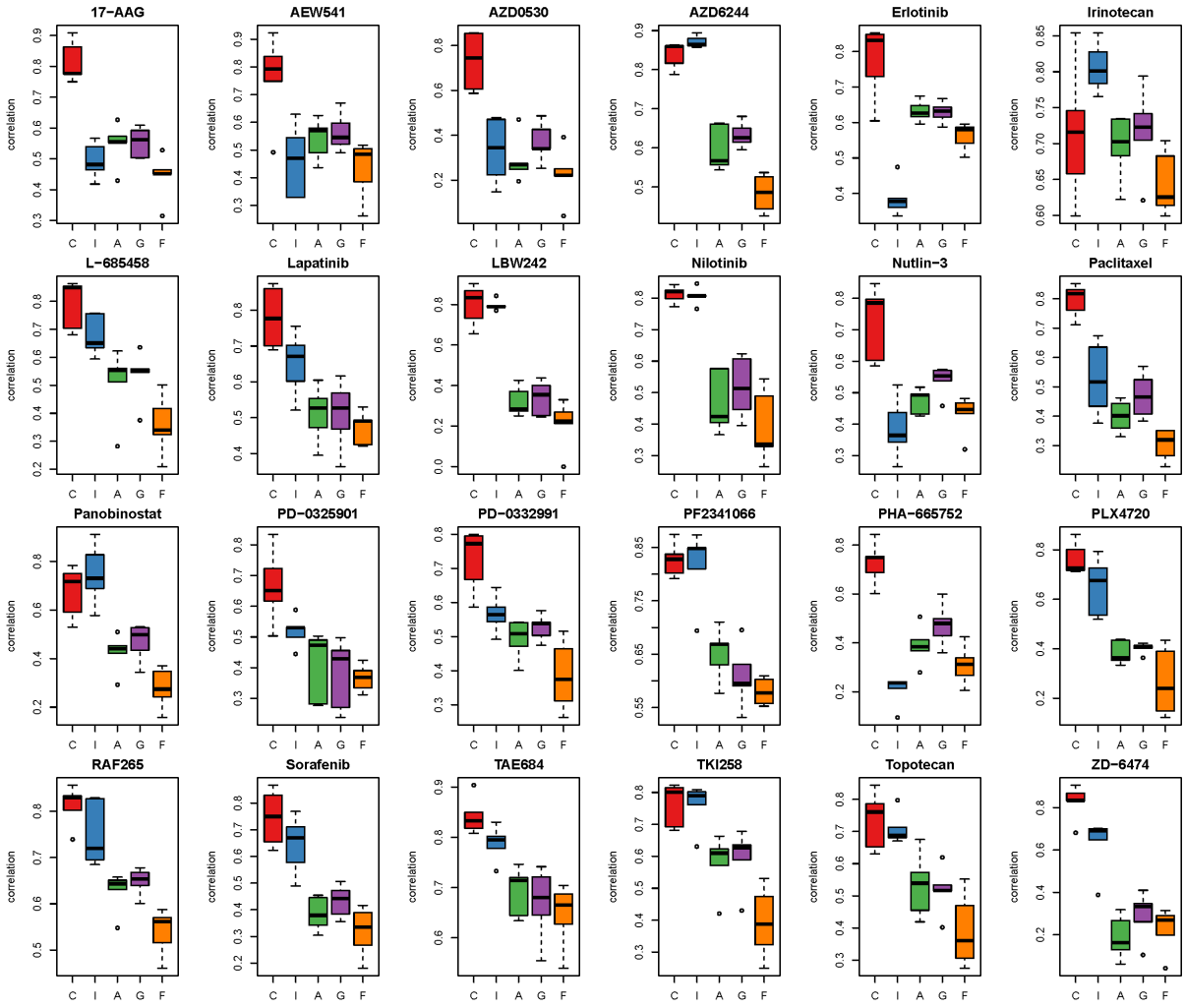


Figure 4.4. The distributions of the correlation over 100 repetitions for the five methods, for the 24 drugs. The higher the correlation value, the better the performance. 'C','I','A','G','F' stand for 'CSMR','ICC','LASSO','RIDGE','Random Forest'.

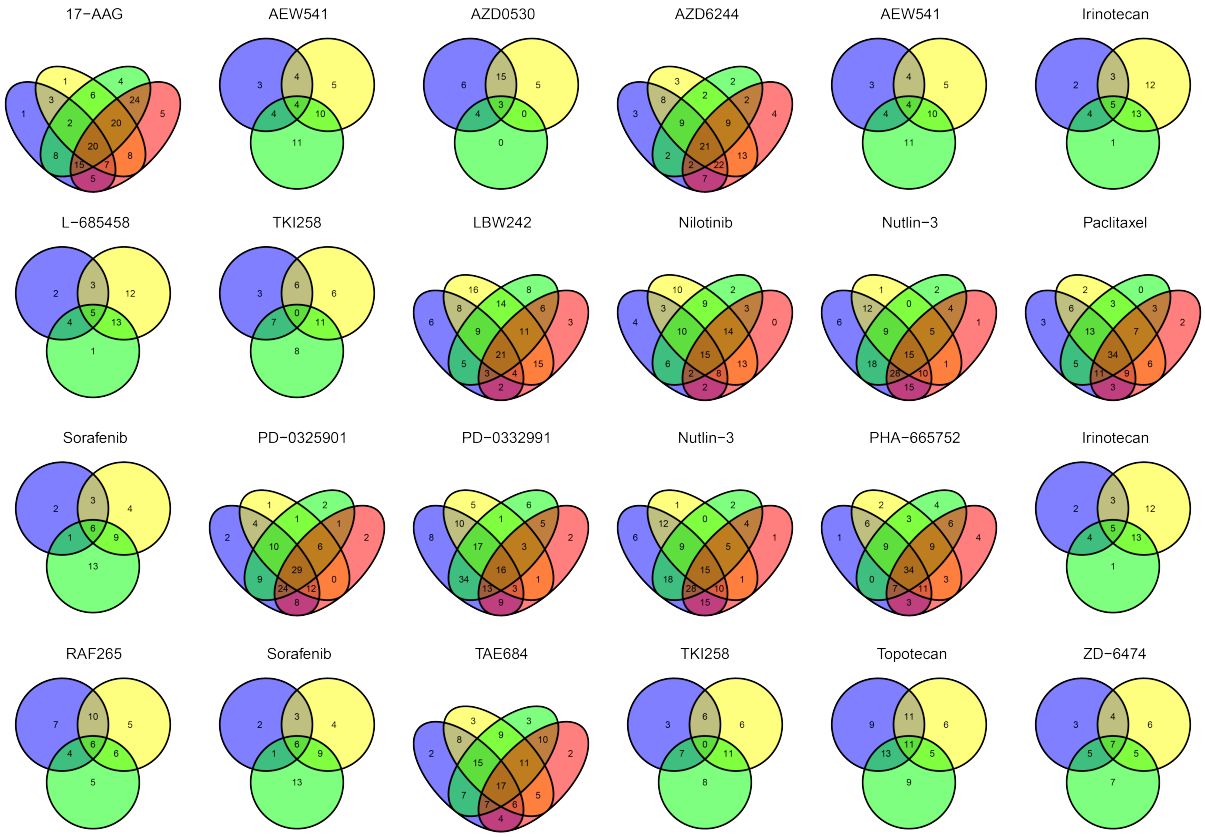


Figure 4.5. The distributions of the correlation over 100 repetitions for the five methods, for the 24 drugs. The higher the correlation value, the better the performance. 'C','I','A','G','F' stand for 'CSMR','ICC','LASSO','RIDGE','Random Forest'.

we pooled all the selected genes together and conducted pathway enrichment analysis against 1,328 pathways collected in [133]. Again, it could be seen that different responses to different drugs have been employed.

4.5 Conclusions

With the recent rapid evolution in genomic technologies, we have now entered a new phase, one in which it is possible to comprehensively characterize the genetic profiles of large population of subjects. Importantly, the development of sequencing technologies has been paired with a transition towards integrating genetic data with phenotypic data, such as in

electronic medical records. Such a synergy has the potential to ultimately facilitate the generation of a data commons useful for identifying relationships between genetic variations and their clinical presentations. Unfortunately, existing big data analysis tools for mining the information rich data commons has not been very impressive with regards to the overall translational or clinical efficacy, owing in large part to the heterogeneous causes of disease. It is hence imperative to unveil the relationship between the genetic variations and the clinical presentations, while taking into account the possible heterogeneity of the study subjects.

In this chapter, we proposed a novel supervised clustering algorithm using penalized mixture regression model, called CSMR, to deal with the challenges in studying the heterogeneous relationships between high dimensional genetic features and a phenotype that serves as a response variable to guide the clustering. CSMR is capable of simultaneous stratification of the sample population and sparse feature-wise characterization of the subgroups. The algorithm was adapted from the classification expectation maximization algorithm, which offers a novel supervised solution to the clustering problem, with substantial improvement on both the computational efficiency and biological interpretability. Experimental evaluation on simulated benchmark datasets with different settings demonstrated that the CSMR can accurately identify the subspaces on which a subset of features are explanatory to the response variables and the feature characteristics of the subspaces, and it outperformed the baseline methods. Application of CSMR on the heterogeneous CCLE dataset demonstrated the superior performance of CSMR over the others. On the CCLE dataset, CSMR is powerful in recapitulating the distinct subgroups hidden in the pool of cell lines with regards to their coping mechanisms to different drugs. CSMR also demonstrated the uniqueness of different subgroups for the same drug, as seen by the distinctly selected genes for the subgroups.

In summary, CSMR represents a big data analysis tool with the potential to bridge the gap between advancements in biotechnology and our understanding of the disease, and resolve the complexity of translating the clinical representations of the disease to the real causes underpinning it. We believe that such a tool will bring new understanding to the molecular basis of a disease, and could be of special relevance in the growing field of personalized medicine.

5. SPATIALLY AND ROBUSTLY HYBRID MIXTURE REGRESSION MODEL FOR INFERENCE OF SPATIAL DEPENDENCE

This chapter focuses on the challenges in investigating the relationship between response variable and a set of explanatory variables over the spatial domain. Our assumption is that the relationships may exhibit complex spatially dynamic patterns that cannot be captured by constant regression coefficients. We proposed the SRMR method which integrates the robust finite mixture Gaussian regression model with spatial constraints, to simultaneously handle the spatial nonstationarity, local homogeneity, and outlier contaminations.

In this chapter, we have three major contributions by considering the spatially and robustly mixture regression.

1. We developed the very first computational concept of spatially dependent mixture regression analysis.

2. We provided the SRMR method that efficiently solves the spatially dependent mixture regression problem, which is also empowered by a statistical inference approach to assess regression significance.

3. SRMR method enables a new type of spatial segmentation analysis to detect overlapped spatial regions of varied dependencies among subset of features, which have high contextual meaningfulness.

Spatial transcriptomics is a newly emerged data type, in which each sample point has a measured high dimensional data and spatial coordinates. Revealing linear dependency that is specific to the samples within a spatial region could provide unseen knowledge in the system. (i) Shifts of functional dependency of genes through different regions may directly suggest the phenotypic difference among spatial regions. It could not only reveal detailed molecular changes but also provide a new perspective in dissecting spatial regions into functionally similar groups. (ii) On spatial transcriptomics data, this approach could be directly applied to identify differential ligand-receptor based or non-ligand based cell-cell and cell-microenvironment interactions, as well as deriving regions that may have specific phenotypic variations or drug resistance. (iii) We have demonstrated the method could

be applied to other fields such as economics or geographic data to study regional specific dependency.

5.1 Introduction

10x Genomics spatial transcriptomics (ST) is a recent commercialized technique to measure spatial coordinates associated gene expression signal from a biological tissue sample, which it has a broad utilization in biomedical research. A typical ST data is a matrix consisting of ∼15,000 genes (rows) in ∼4,000 individual spatial spots (columns), and each spot has a 2D spatial coordinate. The spatial spots are uniformly distributed. A key challenge in ST data analysis is to infer the spatially dependent and biologically meaningful functional variations from the high dimensional feature matrix (genes by spatial spots).

Compared with existing spatial regression models, our proposed model assumes the existence a few distinct regression models that are estimated based on observations that exhibit similar response–predictor relationships. As such, the proposed model not only accounts for nonstationarity in the spatial trend, but also clusters observations into a few distinct and homogenous groups. This provides an advantage on interpretation with a few stationary sub-processes identified that capture the predominant relationships between response and predictor variables. Moreover, the proposed method incorporates robust procedures to handle contaminations from both regression outliers and spatial outliers. By doing so, we robustly segment the spatial domain into distinct local regions with similar regression coefficients, and sporadic locations that are purely outliers.

The remainder of this chapter is organized as follows. In Section 5.2, we will introduce the problem and proposed SRMR algorithm. In Section 5.3, we show the performance comparison of our method with other state of the art methods on synthetic datasets. In Section 4.4, we will apply all methods to two types of real world datasets.

5.2 Methods

5.2.1 Problem Statement

We denote scalar value, vector, and matrix as lowercase character x, bold lowercase character $\mathbf{x}$, and uppercase character X, respectively. Let $\{(\boldsymbol{x}(s_\mathrm{i}), y(s_\mathrm{i})), \mathrm{i} = 1, \ldots, n\}$ represent a set of spatial data that is observed at spatial locations $s_1, \ldots, s_n \in \mathbb{R}^2$, where the response variable $y(s_\mathrm{i})$ is assumed to be spatially correlated, $\boldsymbol{x}(s_\mathrm{i}) = (x_1(s_\mathrm{i}), \ldots, x_p(s_\mathrm{i}))^\mathrm{T}$ is the p-dimensional vector of explanatory variables for the observation located at s_i, and $s_\mathrm{i} = (c_\mathrm{i}^1, c_\mathrm{i}^2)$ is the 2-dimensional coordinate of the ith location. In this chapter, we only describe and validate the SRMR model on 2-dimensional spatial data. Noted, the approach can be directly applied to K-dimensional ($K > 2$) spatial data.

To capture the spatially dependent structure for the response variable, we write a standard generalized linear regression model (GLM) for the i-th spatial location as follows,

$$g(E(y = y(s_\mathrm{i}) \mid \boldsymbol{x} = \boldsymbol{x}(s_\mathrm{i}))) = \sum_{\mathrm{j}=1}^{p} x_\mathrm{j}(s_\mathrm{i})\beta_\mathrm{ji} + \epsilon_\mathrm{i} \tag{5.1}$$

where $\beta_\mathrm{ji}, \mathrm{j} = 1, 2, \ldots, p$, are the regression coefficients for the p predictors, and ϵ_i represents random noise with mean 0 and variance σ_i^2, and $g(\cdot)$ is the probability density function. In this method, we assume linear regression follows same format of probability density function. The intercept can be accommodated by including 1 as an entry of $\boldsymbol{x}(\boldsymbol{s}_\mathrm{i})$. Apparently, unless with sufficient number of repeated measurements for each location, the $\beta_\mathrm{ji}, \sigma_\mathrm{i}$ are non-identifiable. In many cases, there is only a single observation for each spatial location, certain spatial constraints will be enforced to ensure the identifiablity of the model parameters.

Definition 5.2.1. ***Spatially Dependent Mixture Regression.*** *Given a dataset consisting of n observations $\{(\boldsymbol{x}(s_\mathrm{i}), y(s_\mathrm{i})), \mathrm{i} = 1, \ldots, n\}$ from spatial locations $s_1, \ldots, s_n$, the goal of spatially dependent mixture regression is to identify spatial regions $\Pi_1, \ldots, \Pi_K$ and the number K, s.t.,*

$$y(s_\mathrm{i}) = \sum_{\mathrm{j}=1}^{p} x_\mathrm{j}(s_\mathrm{i})\beta_\mathrm{j}^k + \epsilon_\mathrm{i}, \ \mathrm{if} \ s_\mathrm{i} \in \Pi_k$$

, where $\beta_{\mathrm{j}}^k, \mathrm{j} = 1, ..., p, k = 1, ..., K$ *are regression parameters for the* p *predictors in the* k*-th cluster;* $\epsilon_{\mathrm{i}} \sim \mathcal{N}(0, \sigma_k)$*, where* σ_k *represents the noise level of cluster* k*.*

To account for the presence of outliers, we assume that Π_K are non-overlapping subsets of the whole set $\{1, ..., n\}$, and denote the outlier set as Π_0, such that $\Pi_0 = \{1, ..., n\} \setminus \bigcup_{k=1}^{K} \Pi_k$ Two type of outliers will be considered here:

Type 1 Outliers: $y(s_{\mathrm{i}}) \neq \sum_{\mathrm{j}=1}^{p} x_k(s_{\mathrm{i}})\beta_{\mathrm{j}}^k + \epsilon^k, \forall k = 1, ..., K$

Type 2 Outliers: $\exists k, y(s_{\mathrm{i}}) = \sum_{\mathrm{j}=1}^{p} x_k(s_{\mathrm{i}})\beta_{\mathrm{j}}^k + \epsilon^k, s_{\mathrm{i}} \notin \Pi_k$

Here the **Type 1 Outliers** represent the samples do not fit any regression model while the **Type 2 Outliers** represent the ones fit a certain model but do not locate nearby the spatial region.

Noted, pre-assumptions of the spatial regions $\Pi_1, \ldots, \Pi_K$ are needed to enable a valid solution of the spatially dependent mixture regression problem. Such assumptions include a connected spatial region, a compact shape, or high enrichment to a certain region. Noted, as spatially dependent mixture regression assigns each sample into one spatial region Π_k, it directly forms a spatial segmentation method.

5.2.2 Related Works

Spatially smooth regression. Conventional nonstationary spatial regression models such as geographically weighted regression (GWR) [93]–[95] and Bayesian spatially varying coefficient (SVC) [96], [97] models allow regression coefficients to vary smoothly as a function of the spatial domain. For GWR, assuming a linear model with $\boldsymbol{y}$ denoting the observed response vector and $\boldsymbol{X}$ the design matrix, the regression coefficient at the i th location is estimated from $\widehat{\boldsymbol{\beta}}_{\mathrm{i}} = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}_{\mathrm{i}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}_{\mathrm{i}}\boldsymbol{y}$, where $\boldsymbol{W}_{\mathrm{i}}$ is a diagonal weight matrix defined by a kernel function of distance of all other points to point i. The challenge with GWR and SVC models is that they fit as many regression models to the data as there are observations, at the cost of a large computational burden, possible over-fitting and interpretation. A penalized spatial regression model has been developed to automatically detect clusters [98] by incorporating a fused-lasso penalty constructed based on spatial proximity.

Spatial segmentation. Model-based spatial segmentation aims perform a segmentation task to all samples (e.g. pixels in image) based on the input features. Model-based spatial segmentation adopts an energy function $U(\Pi)$ to integrate the spatial information such as neighborhoods with a regular clustering analysis of the features. Intrinsically, such methods leverage spatial and data consistency to segment spatial regions, i.e., only considering the covariance of independent variables, which cannot solve the spatially dependent regression problem.

5.2.3 SRMR algorithm and Mathematical Consideration

To solve the problem of spatially dependent mixture regression, computational challenges arise from three aspects: (1) the mixture regression model and spatial consistency do not form one unified likelihood function, which prohibits a direct solution by using EM algorithm, (2) detection spatial regions should depend on both goodness of fitting and spatial consistency, and (3) there is lack of a validate approach to assess the statistical significance of mixture regression models.

In sight of the challenge, we developed the spatial robust mixture regression (SRMR) algorithm to conduct simultaneous outlier detection and spatially dependent mixture regression estimation. The underlying idea is that by assuming a likelihood function of spatial regions p_{spa} and introducing a tuning parameter $\lambda \in (0, 1)$ to link p_{spa} with the likelihood of mixture regression p_{reg}, a surrogate likelihood function $(1 - \lambda)p_{reg} + \lambda p_{spa}$ is developed to enable a modified EM-algorithm (Algorithm 5). The inputs of Algorithm 5 include the response and independent variables, spatial coordinates, and the hyper parameter λ. It conducts a simplified spatially dependent mixture regression fitting by assuming there is only Type 2 outliers, i.e., the sample fit one mixture model but do not locate in the corresponding spatial region. Hence, Algorithm 5 fits a conventional mixture regression model and computes the spatial regions that are top enriched by the samples fit each regression component. In this study, we assume the spatial likelihood follows $p_{spa}(z_{\mathrm{i}} = k \mid s_{\mathrm{i}}, w) \propto \|s_{\mathrm{i}}, w\|_2$, where z_{i} represents the class of sample i and $\|s_{\mathrm{i}}, w\|_2$ represents the Euclidean distance between the spatial coordinate of the sample s_{i} and the centers of the spatial regions w, i.e., assuming

the spatial regions form a compact shape. Specifically, a voting step (C-step) is introduced in Algorithm 5, which identifies Type 2 outliers by the ones whose most likely regression component and spatial region are not consistent. Noted, as all the input samples are utilized in the estimation of the mixture regression model, Algorithm 5 is always convergent.

Based on the Algorithm 5, we developed the SRMR framework (Algorithm 6). In SRMR, we iteratively conduct the Type 2 outlier only spatially dependent mixture regression by using the Algorithm 5 and identify Type 1 outliers by running a robust linear regression on all the samples predicted to each spatial region. The underlying consideration is that only one regression component is consisted within each identified spatial region, which could be effectively identified by a conventional robust regression approach (**RLM**). In SRMS, we implement the trimmed likelihood estimation based robust mixture regression. The inputs of SRMR is the same as the input of Algorithm 5 plus the maximal iteration number L_0 and a random seed. The outputs of SRMR include the identified mixture regression models and outliers. The component of each non-outlier samples can be further assigned by maximal likelihood. In SRMR, we utilize the same BIC function for conventional robust mixture regression analysis.

5.2.4 Statistical Inference

Hypothesis testing for spatial regions

We conducted a geometry based approach to estimate the significance to observe a spatial region of a certain size. Noted, we utilized the compact spatial shape assumption in SRMR, which could be considered as a round shape. For a round shape with a diameter of r, the number of the shapes needed to cover a rectangular spatial region can be computed by $0.28m \times n/r^2$, which serves as a weight to correct the p value assessed from each single component robust regression as detailed following.

Hypothesis testing for robust linear regression

We discuss hypothesis testing of the significance a robust linear regression model parameterized by $\hat{\boldsymbol{\theta}} = \{\hat{\boldsymbol{\beta}}, \hat{\sigma}, \hat{\boldsymbol{\eta}}\}$, which represents the robust regression coefficients estimator, standard deviation estimator, and the index of the outlying samples respectively. A boot-

Input: Response vector Y; independent variables in matrix $X^{N\times(P+1)}$; the number of mixing component K; size of initialization random sample n_0; 2-dimentional spatial coordinates $S^{N\times 2}$; hyperparameter λ.

Output: Partition $\mathcal{C}^* = \bigcup_{k=1}^{K} C_k$; Mixture regression model parameter estimate $\boldsymbol{\theta}^*$; spatial centriod parameter w^* ; type 2 outlier set U^*

Initialization: $\mathcal{C} = \{C_1, ..., C_K\}, C_{\mathrm{i}} \subseteq \{1, ..., N\}$ based on coordinate S; compute centroid point $w = \{w_1, ..., w_K\}$ with $\mathcal{C}$

for $m = 0, ..., L_0$ *or until convergence* **do**

E-step: Compute for $\mathrm{i} = 1, ..., N$ and $k = 1, ..., K$, the hybrid posterior probabilities $p_{\mathrm{i}k}^{(m)}$ by

$p_{\mathrm{i}k}^{(m)} = (1-\lambda) p_{reg}(z_{\mathrm{i}} = k \mid \boldsymbol{x}_{\mathrm{i}}, y_{\mathrm{i}}, \boldsymbol{\theta}^{(m)}) + \lambda p_{spa}(z_{\mathrm{i}} = k \mid S_{\mathrm{i},:}, w^{(m)})$

C-step: For $\mathrm{i} = 1, ..., K$, assign $C_k^{(m)} = \{\mathrm{i} \mid \underset{l\in\{1,...,K\}}{\operatorname{argmax}}\ p_{il}^{(m)} = k, \mathrm{i} = 1, ..., N\}$, and let $n_k^{(m)}$ be the size of $C_k^{(m)}$,

$U_k^{(m)} = \{\mathrm{i} \mid z_{\mathrm{i}r} \neq z_{\mathrm{i}s}, z_{\mathrm{i}r} = \underset{z_{\mathrm{i}r}\in\{1,...,K\}}{\operatorname{argmax}}\ p_{reg}, z_{\mathrm{i}s} = \underset{z_{\mathrm{i}s}\in\{1,...,K\}}{\operatorname{argmax}}\ p_{spa}\}$

M-step: For $k = 1, ..., K$, the parameters are then updated by

$\boldsymbol{\pi}_k^{(m+1)} = \frac{n_k^{(m)}}{\sum_{l=1}^{K} n_l^{(m)}}$, $(\boldsymbol{\beta}_k^{(m+1)}, \sigma_k^{2(m+1)}) = \mathbf{OLS}(Y_{C_k^{(m)}}, X_{C_k^{(m)},:})$,

$w_k = \frac{1}{n_k^{(m)}} \sum_{\mathrm{i}\in C_k} S_{\mathrm{i},:}$

end

Algorithm 5: Hybrid Mixture Regression (HMR)

Input: Response vector Y; independent variables in matrix $X_{N\times(P+1)}$; the number of mixing component, K; size of initialization random sample, n_0; the maximum number of iteration L_0; the number of random starts J

Output: Partition $\mathcal{C}^* = \bigcup_{k=1}^{K} C_k$; robust FMGR parameter estimate $\boldsymbol{\theta}^* = \boldsymbol{\theta}^{J_0}$; outlier set $U^* = U^{J_0}$

for $\mathrm{j} = 0, ..., J$ **do**

  Initialization: $U^{(old)} = \{1, ..., N\}; U^{(cur)} = \emptyset; L = 0$

  for *k=1,...,K* **do**

    Draw a random sample of size n_0 from set $\{1, ..., N\}$, indexed by I_k

    Run robust linear regression: $(M_k, \beta_k, \sigma_k) =: \mathbf{RLM}(y_{I_K} \sim X_{I_k})$

    Initialize posterior probability: $p_{\mathrm{i}}k = \mathcal{N}(y_{\mathrm{i}} - x_{\mathrm{i}}^T \boldsymbol{\beta}_k; 0, \sigma_k^2), \mathrm{i} = 1, ..., N$

  end

  while $U^{(old)} \neq U^{(cur)} \& L < L_0$ **do**

    Let $U^{(old)} = U^{(cur)}; L = L + 1$

    for *k=1,...,K* **do**

      Let I_k be sample indices most likely in cluster k

      Let $U_k^{(cur)}$ be type 1 outliers of $Y_{I_k} \sim X_{I_k,:}$, using least trimmed squares robust regression

    end

    $U_{reg}^{(cur)} = \bigcup_k U_k^{(cur)}; S = \{1, ..., N\} - U_{reg}^{(cur)}$

    Update $(\boldsymbol{\theta}, w)$ by **HMR** with the rest of samples in S, let $U_{spa}^{(cur)}$ be type 2 outliers, and W be the hybrid posterior probability

  end

  $U^{\mathrm{j}} = U_{reg}^{(cur)} + U_{spa}^{(cur)}, \boldsymbol{\theta}^{\mathrm{j}} = \boldsymbol{\theta}, w^{\mathrm{j}} = w$

  Let F^{j} be a length-N binary vector whose i$-$th entry is 1 only if $\mathrm{i} \in U^{\mathrm{j}}$

end

Denote J_0 as the one such that F_{J_0} is closet to the mean of $\{F_{\mathrm{j}}, \mathrm{j} = 1, ..., J\}$

Algorithm 6: Spatial Robust Mixture Regression (SRMR)

strap procedure is adopted to test the null hypothesis $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$. We perform the following steps.

Step 1: Calculating the residuals for all observations, including the outlier samples, under regression parameter $\hat{\boldsymbol{\beta}}, \hat{\sigma}$, denoted as $\boldsymbol{\epsilon} = \{\epsilon_1, ..., \epsilon_n\}$. Let $\boldsymbol{\epsilon}_{out}$ be the residuals corresponding to outlying samples, and ϵ_0 be smallest absolute residual in $\boldsymbol{\epsilon}_{out}$.

Step 2: Generate iid sample $\tilde{\epsilon}_1, \ldots, \tilde{\epsilon}_n$ from the normal distribution $\mathcal{N}(0, \hat{\sigma})$, denoted as $\tilde{\boldsymbol{\epsilon}} = (\tilde{\epsilon}_1, \ldots, \tilde{\epsilon}_n)$.

Step 3: Calculate the percentage of samples in $\tilde{\boldsymbol{\epsilon}}$ whose absolute values are larger than ϵ_0, and denote it as p_0.

Step 4: Repeat steps 2-3 for B times, and the statistical significance is evaluated as the average of p_0 for the B times.

5.2.5 Algorithm Discussion

Several prominent features make our proposed approach attractive. First instead of using a robust estimation criterion or complex heavy-tailed distributions to robustify the mixture regression model, our method is built upon a spatial regression model so as to facilitate computation and model interpretation. Second we adopt a sparse and scale-dependent mean-shift parameterization. Each observation is allowed to have potentially different outlying effects across different regression components, which is very flexible. Compared to existing spatial regression methods, our approach allows an efficient solution via the accelerated penalized regression approach, and different information criteria (such as AIC and BIC) can be used to adaptively determine the proportion of outliers. In the next section, we utilized extensive simulations to demonstrate the performance of SRMR and its highly robustness to both gross outliers and high leverage points.

5.3 Experiments by using Simulated Data

5.3.1 Selection of Baseline Methods

We collected in total nine existing methods to represent the current works. In the field of mixture regression, Pan et al. [134] proposed DC-ADMM which cluster mixture content

in a group pursuit way. It has an implementation as "PRclust" [1] *R* package. In the field of robust mixture regression, we collected two state-of-the-art algorithms, Trimmed Likelihood Estimation (TLE) and Component-wise adaptive Trimming Likelihood Estimation (CTLE) from *R* package "RobMixReg" [2] in CRAN [135]. In the field of spatial smooth regression, we collected four algorithms, spatially clustered coefficient regression (SCC)[3] [98], Spatialculster [4] [136], Spdep [5] [137], and ClustGeo [6] [138]. However, only ClustGeo can be executed under our formulation. In the field of segmentation methods based on Markov Random Field, we collected two methods FRGMM [7] [101] and mrf2d [8] [139]. However, these two methods aim to clustering image pixels, which requires natural spatial orders from neighborhood pixels as input, and hence cannot be applied to solve our problem. Finally, we used four baseline methods DC-ADMM, TLE, CTLE, and ClustGeo to perform comparison experiments. All baseline methods used with their default parameters, except **nit** parameter in TLE and CTLERob were set as 10. For DC-ADMM, we used **stability-prclust** function to select the best parameter, followed by the instruction. For ClustGeo, we used **choicealpha** function to select best parameter.

5.3.2 Simulation Settings

To simulate spatially dependent linear relationships, we first generate a univariate independent variable x from uniform distribution $X \sim U(-2, 2)$ and dependent variable y by $y_{\mathrm{i}} = \beta_k x_{\mathrm{i}} + \sigma_k, k = 1, ..., K, \mathrm{i} = 1, ..., n$, where K is number of mixture models and β is regression coefficient. Spatial coordinate of each sample s_{i} was generated from a multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$, where μ determines the center and Σ determines the range and shape of each spatial region. We use $\mu_1 = [1, 1]^T, \mu_2 = [-1, -1]^T, \Sigma = diag(0.1, 0.1)$ as the default experimental setting, i.e. $K = 2$ of two distinct and non-uniformly distributed

[1]↑https://github.com/ChongWu-Biostat/prclust
[2]↑https://cran.r-project.org/web/packages/RobMixReg/
[3]↑https://github.com/furong-tamu/Supplementary-files-for-SCC
[4]↑https://github.com/mpadge/spatialcluster
[5]↑https://github.com/r-spatial/spdep/
[6]↑https://cran.r-project.org/web/packages/ClustGeo/
[7]↑https://sites.google.com/site/nguyen1j/home/10-code
[8]↑https://freguglia.github.io/mrf2d/

spatial regions. The two types of outliers wer further simulated. We simulated Type 1 outliers by a rejection sampling approach. Specifically, we first samples independent (x, y) from $(U(-2, 2), U(-8, 8))$ and only accept the ones whose Euclidean distance to the regression lines larger than two as Type 1 outliers. To simulate the Type 2 outliers of a certain ratio, we randomly select the ratio of samples and reverse their spatial coordinate $s_{\mathrm{i}} = (c_{\mathrm{i}}^1, c_{\mathrm{i}}^2)$ by $s_{\mathrm{i}}^o = (-c_{\mathrm{i}}^1, -c_{\mathrm{i}}^2)$.

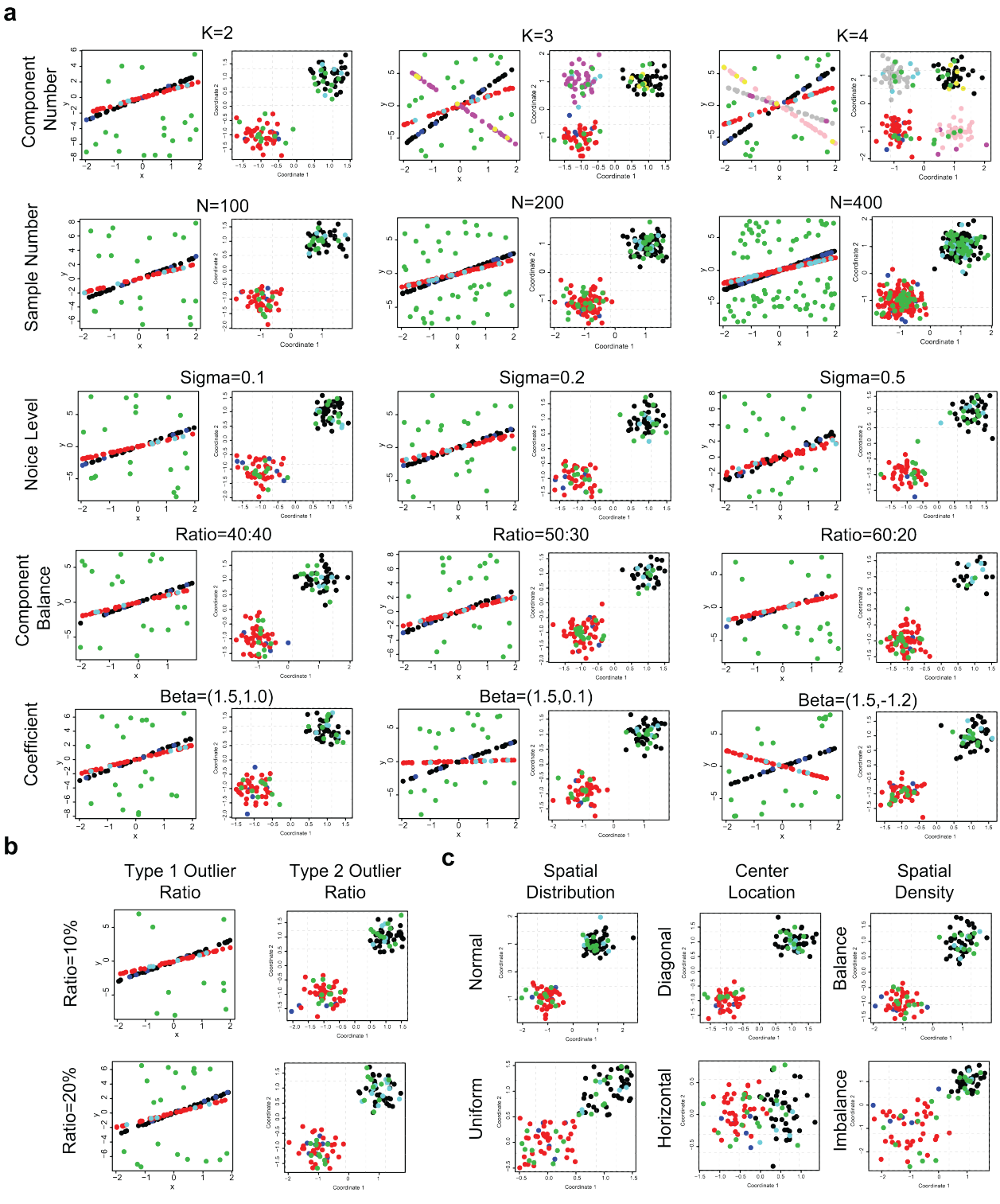
We conducted the synthetic data based experiments for three types of method evaluation:

(1) We evaluated the general performance of SRMR and baseline methods in solving the spatially dependent mixture regression problem by the following experimental setups (Figure 5.1A). Each time we perturbed one of the five factors and fixed the others, including number of mixture regression models $K = \{2, 3, 4\}$, total sample size $N = \{100, 200, 400\}$, error of linear regression $\sigma = \{0.1, 0.2, 0.5\}$, rate of samples belong to (model_1, model_2, outliers)=$\{(0.4, 0.4, 0.2), (0.5, 0.3, 0.2), (0.6, 0.2, 0.2)\}$ (only for K=2), and coefficients of linear regression model $\boldsymbol{\beta} = \{(1.5, 1.0), (1.5, 0.1), (1.5, -1.2)\}$ (only for K=2).

(2) We validated the robustness of SRMR and baseline methods in handling the two types of outliers, namely Type 1 and 2 outliers by perturbing their ratio from 10% to 20% (Figure 5.1B).

(3) We validated the capability of SRMR and baseline methods in detecting different shapes and distributions of spatial regions. We simulated the spatial coordinates from a multivariate normal distribution or a multivariate uniform distribution, the former one simulates a round and dense spatial region while the later one generates uniformly distributed 2D coordinates. The simulated shapes are showcased in Figure 5.1C. In addition, we also evaluated if SRMR is sensitive to different relative positions of the spatial regions. We simulated two types of relative location of spatial regions, namely (i) diagonal distribution by setting $\mu_1 = [1, 1]^T, \mu_2 = [-1, -1]^T$ and (ii) horizontal distribution by setting $\mu_1 = [0.5, 0]^T, \mu_2 = [-0.5, 0]^T$. To simulate spatial regions of imbalanced densities, we perturbed the covariance matrix of the spatial coordinates from $diag(0.1, 0.1)$ to $diag(0.5, 0.1)$.

Figure 5.1. Experiment Setting. Sub-figures without grid represent linear relationship and sub-figures with grid represent spatial coordinates. For (b) and (c), we only show partial plot which control factor is changed instead of full plot (linear relationship and spatial coordinate) as (a). (a) contains five different scenarios in terms of mixture regression. (b) contains two scenarios to deal with Type 1 and Type 2 outliers. (c) contains three scenarios for detecting different shapes and distributions of spatial regions.


a
K=2
K=3
K=4
Component Number
N=100
N=200
N=400
Sample Number
Sigma=0.1
Sigma=0.2
Sigma=0.5
Noice Level
Ratio=40:40
Ratio=50:30
Ratio=60:20
Component Balance
Beta=(1.5,1.0)
Beta=(1.5,0.1)
Beta=(1.5,-1.2)
Coefficient
b
Type 1 Outlier Ratio
Type 2 Outlier Ratio
Ratio=10%
Ratio=20%
c
Spatial Distribution
Center Location
Spatial Density
Normal
Diagonal
Balance
Uniform
Horizontal
Imbalance

5.3.3 Evaluation Metrics

We evaluated the performance of SRMR and baseline methods on synthetic datasets, based on how accurate the methods can identify the simulated mixture regression models and corresponding spatial regions, and distinguish the two types of outliers. Four evaluation metrics were utilized in the synthetic data based evaluations:

1) *Rand Index (RI)*: $= \frac{\text{number of agreeing pairs}}{\text{number of total pairs}}$ computes a similarity measure between two clusters by considering counting the sample pairs that are assigned in the same or different clusters in the predicted and true clusters.

2) *Adjust Rand Index (ARI)*: $= \frac{\text{RI- Expected (RI)}}{\text{max(RI) - Expected(RI)}}$, which is a corrected-for-chance version of RI.

3) *Accuracy Rate (ACC)* for outliear detection. ACC: $= \frac{\text{detected true outliers}}{\text{true outliers}}$ measures the accuracy for distinguishing the Type 1 and Type 2 outliers.

4) *Error of Predicted Coefficients (PCE)*: $= \sum_{k=1}^{K}(\beta_k - \beta^p_{l(k)})^2$ measures the distance between the true regression coefficient β_k of the regression components $k = 1, ..., K$ and predicted regression coefficient β^p. Here $l(k) = \underset{\text{j}}{\text{argmin}}(\beta_k - \beta^p_{\text{j}})^2$, i.e., $\beta^p_l(k)$ is the predicted coefficient closest to β_k.

5.3.4 Methods Comparisons and Performance Evaluations

We organized the synthetic data experiment results in Table 5.1 into three sections: mixture regression, robustness and spatial patterns. Overall, SRMR outperforms baseline methods in all 10 experiment settings under almost all evaluation metrics.

In Table 5.1, the first section (1st-5th blocks) illustrated the performance of SRMR and other methods in terms of the accuracies in detecting the heterogeneous linear dependencies in different scenarios, with regards to sample size, number of components, noise level, cluster balance and strength of regression coefficients. SRMR could detect the clusters and regression coefficients for each cluster very accurately, for different sample sizes, components, and it is robust to the different noise levels, imbalance of cluster sizes and small regression coefficients. Notably, because it incorporates spatial information, it is able to differentiate two clusters with very similar regression coefficients but different spatial locations. Since DC-ADMM and

ClustGeo are designed for clustering, but not regression, the evaluation metrics ACC and PCE for these two methods are filled with NaN. Although DC-ADMM proposed using a novel formulation for clustering, it cannot handle outliers or incorporate spatial information. Thus, the performance of DC-ADMM is the lowest in most of cases. As noise level of regression line increased, the power of ordinary robust mixture regression methods TLE and CTLE decreased, leading to lower RI and ARI score. When the clusters become more and more imbalanced, the RI and ARI scores of all of the baseline methods get much worse. When two clusters have very similar regression parameters, but are far away in terms of spatial locations, TLE and CTLE cannot differentiate the two clusters, as they didn't account for spatial proximity, causing low RI and ARI score.

The second section (6th-7th blocks) of Table 5.1 illustrated the performance of all methods in terms of robustness to outlier contamination, including Type 1 outliers and Type 2 outliers. SRMR is highly robust to both regression outliers and spatial outliers, and the clustering accuracies and parameter estimates are almost unaffected in the presence of outliers. This is because SRMR adopted a trimmed likelihood approach, and it is expected that the outliers will not be taken into model estimations. Since DC-ADMM and ClustGeo are not designed to handle the neither Type 1 or Type 2 outliers, their performance consistently worse than TLE, CTLE, and SRMR. While TLE and CTLE could handle regression outliers, they have no control over the spatial proximity, and hence they are very sensitive Type 2 outliers, i.e., spatial outliers. ACC of TLE and CTLE is around 70% due to spatial heterogeneity while SRMR has 100% accuracy rate in all scenarios.

The third section (8th-10blocks) in Table 5.1 illustrate the performance of all methods for different spatial patterns, regarding the shape, center and density of the spatial clusters. SRMR is designed to detect heterogeneious linear dependencies that is robust to both regression outliers and spatial outliers, and its performance is consistently desirably with regards to different spatial patterns. When the spatial distribution of the clusters are changed from multivariate normal to multivariate uniform, it means the shape of the clusters are less sphear, and more diffused. When the center of spatial coordinate changed from diagonal to horizontal, the boundary of two spatial centers became blurred, meaning there are more overlap between neighbouring clusters. The performance of TLE, CTLE and ClustGeo got

worse with more cluster overlaps, while SRMR is robust to this complex situation thanks to the integration of both regression and spatial similarity. ClustGEO is sensitive to the imbalanced density of different clusters, while SRMR is unaffected.

In summary, SRMR is the only method that could model the linear dependency between response and predictors that vary in the spatial domain, and detect clusters of observations with both similarities in regression parameters and spatial proximity. And it is robust to both outliers in regression fitting and spatial locations. It has produced highly favorable performance in different simulation settings, with regards to different levels of regression/spatial noise, outliers, and mixture imbalance.

5.4 Experiments by using Real-world Data

We further validated SRMR on two real-world datasets, namely (1) a geospatial economics data collected from 298 cities of China and (2) a spatial transcriptomics data collected from 3,798 spatial spots on a 2D breast cancer tissue. The synthetic data based experiments clearly suggested that SRMR is the only method that can effectively solve the spatially dependent mixture regression problem compared to the baseline methods. In the real-world data based experiments, we mainly focused on illustrating the contextual meaning of the spatial regions and corresponding regression models identified by SRMR. We also evaluated the goodness of fitting and significance of the spatially dependent mixture regression models as well as the running time of the tested methods.

5.4.1 Application on Geospatial Economics Data

We collected 7 economic features, namely total GDP, public income, public spend, educational spend, technology spend, population, and averaged personal income, and latitude and longitudes, for 298 cities in China. We evaluated SRMR and baseline methods to this data set. We utilized each of the eight features as a dependent variable and selected others as independent variables When applying SRMR and other regression models, while all the features were utilized as the input of ClustGeo. Similar to the synthetic data based experiments, SRMR is the only method can identify spatially dependent mixture regression

Table 5.1. Synthetic Data Performance

			RI	*ARI*	*ACC*	*PCE*	*RI*	*ARI*	*ACC*	*PCE*	*RI*	*ARI*	*ACC*	*PCE*
Mixture Regression	Component Number		**K=2**				**K=3**				**K=4**			
		DC-ADMM	0.66	0.14	NaN	NaN	0.77	0.15	NaN	NaN	0.79	0.15	NaN	NaN
		TLE	0.67	0.89	0.79	0.03	0.76	0.97	0.75	2.65	0.74	0.94	0.72	2.84
		CTLE	0.88	0.88	0.67	**0.01**	0.93	0.97	0.58	**0.01**	0.94	0.96	0.51	**0.01**
		ClustGeo	0.73	0.95	NaN	NaN	0.75	0.68	NaN	NaN	0.83	0.76	NaN	NaN
		SRMR	**0.92**	**1**	**0.98**	**0.01**	**0.95**	**1**	**1**	**0.01**	**0.95**	**1**	**1**	**0.01**
	Sample Number		**N=100**				**N=200**				**N=400**			
		DC-ADMM	0.66	0.14	NaN	NaN	0.66	0.14	NaN	NaN	0.65	0.15	NaN	NaN
		TLE	0.67	0.88	0.79	0.03	0.67	0.89	0.8	0.02	0.67	0.89	0.79	0.01
		CTLE	0.88	0.89	0.67	0.01	0.88	0.89	0.67	**0.01**	0.88	0.89	0.67	**0**
		ClustGeo	0.53	0.53	NaN	NaN	0.52	0.5	NaN	NaN	0.51	0.51	NaN	NaN
		SRMR	**0.92**	**1**	**0.98**	**0.01**	**0.92**	**1**	**0.99**	**0.01**	**0.92**	**1**	**0.99**	**0**
	Noise Level		**Sigma=0.1**				**Sigma=0.2**				**Sigma=0.5**			
		DC-ADMM	0.66	0.15	NaN	NaN	0.66	0.15	NaN	NaN	0.66	0.15	NaN	NaN
		TLE	0.67	0.88	0.79	0.02	0.65	0.77	0.79	0.08	0.61	0.57	0.79	0.22
		CTLE	0.88	0.89	0.67	**0.01**	0.84	0.79	0.67	**0.02**	0.76	0.6	0.63	0.14
		ClustGeo	0.53	0.51	NaN	NaN	0.74	0.97	NaN	NaN	0.53	0.53	NaN	NaN
		SRMR	**0.92**	**1**	**0.99**	**0.01**	**0.88**	**1**	**0.97**	**0.02**	**0.82**	**1**	**0.94**	**0.07**
	Component Balance		**Ratio=40:40**				**Ratio=50:30**				**Ratio=60:20**			
		DC-ADMM	0.66	0.15	NaN	NaN	0.66	0.17	NaN	NaN	0.65	0.21	NaN	NaN
		TLE	0.67	0.89	0.8	0.03	0.69	0.86	0.82	0.18	0.62	0.72	0.85	0.47
		CTLE	0.88	0.9	0.67	**0.01**	0.88	0.91	0.67	**0.01**	0.85	0.94	0.34	0.37
		ClustGeo	0.52	0.5	NaN	NaN	0.73	0.98	NaN	NaN	0.67	0.91	NaN	NaN
		SRMR	**0.92**	**1**	**0.99**	**0.01**	**0.92**	**1**	**0.99**	**0.01**	**0.91**	**1**	**0.98**	**0.01**
	Coefficient		**Beta=(1.5,1.0)**				**Beta=(1.5,0.1)**				**Beta=(1.5,-1.2)**			
		DC-ADMM	0.66	0.15	NaN	NaN	0.71	0.28	NaN	NaN	0.74	0.25	NaN	NaN
		TLE	0.67	0.89	0.79	0.03	0.69	0.96	0.79	0.03	0.72	0.98	0.79	0.13
		CTLE	0.88	0.89	0.67	**0.01**	0.91	0.96	0.67	**0.01**	0.91	0.98	0.67	**0.01**
		ClustGeo	0.72	0.94	NaN	NaN	0.51	0.56	NaN	NaN	0.5	0.52	NaN	NaN
		SRMR	**0.92**	**1**	**0.99**	**0.01**	**0.94**	**1**	**1**	**0.01**	**0.94**	**1**	**1**	**0.01**
Robustness	Type 1 Outlier Ratio		**Ratio=10%**				**Ratio=20%**							
		DC-ADMM	0.61	0.11	NaN	NaN	0.66	0.14	NaN	NaN				
		TLE	0.64	0.87	0.72	0.05	0.67	0.88	0.79	0.03				
		CTLE	0.86	0.88	0.5	**0.01**	0.88	0.89	0.67	**0.01**				
		ClustGeo	0.55	0.42	NaN	NaN	0.52	0.52	NaN	NaN				
		SRMR	**0.94**	**1**	**0.98**	**0.01**	**0.92**	**1**	**0.99**	**0.01**				
	Type 2 Outlier Ratio		**Ratio=10%**				**Ratio=20%**							
		DC-ADMM	0.73	0.24	NaN	NaN	0.75	0.21	NaN	NaN				
		TLE	0.72	0.98	0.79	0.17	0.7	0.97	0.69	0.17				
		CTLE	0.92	0.98	0.67	**0.01**	0.87	0.98	0.5	**0.01**				
		ClustGeo	0.71	0.9	NaN	NaN	0.46	0.53	NaN	NaN				
		SRMR	**0.94**	**1**	**1**	**0.01**	**0.89**	**1**	**1**	**0.01**				
Spatial Pattern	Spatial Distribution		**Multivariate Normal**				**Multivariate Uniform**							
		DC-ADMM	0.74	0.25	NaN	NaN	0.74	0.24	NaN	NaN				
		TLE	0.72	0.98	0.79	0.09	0.71	0.98	0.78	0.14				
		CTLE	0.92	0.98	0.67	**0.01**	0.92	0.98	0.67	**0.01**				
		ClustGeo	0.7	0.89	NaN	NaN	0.73	0.96	NaN	NaN				
		SRMR	**0.94**	**1**	**1**	**0.01**	**0.94**	**1**	**0.99**	**0.01**				
	Center Location		**Diagonal**				**Horizontal**							
		DC-ADMM	0.66	0.14	NaN	NaN	0.66	0.14	NaN	NaN				
		TLE	0.67	0.89	0.8	0.04	0.67	0.87	0.79	0.03				
		CTLE	0.88	0.89	0.67	**0.01**	**0.88**	0.88	0.67	**0.01**				
		ClustGeo	0.73	0.96	NaN	NaN	0.66	0.78	NaN	NaN				
		SRMR	**0.92**	**1**	**0.99**	**0.01**	0.86	**0.99**	**0.97**	**0.01**				
	Spatial Density		**Balance**				**Imbalance**							
		DC-ADMM	0.66	0.14	NaN	NaN	0.66	0.14	NaN	NaN				
		TLE	0.67	0.89	0.78	0.03	0.67	0.89	0.79	0.04				
		CTLE	0.88	0.9	0.67	**0.01**	0.88	0.9	0.67	**0.01**				
		ClustGeo	0.52	0.51	NaN	NaN	0.5	0.41	NaN	NaN				
		SRMR	**0.92**	**1**	**0.99**	**0.01**	**0.91**	**1**	**0.98**	**0.01**				

models. In contrast, TLE, CTLERob, and DC-ADMM only detected spatial independent regression models, and ClustGeo output a spatial segmentation based on all features.

For a clear visualization and explanation, we illustrated two univariate regressions of *Educational Spend* (ES) $\sim$ GDP and $Income$ $\sim$ GDP. For both ES and $Income$, SRMR identified four spatial regions corresponding to the north-east, middle-east, south-east and west regions of China (Figure 5.2 a1, Figure 5.2 a2). The spatial regions detected by SRMR show distinct different dependency of ES and $Income$ with GDP. Specifically, ES is positively associated with GDP in the middle-east ($ES = 0.24 \cdot GDP + 10.9$) and north-east China ($ES = 0.4 \cdot GDP + 9.17$). The south-east cities have more stable ES, which less depends on GDP ($ES = 0.8 \cdot GDP + 4.19$), while a negative association of ES and GDP are observed in the west cities ($ES = -0.39 \cdot GDP + 17.09$). The high dependency in middle-east and north-east cities and less dependency in south-east cities are consistent to our knowledge, as the middle-east and north-east China are promoting the education system basis while the education systems south-east China are relatively stable. We also checked the cities in the west China that have high GDP but low ES. Such cities include Dongying, Ordos, Karamay, etc., which are developing more neo energy business rather than education in the recent years. Similar observations were also made in the $Income$ $\sim$ GDP model (Figure 5.2a2). The SRMR outputs suggested the personal $Income$ in the north-east, south-east and west cities less depends on GDP while more positive dependency between $Income$ and GDP was observed in middle-east cities, especially the well developed cities Beijing, Shanghai, Tianjin, Hangzhou, etc. On the other hand, on both *Educational Spend* (ES) $\sim$ GDP and $Income$ $\sim$ GDP, TLE and CTLERob failed to identify such spatial dependent and contextual meaningful patterns while both of them tend to over-fit the mixture of regressions (Figure 5.2 a3, Figure 5.2 a4). DC-ADMM identified all cities as one class (Figure 5.2 a5) while ClustGeo identified three distinct non-overlapping spatial regions without offering explainable regional specific feature dependencies (Figure 5.2 a6).

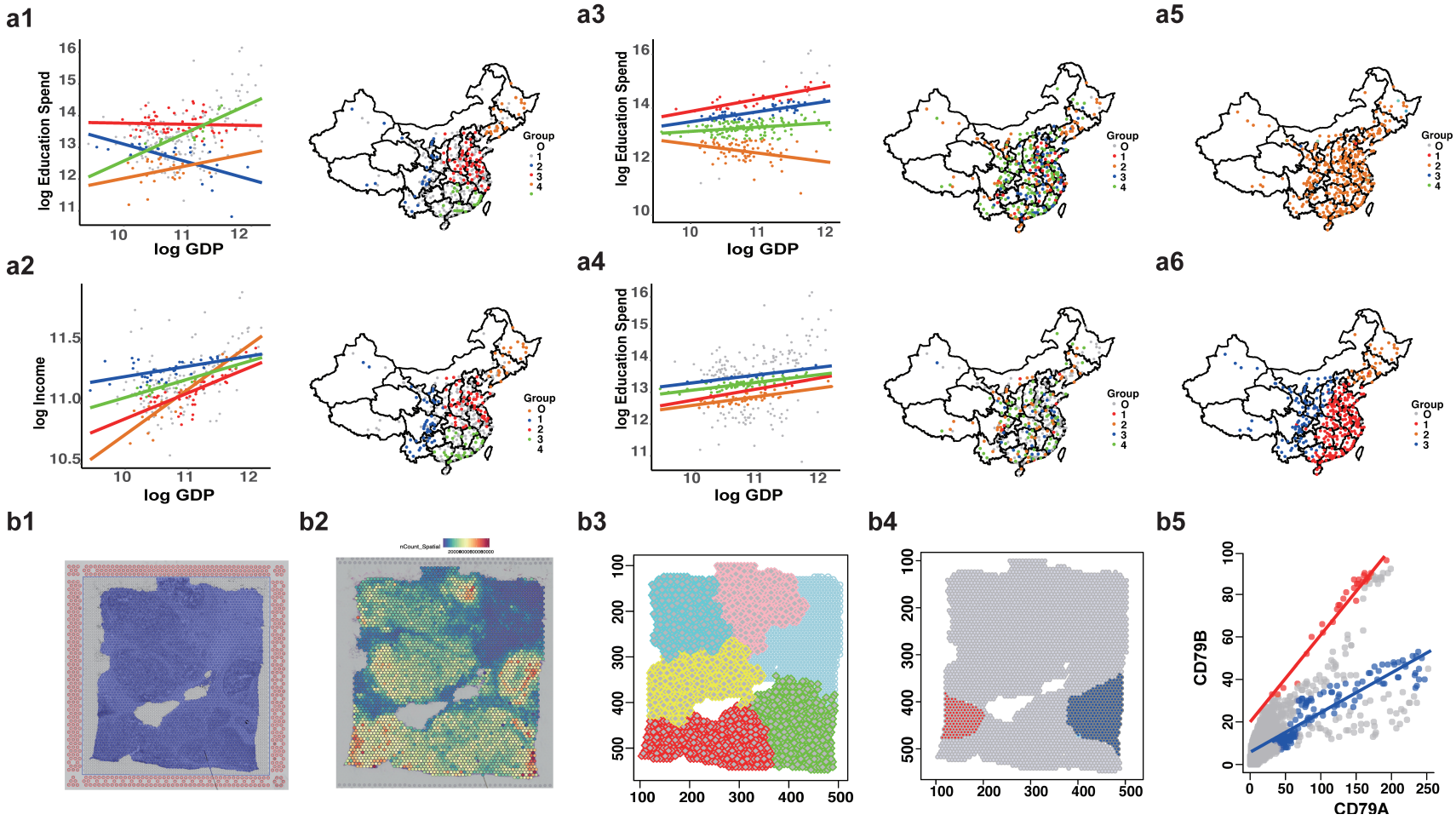


Figure 5.2. Real-world data based experiments. a1: SRMR, a2: SRMR, a3: TLE, a4: CTLE, a5: DC-ADMM, a6: ClustGeo; a1,a3-a6: $ES \sim GDP$, a2: $Income \sim GDP$. a1-a6: cities of different regression components are red, blue, green and orange colored, while the outliers are colored by grey.

5.4.2 Application on Spatial Transcriptomics Data

Here we illustrate that SRMR enables a new type of ST data analysis by simultaneously identify spatial regions in which the expression level of genes show different level of dependency, which directly annotate the biological meaning of each detected region.

We applied SRMR and baseline methods on the v1.1 ST data of breast cancer provided by 10xgenomics.com, consisting of 13,161 genes and 3,798 spatial spots. We first selected 500 genes that having high expression level and having known tumor micro-environment related functions. We fit the regression model $\text{Gene}_1 \sim \text{Gene}_2$ for each pair of the 500 genes by using SRMR, TLE, CTLERob and DC-ADMM and conducted ClustGeo by using all the 500 genes. Similar to the synthetic and Geospatial data, SRMR is the only method that detected spatially dependent mixture regression models in the ST data. General spatial segmentation, such as ClustGeo, identifies spatial regions by using the whole feature matrix (Figure 5.2 b3), which is consistent to the distribution of the averaged gene expression signal level (Figure 5.2 b2). On the other hand, we identified more than 500 overlapped spatial regions by using SRMR, each having varied dependency among certain genes. Figure 5.2 b4 showcased two distinct spatial regions only identified by SRMR, which have varied dependency between the CD79A and CD79B genes as shown in Figure 5.2 b5. CD79A/B are key genes involved in maturation and functional variation of B cells. The varied dependency of CD79A and CD79B characterizes distinct sub-regions in one breast cancer tissue that potentially have different immune activities and responses to immuno-therapy.

In summary, compared with baseline methods, SRMR is the only method can effectively solve the spatially dependent mixture regression problem on the two real-world data. For the analysis of a single regression model in the real-world data, the running time of SRMR, TLE and CTLERob are about 15s, 10s and 2s, respectively. The running time of SRMR is slower, but also comparable to the baseline robust mixture regression approaches. The running time of DC-ADMM and ClustGeo are about 0.01s.

5.5 Conclusions

We developed a new statistical model of high dimensional data with matched spatial information, namely spatially dependent mixture regression. We also developed spatial robust mixture regression (SRMR) analysis as an effective solution of the problem. SRMR is empowered by an inference scheme to assess statistical significance of spatial dependent finite mixture regression models. On both synthetic and real-world data based experiments, we demonstrated that SRMR is the only capability can solve the spatially dependent mixture regression problem. Particularly, SRMR enables a new type of spatial segmentation analysis by detecting large sets of spatial regions having varied dependency among certain features. Compared with conventional spatial segmentation analysis, the regions identified by SRMR characterize more spatial dependent variations conceived in the data and enable better contextual explanation. The source codes of SRMR and the analysis of this study are provided at https://github.com/changwn/SRMR.[9]

[9]↑https://github.com/changwn/SRMR

Part II

Deep Learning based Systems Biology Model for Human Metabolic Flux Estimation

6. A DEEP NEURAL NETWORK MODEL TO ESTIMATE CELL-WISE METABOLIC FLUX USING scRNA-SEQ DATA

This chapter focuses on human metabolic flux estimation with deep learning approach. The metabolic heterogeneity, and metabolic interplay between cells have been known as significant contributors to disease treatment resistance. However, with the lack of a mature high-throughput single cell metabolomics technology, we are yet to establish systematic understanding of the intra-tissue metabolic heterogeneity and cooperative mechanisms. To mitigate this knowledge gap, we developed a novel computational method, namely scFEA (single cell Flux Estimation Analysis), to infer cell-wise fluxome from single cell RNA-sequencing (scRNA-seq) data. scFEA is empowered by a systematically reconstructed human metabolic map as a factor graph, a novel probabilistic model to leverage the flux balance constraints on scRNA-seq data. The intricate information cascade from transcriptome to metabolome was captured using multi-layer neural networks to capitulate the non-linear dependency between enzymatic gene expressions and reaction rates.

In this chapter, we have three major contributions on estimating human metabolic flux at single cell level resolution.

1. We proposed a metabolic map reduction approach based on network topology and gene expression status.

2. We used a multi-layers neural network model to capture the non-linear dependency of metabolic flux on the enzymatic gene expressions.

3. In our method, we built a novel neural network architecture and solution to maximize the overall flux balance of intermediate substrates throughout all cells.

Gene expression patterns over the topological structure of biological networks is deterministic of biological functions. Several studies integrated the topology of biological networks with gene expression data to model activity of transcriptional regulation or metabolic networks, such as using the expression levels over pathways to portray metabolic changes. However, functional activities can vary dramatically from cell to cell due to their high plasticity, while most published work tend to fir one model for predefined cell groups, disregarding the intrinsic heterogeneity among cells. In this work, we proposed an advanced method

to provide a reliable estimation of cell-wise functional activity and states from scRNA-seq data, that could translate single cell transcriptomes to chemical mass carrying fluxomes in each cell. It is critical to capture metabolic activity level at single cell resolution for studying the heterogeneity of metabolic fluxes. However, the single cell resolution metabolomics data is still in immature stage, we cannot observe metabolic activity level for cells. Fortunately, the large availability of scRNA-seq data provides a potential solution of cell-wise metabolic flux from a different perspective, i.e. utilizing the non-linear dependency between gene expression and metabolic flux changes to estimate the metabolic activity level. To handle such non-linear dependency, we proposed a novel graph neural network architecture by using multi-layer neural network to capture the non-linear dependency for each metabolic reaction. In addition, this problem can be also regarded as a one-shot learning problem since the transcriptomes of cells only measure the one moment of cells. Taking traffic flow as an example, we only have one frame of video of traffic flow and we need to estimate traffic flow by only using this one frame picture at different locations of highway. This work is a very first attempt to solve mass carrying flux by using non-real time data, which provided a set of novel mathematical models and algorithms that can be broadly utilized in other real-world flow estimation problems, such as estimation and monitor of traffic and currency flow.

6.1 Background

Metabolic dysregulation is a hallmark of many disease types including cancer, diabetes, cardiovascular disease and Alzheimer's disease [140]–[146]. In cancer, the diseased cells are well understood to rewire their metabolism and energy production to support rapid proliferation, sustain viability, and promote acquired drug resistance [147]–[150]. Here, the diseased cells often react differently to the microenvironmental stress. Such heterogeneity often results in an increased repertoire of possible cellular responses to compromise the efficacy of drug therapies, leading to the enhanced survival of the entire diseased cell population [151], [152]. The metabolome is an excellent indicator of phenotypic heterogeneity due to its high dynamics and plasticity [153]. Unfortunately, current high-throughput metabolic profiling has been largely applied to bulk cell or tissue samples, from which we could only observe an

averaged metabolic signal over a large number of cells; while single cell metabolomics is still in its infancy, due to its relatively low throughput and low sensitivity [153]–[159]. Overall, our understanding of metabolic dysregulation of human disease has been immensely limited by our technology to study the metabolic landscape at single-cell level and in the context of their tissue microenvironment [160]–[167].

Single cell RNA-seq (scRNA-seq) data has been widely utilized to characterize cell type specific transcriptional states and its underlying phenotypic switches in a complex tissue [168]–[177]. Realizing the strong connections between transcriptomic and metabolomic profiles [169], [173], [178]–[181], scRNA-seq data has found its application in portraying metabolic variations. Using scRNA-Seq data, the existing research studied metabolic changes of pre-defined cell groups relying on differential expression and enrichment analysis of key metabolic enzymes and pathways [168], [170]–[175], [177]. However, for this type of analysis, the node/edge structures in a metabolic pathway graph, or the mass balance constraints of metabolic network is not considered. Studies coupling single cell transcriptomics data and the Flux Balance Analysis (FBA) at steady-state framework have only recently emerged [169], [176]. The FBA describes the potential flux over the topological structure of a metabolic network, with a set of equations governing the mass balance at steady state. The advantage of incorporating FBA into the model is two-fold: considering the chemical stoichiometry in FBA could lead to more accurate estimation of the metabolite abundance; flux estimation for each individual metabolite can be solved, leading to high-resolution characterization of the metabolic profiling. Damiani et al developed scFBA that utilizes the cell group specific gene expression status derived from scRNA-seq data to regularize the network topology for FBA [169]. Wagner et al proposed a method, namely Compass, which maximizes the coherence between scRNA-seq expression profile and predicted flux in solution space of FBA [181]. However, as stated in the original works, the stringent flux balance and steady-state assumption in scFBA and Compass may not be rational for certain disease types with constantly severe "imbalance" of many metabolites, namely cancer. Another limitation of the FBA-based methods is that the single cells' gene expression is not used directly to model metabolic flux. Both scFBA and Compass used single cell gene expression as certain constraints to guide the search in the solution space of flux balance condition. In addition, both

models are intended for modeling the fluxes for cells of pre-defined groups, instead of at a single cell resolution, and they are restricted to a small portion of the whole metabolic map. Therefore, it remains an urgent task to design advanced computational tools for reliable estimation of cell-wise metabolic flux and states by translating single cell transcriptomes to single cell fluxomes. Such a tool is vital to unravel the principles of how the disease microenvironment may affect the metabolic phenotypes for the heterogeneous cell types [169], [170].

Computational challenges to estimate cell-wise metabolic flux arise from the following aspects: (1) multiple key factors determine cells' metabolic states, including exogeneous nutrient availability, leading to the discrepancy of cell type specific markers and metabolic phenotypes and states; (2) the whole metabolic network is of high complexity, hence a proper computational reduction and reconstruction of the network is needed to reach a balance between resolution of metabolic state characterization and computational feasibility; (3) the intricate non-linear dependency between transcriptomic expressions and metabolic reaction rates calls for a more sophisticated model to fully capitulate the relationships; and (4) alternative enzymes with similar functions may result in common metabolic phenotypes, however, exactly which enzymes share such common effect to the metabolic flux change remains largely unknown.

In this study, we developed a novel computational method, namely single-cell Flux Estimation Analysis (scFEA), to estimate the relative rate of metabolic flux at single cell resolution from scRNA-seq data. Specifically, scFEA can effectively solve the above challenges with the following computational innovations: (i) an optimization function derived based upon a probabilistic model to consider the flux balance constraints among a large number of single cells with varied metabolic fluxomes, (ii) a metabolic map reduction approach based on network topology and gene expression status, (iii) a multi-layer neural network model to capture the non-linear dependency of metabolic flux on the enzymatic gene expressions, and (iv) a novel graph neural network architecture and solution to maximize the overall flux balance of intermediate substrates throughout all cells. The central hypotheses of scFEA are (1) the flux variations of a metabolic module can be modeled as a non-linear function of the transcriptomic-level changes of the catalyzing enzymes; and (2) the total

flux imbalance of all intermediate substrates should be minimized throughout all single cells. The cell-wise fluxome estimated by scFEA enables a series of downstream analysis, including identification of cell or tissue level metabolic stress, sensitivity evaluation of individual enzymes to the whole metabolic network, and inference of cell-tissue and cell-cell metabolic exchanges. To validate scFEA, we generated an scRNA-seq dataset with matched tissue level metabolomic profiles under different biochemical perturbations. Applications of scFEA on synthetic datasets, the newly generated dataset with matched scRNA-seq and metabolomic profiles, and six other independent real-world datasets, validated the prediction accuracy, robustness, and biological interpretability of scFEA.

6.2 Methods and Materials

6.2.1 Collection of human metabolic map

We consider the human metabolic network as composed of different reaction types including metabolism, transport (including uptake and export), and biosynthesis. As detailed in Results, the reconstructed network consists of 22 super module classes of 169 modules. All reactions related to metabolism were collected from the Kyoto Encyclopedia of Genes and Genomes database (KEGG) [182]. In total, 11 metabolism related super modules were manually summarized, which is comprised of glycolysis, TCA cycle, pentose phosphate, fatty acids metabolism and synthesis, metabolism of amino acids namely serine, aspartate, beta-alanine, glutamate, leucine/valine/isoleucine and urea cycle, propionyl-CoA and spermidine metabolism [183]. The 11 metabolism super modules contain 1388 reactions, 317 enzymes, which corresponds to 563 genes.

Transporters enable the trafficking of molecules in and out of cell membranes. We collected the human transporter proteins, their corresponding genes and metabolite substrates from the Transporter Classification Database [184], [185]. In total, 80 transporter genes, and 35 related metabolites were collected.

An essential part of metabolic map is the biosynthesis pathways. KEGG database and literature [146], [186]–[192] are the main information sources used for building biosynthesis modules. We collected 69 biosynthesis modules forming 10 super modules, namely biosynthe-

sis of hyaluronic acid, glycogen, glycosaminoglycan, N-linked glycan, O-linked glycan, sialic acid, glycan, purine, pyrimidine, and steroid hormones. Overall, the biosynthesis modules include 459 genes of 269 enzymes catalyzing 869 reactions.

6.2.2 Selecting genes of significant expression

We applied our inhouse method, LTMG, to determine the expression status of each gene in each single cell. LTMG considers the multi-modality of the expression profile of each gene throughout all the single cells, by assuming that the gene's expression follows a mixture of suppressed state and activated states, as represented by the following likelihood function [78].

$$\prod_{\mathrm{j}=1}^{N}(\sum_{\mathrm{i}=1}^{S} a_{\mathrm{i}} p_{\mathrm{i}}(x_{\mathrm{j}} \mid u_{\mathrm{i}}, \sigma_{\mathrm{i}}) + a_S + p_{S+1}(x_{\mathrm{j}} \mid u_{S+1}, \sigma_{S+1})) \tag{6.1}$$

,where $x_{\mathrm{j}}, \mathrm{j} = 1, ..., N$ are the expression profile of gene x in N cells, the index $1...S$ are the S active expression states and $S+1$ is the suppressed expression state, a_{i} is the proportion of each state with $a_1 + ... + a_{S+1} = 1$, $a_{1...S} > 0$ and $a_{S+1} \geq 0$, $p_{\mathrm{i}}, u_{\mathrm{i}}$ and σ_{i} are the pdf, mean and standard deviation of each expression state. Specifically, LTMG considers the distribution of ech mixing component, p_{i}, as a left truncated Gaussian distribution, to account for the noise of drop out events. In this work, LTMG was used to fit to each gene's expression and a gene was determined to have significant expression if $\sum_{\mathrm{i}=1}^{S} a_{\mathrm{i}} \geq 0.1$, i.e., the gene has active expression states in at least 10% cells.

6.2.3 Pre-filtering of active modules based on gene expression

Each metabolic module contains an input, an output, and a number of enzymes catalyzing the reactions. A reaction is considered as disconnected if none of the genes encoding its catalyzing enzymes is significantly expressed. A metabolic module is considered as blocked if there is no connected path from the input to the output. Considering the common drop-out events in scRNA-seq data, especially for the drop-seq data, we adopted a conservative approach to pre-trim the metabolic modules: essentially, a module will be removed from

further analysis if none of the genes involved in all reactions of this module has significantly active expressions.

6.2.4 scFEA model setup and a belief propagation based solution of the flux model

Model setup We developed a novel optimization strategy to minimize L similar to the idea of belief propagation [193]. Specifically, the flux balance of each metabolite C_k, $L_K = \sum_{\mathrm{j}=1}^{N}(\sum_{m\in F_{in}^{c_k}} Flux_{m,\mathrm{j}} - \sum_{m'\in F_{out}^{c_k}} Flux_{m',\mathrm{j}})^2$, will be iteratively optimized, by taking into account all the Hop-2 neighbors in the factor graph (metabolites), denoted as $Ne(C_k)$, and Hop-4 neighbors (metabolites), i.e., $Ne^2(C_k) := \{C_{k'} \mid C_{k'} \in Ne(Ne(C_k)) \setminus C_k\}$. Specifically, for a more efficient optimization, we adopt the idea of belief propagation by minimizing a reweighted flux imbalance: $L_K^* = L_K + \sum_{C'_k \in Ne^2(C_k)} W_{k'} L_{k'}$ at each iteration, where $W_{k'}$ is a weight value $(0, 1]$ representing the reliability of the current flux balance of $C_{k'}$. We set $W_{k'} = \exp -\frac{\sum_{C'_k \in Ne()Ne(C_{k'})\setminus\{C_{k'},C_k\}} L_{k'}}{|Ne^2(C_{k'})\setminus\{C_{k'},C_k\}|}$ as an exponential function of the negative averaged imbalance level of 2-hop neighbors (metabolite) of $C_{k'}$ excluding C_k, with higher $W_{k'}$ denoting lower imbalance level of the metabolites. The intuition is that the more reliable the current flux is estimated for the modules involving $C_{k'}$, which is reflected by the averaged imbalance level of its 2-hop neighbors, a higher weight $W_{k'}$ should be given to $C_{k'}$. Therefore, that when minimizing L_K, a disruption of the flux balance of $C_{k'}$ of higher weight will be more heavily penalized, and less desirable.

Neural network model setup For each module, a neural network is used to represent the non-linear dependency between gene expressions and reaction rates. Each neural network has a_1 hidden layers each with a_2 hidden nodes, and one output node. In this study, we took $a_1 = 3$ and $a_2 = 8$. A Hyperbolic Tangent activation function, $Tanhshrink(x) = x - \tanh x$ is used. The number of nodes and the number of hidden layers determines the complexity of network structure, which impacts the convergence time of optimization. Too simple structure may not fully capture the non-linear relationship, while too complex structure cause difficulty to train all parameters and reach convergence. Our organized metabolic modules have an average gene number of 8, which determine the input nodes of scFEA. Since scFEA has

169 parallel subnetworks for each metabolic module, we decide that three hidden layers can leverage the level of non-linearity and overfitting, and ensure a feasible computational cost.

6.2.5 Clustering analysis of cells with distinct metabolic states

scFEA adopts an attributed graph clustering approach to identify the groups of cells and metabolic modules forming a distinct metabolic state. Two clustering approaches were provided to the results of scFEA for different purposes, namely clustering of (1) metabolic modules based on the metabolic map and the predicted flux, and (2) cells sharing a common state on the overall metabolic map based on the predicted flux.

6.2.6 Analysis of cell group specific metabolic stress and metabolic exchanges among cell groups

The cell-wise metabolic flux estimated by scFEA enables the analysis of metabolic stress. For a pre-defined cell group such as cells of the same type, the total imbalance of each compound will be computed and ranked. One-way t-test was applied to test if the imbalance is significantly different from 0. The metabolic exchange among different cell groups from one tissue sample were identified as the metabolites with different signs of metabolic imbalance in different cell groups, such as accumulation and depletion, or exporting or importing. Tissue level metabolic stress is computed as the total imbalance throughout multiple cells.

6.2.7 Perturbation analysis

In scFEA, to evaluate the impact of the change in gene expression on the whole metabolic map, a perturbation analysis is conducted which includes three components: (1) the direct impact of each gene G_{i}^{m} to the flux module m can be directly computed by its derivative $\frac{df_{nn}^{m}}{dG_{\mathrm{i}}^{m}}$ for all the modules containing G_{i}^{m}; (2) the impact of the flux change of one module A on a target module B could be estimated as the variations of flux in B calculated under different values of flux in A, while keeping the other parameters/input fixed, i.e., a Monte-Carlo based method; (3) the impact of each gene's expression to the flux of distant modules

can be evaluated by integrating (1) and (2) using a chain rule, i.e. first computing the flux change of the modules containing the gene and then evaluating the change of other modules.

6.2.8 Patient-derived cell line models of pancreatic cancer

Pa03C cells were obtained from Dr. Anirban Maitra's lab at The Johns Hopkins University (Jones et al. 2008). All cells were maintained at 37°C in 5% CO2 and grown in DMEM (Invitrogen; Carlsbad, CA) with 10% Serum (Hyclone; Logan, UT). Cell line identity was confirmed by DNA fingerprint analysis (IDEXX BioResearch, Columbia, MO) for species and baseline short-tandem repeat analysis testing in February 2017. All cell lines were 100% human, and a nine-marker short tandem repeat analysis is on file. They were also confirmed to be mycoplasma free.

6.2.9 scRNA-seq experiment

Cells were transfected with either Scrambled (SCR) (5′ CCAUGAGGUCAGCAUGGUCUG 3′, 5′ GACCAUGCUGACCUCAUGGAA 3′) or siAPEX1 (5′ GUCUGGUACGACUGGAGUACC 3′, 5′ UACUCCAGUCGUACCAGACCU 3′ siRNA). Briefly, 1×105 cells are plated per well of a 6-well plate and allowed to attach overnight. The next day, Lipofectamine RNAiMAX reagent (Invitrogen, Carlsbad, CA) was used to transfect in the *APEX1* and SCR siRNA at 20 nM following the manufacturer's indicated protocol. Opti-MEM, siRNA, and Lipofectamine was left on the cells for 16 h and then regular DMEM media with 10% Serum was added.

Three days post-transfection, SCR/siAPEX1 cells were collected and loaded into 96-well microfluidic C1 Fluidigm array (Fluidigm, South San Francisco, CA, USA). All chambers were visually assessed and any chamber containing dead or multiple cells was excluded. The SMARTer system (Clontech, Mountain View, CA) was used to generate cDNA from captured single cells. The dscDNA quantity and quality was assessed using an Agilent Bioanalyzer (Agilent Technologies, Santa Clara, CA, USA) with the High Sensitivity DNA Chip. The Purdue Genomics Facility prepared libraries using a Nextera kit (Illumina, San Diego, CA).

Unstrained 2x100 bp reads were sequenced using the HiSeq 2500 on rapid run mode in one lane.

6.2.10 scRNA-seq data processing and analysis

FastQC was applied to evaluate the quality of the single cell RNA sequencing data. Counts were called for each cell sample by using STAR alignment pipeline against human GRCh38 reference genome. Cells with less than 250 or more than 10000 non-zero expressed genes were excluded from the analysis. Cells with more than 15% counts mapped to the mitochondrial genome were excluded as low quality cells, resulting 40 *APEX1* KD and 48 Control cells under hypoxia condition and 27 *APEX1* KD and 46 Control cells under normoxia condition for further analysis. We utilized our in-house left truncated mixture Gaussian model to identify differentially expressed genes [78]. Pathway enrichment analysis of the genes in the identified bi-clusters are computed using hypergeometric test against the 1329 canonical pathway in MSigDB database [133], with $p<0.001$ as a significance cutoff.

6.2.11 Metabolomic profiling and data analysis

To address the function of the mitochondria, S-1 Mitoplates (Biolog, Hayward, CA) Mitochondrial Function Assay were performed following the manufacturer's protocol. The assay covers 14 metabolites in central metabolic pathways, namely glucose, glucose-1 phosphate, glucose-6 phosphate, pyruvate, and lactate in the glycolysis pathway, citrate, 2-oxoglutarate, succinate, fumarate, malate in the TCA cycle, and amino acids glutamate, glutamine, serine, and ornithine. Specifically, assay mix (60 minutes at 37°C) was added to the plates to dissolve the substrates. We collected, counted, resuspended PDAC cells in provided buffer and plated them at 5x104 cells/well after treatment . Readings at 590nm were taken every 5 min for 4 hours at 37°C. Experiments were performed in triplicate with 3 biological replicates for the siAPEX1 and control PDAC cells under the normoxia condition. Raw data was analyzed using Graphpad Prism 8, and statistical significance was determined using the 2-way ANOVA and p-values <0.05 were considered statistically significant

6.2.12 qRT-PCR

qRT-PCR was used to measure the mRNA expression levels of the various genes identified from the scRNA-seq analysis. Following transfection, total RNA was extracted from cells using the Qiagen RNeasy Mini kit (Qiagen, Valencia, CA) according to the manufacturer's instructions. First-strand cDNA was obtained from RNA using random hexamers and MultiScribe reverse transcriptase (Applied Biosystems, Foster City, CA). Quantitative PCR was performed using SYBR Green Real Time PCR master mix (Applied Biosystems, Foster City, CA) in a CFX96 Real Time detection system (Bio-Rad, Hercules, CA). The relative quantitative mRNA level was determined using the comparative Ct method using ribosomal protein L6 (*RPL6*) as the reference gene. Experiments were performed in triplicate for each sample. Statistical analysis performed using the $2-\Delta\Delta$CT method and analysis of covariance (ANCOVA) models, as previously published [194].

6.3 Results

6.3.1 Systems biology considerations, hypotheses, and analysis pipeline of scFEA

The reaction rate of a simple enzyme catalyzed metabolic reaction follows the Michaelis-Menten kinetic model: $V = K_{cat}\frac{[E][S]}{K_m+[S]}$, which is a non-linear function of enzyme concentration $[E]$, substrate concentration $[S]$, and kinetic parameters K_{cat} and K_m. . On one hand, the reaction rate is approximately a linear function of the enzyme concentration when the substrate concentration is much larger than K_m, i.e., when $\frac{[S]}{K_m+[S]}$ is ~ 1; on the other hand, the enzyme concentrations could often serve as a surrogate for the substrate concentration considering the regulatory effect of substrate availability on the enzyme transcription. Overall, we consider the reaction rate to be an (non-)linear function of the enzyme concentration. Obviously, the flux of a reaction chain is mostly determined by the rate limiting steps, which depend on the flux distribution, substrate concentration, and kinetic parameters. Hence, the rate limiting steps are often context specific and unknow because of the dynamics of the physiological and biochemical conditions of the cells. Based on these considerations, we developed scFEA, to estimate cell-wise metabolic flux from scRNA-seq data. scFEA consists of

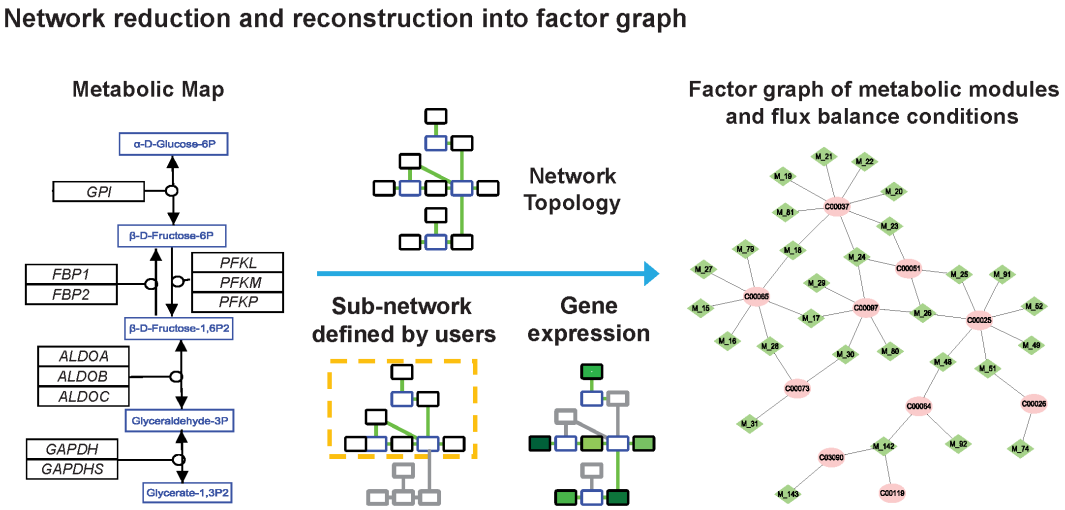


Figure 6.1. Metabolic reduction and reconstruction. A metabolic map was reduced and reconstructed into a factor graph based on network topology, significantly non-zero gene expressions and users' input.

three major computational components, namely: (1) network reduction and reconstruction, (2) estimation of cell-wise metabolic flux, and (3) downstream analyses including estimation of metabolic stress, perturbation of metabolic genes, and clustering of cells with different metabolic states (Figure 6.3). The required input of scFEA is an scRNA-seq dataset, while optional inputs, including cell group labels or subset of metabolic reactions of interest, can be specified for additional analysis.

To reduce the complexity of the metabolic map, we reconstructed it into a factor graph composed by connected metabolic modules as variables and intermediate metabolites as factors (Figure 6.1). Specifically, connected reactions are merged into one module, if changes in the reaction rates within the module do not affect the rates of the rest of the reactions, given a fixed flux rate of the module. In other words, the estimated flux of a module stays the same with or without merging the reactions, under the flux balance condition. This approach increases the robustness of flux estimation and reduces the computational complexity.

The central computational component of scFEA is a novel graph neural network architecture, which models cell-wise metabolic flux of each metabolic module using gene expression levels of the catalyzing enzymes (Figure 6.2). We hypothesize that the metabolic flux throughout all the single cells in a tissue sample should minimize the overall imbalance of

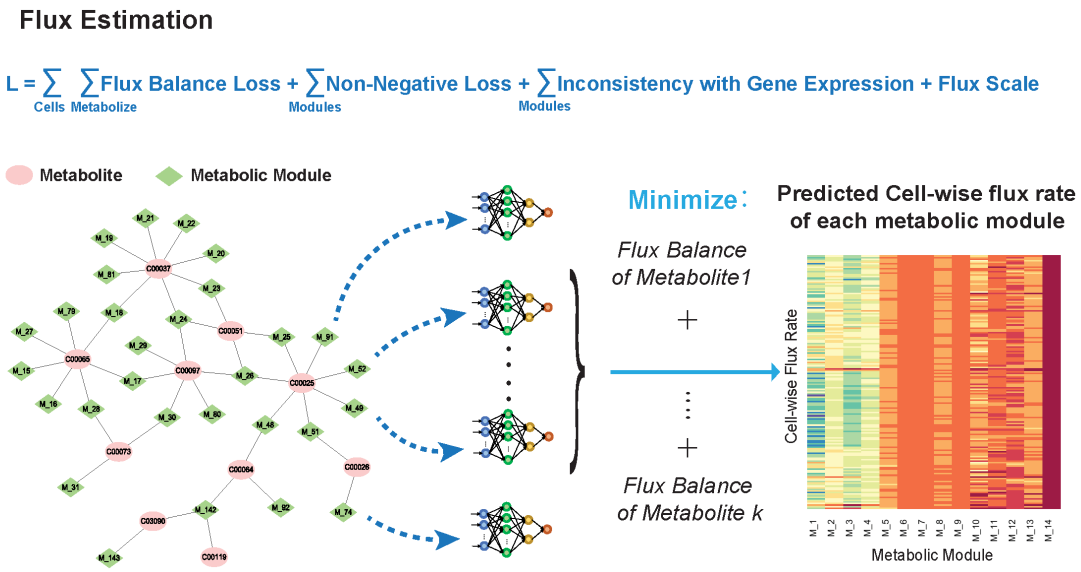


Figure 6.2. A novel graph neural network architecture based prediction of cell-wise fluxome. A loss function (L) composed by loss terms of flux balance, non-negative flux, coherence between predicted flux and gene expression, and constraint of flux scale, were utilized to estimate cell-wise metabolic flux from scRNA-seq data.

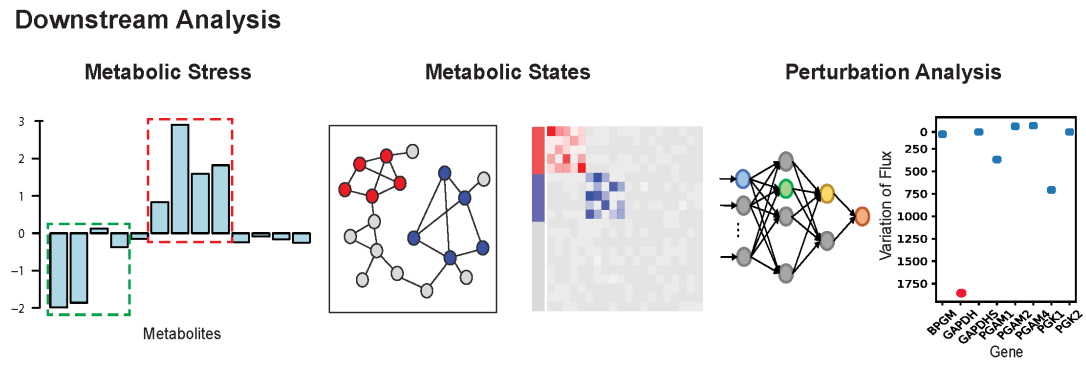


Figure 6.3. Downstream analysis of scFEA is provided, including inference of metabolic stress, cell and module clusters of distinct metabolic states, and the genes of top impact to the whole metabolic flux.

the in-/out-flux of intermediate substrates. The rationality of this assumption is that cells within the same tissue exchange metabolites with each other, hence the total flux balance constraint on all the single cells from one tissue sample is more robust than in individual cells. In scFEA, we utilize the gene expression variations to reflect the protein level change of enzymes and transporters. Note that this assumption is supported by many existing studies that reveal the high explainability of the transcriptome for the proteome [165], [195], [196]. We assume the flux variations of a module generally impacts its neighboring modules, which can be reflected by aggregating the expression variations of the genes in its neighborhood over the metabolic network. The non-linear dependency between gene expression and metabolic flux is modeled as a fully connected neural network of 2-4 layers, which could be considered as a non-linear approximation of the Michaelis-Menten model. To solve the neural network parameters, scFEA minimizes a loss function that mimics the overall flux imbalance of all modules in all cells, with further non-negativity and other prior assumptions on the module fluxome. The large number of single-cell in an scRNA-seq data grants sufficient statistical power to detect the flux variations and avoids the overfitting of the neural network training. It is noteworthy the parameters of the neural network could serve as sensitivity measures of the metabolic flux balance to the variations of the genes. In other words, genes with higher impact are likely to be associated with rate limiting reactions under the particular context.

The estimated cell-wise metabolic flux enables the prediction of (i) the metabolites or pathways with high imbalance in certain cell group, (ii) groups of metabolic modules or cells with varied metabolic states, and (iii) the metabolic genes whose perturbation highly impacts the overall metabolic flux (Figure 6.3). In this study, we mainly focus on solving cell-wise metabolic flux and states, and method validations in human cells. A capability for mouse data analysis is also provided in the software package of "scFEA".

6.3.2 Metabolic map reduction and reconstruction

The whole metabolic network in human and mouse have been well studied. However, while databases including the Kyoto Encyclopedia of Genes and Genomes (KEGG) provide well categorized metabolic pathways and the comprehensive set of metabolic genes [182], the

network topological structure needs to be further optimized for fluxome estimation, due to the following reasons: (1) the flux balance constraints depend on the optimization goal or computational assumption, such as the balance of carbon, redox or pH, (2) the network complexity needs to be reduced to enable computational feasibility, and (3) a manual correction and annotation of the directions of reactions and transporters is in need. In addition, cells of different types or physiological states naturally have varied metabolic states. In scFEA, we first manually curated and annotated the metabolic map of human and mouse retrieved from KEGG database. The global metabolic map is further reduced and reconstructed into a factor graph based on its topological property. scFEA also allows the selection of a connected sub-network in the global metabolic network for flux estimation.

Collection of human and mouse metabolic map. The metabolic map consists of pathways and reactions that fall under four major types, namely import, metabolism, biosynthesis, and export. To ensure a comprehensive coverage of the global metabolic map, we collected reactions of metabolism and biosynthesis as well as transporters for import and export from different data sources. Specifically, metabolic reactions were directly retrieved from KEGG database [182]; the transporters and annotations of import and export reactions were accessed from the transporter classification database [197]; biosynthesis reactions were collected from the biosynthesis pathways encoded in KEGG and curated by using additional literatures. The final metabolic map covers the metabolism, transport, and biosynthesis of carbohydrate, amino acids, fatty acids and lipids, glycan, and nucleic acids in human and mouse, including 862 genes of 390 enzymes, 1880 reactions, 1219 metabolites, and 116 transporter genes of 35 metabolites in human.

6.3.3 Mathematical formulation of metabolic flux estimation in individual cells

For a clear model setup, we first formulate the metabolic network as a directed factor graph. Here, each metabolic module is represented as a variable, and each compound is represented as a factor node carrying a loss function that evaluates the level of flux imbalance among modules, and the direction represent if a metabolite is the input or output of a metabolic module (Figure 6.4). We denote $FG(C^{1\times K}, RM^{1\times M}, E = \{E_{C\to R}, E_{R\to}\})$ as the

factor graph, where $C^{1\times K} = \{C_k, k = 1, ..., K\}$ is the set of K compounds, $RM^{1\times M} = \{R_m, m = 1, ..., M\}$ is the set of M metabolic modules, $E_{R\to C}$ and $E_{C\to R}$ represent direct edges from module to compound and from compound to module, respectively. For the k-th compound C_k, we define the set of reactions producing and consuming C_k as $F_{\text{in}}^{C_k} = \{R_m \mid (R_m \to C_k) \in E_{R\to C}\}$ and C_k as $F_{out}^{C_k} = \{R_m \mid (C_k \to R_m) \in E_{C\to R}\}$, which is derived from the stoichiometric matrix of the whole metabolic map. For an scRNA-seq data set with N cells, we denote $Flux_{m,\text{j}}$ as the flux of the m-th module in the cell $\text{j}, \text{j} = 1, ..., N$ and let $F_{\text{j}} = \{Flux_{1,\text{j}}, ..., Flux_{M,\text{j}}\}$. Our computational hypothesis is that the total flux imbalance of the intermediate metabolites throughout all the collected cells should be minimized, based on which we developed the likelihood function of the flux of all modules throughout all cells as:

$$\phi(C, F) = \prod_{\text{j}=1}^{N} \prod_{k=1}^{K} \phi(C_{k,\text{j}} \mid F_{\text{j}})\varphi(F_{\text{j}}) \tag{6.2}$$

, where $\phi(C_{k,\text{j}} \mid F_{\text{j}}) = \phi(C_{k,\text{j}} \mid F_{\text{in}}^{C_k}, F_{out}^{C_k}) \propto \text{e}^{-\frac{\lambda(\sum_{m\in F_{\text{in}}^{C_k}} Flux_{m,\text{j}} - \sum_{m\in F_{out}^{C_k}} Flux_{m,\text{j}})^2}{2}}$ and $\varphi(F_{\text{j}})$ represents the prior distribution of the fluxome in cell j, and λ is a tuning hyperparameter. scFEA models the flux of reach reaction, $Flux_{m,\text{j}}$, as a nonlinear function of the expression changes of the genes associated with the module. Denote $G^m = \{G_1^m, ..., G_{\text{i}_m}^M\}$ as the genes associated with the reactions in R_m, and $G^m = \{G_1^m, ..., G_{\text{i}_m}^M\}$ as their expressions in sample j, where i_m stands for the number of genes in R_m. WE model $Flux_{m,\text{j}} = f_{nn}^m(G_{\text{j}}^m \mid \theta_m)$ as a multi-layer fully connected nural network with the input G_{j}^m, where θ_m denotes the parameters of the neural network (Figure 6.6). It is noteworthy that the cell group and tissue context specific distribution of the flux $\varphi(F_{\text{j}})$ and the reaction parameters θ_m are always unknown. Apparently, without further constraints, $Flux_{m,\text{j}} = 0$ is a trivial solution. To provide a robust and rational solution, we introduced two additional constraints to $Flux_{m,\text{j}}$ namely (1) the predicted flux, $Flux_{m,\text{j}}$, should be non-negative; and (2) within a super module (Figure 6.5), the total predicted flux should be correlated with gene expression variation. The second assumption assumes that the metabolic flux variation within large metabolic modules should be coherent to their gene expression change, which is supported by recent studies [169], [181]. This assumption effectively avoids the trivial solution. Hence, instead

of directly maximize $\phi(C,\ F)$, we solve the θ_m and cell-wise flux $Flux_{m,\mathrm{j}}$ by minimizing the following loss function L:

$$\begin{aligned} L = \sum_{\mathrm{j}=1}^{N}\sum_{k=1}^{K}(\sum_{m\in F_{in}^{C_k}} - \sum_{m'\in F_{out}^{C_k}})^2 + \alpha\sum_{\mathrm{j}=1}^{N}\sum_{m=1}^{M}(\mid Flux_{m,\mathrm{j}} - Flux_{m,\mathrm{j}}) \\ +\beta\sum_{\mathrm{j}=1}^{N}[1- \mid cor(Flux_{:,\mathrm{j}}^{SM}, GE_{:,\mathrm{j}}^{SM}) \mid] + \gamma\sum_{\mathrm{j}=1}^{N}(\sum_{m=1}^{M} \mid Flux_{m,\mathrm{j}} \mid -TA_{\mathrm{j}})^2 \end{aligned} \tag{6.3}$$

where $\alpha,\ \beta,\ \gamma$ are hyperparameters, *cor* represents Pearson's correlation coefficients; $Flux_{\square}^{SM}$ and$GE_{\square}^{SM}$ are two $NSM \times N$ matrices, here NSM is number of super modules, $Flux_{m,\mathrm{j}}^{SM}$ represents the sum of the flux of the modules in the super module m, $GE_{m,\mathrm{j}}^{SM}$ represents the sum of expression of the genes in the super module m, in cell j, and TA_{j} is a surrogate for total metabolic activity level of cell j, which is assigned as the total expression of metabolic genes in cell j. The first, second, third and fourth terms of L are related to constraints on flux balance, non-negative flux, the coherence between predicted flux and total gene expression level of each super-module, and the relative scale of flux, respectively. Here Pearson's correlation, which is scale-free, is utilized to model the coherence between gene expression and predicted flux, as genes may have varied intrinsic expression range. Our empirical and robustness analyses suggested that $\alpha = 1, \beta = 0, \gamma = 1$ and $\alpha = 1, \beta = 1, \gamma = 1$ result in a good leverage of the flux balance loss and other constraints for Smart-seq2 and 10x Genomics data, respectively.

It is noteworthy that the above formulation defines a new graph neural network architecture for flux estimation over a factor graph: on one hand, each variable is defined as a neural network of biologically meaningful attributes, i.e., the genes participating in each metabolic module; on the other hand, the information aggregation between adjacent variables is constrained by the balance of the in- and out- flux of each intermediate metabolites. Noted, the number of intermediate constraints (K) and large number of cells (N) of scRNA-seq data ensures the identifiability of θ_m for the multi-layer f_{nn}^{m} at at a certain complexity level.

The challenges to minimize the loss function L include the following: (1) the balance of one intermediate substrate is influenced by multiple modules, hence updating the module flux one at a time may not be computationally efficient, and (2) the updating strategy for

a large group of fluxes cannot be theoretically derived. The two challenges prohibit a direct utilization of back propagation or gradient descending methods. We developed an effective optimization strategy for L by adopting the idea of information transfer in belief propagation, which has been commonly utilized in analyzing cyclic networks such as Markov random field [198]. Specifically, L is minimized by iteratively minimizing the flux imbalance of C_k and the weighted sum of the flux imbalance of the Hop-2 neighbors of C_k in the factor graph, as the L_k^* defined below:

$$L_k^* = \sum_{\mathrm{j}=1}^{N}\Big(\sum_{m\in F_{in}^{C_k}} - \sum_{m'\in F_{out}^{C_k}}\Big)^2 + \sum_{k'} W_{k'} \sum_{\mathrm{j}=1}^{N}\Big(\sum_{m\in F_{in}^{C_k}} - \sum_{m'\in F_{out}^{C_k}}\Big)^2 \tag{6.4}$$

where $C_{k'}$ are the Hop-2 neighbors of C_k, $W_{k'}$ is proportional to the current total imbalance of all the Hop-2 neighbors of $C_{k'}$, except for C_k itself. Here the Hop-2 neighbors of a compound (or module) on the factor graph is defined as all other compounds (or modules) having a connection with the modules (or compounds) who connect to the compound (or module). Such a regional perturbation strategy over the whole graph can effectively leverage the search of global minimum and computational feasibility.

The output of scFEA includes f_{nn}^m, θ_m for each module and predicted cell-wise metabolic flux $Flux_{m,\mathrm{j}}$. It is noteworthy the predicted flux $Flux_{m,\mathrm{j}}$ is a relative measure of unfixed scale. However, $Flux_{m,\mathrm{j}}$ is comparable among cells ($Flux_{m,:}$) or metabolic modules ($Flux_{:,\mathrm{j}}$).

6.3.4 Method validation on a scRNA-seq data with perturbed metabolic conditions and matched metabolomics data

To validate the cell-wise flux estimated by scFEA, we generated an scRNA-seq dataset consisting of 162 patient-derived pancreatic cancer cells (Pa03c cell) under two crossed experimental conditions: *APEX1* knockdown (*APEX1* KD) or control, and under hypoxia or normoxia conditions (see detailed experimental procedure and data processing in Methods). Metabolomics profiling of 14 metabolites were collected on bulk wildtype Pa03c cells and *APEX1* inhibition cells under the normoxia conditions, each with three replicates. The 14 metabolites include glucose, glucose-1 phosphate, glucose-6 phosphate, pyruvate, and lac-

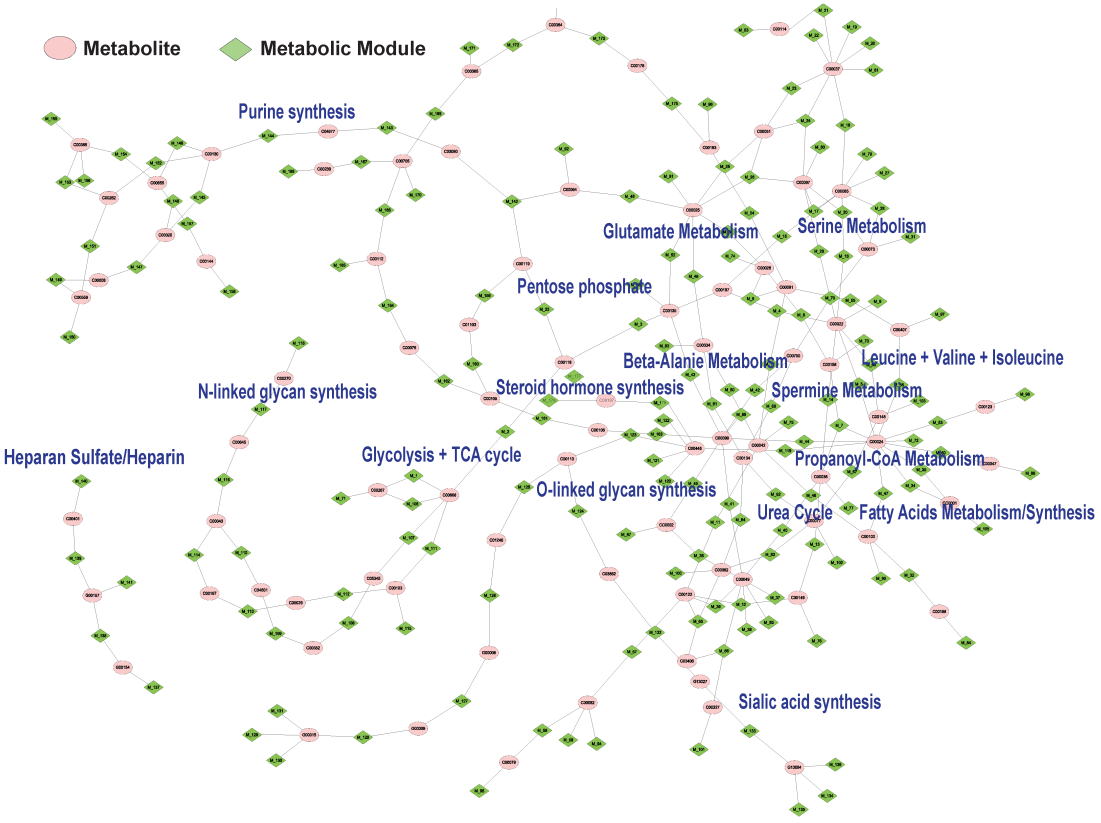


Figure 6.4. Factor graph representation of the reconstructed human metabolic map, in which the modules and metabolites were colored by green and pick.

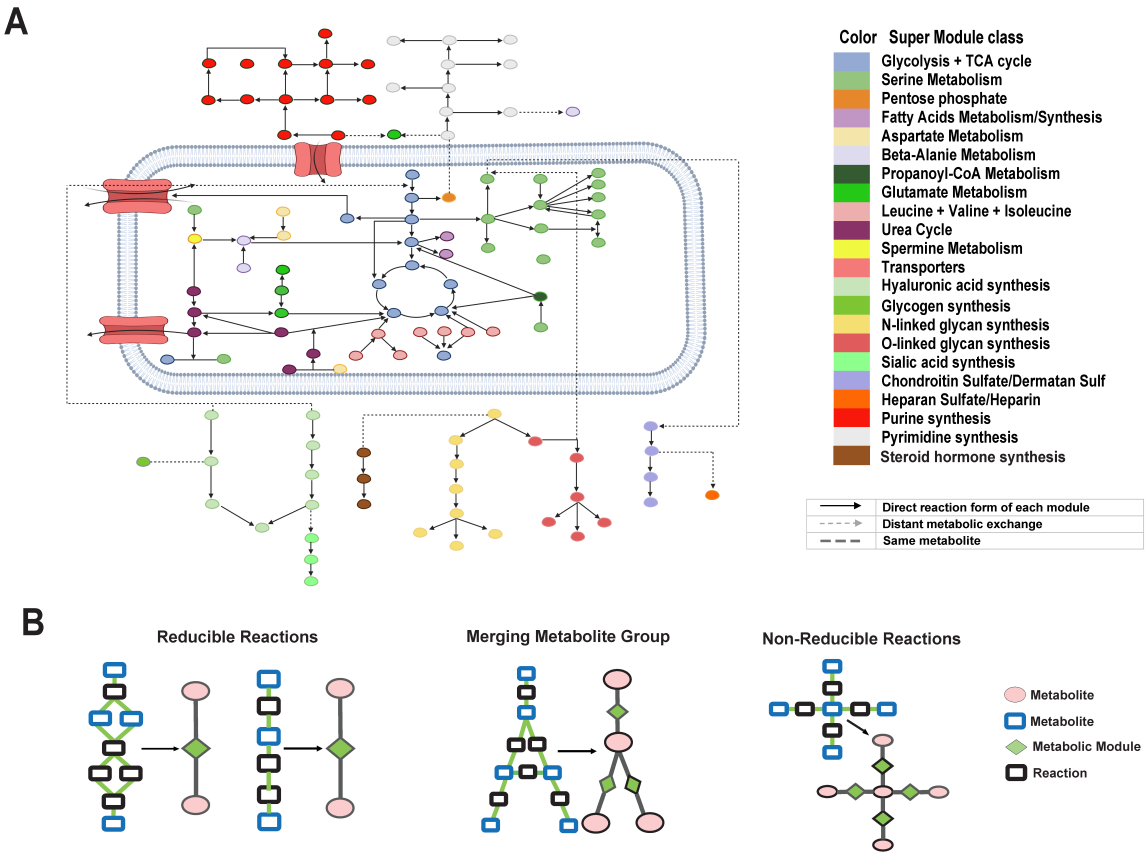


Figure 6.5. Reduced and reconstructed human metabolic map. (A) Collected human metabolic modules and super module classes. (B) Examples of how the network motifs in the metabolic map are simplified into metabolic modules, where the reactions and metabolites are represented by black and blue rectangles, and modules and metabolites are colored by green and pink. Chain-like reactions can be directly simplified; a complicate module connected by multiple branches can be shrunk into one point linked with the multiple in/out branches; and complicated intersections cannot be simplified.

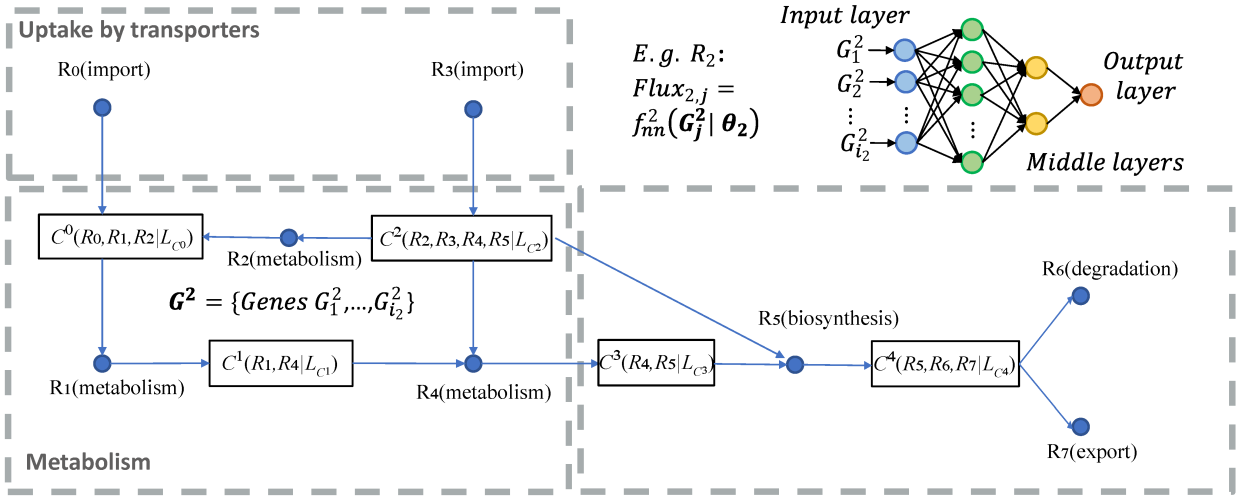


Figure 6.6. A toy model of the factor graph of metabolic modules, flux balance conditions, and the flux model for the module R_2 (top-right). In the factor graph, each C (metabolites) corresponds to one flux balance condition and serves as a factor, and each R (can be a reaction or a module) is a variable. For example, $C^0(R_0, R_1, R_2 \mid L_{C^0})$ simplyrepresents that the metabolite C^0 is determined by the flux balance loss of R_0, R_1, R_2, here L_{C^0} is the flux balance term of C^0. Import and export/degradation reactions are considered as habing no input or output substrates.

tate in the glycolysis pathway, citrate, 2-oxoglutarate, succinate, fumarate, malate in the TCA cycle, and amino acids glutamate, glutamine, serine, and ornithine. We utilized the Smart-seq2-fluidigm protocol for single cell RNA sequencing. It allows for saturated gene detection of each single cell, to enable a more accurate modeling of metabolic flux. *APEX1* is a multifunctional protein that interacts with multiple transcriptional factors (TFs) to regulate cellular responses to hypoxia and oxidative stress [199]. Our previous studies identified significant roles of APEX1 in the regulation of Pa03c cells' response to metabolic environment changes [78], [200].

To the best of our knowledge, scFEA is the first computational tool to estimate metabolic flux at single cell level. Without baseline methods for comparisons, we validate scFEA by examining the consistency between the metabolic flux variation predicted by scFEA and experimental observations. We identified 126 up- and 443 down- regulated genes in *APEX1* KD vs Control under the normoxia condition, and 260 up- and 1496 down- regulated genes under hypoxia condition. Pathway enrichment analysis showed that the TCA cycle (normoxia: p=0.003, hypoxia: p=1.12e-07) and oxidative phosphorylation (normoxia: p=3.17e-4, hypoxia: p=1.77e-08) pathways are significantly enriched by down regulated genes, under both normoxia and hypoxia conditions. This suggests that the knock down of *APEX1* may lead to inhibited cellular aerobic respiration. In addition, genes regulated by *HIF1A* (hypoxia-inducible factor 1-alpha), including glycolysis and TCA cycle genes, were observed to be up-regulated in hypoxia conditions compared with normoxia conditions, in the control Pa03c cells. This is consistent to the common knowledge of hypoxia response. Out of the 14 metabolites, we have seen increase of abundance in glucose, glucose-1 phosphate, glucose-6 phosphate, and lactate, and decrease in 2-oxoglutarate, succinate, fumarate, and malate in *APEX1*-KD vs control cells under the normoxia condition. In summary, analysis of the single cell gene expression and bulk cell metabolomic data revealed that knockdown of *APEX1* affects the cells' glucose metabolism and inhibits the cells' TCA cycle pathway, under both normoxia and hypoxia condition. Figure 6.7 illustrates the variation of genes and metabolites involved in glycolysis, pentose phosphorylation, TCA cycle, glutaminolysis and aspartate metabolism pathways in *APEX1*-KD vs control under normoxia condition. We

conducted a qRT-PCR experiment to confirm the down regulated genes in glycolysis, TCA cycle and oxidative phosphorylation pathways (Figure 6.8).

Consistency between the scFEA predicted flux variation and the metabolomics data.

We applied scFEA to the aforementioned scRNA-seq data of the four conditions, with hyperparameters $\alpha = 1, \beta = 0, \gamma = 1$. We first focus on the normoxia conditions where matched single cell expression and metabolomics data are available. scFEA predicted decreased metabolic flux for the modules in glycolysis and TCA cycle in *APEX1*-KD vs control, i.e., glucose → D-Glucose 1-phosphate (G1P) → alpha-D-Glucose 6-phosphate (G6P) → glyceraldhyde-3P (G3P) → 3-Phospho-D-glyceroyl phosphate (3PD) → pyruvate → Acetyl-CoA → citrate → 2-Oxoglutarate (2OG) → succinate-CoA → succinate → fumarate → malate → oxaloacetate (OAA) and pyruvate → lactate. Particularly, the reactions towards the downstream of this reaction chain has even lower flux in *APEX1*-KD vs control (Figure 6.9A). We then examined the Pearson's correlation between the averaged predicted flux change with the observed metabolomic change of intermediate metabolites in glycolysis and TCA cycle pathways. In APEX1-KD vs control cells under normoxia condition, we observed a Pearson's correlation coefficient (PCC) of 0.86 (p=0.006) (Figure 6.9A), suggesting the high consistency between predicted flux variation with the observed metabolic changes. Using metabolomics data, we observed increase of production for glucose, G1P, G3P and lactate, and decrease of production for 2OG, succinate, fumarate, and malate in *APEX1*-KD vs control (Figure 6.9B). By Michaelis Menten model, the substrates of largely varied concentration determine the reaction rate in a non-linear manner (close to linear when the reaction is less saturated). Hence, variations in the concentration of the metabolites with one dominating out-flux could partially reflect the changes of the out-flux rate. We also correlated the metabolomic change with the averaged expression change of the enzymes catalyzing the reactions. However, no significant correlation was observed (PCC=-0.03, p=0.943, Figure 6.9B), suggesting that single cell gene expression alone doesn't produce a good estimate of single cell metabolomic landscape. In addition, ssGSEA (single sample gene set enrichment analysis) has been utilized to model cell-wise pathway activity in scRNA-seq data [201]. Here, we showed that scFEA predicted metabolic flux is much more consistent to the true

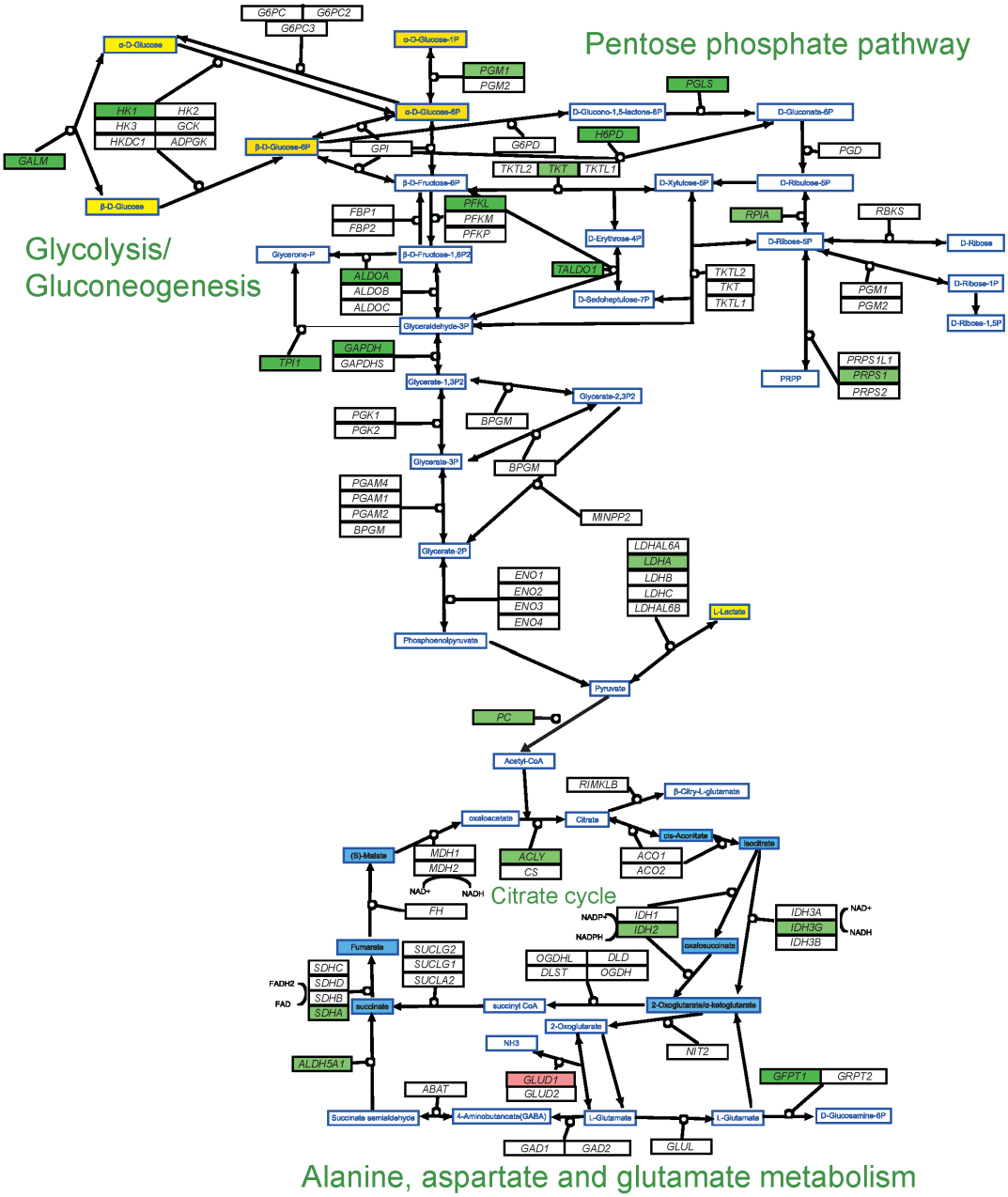
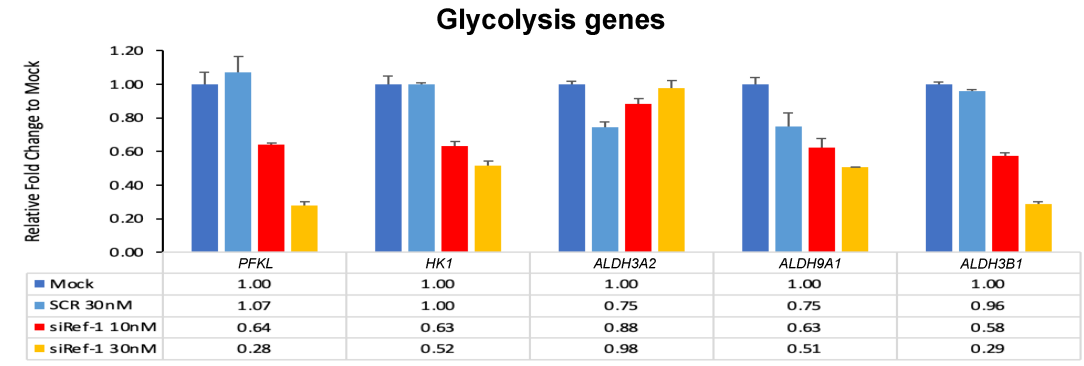
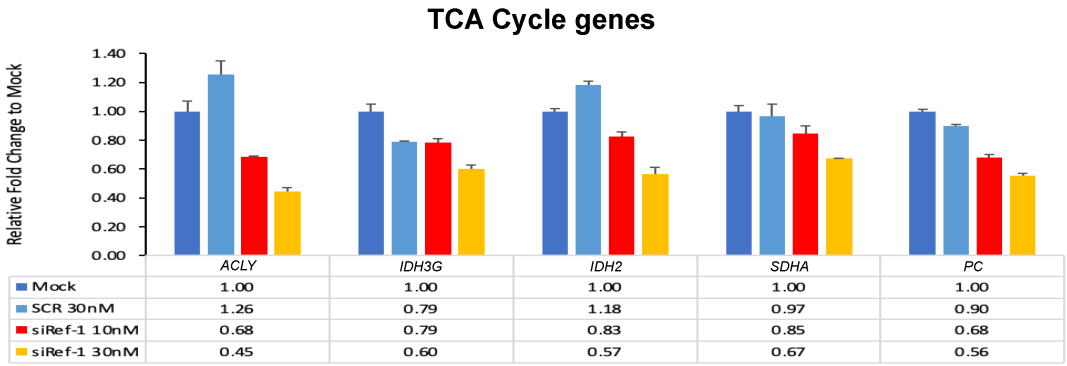


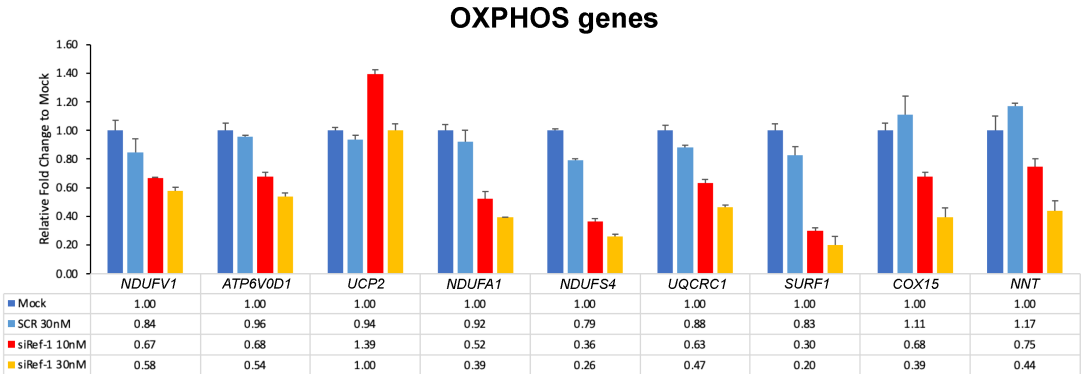
Figure 6.7. Gene expression and metabolomic variations of the glycolysis, pentose phosphate, TCA cycle, glutamine, and aspartate metabolic pathways in *APEX1*-KD vs control under normoxia condition. Genes/metabolites were shown in rectangular boxes with black/blue borders, up/down regulated genes were colored in red/green, increased and decreased metabolites were colored in yellow/blue, respectively. The darker color suggests a higher variation.



	PFKL	HK1	ALDH3A2	ALDH9A1	ALDH3B1
Mock	1.00	1.00	1.00	1.00	1.00
SCR 30nM	1.07	1.00	0.75	0.75	0.96
siRef-1 10nM	0.64	0.63	0.88	0.63	0.58
siRef-1 30nM	0.28	0.52	0.98	0.51	0.29



	ACLY	IDH3G	IDH2	SDHA	PC
Mock	1.00	1.00	1.00	1.00	1.00
SCR 30nM	1.26	0.79	1.18	0.97	0.90
siRef-1 10nM	0.68	0.79	0.83	0.85	0.68
siRef-1 30nM	0.45	0.60	0.57	0.67	0.56



	NDUFV1	ATP6V0D1	UCP2	NDUFA1	NDUFS4	UQCRC1	SURF1	COX15	NNT
Mock	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
SCR 30nM	0.84	0.96	0.94	0.92	0.79	0.88	0.83	1.11	1.17
siRef-1 10nM	0.67	0.68	1.39	0.52	0.36	0.63	0.30	0.68	0.75
siRef-1 30nM	0.58	0.54	1.00	0.39	0.26	0.47	0.20	0.39	0.44

Figure 6.8. qRT-PCR results. Mock and SCR are controls and siRef-1 are knock down of *APEX1*.

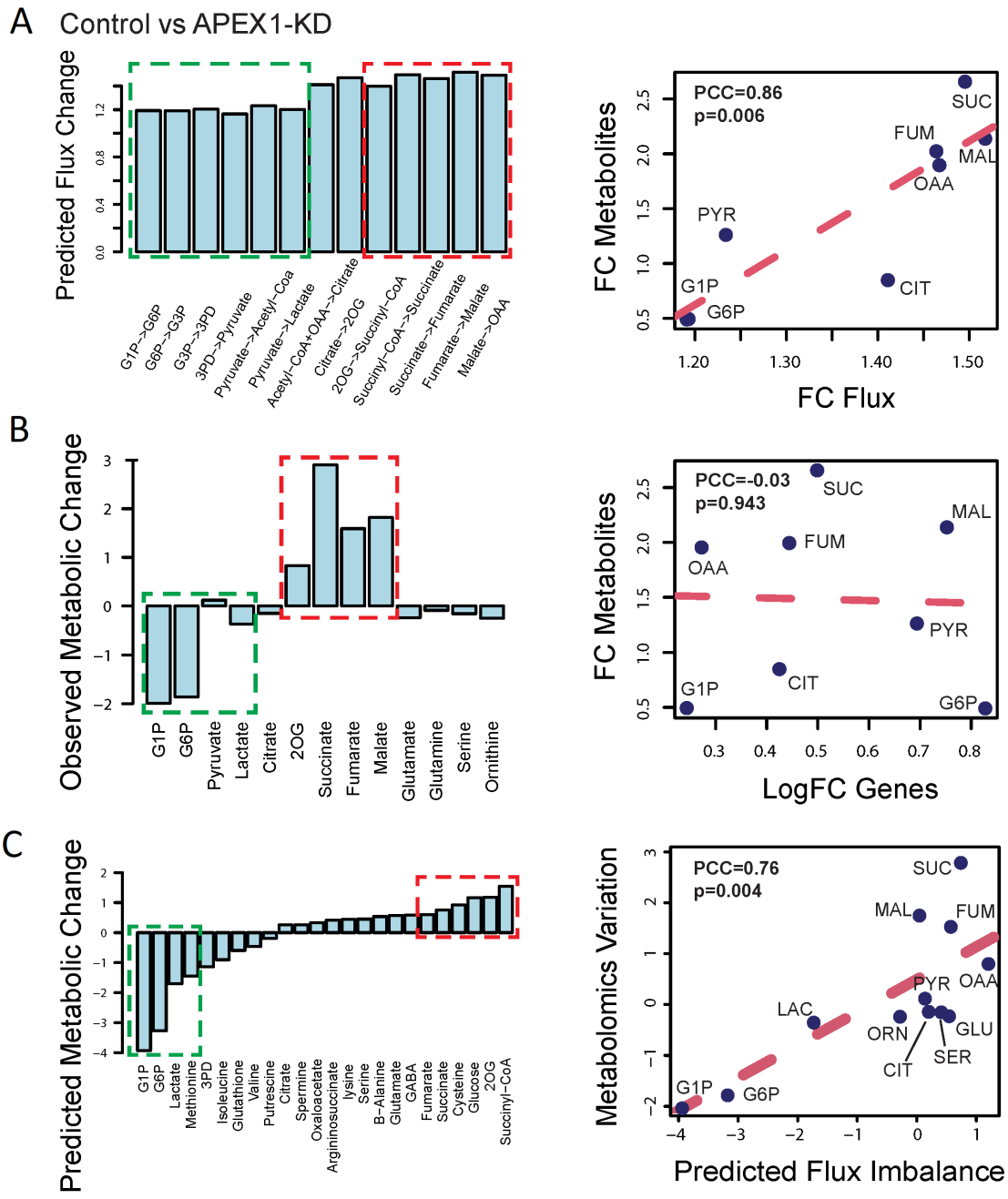
metabolomics changes, as it leveraged the non-linear relationships between gene expression and enzymatic reaction rate, and the flux balance constraints of the metabolites.

High consistency of the predicted metabolic stress with experimentally observed metabolomic changes.

scFEA allows us to investigate the cell-wise metabolic stress, which was defined as the imbalance of the in-/out-fluxes of each intermediate metabolites in each cell. Figure 6.9C shows that the G1P, G6P and lactate were accumulating while 2OG, succinate, succinyl-CoA, and fumarate were depleted in *APEX1*-KD vs control. A PCC of 0.75 (p=0.004) was observed between the predicted metabolic stress and the true metabolic change, on 12 metabolites with both measured metabolomic profile and predicted metabolic stress. This demonstrates the high accuracy of the predicted and observed metabolic stress level. Figure 6.10A shows the predicted cell-wise fluxome of the glycolysis and TCA cycle modules for cells of the four conditions. We observed, in general, higher flux of the glycolytic modules than the TCA cycle modules, with the largest average flux gap seen on Pyruvate → Acetyl-CoA and Acetyl-CoA → Citrate. In addition, the flux of the downstream reactions (citrate → 2OG → succinyl-CoA → succinate) of the TCA cycle is lower than the upstream reactions (succinate → fumarate → malate → OAA). A possible explanation for the leaky metabolic flux is that some of the intermediate substrates flow to other branches, majorly for biosynthesis of amino acids. Among the four conditions, we identified that the hypoxia control group has the highest flux rate of glycolysis and TCA cycle modules. Clearly, the inhibition of *APEX1* significantly decreased the metabolism rate of glucose. Seeing the accumulations of glycolytic substrates and depletions of TCA cycle substrates, we speculate that the knock-down of *APEX1* may directly impact the downstream part of glycolysis, the whole TCA cycle and further oxidative phosphorylation, leading to accumulation of G1P and G6P as a result of the blockage. Up regulation of glucose transporters was also observed in *APEX1* KD vs control, further suggesting the accumulation of glycolytic substrates.

Perturbation analysis to detect key flux determining genes.

Figure 6.9. (A) Predicted flux fold change (left, x-axis: metabolic module, y-axis: predicted flux change) in control vs *APEX1*-KD, and correlation between fold change of predicted flux and observed metabolite change (right, x-axis: fold change of predicted flux, y-axis: fold change of observed metabolite abundance, each data point is one metabolite, PYR: pyruvate, CIT: citrate, FUM: fumarate, SUC: succinate, MAL: malate). (B) Observed metabolomic change (left, x-axis: metabolites, y-axis: abundance difference observed in the tissue level metabolomics data) in control vs *APEX1*-KD, and correlation between log fold change of gene expressions involved in each reaction and observed metabolomics change (right, x-axis: log fold change of the averaged expression of the genes involved in each reaction, y-axis: fold change of observed metabolites abundance observed in the metabolomics data, each data point is one metabolite). (C) Predicted metabolic stress (left, x-axis: metabolites, y-axis: predicted abundance difference) in control vs *APEX1*-KD and correlation between predicted metabolic stress and observed difference in metabolite abundance (right, x-axis: top scFEA predicted imbalance of the in-/out-flux of intermediate metabolites, y-axis: difference of observed metabolomic abundance, in control vs *APEX1*-KD, each data point is one metabolite: LAC: lactate, SER: serine, GLU: glutamine, ORN: ornithine). In (A-C) all comparisons were made by comparing control vs *APEX1*-KD under normoxia. Noted, the fold change of metabolomic abundance is used in calculating the correlation in A-B and difference of metabolomic abundance is used in B. The green and red dash-blocks represents the accumulated (green) and depleted (red) metabolites in Control vs *APEX1*-KD.


A
Control vs APEX1-KD
Predicted Flux Change
G1P->G6P
G6P->G3P
G3P->3PD
3PD->Pyruvate
Pyruvate->Acetyl-CoA
Pyruvate->Lactate
Acetyl-CoA+OAA->Citrate
Citrate->2OG
2OG->Succinyl-CoA
Succinyl-CoA->Succinate
Succinate->Fumarate
Fumarate->Malate
Malate->OAA
PCC=0.86
p=0.006
FC Metabolites
FC Flux
SUC
FUM
MAL
OAA
PYR
G1P
G6P
CIT
B
Observed Metabolic Change
G1P
G6P
Pyruvate
Lactate
Citrate
2OG
Succinate
Fumarate
Malate
Glutamate
Glutamine
Serine
Ornithine
PCC=-0.03
p=0.943
LogFC Genes
C
Predicted Metabolic Change
Metabolomics Variation
Predicted Flux Imbalance
PCC=0.76
p=0.004
LAC
ORN
GLU
SER

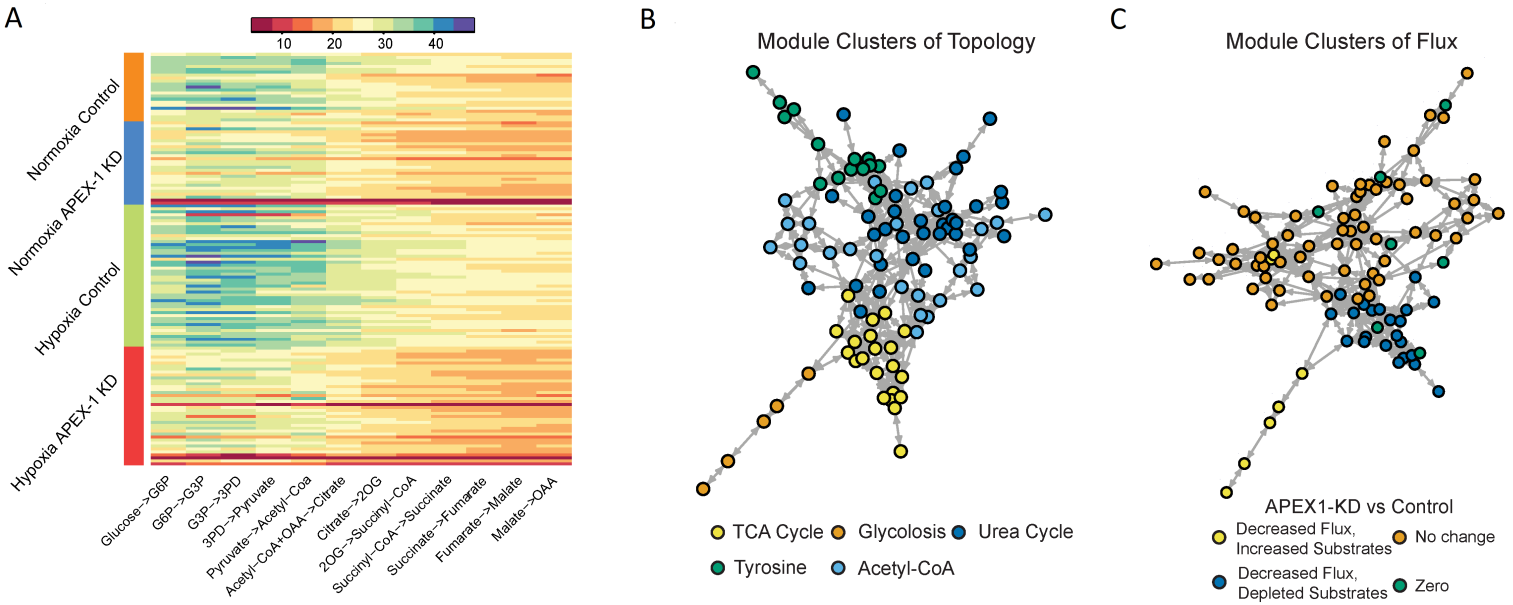


Figure 6.10. (A) Profile of the predicted fluxome of 13 glycolytic and TCA cycle modules. Here each column represents the flux between two metabolites, shown on the x-axis, for all the cells of the four experimental conditions, shown on the y-axis. For two neighboring fluxes, the product of the reaction on the left is the substrate of the reaction on the right, and in a perfectly balanced flux condition, the two neighboring fluxes should be equal. (B) Clusters of metabolic modules inferred by using the network connectivity structure only. (C) Clusters of metabolic modules inferred by using the network topological structure (weight of 0.3) combined with predicted fluxome (weight of 0.7).

We also conducted a perturbation analysis to detect the key genes with high impact on each metabolic module. The following genes were identified to have the highest impact on metabolic flux: *HK1* and *HK2* (Glucose→G6P, EC: 2.7.1.1), *ALDOA*, *PFKL* and *GPI* (G6P→G3P, EC: 5.3.1.9), *GAPDH* and *PGK1* (G3P→3PD, EC: 1.2.1.12, 2.7.2.3), *ENO1*, *PGAM1*, and *PKM* (3PD→Pyruvate, EC: 5.4.2.11, 4.2.1.11), *PDHA2* (Pyruvate→Acetyl-Coa, EC: 1.2.4.1), *LDHA* (Pyruvate→Lactate, EC: 1.1.1.27), *ACLY* (Acetyl-CoA+OAA→Citrate, EC: 2.3.3.8), *IDH2* (Citrate→2OG, EC: 1.1.1.42), *DLD* and *OGDH* (2OG→Succinyl-CoA, EC: 1.2.4.2), *SUCLG1* (Succinyl-CoA→Succinate, EC: 6.2.1.4), *SDHA* (Succinate→Fumarate, EC: 1.3.5.1), *FH* (Fumarate→Malate, EC: 4.2.1.2), *MDH1* (Malate→OAA, EC: 1.1.1.37). A qRT-PCR experiment was conducted to confirm the down regulation of the above key metabolic genes, including *HK1, PFKL, ACLY, SDHA, and IDH2*. We also compared the predicted high impact enzymes in the modules containing multiple enzymes (seven in total) with the rate limiting enzymes reported in Rate-Limiting Enzymes database (RLEdb) [202]. We observed that six out of the seven predicted high impact enzymes, namely 2.7.1.1, 1.2.1.12, 2.7.2.3, 5.4.2.11, 1.2.4.1, and 1.2.4.2, have been reported in RLEdb, suggesting a significant enrichment (p=0.0005 by Fisher's exact test) of our predictions to RLEdb. We further conducted a module level perturbation analysis by increasing or decreasing the expression of genes in a certain module. Consistent to our experimental observations, a decrease of expression on genes of the downstream part of glycolysis pathway in the control cells will lower the flux of the TCA cycle, causing the accumulation of glycolytic intermediate substrates and depletion of TCA cycle metabolites.

Detecting groups of metabolic modules with similar variations and cells with distinct metabolic states.

We also applied scFEA to a larger metabolic map, with the 11 metabolic super modules and transporters. Figure 6.10B illustrated five distinct groups of metabolic modules derived using a spectral clustering method purely based on their network topology, namely (1) glycolysis, (2) TCA cycle and glutamine metabolism related modules, (3) tyrosine and serine metabolism, (4) urea cycle related modules, and (5) acetyl-coA related metabolisms such as fatty acids and propanoyl-CoA metabolisms. To examine the high-level structure based on

the flow of flux, we conducted a clustering analysis of the metabolic modules by considering both the network connectivity and flux similarity. The distance between two modules R_{i} and R_{j} is defined as $\alpha d(R_{\mathrm{i}}, R_{\mathrm{j}}) + (1-\alpha) d^F(R_{\mathrm{i}}, R_{\mathrm{j}})$, where $d(R_{\mathrm{i}}, R_{\mathrm{j}})$ is the normalized spectral distance based on the metabolic network connectivity, and $d^F(R_{\mathrm{i}}, R_{\mathrm{j}})$ is the normalized similarity based on the estimated flux of all the normoxia cells. Here $\alpha = 0.3$ is used in the analysis. Figure 6.10C shows the metabolic module clusters by integrating topological structure and flux similarity. Four distinct clusters were identified, including (1) glycolysis and fatty acids metabolism of decreased flux and accumulated substrates in APEX1-KD vs control, (2) TCA cycle and pyruvate metabolism with decreased flux and depleted substrates, (3) metabolism of amino acids and other metabolites with unchanged flux and metabolites, and (4) a few other modules of 0 flux rates, respectively. This observation further validated the rationality of scFEA predicted fluxome.

We also conducted cell clustering based on the estimated single cell flux. Non-surprisingly, the cell clusters coincide with experimental conditions, forming five group of cells of high, intermediate, and low metabolic rates, high lactate production and low TCA-cycle rate (Figure 6.11).

6.3.5 Method validation and robustness analysis on synthetic and independent real-world data sets

Method validation on independent real-world data.

We also validated scFEA on an independent scRNA-seq data of perivascular adipose tissue derived mesenchymal stem cells (PV-ADSC) (GSE132581) [203] by using hyperparameters $\alpha = 1, \beta = 0$, and $\gamma = 1$. To the best of our knowledge, this data set, in addition to our newly generated data set, are the only two scRNA-seq data with matched tissue level targeted metabolomics profiling available in the public domain. We first re-conducted cell clustering analysis and identified two distinct PV-ADSC cell clusters corresponding to different levels of differentiation, as reported in the original work (Figure 6.12A). Here, the clusters were visualized using UMAP [204]. Due to the small sample size (85 cells), scFEA was applied to estimate only the fluxome of glycolysis and TCA cycle pathways. We observed an increased flux of glycolytic reactions (p<1.56e-6), lactate production (p=0.002),

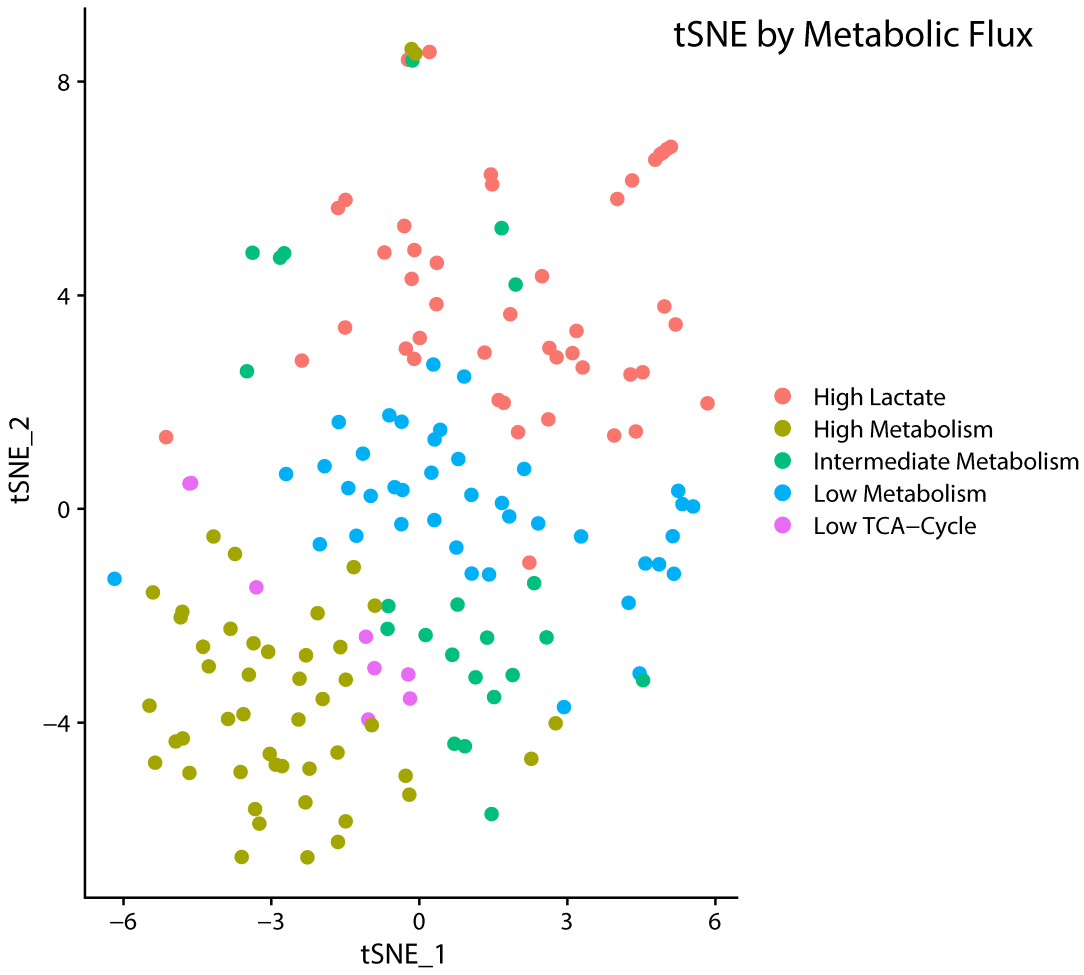


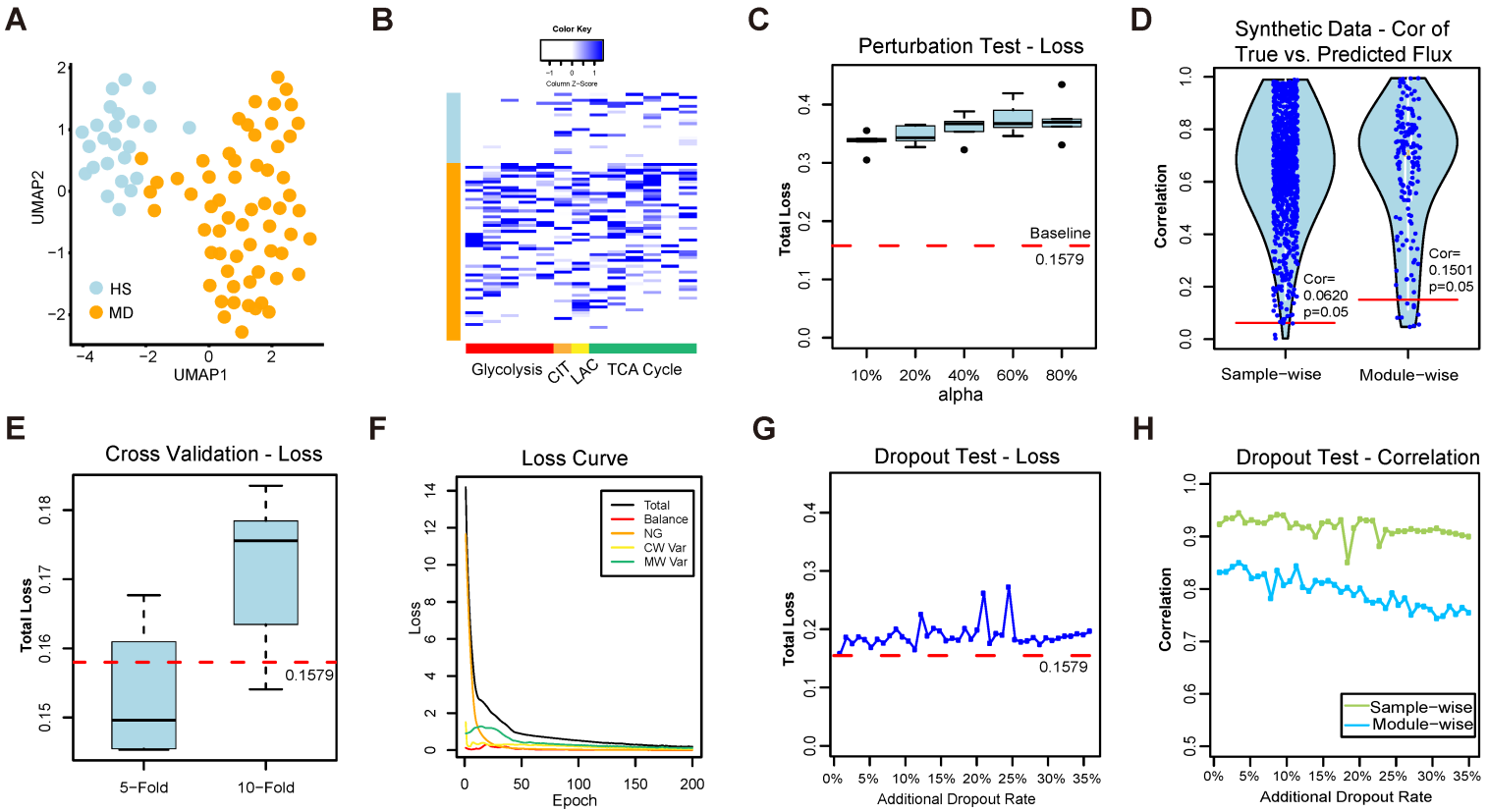
Figure 6.11. tSNE plot of the cell clusters generated based on metabolic flux of the pancreatic cancer cell line data.

and the reactions from cis-aconitate to oxaloacetate in TCA cycle ($p<0.02$) in the more differentiated (MD) vs the high stemness (HS) PV-ADSC cells. The reactions from acetyl-CoA to citrate were not significantly changed ($p=0.887$) (Figure 6.12B). This is consistent to the observations made on the metabolomics data in the original work, i.e., the glycolytic intermediate metabolites, lactate production, and metabolites in the later part of TCA cycle were elevated in the MD cells, while citrate was not significantly changed. We also analyzed the metabolic modules of two amino acids super modules with metabolomics profile reported in the original study, namely valine and isoleucine metabolism and glutamate and glutathione metabolism (Figure 6.13). Elevated valine and isoleucine metabolic flux in MD vs HS cells has been predicted by scFEA, which is consistent to the original report. scFEA also predicted an increased flux of the modules from glutathione $\rightarrow$ glutamate $\rightarrow$ glutamine $\rightarrow$ TCA cycle, this could explain the increased flux rate of TCA cycle but less increase in citrate production. The original study only reported a depletion of glutathione and glutamate, our metabolic stress analysis also predicted more decreased glutathione and glutamate in MD vs HS cells. Our analysis suggested that the elevated glutamate and glutathione metabolism is to fuel the substrate source for TCA cycle in MD cells, which depleted the concentration of glutathione and glutamate.

Method validation on randomly shuffled gene expression profile.

In scFEA, we assume that the flux distribution in each single cell should be constrained by the flux balance condition while the reaction rate of each module could be modeled as a non-linear function of the gene expressions involved in this module. These two assumptions suggested that the distribution of the gene expressions involved in the metabolic modules was constrained by a set of equations governed by the metabolic flux distribution and the flux balance condition. One existing evidence directly supports our assumptions is that the expression of closely related metabolic genes tend to be co-up or co-down regulated [205], [206]. To further validate our assumption, we randomly shuffle the expression profile of each gene in a certain proportion (10%, 20%, 40%, 60% and 80%) of cells in our pancreatic cancer cell line data, and applied scFEA to each shuffled data. We observed that the minimized total loss is positively associated with the level of perturbations (Figure 6.12C) and the original

Figure 6.12. Methods validations on real-world and synthetic datasets. (A) UMAP-based clustering visualization of the GSE132581 PV-ADSC data, here HS and MD stand for PV-ADSC of HS and more differentiation, respectively. (B) Distribution of predicted cell-wise flux of glycolytic and TCA cycle modules. Each row is one cell, where row side color bar represents HS and MD PV-ADSC by blue and orange, respectively. Each column is one module. The left five columns (red labeled) are glycolytic modules from glucose to acetyl-CoA, the CIT column (orange labeled) is the reaction from acetyl-CoA to Citrate, the LAC column (yellow labeled) is the reaction from pyruvate to lactate, and the right six columns (green labeled) are TCA cycle modules from citrate to oxaloacetic acid. (C) The total loss (y-axis) for cases where different proportion (x-axis) of cell samples have randomly shuffled gene expressions of the pancreatic cancer cell line data. The baseline loss 0.1579 was computed using the original expression profile of all 166 cells. (D) The sample-wise and module-wise correlation (y-axis) between the true and predicted module flux in synthetic data-based method validation with multiple repetitions, here Cor=0.5775 (p=0.05) and 0.5778 (p=0.05) correspond to the sample-wise and module-wise correlation, respectively. (E) Total loss (y-axis) computed under 5-/10-fold cross validation (x-axis) vs baseline loss. (F) Convergency of the total loss and four loss terms during the training of neural networks on the pancreatic cancer cell line data. (G) Total loss (y-axis) computed from the robustness test by adding 0%-35 artificial dropouts to the original data (50.22% zero rate) vs baseline loss. (H) Sample-wise and module-wise correlation (y-axis) of the module flux predicted from the data with 0%-35 additional artificial dropouts with the module flux predicted from the original data.


A
UMAP2
UMAP1
HS
MD
B
Color Key
Column Z-Score
Glycolysis
CIT
LAC
TCA Cycle
C
Perturbation Test - Loss
Total Loss
Baseline
0.1579
10%
20%
40%
60%
80%
alpha
D
Synthetic Data - Cor of
True vs. Predicted Flux
Correlation
Cor=
0.0620
p=0.05
Cor=
0.1501
p=0.05
Sample−wise
Module−wise
E
Cross Validation - Loss
Total Loss
0.1579
5−Fold
10−Fold
F
Loss Curve
Loss
Epoch
Total
Balance
NG
CW Var
MW Var
G
Dropout Test - Loss
Total Loss
0.1579
Additional Dropout Rate
H
Dropout Test - Correlation
Correlation
Sample−wise
Module−wise
Additional Dropout Rate

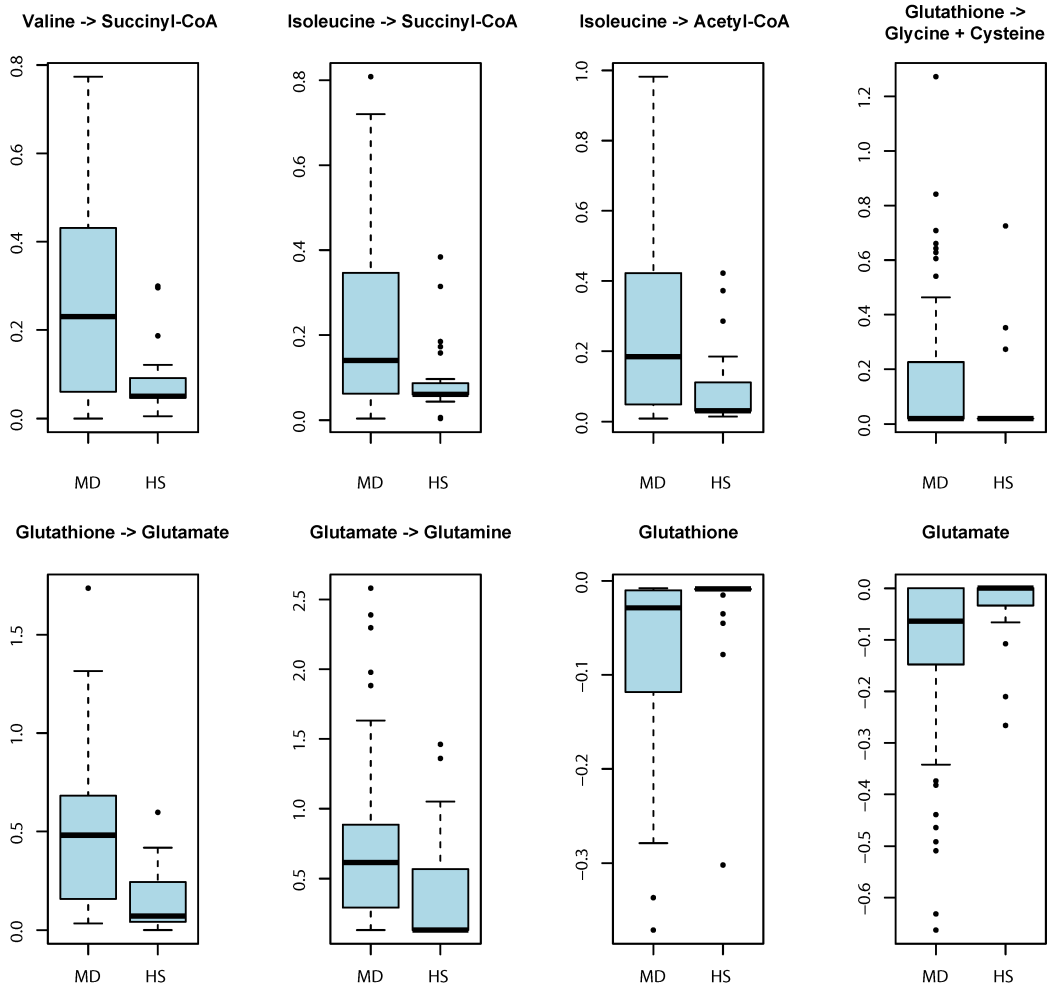


Figure 6.13. Boxplots of the predicted fluxes of Valine -> Succinyl-CoA, Isoleucine -> Succinyl-CoA, Isoleucine -> Acetly-CoA, Glutathione -> Glycine + Cysteine, Glutathione -> Glutamate, Glutamate -> Glutamine and predicted changes in the abundance of Glutathione and Glutamate in the PV-ADSC of high stemness (HS) and more differentiation (MD).

scRNA-seq data achieved the smallest total loss, which partially support our underlying assumption.

Method validation on synthetic data.

We simulated matched metabolic flux and gene expression data on 1,000 single cells. For the 1,000 cells, we first randomly generated different flux distribution of 169 connected modules from the solution space satisfying flux balance condition of these modules. The expression profile of the genes involved in each module was reversely simulated by assuming that its flux follows a fixed non-linear function of the gene expressions. Detailed data simulation approach was provided in Supplementary Methods. We applied scFEA on the simulated single cell gene expression profile and compared the fluxome predicted by scFEA and known fluxome. We observed that scFEA predicted fluxes are highly consistent to the true flux distribution, on both directions of the cells and metabolic modules (Figure 6.12D). Specifically, more than 99.6% single cells achieved at least 0.0620 (p=0.05) sample-wise correlation and more than 84.79% modules achieved at least 0.1501 (p=0.05) module-wise correlation. Our analysis demonstrated that under the assumption of scFEA, i.e., if the flux balance constraint and non-linear dependency between gene expression and metabolic hold, the formulation and solution strategy of scFEA could accurately estimate the cell-wise fluxome from single cell gene expression data.

Robustness analysis based on perturbed sample inputs, cross-validation, and analysis of hyperparameters.

We also tested the robustness of scFEA by 5-/10-fold cross validations on the pancreatic cancer cell line data. Compared with the baseline total loss achieved by using all cells, the total loss of the testing data does not change significantly when using different training cells to train the model (Figure 6.12E). In training the neural networks, scFEA used Adam as the optimizer [207], which can adaptively adjust the learning rate. To choose the most suitable hyperparameters of the four terms in the loss function, we conducted experiments by changing the relative scale of any two terms and fixing the rest two on the pancreatic cancer cell line data. We change the relative ratio of two hyperparameters from 10 to 1000. Our experi-

ments suggested a similar optimal solution can always be achieved under our hyperparameter perturbation range (Figure 6.14). Figure 6.12F showcases the convergence of the four loss terms and total loss in the model fitting of the pancreatic cancer cell line data. In addition, the applications on six real-world data (see further results) and simulated data suggested that the default hyperparameters always generate results of good convergence of the total loss and high biological implications. The default hyperparameters of the current version and details in hyperparameter tunning codes were provided via https://github.com/changwn/scFEA.

Robustness analysis with respect to different level of drop-out

To further examine the method's robustness, we simulated different levels of additional dropout events to our pancreatic cancer cell line data. Our data was collected by using the Smart-seq2-fluidigm protocol, whose original ratio of zero expressions of the metabolic genes is 50.22%. We simulated additional drop-out rate ranging from 4.34% to 34.78%, to reach a typical drop-out level of a droplet based scRNA-seq data (85%), and applied scFEA on the tampered data. We observed that the total loss slightly increases from 0.1649 to 0.2722 when the zero ratio increased from 50% to 85% (Figure 6.12G). The module-wise and cell-wise correlation between the flux estimated from the original data and the tampered data are consistently higher than 0.7437 and 0.8505 (Figure 6.12H), suggesting the high robustness of scFEA with respect to different level of drop-out events.

6.3.6 Application of scFEA on scRNA-seq data of tumor and brain microenvironment revealed distinct metabolic stress, exchange and varied metabolic states in different types of cells

In this section, we majorly focused on validating the computational concept and applicability of scFEA on five real-world datasets, including two scRNA-seq data of cancer microenvironment, one single nuclei RNA-seq data of brain tissue, and one spatial transcriptomics data of breast cancer tissue. The data information is detailed in Supplementary Methods. All 169 metabolic modules across the whole metabolomic network were utilized in the analysis. Due to the lack of matched metabolomics information, we focused on demon-

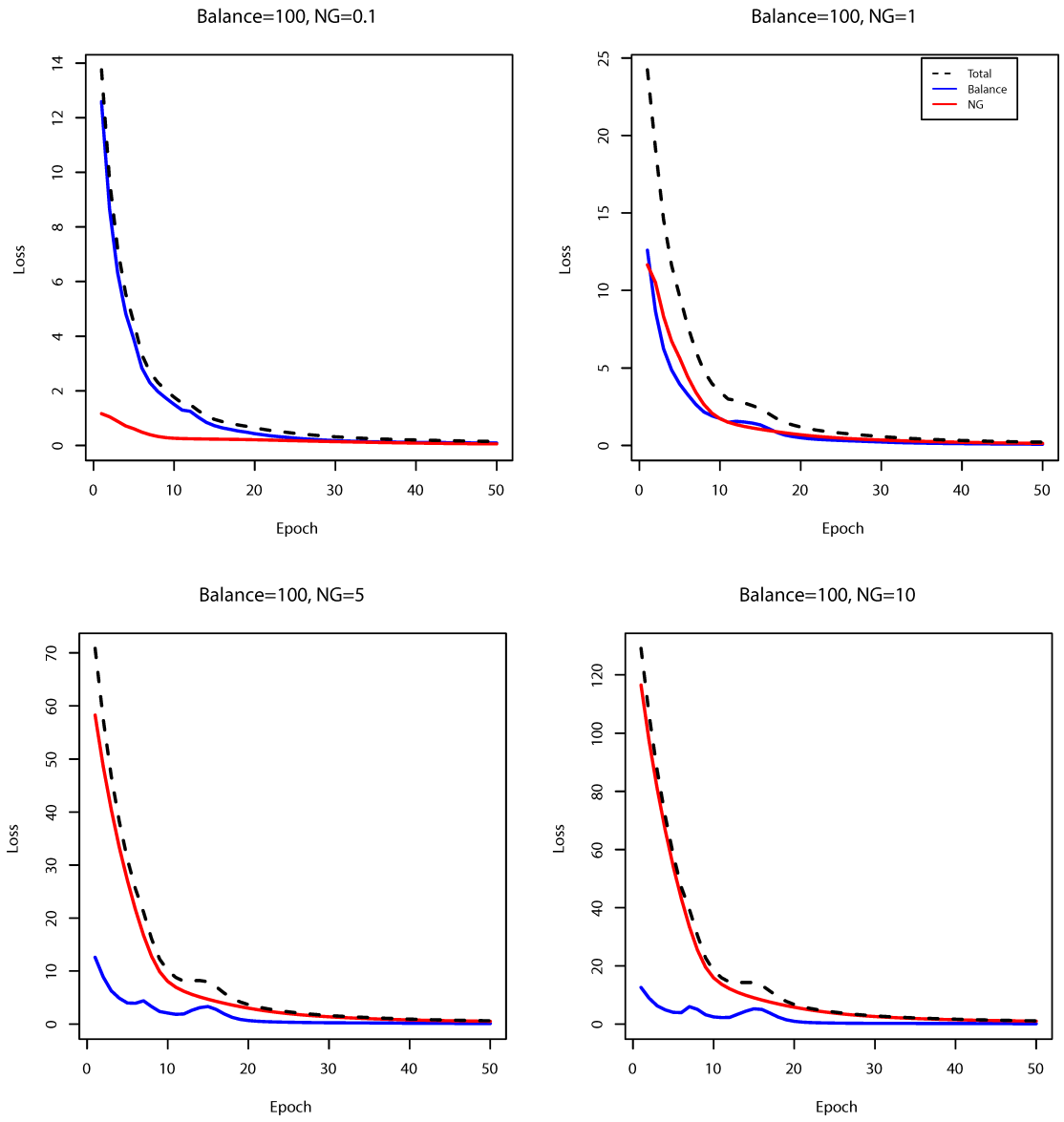


Figure 6.14. Convergency of the flux balance loss and non-negative loss during the training of scFEA on the pancreatic cancer cell line data. The hyperparameters of the two loss were set differently to form four experiments. The flux balance loss, non-negative loss and total loss were blue, red and black-dash colored.

strating the capability of scFEA in inferring metabolic flux, metabolic stress, and subgroups of cells and metabolic modules having distinct variations, on these data sets.

Application on scRNA-seq data of cancer microenvironment

We applied scFEA on two publicly available scRNA-seq datasets collected from the microenvironment of melanoma (GSE72056) and head and neck cancer (GSE103322) by using hyperparameters $\alpha = 1, \beta = 0$, and $\gamma = 1$. In both data sets, we generated UMP based cell and cell group visualization by using predicted fluxomes of the 169 modules (Figure 6.15A-D). We identified that the metabolic flux distributions are quite homogeneous within cancer cells, while being distinct from immune and stromal cells in both data sets (Figure 6.15A, 6.15C). Distinct cell clusters of immune and stromal cells corresponding to varied metabolic fluxomes were also identified (Figure 6.15B, 6.15D). A possible explanation is that cancer cells having a reprogrammed metabolism are more robust to the biochemical variations than immune and stromal cells in the tumor microenvironment.

We observed that the malignant cells have the highest metabolic rates in most metabolic reactions comparing to other cell types in both melanoma and head and neck cancer, especially for the glucose and amino acids metabolic modules (Figure 6.15E, 6.15F). On average, the TCA cycle and lactate production account for 43.4% and 52.5% of the total glycolysis flux in head and neck cancer, and 65.3% and 46.1% of the total glycolysis flux (with additional carbon flow from other metabolites such as glutaminolysis) in melanoma, respectively. In the non-malignant cells, the ratio of lactate production is much lower. Our observation clearly suggested the existence of Warburg effect and metabolic shift in cancer cells, which is consistent to our previous findings of high lactate production in melanoma [208].

We identified that the malignant cells have the highest metabolic stress, which is defined as the total imbalance of intermediate substrates, followed by fibroblast and endothelial cells, and then immune cells. Similar to the pancreatic cancer cell line data, we identified that both cancer and stromal cells in both cancer types tend to have depleted glucose, G1P and G6P. In addition, cancer cells tend to suffer from a high depletion level of acetyl-coA. On the other hand, TCA cycle intermediates and amino acids tend to be accumulated in

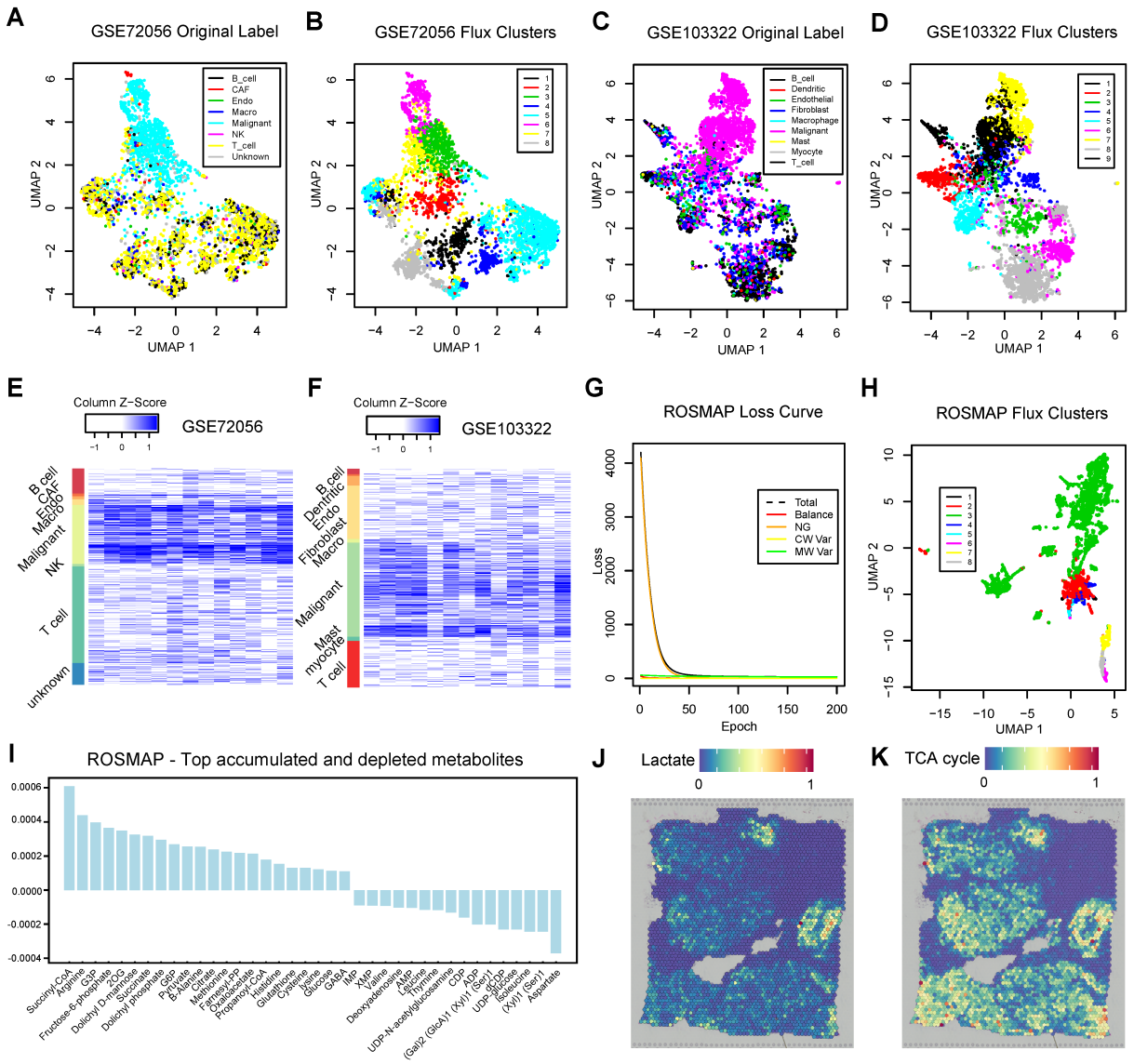
cancer cells. These observations are consistent to the findings derived from quantitative metabolomics data collected on solid cancer [178].

We noticed that the direction of imbalance for most intermediate metabolites seem to be the same throughout different cell types, though the imbalance level is much lower in stromal cells comparing to cancer cells. A possible explanation is that these cells were collected in a small region of the same microenvironment, and the similar stresses, such as hypoxia and altered pH level, cause a similar impact on the metabolic landscape of cells of different types.

Application on droplet based snRNA-seq data of Alzheimer's disease

We also applied scFEA on the ROSMAP snRNA-seq data (single nuclei RNA sequencing) collected from cells in the central nervous systems of Alzheimer's disease (AD) patients and healthy donors [209] by using hyperparameters $\alpha = 1, \beta = 1$, and $\gamma = 1$. Specifically, the ROSMAP snRNA-seq data was collected using the 10x Genomics Chromium droplet-based protocol. Comparing to the Smart-seq based scRNA-seq data, droplet based data often have lower total expression signals and higher dropout rate. scFEA has been successfully applied on this data set. Changes of the total loss over the running epochs suggested the total loss converge to a small value during the training of the scFEA model (Figure 6.15G). Specifically, the flux imbalance loss forms the major loss term in the beginning of the training and quickly converges to a small value, suggesting a solution with good flux balance condition has been identified in this data set. Based on the scFEA predicted flux, we identified that metabolic activity is higher in neuron cells than in other brain cell types. Cell clusters of different metabolic states were computed (Figure 6.15H), in which a large cluster consisting of cells with more active metabolism has been identified (green labeled). We further studied on the metabolic stress of this cell cluster, which is enriched by neuron cells from AD patients (Figure 6.15I). We found that glucose, glycolytic and TCA cycle substrates, and glutathione are among the top accumulated metabolites. Suppressed glycolysis and dysfunctional TCA cycle that may lead to increased glucose and other intermediate metabolites, and elevated glutathione in response to reactive oxygen species, have been reported in AD [210]–[212]. On the other hand, molecules involve in DNA synthesis and valine/leucine/isoleucine metabolism are most depleted in the AD neuron cells, which are consistent to the recently

Figure 6.15. Application on two tumor scRNA-seq datasets, ROSMAP, and one breast cancer spatial transcriptomics dataset. (A) UMAP-based clustering visualization using predicted metabolic fluxes of the GSE72056 melanoma data, the cell label was provided in original work. (B) UMAP-based clustering visualization using predicted metabolic fluxes of the GSE72056, k-means clustering was used for cell clustering. (C) UMAP-based clustering visualization using predicted metabolic fluxes of the GSE103322 head and neck cancer data, the cell label was provided in original work. (D) UMAP-based clustering visualization using predicted metabolic fluxes of the GSE103322, k-means clustering was used for cell clustering. (E) Distribution of predicted cell-wise flux of glycolytic and TCA cycle modules of GSE72056 melanoma data. Each row is one cell, where row side color bar represents 8 cell types. Each column is one module. The left five columns are glycolytic modules from glucose to acetyl-CoA, the 6th column is the reaction from acetyl-CoA to Citrate, the 7th column is the reaction from pyruvate to lactate, and the right-most six columns (8-13 columns) are TCA cycle modules from citrate to oxaloacetic acid. (F) Distribution of predicted cell-wise flux of glycolytic and TCA cycle modules of GSE103322 head and neck cancer data. Each row is one cell, where row side color bar represents 9 cell types, respectively. The column is same as (E). (G) UMAP-based clustering visualization using predicted metabolic fluxes of the ROSMAP data. k-means clustering was used for cell clustering. (H) Convergency curve of the total loss and four loss terms during the training of neural networks on the ROSMAP data. (I) Top accumulated and depleted metabolites predicted in the AD neuron cells in the ROSMAP data. The y-axis is metabolism stress level (or level of accumulation and depletion), where a positive value represents accumulation while a negative value represents depletion. The x-axis are metabolites in a decreasing order of the accumulation level. (J) scFEA predicted flux rate of lactate product on the spatial breast cancer data. The color of each point represents the spatial-wise predicted lactate product rate. The spatial plot is overlaid on the tissue slice image. (K) scFEA predicted flux rate of TCA cycle (citrate to 2OG) on the spatial breast cancer data.


A
GSE72056 Original Label
B_cell
CAF
Endo
Macro
Malignant
NK
T_cell
Unknown
UMAP 1
UMAP 2
B
GSE72056 Flux Clusters
C
GSE103322 Original Label
B_cell
Dendritic
Endothelial
Fibroblast
Macrophage
Malignant
Mast
Myocyte
T_cell
D
GSE103322 Flux Clusters
E
Column Z-Score
GSE72056
B cell
CAF
Endo
Macro
Malignant
NK
T cell
unknown
F
Column Z-Score
GSE103322
B cell
Dendritic
Endo
Fibroblast
Macro
Malignant
Mast
myocyte
T cell
G
ROSMAP Loss Curve
Total
Balance
NG
CW Var
MW Var
Loss
Epoch
H
ROSMAP Flux Clusters
I
ROSMAP - Top accumulated and depleted metabolites
Succinyl-CoA
Arginine
G3P
Fructose-6-phosphate
Dolichyl D-mannose
OOG
Succinate
Dolichyl phosphate
G6P
Pyruvate
Β-Alanine
Citrate
Methionine
Farnesyl-PP
Oxaloacetate
Propanoyl-CoA
Histidine
Glutathione
Cysteine
Serine
Glucose
GABA
IMP
XMP
Valine
Deoxyadenosine
AMP
Leucine
Thymine
CDP
ADP
UDP-N-acetylglucosamine
(Gal)2 (GlcA)1 (Xyl)1 (Ser)1
GDP
UDP-glucose
Adenosine
(Xyl)1 (Ser)1
Aspartate
J
Lactate
K
TCA cycle

reported observations of suppressed DNA synthesis and valine metabolism in AD [213], [214]. We predicted aspartate and metabolites involved in glycosaminoglycan synthesis are largely depleted in the AD neuron cells. Previous studies reported the association of these metabolites to AD [215], [216], however, their abundance change has been less studied. We anticipate that the cell-wise metabolic stress prediction enabled by scFEA could offer novel and systematic insight for biomarker prioritization.

6.4 Discussion

Despite the ample knowledge we have gained on metabolic dysregulation for many disease types, there are still major gaps in our understanding of the integrated behavior and metabolic heterogeneity of cells in the context of tissue microenvironment. Essentially, the metabolic behavior can vary dramatically from cell to cell driven by the need to cope with various dynamic metabolic stress. Large amount of scRNA-seq has proven its potential in delivering information on a cell functioning state and its underlying phenotypic switches. Designing advanced computational tools to harness the power of large scale scRNA-seq data for reliable prediction of cell-wise metabolic flux and states is crucial to bridge the technological gap of single cell metabolomics. Knowledge derived therefrom will substantially benefit our understanding of the metabolic regulation intrinsic to diseased cells, and on microenvironmental factors imposed upon the diseased cells, and shed light on potential therapeutic vulnerabilities.

scFEA is developed to predict metabolic flux at single cell resolution from scRNA-seq data, in order to construct a compendium of metabolic states for different cell types and tissue contexts, and their relevance to various disease phenotypes. To experimentally validate scFEA, we generated an scRNA-seq data of a patient derived pancreatic cancer cells under four conditions of perturbed oxygen level and metabolic regulators, and matched tissue level metabolomics data and qRT-PCR profiles of key metabolic genes. We validated that the variations of metabolic flux predicted by scFEA are highly consistent with the observed metabolomic changes under different conditions. We also applied scFEA on in-drop or droplet based scRNA-seq data and spatial transcriptomics data. Our analysis suggested

that scFEA could robustly predict cell and cell group specific metabolic shift for the data generated from different protocols. Notably, the fluxome estimated by scFEA enables a series of downstream analysis including identification of cell or tissue level metabolic stress, sensitivity evaluation of enzymes to the whole metabolic flux, and inference of cell-tissue and cell-cell metabolic exchanges.

The scFEA model has the following advantages: (1) the model characterizes true biological flux by leveraging the metabolic network structure, and it is generally applicable as it requires only the input of scRNA-seq data; (2) the number of constraints, i.e. the number of flux balance conditions multiplied by the single cell number, is larger than the number of parameters, avoiding possible overfitting; and (3) The neural network based flux estimation can well handle the non-linear dependency between enzymatic gene expression and reaction rates. Notably, in the network reduction and reconstruction of scFEA, connected reactions were merged to form one metabolic module. The neural network model allows for a non-linear dependency between gene expression and module flux. Theoretically, the flux rate could be determined by an "OR" operation of the high expression of any gene involved in the module. scFEA utilizes neighboring genes on the metabolic map to infer the metabolic flux of connected metabolic reactions, which increases robustness to dropout events and prediction accuracy. Our analysis suggested that scFEA is capable of identifying the interactive effect of multiple rate-limiting-enzymes in one module.

scFEA seeks for a constrained optimization of flux balance, where each flux was modeled as a non-linear function of the relevant enzymatic gene expression levels. The flux of each module is constrained by three additional loss terms, namely (1) non-negativity, (2) consistency between predicted metabolic flux and gene expression level, and (3) the cell-wise total metabolic activity, TA_{j}. Although our current setting has been validated using matched scRNA-seq and metabolomics data, applications to publicly available cancer data suggested a similar trend metabolic "imbalance" for both cancer and stromal cells. Our analysis suggested that setting $A_m\mathrm{j}$ for each super module m in cell j may increase the flexibility of cell specific metabolic imbalance, but at the price of possible over-fitting. A more sensitive approach is to train a specific model for each pre-defined cell group. The biological rationale is that the neural network parameters contain the information of "kinetic parameters" that

link gene expression with metabolic rate, which may differ among distant cell types, or cells under different conditions. Hence it is rational to assume cell type specific parameters.

In this study, we did not provide a theoretical proof of the correctness of the scFEA model. Future efforts on generating high quality validating data or more comprehensive systems biological derivations could improve the understanding of the dependency between gene expression and metabolic state in individual cells. Noted, compared with the existing FBA based solutions, which tend to ignore kinetics and assume stringent flux balance condition, our new model treating flux balance as a loss function and seeking for modeling the non-linear dependency between transcriptome and fluxome is more flexible, robust, and scientifically reasonable. Unlike other FBA based approaches, scFEA does not require a prior knowledge of the imports and exports of the whole system. The flux distribution, including the in-/out-fluxes of the system, is estimated by minimizing the loss terms through a large number of cells. We consider such a consideration is more suitable for cell-wise flux estimation since the in and out fluxes are always cell and context specific and unknown. Noted, while the flux balance in scFEA model is robust to the stoichiometric coefficients, the predicted fluxome are represented by a relative reaction rates scaled by total metabolic activity.

The neural network based optimization framework of scFEA could enable a potential integration of metabolomics data, kinetic parameters, spatial information, or other prior knowledge of metabolic imbalance, to better characterize cell and tissue level metabolic shifts of the target system. One future direction is to utilize metabolomics data, kinetic parameters or other prior knowledge to better design the first layer of the neural network in modeling the flux of each module. Spatial information can be utilized to preselect group of cells for training spatially dependent model. Another future direction is to implement the current flux estimation analysis in spatial transcriptomics to infer (1) metabolic shifts specific to spatial patterns and (2) metabolic exchange between adjacent cells. This application to spatial transcriptomics data will be particularly interesting for cancer studies, to reveal how the metabolism and macromolecule biosynthesis in stromal cells such as cancer associated fibroblast, affect the adjacent cancer cells.

Overall, scFEA can efficiently delineate the sophisticated metabolic flux and imbalance specific to certain cell groups. We anticipate that it has the potential to decipher metabolic

heterogeneity, and tease out the metabolomic susceptibility to certain drugs, and ultimately warrant novel mechanistic and therapeutic insights of a diseased biological system at an unprecedented resolution.

7. CONCLUSION

7.1 Thesis Summary

The above projects show the major results of my research in the past five years, which focused on elucidating the heterogeneity of disease and modeling the human metabolic network. Through the thesis, I proposed several computational methods under topic of subspace learning and one systems biology model under topic of human metabolic flux. These two parts try to interpret the mechanism of human disease from different aspects.

In part I, we proposed a series of algorithms which formed a systematic way of subspace learning using mixture model in biomedical data. In Chapter 3, we focus on robust mixture regression to handle outliers. We proposed CAT method to perform outlier detection and parameter estimation simultaneously. In Chapter 4, our interest moves to another case where we have an external biological or clinical response variable to perform clustering of the samples and regression of the response on the features at the same time. We proposed CSMR method to resolve the complexity of translating the clinical representations of the disease to the real causes underpinning it. In Chapter 5, we proposed SRMR method to investigate the relationship between a response variable and a set of explanatory variables over the spatial domain. Our method integrated the robust finite mixture Gaussian regression model with spatial constraints, to simultaneously handle the spatial nonstationarity, local homogeneity, and outlier contamination.

All models proposed in part I showed promising results in both simulations and experiments on real biomedical data. The proposed algorithms are useful computational tools for population research and disease studies.

Particularly, we integrated the CAT method and the CSMR method into the R package "RobMixReg". Based on our knowledge, this is the first comprehensive solution to mining the latent relationships in biomedical data, ranging from the regulatory relationships between different molecular elements, to the relationships between phenotypes and omics features. To avoid confusion, the CSMR method and the SRMR method also have an independent GitHub repository respectively. Till 03/2022, our R package "RobMixReg" has been downloaded around 9,500 times. We believe that our proposed methods will be a valuable addition

to the biomedical research community due to the following contribution: 1) for low dimensional predictor, RobMixReg allows for robust parameter estimations, and detection of the outliers with adaptive trimming; 2) RobMixReg allows for different mixture components to have flexible forms of predictors, which maximally explores the heterogeneous relationships among the hidden subgroups; 3) RobMixReg handles the high-dimensional predictors by regularizing the regression coefficients of each component, with a data-driven level of sparsity; 4) RobMixReg provides the capability for order selection.

In part II, we developed a novel systems biology model to estimate the cell-wise metabolic reaction rates from scRNA-seq data. In this study, we proposed a novel computational method scFEA. To the best of our knowledge, scFEA is the first of its kind of to tackle metabolic heterogeneity using scRNA-seq data. We consider this work with the following five key novelties:

(I) The first computational method that estimates metabolic flux and states at single cell level. The proposed method framework consists of five novel capabilities currently unavailable in the public domain: (1) curating more than 171 metabolic modules of import, metabolism, biosynthesis and exports that cover the whole metabolic map, (2) discovering a compendium of metabolic states, as well as hidden groups of cells possessing the metabolic states, (3) detecting the metabolites and reactions most susceptible to metabolic imbalance in each cell, (4) evaluating the impact of perturbations in enzymes and metabolites to the whole metabolic map, and (5) inferring cell-cell and cell-tissue metabolic exchanges, and the biochemical states at single cell and tissue level.

(II) Advanced computational model. We implemented a novel neural network architecture with the topology structure of metabolic network and the flux balance constraints to effectively capture the non-linear relationships between enzymatic gene expressions and metabolic reaction rates.

(III) Model performances. To experimentally validate scFEA, we generated an scRNA-seq data of a patient derived pancreatic cancer cells under four conditions of perturbed oxygen level and metabolic regulators, and matched tissue level metabolomics data and qRT-PCR profiles of key metabolic genes. We validated that the variations of metabolic

flux predicted by scFEA are highly consistent with the observed metabolomic changes under different conditions.

(IV) Biological insights derived from the application of scFEA. The fluxome predicted by scFEA in cancer cells suggested the accumulation of glycolytic metabolites and depletion of TCA cycle metabolites, caused by suppression of the glycolysis pathway and TCA cycle pathways in both normoxia and hypoxia conditions. Application of scFEA on the data of cancer microenvironment identified different metabolic activities and imbalances between cancer and stromal cells.

(V) Data usage and broader applications. The scFEA can be generally applied to scRNA-seq datasets to estimate cell-wise metabolic flux and build a compendium of well annotated cell type/ tissue specific metabolic states. The method could be reinforced by seamlessly integrating additional metabolomics and proteomics data, spatial, kinetic and other prior information of the reactions. The advanced downstream analysis of cell-wise metabolic stress, metabolic perturbation, and cell-tissue metabolic communication can assist the utilization of a wide spectrum of scRNA-seq data generated in large consortia or individual labs.

Over the past four months, we received more than 100 inquiries regarding the utilization of scFEA, including scientists from top institutions over the world. Further to let more researcher can use our method, we developed a webserver scFLUX as an online tool and code free environment for biology background user. We believe this webserver and our scFEA method provide a novel way to perform metabolic analysis for biomedical community.

7.2 Future Research Directions

In this section, we will discuss some limitation of our current methods and propose several further work and directions.

7.2.1 Subspace Learning

In part I, we proposed a systematic way to study heterogeneity using mixture model and three proposed methods have good performance on the designed scenario. However, three proposed methods still have their own limitation respectively. For CAT method, we

proposed to detect the outlier and estimate parameters of mixture regression model simultaneously. However, CAT was proposed to solve mixture regression problem for two vectors instead of high dimensional data. For example in our real application of CAT method, we performed CAT to analysis the relationship between gene expression of *CREB3L1* and the methylation profile of *cg16012690*. They are two vectors of length 299 which represent 299 colon adenocarcinoma patients. Thus, we cannot adopt CAT method to high dimension case where the input is one matrix and one vector.

For CSMR method, we can handle the high dimensional data because we induced penalized mixture regression model to solve the feature selection problem and the hidden cluster problem. However, this CSMR method has two limitations. The first limitation is that CSMR cannot handle outlier contamination. Although we solve the high dimension issue comparing with CAT, we loss the capability of robustness in CAT. The second limitation is that CSMR becomes less stable when total number of parameters need to be estimated increased a lot comparing with number of observations. In other words, it is still very difficult to estimate all coefficient parameters in each clustering with limited observations. Thus, we highlighted the sparsity assumption of CSMR as before.

Thus, one possible future direction to solve the above limitations of CAT and CSMR synthetically. We need to propose a regularized mixture regression model for high-dimensional data which can handle outlier contamination as well.

For SRMR, our method integrates the robust finite mixture Gaussian regression model with spatial constraints, to simultaneously handle the spatial nonstationarity, local homogeneity, and outlier contamination. The first limitation of SRMR is that we only describe the proposed SRMR model on 2-dimensional spatial data. For example, our current spatial coordination is a 2-dimensional vector including the x-axis and the y-axis coordinates for each sample. The future research includes extending 2-dimensional spatial data to high dimensional. Although high dimensional spatial data is difficult to visualize than 2-dimensional or 3-dimensional coordinates, we can still extend the spatial constraint to K dimensional by proposing a definition of spatial center for high dimensional spatial data. The second limitation of SRMR is that current version SRMR only works for two low dimensional variables. This issue is similar to CAT, we can only analysis the relationship between two genes or one

gene and one methylation profile. The second future research direction for SRMR is proposing a novel model to extend same capability to high dimensional data. For example, input variables could be a matrix instead of vector. This could significantly promote the speed of mechanism discovery for human disease. With respect to spatial consideration, the third future research direction for SRMR is adding spatial smooth process or smooth consideration which can improve the stability of the result. For example, in spatial transcriptomics, we assume better result if we consider one spatial point and its several neighbors together than only considering one point itself.

7.2.2 Computational Modeling of Metabolic Flux

In part II, our proposed method scFEA can efficiently delineate the sophisticated metabolic flux at single cell resolution. However, we did not provide a theoretical proof of the correctness of the scFEA model. Future efforts on generating high quality validating data could provide us deeper insights on the understanding of the dependency between gene expression and metabolic state in individual cells.

In addition, current scFEA utilize PyTorch built-in optimization algorithm to seeking minimal loss. However, this strategy did not consider the topological structure of metabolic network where metabolic modules are connected with known links. Our future direction is to propose a new optimization method over the metabolic network which not only consider computational loss but also consider topology structure of metabolic network.

With the capability of reconstruction of human metabolic network, we can also extend this framework to other small system such as methionine metabolism. This specific analysis could improve our understanding of cell proliferation and gene regulation. This methionine project is our one ongoing project, initial results already show some promising and interesting discovery. Other future direction includes establishing neurotransmitter metabolic framework. Another ongoing project already built several small metabolic systems including acetylcholine, dopamine, histamine, and serotonin. These neurotransmitters play an important role in neuron cells about how human response to environment in a metabolic level activity. This neurotransmitter project could help us understanding the mechanism

of Alzheimer's disease in human brain where neuron cells and microglia cells dominate the region.

REFERENCES

[1] E. Callaway, "'it will change everything': Deepmind's ai makes gigantic leap in solving protein structures," *Nature*, vol. 588, no. 7837, pp. 203–205, 2020. DOI: https://doi.org/10.1038/d41586-020-03348-4.

[2] I. Jovčevska, "Next generation sequencing and machine learning technologies are painting the epigenetic portrait of glioblastoma," *Frontiers in oncology*, vol. 10, p. 798, 2020. DOI: https://doi.org/10.3389/fonc.2020.00798.

[3] X. Li and C.-Y. Wang, "From bulk, single-cell to spatial rna sequencing," *International Journal of Oral Science*, vol. 13, no. 1, pp. 1–6, 2021. DOI: https://doi.org/10.1038/s41368-021-00146-0.

[4] R. C. Olby, *The path to the double helix: the discovery of DNA*. Courier Corporation, 1994, ISBN: 0486681173.

[5] L. M. Baudhuin, S. A. Lagerstedt, E. W. Klee, N. Fadra, D. Oglesbee, and M. J. Ferber, "Confirming variants in next-generation sequencing panel testing by sanger sequencing," *The Journal of Molecular Diagnostics*, vol. 17, no. 4, pp. 456–461, 2015. DOI: https://doi.org/10.1016/j.jmoldx.2015.03.004.

[6] D. Faure and D. Joly, "Next-generation sequencing as a powerful motor for advances in the biological and environmental sciences," *Genetica*, vol. 143, no. 2, pp. 129–132, 2015. DOI: https://doi.org/10.1007/s10709-015-9831-8.

[7] J. K. Kulski, "Next-generation sequencing—an overview of the history, tools, and "omic" applications," *Next generation sequencing-advances, applications and challenges*, vol. 10, p. 61 964, 2016. DOI: https://doi.org/10.5772/61964.

[8] J. D. Watson, "The human genome project: Past, present, and future," *Science*, vol. 248, no. 4951, pp. 44–49, 1990. DOI: https://doi.org/10.1126/science.2181665.

[9] E. Clough and T. Barrett, "The gene expression omnibus database," in *Statistical genomics*, Springer, 2016, pp. 93–110. DOI: https://doi.org/10.1007/978-1-4939-3578-9_5.

[10] N. de Souza, "The encode project," *Nature methods*, vol. 9, no. 11, pp. 1046–1046, 2012. DOI: https://doi.org/10.1038/nmeth.2238.

[11] K. Tomczak, P. Czerwińska, and M. Wiznerowicz, "The cancer genome atlas (tcga): An immeasurable source of knowledge," *Contemporary oncology*, vol. 19, no. 1A, A68, 2015. DOI: https://doi.org/10.5114/wo.2014.47136.

[12] J. Lonsdale, J. Thomas, M. Salvatore, R. Phillips, E. Lo, S. Shad, R. Hasz, G. Walters, F. Garcia, N. Young, *et al.*, "The genotype-tissue expression (gtex) project," *Nature genetics*, vol. 45, no. 6, pp. 580–585, 2013. DOI: https://doi.org/10.1038/ng.2653.

[13] J. November, "More than moore's mores: Computers, genomics, and the embrace of innovation," *Journal of the History of Biology*, vol. 51, no. 4, pp. 807–840, 2018.

[14] V. Chaitankar, G. Karakülah, R. Ratnapriya, F. O. Giuste, M. J. Brooks, and A. Swaroop, "Next generation sequencing technology and genomewide data analysis: Perspectives for retinal research," *Progress in retinal and eye research*, vol. 55, pp. 1–31, 2016. DOI: https://doi.org/10.1016/j.preteyeres.2016.06.001.

[15] P. O. Brown and D. Botstein, "Exploring the new world of the genome with dna microarrays," *Nature genetics*, vol. 21, no. 1, pp. 33–37, 1999. DOI: https://doi.org/10.1038/4462.

[16] J. C. Marioni, C. E. Mason, S. M. Mane, M. Stephens, and Y. Gilad, "Rna-seq: An assessment of technical reproducibility and comparison with gene expression arrays," *Genome research*, vol. 18, no. 9, pp. 1509–1517, 2008. DOI: https://doi.org/10.1101/gr.079558.108.

[17] S. Zhao, W.-P. Fung-Leung, A. Bittner, K. Ngo, and X. Liu, "Comparison of rna-seq and microarray in transcriptome profiling of activated t cells," *PloS one*, vol. 9, no. 1, e78644, 2014. DOI: https://doi.org/10.1371/journal.pone.0078644.

[18] A. A. Pollen, T. J. Nowakowski, J. Shuga, X. Wang, A. A. Leyrat, J. H. Lui, N. Li, L. Szpankowski, B. Fowler, P. Chen, *et al.*, "Low-coverage single-cell mrna sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex," *Nature biotechnology*, vol. 32, no. 10, pp. 1053–1058, 2014. DOI: https://doi.org/10.1038/nbt.2967.

[19] L. Zhang, X. Yu, L. Zheng, Y. Zhang, Y. Li, Q. Fang, R. Gao, B. Kang, Q. Zhang, J. Y. Huang, *et al.*, "Lineage tracking reveals dynamic relationships of t cells in colorectal cancer," *Nature*, vol. 564, no. 7735, pp. 268–272, 2018. DOI: https://doi.org/10.1038/s41586-018-0694-x.

[20] K. B. Halpern, R. Shenhav, O. Matcovitch-Natan, B. Toth, D. Lemze, M. Golan, E. E. Massasa, S. Baydatch, S. Landen, A. E. Moor, *et al.*, "Single-cell spatial reconstruction reveals global division of labour in the mammalian liver," *Nature*, vol. 542, no. 7641, pp. 352–356, 2017. DOI: https://doi.org/10.1038/nature21065.

[21] A. Grover, A. Sanjuan-Pla, S. Thongjuea, J. Carrelha, A. Giustacchini, A. Gambardella, I. Macaulay, E. Mancini, T. C. Luis, A. Mead, *et al.*, "Single-cell rna sequencing reveals molecular and functional platelet bias of aged haematopoietic stem cells," *Nature communications*, vol. 7, no. 1, pp. 1–12, 2016. DOI: https://doi.org/10.1038/ncomms11075.

[22] R. Fisher, L. Pusztai, and C. Swanton, "Cancer heterogeneity: Implications for targeted therapeutics," *British journal of cancer*, vol. 108, no. 3, pp. 479–485, 2013. DOI: https://doi.org/10.1038/bjc.2012.581.

[23] A. A. Kolodziejczyk, J. K. Kim, V. Svensson, J. C. Marioni, and S. A. Teichmann, "The technology and biology of single-cell rna sequencing," *Molecular cell*, vol. 58, no. 4, pp. 610–620, 2015. DOI: https://doi.org/10.1016/j.molcel.2015.04.005.

[24] F. Tang, C. Barbacioru, Y. Wang, E. Nordman, C. Lee, N. Xu, X. Wang, J. Bodeau, B. B. Tuch, A. Siddiqui, *et al.*, "Mrna-seq whole-transcriptome analysis of a single cell," *Nature methods*, vol. 6, no. 5, pp. 377–382, 2009. DOI: https://doi.org/10.1038/nmeth.1315.

[25] J. A. Benitez, S. Cheng, and Q. Deng, "Revealing allele-specific gene expression by single-cell transcriptomics," *The international journal of biochemistry & cell biology*, vol. 90, pp. 155–160, 2017. DOI: https://doi.org/10.1016/j.biocel.2017.05.029.

[26] D. A. Lawson, K. Kessenbrock, R. T. Davis, N. Pervolarakis, and Z. Werb, "Tumour heterogeneity and metastasis at single-cell resolution," *Nature cell biology*, vol. 20, no. 12, pp. 1349–1360, 2018. DOI: https://doi.org/10.1038/s41556-018-0236-7.

[27] G. S. Kinker, A. C. Greenwald, R. Tal, Z. Orlova, M. S. Cuoco, J. M. McFarland, A. Warren, C. Rodman, J. A. Roth, S. A. Bender, *et al.*, "Pan-cancer single-cell rna-seq identifies recurring programs of cellular heterogeneity," *Nature genetics*, vol. 52, no. 11, pp. 1208–1218, 2020. DOI: https://doi.org/10.1038/s41588-020-00726-6.

[28] F. Wu, J. Fan, Y. He, A. Xiong, J. Yu, Y. Li, Y. Zhang, W. Zhao, F. Zhou, W. Li, *et al.*, "Single-cell profiling of tumor heterogeneity and the microenvironment in advanced non-small cell lung cancer," *Nature communications*, vol. 12, no. 1, pp. 1–11, 2021. DOI: https://doi.org/10.1038/s41467-021-22801-0.

[29] S. Picelli, Å. K. Björklund, O. R. Faridani, S. Sagasser, G. Winberg, and R. Sandberg, "Smart-seq2 for sensitive full-length transcriptome profiling in single cells," *Nature methods*, vol. 10, no. 11, pp. 1096–1098, 2013. DOI: https://doi.org/10.1038/nmeth.2639.

[30] X. Wang, Y. He, Q. Zhang, X. Ren, and Z. Zhang, "Direct comparative analyses of 10x genomics chromium and smart-seq2," *Genomics, proteomics & bioinformatics*, vol. 19, no. 2, pp. 253–266, 2021. DOI: https://doi.org/10.1016/j.gpb.2020.02.005.

[31] C. Gao, M. Zhang, and L. Chen, "The comparison of two single-cell sequencing platforms: Bd rhapsody and 10x genomics chromium," *Current Genomics*, vol. 21, no. 8, pp. 602–609, 2020. DOI: https://doi.org/10.2174/1389202921999200625220812.

[32] A. McDavid, G. Finak, P. K. Chattopadyay, M. Dominguez, L. Lamoreaux, S. S. Ma, M. Roederer, and R. Gottardo, "Data exploration, quality control and testing in single-cell qpcr-based gene expression experiments," *Bioinformatics*, vol. 29, no. 4, pp. 461–467, 2013. DOI: https://doi.org/10.1093/bioinformatics/bts714.

[33] W. Hou, Z. Ji, H. Ji, and S. C. Hicks, "A systematic evaluation of single-cell rna-sequencing imputation methods," *Genome biology*, vol. 21, no. 1, pp. 1–30, 2020. DOI: https://doi.org/10.1186/s13059-020-02132-x.

[34] M. Huang, J. Wang, E. Torre, H. Dueck, S. Shaffer, R. Bonasio, J. I. Murray, A. Raj, M. Li, and N. R. Zhang, "Saver: Gene expression recovery for single-cell rna sequencing," *Nature methods*, vol. 15, no. 7, pp. 539–542, 2018. DOI: https://doi.org/10.1038/s41592-018-0033-z.

[35] W. V. Li and J. J. Li, "An accurate and robust imputation method scimpute for single-cell rna-seq data," *Nature communications*, vol. 9, no. 1, pp. 1–9, 2018. DOI: https://doi.org/10.1038/s41467-018-03405-7.

[36] D. Van Dijk, R. Sharma, J. Nainys, K. Yim, P. Kathail, A. J. Carr, C. Burdziak, K. R. Moon, C. L. Chaffer, D. Pattabiraman, *et al.*, "Recovering gene interactions from single-cell data using data diffusion," *Cell*, vol. 174, no. 3, pp. 716–729, 2018. DOI: https://doi.org/10.1016/j.cell.2018.05.061.

[37] R. Lopez, J. Regier, M. B. Cole, M. I. Jordan, and N. Yosef, "Deep generative modeling for single-cell transcriptomics," *Nature methods*, vol. 15, no. 12, pp. 1053–1058, 2018. DOI: https://doi.org/10.1038/s41592-018-0229-2.

[38] P. L. Ståhl, F. Salmén, S. Vickovic, A. Lundmark, J. F. Navarro, J. Magnusson, S. Giacomello, M. Asp, J. O. Westholm, M. Huss, *et al.*, "Visualization and analysis of gene expression in tissue sections by spatial transcriptomics," *Science*, vol. 353, no. 6294, pp. 78–82, 2016. DOI: https://doi.org/10.1126/science.aaf2403.

[39] R. Satija, J. A. Farrell, D. Gennert, A. F. Schier, and A. Regev, "Spatial reconstruction of single-cell gene expression data," *Nature biotechnology*, vol. 33, no. 5, pp. 495–502, 2015. DOI: https://doi.org/10.1038/nbt.3192.

[40] V. Marx, "Method of the year: Spatially resolved transcriptomics," *Nature methods*, vol. 18, no. 1, pp. 9–14, 2021. DOI: https://doi.org/10.1038/s41592-020-01033-y.

[41] M. Asp, J. Bergenstråhle, and J. Lundeberg, "Spatially resolved transcriptomes—next generation tools for tissue exploration," *BioEssays*, vol. 42, no. 10, p. 1 900 221, 2020. DOI: https://doi.org/10.1002/bies.201900221.

[42] M. Efremova, M. Vento-Tormo, S. A. Teichmann, and R. Vento-Tormo, "Cellphonedb: Inferring cell–cell communication from combined expression of multi-subunit ligand–receptor complexes," *Nature protocols*, vol. 15, no. 4, pp. 1484–1506, 2020. DOI: https://doi.org/10.1038/s41596-020-0292-x.

[43] J. A. Ramilowski, T. Goldberg, J. Harshbarger, E. Kloppmann, M. Lizio, V. P. Satagopam, M. Itoh, H. Kawaji, P. Carninci, B. Rost, *et al.*, "A draft network of ligand–receptor-mediated multicellular signalling in human," *Nature communications*, vol. 6, no. 1, pp. 1–12, 2015. DOI: https://doi.org/10.1038/ncomms8866.

[44] W.-T. Chen, A. Lu, K. Craessaerts, B. Pavie, C. S. Frigerio, N. Corthout, X. Qian, J. Laláková, M. Kühnemund, I. Voytyuk, *et al.*, "Spatial transcriptomics and in situ sequencing to study alzheimer's disease," *Cell*, vol. 182, no. 4, pp. 976–991, 2020. DOI: https://doi.org/10.1016/j.cell.2020.06.038.

[45] A. Rao, D. Barkley, G. S. França, and I. Yanai, "Exploring tissue architecture using spatial transcriptomics," *Nature*, vol. 596, no. 7871, pp. 211–220, 2021. DOI: https://doi.org/10.1038/s41586-021-03634-9.

[46] B. L. Gudenas and L. Wang, "Gene coexpression networks in human brain developmental transcriptomes implicate the association of long noncoding rnas with intellectual disability," *Bioinformatics and Biology insights*, vol. 9, BBI–S29435, 2015. DOI: https://doi.org/10.4137/BBI.S29435.

[47] A. O. Fadaka, O. O. Bakare, A. Pretorius, and A. Klein, "Genomic profiling of microrna target genes in colorectal cancer," *Tumor Biology*, vol. 42, no. 6, p. 1 010 428 320 933 512, 2020. DOI: https://doi.org/10.1177/1010428320933512.

[48] L. Parsons, E. Haque, and H. Liu, "Subspace clustering for high dimensional data: A review," *Acm sigkdd explorations newsletter*, vol. 6, no. 1, pp. 90–105, 2004. DOI: https://doi.org/10.1145/1007730.1007731.

[49] A. Rudi, G. D. Canas, and L. Rosasco, "On the sample complexity of subspace learning," *Advances in Neural Information Processing Systems*, vol. 26, 2013. DOI: https://doi.org/10.48550/arXiv.1408.5032.

[50] B. Yang, T.-T. Xin, S.-M. Pang, M. Wang, and Y.-J. Wang, "Deep subspace mutual learning for cancer subtypes prediction," *Bioinformatics*, vol. 37, no. 21, pp. 3715–3722, 2021. DOI: https://doi.org/10.1093/bioinformatics/btab625.

[51] M. Masid and V. Hatzimanikatis, "Quantitative modeling of human metabolism: A call for a community effort," *Current Opinion in Systems Biology*, vol. 26, pp. 109–115, 2021. DOI: https://doi.org/10.1016/j.coisb.2021.04.008.

[52] A. M. Bersani, E. Bersani, G. Dell'Acqua, and M. G. Pedersen, "New trends and perspectives in nonlinear intracellular dynamics: One century from michaelis–menten paper," *Continuum Mechanics and Thermodynamics*, vol. 27, no. 4, pp. 659–684, 2015. DOI: https://doi.org/10.1007/s00161-014-0367-4.

[53] L. Michaelis, M. L. Menten, *et al.*, "Die kinetik der invertinwirkung," *Biochem. z*, vol. 49, no. 333-369, p. 352, 1913. DOI: https://doi.org/10.1111/febs.12598.

[54] S. Goldfeld and R. Quandt, "The estimation of structural shifts by switching regressions," in *Annals of Economic and Social Measurement, Volume 2, number 4*, NBER, 1973, pp. 475–485. DOI: http://www.nber.org/chapters/c9938.pdf.

[55] D. Böhning, *Computer-assisted analysis of mixtures and applications: meta-analysis, disease mapping and others*. CRC press, 1999, vol. 81, ISBN: 0849303850.

[56] C. Hennig, "Identifiablity of models for clusterwise linear regression," *Journal of Classification*, vol. 17, no. 2, 2000. DOI: https://doi.org/10.1007/s003570000022.

[57] W. Jiang and M. A. Tanner, "Hierarchical mixtures-of-experts for exponential family regression models: Approximation and maximum likelihood estimation," *Annals of Statistics*, pp. 987–1011, 1999. DOI: https://doi.org/10.1214/aos/1018031265.

[58] G. J. McLachlan and D. Peel, *Finite mixture models*. John Wiley & Sons, 2004. DOI: https://doi.org/10.1146/annurev-statistics-031017-100325.

[59] L. Xu and M. I. Jordan, "On convergence properties of the em algorithm for gaussian mixtures," *Neural computation*, vol. 8, no. 1, pp. 129–151, 1996. DOI: https://doi.org/10.1162/neco.1996.8.1.129.

[60] S. Frühwirth-Schnatter, *Finite mixture and Markov switching models*. Springer Science & Business Media, 2006. DOI: https://doi.org/10.1007/978-0-387-35768-3.

[61] C. Yu, W. Yao, and G. Yang, "A selective overview and comparison of robust mixture regression estimators," *International Statistical Review*, vol. 88, no. 1, pp. 176–202, 2020. DOI: https://doi.org/10.1111/insr.12349.

[62] M. Markatou, "Mixture models, robustness, and the weighted likelihood methodology," *Biometrics*, vol. 56, no. 2, pp. 483–486, 2000. DOI: https://doi.org/10.1111/j.0006-341X.2000.00483.x.

[63] H.-b. Shen, J. Yang, and S.-t. Wang, "Outlier detecting in fuzzy switching regression models," in *International Conference on Artificial Intelligence: Methodology, Systems, and Applications*, Springer, 2004, pp. 208–215. DOI: https://doi.org/10.1007/978-3-540-30106-6_21.

[64] X. Bai, W. Yao, and J. E. Boyer, "Robust fitting of mixture regression models," *Computational Statistics & Data Analysis*, vol. 56, no. 7, pp. 2347–2359, 2012. DOI: https://doi.org/10.1016/j.csda.2012.01.016.

[65] S. Bashir and E. Carter, "Robust mixture of linear regression models," *Communications in Statistics-Theory and Methods*, vol. 41, no. 18, pp. 3371–3388, 2012. DOI: https://doi.org/10.1080/03610926.2011.558655.

[66] W. Song, W. Yao, and Y. Xing, "Robust mixture regression model fitting by laplace distribution," *Computational Statistics & Data Analysis*, vol. 71, pp. 128–137, 2014. DOI: https://doi.org/10.1016/j.csda.2013.06.022.

[67] W. Yao, Y. Wei, and C. Yu, "Robust mixture regression using the t-distribution," *Computational Statistics & Data Analysis*, vol. 71, pp. 116–127, 2014. DOI: https://doi.org/10.1016/j.csda.2013.07.019.

[68] D. Peel and G. J. McLachlan, "Robust mixture modelling using the t distribution," *Statistics and computing*, vol. 10, no. 4, pp. 339–348, 2000. DOI: https://doi.org/10.1023/A:1008981510081.

[69] C. Yu, W. Yao, and K. Chen, "A new method for robust mixture regression," *Canadian Journal of Statistics*, vol. 45, no. 1, pp. 77–94, 2017. DOI: https://doi.org/10.1002/cjs.11310.

[70] N. Neykov, P. Filzmoser, R. Dimova, and P. Neytchev, "Robust fitting of mixtures using the trimmed likelihood estimator," *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 299–308, 2007. DOI: https://doi.org/10.1016/j.csda.2006.12.024.

[71] G. Celeux and G. Govaert, "A classification em algorithm for clustering and two stochastic versions," *Computational statistics & Data analysis*, vol. 14, no. 3, pp. 315–332, 1992. DOI: https://doi.org/10.1016/0167-9473(92)90042-E.

[72] F. Z. Doğru and O. Arslan, "Robust mixture regression modeling using the least trimmed squares (lts)-estimation method," *Communications in Statistics-Simulation and Computation*, vol. 47, no. 7, pp. 2184–2196, 2018. DOI: https://doi.org/10.1080/03610918.2017.1341528.

[73] P. J. Rousseeuw, "Least median of squares regression," *Journal of the American statistical association*, vol. 79, no. 388, pp. 871–880, 1984. DOI: https://doi.org/10.1080/01621459.1984.10477105.

[74] L. A. García-Escudero, A. Gordaliza, A. Mayo-Íscar, and R. San Martín, "Robust clusterwise linear regression through trimming," *Computational Statistics & Data Analysis*, vol. 54, no. 12, pp. 3057–3069, 2010. DOI: https://doi.org/10.1016/j.csda.2009.07.002.

[75] L. A. García-Escudero, A. Gordaliza, F. Greselin, S. Ingrassia, and A. Mayo-Íscar, "Robust estimation of mixtures of regressions with random covariates, via trimming and constraints," *Statistics and Computing*, vol. 27, no. 2, pp. 377–402, 2017. DOI: https://doi.org/10.1007/s11222-016-9628-3.

[76] C. Fraley and A. E. Raftery, "How many clusters? which clustering method? answers via model-based cluster analysis," *The computer journal*, vol. 41, no. 8, pp. 578–588, 1998. DOI: https://doi.org/10.1093/comjnl/41.8.578.

[77] C. Fraley and A. E. Raftery, "Model-based clustering, discriminant analysis, and density estimation," *Journal of the American statistical Association*, vol. 97, no. 458, pp. 611–631, 2002. DOI: https://doi.org/10.1198/016214502760047131.

[78] C. Wan, W. Chang, Y. Zhang, F. Shah, X. Lu, Y. Zang, A. Zhang, S. Cao, M. L. Fishel, Q. Ma, *et al.*, "Ltmg: A novel statistical modeling of transcriptional expression states in single-cell rna-seq data," *Nucleic acids research*, vol. 47, no. 18, e111–e111, 2019. DOI: https://doi.org/10.1093/nar/gkz655.

[79] J. Xie, A. Ma, Y. Zhang, B. Liu, S. Cao, C. Wang, J. Xu, C. Zhang, and Q. Ma, "Qubic2: A novel and robust biclustering algorithm for analyses and interpretation of large-scale rna-seq data," *Bioinformatics*, vol. 36, no. 4, pp. 1143–1149, 2020. DOI: https://doi.org/10.1093/bioinformatics/btz692.

[80] C. Wan, W. Chang, T. Zhao, M. Li, S. Cao, and C. Zhang, "Mebf: A fast and efficient boolean matrix factorization method," *arXiv preprint arXiv:1909.03991*, 2019. DOI: https://doi.org/10.48550/arXiv.1909.03991.

[81] C. Wan, W. Chang, T. Zhao, S. Cao, and C. Zhang, *Denoising individual bias for a fairer binary submatrix detection*, 2020. DOI: https://doi.org/10.1145/3340531.3412156. arXiv: 2007.15816 [cs.LG].

[82] S. Cao, W. Chang, C. Wan, Y. Zang, J. Zhao, J. Chen, B. Li, Q. Ma, and C. Zhang, "Bi-clustering based biological and clinical characterization of colorectal cancer in complementary to cms classification," *bioRxiv*, p. 508 275, 2018. DOI: https://doi.org/10.1101/508275.

[83] A. Khalili and J. Chen, "Variable selection in finite mixture of regression models," *Journal of the american Statistical association*, vol. 102, no. 479, pp. 1025–1038, 2007. DOI: https://doi.org/10.1198/016214507000000590.

[84] N. Städler, P. Bühlmann, and S. Van De Geer, "L1-penalization for mixture regression models," *Test*, vol. 19, no. 2, pp. 209–256, 2010. DOI: https://doi.org/10.1007/s11749-010-0197-z.

[85] J. Fan and J. Lv, "Comments on: L1-penalization for mixture regression models," *Test*, vol. 19, no. 2, pp. 264–269, 2010. DOI: https://doi.org/10.1007/s11749-010-0200-8.

[86] L. R. Lloyd-Jones, H. D. Nguyen, and G. J. McLachlan, "A globally convergent algorithm for lasso-penalized mixture of linear regression models," *Computational Statistics & Data Analysis*, vol. 119, pp. 19–38, 2018. DOI: https://doi.org/10.1016/j.csda.2017.09.003.

[87] Q. Li, R. Shi, and F. Liang, "Drug sensitivity prediction with high-dimensional mixture regression," *PloS one*, vol. 14, no. 2, 2019. DOI: https://doi.org/10.1371/journal.pone.0212108.

[88] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996. DOI: https://doi.org/10.1111/j.2517-6161.1996.tb02080.x.

[89] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American statistical Association*, vol. 96, no. 456, pp. 1348–1360, 2001. DOI: https://doi.org/10.1198/016214501753382273.

[90] E. Devijver *et al.*, "Finite mixture regression: A sparse variable selection by model selection for clustering," *Electronic journal of statistics*, vol. 9, no. 2, pp. 2642–2674, 2015. DOI: https://doi.org/10.1214/15-EJS1082.

[91] W. Dolde and D. Tirtiroglu, "Temporal and spatial information diffusion in real estate price changes and variances," *Real Estate Economics*, vol. 25, no. 4, pp. 539–565, 1997. DOI: https://doi.org/10.1111/1540-6229.00727.

[92] M. P. Van Den Heuvel and H. E. H. Pol, "Exploring the brain network: A review on resting-state fmri functional connectivity," *European neuropsychopharmacology*, vol. 20, no. 8, pp. 519–534, 2010. DOI: https://doi.org/10.1016/j.euroneuro.2010.03.008.

[93] C. Brunsdon, A. S. Fotheringham, and M. E. Charlton, "Geographically weighted regression: A method for exploring spatial nonstationarity," *Geographical analysis*, vol. 28, no. 4, pp. 281–298, 1996. DOI: https://doi.org/10.1111/j.1538-4632.1996.tb00936.x.

[94] A. S. Fotheringham, C. Brunsdon, and M. Charlton, *Geographically weighted regression: the analysis of spatially varying relationships*. John Wiley & Sons, 2003, ISBN: 0470855258.

[95] D. C. Wheeler and A. Páez, "Geographically weighted regression," in *Handbook of applied spatial analysis*, Springer, 2010, pp. 461–486. DOI: https://doi.org/10.1007/978-3-642-03647-7_22.

[96] M. Fuentes, "Spectral methods for nonstationary spatial processes," *Biometrika*, vol. 89, no. 1, pp. 197–210, 2002. DOI: https://doi.org/10.1093/biomet/89.1.197.

[97] S. Banerjee, B. P. Carlin, and A. E. Gelfand, *Hierarchical modeling and analysis for spatial data*. CRC press, 2014. DOI: https://doi.org/10.1201/9780203487808.

[98] F. Li and H. Sang, "Spatial homogeneity pursuit of regression coefficients for large datasets," *Journal of the American Statistical Association*, vol. 114, no. 527, pp. 1050–1062, 2019. DOI: https://doi.org/10.1080/01621459.2018.1529595.

[99] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and smoothness via the fused lasso," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 1, pp. 91–108, 2005. DOI: https://doi.org/10.1111/j.1467-9868.2005.00490.x.

[100] T. M. Nguyen and Q. J. Wu, "Gaussian-mixture-model-based spatial neighborhood relationships for pixel labeling problem," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 1, pp. 193–202, 2011. DOI: https://doi.org/10.1109/TSMCB.2011.2161284.

[101] T. M. Nguyen and Q. J. Wu, "Fast and robust spatially constrained gaussian mixture model for image segmentation," *IEEE transactions on circuits and systems for video technology*, vol. 23, no. 4, pp. 621–635, 2012. DOI: https://doi.org/10.1109/TCSVT.2012.2211176.

[102] G. J. McLachlan, S. X. Lee, and S. I. Rathnayake, "Finite mixture models," *Annual review of statistics and its application*, vol. 6, pp. 355–378, 2019. DOI: https://doi.org/10.1146/annurev-statistics-031017-100325.

[103] D. A. Reynolds, "Gaussian mixture models.," *Encyclopedia of biometrics*, vol. 741, no. 659-663, 2009. DOI: https://doi.org/10.1007/978-1-4899-7488-4_196.

[104] H. G. Sung, *Gaussian mixture regression and classification*. Rice University, 2004.

[105] G. J. McLachlan and T. Krishnan, *The EM algorithm and extensions*. John Wiley & Sons, 2007, vol. 382. DOI: https://doi.org/10.1016/S0169-7161(05)80138-5.

[106] J. Guinney, R. Dienstmann, X. Wang, A. De Reyniès, A. Schlicker, C. Soneson, L. Marisa, P. Roepman, G. Nyamundanda, P. Angelino, *et al.*, "The consensus molecular subtypes of colorectal cancer," *Nature medicine*, vol. 21, no. 11, pp. 1350–1356, 2015. DOI: https://doi.org/10.1038/nm.3967.

[107] S. Faria and G. Soromenho, "Fitting mixtures of linear regressions," *Journal of Statistical Computation and Simulation*, vol. 80, no. 2, pp. 201–225, 2010. DOI: https://doi.org/10.1080/00949650802590261.

[108] A. F. Siegel, "Robust regression using repeated medians," *Biometrika*, vol. 69, no. 1, pp. 242–244, 1982. DOI: https://doi.org/10.1093/biomet/69.1.242.

[109] J. Blömer, S. Brauer, and K. Bujna, "Hard-clustering with gaussian mixture models," *arXiv preprint arXiv:1603.06478*, 2016. DOI: https://doi.org/10.48550/arXiv.1603.06478.

[110] D. L. Donoho and P. J. Huber, "The notion of breakdown point," *A festschrift for Erich L. Lehmann*, vol. 157184, 1983.

[111] C. Yu and W. Yao, "Robust linear regression: A review and comparison," *Communications in Statistics-Simulation and Computation*, vol. 46, no. 8, pp. 6261–6282, 2017. DOI: https://doi.org/10.1080/03610918.2016.1202271.

[112] P. J. Rousseeuw and A. M. Leroy, *Robust regression and outlier detection*. John wiley & sons, 2005, vol. 589, ISBN: 0471725374.

[113] P. Rousseeuw and V. Yohai, "Robust regression by means of s-estimators," in *Robust and nonlinear time series analysis*, Springer, 1984, pp. 256–272. DOI: https://doi.org/10.1007/978-1-4615-7821-5_15.

[114] G. Pison, S. Van Aelst, and G. Willems, "Small sample corrections for lts and mcd," *Metrika*, vol. 55, no. 1-2, pp. 111–123, 2002. DOI: https://doi.org/10.1007/s001840200191.

[115] A. M. Leroy and P. J. Rousseeuw, "Robust regression and outlier detection," *Wiley Series in Probability and Mathematical Statistics, New York: Wiley, 1987*, 1987.

[116] P. J. Rousseeuw and K. V. Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212–223, 1999. DOI: https://doi.org/10.1080/00401706.1999.10485670.

[117] F. Leisch, "Flexmix: A general framework for finite mixture models and latent glass regression in r," *Journal of Statistical Software*, vol. 11, no. 8, pp. 1–18, 2004. DOI: https://doi.org/10.18637/jss.v011.i08.

[118] G. Celeux, M. Hurn, and C. P. Robert, "Computational and inferential difficulties with mixture posterior distributions," *Journal of the American Statistical Association*, vol. 95, no. 451, pp. 957–970, 2000. DOI: https://doi.org/10.1080/01621459.2000.10474285.

[119] M. Stephens, "Dealing with label switching in mixture models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 62, no. 4, pp. 795–809, 2000. DOI: https://doi.org/10.1111/1467-9868.00265.

[120] W. Yao and B. G. Lindsay, "Bayesian mixture labeling by highest posterior density," *Journal of the American Statistical Association*, vol. 104, no. 486, pp. 758–767, 2009. DOI: https://doi.org/10.1198/jasa.2009.0237.

[121] B. Denard, C. Lee, and J. Ye, "Doxorubicin blocks proliferation of cancer cells through proteolytic activation of creb3l1," *Elife*, vol. 1, e00090, 2012. DOI: https://doi.org/10.7554/eLife.00090.001.

[122] Q. Chen, "Regulation of expression and regulated intramembrane proteolysis of creb3l1," Ph.D. dissertation, UT Southwestern Medical Center, 2013. DOI: https://hdl.handle.net/2152.5/1728.

[123] J. N. Weinstein, E. A. Collisson, G. B. Mills, K. R. M. Shaw, B. A. Ozenberger, K. Ellrott, I. Shmulevich, C. Sander, J. M. Stuart, C. G. A. R. Network, *et al.*, "The cancer genome atlas pan-cancer analysis project," *Nature genetics*, vol. 45, no. 10, p. 1113, 2013. DOI: Thecancergenomeatlaspan-canceranalysisproject.

[124] C. Curtis, S. P. Shah, S.-F. Chin, G. Turashvili, O. M. Rueda, M. J. Dunning, D. Speed, A. G. Lynch, S. Samarajiwa, Y. Yuan, *et al.*, "The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups," *Nature*, vol. 486, no. 7403, pp. 346–352, 2012. DOI: https://doi.org/10.1038/nature10983.

[125] A. Schlicker, G. Beran, C. M. Chresta, G. McWalter, A. Pritchard, S. Weston, S. Runswick, S. Davenport, K. Heathcote, D. A. Castro, *et al.*, "Subtypes of primary colorectal tumors correlate with response to targeted treatment in colorectal cell lines," *BMC medical genomics*, vol. 5, no. 1, p. 66, 2012. DOI: https://doi.org/10.1186/1755-8794-5-66.

[126] A. Marusyk and K. Polyak, "Tumor heterogeneity: Causes and consequences," *Biochimica et Biophysica Acta (BBA)-Reviews on Cancer*, vol. 1805, no. 1, pp. 105–117, 2010. DOI: https://doi.org/10.1016/j.bbcan.2009.11.002.

[127] M. Köbel, S. E. Kalloger, N. Boyd, S. McKinney, E. Mehl, C. Palmer, S. Leung, N. J. Bowen, D. N. Ionescu, A. Rajput, *et al.*, "Ovarian carcinoma subtypes are different diseases: Implications for biomarker studies," *PLoS medicine*, vol. 5, no. 12, e232, 2008. DOI: https://doi.org/10.1371/journal.pmed.0050232.

[128] T. Hastie and J. Qian, "Glmnet vignette," *Retrieved June*, vol. 9, no. 2016, pp. 1–30, 2014. DOI: https://dx.doi.org/10.18637/jss.v033.i01.

[129] W. Pan and X. Shen, "Penalized model-based clustering with application to variable selection," *Journal of Machine Learning Research*, vol. 8, no. May, pp. 1145–1164, 2007.

[130] A. Khalili, J. Chen, and S. Lin, "Feature selection in finite mixture of sparse normal linear models in high-dimensional feature space," *Biostatistics*, vol. 12, no. 1, pp. 156–172, 2011. DOI: https://doi.org/10.1093/biostatistics/kxq048.

[131] J. Barretina, G. Caponigro, N. Stransky, K. Venkatesan, A. A. Margolin, S. Kim, C. J. Wilson, J. Lehár, G. V. Kryukov, D. Sonkin, *et al.*, "The cancer cell line encyclopedia enables predictive modelling of anticancer drug sensitivity," *Nature*, vol. 483, no. 7391, pp. 603–607, 2012. DOI: https://doi.org/10.1038/nature11003.

[132] I. S. Jang, E. C. Neto, J. Guinney, S. H. Friend, and A. A. Margolin, "Systematic assessment of analytical methods for drug sensitivity prediction from cancer cell line data," in *Biocomputing 2014*, World Scientific, 2014, pp. 63–74. DOI: https://doi.org/10.1142/9789814583220_0007.

[133] A. Liberzon, A. Subramanian, R. Pinchback, H. Thorvaldsdóttir, P. Tamayo, and J. P. Mesirov, "Molecular signatures database (msigdb) 3.0," *Bioinformatics*, vol. 27, no. 12, pp. 1739–1740, 2011. DOI: https://doi.org/10.1093/bioinformatics/btr260.

[134] C. Wu, S. Kwon, X. Shen, and W. Pan, "A new algorithm and theory for penalized regression-based clustering," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 6479–6503, 2016. DOI: https://dl.acm.org/doi/10.5555/2946645.3053470.

[135] W. Chang, C. Wan, C. Yu, W. Yao, C. Zhang, and S. Cao, "Robmixreg: An r package for robust, flexible and high dimensional mixture regression," *bioRxiv*, 2020. DOI: https://doi.org/10.1101/2020.08.02.233460.

[136] D. Guo, "Regionalization with dynamically constrained agglomerative clustering and partitioning (redcap)," *International Journal of Geographical Information Science*, vol. 22, no. 7, pp. 801–823, 2008. DOI: https://doi.org/10.1080/13658810701674970.

[137] R. Bivand, L. Anselin, O. Berke, A. Bernat, M. Carvalho, Y. Chun, C. Dormann, S. Dray, R. Halbersma, N. Lewin-Koh, *et al.*, *Spdep: Spatial dependence: Weighting schemes, statistics and models*, 2011. DOI: https://r-spatial.github.io/spdep/.

[138] M. Chavent, V. Kuentz-Simonet, A. Labenne, and J. Saracco, "Clustgeo: An r package for hierarchical clustering with spatial constraints," *Computational Statistics*, vol. 33, no. 4, pp. 1799–1822, 2018. DOI: https://doi.org/10.1007/s00180-018-0791-1.

[139] V. Freguglia and N. L. Garcia, "Mrf2d: Markov random field image models in r," *arXiv preprint arXiv:2006.00383*, 2020. DOI: https://doi.org/10.48550/arXiv.2006.00383.

[140] M. P. Mattson and S. L. Chan, "Dysregulation of cellular calcium homeostasis in alzheimer's disease," *Journal of Molecular Neuroscience*, vol. 17, no. 2, pp. 205–224, 2001. DOI: https://doi.org/10.1385/JMN:17:2:205.

[141] E. Rask, T. Olsson, S. Soderberg, R. Andrew, D. E. Livingstone, O. Johnson, and B. R. Walker, "Tissue-specific dysregulation of cortisol metabolism in human obesity," *The Journal of clinical endocrinology & metabolism*, vol. 86, no. 3, pp. 1418–1421, 2001. DOI: https://doi.org/10.1210/jcem.86.3.7453.

[142] Y. Matsuzawa, "Therapy insight: Adipocytokines in metabolic syndrome and related cardiovascular disease," *Nature clinical practice Cardiovascular medicine*, vol. 3, no. 1, pp. 35–42, 2006. DOI: https://doi.org/10.1038/ncpcardio0380.

[143] L. Dunn, G. F. Allen, A. Mamais, H. Ling, A. Li, K. E. Duberley, I. P. Hargreaves, S. Pope, J. L. Holton, A. Lees, *et al.*, "Dysregulation of glucose metabolism is an early event in sporadic parkinson's disease," *Neurobiology of aging*, vol. 35, no. 5, pp. 1111–1115, 2014. DOI: https://doi.org/10.1016/j.neurobiolaging.2013.11.001.

[144] M. D. Hirschey, R. J. DeBerardinis, A. M. E. Diehl, J. E. Drew, C. Frezza, M. F. Green, L. W. Jones, Y. H. Ko, A. Le, M. A. Lea, *et al.*, "Dysregulated metabolism contributes to oncogenesis," in *Seminars in cancer biology*, Elsevier, vol. 35, 2015, S129–S150. DOI: https://doi.org/10.1016/j.semcancer.2015.10.002.

[145] K. D. Kochanek, S. L. Murphy, J. Xu, and E. Arias, "Deaths: Final data for 2017," 2019. DOI: https://www.cdc.gov/nchs/data/nvsr/nvsr68/nvsr68_09-508.pdf.

[146] H. Sun, Y. Zhou, M. F. Skaro, Y. Wu, Z. Qu, F. Mao, S. Zhao, and Y. Xu, "Metabolic reprogramming in cancer is induced to increase proton production," *Cancer Research*, vol. 80, no. 5, pp. 1143–1155, 2020. DOI: https://doi.org/10.1158/0008-5472.CAN-19-3392.

[147] C. Thompson, D. Bauer, J. Lum, G. Hatzivassiliou, W.-X. ZONG, F. Zhao, D. Ditsworth, M. Buzzai, and T. Lindsten, "How do cancer cells acquire the fuel needed to support cell growth?" In *Cold Spring Harbor symposia on quantitative biology*, Cold Spring Harbor Laboratory Press, vol. 70, 2005, pp. 357–362. DOI: https://doi.org/10.1101/sqb.2005.70.011.

[148] R. J. DeBerardinis, J. J. Lum, G. Hatzivassiliou, and C. B. Thompson, "The biology of cancer: Metabolic reprogramming fuels cell growth and proliferation," *Cell metabolism*, vol. 7, no. 1, pp. 11–20, 2008. DOI: https://doi.org/10.1016/j.cmet.2007.10.002.

[149] D. Hanahan and R. A. Weinberg, "Hallmarks of cancer: The next generation," *cell*, vol. 144, no. 5, pp. 646–674, 2011. DOI: https://doi.org/10.1016/j.cell.2011.02.013.

[150] P. S. Ward and C. B. Thompson, "Metabolic reprogramming: A cancer hallmark even warburg did not anticipate," *Cancer cell*, vol. 21, no. 3, pp. 297–308, 2012. DOI: https://doi.org/10.1016/j.ccr.2012.02.014.

[151] A. L. Bishop, F. A. Rab, E. R. Sumner, and S. V. Avery, "Phenotypic heterogeneity can enhance rare-cell survival in 'stress-sensitive'yeast populations," *Molecular microbiology*, vol. 63, no. 2, pp. 507–520, 2007. DOI: https://doi.org/10.1111/j.1365-2958.2006.05504.x.

[152] M. E. Lidstrom and M. C. Konopka, "The role of physiological heterogeneity in microbial population behavior," *Nature chemical biology*, vol. 6, no. 10, pp. 705–712, 2010. DOI: https://doi.org/10.1038/nchembio.436.

[153] R. Zenobi, "Single-cell metabolomics: Analytical and biological perspectives," *Science*, vol. 342, no. 6163, p. 1 243 259, 2013. DOI: https://doi.org/10.1126/science.1243259.

[154] M. Fessenden, "Metabolomics: Small molecules, single cells," *Nature*, vol. 540, no. 7631, pp. 153–155, 2016. DOI: https://doi.org/10.1038/540153a.

[155] S. Emara, S. Amer, A. Ali, Y. Abouleila, A. Oga, and T. Masujima, "Single-cell metabolomics," *Metabolomics: from fundamentals to clinical applications*, pp. 323–343, 2017. DOI: https://doi.org/10.1007/978-3-319-47656-8_13.

[156] M. Zampieri, K. Sekar, N. Zamboni, and U. Sauer, "Frontiers of high-throughput metabolomics," *Current opinion in chemical biology*, vol. 36, pp. 15–23, 2017. DOI: https://doi.org/10.1016/j.cbpa.2016.12.006.

[157] A. Ali, Y. Abouleila, Y. Shimizu, E. Hiyama, S. Emara, A. Mashaghi, and T. Hankemeier, "Single-cell metabolomics by mass spectrometry: Advances, challenges, and future applications," *TrAC Trends in Analytical Chemistry*, vol. 120, p. 115 436, 2019. DOI: https://doi.org/10.1016/j.trac.2019.02.033.

[158] K. D. Duncan, J. Fyrestam, and I. Lanekoff, "Advances in mass spectrometry based single-cell metabolomics," *Analyst*, vol. 144, no. 3, pp. 782–793, 2019. DOI: https://doi.org/10.1039/C8AN01581C.

[159] P. J. Ahl, R. A. Hopkins, W. W. Xiang, B. Au, N. Kaliaperumal, A.-M. Fairhurst, and J. E. Connolly, "Met-flow, a strategy for single-cell metabolic analysis highlights dynamic changes in immune subpopulations," *Communications biology*, vol. 3, no. 1, pp. 1–15, 2020. DOI: https://doi.org/10.1038/s42003-020-1027-9.

[160] R. Jaenisch and A. Bird, "Epigenetic regulation of gene expression: How the genome integrates intrinsic and environmental signals," *Nature genetics*, vol. 33, no. 3, pp. 245–254, 2003. DOI: https://doi.org/10.1038/ng1089.

[161] A. P. Feinberg, "Phenotypic plasticity and the epigenetics of human disease," *Nature*, vol. 447, no. 7143, pp. 433–440, 2007. DOI: https://doi.org/10.1038/nature05919.

[162] N. D. Heintzman, R. K. Stuart, G. Hon, Y. Fu, C. W. Ching, R. D. Hawkins, L. O. Barrera, S. Van Calcar, C. Qu, K. A. Ching, *et al.*, "Distinct and predictive chromatin signatures of transcriptional promoters and enhancers in the human genome," *Nature genetics*, vol. 39, no. 3, pp. 311–318, 2007. DOI: https://doi.org/10.1038/ng1966.

[163] R. A. Harris, T. Wang, C. Coarfa, R. P. Nagarajan, C. Hong, S. L. Downey, B. E. Johnson, S. D. Fouse, A. Delaney, Y. Zhao, *et al.*, "Comparison of sequencing-based methods to profile dna methylation and identification of monoallelic epigenetic modifications," *Nature biotechnology*, vol. 28, no. 10, pp. 1097–1105, 2010. DOI: https://doi.org/10.1038/nbt.1682.

[164] E. P. Consortium *et al.*, "An integrated encyclopedia of dna elements in the human genome," *Nature*, vol. 489, no. 7414, p. 57, 2012. DOI: https://doi.org/10.1038/nature11247.

[165] A. Kundaje, W. Meuleman, J. Ernst, M. Bilenky, A. Yen, A. Heravi-Moussavi, P. Kheradpour, Z. Zhang, J. Wang, M. J. Ziller, *et al.*, "Integrative analysis of 111 reference human epigenomes," *Nature*, vol. 518, no. 7539, pp. 317–330, 2015. DOI: https://doi.org/10.1038/nature14248.

[166] M. Robertson-Tessi, R. J. Gillies, R. A. Gatenby, and A. R. Anderson, "Impact of metabolic heterogeneity on tumor growth, invasion, and treatment outcomes," *Cancer research*, vol. 75, no. 8, pp. 1567–1579, 2015. DOI: https://doi.org/10.1158/0008-5472.CAN-14-1428.

[167] J. Kim and R. J. DeBerardinis, "Mechanisms and implications of metabolic heterogeneity in cancer," *Cell metabolism*, vol. 30, no. 3, pp. 434–446, 2019. DOI: https://doi.org/10.1016/j.cmet.2019.08.013.

[168] A. E. Vasdekis and G. Stephanopoulos, "Review of methods to probe single cell metabolism and bioenergetics," *Metabolic engineering*, vol. 27, pp. 115–135, 2015. DOI: https://doi.org/10.1016/j.ymben.2014.09.007.

[169] C. Damiani, D. Maspero, M. Di Filippo, R. Colombo, D. Pescini, A. Graudenzi, H. V. Westerhoff, L. Alberghina, M. Vanoni, and G. Mauri, "Integration of single-cell rna-seq data into population models to characterize cancer metabolism," *PLoS computational biology*, vol. 15, no. 2, e1006733, 2019. DOI: https://doi.org/10.1371/journal.pcbi.1006733.

[170] T. M. Evers, M. Hochane, S. J. Tans, R. M. Heeren, S. Semrau, P. Nemes, and A. Mashaghi, *Deciphering metabolic heterogeneity by single-cell analysis*, 2019. DOI: https://doi.org/10.1021/acs.analchem.9b02410.

[171] H. Honkoop, D. E. de Bakker, A. Aharonov, F. Kruse, A. Shakked, P. D. Nguyen, C. de Heus, L. Garric, M. J. Muraro, A. Shoffner, *et al.*, "Single-cell analysis uncovers that metabolic reprogramming by erbb2 signaling is essential for cardiomyocyte proliferation in the regenerating heart," *Elife*, vol. 8, e50163, 2019. DOI: https://doi.org/10.7554/eLife.50163.

[172] M. S. Saurty-Seerunghen, L. Bellenger, E. A. El-Habr, V. Delaunay, D. Garnier, H. Chneiweiss, C. Antoniewski, G. Morvan-Dubois, and M.-P. Junier, "Capture at the single cell level of metabolic modules distinguishing aggressive and indolent glioblastoma cells," *Acta neuropathologica communications*, vol. 7, no. 1, pp. 1–16, 2019. DOI: https://doi.org/10.1186/s40478-019-0819-y.

[173] Z. Xiao, Z. Dai, and J. W. Locasale, "Metabolic landscape of the tumor microenvironment at single cell resolution," *Nature communications*, vol. 10, no. 1, pp. 1–12, 2019. DOI: https://doi.org/10.1038/s41467-019-11738-0.

[174] K. Rohlenova, J. Goveia, M. García-Caballero, A. Subramanian, J. Kalucka, L. Treps, K. D. Falkenberg, L. P. de Rooij, Y. Zheng, L. Lin, *et al.*, "Single-cell rna sequencing maps endothelial metabolic plasticity in pathological angiogenesis," *Cell metabolism*, vol. 31, no. 4, pp. 862–877, 2020. DOI: https://doi.org/10.1016/j.cmet.2020.03.009.

[175] Z. Xiao, J. W. Locasale, and Z. Dai, "Metabolism in the tumor microenvironment: Insights from single-cell analysis," *Oncoimmunology*, vol. 9, no. 1, p. 1 726 556, 2020. DOI: https://doi.org/10.1080/2162402X.2020.1726556.

[176] Y. Zhang, M. S. Kim, E. Nguyen, and D. M. Taylor, "Modeling metabolic variation with single-cell expression data," *bioRxiv*, 2020. DOI: https://doi.org/10.1101/2020.01.28.923680.

[177] L. S. Levine, K. J. Hiam, D. M. Marquez, I. Tenvooren, D. C. Contreras, J. C. Rathmell, and M. H. Spitzer, "Single-cell metabolic dynamics of early activated cd8 t cells during the primary immune response to infection," *bioRxiv*, 2020. DOI: https://doi.org/10.1101/2020.01.21.911545.

[178] A. Hirayama, K. Kami, M. Sugimoto, M. Sugawara, N. Toki, H. Onozuka, T. Kinoshita, N. Saito, A. Ochiai, M. Tomita, *et al.*, "Quantitative metabolome profiling of colon and stomach cancer microenvironment by capillary electrophoresis time-of-flight mass spectrometry," *Cancer research*, vol. 69, no. 11, pp. 4918–4925, 2009. DOI: https://doi.org/10.1158/0008-5472.CAN-08-4806.

[179] D. Lee, K. Smallbone, W. B. Dunn, E. Murabito, C. L. Winder, D. B. Kell, P. Mendes, and N. Swainston, "Improving metabolic flux predictions using absolute gene expression data," *BMC systems biology*, vol. 6, no. 1, pp. 1–9, 2012. DOI: https://doi.org/10.1186/1752-0509-6-73.

[180] M. Mehrmohamadi, X. Liu, A. A. Shestov, and J. W. Locasale, "Characterization of the usage of the serine metabolic network in human cancer," *Cell reports*, vol. 9, no. 4, pp. 1507–1519, 2014. DOI: https://doi.org/10.1016/j.celrep.2014.10.026.

[181] A. Wagner, C. Wang, D. DeTomaso, J. Avila-Pacheco, S. Zaghouani, J. Fessler, S. Eyzaguirre, E. Akama-Garren, K. Pierce, N. Ron-Harel, *et al.*, "In silico modeling of metabolic state in single th17 cells reveals novel regulators of inflammation and autoimmunity," *bioRxiv*, 2020. DOI: https://doi.org/10.1101/2020.01.23.912717.

[182] M. Kanehisa and S. Goto, "Kegg: Kyoto encyclopedia of genes and genomes," *Nucleic acids research*, vol. 28, no. 1, pp. 27–30, 2000. DOI: https://doi.org/10.1093/nar/28.1.27.

[183] S. Cao, X. Zhu, C. Zhang, H. Qian, H.-B. Schuttler, J. Gong, and Y. Xu, "Competition between dna methylation, nucleotide synthesis, and antioxidation in cancer versus normal tissues," *Cancer Research*, vol. 77, no. 15, pp. 4185–4195, 2017. DOI: https://doi.org/10.1158/0008-5472.CAN-17-0262.

[184] L. Lin, S. W. Yee, R. B. Kim, and K. M. Giacomini, "Slc transporters as therapeutic targets: Emerging opportunities," *Nature reviews Drug discovery*, vol. 14, no. 8, pp. 543–560, 2015. DOI: https://doi.org/10.1038/nrd4626.

[185] Y. D. Bhutia, E. Babu, S. Ramachandran, S. Yang, M. Thangaraju, and V. Ganapathy, "Slc transporters as a novel class of tumour suppressors: Identity, function and molecular mechanisms," *Biochemical Journal*, vol. 473, no. 9, pp. 1113–1124, 2016. DOI: https://doi.org/10.1042/BJ20150751.

[186] B. A. Moffatt and H. Ashihara, "Purine and pyrimidine nucleotide synthesis and metabolism," *The Arabidopsis Book/American Society of Plant Biologists*, vol. 1, 2002. DOI: https://doi.org/10.1199/tab.0018.

[187] P. L. DeAngelis, J. Liu, and R. J. Linhardt, "Chemoenzymatic synthesis of glycosaminoglycans: Re-creating, re-modeling and re-designing nature's longest or most complex carbohydrate chains," *Glycobiology*, vol. 23, no. 7, pp. 764–777, 2013. DOI: https://doi.org/10.1093/glycob/cwt016.

[188] C. Zhang, S. Cao, B. P. Toole, and Y. Xu, "Cancer may be a pathway to cell survival under persistent hypoxia and elevated ros: A model for solid-cancer initiation and early development," *International journal of cancer*, vol. 136, no. 9, pp. 2001–2011, 2015. DOI: https://doi.org/10.1002/ijc.28975.

[189] C. Zhang, C. Liu, S. Cao, and Y. Xu, "Elucidation of drivers of high-level production of lactates throughout a cancer development," *Journal of molecular cell biology*, vol. 7, no. 3, pp. 267–279, 2015. DOI: https://doi.org/10.1093/jmcb/mjv031.

[190] L. Krasnova and C.-H. Wong, "Understanding the chemistry and biology of glycosylation with glycan synthesis," *Annual review of biochemistry*, vol. 85, pp. 599–630, 2016. DOI: https://doi.org/10.1146/annurev-biochem-060614-034420.

[191] M. Zulueta, S.-Y. Lin, Y.-P. Hu, and S.-C. Hung, "Synthesis of glycosaminoglycans," *Glycochemical synthesis: strategies and applications (ed. Hung SC, Zulueta MM)*, pp. 235–261, 2016. DOI: https://doi.org/10.1021/acs.chemrev.6b00010.

[192] C. Gao and K. J. Edgar, "Efficient synthesis of glycosaminoglycan analogs," *Biomacromolecules*, vol. 20, no. 2, pp. 608–617, 2018. DOI: https://doi.org/10.1021/acs.biomac.8b01150.

[193] J. S. Yedidia, W. Freeman, and Y. Weiss, "Generalized belief propagation," *Advances in neural information processing systems*, vol. 13, 2000. DOI: https://dl.acm.org/doi/10.5555/3008751.3008848.

[194] M. L. Fishel, X. Wu, C. M. Devlin, D. P. Logsdon, Y. Jiang, M. Luo, Y. He, Z. Yu, Y. Tong, K. P. Lipking, *et al.*, "Apurinic/apyrimidinic endonuclease/redox factor-1 (ape1/ref-1) redox function negatively regulates nrf2," *Journal of Biological Chemistry*, vol. 290, no. 5, pp. 3057–3068, 2015. DOI: https://doi.org/10.1074/jbc.M114.621995.

[195] S. Schnell, "Validity of the michaelis–menten equation–steady-state or reactant stationary assumption: That is the question," *The FEBS journal*, vol. 281, no. 2, pp. 464–472, 2014. DOI: https://doi.org/10.1111/febs.12564.

[196] Y. Liu, A. Beyer, and R. Aebersold, "On the dependency of cellular protein levels on mrna abundance," *Cell*, vol. 165, no. 3, pp. 535–550, 2016. DOI: https://doi.org/10.1016/j.cell.2016.03.014.

[197] M. H. Saier Jr, C. V. Tran, and R. D. Barabote, "Tcdb: The transporter classification database for membrane transport protein analyses and information," *Nucleic acids research*, vol. 34, no. suppl_1, pp. D181–D186, 2006. DOI: https://doi.org/10.1093/nar/gkj001.

[198] X. Lan, S. Roth, D. Huttenlocher, and M. J. Black, "Efficient belief propagation with learned higher-order markov random fields," in *European conference on computer vision*, Springer, 2006, pp. 269–282. DOI: https://doi.org/10.1007/11744047_21.

[199] M. R Kelley, M. M Georgiadis, and M. L Fishel, "Ape1/ref-1role in redox signaling: Translational applications of targeting the redox function of the dna repair/redox protein ape1/ref-1," *Current molecular pharmacology*, vol. 5, no. 1, pp. 36–53, 2012. DOI: https://doi.org/10.2174/1874467211205010036.

[200] F. Shah, E. Goossens, N. M. Atallah, M. Grimard, M. R. Kelley, and M. L. Fishel, "Ape 1/ref-1 knockdown in pancreatic ductal adenocarcinoma–characterizing gene expression changes and identifying novel pathways using single-cell rna sequencing," *Molecular oncology*, vol. 11, no. 12, pp. 1711–1732, 2017. DOI: https://doi.org/10.1002/1878-0261.12138.

[201] Y.-P. Chen, J.-H. Yin, W.-F. Li, H.-J. Li, D.-P. Chen, C.-J. Zhang, J.-W. Lv, Y.-Q. Wang, X.-M. Li, J.-Y. Li, *et al.*, "Single-cell transcriptomics reveals regulators underlying immune cell diversity and immune subtypes associated with prognosis in nasopharyngeal carcinoma," *Cell research*, vol. 30, no. 11, pp. 1024–1042, 2020. DOI: https://doi.org/10.1038/s41422-020-0374-x.

[202] M. Zhao, X. Chen, G. Gao, L. Tao, and L. Wei, "Rledb: A database of rate-limiting enzymes and their regulation in human, rat, mouse, yeast and e. coli," *Cell research*, vol. 19, no. 6, pp. 793–795, 2009. DOI: https://doi.org/10.1038/cr.2009.61.

[203] W. Gu, W. N. Nowak, Y. Xie, A. Le Bras, Y. Hu, J. Deng, S. Issa Bhaloo, Y. Lu, H. Yuan, E. Fidanis, *et al.*, "Single-cell rna-sequencing and metabolomics analyses reveal the contribution of perivascular adipose tissue stem cells to vascular remodeling," *Arteriosclerosis, thrombosis, and vascular biology*, vol. 39, no. 10, pp. 2049–2066, 2019. DOI: https://doi.org/10.1161/ATVBAHA.119.312732.

[204] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018. DOI: https://doi.org/10.48550/arXiv.1802.03426.

[205] J. A. van der Knaap and C. P. Verrijzer, "Undercover: Gene control by metabolites and metabolic enzymes," *Genes & development*, vol. 30, no. 21, pp. 2345–2369, 2016. DOI: https://doi.org/10.1101/gad.289140.116.

[206] X. Li, G. Egervari, Y. Wang, S. L. Berger, and Z. Lu, "Regulation of chromatin and gene expression by metabolic enzymes and metabolites," *Nature reviews Molecular cell biology*, vol. 19, no. 9, pp. 563–578, 2018. DOI: https://doi.org/10.1038/s41580-018-0029-7.

[207] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. DOI: https://doi.org/10.48550/arXiv.1412.6980.

[208] K. Xu, X. Mao, M. Mehta, J. Cui, C. Zhang, and Y. Xu, "A comparative study of gene-expression data of basal cell carcinoma and melanoma reveals new insights about the two cancers," *PloS one*, vol. 7, no. 1, e30750, 2012. DOI: https://doi.org/10.1371/journal.pone.0030750.

[209] H. Mathys, J. Davila-Velderrain, Z. Peng, F. Gao, S. Mohammadi, J. Z. Young, M. Menon, L. He, F. Abdurrob, X. Jiang, *et al.*, "Single-cell transcriptomic analysis of alzheimer's disease," *Nature*, vol. 570, no. 7761, pp. 332–337, 2019. DOI: https://doi.org/10.1038/s41586-019-1195-2.

[210] H. Atamna and W. H. Frey II, "Mechanisms of mitochondrial dysfunction and energy deficiency in alzheimer's disease," *Mitochondrion*, vol. 7, no. 5, pp. 297–310, 2007. DOI: https://doi.org/10.1016/j.mito.2007.06.001.

[211] P. K. Mandal, D. Shukla, M. Tripathi, and L. Ersland, "Cognitive improvement with glutathione supplement in alzheimer's disease: A way forward," *Journal of Alzheimer's Disease*, vol. 68, no. 2, pp. 531–535, 2019. DOI: https://doi.org/10.3233/JAD-181054.

[212] J. Le Douce, M. Maugard, J. Veran, M. Matos, P. Jégo, P.-A. Vigneron, E. Faivre, X. Toussay, M. Vandenberghe, Y. Balbastre, *et al.*, "Impairment of glycolysis-derived l-serine production in astrocytes contributes to cognitive deficits in alzheimer's disease," *Cell metabolism*, vol. 31, no. 3, pp. 503–517, 2020. DOI: https://doi.org/10.1016/j.cmet.2020.02.004.

[213] Y. B. Yurov, S. G. Vorsanova, and I. Y. Iourov, "The dna replication stress hypothesis of alzheimer's disease," *TheScientificWorldJOURNAL*, vol. 11, pp. 2602–2612, 2011. DOI: https://doi.org/10.1100/2011/625690.

[214] B. Polis and A. O. Samson, "Role of the metabolism of branched-chain amino acids in the development of alzheimer's disease and other metabolic disorders," *Neural regeneration research*, vol. 15, no. 8, p. 1460, 2020. DOI: https://dx.doi.org/10.4103/F1673-5374.274328.

[215] P. M. Doraiswamy, "The role of the n-methyl-d-aspartate receptor in alzheimer's disease: Therapeutic potential," *Current neurology and neuroscience reports*, vol. 3, no. 5, pp. 373–378, 2003. DOI: https://doi.org/10.1007/s11910-003-0019-8.

[216] M. B. Huynh, M. O. Ouidja, S. Chantepie, G. Carpentier, A. Maïza, G. Zhang, J. Vilares, R. Raisman-Vozari, and D. Papy-Garcia, "Glycosaminoglycans from alzheimer's disease hippocampus have altered capacities to bind and regulate growth factors activities and to bind tau," *PloS one*, vol. 14, no. 1, e0209573, 2019. DOI: https://doi.org/10.1371/journal.pone.0209573.

VITA

Wennan Chang received the B.E. and M.S. degree in College of Computer and Control Engineering from Nankai University in 2014 and 2017, respectively. He began his PhD studies in Electrical and Computer Engineering at Purdue Unversity and works as a graduate research assistance in the Biomedical Data Research Lab at Indiana University School of Medicine. He worked under the co-supervision of Professor Chi Zhang and Professor Mireille Boutin. His research interests are broadly at the intersection of algorithms, mathematical modeling and optimization, with a focus on applications in machine learning and cancer genomic data science.