STATISTICAL INTEGRATIVE OMICS METHODS FOR DISEASE SUBTYPE DISCOVERY

by

Zhiguang Huo

M.S in Physics, University of Pittsburgh, 2012

B.S in Physics, Harbin Institute of Technology, China, 2011

Submitted to the Graduate Faculty of

the Department of Biostatistics

Graduate School of Public health in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2017

UNIVERSITY OF PITTSBURGH

GRADUATE SCHOOL OF PUBLIC HEALTH

This dissertation was presented

by

Zhiguang Huo

It was defended on

March 30th 2017

and approved by

George C. Tseng, ScD, Professor, Department of Biostatistics, Graduate School of Public Health, University of Pittsburgh

YongSeok Park, PhD, Assistant Professor, Department of Biostatistics, Graduate School of Public Health, University of Pittsburgh

Abdus S. Wahed, PhD, Associate Professor, Department of Biostatistics, Graduate School of Public Health, University of Pittsburgh

Stewart J. Anderson, PhD, Professor, Department of Biostatistics, Graduate School of Public Health, University of Pittsburgh

Zhao Ren, PhD, Assistant Professor, Department of Statistics, Dietrich School of Arts and Sciences, University of Pittsburgh

Dissertation Director: **George C. Tseng**, ScD, Professor, Department of Biostatistics, Graduate School of Public Health, University of Pittsburgh

STATISTICAL INTEGRATIVE OMICS METHODS FOR DISEASE SUBTYPE DISCOVERY

Zhiguang Huo, PhD

University of Pittsburgh, 2017

ABSTRACT

Disease phenotyping using omics data has become a popular approach that can potentially lead to better personalized treatment. Identifying disease subtypes via unsupervised machine learning is the first step towards this goal. With the accumulation of massive high-throughput omics data sets, omics data integration becomes essential to improve statistical power and reproducibility. In this dissertation, two directions from sparse K-means method will be extended.

The first extension is a meta-analytic framework to identify novel disease subtypes when expression profiles from multiple cohorts are available. The lasso regularization and meta-analysis can identify a unique set of gene features for subtype characterization. By adding pattern matching reward function, consistency of subtype signatures across studies can be achieved.

The second extension is using integrating multi-level omics datasets by incorporating prior biological knowledge using sparse overlapping group lasso approach. An algorithm using alternating direction method of multiplier (ADMM) will be applied for fast optimization.

For both topics, simulation and real applications in breast cancer and leukemia will show the superior clustering accuracy, feature selection and functional annotation. These methods will improved statistical power, prediction accuracy and reproducibility of disease subtype discovery analysis.

Contribution to public health: The proposed methods are able to identify disease subtypes from complex multi-level or multi-cohort omics data. Disease subtype definition is essential to deliver personalized medicine, since treating different subtypes by its most appropriate medicine will achieve the most effective treatment effect and eliminate side effect. Omics data itself can provide better definition of disease subtypes than regular pathological approaches. By multi-level or multi-cohort omics data, we are able to gain statistical power and reproducibility, and the resulting subtype definition is much reliable, convincing and reproducible than single study analysis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

First and most importantly, I am very lucky to study and work with my advisor Dr. George Tseng, as he would recruit me without any statistics background. His passionate and vision toward research, enthusiasm toward life and faith, character and attitude to treat people and work have deeply influenced me and my career path. I really appreciate his guidance, encouragement and support during my Ph.D periods and I will never forget these memorable and happy times.

I want to thank my co-advisor, Dr. YongSeok Park. I appreciate his guidance and earnest advise and experience for me. It happened that I lowered down the priority of his projects but he would always encourage me to do what I mostly like to do first. I really appreciate his patience and tolerance on me.

I also want to thank all my other committee members Dr. Wahed, Dr. Anderson and Dr. Ren. They have provided me a great amount of help, suggestion, consultant and encouragement for my research and career choice.

I want to thank all my lab mates for accompanying me during my Ph.D periods. I have learned too many things from them. With their discussion, help and interaction, I always feel passionate and never feel lonely.

I would like to dedicate the work's best aspects and express my heartfelt gratitude to my parents and fiancee Shu Wang. To my parents, who cultivated me meticulously to be honest, independent and self-motivated. To my fiancee, who stands by me, arranges for me and encourages me through the happiness and suffering.

1.0 INTRODUCTION

In this section, background knowledge for my dissertation is introduced. It contains several subsections. Genomics background and techniques serves as a overview of datasets used in this dissertation (Section 1.1). Subtype discovery via transcriptomic data describes the biological motivation of this dissertation (Section 1.2). Basic bioinformatics approach to high-throughput genomic data serves as the foundation on which the proposed methods will depend on (Section 1.3). Statistical omics data integration, including meta analysis and integrative analysis, serves as the motivation for developing these methods (Section 1.4). Section 1.5 will give an overview of the dissertation structure and brief introduce the purpose of each Chapter.

1.1 VARIOUS TYPES OF OMICS DATA

Omics represents the study of genomics, proteomics or metabolomics, which are all with root -omics. Genomics aims at the collective characterization and quantification of genes, Omics aims at the collective characterization and quantification of genes, environmental effect to genes and their interactions. It involves a wide variety of genetic aspects including transcription, translation, modification, protein-protein interaction and DNA structure, gene fusion. These genomic phenomena have brought up diverse omics data types. This section will introduce these types of data at DNA, RNA, epigenetics levels. Properly integrating these types of omics data and combining data from different sources is very challenging and my dissertation will solve a couple challenges in this field.

1.1.1 Omics data at DNA level

Deoxyribonucleic acid (DNA) is a nucleic acid which carries majority of the genetic information and controls the development and replication of all living organisms. DNA molecules are consisted of double strands coiled around each other, with compromised nucleotides on each base position. Each nucleotide composes one of these nitrogen-containing nucleotides, guanine (G), adenine (A), thymine (T), or cytosine (C). DNA is a sequence of these nucleotides, which is folded as chromosomes inside nuclear. The human genome contains 46 chromosomes (23 pairs) with approximately 3 billion base pairs of DNA. Fragments of DNA could be transcripted into messenger RNA, which will form protein sequence and affect the phenotype of the organism. DNA replicates itself during cell division and the copies of sibling cells have the identical genetic information as their parent. Single nucleotide polymerphisms (SNPs) is an inheritable mutation at a single base pair among different members of a population. Common measurement of genetic variations is often referred as SNP genotyping. Many studies have shown that SNP is associated with phenotypic variation, response to environment and anthropometric behavior. This type of genetic variation is usually studied in the whole genome scale and the analysis of SNP is called genome-wide association study (GWAS). Mutation is a permanent alteration of nucleotide sequence. The resulting change of DNA is not repairable and the errors will proceed to DNA replication and RNA transcription. There are two types of major mutations: somatic mutation and germline mutation. Somatic mutation is a genetic structure variance which is not inheritable from a parent or to an offspring. It happens all the time for living organisms. Germline mutation occurs in sperm or ova, which is heritable since it is in the lineage of germ cells. This type of mutation will be transmitted to offspring. Mutation is associated with cancer since the error in RNA transcription would result in undesirable protein which could lead to cancer. Copy number variation (CNV) is a structure variation of DNA segment. It corresponds to relative long regions of DNA being altered (either duplicated or deleted). DNA is double strand so normal copy number is 2. Duplicated copy will be greater than 2 and deleted copy will be smaller than 2. It is observed to relate with diseases and also accounts for regulation of genes expression and other genomic process.

1.1.2 Omics data at RNA level

Ribonucleic acid (RNA) is another nucleic acid involved in various biological processes: coding, regulation and expression of a gene. Although DNA and RNA are all chains of nucleotides, but RNA is single strand while DNA is double strands. Cellular processes utilize messenger RNA (mRNA) to transmit genetic information and synthesize proteins. mRNA plays an important role for many biological process and the amount of mRNA (often called gene expression) is directed associated to protein formation, external phenotype, cellular pathway, disease mechanism. mRNA is also called transcriptomic data since it is transcripted from DNA. Normally the expression of mRNA is positively correlation with CNV of the same gene, since more copies of DNA tend to transcript more mRNA. But this is not necessarily true since in reality there are far more complicated biological mechanism which would prevent the mRNA to be overly expressed given the existence of more copies of DNA. Besides mRNA, there are other types of RNA, including rRNA, tRNA, snRNA, miRNA. miRNA (short for microRNA) is a small, non-coding RNA which could regulate gene expression via facilitating or silencing gene expression. miRNA can target on a group of genes (usually hundreds) and such predicted information are publicly available in several popular databases (e.g. http://www.microrna.org/microrna/home.do, http://www.broadinstitute.org/gsea/msigdb/collections.jsp).

1.1.3 Omics data of epigenetics

Epigenetics include DNA methylation, histone modification and chromatin structure change and it plays a pivotal role in gene regulation. Although monozygous twins have identical genotypes, their phenotype may be discordant, such as susceptibilities to disease and many anthropomorphic features (Fraga et al., 2005), because of the existence of epigenetic difference. DNA methylation is one of the most crucial epigenetic effects, which will help regulate the gene expression and silencing(Kulis and Esteller, 2010) and many other cellular processes, including development of embryo, changing chromatin architecture, inactivation of X chromosome, genomic imprinting and histone modification (Robertson, 2005). It is intensely studied and it has been discovered that DNA methylation is correlated with other

epigenetic effect such as histone modification and changing chromatin architecture. Cancers are related to the alteration of DNA methylation pattern. For instance, hypermethylation of tumor suppress genes will inactivate their transcriptional function and lead to loss of regular cellular function (Esteller, 2007). DNA methylation happens when addition of a methyl group (-CH_3) occurs at the fifth position of the cytosine (C). Most common DNA methylation are observed in CpG dinucleotides, where cytosine (C) is adjacent to guanine (G). Both microarray technique and NGS technique (see Section 1.1.4) allow us to measure the methylation value. Commonly the methylation value is measured in a mixture of cells given a subject. To quantify the methylation level, researchers define beta value, which represents for a CpG site, the percentage of methylated events out of all methylated and unmethylated events. As a result, beta value is between 0 and 1. Normally methylation level is negatively correlated with mRNA expression of the same gene, since the methyl group on DNA usually inhabits the transcription of mRNA.

1.1.4 Experimental techniques

1.1.4.1 Microarray Traditionally in molecular biology, researchers could only study some specific genes, which is very time consuming and expensive. DNA Microarray is a break-through technique that could address this problem that was thought as impossible. This technique enable researchers to obtain the expression of tens of thousands of genes simultaneously. This facilitates the biological community to understand the mechanism of many diseases and fundamental aspects of organ growth and development. A microarray experiment will generate DNA templates which can target specific genes and hybridize mRNA molecule onto it. And an array is consisted to a lot of DNA samples. The gene expression level will be evaluated by the amount of mRNA which bound to the DNA templates. Microarray is also called gene-chips, which has been widely used for decades and its technique has been updated constantly. It enjoys tremendous popularity for its low cost and high-throughput (measuring tens of thousands of genes at the same time).

There are three major types of microarray. The first one is expression profiling, in which mRNA or miRNA expression level could be harvested by using the hybridization

technique. The detected biomarkers could be hypothesized to be associated with cancer, other disease, treatment or response to environment. The second type is SNP array detection, in which single nucleotide polymorphism will be identified. SNP genotyping data and copy number variation data could be obtained from this type of array. By certain antibody treatment, methylation level could be detected by this type of array. The third type is ChIP-chip, which will attach antibodies to protein of interest and immunoprecipitate the DNA/protein complex. This procedure will help identify binding sites for transcriptional regulators (including transcription factors, histones and other DNA-binding proteins).

1.1.4.2 Next generation sequencing In the field, the traditional sequencing technique – Sanger sequencing has been dominating for more than 30 years. In 2001, the Human Genome Project sequenced the blue print of human genome and it greatly motivated people to explore our genetic mechanism from a sequencing perspective. However, it is not only expensive but also only able to target several specified genes. Thus, researchers are hunger for high throughput techniques. Next generation sequencing utilizes high-throughput DNA sequencing techniques: DNA sequence are smashed into fragments, which are called reads, and sequenced in parallel, yielding substantially throughput. Alignment algorithms will assemble these short reads to the reference genome. By reconstructing the whole genome, we are able to know the exact nucleotide order present in DNA and the coverage of of segment at any position. Therefore a wide variety of genomic features could be measured. Through specific locus base, we could detect SNP/indel, structural variation and somatic mutation. Through coverage, we will be able to detect copy number variation and mRNA expression. By some extra bisulfite treatment technique, sequencing can also measure methylation. Besides novel genomic feature such as isoform of mRNA and fusion genes could be detected. Nowadays, millions of fragments of DNA from a single sample can be sequenced in parallel and the entire genome can be sequenced within one day. This technique has dramatically accelerated people's understanding about human genome.

1.1.5 Databases for Omics data

With advances in biological techniques, researchers are able to measure many different types of omics data. This is often referred as high-throughput technology since relatively small-size detection equipment could generate large amount of information. Omics datasets are booming and accumulating in the past 10 years. The price to generate the data keeps dropping down and large amount of datasets are available in the public domain. Over the years large amount of omics data are accumulated in public databases and depositories; for example, The Cancer Genome Atlas (TCGA) `http://cancergenome.nih.gov`, Gene Expression Omnibus (GEO) `http://www.ncbi.nlm.nih.gov/geo/`, Sequence Read Archive (SRA) `http://www.ncbi.nlm.nih.gov/sra`, just to name a few. These datasets provide unprecedented opportunities to reveal cancer mechanisms via combining multiple cohorts or multiple-level omics data types (a.k.a. horizontal omics meta-analysis and vertical omics integrative analysis; see below) (Tseng et al., 2012). My dissertation proposes integrative and/or meta-analysis approaches to tackle this problem.

1.2 SUBTYPE DISCOVERY VIA TRANSCRIPTOMIC DATA

Many complex diseases were once thought of as a single disease but modern transcriptomic studies have revealed their disease subtypes that contain different disease mechanisms, survival outcomes and treatment responses. Representative diseases include leukemia (Golub et al., 1999), lymphoma (Rosenwald et al., 2002), glioblastoma (Parsons et al., 2008; Verhaak et al., 2010), breast cancer (Lehmann et al., 2011; Parker et al., 2009), colorectal cancer (Sadanandam et al., 2013) and ovarian cancer (Tothill et al., 2008). Taking breast cancer as an example, Perou et al. (2000) was among the first to apply gene expression profile information to identify clinically meaningful subtypes of breast cancer, such as Luminal A, Luminal B, Her2-enriched and Basal-like. Many independent studies have followed the approach on different cohorts and identified similar breast cancer subtypes (Ivshina et al., 2006; Loi et al., 2007; Sørlie et al., 2001; van't Veer et al., 2002; Wang et al., 2005). Although the breast

cancer subtype classification models have been shown to cross-validate across studies with moderately satisfying consistency (Sørlie et al., 2003), each study claims a different intrinsic gene set (i.e. the list of genes used to define subtype classification) and a different characterization of cancer subtypes (Mackay et al., 2011), making it difficult to classify new patients with confidence in clinical applications. Parker et al. (2009) combined five transcriptomic studies using pre-existing subtype classifications from each study and identified 50 genes most reproducible in the subtype classification by Prediction Analysis of Microarray (PAM) (Tibshirani et al., 2002). These signature genes (often called PAM50) have been widely followed up and validated thereafter but, from a statistical point of view, the construction of PAM50 genes was an ad hoc framework and did not fully integrate information of multiple transcriptomic studies. In a parallel line, Wirapati et al. (2008) performed meta-analysis of breast cancer subtyping based on three pre-selected genes (ER, HER2 and ERBB2) and the consequential subtypes were associated with the prior gene selections. These subtypes usually have strong clinical relevance since they show different clinical outcome, and might be responsive to different treatments (Abramson et al., 2015). However, single cohort/single omics (e.g. transcriptome) analysis suffers from sample size limitation and reproducibility issues (Simon et al., 2003; Simon, 2005; Domany, 2014). Section 1.1.5 describes large amount of genomic data are available in public domain. By integrating these datasets will increase statistical power, credibility and reproducibility.

1.3 HIGH-THROUGHPUT GENOMIC DATA ANALYSIS

In this section we will introduce several commonly used high-throughput data analysis methods. They are also the foundation for methodology development and result evaluation.

1.3.1 Differential expressed gene detection

Differential expressed gene detection is the most commonly used genomics analysis. It could describe under different environment (case vs control), which gene expression will be altered.

These detected biomarkers can be further utilized to characterize the disease and predict the patients. It is also the foundation for other downstream analysis, such as pathway enrichment analysis machine learning analysis. For continuous expression data including microarray and RNAseq FPKM data, traditional statistical methods include the student t test and the Wisconsin rank sum test for two class comparisons. Anova is often applied to multiple classes comparison and linear regression will be applied to continuous covariate. Cox proportional hazard model will be suitable for time to event data. However, there are limitations on traditional methods. For example, with t test, sometimes features with small effect size could also be chosen because of low variance, through they are not of biological interest. Advanced method includes SAM (Tusher et al., 2001), LIMMA (Smyth, 2005) which will partially account for these limitations. For RNAseq count data, edgeR(Robinson et al., 2010) is a popular tool to detect DE genes. Meanwhile, other omics datatype may have their own hypothesis. For methylation, people will detect differential methylation using t test (Hansen et al., 2012) or logistic regression model(Akalin et al., 2012). However these methods don't fully account for the design and mechanism of methylation. The Beta binomial model (Park et al., 2014) will fully characterize these properties. For each of these data setting, permutation test is also a very powerful approach to detect DE genes. However, it suffers from heavy computing.

Another problem in differential expressed genes detection is multiple comparison. In genomics setting, we have at least tens of thousands of features. Assuming there is no DE signal and all genes are from null. In this case, the p values will be uniformly distributed between 0 and 1. By chance we will get very signifiant p values. Therefore, we couldn't simply use the standard 0.05 p value threshold. To address this problem, there are two multiple comparison control methods. The first one is family-wise error rate (FWER) (Hochberg and Tamhane, 2009), which gives the probability of at least one false positive. In genomic setting, it is often too stringent to control FWER. Another commonly used measurement is false discovery rate (FDR) (Benjamini and Hochberg, 1995). This indicates within our discoveries, what the percentage of false positives is. These two multiple comparison procedures are widely applied in genetics and genomics studies.

1.3.2 Pathway enrichment analysis

In genomics analysis, we could obtain a list of genes that are related to disease or treatment. These genes could be differential expression genes (from DE analysis) or co-expression genes (from cluster analysis). Pathway enrichment analysis is to pursue a functional annotation for the outcome gene list. A pathway database is a collection of genes, which are known to be associated with specific biological states, chemical perturbations or other environmental or treatment factors. A lot commonly used pathway databases can be obtained from public domain. To list a few, Gene Ontology (GO) http://geneontology.org, KEGG http://www.genome.jp/kegg/, Biocarta http://www.biocarta.com, Reactome http://www.reactome.org, MSigDB http://www.broadinstitute.org/gsea/msigdb/index.jsp. Pathway enrichment result could serve as a validating purpose if the pathway result is highly associated with the experimental setting,

Several hypothesis testing methods can be used to examine the association between the experimental outcome gene list and a pathway (a list of pathway genes). Fisher's exact tests or Chi-square tests could test this association very well. For these two tests we only need to use the outcome gene list and don't need the significance score (p value of each gene). We could construct a 2×2 table by two conditions: whether the gene is in outcome gene list and whether the gene is in the pathway database. The null hypothesis is the proportion of outcome genes inside a pathway is independent of the proportion of outcome genes outside a pathway. Fisher's exact tests or Chi-square tests can illustrate how significant a outcome gene proportion is different between inside or outside pathways. Another approach is to utilize a Kolmogorov-Smirnov test (KS test). For this test, we need to use the gene list and significance score. The null hypothesis is the distribution of significance score of all genes inside a pathway is same as the distribution of significance score of all genes outside a pathway. The KS test could could tell how significant there is a significance score difference between inside or outside pathways. Pathway enrichment analysis will be used to examine whether the detected co-expression gene list or DE gene list is biological meaningful in result sections of later chapters.

1.3.3 Transcriptomic clustering analysis

1.3.3.1 General clustering analysis algorithms Unsupervised machine learning aims to group a set of objects into clusters without the prior knowledge of class labels. It is widely used in genomics research and other machine learning field. Clustering analysis could help to discover disease subtypes, which could better characterize the disease property and guide to precision medicine. In the literature, hierarchical clustering (Ward Jr, 1963) generates clustering result at each level of hierarchy by combining clusters result of the next lower level. K-means (MacQueen et al., 1967) is popular due to its simplicity and fast computing. It aims to minimize the within cluster sum of square by iteratively optimize its cluster assignment and cluster labels. Self organized map (SOM)(Kohonen, 1998) generates clustering diagram by mapping a high-dimensional distribution to a low-dimensional grid. Mean shift clustering (Cheng, 1995) performs clustering by seeking for modes through non-parametric iteration. In terms of transcriptomic clustering analysis, popular methods include hierarchical clustering (Eisen et al., 1998), K-means (Dudoit and Fridlyand, 2002), mixture model-based approaches (Xie et al., 2008; McLachlan et al., 2002) and non-parametric approaches (Qin, 2006), for analysis of single transcriptomic study. Resampling and ensemble methods have been used to improve stability of the clustering analysis (Kim et al., 2009; Swift et al., 2004) or to pursue tight clusters by leaving scattered samples that are different from major clusters (Tseng, 2007; Tseng and Wong, 2005; Maitra and Ramler, 2009). Witten and Tibshirani (2010) proposed a sparse K-means algorithm that can effectively select gene features and perform sample clustering simultaneously.

1.3.3.2 K-means and sparse K-means K-means algorithm (Hartigan and Wong, 1979) has been a popular clustering method due to its simplicity and fast computation. Consider X_{jl} the gene expression intensity of gene j and sample l. The method aims to minimize the within-cluster sum of squares (WCSS):

$$\min_{C} \sum_{j=1}^{p} WCSS_j(C) = \min_{C} \sum_{j=1}^{p} \sum_{k=1}^{K} \frac{1}{n_k} \sum_{l,m \in C_k} d_{lm,j} \tag{1.3.1}$$

where p is the number of genes (features), K is the number of clusters, $C = (C_1, C_2, \ldots, C_K)$ denotes the clustering result containing partitions of all samples into K clusters, n_k is the number of samples in cluster k and $d_{lm,j} = (X_{jl} - X_{jm})^2$ denotes the squared Euclidean distance of gene j between sample l and m. Although the initial development of K-means was a heuristic algorithm, it was shown to be a special classification likelihood method in model-based clustering when data from each cluster come from Gaussian distribution with identical and spherical covariance structure (Tseng, 2007).

One major drawback of K-means is that it utilizes all p features with equal weights in the distance calculation. In genomic applications, p is usually high but biologically only a small subset of genes should contribute to the sample clustering. Witten and Tibshirani (2010) proposed a sparse K-means approach with lasso regularization on gene-specific weights to tackle this problem. One significant contribution of their sparse approach was the observation that direct application of lasso regularization to Equation 1.3.1 will result in a meaningless null solution. Instead, they utilized the fact that minimizing $WCSS$ is equivalent to maximizing between-cluster sum of squares ($BCSS$) since $WCSS$ and $BCSS$ add up to a constant value of total sum of squares ($TSS_j = BCSS_j(C) + WCSS_j(C)$). The optimization in Equation 1.3.1 is equivalent to

$$\max_C \sum_{j=1}^{p} BCSS_j(C) = \max_C \sum_{j=1}^{p} \left[\frac{1}{n} \sum_{l,m} d_{lm,j} - \sum_{k=1}^{K} \frac{1}{n_k} \sum_{l,m \in C_k} d_{lm,j} \right] \tag{1.3.2}$$

The lasso regularization on gene-specific weights in Equation 1.3.2 gives the following sparse K-means objective function:

$$\begin{aligned} &\max_{C,\mathbf{w}} \sum_{j=1}^{p} w_j BCSS_j(C) \\ &\text{subject to } \|\mathbf{w}\|_2 \leq 1, \|\mathbf{w}\|_1 \leq \mu, w_j \geq 0, \forall j, \end{aligned} \tag{1.3.3}$$

where w_j denotes weight for gene j, $C = (C_1, \ldots, C_K)$ is the clustering result, K is the pre-estimated number of clusters and $\|\mathbf{w}\|_1$ and $\|\mathbf{w}\|_2$ are the l_1 and l_2 norms of the weight vector $\mathbf{w} = (w_1, \ldots, w_p)$. The regularization shrinks most gene weights to zero and μ is a tuning parameter to control the number of non-zero weights (i.e. the number of intrinsic genes for subtype characterization).

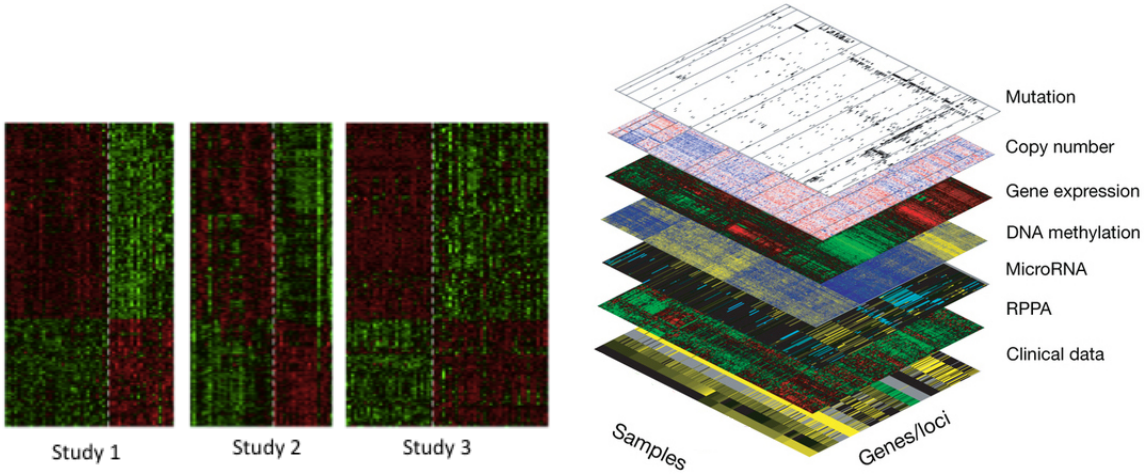
These methods serve as the background for Chapter 2 and Chapter 3, aim to develop methodology to discover disease subtypes from omics data.

1.4 STATISTICAL DATA INTEGRATION

As described in Section 1.1.5, large amounts of omics data are accumulating and become publicly available. How to combine these dataset to strengthen statistical analysis becomes a natural question. Reproducibility problem is also emphasized in the literature. Ioannidis et al. (2008) evaluated the replication of 18 microarray-based gene expression analyses, but it turned out reproducibility was low. Integration of different omics data type and/or different cohorts will help improve reproducible and draw robust inference. To extend single-study techniques towards integration of multiple omics data sets, Tseng et al. (2012) categorized omics data integration into two major types: (A) Horizontal omics meta-analysis and (B) Vertical omics integrative analysis. Figure 1 has shown two directions for data integration including horizontal genomics meta analysis and vertical genomic integrative analysis. For horizontal meta-analysis (Figure 1(a)), multiple studies of the same omics data type (e.g. transcriptome) from different cohorts are combined to increase sample size and statistical power, a strategy often used in differential expression analysis (Ramasamy et al., 2008), pathway analysis (Shen and Tseng, 2010), network analysis (Zhu et al., 2016) or subtype discovery (Huo et al., 2016). In contrast, vertical integrative analysis (Figure 1(b)) aims to integrate multi-level omics data from the same patient cohort (e.g. gene expression data, genome-wide profiling of somatic mutation, DNA copy number, DNA methylation, or microRNA expression from the same set of biological samples (Richardson et al., 2016)).

1.4.1 Horizontal meta analysis

Horizontal genomics meta analysis aims to combine multiple transcriptomic studies or other omics data type. Most methods have been developed to improve differential analysis (candidate marker detection)(Chang et al., 2013) and pathway analysis(Wang et al., 2012). Hor-



(a) Horizontal meta-analysis

(b) Vertical integration

Figure 1: **Background for omics data integration.**
Meta analysis omics data integration (Horizontal direction) and integrative omics data integration (Vertical direction)

izontal genomics meta analysis of differential expressed genes includes method combining p value: Fisher(Fisher, 1925), Stoffer(Stouffer, 1949), maxP, minP, roP(Song and Tseng, 2014), AW(Li et al., 2011); other approach to combine effect sizes, or Bayesian approach(Scharpf et al., 2009). Horizontal genomics meta analysis could increase statistical power and over-come the difficulty that signal in single cohort is weak and not that reproducible.

As high-throughput experiments become affordable and prevalent, many data sets of the same omic type (e.g. transcriptome) and of a related disease hypothesis have often been col-lected and meta-analyzed, as is described in Section 1.4. The meta-analysis to combine mul-tiple studies has brought new statistical challenges. When multiple transcriptomic studies are combined, most methods have been developed to improve differential analysis (candidate marker detection) and pathway analysis (See Section 1.4). These methods mostly extend from traditional meta-analysis by combining effect sizes or p-values of multiple studies to a genome-wide scale (see review papers for microarray and GWAS meta-analysis by (Tseng et al., 2012; Begum et al., 2012) for details).

1.4.2 Vertical integrative analysis

Vertical genomic integrative analysis aims to integrate multiple types of omics data from a given cohort. The input omics data source include but not limit to transcriptome profile, genotypes, DNA copy number alteration, DNA methylation, microRNA and proteinomics. For instance, IBAG (Wang et al., 2013) uses model-based integration approach, combining data obtained from multiple platforms into one integrative model to discover clinically relevant biomarker and/or to predict clinical outcome. Icluster (Shen et al., 2009) uses dimension reduction and latent variable techniques to combine different omics data and find disease subtypes. Vertical genomic integrative analysis will make the result more consistent within cohort and biological meaningful. Meanwhile by recruiting different sources of omics data, statistical power will increase and the conclusion will be more convincing.

1.5 OVERVIEW OF THE DISSERTATION

My dissertation contains six chapters. Chapter 1 contains overall introduction of datasets, experimental techniques, high through-put analysis methods, motivation of genomic integrative analysis. These contents serve as the background knowledge for the methodology development for Chapter 2, 3.

Chapter 2 is Meta-analytic framework for sparse K-means to identify disease subtypes in multiple transcriptomic studies. This is a meta-analysis framework for disease subtyping combining multiple omics cohorts. Intuitively, this method is better than single study sparse Kmeans since meta analysis will achieve unified feature selection; increase credibility of inference by recruiting more studies. It has also been shown to have better performance, including clustering accuracy in simulation and resampling accuracy and stability in real data application than single study sparse clustering. The content in this Chapter has been published in Journal of the American Statistical Association (Huo et al., 2016).

Chapter 3 is integrative sparse Kmeans to identify disease subtypes form multiple omics data types of the same patient cohort. This integrative approach via combining multiple

omics data types will better characterize the disease subtypes. We also consider external group information, which makes the selected for the disease subtype more meaningful. Performance will be evaluated in simulation and real data comparing to sparse Kmeans without group, or other integrative approach such as iCluster. The content of this chapter is accepted by the Annals of Applied Statistics (Huo and Tseng, 2017).

Chapter 4 is discussion and future work. The future work is about two way integration to identify disease subtypes form multiple omics data types from multiple patient cohorts. This is an extension from both Meta Sparse Kmeans and integrative Sparse Kmeans approach. It combines all the benefits from Chapter 2 and Chapter 3. by combining multiple omics data types and multiple cohorts and incorporating external group information, feature selection and clustering matching. This method would give a comprehensive representation of the clustering result, which will lead to the most convincing result.

2.0 META SPARSE KMEANS

2.1 INTRODUCTION

In section 1.2, we introduced the background for disease subtype discovery via transcriptomic data. With the accumulation of transcriptomic data in public domain, we will gain statistical power and reproducibility by combining multiple studies. In section 1.3.3.1, we introduced popular transcriptomic clustering method for single study. But when it comes to disease subtype discovery, no integrative method for combining multiple transcriptomic studies is available, to the best of our knowledge. In this chapter, we propose a Meta-analytic sparse K-means method (MetaSparse$K$means) (Huo et al., 2016) for combining multiple transcriptomic studies, which identifies disease subtypes and associated gene signatures, and constructs prediction models to classify future new patients. The method contains embedded normalization and scaling to account for potential batch effects from different array platforms and a multi-class correlations (MCC) measure (Lu et al., 2010) to account for different sample proportions of the disease subtypes across studies. A pattern matching reward function is included in the objective function to guarantee consistency of subtype patterns across studies. We will demonstrate improved performance of MetaSparse$K$means by simulations and two real examples in leukemia and breast cancer studies. The content in this Chapter has been published in Journal of the American Statistical Association (Huo et al., 2016).

This chapter is structured as the following. In Section 2.2, we will demonstrate a motivating example to combine three large breast cancer transcriptomic studies for disease subtype discovery. We will describe the input data structure, problem setting and the biological goals to motivate the development of MetaSparse$K$means. In Section 2.3, introduction of classical K-means, sparse K-means and development of MetaSparse$K$means are presented.

Section 3.4 contains simulation results and applications to real data in breast cancer and leukemia. Finally, conclusions and discussions are included in Section 3.5.

2.2 MOTIVATING EXAMPLE

Table 1 shows a summary description of three breast cancer training transcriptomic studies: Wang (Wang et al., 2005), Desmedt (Desmedt et al., 2007) and TCGA (Network, 2012) as well as one testing study METABRIC (Curtis et al., 2012) with large sample size (n=1981) and survival information. In the training set, each study contains about 150-500 samples. Wang and Desmedt applied Affymetrix U133A chip that generated log-intensities ranging between 2.104 and 14.389, while TCGA adopted Agilent Custom 244K array that produced log-ratio intensities ranging between -13.816 and 14.207. All probes in three studies were matched to gene symbols before meta-analysis. When multiple probes matched to one gene symbol, the probe that with the largest inter-quartile range (IQR) was used (Gentleman et al., 2005). 11,058 genes were matched across studies and three gene expression matrices (11,058×260, 11,058×164 and 11,058×533) were used as input data for disease subtype discovery. In such a meta-analysis framework of sample clustering analysis, we pursue two goals simultaneously: identification of a gene set (often called "intrinsic gene set") for subtype characterization and clustering of samples in each study. Five major analytical issues (or procedures) have to be considered in the new meta-analytic framework: (A) combine information from multiple studies and perform feature (gene) selection; (B) use the combined information to perform clustering on each study; (C) accommodate potential batch effect across studies and the fact that each study contains different mixture proportions of the subtypes. (e.g. study 1 contains 20% of the first subtype while study 2 contains 35%); (D) guarantee that subtypes across studies can be matched with consistent gene signature and pattern; (E) construct a prediction model based on the combined analysis to predict future patients. In the following method section, we will develop a MetaSparse$K$means method to answer all five issues described above. Figure 2(d)-2(f) illustrate the heatmap result of our developed method on the motivating example (details will be discussed in the Result Sec-

Table 1: Breast Cancer Data information

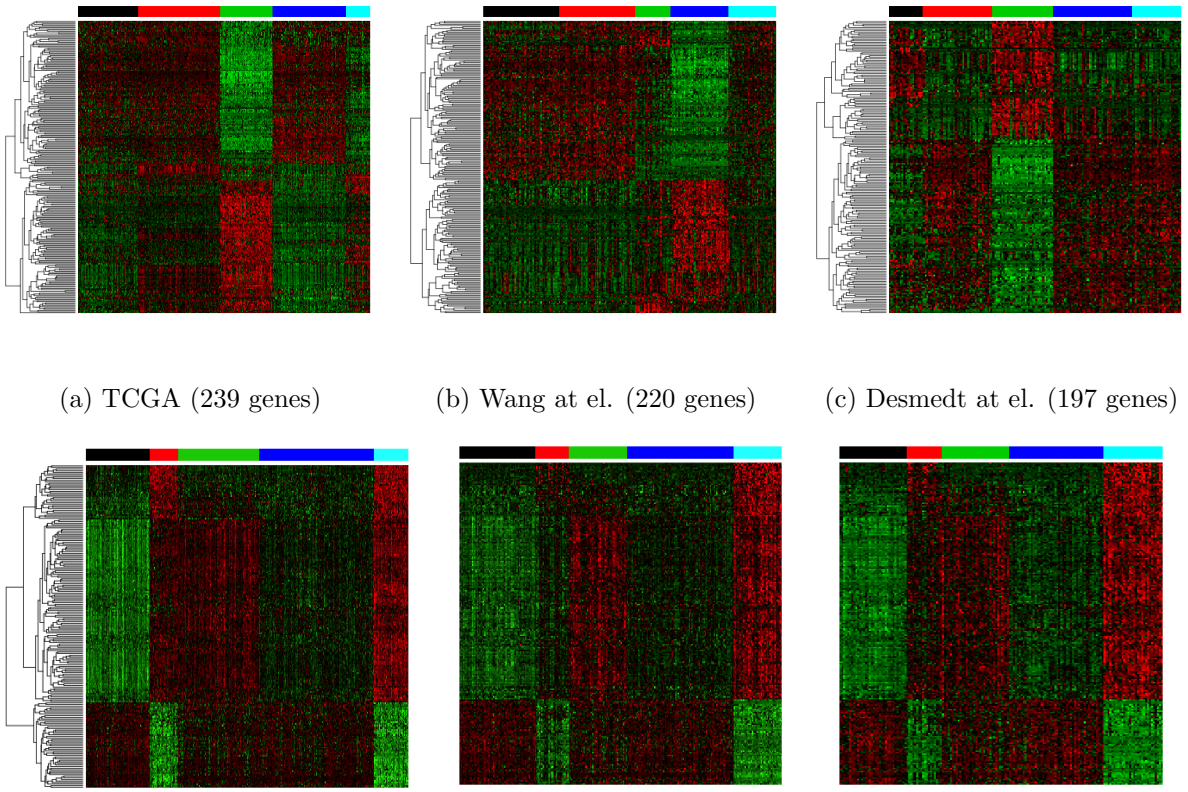
	Training			Testing
Study Name	TCGA	Wang at el.	Desmedt at el.	METABRIC
Platform	Agilent	Affymetrix	Affymetrix	Illumina
Number of genes	17,814	12,704	12,704	19,602
Number of patients	533	260	164	1,981
Range of intensity	$[-13.816, 14.207]$	$[3.085, 14.389]$	$[2.104, 14.160]$	[-1.262 16.618]
Mean intensity	0.003	6.797	5.523	6.954
Standard deviation	1.34	1.71	1.84	1.70

tion 2.4.3). 203 genes (on the rows of heatmaps) were simultaneously selected to characterize the disease subtypes. Clustering results were shown on the color bars above the heatmaps. The expression patterns of the five disease subtypes were matched well across studies from visual inspection in the heatmaps and a classification model was constructed to predict future patients. In contrast, Figure 2(a)-2(c) show sparse K-means clustering results when applied to each study separately. Each study generates different gene selection (220, 197, 239 genes respectively) and cluster patterns that are difficult to be integrated to predict a future patient. Throughout this chapter, we will develop and illustrate the method for combining multiple transcriptomics studies, but the method is also applicable to meta-analysis of other types of omics data, such as miRNA, methylation or copy number variation.

2.3 METHOD

2.3.1 MetaSparse$K$means

We have introduced K-means and sparse K-means in Section 1.3.3.2. Equation 1.3.3 identifies gene features and performs sample clustering simultaneously for a given transcriptomic study. To extend it for combining $S(S \geq 2)$ transciptomic studies, a naive solution is to



(a) TCGA (239 genes) (b) Wang at el. (220 genes) (c) Desmedt at el. (197 genes)

(d) TCGA (203 genes) (e) Wang at el. (203 genes) (f) Desmedt at el. (203 genes)

Figure 2: Individual study clustering and MetaSparse$K$means result for 3 breast cancer datasets.

Rows represent genes and columns represent samples. Red and green color represent higher and lower expression. In each study, the patients are divided into 5 clusters, represented by 5 unique colors in the color bar above the heatmaps. 2(a)-2(c): Sparse K-means result from three studies separately. 2(d)-2(f): MetaSparse$K$means result.

consider optimization of the sum over S studies:

$$\begin{aligned} &\arg\max_{C^{(s)},\mathbf{z}} \sum_{j=1}^{p} z_j \times \sum_{s=1}^{S} BCSS_j^{(s)}(C^{(s)}) \\ &\text{subject to } \|\mathbf{z}\|_2 \le 1, \|\mathbf{z}\|_1 \le \mu, z_j \ge 0, \forall j. \end{aligned} \tag{2.3.1}$$

where superscript of (s) in $BCSS^{(s)}$ and $C^{(s)}$ denotes the $BCSS$ and clustering in study s ($1 \le s \le S$). A notable feature of Equation 2.3.1 is that the weights z_j are identical across all studies and thus it generates a common intrinsic gene set together with clustering of samples in each study $C^{(s)} = (C_1^{(s)}, \cdots, C_{K_s}^{(s)})$ (K_s is the number of clusters in study s). In this chapter, K_s is assumed to be equal to K (equal number of clusters across studies) and its extension is discussed later. A downside for Equation 2.3.1 is that it treats all studies equally without considering that different studies may contain different sample sizes and intensity ranges as shown in Table 1. As a result, studies with larger sample sizes and higher intensity variability ranges will dominate the analysis in Equation 2.3.1. To fix this problem, we propose to standardize $BCSS$ score by TSS below:

$$\begin{aligned} &\arg\max_{C^{(s)},\mathbf{z}} \sum_{j=1}^{p} z_j \times \sum_{s=1}^{S} \frac{1}{S} \frac{BCSS_j^{(s)}(C^{(s)})}{TSS_j^{(s)}} \\ &\text{subject to } \|\mathbf{z}\|_2 \le 1, \|\mathbf{z}\|_1 \le \mu, z_j \ge 0, \forall j. \end{aligned} \tag{2.3.2}$$

Note that the standardized $BCSS$ score in each study is always bounded between 0 and 1. The formulation so far answers issues (A)-(C) in Section 2.2 by generating a common intrinsic gene set, clustering samples in each study and accommodating different sample sizes and intensity ranges among studies. In Equation 2.3.2, the contribution of $BCSS/TSS$ is equal from each study and is not adjusted by sample size (denoted as equal weight or EW). Alternative option is to replace the $1/S$ term with $n_s/\sum_s n_s$ (n_s is the sample size of study s) so that studies with larger sample size contribute greater in the clustering formation (denoted by unequal weight or UW). In the simulation section (Figure 5), EW and UW are compared. Conceptually, when studies are homogeneous, UW performs better by accounting

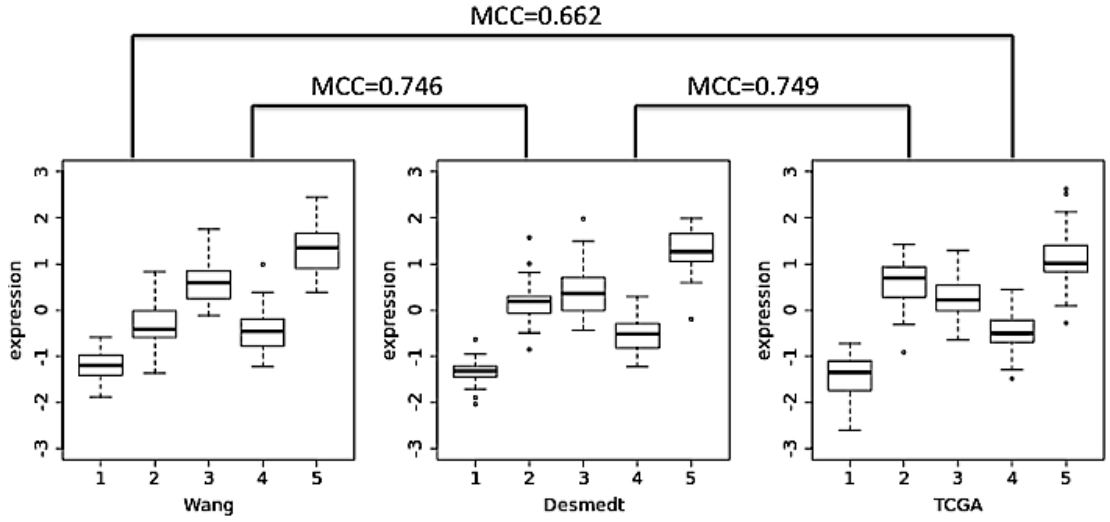
for sample size. But when studies contain heterogeneous information, EW is expected to be more robust and will be recommended in real applications.

A next issue in this meta-analytic framework is to match the cluster patterns obtained from different studies (issue (D) in Section 2.2). For example, samples of the light blue cluster in all three studies in Figure 2(d)-2(f) are up-regulated (red) in the upper part of genes and down-regulated (green) in the lower part of genes. Equation 2.3.2 guarantees to generate sample clusters with good separability in each study but does not warrant such subtype matching across studies. To achieve this purpose, we added pattern matching reward function $f_j^{match}(M)$ in the objective function:

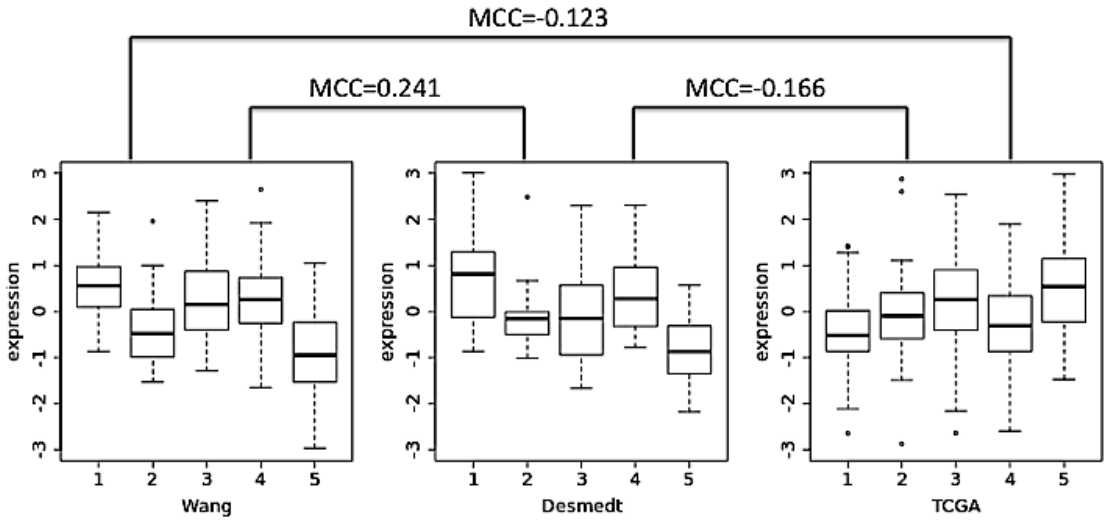
$$\begin{aligned} &\max_{C^{(s)},\mathbf{z},M} \sum_{j=1}^{p} z_j \times \left[\frac{1}{S} \sum_{s=1}^{S} \frac{BCSS_j^{(s)}(C^{(s)}(K))}{TSS_j^{(s)}} + \lambda \times f_j^{match}(M) \right] \\ &\text{subject to } \|\mathbf{z}\|_2 \leq 1, \|\mathbf{z}\|_1 \leq \mu, z_j \geq 0, \forall j, \end{aligned} \tag{2.3.3}$$

where M is the cluster matching enumeration across S studies, $M = M(C^{(1)}, \ldots, C^{(S)})$. For example, when $S = 3$ and $K = 3$, denote $(C_1^{(1)} - C_3^{(2)} - C_1^{(3)}, C_2^{(1)} - C_2^{(2)} - C_3^{(3)}, C_3^{(1)} - C_1^{(2)} - C_2^{(3)})$ as a possible matching function of M, where the first cluster in study 1, the third cluster in study 2 and the first cluster in study 3 are matched with similar gene expression pattern to represent the first disease subtype. Similarly, patients in the second clusters in study 1, second cluster in study 2 and third cluster in study 3 form the second disease subtype and so on. Under this notation, the total number of possible pattern matching of M is $(K!)^{(S-1)}$. M can be regarded as a cluster label reordering operator for all S studies: $M = (\phi^{(1)}(C^{(1)}), \phi^{(2)}(C^{(2)}), \ldots, \phi^{(S)}(C^{(S)}))$, where $\phi^{(s)}(C^{(s)})$ maps the K clusters in the s^{th} study $C^{(s)} = (C_1^{(s)}, C_2^{(s)}, \ldots, C_K^{(s)})$ to disease subtype $1, 2, \ldots, K$. In the example above, the corresponding mapping is $\phi^{(1)}(C_1^{(1)}, C_2^{(1)}, C_3^{(1)}) = (1, 2, 3)$, $\phi^{(2)}(C_1^{(2)}, C_2^{(2)}, C_3^{(2)}) = (3, 2, 1)$, $\phi^{(3)}(C_1^{(3)}, C_2^{(3)}, C_3^{(3)}) = (1, 3, 2)$.

The pattern matching reward function $f_j^{match}(M)$ borrows the concept from multi-class correlation (MCC) (Lu et al., 2010) that was developed to quantify concordant multi-class (more than two classes) expression pattern for candidate marker detection in the meta-analysis of multiple transcriptomic studies. Traditionally, one can calculate the Pearson



(a) gene CENPA with similar pattern in all studies



(b) gene TUBGCP4 with discordant pattern in different studies

Figure 3: Two real gene examples to show the idea of MCC.

The x axis is the cluster index and y axis is the expression intensity. Gene CENPA shows similar patterns across studies and MCCs are large (Figure 3(a)). Gene TUBGCP4 shows discordant patterns across studies and MCCs are smaller (Figure 3(b)).

correlation of two vectors with equal lengths. However, our pattern matching score needs to consider the correlation of identical number of clusters with unequal number of samples in each cluster. For example, Figure 3(a) shows the expression pattern of a given gene CENPA in the three breast cancer studies, each with 5 clusters of samples. All studies have relatively high expression in cluster 5, intermediate expression level in cluster 2 and 3, and lower expression in cluster 1 and 4. This is our desired concordant pattern gene which would generate high total MCC scores. Figure 3(b) shows a gene with different cluster patterns in different studies. In Wang the pattern is higher expression in cluster 1, 3 and 4, and lower expression in cluster 2 and 5. The TCGA study, however, does not have a clear pattern. Desmedt is somewhat similar to Wang but very different from TCGA. Since the patterns are not consistent across studies, the total MCC scores in this case should be lower.

Below we describe the MCC score definition from the empirical distributions of each cluster in a pair of studies study (See Lu et al. (2010) for more details). Consider $D_X = \{x_{ki}\}$ $(1 \le k \le K, 1 \le i \le n_k)$ to represent expression intensity of class k and sample i for the first study and $D_Y = \{y_{kj}\}$ $(1 \le k \le K, 1 \le j \le m_k)$ for the second study, where n_k and m_k are the number of samples of class k in the first and second studies. We first define an imaginary bivariate distribution $(\mathbb{X}, \mathbb{Y})$ that is a mixture of the K independent bivariate distributions $(X_1, Y_1), \ldots, (X_K, Y_K)$ with equal probability where X_k and Y_k are empirical distributions from $\{x_{k1}, \ldots, x_{kn_k}\}$ and $\{y_{k1}, \ldots, y_{km_k}\}$ (i.e. the CDF of $(\mathbb{X}, \mathbb{Y})$ is $G_{\mathbb{X},\mathbb{Y}}(x, y) = \frac{1}{K}\sum_{k=1}^{K} G_{X_k,Y_k}(x, y) = \frac{1}{K}\sum_{k=1}^{K} G_{X_k}(x)G_{Y_k}(y)$). MCC score is defined as the Pearson correlation of X and Y as shown below

$$MCC(D_X, D_Y) = \mathrm{cor}(\mathbb{X}, \mathbb{Y}) = \frac{(\sum_{k=1}^{K} \mu_{X_k}\mu_{Y_k}) - K\bar{\mu}_X\bar{\mu}_Y}{\sqrt{\left[\sum_{k=1}^{K} \sigma_{X_k}^2 + \sum_{k=1}^{K} \left(\mu_{X_k} - \bar{\mu}_X\right)^2\right]\left[\sum_{k=1}^{K} \sigma_{Y_k}^2 + \sum_{k=1}^{K} \left(\mu_{Y_k} - \bar{\mu}_Y\right)^2\right]}}$$

,where $\mu_{X_k} = \sum_{i=1}^{n_k} x_{ki}/n_k$, $\mu_{Y_k} = \sum_{j=1}^{m_k} y_{kj}/m_k$, $\sigma_{X_k}^2 = \sum_{i=1}^{n_k} \left(x_{ki} - \mu_{X_k}\right)^2/n_k$, $\sigma_{Y_k}^2 = \sum_{j=1}^{m_k} \left(y_{kj} - \mu_{Y_k}\right)^2/m_k$, $\bar{\mu}_X = \sum_{k=1}^{K} n_k\mu_{X_k}/\sum_{k=1}^{K} n_k$, $\bar{\mu}_Y = \sum_{k=1}^{K} m_k\mu_{Y_k}/\sum_{k=1}^{K} m_k$.

It is worth noting that MCC is defined from conventional Pearson correlation and is restricted between -1 and 1. When $n_1 = \ldots = n_k = n$ and $m_1 = \ldots = m_k = m$, MCC

reduces to

$$MCC = \frac{r_{\vec{\mu}_X \vec{\mu}_Y}}{\sqrt{\frac{1}{F_X} \cdot \frac{K}{K-1} + 1}\sqrt{\frac{1}{F_Y} \cdot \frac{K}{K-1} + 1}}$$

,where $r_{\vec{\mu}_X \vec{\mu}_Y} = \frac{\sum_k (\mu_{X_k} - \bar{\mu}_X)(\mu_{Y_k} - \bar{\mu}_Y)}{\sqrt{\sum_k (\mu_{X_k} - \bar{\mu}_X)^2}\sqrt{\sum_k (\mu_{Y_k} - \bar{\mu}_Y)^2}}$ is the sample correlation of $\vec{\mu}_X = (\mu_{X_1}, \ldots, \mu_{X_k})$ and $\vec{\mu}_Y = (\mu_{Y_1}, \ldots, \mu_{Y_k})$. $F_X = \frac{\sum_k (\mu_{X_k} - \bar{\mu}_X)^2/(K-1)}{\sum_k \sum_i (x_{ki} - \mu_{X_k})^2/\big((n-1)K\big)}$ and $F_Y = \frac{\sum_k (\mu_{Y_k} - \bar{\mu}_Y)^2/(K-1)}{\sum_k \sum_j (y_{kj} - \mu_{Y_k})^2/\big((m-1)K\big)}$ are exactly the F-statistics in ANOVA for D_X and D_Y. When the within-class variation is much smaller than the between-class variation, F_X and F_Y become large. MCC converges to $r_{\vec{\mu}_X \vec{\mu}_Y}$ as expected.

Finally, the pattern matching reward function is defined as the average of MCC of all pairs of studies as below:

$$f_j^{match}(M) = \left(\frac{1}{\binom{S}{2}} \sum_{s,s' \in S} MCC_j(\phi^{(s)}(C^{(s)}), \phi^{(s')}(C^{(s')})) + 1 \right) / 2$$

where s and s' denote any two studies from all S studies and $\phi^{(s)}(C^{(s)})$ was previously defined for cluster matching function M. Note that the pattern matching reward function is transformed to guarantee taking values between 0 and 1.

In summary, the objective function of MetaSparse$K$means in Equation 2.3.3 generates a common feature set from the non-zero estimated weights and sample clustering in each study. The first term in Equation 2.3.3 ensures good cluster separation in each study, the second term guarantees the consistent patterns of identified disease subtypes across studies and the l_1 penalty generates sparsity on gene weights to facilitate feature selection.

2.3.2 Implementation of MetaSparse$K$means

In this subsection, we discuss the optimization procedure, parameter estimation and how the classification model from the clustering can predict a future patients cohort.

2.3.2.1 Optimization without pattern matching reward function For clarity of demonstration, we first illustrate the optimization procedure without reward function as shown in Equation 2.3.2. The algorithm is a simple extension from Witten and Tibshirani (2010).

1. Initialize $\mathbf{z}$ such that $z_j = \frac{sd_j}{sd_1+...+sd_p} \times \mu$, where sd_j is the standard deviation of gene j.

2. Fix $\mathbf{z}$, update $C^{(s)}$ for study s ($\forall s \in S$) by optimizing Equation 2.3.2 applying conventional weighted K-means.

3. Fix $C^{(s)}$, update $\mathbf{z}$ by optimizing Equation 2.3.2 following Karush-Kuhn-Tucker (KKT) condition.

4. Iterate Step 2-3 until converge.

In Step 1, we apply unequal initialization weight that is proportional to the standard deviation of each gene. We have found better performance of this initialization compared to equal weight initialization suggested in (Witten and Tibshirani, 2010). In Step 2, since the weights are fixed and $TSS_j^{(s)}$ is irrelevant to the clustering result, the optimization is essentially to repeat regular K-means algorithm with weighted gene structure for each study independently. In Step 3, fixing $a_j = \sum_{s=1}^{S} \frac{BCSS_j^{(s)}(C^{(s)}(K))}{TSS_j^{(s)}}$, optimization of weights $\mathbf{z}$ is a convex optimization problem that leads to $z_j = \frac{\Gamma_\Delta(a_j)}{\|\Gamma_\Delta(a_j)\|_2}$ following KKT condition, where Γ is the soft-thresholding operator which is defined as $\Gamma_\Delta(x) = \max(x - \Delta, 0)$. $\Delta > 0$ is chosen such that $\|\mathbf{z}\|_1 = \mu$; otherwise $\Delta = 0$ if $\|\mathbf{z}\|_1 < \mu$. Readers may refer to (Boyd and Vandenberghe, 2004; Witten and Tibshirani, 2010) for more details. Finally, Steps 2 and 3 are iterated until convergence of the weight estimate (i.e. $\frac{\sum_{j=1}^{p} |z_j^{(r)} - z_j^{(r-1)}|}{\sum_{j=1}^{p} |z_j^{(r-1)}|} < 10^{-4}$), where $z_j^{(r)}$ represents the z_j estimate in the r^{th} iteration. In our simulation and real data experiences, the algorithm usually converges within 20 iterations.

2.3.2.2 Optimization with pattern matching reward function When the pattern matching reward function is added, the iterative optimization has an additional step to estimate the best clustering matching across studies M. In this case we split optimization of

Equation 2.3.3 into 3 parts:

$$C^{(s)+} = \arg\max_{C^{(s)}} \sum_{j=1}^{p} z_j \times \left[\frac{1}{S} \sum_{s=1}^{S} \frac{BCSS_j^{(s)}(C^{(s)})}{TSS_j^{(s)}} \right] \tag{2.3.4a}$$

$$M^{+} = \arg\max_{M} \sum_{j=1}^{p} z_j \times f_j^{match}(M) \tag{2.3.4b}$$

$$\mathbf{z}^{+} = \arg\max_{\mathbf{z}} \sum_{j=1}^{p} z_j \times \left[\frac{1}{S} \sum_{s=1}^{S} \frac{BCSS_j^{(s)}(C^{(s)})}{TSS_j^{(s)}} + \lambda \times f_j^{match}(M) \right]$$

$$\text{subject to } \|\mathbf{z}\|_2 \leq 1, \|\mathbf{z}\|_1 \leq \mu, z_j \geq 0, \forall j, \tag{2.3.4c}$$

where $C^{(s)+}, M^{+}, \mathbf{z}^{+}$ are the updating rule in the iteration. The optimization algorithm becomes:

1. Initialize $\mathbf{z}$ such that $z_j = \frac{sd_j}{sd_1+...+sd_p} \times \mu$, where sd_j is the standard deviation of gene j.
2. Fix $\mathbf{z}$, for $\forall s \in S$, update $C^{(s)}$ by weighted K-means according to Equation 2.3.4a.
3. Fix $\mathbf{z}$ and $C^{(s)}$, update M by using exhaustive search or simulated annealing (see below) according to Equation 2.3.4b.
4. Fix $C^{(s)}$ and M, update $\mathbf{z}$ by KKT condition according to Equation 2.3.4c.
5. Iterate Step 2-4 until converge.

One potential concern in Equation 2.3.4a is the lack of consideration of $f_j^{match}(M)$. Including $f_j^{match}(M)$ in Equation 2.3.4a will greatly complicate the optimization for $C^{(s)}$. We decided to remove this term so that $C^{(s)}$ can be efficiently estimated in each study separately and then update M right after updating $C^{(s)}$. The simplified algorithm performed well in all our applications.

When updating M in Equation 2.3.4b, exhaustive search requires evaluation of all possible $(K!)^{S-1}$ combinations. In our motivating example of $K = 5$ and $S = 3$, it takes 14,400 evaluations. The number of evaluations increases to 207.36 million when S increases to 5. As an alternative, we propose a linear stepwise search to reduce the computational burden. In the first step, we match the first two studies with the largest sample sizes. Then the third study is added to match with existing patterns and the procedure continues by adding one study at a time. This approach will reduce to $(K!) \times (S-1)$ possible evaluations. The search space will reduce from exponential order to linear order of the number of studies. In

the case of $K = 5$ and $S = 5$, the number of evaluations reduces from 207.36 million to 480. In case that the linear stepwise search may reach an undesirable suboptimal solution, we propose a third approach to apply stepwise search solution as an initial value to a simulated annealing algorithm (Kirkpatrick et al., 1983) (see Appendix for detailed algorithm). Simulated annealing is an MCMC-based stochastic optimization algorithm for non-convex function. We expect that the third approach will achieve the best balance for affordable computing time while maintaining high clustering accuracy (Table **??**). The computing load and performance of these three matching approach will be evaluated in Section 2.4.4. In our software package, we suggest to perform exhaustive search when $(K!)^{S-1} \leq 14,400$ and automatically switch to simulated annealing otherwise.

2.3.2.3 Parameter selection In the MetaSparse$K$means formulation above, the number of clusters K are assumed pre-specified. In practice, it has to be estimated from data. The issue of estimating the number of clusters has received wide attention in the literature (Milligan and Cooper, 1985; Kaufman and Rousseeuw, 2009; Sugar and James, 2003). Here, we suggest the numbers of clusters to be estimated in each study separately using conventional methods such as prediction strength (Tibshirani and Walther, 2005) or gap statistics (Tibshirani et al., 2001) and jointly compared across studies (such that the numbers of clusters are roughly the same for all studies) for a final decision before applying MetaSparse$K$means. Below we assume that a common K is pre-estimated for all studies.

Another important parameter to be estimated is μ that controls the number of non-zero weights in the lasso regularization. Larger μ results in larger number of non-zero weights (i.e. the number of intrinsic genes to characterize the subtypes). We follow and extend the gap statistic procedure in sparse K-means (Witten and Tibshirani, 2010) to estimate μ:

1. For each gene feature in each study, randomly permute the gene expression row vector (permute samples). This creates a permuted data set $X^{(1)}$. Repeat for B times to generate $X^{(1)}, X^{(2)}, \ldots, X^{(B)}$.
2. For each potential tuning parameter μ, compute the gap statistics as below.

$$\text{Gap}(\mu) = O(\mu) - \frac{1}{B}\sum_{b=1}^{B} O_b(\mu), \tag{2.3.5}$$

where $O(\mu) = \sum_{j=1}^{p} z_j^* [\frac{1}{S}(\sum_{s=1}^{S} \frac{BCSS_j^{(s)}(C^{(s)*}(K))}{TSS_j^{(s)}}) + \lambda \times f_j^{match}(M^*)]$ is from observed data, where $\mathbf{z}^*, C^*(K), M^*$ are the maximizers of the objective function. $O_b(\mu)$ is similar to $O(\mu)$ but it is from permuted data $X^{(b)}$

3. For a range of selections of μ, select μ^* such that the gap statistics in Euqation 3.3.6 is maximized. Figure 11 shows the candidate region and the corresponding gene numbers of different μ for a simulated dataset that will be discussed in Section 3.4.1.

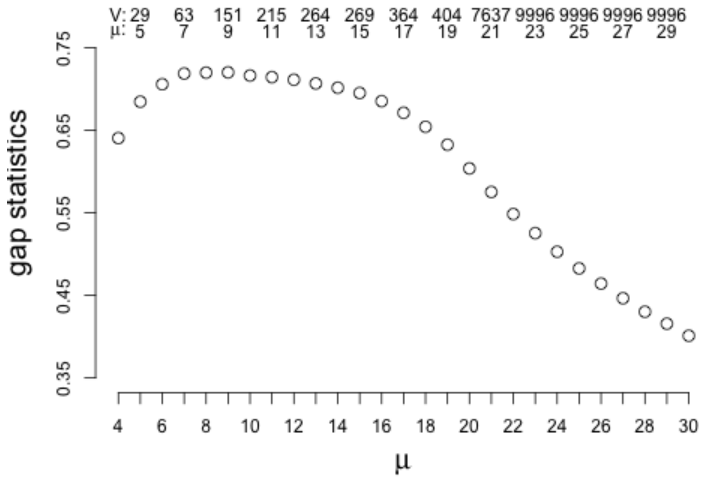


Figure 4: Gap statistics to select μ in simulated data with biological variance $\sigma_1 = 1$.
X-axis: μ; y-axis: gap statistics. V and μ on top give the number of non-zero weight features and corresponding tuning parameter. Gap statistics is maximized at $\mu = 9$, which is coresponding to 151 genes.

Our simulation has shown good performance of the gap statistics procedure but the performance may vary in real data. In practice, the users may test different selections of μ and examine the change of clustering assignment. In general, slight change of μ (or equivalently the number of selected genes) does not greatly change the clustering result. Another possibility is to use clinical or survival information to guide estimation of μ although we chose not to do so in the breast cancer example to avoid re-using the survival information in the evaluation.

Finally, the parameter λ controls the balance of the standardized $BCSS$ and pattern matching rewards in Equation 2.3.3. The former term drives the optimization to seek for clear cluster separations while the latter term emphasizes on concordant pattern of disease subtypes across studies. We performed sensitivity analysis on λ in the applications and found that slightly changing λ had little impact on the final clustering result in most cases. Since considerations of both terms are biologically important, we suggest to use $\lambda = 0.5$ in general unless users have particular reasons to change. Note that the first and second terms in Equation 2.3.3 are standardized to range between 0 and 1 and are at comparable scales.

2.3.2.4 Data visualization To generate heatmaps similar to Figure 2(a)-2(f), data normalization is necessary so genes at different expression scales can be presented simultaneously. Conventional wisdom in microarray analysis is to standardize each gene vector to have zero mean and unit variance in each study independently. This is, however, not applicable in our situation since the sample proportions of each disease subtype are not equal across studies. We instead applied a ratio-adjusted gene-wise normalization (Cheng et al., 2009) that accounts for differential subtype mixture proportions in the studies.

2.3.2.5 Classification of a future patient cohort For a future dataset that possibly comes from a different experimental platform, models from MetaSparse$K$means can help cluster the new cohort and match the signature patterns to determine the subtypes. The algorithm goes with two steps:

1. The optimal weights z^* from MetaSparse$K$means algorithm on training data are used to cluster patients of the new cohort using conventional K-means with pre-specified weighted gene structure:

$$C^{(new)} = \arg\min_{C} \sum_{j=1}^{p} z_j^* \sum_{k=1}^{K} \frac{1}{n_k} \sum_{l,m \in C_k} d_{lm,j}$$

2. The generated clusters $C^{(new)}$ are then matched back to disease subtypes determined by MetaSparse$K$means training results. Specifically, we ask for the best cluster pattern

matching of the new clusters to the original subtypes. Since the matching in the training studies are fixed, the optimization only requires MCC calculation of new cohort clustering $C^{(new)}$ with clustering of each training study $C^{(1)}, \ldots, C^{(S)}$.

$$M^{(new*)} = \arg\max_{M^{(new)}} \sum_{j=1}^{p} \sum_{s \in S} z_j^* MCC_j(\phi^{(s)}(C^{(s)}), \phi^{(new)}(C^{(new)}))$$

2.3.2.6 Extensions for practical applications Below we discuss two extensions for practical applications. Firstly, our framework has applied equal K in all studies. The question is whether and how to allow variable K across studies. Biologically, it is not reasonable to have wildly different number of disease subtypes across studies. Thus, we decided not to extend the algorithm for automatically searching variable K. Instead, we suggest the users to apply equal K and perform ad hoc analysis if evidence shows that some studies have almost no samples for a particular subtype or an additional subtype is needed (e.g. reduce from K=(5,5,5) to $K = (5, 4, 5)$ in the second study). Secondly, the number of genes may reduce greatly in the gene matching step if one or two studies apply an old array platform with less comprehensive coverage of the genome. In this case, our framework can easily extend to allow missing genes in partial studies (by simply ignoring the terms of a specific missing gene in a study). We have included this function in the software package and suggest to include genes as long as they appear in $> 70\%$ of studies.

2.4 RESULT

We evaluated MetaSparse$K$means on simulation datasets as well as two real multi-center examples in leukemia and breast cancer. In the simulation datasets, we showed that MetaSparse$K$means could recover the underlying true clusters with higher accuracy than single study analysis. We also showed that MetaSparse$K$means using equal weight (EW) is superior than MetaSparse$K$means using unequal weight (UW) in heterogenous scenario and reversely MetaSparse$K$means UW is superior than MetaSparse$K$means EW in homoge-

nous scenario. In leukemia dataset, we demonstrated that MetaSparse$K$means obtained unified gene selection and stable cluster pattern while single study analysis by sparse K-means claimed different gene selections and unmatched cluster patterns in different studies. In the breast cancer dataset, we applied MetaSparse$K$means to 3 breast cancer studies and showed that MetaSparse$K$means had better performance than single study sparse K-means. The classification model was used to predict the fourth METABRIC dataset and the meta-analyzed model generated more significant survival differences than the prediction based on single study models. Lastly we evaluated the computation time and accuracy for MetaSparse$K$means using different matching algorithm. MCMC (with linear stepwise search initial) will balance the computing load and optimization performance.

2.4.1 Simulation

2.4.1.1 Simulation setting To evaluate the performance of MetaSparse$K$means and compare with sparse K-means, we simulated $S(S = 3)$ studies with $K(K = 3)$ subtypes in each study. To best mimic the nature of microarray study, we will simulate confounding variables, gene correlation structure and noise genes (e.g. housekeeping genes or unexpressed genes). Below are the detailed generative steps to create subtype predictive genes, confounder impacted genes and noise genes.

(a) Subtype predictive genes.

1. We simulate $N_{k1} \sim \text{POI}(400)$, $N_{k2} \sim \text{POI}(200)$, $N_{k3} \sim \text{POI}(100)$ samples for subtype $k(1 \leq k \leq 3)$ in study $s(1 \leq s \leq 3)$. The number of subjects in study s is $N_s = \sum_k N_{ks}$.
2. Sample $M = 20$ gene modules $(1 \leq m \leq 20)$. In each module, sample n_m genes where $n_m \sim \text{POI}(20)$. Therefore, there will be an average of 400 subtype predictive genes.
3. μ_{sik} is the template gene expression of study $s(1 \leq s \leq S)$, subtype $k(1 \leq k \leq 3)$ and module $m(1 \leq m \leq M)$. For the first study, sample the template gene expression $\mu_{1km} \sim \text{UNIF}(4, 10)$ with constrain $\max_{p,q} |\mu_{1pm} - \mu_{1qm}| \geq 1$, where p, q denote two subtypes. For the second and third study, set $\mu_{2km} = \mu_{3km} = \mu_{1km}, \forall k, m$. This part define the subtype mean intensity for each module in all studies. To simulate the

situation that the first study (with the largest sample size) containing stronger signal, we introduced a new parameter f (for fold) to recalculate the template gene expression for the first study μ_{1km}: $\mu^{\star}_{1km} = (\mu_{1km} - \min_{k,m}\{\mu_{1km}\}) \times f + \min_{k,m}\{\mu_{1km}\}$, We set $f = 1$ unless otherwise mentioned.

4. Add biological variation σ_1^2 to the template gene expression and simulate $X^{'}_{skmi} \sim \mathrm{N}(\mu_{skm}, \sigma_1^2)$ for each module m, subject $i(1 \leq i \leq N_{ks})$ of subtype k and study s.
5. Sample the covariance matrix Σ_{mks} for genes in module m, subtype k and study s, where $1 \leq m \leq 20$, $1 \leq k \leq 3$ and $1 \leq s \leq 3$. First sample $\Sigma^{'}_{mks} \sim \mathrm{W}^{-1}(\Phi, 60)$, where $\Phi = 0.5I_{n_m \times n_m} + 0.5J_{n_m \times n_m}$, W^{-1} denotes the inverse Wishart distribution, I is the identity matrix and J is the matrix with all elements equal 1. Then Σ_{mks} is calculated by standardizing $\Sigma^{'}_{mks}$ such that the diagonal elements are all 1's.
6. Sample gene expression levels of genes in cluster m as $(X_{1skmi}, \ldots, X_{n_m skmi})^{\top} \sim \mathrm{MVN}(X^{'}_{skmi}, \Sigma_{mks})$, where $1 \leq i \leq N_{ks}$, $1 \leq m \leq M$, $1 \leq k \leq 3$ and $1 \leq s \leq 3$.

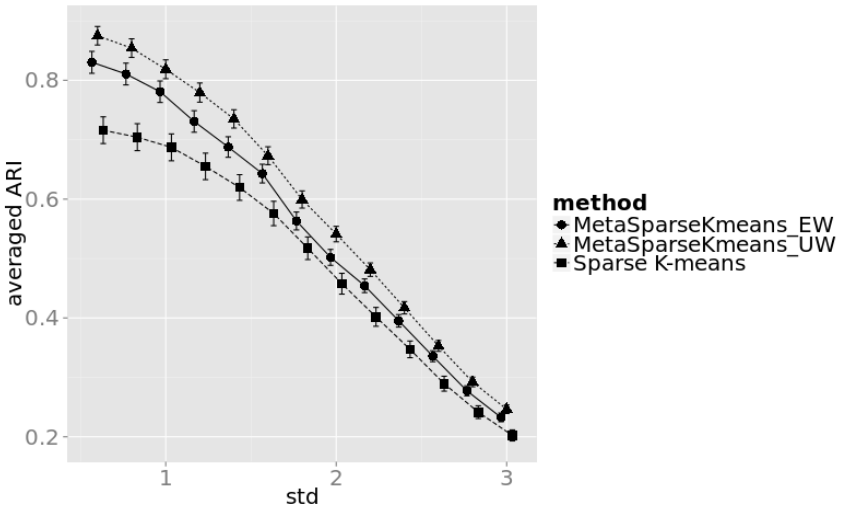
(b) Confounder impacted genes.

1. Sample 4 confounding variables. In practice, confounding variables can be gender, race, other demographic factors or disease stage etc. They will add heterogeneity to each study to complicate disease subtype discovery. For each confounding variable c, we will sample $R = 15$ modules. For each of these modules $r_c(1 \leq r_c \leq R)$, sample number of genes $n_{r_c} \sim \mathrm{POI}(20)$. These genes will be the same for all 3 studies. Therefore, there will be an average of 1,200 confounder impacted genes.
2. For each study $s(1 \leq s \leq 3)$ and each confounding variable c, sample the number of confounder subclass $h_{sc} \sim \mathrm{POI}(3)$ with constraint $h_{sc} > 1$. The N_s samples in study s will be randomly divided into h_{sc} subclasses.
3. Sample confounding template gene expression $\mu_{slrc} \sim \mathrm{UNIF}(4, 10)$ for confounder c, gene module r, subclass $l(1 \leq l \leq h_{sc})$ and study s. We recalculate $\mu^{\star}_{1lrc} = (\mu_{1lrc} - \min_{lrc}\{\mu_{1lrc}\}) \times f + \min_{lrc}\{\mu_{1lrc}\}$, which is similar to Step 3. Add biological variation σ_1^2 to the confounding template gene expression $X^{'}_{scrli} \sim \mathrm{N}(\mu_{slrc}, \sigma_1^2)$. Similar to Step 6 and 7, we simulate gene correlation structure within modules of confounder impacted genes.

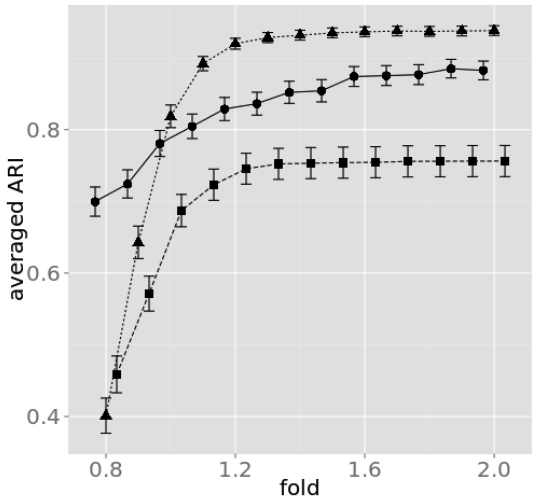
(c) Noise genes.

1. Sample 8,400 noise genes denoted by $g(1 \leq g \leq 8,400)$. For each study, we generate the mean template gene expression $\mu_{sg} \sim \text{UNIF}(4, 10)$. Then we add biological variation variance $\sigma_2^2 = 1$ to generate $X_{sgi} \sim N(\mu_{sg}, \sigma_2^2), 1 \leq i \leq N_s$. Gene expression level generated here will be relatively stable. Therefore these genes could be regarded as housekeeping genes if their expression are high, or un-expressed genes if their expression are low.

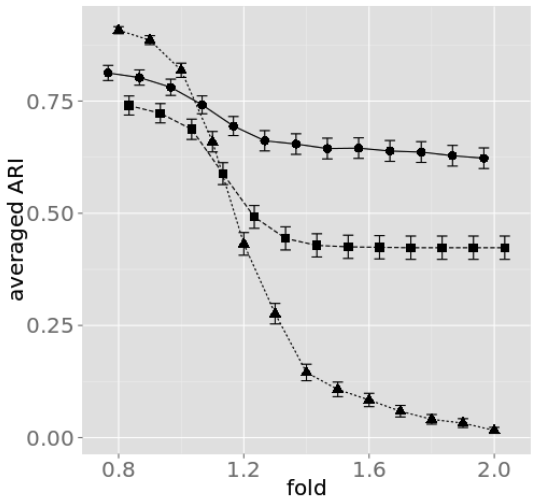
2.4.1.2 Simulation result In this section we compared the performance of MetaSparse$K$means using equal weight (EW) and unequal weight (UW), and compared metaSparse$K$means with single study sparse K-means result. The tuning parameter for MetaSparse$K$means was selected from gap statistics. For a fair comparison, we selected the tuning parameter in single study such that the number of selected genes are similar to the number in MetaSparse$K$means. We compared the results by adjusted Rand index (Hubert and Arabie, 1985) (ARI) with the underlying truth in each study. The ARIs were averaged over 3 studies. Figure 5(a) shows the performance of three methods for $B = 100$ simulations and $\sigma_1 = 0.6, 0.8 \sim 3$ (error bars represent mean $\pm$ standard error). When the biological variation increases, performance of all three methods decreases. MetaSparse$K$means (both EW and UW) outperforms individual analysis. Figure 5(b) shows the performance of three methods when the subtype predictive gene fold change in the largest study f varies: $f = 0.8, 0.9 \sim 2$ (error bars represent mean $\pm$ standard error). When the largest study has stronger signal $f > 1$, performance of MetaSparse$K$means-UW is better than MetaSparse$K$means-EW. When the largest study has weaker signal $f < 1$, performance of MetaSparse$K$means-EW is better than MetaSparse$K$means-UW. Figure 5(c) shows a third simulation when the fold change of the confounding impacted genes in the largest study varies: $f = 0.8, 0.9 \sim 2$ (error bars represent mean $\pm$ standard error). When the largest study has strong confounding effect (i.e. heterogeneous compared to other studies) $f > 1$, MetaSparse$K$means-UW has worse performance than MetaSparse$K$means-EW and can be even worse than individual study clustering. When the studies are more homogeneous $f < 1$, performance of MetaSparse$K$means-UW is superior.



(a) Vary biological variance



(b) Vary subtype signal in the largest study



(c) Vary confounding effect in the largest study

Figure 5: Simulation result comparing MetaSparse$K$means.

Simulation result comparing MetaSparse$K$means(EW), MetaSparse$K$means (UW) and sparse K-means under different scenarios. Figure 5(a): varying biological variance. Figure 5(b): varying subtype predictive gene intensity in the first study with the largest sample size. Figure 5(c): varying confounding impacted gene intensity in the first study with the largest sample size.

Table 2: Leukemia dataset information

Study Name	Verhaak at el.	Balgobind at el.	Kohlmann at el.
Number of probes	48,788	48,788	48,788
Number of patients	89	74	105
True class label $\star$	(33, 21, 35)	(27, 19, 28)	(28, 37, 40)
Data range	$[4.907, 14.159]$	$[3.169, 15.132]$	$[0, 1]$
Mean intensity	6.163	6.093	0.309
Standard deviation	1.543	1.334	0.196
Platform	Affymetrix human genome u133 plus 2.0 array		
$\star$: true class labels are the number of samples for (inv(16), t(15:17), t(8,21))			

2.4.2 Leukemia example

Table 2 shows a summary description of three Leukemia transcriptomic studies: Verhaak (Verhaak et al., 2009), Balgobind (Balgobind et al., 2011), Kohlmann (Kohlmann et al., 2008). We only considered samples from acute myeloid leukemia (AML) with subtype inv(16)(inversions in chromosome 16), t(15;17)(translocations between chromosome 15 and 17), t(8;21)(translocations between chromosome 8 and 21). These three gene-translocation AML subtypes have been well-studied with different survival, treatment response and prognosis outcomes. We treat these class labels as the underlying truth to evaluate the clustering performance. The expression data for Verhaak, Balgobind ranged from around $[3.169, 15.132]$ while Kohlmann ranged in $[0, 1]$. All the datasets were downloaded directly from NCBI GEO website. Originally there were 54,613 probe sets and we filtered out probes with 0 standard deviation in any study. In the end 48,788 probes were remained matched across studies. Three gene expression matrices with sample size 89, 74 and 105 were used as input data for disease subtype discovery.

To compare the performance between MetaSparse$K$means and single sparse K-means, we chose μ such that the number of selected probe sets was around 200-300 in each method.

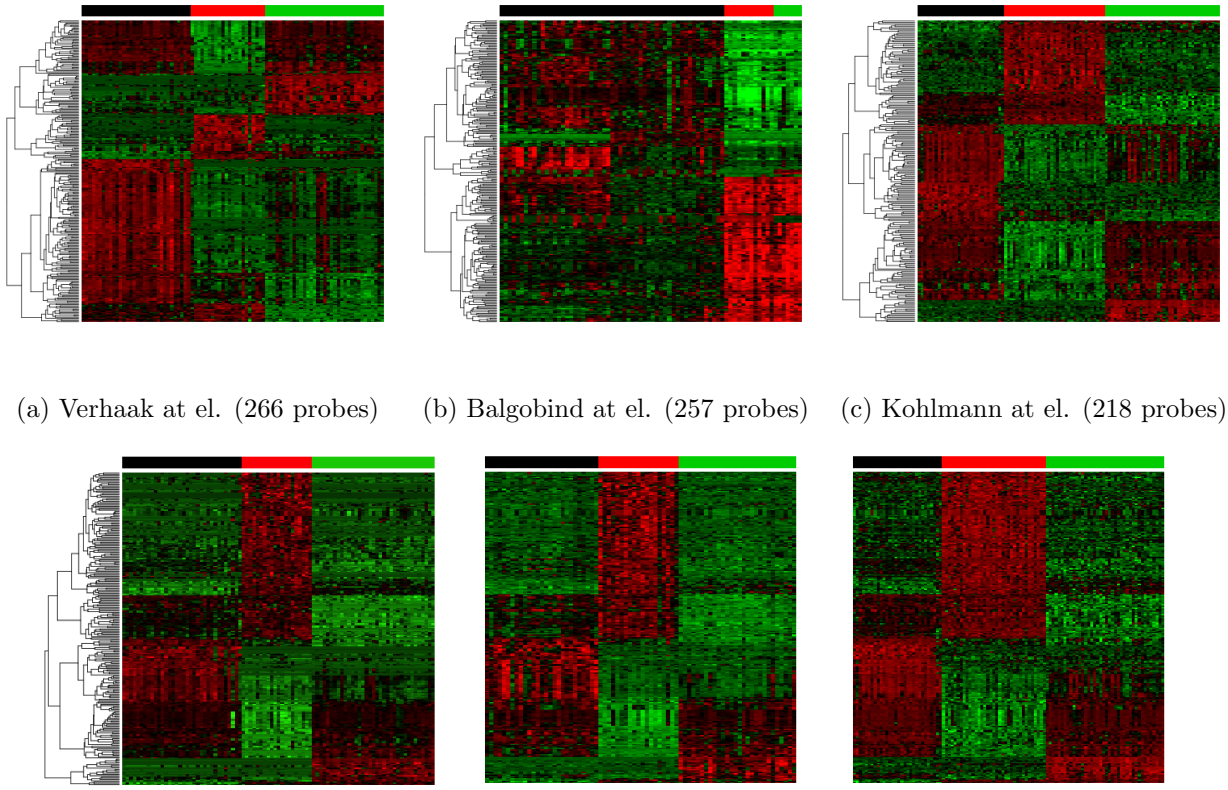
Table 3: Comparison between MetaSparse$K$means and sparse K-means on Leukemia dataset

	MSKM	Verhaak	Balgobind	Kohlmann
μ	12	10	10	10
Number of selected probes	245	266	257	218
ARI	0.97/1/0.95	0.97	0.41	0.95

Figure 6(a)-6(c) show heatmap of clustering results from each single study sparse K-means. Each study generated three disease subtypes using different intrinsic gene sets, making it difficult to classify future patients with a unified classification rule. Figure 6(d)-6(f) demonstrate heatmap from MetaSparse$K$means clustering using 245 probe sets. We not only obtained a common intrinsic gene set, but also observed clear consistent patterns of the three disease subtypes across the three studies. Table 3 shows the ARI of each clustering result with the underlying leukemia subtype truth. Single study analysis in Verhaak and Kohlmann produced almost perfect clustering ($ARI = 0.97$ and 0.95) while Balgobind gave a poor $ARI = 0.41$. The MetaSparse$K$means generated improved ARIs in each study ($ARI = 0.97$, 1 and 0.95).

2.4.3 Breast cancer example

2.4.3.1 Clustering result and survival association As shown in the motivating example in Figure 2(a)-2(c), single study sparse K-means generated different sets of intrinsic genes. MetaSparse$K$means obtained 203 common intrinsic genes to cluster the patients into five disease subtypes with consistent expression pattern across studies. Since the underlying true cancer subtypes are not available in this example, we applied the models from each method to classify an independent testing cohort METABRIC (Curtis et al., 2012), which contained 1,981 samples from Illumina HT12 arrays. This serves the purpose of extending the training model to a validating dataset. Figure 7(a) shows the subtype prediction patterns from MetaSparse$K$means method. We can clearly see that the resulting expres-



(a) Verhaak at el. (266 probes) (b) Balgobind at el. (257 probes) (c) Kohlmann at el. (218 probes)

(d) Verhaak at el. (245 probes) (e) Balgobind at el. (245 probes) (f) Kohlmann at el. (245 probes)

Figure 6: Leukemia results after MetaSparse$K$means.

The three figures on top are heatmaps of Leukemia dataset after sparse K-means. The three figures on bottom are results from MetaSparse$K$means.

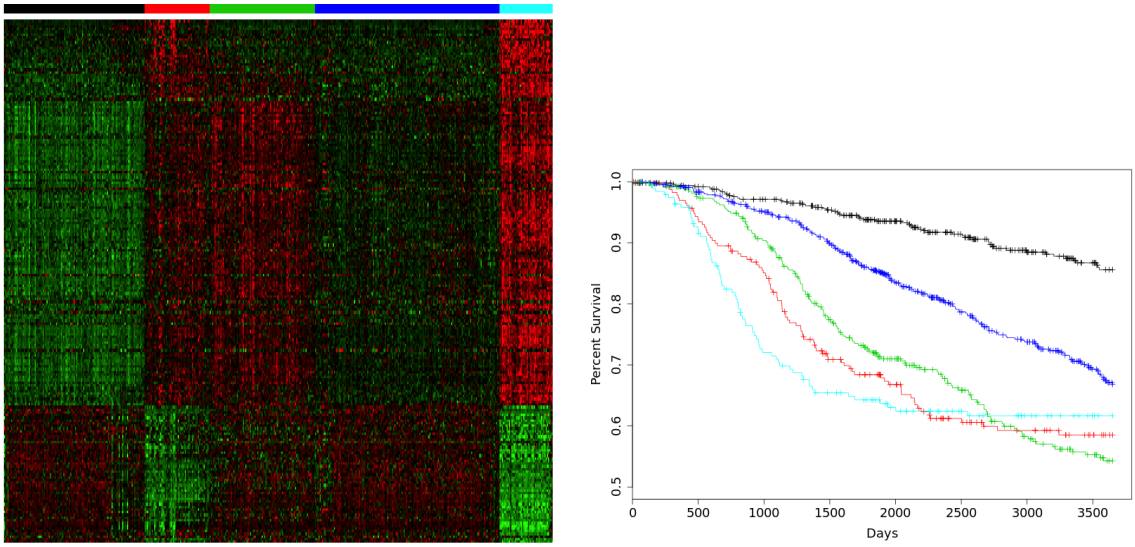
Table 4: Survival analysis in METABRIC

Model	# of Samples	# of selected genes	p value
Meta(TCGA+Wang+Desmedt)	533+260+164	203(194)	3.79×10^{-25}
TCGA	533	239(233)	1.46×10^{-19}
Wang	260	220(214)	3.31×10^{-14}
Desmedt	164	197(193)	7.81×10^{-14}
PAM50		50	1.01×10^{-20}

Classification models trained in each single study and combined meta-framework are applied to METABRIC. P-value of survival differences of identified subgroups were evaluated based on log-rank test. The previously published PAM50 model was also compared. The number in () indicates the actual number of genes used in the prediction model since a few genes were not observed in the METABRIC array platform.

sion patterns are consistent with those from three training studies in Figure 2(d)-2(f). The Kaplan-Meier survival curves of the five disease subtypes are well-separated with p-value 3.79×10^{-25} from log-rank test (Figure 7(b)). The survival separation demonstrates high potential of clinical utility of the discovered disease subtypes. Note that although only 194 out of 203 genes appeared in the METABRIC dataset, those genes still had enough power to separate the subtypes. Table 4 shows log-rank p-value of survival separation from each individual sparse K-means classification and PAM50. MetaSparse$K$means generated the best survival separation of the subtypes. PAM50 is currently the most well-accepted transcriptomic subtype definition of breast cancer. We have further compared the clustering results from MetaSparse$K$means and PAM50 in the Appendix and Supplement Table 12

2.4.3.2 Pathway Enrichment In order to evaluate whether the genes obtained from each model are biologically meaningful, pathway enrichment analysis was perform using Fisher's exact tests by testing association of selected intrinsic genes and genes in a particular pathway. We applied the BioCarta Database obtained from MSigDB (http://www.broadinstitute.org/gsea/msigdb/collections.jsp#C2). This database contains 217 cu-



(a) Heatmap of METABRIC based on 194 genes from MetaSparse$K$means.

(b) Survival curves of the 5 subtypes from MetaSparse$K$means validation. The color is corresponding to the subtype color in the heatmap.

Figure 7: Clinical result of METABRIC dataset

rated cancer related pathways and is particularly suited to evaluate the breast cancer example. Figure 8 shows the jitter plot pathway enrichment q-values at log-scale (base 10). The horizontal solid line corresponds to the $q = 0.05$ significance level threshold. The pathway enrichment result from MetaSparse$K$means yielded more significant pathways than the individual models(7 significant pathway in MetaSparse$K$means versus 1 in individual sparse K-means). All 8 significant pathways are listed in Table 5.

2.4.3.3 Accuracy and stability analysis We have performed additional subsampling evaluation on breast cancers studies to evaluate the accuracy and stability of MetaSparse$K$means compared to single study analysis. For accuracy, since TCGA had larger sample size than the other two studies, we randomly subsampled $50\%, 60\%, 70\%, 80\%, 90\%$ of samples in TCGA for evaluation. Sparse K-means was applied to the whole TCGA data (n=533) without considering Wang and Desmedt to generate sample clustering $C_{TCGA,all}$ and this result was treated as a pseudo-gold standard. Sparse K-means was then similarly ap-

Table 5: Eight significant BIOCARTA pathways.

Pathway name	MetaSparse$K$means	TCGA	Wang	Desmedt
BIOCARTA SRCRPTP PATHWAY	*0.0255	1	1	1
BIOCARTA MCM PATHWAY	$^\star 6.47 \times 10^{-6}$	1	1	1
BIOCARTA G1 PATHWAY	*0.0427	1	1	1
BIOCARTA G2 PATHWAY	*0.0367	1	1	1
BIOCARTA P27 PATHWAY	*0.0472	1	1	1
BIOCARTA RANMS PATHWAY	*0.0229	1	1	1
BIOCARTA PTC1 PATHWAY	*0.0287	1	1	1
BIOCARTA HER2 PATHWAY	0.149	0.170	*0.0078	0.0817

The p-values were obtained using Fisher's exact tests based on selected genes from MetaSparse$K$means or individual study clustering and Benjamini-Hochberg correction (Benjamini and Hochberg, 1995) was applied to generate q-values in the table. *: q-value smaller than 0.05 cutoff.

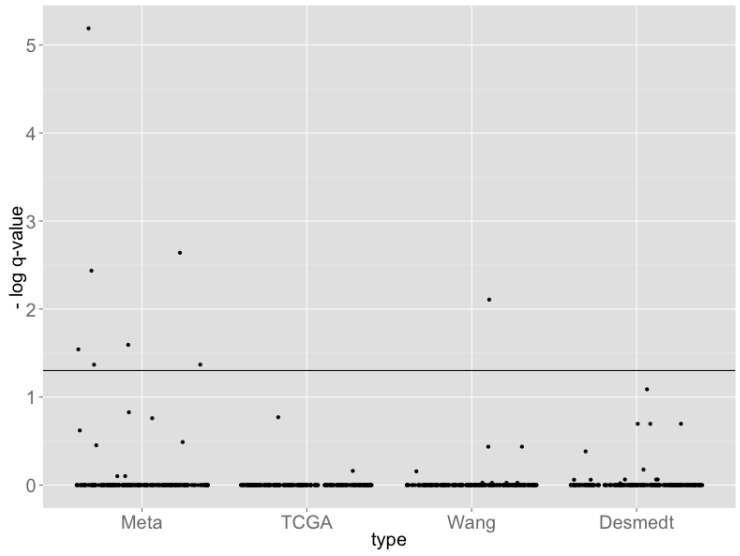


Figure 8: Pathway enrichment result from four different models (Meta, TCGA, Wang, Desmedt).

Clustering from meta-analysis identified intrinsic genes more associated to cancer related pathways.

plied to 100 independently subsampled $p\%(p = 50, 60, 70, 80, 90)$ TCGA dataset to generate clustering result $C^{(b)}_{TCGA,p\%}$ $(1 \leq b \leq 100)$. The adjusted Rand index (ARI) was calculated between $C^{(b)}_{TCGA,p\%}$ and $C_{TCGA,all}$ and the trajectories with error bar (standard error) are shown in Figure 9(a) (blue). Similar analysis was performed for MetaSparse$K$means when the TCGA subtype clustering results were combined with Wang and Desmedt for clustering and the ARI results were shown in red. In this analysis, we used the large sample size of TCGA data to generate the subtype clustering result and treated it as a pseudo-gold standard. The data subsampling represented the situation when sample size was not large and the ARI value represented an indirect evidence of the clustering accuracy. Figure 9(a) demonstrates a clearly better accuracy for MetaSparse$K$means than single study sparse K-means and the increased power evidently comes from the incorporated information from the other two studies, Wang and Desmedt.

For stability, we performed similar subsampling in TCGA data as before. But instead of comparing to the whole data clustering results, we restricted to all pair-wise comparison of subsampled data. For a given $p\%$ subsampling rate, B $(B = 100)$ TCGA subsampled data were generated and sparse K-means were applied to each subsampled dataset. ARIs were calculated for each pair-wise comparison that generated $C^{100}_2 = 4950$ ARIs and the trajectories with error bar (standard error) are shown in Figure 9(b) (blue). Similar analysis for MetaSparse$K$means was performed where Wang and Desmedt were combined with subsampled TCGA data in the subtype clustering (red in Figure 9(b)). The result showed that MetaSparse$K$means generated more stable disease subtype assignments than single study sparse K-means by incorporating information from the other two studies. Note that when comparing two $p\%$ subsampled clustering results, only overlapped samples were considered in the ARI calculation.

2.4.4 Computation time and matching accuracy

To evaluate computation time for the MetaSparse$K$means algorithm using different pattern matching algorithms, we will use the simulation scenario in Section 3.4.1 with different S, K and σ. We use two criteria to evaluate the accuracy for using different matching algorithms

Table 6: Computing time for different matching methods

	Algorithm	S=3	S=5	S=15
K=3	Exhaustive	2.604	5.614	$> 2.9 \times 10^4$
	Stepwise	2.854	5.290	18.024
	MCMC	4.288	7.429	35.736
K=5	Exhaustive	15.616	$> 2.9 \times 10^4$	$> 2.9 \times 10^4$
	Stepwise	8.738	13.951	39.273
	MCMC	11.645	16.541	78.687

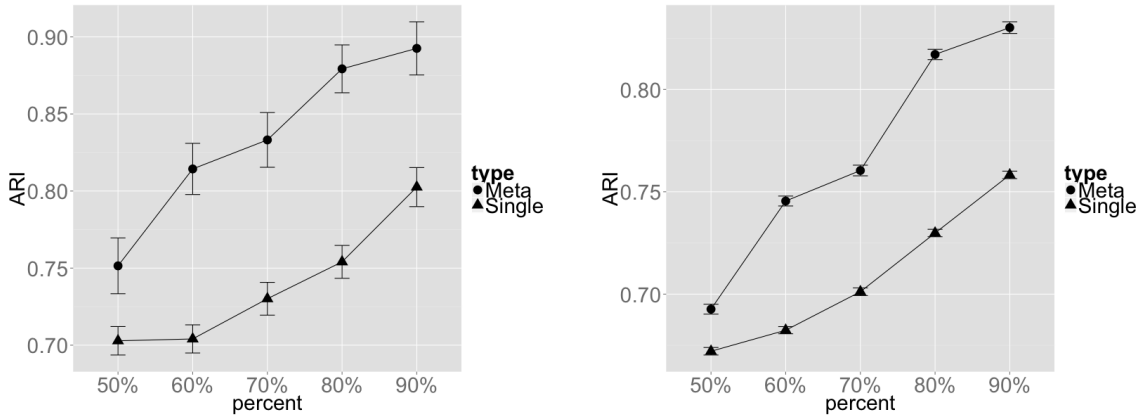
Computing time in minutes comparing different combination of S and K using a regular desktop computer.

described in Section 2.3.2.2: percent of reaching global optimal based on Equation 2.3.4b, and the resulting cluster agreement with the underlying truth using ARI. Table 6 shows that stepwise and MCMC searching greatly reduced computing time for large S. Even in a large meta-analysis of $S = 15$ and $K = 5$, computing time was at 39 and 79 minutes without using any powerful machine or parallel programming. In Table 7, we fixed $S = 3$ and $K = 3$ and varied biological variance $\sigma = 2, 6$ and did 100 simulations for each σ. On the left, the performance of matching score is evaluated by comparing with exhaustive matching score. We observed that stepwise matching sometime will deviate from the optimal matching, but MCMC (with stepwise initial) can increase the chance to the best matching. On the right, we evaluated the final cluster agreement. We observed that all of the three methods would achieved similar performance. The result demonstrates that MCMC achieves the best balance between computing load and optimization performance. Besides, in our real data examples, all three matching algorithms will yield the same clustering result.

Table 7: Accuracy for different matching methods

	% of optimal		accuracy	
Variance	$\sigma = 2$	$\sigma = 6$	$\sigma = 2$	$\sigma = 6$
Exhaustive	100%	100%	0.829 ± 0.031	0.020 ± 0.002
Stepwise	93.3%	92.8%	0.828 ± 0.031	0.020 ± 0.002
MCMC	100%	100%	0.828 ± 0.031	0.020 ± 0.002

Performance comparing with the best matching score (percentage of agreement with optimal matching) and clustering accuracy by ARI (mean estimate± standard error) under different biological variances ($\sigma = 2$ and $\sigma = 6$).



(a) accuracy

(b) robustness

Figure 9: Accuracy comparison of MetaSparse$K$means and sparse K-means.

Figure 9(a) compares the accuracy of MetaSparse$K$means and sparse K-means. For sparse K-means we used the TCGA data (n=533) only and for MetaSparse$K$means we combined TCGA, Wang and Desmedt. At each sub-sampling point, ARI was calculated 100 times and averaged. Figure 9(b) compares the stability of MetaSparse$K$means and sparse K-means. At each sub-sampling point, ARI was calculated 4950 times and averaged.

2.5 DISCUSSION

Disease phenotyping and subtype discovery have received increasing attention since high-throughput experimental data have become more and more affordable and prevalent. In the literature, such a modeling is usually performed in a single study and attempts have been made to validate in other studies. As more and more studies of the same disease are available, combining multiple studies for simultaneous subtype clustering is an appealing approach to identify a common set of intrinsic genes and a common model of subtype definition for future prediction. In this chapter, we developed a MetaSparse$K$means framework that can achieve this goal. Simulations and applications to leukemia datasets and breast cancer datasets demonstrated improved performance by meta-analysis. We demonstrated a superior accuracy and stability of MetaSparse$K$means compared to individual analysis counterpart in the breast cancer example. We also performed an validation on a large independent METABRIC study which evaluated its potential clinical significance by survival analysis and demonstrated the better pathway association of the identified intrinsic genes with cancer related pathways.

Although MetaSparse$K$means was mainly applied to transcriptomic studies in this chapter, it can also be applied to other high-throughput omics data such as methylation, copy number variation, miRNA and proteomics. There are a few potential extensions of MetaSparse$K$means. First of all, the feature selection in sparse K-means ignores prior knowledge or dependence structure between features. For example, if features contain both gene expression and methylation, the inter-relationship between multi-omics data may be modeled to improve the analysis and interpretation. Secondly, the gap statistic usually leads to a candidate region with near optimal μ and we selected μ corresponding to less number of features. One may design a penalized gap statistics for which μ could be automatically selected. Thirdly, disease-related genes or pathways may be available in well-studied diseases. Incorporating these prior biological information may generate more biologically relevant results and is a future direction. Finally, subtypes identified by MetaSparse$K$means do not necessarily guarantee association with clinical outcome (e.g. survival, tumor stage, tumor grade etc). It is possible that less obvious subtypes with important clinical association may

be masked by strong subtypes with no clinical importance. A guided clustering approach incorporating prior clinical information may help identify clinically relevant disease subtypes.

MetaSparse$K$means inherits fast computation from K-means algorithms. The stepwise search algorithm and simulated annealing also provide a viable solution to the large searching space of cluster matching when the number of studies is large. In the breast cancer example ($K = 5$ and $S = 3$), MetaSparse$K$means took only about 8 minutes for exhaustive search using a regular laptop (CPU 2GHz and 4GB RAM). An R package MetaSparse$K$means is available to perform the analysis.

3.0 INTEGRATIVE SPARSE KMEANS

3.1 INTRODUCTION

In section 1.2, we introduced the background for disease subtype discovery via transcriptomic data. In section 3.1, we introduced the background of horizontal meta-analysis for disease subtype discovery via combining multiple transcriptomic studies. Over the years large amount of omics data are accumulated in public databases and depositories, vertical integrative analysis is appealing since we are able to draw robust conclusion by taking into account the regulatory relationships between different levels of omics data. Omics integrative analysis has been found successful in many applications: (e.g. breast cancer (Koboldt et al., 2012), stomach cancer (Bass et al., 2014)). On the other hand, tremendous amount of biological information has been accumulated in public databases. Proper usage of these prior information (e.g. pathway information and miRNA targeting gene database) can greatly guide the modeling of omics integrative analysis.

In this chapter, we focus on vertical omics integrative analysis for disease subtype discovery. Several methods for this purpose have been proposed in the literature. Lock and Dunson (2013) fitted a finite Dirichlet mixture model to perform Bayesian consensus clustering that allows common clustering across omics types as well as omics-type-specific clustering. The model, however, does not perform proper feature selection and thus is not suitable for high-dimensional omics data. Shen et al. (2009) proposed a latent variable factor model (namely iCluster) to cluster cancer samples by integrating multi-omics data. The method does not incorporate prior biological knowledge and requires extensive computing due to EM algorithm with large matrix operation. We will use the popular iCluster method as the baseline method to compare in this chapter. The content of this chapter is accepted by the Annals of Applied Statistics (Huo and Tseng, 2017).

The central question we ask in this Chapter is: "Can we identify cancer subtypes by simultaneously integrating multi-level omics datasets and/or utilizing existing biological knowledge to increase accuracy and interpretation?" Several statistical challenges will arise when we attempt to achieve this goal: (1) If multi-level omics data are available for a given patient cohort, what kind of method is effective to achieve robust and accurate disease subtype detection via integrating multi-omics data? (2) Since only a small subset of intrinsic omics features are relevant to the disease subtype characterization, how can we perform effective feature selection in the high-dimensional integrative analysis? (3) With the rich biological information (e.g. targeted genes of each miRNA or potential cis-acting regulatory mechanism between copy number variation, methylation and gene expression), how can we fully utilize the prior information to guide feature selection and clustering? In this chapter, we propose an integrative sparse K-means (IS-Kmeans) (Huo and Tseng, 2017) approach by extending the sparse K-means algorithm with overlapping group lasso technique to accommodate the three goals described above. The lasso penalty in the sparse K-means method allows effective feature selection for clustering. In the literature, (non-overlapping) group lasso (Yuan and Lin, 2006) has been developed in a regression setting to encourage features of the same group to be selected or excluded together. The approach, however, has two major drawbacks: (1) it does not allow sparsity within groups (i.e. a group of features are either all selected or all excluded), and (2) the penalty function does not allow overlapping groups. For the first issue, Simon et al. (2013) proposed a sparse group lasso with both an l_1 lasso penalty and a group lasso penalty to allow sparsity within groups while the approach does not allow overlapping groups. For the latter issue, overlapping group information from biological knowledge is frequently encountered in many applications. In genomic application, for example, the targeted genes of two miRNAs are often overlapped or two pathways may contain overlapping genes. Jacob et al. (2009) proposed a duplication technique to allow overlapping groups in regression setting while the approach does not allow sparsity within groups. In this chapter, we attempt to simultaneously overcome both aforementioned difficulties in a clustering setting, which brings optimization challenges beyond the duplication technique by Jacob et al. (2009) and the sparse group lasso optimization by Simon et al.

(2013). In our proposed IS-Kmeans method, we will develop a novel reformulation of l_1 lasso penalty and overlapping group lasso penalty so that a fast optimization technique using alternating direction method of multiplier (ADMM) (Boyd et al., 2011) can be applied (see Section 3.3.3.1).

The rest of the chapter is structured as following. Section 3.2 gives a motivating example. Section 3.3 establishes the method and optimization procedure. Section 3.4.1-3.4.3 comprehensively compares the proposed method with the popular iCluster method using simulation and two breast cancer applications on multi-level omics data. Section 3.4.4 provides another type of IS-Kmeans application of pathway-guided clustering on single transcriptomic study. Section 3.5 includes final conclusion and discussion.

3.2 MOTIVATING EXAMPLE

Figure 10A shows a clustering result using single study sparse K-means (detailed algorithm see Section 1.3.3.2) on the mRNA, methylation and copy number variation (CNV) datasets separately from 770 samples in TCGA. As expected, they generate very different disease subtyping without regulatory inference across mRNA, methylation and CNV. In this example, single study sparse K-means fails to consider that different omics features belonging to the same genes are likely to contain cis-acting regulatory mechanisms related to the disease subtypes. Figure 10B combines the three datasets to perform IS-Kmeans. The IS-Kmeans generates a single disease subtyping and takes into account of the prior regulatory knowledge between mRNA, methylation and CNV. The prior knowledge can also be pathway database (e.g. KEGG, BioCarta and Reactome) or knowledge of miRNA targets prediction databases (e.g. PicTar, TargetScan, DIANA-microT, miRanda, rna22 and PITA) (Witkos et al., 2011; Fan and Kurgan, 2015). Incorporating such prior information of feature grouping increases statistical power and interpretation. Figure 10C shows a simple example of such group prior knowledge. Pathway $\mathcal{J}_1$ includes mRNA1, mRNA2, mRNA3 and mRNA6 while pathway $\mathcal{J}_2$ includes mRNA3, mRNA4, mRNA5 and mRNA7. Note that mRNA3 appears in both pathway $\mathcal{J}_1$ and $\mathcal{J}_2$, which requires our algorithm to allow overlapping groups. Our goal is

to develop a sparse clustering algorithm integrating multi-level omics datasets and the aforementioned prior regulatory knowledge by overlapping group lasso. The algorithm is also suitable for single omics dataset with incorporating prior overlapping pathway information (see the leukemia examples in Section 3.4.4).

3.3 METHOD

3.3.1 Integrative Sparse K-means (IS-Kmeans)

We have introduced K-means and sparse K-means in Section 1.3.3.2. We extend the sparse K-means objective function to group structured sparse K-means. Here we consider J to be the total number of features combing all levels of omics datasets. In order to make features of different omics data types on the same scale and comparable, we normalized $BCSS_j$ by TSS_j and denote

$$R_j(C) = \frac{BCSS_j(C)}{TSS_j}$$

We put the overlapping group lasso penalty term $\Omega(\mathbf{z})$ in the objective function.

$$\begin{aligned} &\min_{C,\mathbf{z}} -\sum_{j=1}^{J} z_j R_j(C) + \gamma\alpha\|\mathbf{z}\|_1 + \gamma(1-\alpha)\Omega(\mathbf{z}) \\ &\text{subject to } \|\mathbf{z}\|_2 \leq 1, z_j \geq 0, \end{aligned} \tag{3.3.1}$$

where γ is the penalty tuning parameter controlling the numbers of non-zero features, $\alpha \in [0,1]$ is a term controlling the balance between individual feature penalty and group feature penalty. If $\alpha = 1$, there is no group feature penalty term and the objective function is equivalent to sparse K-means objective function after standardizing each feature. If $\alpha = 0$, there is no individual feature penalty and only group feature penalty exists. The overlapping group lasso penalty term is defined as

$$\Omega(\mathbf{z}) = \sum_{1\leq g\leq \mathcal{G}_0} w_g \|\mathbf{m}_g \circ \mathbf{z}\|_2,$$

where $\mathcal{G}_0$ is the number of (possibly overlapping) feature groups from prior biological knowledge, $w_g \in \mathbb{R}$ is the group weight coefficient for group g, $\mathbf{m}_g = (\mathbf{m}_{g1}, \ldots, \mathbf{m}_{gJ})$ is the design

vector of the g^{th} feature group and $\circ$ represents Hadamard product. The design of w_g and $\mathbf{m}_g$ is discussed in Section 3.3.2. Note that features with no group information are also treated as a group by itself (a group only contains a feature); such a design is to avoid bias towards a feature with no group information by receiving no penalization. The feature groups can either come from existing biological databases (e.g. pathway or miRNA target database), or from basic biological cis-regulatory knowledge (CNV and methylation features in the neighborhood of a nearby gene region). The first term in Equation 3.3.1 encourages large weights for features with strong clustering separability. The second term is an l_1 norm lasso penalty to encourage sparsity. Finally, $\Omega(\mathbf{z})$ serves as overlapping group lasso to encourage features in the prior knowledge groups to be selected simultaneously (or discarded together). The intuition of group lasso is that if we transform the Lagrange form of $\Omega(\mathbf{z})$ to its constraint form, it becomes an elliptic constraint and features of the same group are preferred to be selected together (Yuan and Lin, 2006; Jacob et al., 2009). The combination of l_1 norm lasso penalty and overlapping group lasso penalty $\Omega(\mathbf{z})$ serves to achieve a sparse feature selection and also encourages (but does not force) features of the same group to be selected together.

Remark. *Since different types of omics datasets may have different value ranges and distributions, additional normalization may be needed in the preprocessing. For example, the commonly-used beta values from methy-seq (defined as "methylation counts"/"total counts") represent the proportions of methylation and range between 0 and 1. A logit transformation to so-called M-values is closer to Gaussian distribution and is more suitable to integrate with other omics data. Similarly, log-transformation of expression intensities from microarray, log-transformation of RPKM/TPM (summarized expression values) from RNA-seq and log-ratio values of CNV values from SNP arrays have been shown to be roughly Gaussian distributed and are proper for multi-omics integration. Another possibility is by replacing Euclidean distance to an appropriate distance measurement (e.g. Gower's distance for binary categorical and ordinal data, and Bray-Curtis dissimilarity for count data). Under this scenario, Equation 3.3.1 remains valid under such modification and we only need to incorporate partition around medoids (PAM) (Kaufman and Rousseeuw, 1987) instead of K-means in the optimization procedure in Section 3.3.3.1. However, heterogeneity of different distance measurement may require extra different sparsity penalties and this is beyond consideration in this dissertation.*

3.3.2 Design of overlapping group lasso penalty

In this section, we discuss and justify the design of overlapping group lasso penalty for w_g and $\mathbf{m}_g$. We denote by $\mathcal{J}_g$ as the collection of features in group g ($1 \leq g \leq \mathcal{G}_0$) and define frequency of feature j appearing in different groups: $h(j) = \sum_{1 \leq g \leq \mathcal{G}_0} \mathbb{I}\{j \in \mathcal{J}_g\}$. We also define the intrinsic feature set $\mathcal{I}$ (i.e. features that contribute to the underlying true sample clustering) and the non-intrinsic feature set $\bar{\mathcal{I}}$. We first state an "Unbiased Feature Selection" principle under a simplified situation:

Definition 3.3.1 ("Unbiased Feature Selection" principle)**.** *Suppose equal separation ability in all intrinsic features $\mathcal{I} = \{j : R_j = R > 0\}$ and no separation ability in non-intrinsic features $\bar{\mathcal{I}} = \{j : R_j = 0\}$ under the true clustering label. The proposed overlapping group lasso design (w_g and $\mathbf{m}_g$) is said to satisfy the "Unbiased Feature Selection" principle if under Equation 3.3.1, it generates equal weights $z_j = 1/\sqrt{|\mathcal{I}|}$ for $j \in \mathcal{I}$ and $z_j = 0$ for $j \in \bar{\mathcal{I}}$ given any prior knowledge of feature groups $\mathcal{J}_g$, $1 \leq g \leq \mathcal{G}_0$.*

The theorem below states an overlapping group lasso penalty design that satisfies "Unbiased Feature Selection" principle when all features are intrinsic features (i.e. $\bar{\mathcal{I}} = \phi$).

Theorem 3.3.1. *Consider $\Omega(\mathbf{z}) = \sum_{1 \leq g \leq \mathcal{G}_0} w_g \|\mathbf{m}_g \circ \mathbf{z}\|_2$ and $\mathbf{m}_g = (\mathbf{m}_{g1}, \ldots, \mathbf{m}_{gj}, \ldots, \mathbf{m}_{gJ})$ in Equation 3.3.1. Suppose equal separation ability for all features $R_1 = \ldots = R_J = R$ ($\bar{\mathcal{I}} = \phi$) and further assume $R > \gamma$. The design of $\mathbf{m}_{gj} = \mathbb{I}\{j \in \mathcal{J}_g\}/\sqrt{h(j)}$, $w_g = \sqrt{\sum_{j \in \mathcal{J}_g} 1/h(j)}$ satisfies the "Unbiased Feature Selection" principle such that optimum solution of $\boldsymbol{z}$ from Equation 3.3.1 generates $z_j = 1/\sqrt{J}$, $\forall j$.*

Theorem 3.3.1 gives a design of overlapping group lasso penalty such that given equal separation ability for all features, the feature selection is not biased by the prior group knowledge. When all the groups are non-overlapping, $h(j) = 1, \forall j$, then

$$\Omega(\mathbf{z}) = \sum_{0 \leq g \leq \mathcal{G}_0} \left(\sqrt{|\mathcal{J}_g|} \sqrt{\sum_{j \in \mathcal{J}_g} z_j^2} \right),$$

where $|\mathcal{J}_g|$ is number of features in group $\mathcal{J}_g$, which is the non-overlapping group lasso penalty (Yuan and Lin, 2006). However, this weight design $\left(w_g = \sqrt{\sum_{j \in \mathcal{J}_g} 1/h(j)}\right)$ is not applicable when the underlying intrinsic feature set is sparse (i.e. $\bar{\mathcal{I}} \neq \phi$). If there are many non-intrinsic features inside group g, the intrinsic features in group g is over penalized since w_g is inflated by the contribution of non-intrinsic features. Therefore, we propose the following overlapping group lasso penalty and show that the design satisfies "Unbiased Feature Selection" principle when intrinsic feature set is sparse.

$$\begin{aligned} \mathbf{m}_{gj} &= \mathbb{I}\{j \in \mathcal{J}_g\}/\sqrt{h(j)} \\ w_g &= \sqrt{\sum_{j \in (\mathcal{J}_g \cap \mathcal{I})} 1/h(j)} \end{aligned} \tag{3.3.2}$$

Theorem 3.3.2. *Suppose the intrinsic feature set $\mathcal{I} = \{j : R_j = R > 0\}$ and the non-intrinsic feature set $\bar{\mathcal{I}} = \{j : R_j = 0\}$. We further assume $R > \gamma$. The overlapping group lasso penalty in Equation 3.3.2 satisfies the "Unbiased Feature Selection" principle such that the optimum solution of $\boldsymbol{z}$ from Equation 3.3.1 is $z_j = 1/\sqrt{|\mathcal{I}|}$ for $j \in \mathcal{I}$ and $z_j = 0$ for $j \in \bar{\mathcal{I}}$.*

Note that we take into account both the non-intrinsic features and the intrinsic features in the penalty design in Equation 3.3.2. Only intrinsic features contribute to the group weight coefficient w_g. The design vector $\mathbf{m}_g$ remains the same as non-overlapping group lasso. In practice, the intrinsic feature set $\mathcal{I}$ is unknown. We follow the coefficient design of adaptive lasso (Zou, 2006) and adaptive group lasso (Huang et al., 2010), which have been discussed in the literature and they maintain consistency property under certain mild conditions. Specifically, we set $\alpha = 1$ in Equation 3.3.1 where only individual feature penalty is considered and use the solution $\hat{\mathbf{z}}$ to define estimated intrinsic feature set $\hat{\mathcal{I}} = \{j : \hat{\mathbf{z}}_j > 0\}$ and non-intrinsic feature set $\hat{\bar{\mathcal{I}}} = \{j : \hat{\mathbf{z}}_j = 0\}$ for Equation 3.3.2. In the example of Figure 10C, suppose all 7 features are intrinsic genes. Pathway $\mathcal{J}_1$ contains mRNA1, mRNA2, mRNA3 and mRNA6, reflecting prior knowledge from pathway databases.

Similarly, group for pathway $\mathcal{J}_2$ contains mRNA3, mRNA4, mRNA5 and mRNA7. As a result, $\mathbf{m}_1 = (1, 1, 1/2, 0, 0, 1, 0)$ and $\mathbf{m}_2 = (0, 0, 1/2, 1, 1, 0, 1)$ and

$$\begin{aligned}\Omega(\mathbf{z}) =& \sqrt{1+1+1/2+1}\sqrt{z_1^2 + z_2^2 + 1/2 \times z_3^2 + z_6^2} + \\ & \sqrt{1/2+1+1+1}\sqrt{1/2 \times z_3^2 + z_4^2 + z_5^2 + z_7^2}.\end{aligned}$$

Note that in our example mRNA3 is shared by pathway groups $\mathcal{J}_1$ and $\mathcal{J}_2$, representing overlapping group lasso penalty.

3.3.3 Optimization

In this section, we discuss major issues for optimization of Equation 3.3.1. Firstly we introduce transformation of Equation 3.3.1 such that l_1 norm penalty can be absorbed in l_2 norm group penalty. Secondly we introduce the optimization procedure for the proposed objective function. Thirdly, we discuss how to use ADMM to optimize the weight term, which is critical and a difficult problem since it involves both the l_1 norm penalty and overlapping group lasso penalty. Lastly, we discuss the stopping rule for the optimization.

3.3.3.1 Reformulation and iterative optimization We use the fact that $\gamma\alpha\|\mathbf{z}\|_1$ can be re-written as $\gamma\alpha\|\mathbf{z}\|_1 = \gamma\alpha\sum_{j=1}^{J}\|\mathbf{z_j}\|_2$ and $\mathbf{z_j} = (0, \ldots, z_j, \ldots, 0)^\top$ with only the j^{th} element non-zero. In other words, the l_1 norm penalty of a single feature can be deemed as group penalty with only one feature within a group. Therefore we can rewrite objective function Equation 3.3.1 as

$$\min_{C,\mathbf{z}} -\sum_{j=1}^{J} z_j R_j(C) + \sum_{j=1}^{J}\|\gamma\alpha\boldsymbol{\phi_j}\circ\mathbf{z}\|_2 + \sum_{0\le g\le\mathcal{G}_0}\|\gamma(1-\alpha)\mathbf{m}_g\circ\mathbf{z}\|_2 \tag{3.3.3}$$

s.t. $\|\mathbf{z}\|_2 \le 1$, $z_j \ge 0$, where $\boldsymbol{\phi_j} = (\boldsymbol{\phi_j}_1, \ldots, \boldsymbol{\phi_j}_J)$, $\boldsymbol{\phi_j}_i = 1$ if $j = i$ and $\boldsymbol{\phi_j}_i = 0$ if $j \ne i$. We combine J and $\mathcal{G}_0$ groups and the combined groups are of size $\mathcal{G} = J + \mathcal{G}_0$. Define

$$\boldsymbol{\beta}_g = \begin{cases}\gamma\alpha\boldsymbol{\phi_j}, & \text{if } 1 \le g \le J, \\ \gamma(1-\alpha)\mathbf{m}_g, & \text{if } J+1 \le g \le \mathcal{G}.\end{cases}$$

Therefore we can rewrite objective function Equation 3.3.3 as

$$\begin{aligned} &\min -\mathbf{R}(C)^\top \mathbf{z} + \sum_{1\le g\le \mathcal{G}} \|\boldsymbol{\beta}_g \circ \mathbf{z}\|_2 \\ &\text{subject to } \|\mathbf{z}\|_2 \le 1, z_j \ge 0, \end{aligned} \tag{3.3.4}$$

where $\mathbf{R}(C) = (R_1(C), \ldots, R_J(C))^\top$. The optimization procedure are outlined below:

1. Initialize weight $\mathbf{z}$ using the original sparse K-means method without the group lasso term.
2. Given weight $\mathbf{z}$, use weighted K-means to update cluster labels C ($\mathbf{R}$ is the normalized WCSS so minimizing $-\mathbf{R}(C)^\top \mathbf{z}$ is essentially weighed K-means). This is a non-convex problem so multiple random starts are recommended to alleviate local minimum problem.

3. Given the cluster label C, $\mathbf{R}$ is fixed so optimizing the objective function is a convex problem with respect to solving weight $\mathbf{z}$. We use ADMM in the next subsection to update weight $\mathbf{z}$.
4. Iterate 2 and 3 until converge.

The detailed algorithm for Step 3 is outlined in Section 3.3.3.2 and the stopping rules of Step 3 and Step 4 are described in Section 3.3.3.3.

3.3.3.2 Update weight using ADMM Alternating direction method of multiplier (ADMM) (Boyd et al., 2011) is ideal for solving the optimization in Equation 3.3.4. We introduce an auxiliary variable $\mathbf{x}_g$ and write down the augmented Lagrange.

$$\min -\mathbf{R}(C)^\top \mathbf{z} + \sum_{1\le g\le \mathcal{G}} \|\mathbf{x}_g\|_2 + \sum_{1\le g\le \mathcal{G}} \{\mathbf{y}_g^\top (\mathbf{x}_g - \boldsymbol{\beta}_g \circ \mathbf{z}) + \frac{\rho}{2}\|\mathbf{x}_g - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2\} \tag{3.3.5}$$

s.t. $\|\mathbf{z}\|_2 \le 1$, $z_j \ge 0$, and $\mathbf{x}_g = \boldsymbol{\beta}_g \circ \mathbf{z}$. This problem (Equation 3.3.5) is clearly equivalent to the original objective function (Equation 3.3.4), since for any feasible $\mathbf{z}$ the terms added to the objective is zero. ρ is the augmented Lagrange parameter which will be discussed in more detail in Section 3.3.3.4. Here the augmented Lagrange is minimized jointly with respect to the two primal variables $\mathbf{x}_g$, $\mathbf{z}$ and the dual variable $\mathbf{y}_g$. In ADMM, $\mathbf{x}_g$, $\mathbf{z}$ and $\mathbf{y}_g$ are updated in an alternating or sequential fashion (Boyd et al., 2011) and thus the optimization problem

can be decomposed into three parts. Given ($\mathbf{x}_g$, $\mathbf{z}$ and $\mathbf{y}_g$), the new iteration of ($\mathbf{x}_g^+$, $\mathbf{z}^+$ and $\mathbf{y}_g^+$) in Equation 3.3.5 is updated as following.

$$\begin{cases} \mathbf{x}_g^+ = \arg\min_{\mathbf{x}_g} \|\mathbf{x}_g\|_2 + \mathbf{y}_g^\top \mathbf{x}_g + \frac{\rho}{2}\|\mathbf{x}_g - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2 \\ \mathbf{z}^+ = \arg\min_{\mathbf{z}} -\sum z_j R_j - \sum_{1\le g\le \mathcal{G}} \mathbf{y}_g^\top (\boldsymbol{\beta}_g \circ \mathbf{z}) + \frac{\rho}{2}\|\mathbf{x}_g^+ - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2 \\ \quad \text{subject to } \|\mathbf{z}\|_2 \le 1, z_j \ge 0. \\ \mathbf{y}_g^+ = \mathbf{y}_g + \rho(\mathbf{x}_g^+ - \boldsymbol{\beta}_g \circ \mathbf{z}^+) \end{cases}$$

Where the updating equation of $\mathbf{x}_g^+$ and $\mathbf{z}^+$ are derived from Equation 3.3.5 and the the updating equation of $\mathbf{y}_g^+$ is imbedded in ADMM procedure (Boyd et al., 2011). We can derive close form solution for $\mathbf{x}_g$ part and $\mathbf{z}$ part by Karush-Kuhn-Tucker (KKT) condition. Details are given in the Appendix.

1. Define $\mathbf{a}_g = \boldsymbol{\beta}_g \circ \mathbf{z} - \frac{\mathbf{y}_g}{\rho}$, we have $\mathbf{x}_g^+ = (1 - \frac{1}{\rho\|\mathbf{a}_g\|_2})_+ \mathbf{a}_g$, where $(\cdot)_+ = \max(0, \cdot)$.
2. Define $b_j = \sum_{1\le g\le \mathcal{G}} \rho \boldsymbol{\beta}_{gj}^2$ and $c_j = \sum_{1\le g\le \mathcal{G}} \left(\rho \mathbf{x}_{gj}^+ + \mathbf{y}_{gj}\right) \circ \boldsymbol{\beta}_{gj}$, where $\boldsymbol{\beta}_g = (\boldsymbol{\beta}_{g1}, \boldsymbol{\beta}_{g2}, \ldots, \boldsymbol{\beta}_{gJ})^\top$, $\mathbf{x}_g = (\mathbf{x}_{g1}, \mathbf{x}_{g2}, \ldots, \mathbf{x}_{gJ})^\top$ and $\mathbf{y}_g = (\mathbf{y}_{g1}, \mathbf{y}_{g2}, \ldots, \mathbf{y}_{gJ})^\top$. The solution is given as following: we define $f_j(u) = (\frac{R_j + c_j}{b_j + 2u})_+$. If $\sum_j f_j(u)^2 < 1$, $z_j^+ = f_j(0)$ $\forall j$. Otherwise $z_j^+ = f_j(u)$ $\forall j$ and u is selected s.t. $\|\mathbf{z}^+\|_2 = 1$.

3.3.3.3 Stopping rules We have two algorithms which require stopping rules. For ADMM in the optimization of Step 3, the primal residual of group g in ADMM iteration t is: $\mathbf{r}_g^t = \mathbf{x}_g^t - \boldsymbol{\beta}_g \circ \mathbf{z}^t$, and the l_2 norm of primal residual is $r^t = \sqrt{\sum_g \|\mathbf{r}_g^t\|_2^2}$. The l_2 norm of dual residual is: $v^t = \sqrt{\sum_g \|\boldsymbol{\beta}_g \circ (\mathbf{z}^t - \mathbf{z}^{t-1})\|_2^2}$. We set our ADMM stopping criteria such that simultaneously $r^t < 10^{-10}$ and $v^t < 10^{-10}$. For convergence of IS-Kmeans, we iterate weighted K-means (Step 2) and updating weight by ADMM (Step 3) until converge. (i.e. $\frac{\sum_{j=1}^J |z_j^{(c)} - z_j^{(c-1)}|}{\sum_{j=1}^J |z_j^{(c-1)}|} < 10^{-4}$), where $z_j^{(c)}$ represents the z_j estimate in the c^{th} iteration of the IS-Kmeans algorithm.

3.3.3.4 augmented Lagrangian parameter ρ Augmented Lagrangian parameter ρ controls the convergence of ADMM. In fact, large value of ρ will lead to small primal residual by placing a large penalty on violations of primal feasibility. And conversely, small value of ρ tend to produce small dual residual, but it will result in a large primal residual by reducing the penalty on primal feasibility (Boyd et al., 2011). An adaptive scheme of varying ρ to balance the primal and dual residual has been proposed (He et al., 2000; Wang and Liao, 2001) which greatly accelerates ADMM convergence in practice.

$$\rho^{t+1} = \begin{cases} \tau^{\text{incr}}\rho^t, & \text{if } \|r^t\|_2 > \eta\|v^t\|_2, \\ \rho^t/\tau^{\text{decr}}, & \text{if } \|v^t\|_2 > \eta\|r^t\|_2, \\ \rho^t, & \text{otherwise.} \end{cases}$$

We set $\eta = 10$ and $\tau^{\text{incr}} = \tau^{\text{decr}} = 2$. The intuition behind this scheme is to control both primal and dual residuals for converging to zero simultaneously.

3.3.4 Select tuning parameters

In the objective function of IS-Kmeans, the number of clusters K is pre-specified. The issue of estimating K has been widely discussed in the literature and has been well-recognized as a difficult and data-dependent problem. (Milligan and Cooper, 1985; Kaufman and Rousseeuw, 2009). Here, we suggest the number of clusters to be estimated in each study separately using conventional methods such as prediction strength (Tibshirani and Walther, 2005) or gap statistics (Tibshirani et al., 2001) and jointly compared across studies (such that the numbers of clusters are roughly the same for all studies) for a final decision before applying integrative sparse K-means. Below we assume that a common K is pre-estimated for all omics datasets.

Another important parameter to be determined is α, which controls the balance between individual feature penalty and overlapping group penalty. According to the Equation 3.3.1, $\alpha = 1$ means we only emphasize on individual feature penalty and ignore overlapping group penalty. In this case the IS-Kmeans is equivalent to sparse K-means. $\alpha = 0$ means we only emphasize overlapping group penalty and ignore individual feature penalty. Simon et al. (2013) argued that there is no theoretically optimal selection for α because selection of α

relates to multiple factors such as accuracy of prior group information and sparsity within groups. In general, a large α (e.g. $\alpha = 0.95$) is suitable when prior group information may not be accurate or features within selected groups may be sparse. On the other hand, if we expect mild sparsity within groups and high accuracy of prior group information, a small α (e.g. $\alpha = 0.05$) help select features by groups. In Section 3.4.1.2, we have performed simulation of different level of prior group information accuracy ($\theta = 1$ and $\theta = 0.2$) and found that $\alpha = 0.5$ generates robust and high performance results in the sensitivity analysis. As a result, we apply $\alpha = 0.5$ throughout the chapter unless otherwise indicated.

The last tuning parameter is γ, which is the penalty coefficient. When γ is large, we place large penalty on the objective function and end up with less selected features. When γ is small, we put small penalty and will include more features. We follow and extend the gap statistic procedure (Tibshirani et al., 2001) to estimate γ:

1. For each feature in each omics type, randomly permute the gene expression (permute samples). This creates a permuted data set $X^{(1)}$. Repeat for B times to generate $X^{(1)}, X^{(2)}, \ldots, X^{(B)}$.
2. For each potential tuning parameter γ, compute the gap statistics as below.

$$\text{Gap}(\gamma) = O(\gamma) - \frac{1}{B}\sum_{b=1}^{B} O_b(\gamma), \tag{3.3.6}$$

   where $O(\gamma) = -\sum_{j=1}^{J} z_j^* R_j(C^*)$ is from observed data, where $\mathbf{z}^*, C^*$ are the minimizer of the objective function in Equation 3.3.1 given γ. $O_b(\gamma)$ is similar to $O(\gamma)$ but generated from permuted data $X^{(b)}$. Note that for
3. For a range of selections of γ, select γ^* such that the gap statistics in Equation 3.3.6 is minimized.

Figure 11 shows an example of a simulated dataset that will be discussed in Section 3.4.1. In this example, we used $\alpha = 0.5$ for IS-Kmeans and the minimum gap statistics corresponded to 1778 genes, which is very close to the underlying truth 1800. The gap statistics for $\alpha = 0.05, 0.95, 1$ are plotted in Supplementary Figure 13 and they all provided adequate γ estimation. In practice, calculating gap statistics from a chain of γ can be performed efficiently by adopting warm start for adjacent γ's. For example, after calculating $O(\gamma_1)$, the

resulting weights can be used as an initial value for the next nearby $\gamma_2 = \gamma_1 + \Delta$ to calculate $O(\gamma_2)$ in the optimization iteration for fast convergence.

3.4 RESULT

We evaluated integrative sparse K-means (IS-Kmeans) on simulation datasets in Section 3.4.1, multiple-level omics applications using breast cancer TCGA (combining mRNA expression, DNA methylation and copy number variation) and METABRIC (combining mRNA expression and copy number variation) examples in Section 3.4.2 and 3.4.3, and a pathway-guided single transcriptomic application in leukemia in Section 3.4.4. In the simulation, the underlying sample clusters and intrinsic feature set were known and we demonstrated the better performance of IS-Kmeans compared to iCluster and sparse K-means by cluster accuracy, feature selection and computing time. For the TCGA and METABRIC application, the underlying true clustering and intrinsic feature set were not known. We evaluated the performance by clustering similarity using adjusted Rand index (ARI) (Hubert and Arabie, 1985) with subtype definition by PAM50 (Parker et al., 2009), cis-regulatory groups, survival difference between clusters and computing time. In the leukemia examples, the disease subtypes were defined by observable fusion gene aberration. We evaluated the performance by clustering accuracy (ARI) and pathway enrichment analysis on selected genes.

3.4.1 Simulation

3.4.1.1 Simulation setting To assess the performance of integrative sparse K-means with different choices of α and compare to the original sparse K-means and iCluster, we simulated $K = 3$ subtypes characterized by several groups of subtype predictive genes in each of $S = 2$ omics datasets with $1 \leq s \leq S$ as the omics dataset index (e.g. $s = 1$ represents gene expression and $s = 2$ represents DNA methylation). The prior group information was imposed between groups of subtype predictive genes across omics datasets. These prior group

information represent the possibility that a group of genes and DNA methylations might be co-regulated. To best preserve the data nature of genomic studies, we also simulated confounding variables, correlated gene structure and non-informative genes. Below is the generative process:

(a) Subtype predictive genes (intrinsic feature set).

1. Denote by N_k is the number of subjects in subtype $k(1 \leq k \leq 3)$. We simulate $N_1 \sim$ POI(40), $N_2 \sim$ POI(40), $N_3 \sim$ POI(30) and the number of subjects is $N = \sum_k N_k$. Simulate $S = 2$ omics datasets, which share the samples and subtypes. Specifically, we denote $s = 1$ to be the gene expression dataset and $s = 2$ to be the DNA methylation dataset.
2. Simulate $M = 30$ feature modules $(1 \leq m \leq M)$ for each omics dataset. Denote n_{sm} to be the number of features in omics dataset s and module m. For each module in $s = 1$, sample $n_{1m} = 30$ genes. For each module in $s = 2$, sample $n_{2m} = 30$ methylations. Therefore, there will be of 1800 subtype predictive features among two omics datasets.
3. Denote by μ_{skm} is the template gene expression (on log scale) of omics dataset $s(1 \leq s \leq S)$, subtype $k(1 \leq k \leq 3)$ and module $m(1 \leq m \leq M)$. Simulate the template gene expression $\mu_{skm} \sim \mathrm{N}(9, 2^2)$ with constrain $\max_{p,q} |\mu_{spm} - \mu_{sqm}| \geq 1$, where p, q denote two subtypes. This part defines the subtype mean intensity for each module in all omics datasets. Note that since in Equation 3.3.1 we used $R_j = \frac{BCSS_j}{TSS_j}$ for standardization, performance of the algorithm is robust to gene expression distribution (e.g. the Gaussian assumption here).
4. In order to tune the signal of the template gene expression, we introduce a relative effect size $f > 0$, such that $\mu'_{skm} = (\mu_{skm} - \min_k \mu_{skm}) \times f + \min_k \mu_{skm}$. If $f = 1$, we don't tune the signal. If $f < 1$, we decrease the signal and if $f > 1$, we amplify the signal.
5. Add biological variation $\sigma_1^2 = 1$ to the template gene expression and simulate $X'_{skmi} \sim \mathrm{N}(\mu'_{skm}, \sigma_1^2)$ for each module m, subject $i(1 \leq i \leq N_k)$ of subtype k and omics dataset s.

6. Simulate the covariance matrix Σ_{mks} for genes in module m, subtype k and omics dataset s, where $1 \leq m \leq M$, $1 \leq k \leq 3$ and $1 \leq s \leq S$. First simulate $\Sigma'_{mks} \sim \mathrm{W}^{-1}(\Phi, 100)$, where $\Phi = 0.5I_{n_{sm} \times n_{sm}} + 0.5J_{n_{sm} \times n_{sm}}$, W^{-1} denotes the inverse Wishart distribution, I is the identity matrix and J is the matrix with all elements equal 1. Then Σ_{mks} is calculated by standardizing Σ'_{mks} such that the diagonal elements are all 1's.
7. Simulate gene expression levels of genes in cluster m as $(X_{1skmi}, \ldots, X_{n_{sm}skmi})^\top \sim \mathrm{MVN}(X'_{skmi}, \Sigma_{mks})$, where $1 \leq i \leq N_{ks}$, $1 \leq m \leq M$, $1 \leq k \leq 3$ and $1 \leq s \leq S$.

(b) Non-informative genes.

1. Simulate 5000 non-informative genes denoted by $g(1 \leq g \leq 5000)$ in each omics dataset. First, we generate the mean template gene expression $\mu_{sg} \sim \mathrm{N}(9, 2^2)$. Then we add biological variance $\sigma_2^2 = 1$ to generate $X_{sgi} \sim N(\mu_{sg}, \sigma_2^2), 1 \leq i \leq N_s$.

(c) Confounder impacted genes.

1. Simulate $C = 2$ confounding variables. In practice, confounding variables can be gender, race, other demographic factors or disease stage etc. These will add heterogeneity to each study to complicate disease subtype discovery. For each confounding variable $c(1 \leq c \leq C)$, we simulate $R = 10$ modules in each omics dataset. For each of these modules $r_c(1 \leq r_c \leq R)$, sample number of genes $n_{r_c} = 30$. Therefore, totally 600 confounder impacted genes are generated in each omics dataset. This procedure is repeated in all S omics datasets.
2. For each omics dataset $s(1 \leq s \leq S)$ and each confounding variable c, sample the number of confounder subclass $h_{sc} = k$. The N samples in omics dataset s will be randomly divided into h_{sc} subclasses.
3. Simulate confounding template gene expression $\mu_{slrc} \sim \mathrm{N}(9, 2^2)$ for confounder c, gene module r, subclass $l(1 \leq l \leq h_{sc})$ and omics dataset s. Similar to Step 5, we add biological variation σ_1^2 to the confounding template gene expression $X'_{scrli} \sim \mathrm{N}(\mu_{slrc}, \sigma_1^2)$. Similar to Step 6 and 7, we simulate gene correlation structure within modules of confounder impacted genes.

(d) Gene grouping information.

1. We assume omics dataset $s = 1$ and $s = 2$ have prior group information on subtype predictive gene modules. There are $M = 30$ modules in each omics dataset.
2. Suppose subtype predictive genes in the m^{th} module of the first omics dataset are grouped with methylation features in the second omics dataset (totally $n_{1m} + n_{2m} = 30 + 30 = 60$ features are in the same group). With probability $1 - \theta$ ($0 \leq \theta \leq 1$), each feature out of the 60 features will be randomly replaced by a confounder impacted gene or non-informative gene. Note that the same replaced feature can appear in multiple subtype predictive gene groups. We set $\theta = 1$ and 0.2 to reflect 100%, 20% accuracy of prior group information.

3.4.1.2 Simulation result For IS-Kmeans, the tuning parameter γ was selected by gap statistics introduced in Section 3.3.4. Table 8 shows the result of gap statistics to select the best γ in the simulation of $\alpha = 0.5$, $\theta = 1$. The smallest gap statistics was selected at $\gamma = 0.21$ that correspond to selecting 1778 features, which was close to the underlying truth. Similarly, gap statistics result for $\alpha = 1, 0.95, 0.05$ are in the Supplementary Figure 13. For simulation, we generated two scenarios with relative effect size $f = 0.6$ and $f = 0.8$. The complete simulation result of $f = 0.6$ is shown in Table 8 and the result for $f = 0.8$ is in the supplementary materials supplementary Table 13. For iCluster and sparse K-means, we allowed them to choose their own optimum tuning parameters. Note that sparse K-means was adopted to each individual omics datatype. We used ARI (Hubert and Arabie, 1985) and Jaccard index (Jaccard, 1901) to evaluate the clustering and feature selection performance. ARI calculated similarity of the clustering result with the underlying true clustering in simulation (range from -1 to 1 and 1 represents exact same partition compared to the underlying truth). Jaccard index compared the similarity and diversity of two feature sets, defined as the size of the intersection of two feature sets divided by the size of the union of two feature sets (range from 0 to 1 and 1 represent identical feature sets compared to the underlying truth). Clearly, IS-Kmeans outperformed iCluster and individual study sparse K-means in terms of ARI and Jaccard index. IS-Kmeans and sparse K-means outperformed iCluster in terms of computing time. Within IS-Kmeans, we compared feature selection in terms of area under the curve (AUC) of ROC curve, which avoids the issue of tuning

parameter selection. When $\theta = 1$ (representing the grouping information is accurate), smaller α (representing larger emphasize on grouping information) yielded better feature selection performance in terms of AUC as expected. However, when $\theta = 0.2$ (representing many errors in the grouping information), smaller α yielded worse performance in terms of AUC. Note that $\alpha = 0.5$ gives robustness and performs well in the two extremes of $\theta = 1$ and $\theta = 0.2$. In all applications below, we will apply $\alpha = 0.5$ unless otherwise noted.

3.4.1.3 Data perturbation We also evaluated the stability of the algorithm against data perturbation. Instead of Gaussian distribution in the data generative process, we utilized heavy tailed t-distribution to generate the expression. In the simulation setting Step a3, the template gene expression is simulated from a t-distribution with degree of freedom 3, location parameter 9 and scale parameter 2. In Step a4, we set relative effect size $f = 0.6$ and $f = 0.8$ respectively. In Step a5, X'_{skmi} is simulated from a t-distribution with degree of freedom 3, location parameter μ'_{skm} and scale parameter σ_1^2. The result for data perturbation is in supplementary Table 16 and 17. The resulting message remains almost the same as the conclusion in Section 3.4.1.2. Therefore, our proposed algorithm is robust against non-Gaussian or heavy tail distributions.

3.4.2 Integrating TCGA Breast cancer mRNA, CNV and methylation

We downloaded TCGA breast cancer (BRCA) multi-level omics datasets from TCGA NIH official website. TCGA BRCA gene expression (IlluminaHiSeq RNAseqV2) was downloaded on 04/03/2015 with 20,531 genes and 1,095 subjects. TCGA BRCA DNA methylation (Methylation450) was downloaded on 09/12/2015 with 485,577 probes and 894 subjects. TCGA BRCA copy number variation (BI gistic2) was downloaded on 09/12/2015 with 24,776 genes and 1,079 subjects. There were 770 subjects with all these three omics data types. Features (probes/genes) with any missing value were removed. For gene expression, we transformed the FPKM value by $\log_2(\cdot + 1)$, where 1 is a pseudo-count to avoid undefined $\log_2(0)$, such that the transformed value was on continuous scale. For methylation, the Methylation450 platform provided beta value with range $0 < \beta < 1$, where 0 represents un-

methylated and 1 represents methylated. We transformed the beta value to M value, which is defined by a logit transformation ($M = \log_2[\frac{\beta}{1-\beta}]$). Therefore methylation characterized by M value is on continuous scale, similar to mRNA and CNV. If multiple methylation probes matched to the same gene symbol, we selected one methylation probe as representative, which had the largest average correlation with other methylation probes of the same gene symbol. We ended up with 20,147 methylation probes with unique gene symbols.

We filtered out 50% low expression genes (unexpressed genes) and then 50% low variance genes (non-informative genes). 50% low expression genes are genes with the lowest 50% mean of gene expression across samples and 10,250 genes remained after this filtering step. 50% low variance genes are genes with the lowest 50% variance of gene expression across samples and 5,125 genes remained after this filtering step. We obtained 4,815 CNV features and 5,035 methylation features by matching to the 5,125 gene symbols. The features from three different omics datasets that shared the same cis-regulatory annotation (same gene symbol) were grouped together to form 5,125 feature groups. In this case, each group had one mRNA gene expression, one CNV gene and/or one methylation probe. Each group contained candidate multi-omics regulatory information because CNV and methylation could potentially regulate mRNA expression. We applied IS-Kmeans with $\alpha = 0.5$, sparse K-means by directly merging three omics datasets together as well as iCluster. Number of clusters K was set to be 5 since it was well established that breast cancer has 5 subtypes by PAM50 definition (Parker et al., 2009). For a fair comparison, we selected the tuning parameter for each method such that number of selected features are close to 2,000.

For evaluation purpose, we investigated three categories of groups among selected features: G1, G2 and G3. G3 represents feature groups (gene symbol) where all three types (mRNA, CNV and methylation) of features are selected. Similarly, G2 represents feature groups (gene symbol) where only two types of features are selected; G1 represents feature groups (gene symbol) where only one type of feature is selected; We also compared the clustering result with PAM50 subtype definition in terms of ARI. The result is shown in Table 9. Clearly, IS-Kmeans obtained more G2 and G3 features than sparse K-means and iCluster. This is biologically more interpretable but not surprising since IS-Kmeans incorporated the multi-omics regulatory information and we expected feature of the same group were en-

couraged to come out together. Besides, IS-Kmeans has higher ARI compared to sparse K-means and iCluster, indicating the clustering result of IS-Kmeans is closer to PAM50 definition than sparse K-means and iCluster. The 5 by 5 confusion table of IS-Kmeans clustering result and PAM50 subtypes is shown in supplementary Table 14. One should note the the ARI for all these three methods are not very high. This could be because PAM50 was defined by gene expression only and in our scenario we integrated multi-omics information. The heatmaps of IS-Kmeans result is shown in Figure 10B. In terms of computing time, IS-Kmeans is nearly 20 times faster than iCluster.

3.4.3 Integrating METABRIC Breast cancer mRNA and CNV

We tested the performance of IS-Kmeans in another large breast cancer multi-omics (sample size n=1,981) dataset METABRIC (Curtis et al., 2012) with mRNA expression (llumina HumanHT12v3) and CNV (Affymetrix SNP 6.0 chip) and survival information. The datasets are available at https://www.synapse.org/#!Synapse:syn1688369/wiki/27311. There were originally 49,576 probes in gene expression. If multiple probes matched to the same gene symbol, we selected the probe with the largest IQR (interquartile range) to represent the gene. After mapping the probes to gene symbols, we obtained 19,489 mRNA expression features and 18,538 CNV features, which shared 1981 samples. After filtering out 30% low expression mRNA based on mean gene expression across samples and then 30% low variance mRNA based on variance of gene expression across samples, we ended up with 9,504 mRNA features. We obtained 8,696 CNV feature symbols by matching with mRNA feature symbols. Therefore, we had totally 18,200 features and 9,504 feature groups (share the same gene symbol) among 1,981 samples.

We applied IS-Kmeans with $\alpha = 0.5$, sparse K-means by directly merging three omics dataset together as well as iCluster. Number of clusters K was set to be 5 (same reason in TCGA). For a fair comparison, we selected the tuning parameter for each method such that number of selected features are close to 2,000. For evaluation purpose, we similarly defined two categories of groups among selected features. G2 represents feature groups (gene symbol) where both types of features are selected and G1 represents feature groups (gene symbol)

where only one type of feature is selected. We also compared the clustering result with PAM50 subtype definition in terms of ARI. The result is shown in Table 10.

Similar to the TCGA example in Section 3.4.2, IS-Kmeans obtained more G2 features than sparse K-means and iCluster. The log-rank test of survival difference for the clustering result defined by IS-Kmeans is more significant than sparse K-means and iCluster. Furthermore, IS-Kmeans has higher ARI compared to sparse K-means and iCluster, indicating the clustering result of IS-Kmeans is closer to PAM50 definition than sparse K-means and iCluster. The 5 by 5 confusion table of IS-Kmeans clustering result and PAM50 subtypes is in supplementary Table 15. In terms of computing time, IS-Kmeans and sparse K-means are much faster than iCluster.

3.4.4 Three leukemia transcriptomic datasets using pathway database as prior knowledge

In the simulations and applications so far (Section 3.4.1-3.4.3), we have focused on using cis-regulatory mechanism as grouping information for integrating multi-level omics data for sample clustering. In this subsection, we present a different but commonly encountered application of pathway-guided clustering in single transcriptomic study. Specifically, we use pathway information from databases to provide prior overlapping group information (i.e. a pathway is a group containing tens to hundreds of genes and two pathways may contain overlapping genes). A transcriptomic study is used for sample clustering with the overlapping group information. We apply IS-Kmeans to three leukemia transcriptomic datasets (Verhaak et al. (2009), Balgobind et al. (2011) and Kohlmann et al. (2008)) separately and using three pathway databases (KEGG, BioCarta and Reactome) independently, generating nine IS-Kmeans clustering results (see Table 11). Table 2 shows a summary description of the three leukemia transcriptomic studies.

We only considered samples from acute myeloid leukemia (AML) with three fusion gene subtypes: inv(16) (inversions in chromosome 16), t(15;17) (translocations between chromosome 15 and 17), t(8;21) (translocations between chromosome 8 and 21). These three gene-translocation AML subtypes have been well-studied with different survival, treatment

response and prognosis outcomes. Since the three subtypes are observable under microscope, we treated these class labels as the underlying truth to evaluate the clustering performance. The expression data for Verhaak, Balgobind ranged from around $[3.169, 15.132]$ while Kohlmann ranged in $[0, 1]$. All the datasets were downloaded directly from NCBI GEO website. Originally there were 54,613 probe sets in each study. For each study, we removed genes with any missing value in it. If multiple microarray probes matched to the same gene symbol, we selected the probe with the largest interquartile range (IQR) to represent the gene. We ended up with 20,154 unique genes in Verhaak and 20,155 unique genes in Balgobind and Kohlmann. We further filtered out 30% low expression genes in each study, which were defined as 30% of genes with the lowest mean expression. We ended up with 14,108 unique genes in each study.

We obtained the three pathway databases (BioCarta, KEGG and Reactome) from MSigDB (http://www.broadinstitute.org/gsea/msigdb/collections.jsp#C2) as the prior group information to guide feature selection in IS-Kmeans. The original pathway sizes were 217, 186 and 674 for BioCarta, KEGG and Reactome. We only kept pathways with size (number of genes inside pathway) greater or equal to 15 and less or equal to 200 after intersecting with 14,108 unique genes. After gene size restriction, we ended up with 114, 160 and 428 pathways for BioCarta, KEGG and Reactome. Note that these pathway groups have large overlaps (i.e. many genes appear in multiple pathways).

For each of the three studies, we applied IS-Kmeans (with BioCarta, KEGG and Reactome as prior group information respectively), sparse K-means and iCluster. Note that in this example, IS-Kmeans dealt with single omics dataset with prior knowledge. For a fair comparison, we tuned the parameters so that the number of selected features are close to 1,000. The result is shown in Table 11. For Verhaak and Kohlmann, IS-Kmeans and sparse K-means almost recovered the underlying true clustering labels (ARI=0.901-0.932), while iCluster had relatively smaller ARI (ARI=0.733). We investigated the heatmap of the clustering result of Verhaak using iCluster (supplementary Figure 15) to understand reasons of its worse performance (lower ARI) and found that its solution converged to a stable clustering configuration with clear clustering separation. Thus, the worse clustering performance in iCluster likely comes from a local optimum solution. For Balgobind, the

clustering results from IS-Kmeans and sparse K-means had smaller ARI (ARI=0.792) but iCluster performed even worse (ARI=0.214).

To further evaluate functional annotation of the selected intrinsic genes via each method, we explored pathway enrichment analysis (Figure 12) using BioCarta database via Fisher's exact test. Five methods (iCluster, IS-Kmeans (BioCarta), IS-Kmeans (KEGG), IS-Kmeans (Reactome), sparse K-means) were compared. Jittered plot of $-\log_{10}$ p-values are shown in Figure 12. IS-Kmeans (BioCarta) shows the most significant pathways consistently across three studies, this is somewhat expected since we used BioCarta pathway as prior knowledge to guide our feature selection. IS-Kmeans (KEGG) and IS-Kmeans (Reactome) also showed more significant pathways than sparse K-means and iCluster, indicating incorporating prior knowledge indeed improved feature selection (in the sense that the selected feature are more biological meaningful). Note that IS-Kmeans (KEGG) and IS-Kmeans (Reactome) did not have overfitting issue since the test pathway database (BioCarta) was different from the prior knowledge we utilized. Similarly, the results using KEGG and Reactome as testing pathway are in supplementary Figure 15.

3.5 CONCLUSION AND DISCUSSION

Cancer subtype discovery is a critical step for personalized treatment of the disease. In the era of massive omics datasets and biological knowledge, how to effectively integrate omics datasets and/or incorporate existing biological evidence brings new statistical and computational challenges. In this dissertation, we proposed an integrative sparse K-means (IS-Kmeans) approach for this purpose. The existing biological information is incorporated in the model and the resulting sparse features can be further used to characterize the cancer subtype properties in clinical application.

Our proposed IS-Kmeans has the following advantages. Firstly, integrative analysis increases clustering accuracy, statistical power and explainable regulatory flow between different omics types of data. The existing biological information is taken into account by using overlapping group lasso. Fully utilizing the inter-omics regulatory information and

external biological information will increase the accuracy and interpretation of the cancer subtype findings. Secondly, we reformulated the complex objective function into a simplified form where weighted K-means and ADMM can be iteratively applied to optimize the convex sub-problems with closed form solutions. Due to the nature of classification EM algorithm in K-means and close form iteration updates of ADMM, implementation of the IS-Kmeans framework is computationally efficient. IS-Kmeans only takes 10-15 minutes for 15,000 omics features and more than 700 subjects on a standard desktop with single computing thread while iCluster takes almost 4 hours. Thirdly, the resulting sparse features from IS-Kmeans have better interpretation than features selected from iCluster.

IS-Kmeans potentially has the following limitations. The existing biological information is prone to errors and can be updated frequently. Incorporating false biological information may dilute information contained in the data and even lead to biased finding. Therefore, we suggest not to over-weigh the overlapping group lasso term and choose $\alpha = 0.5$ to adjust for the balance between information from existing biological knowledge and information from the omics datasets. The users can, however, tune this parameter depending on the strength of their prior belief of the biological knowledge. Another limitation is that IS-Kmeans can only deal with one cohort with multiple types of omics data. How to effectively combine multiple cohorts with multi-level omics data will be a future work. R package "ISKmeans" incorporates C++ for fast computing and it is publicly available on GitHub https://github.com/Caleb-Huo/IS-Kmeans as well as authors' websites. All the data and code presented in this dissertation are also available on authors' websites.

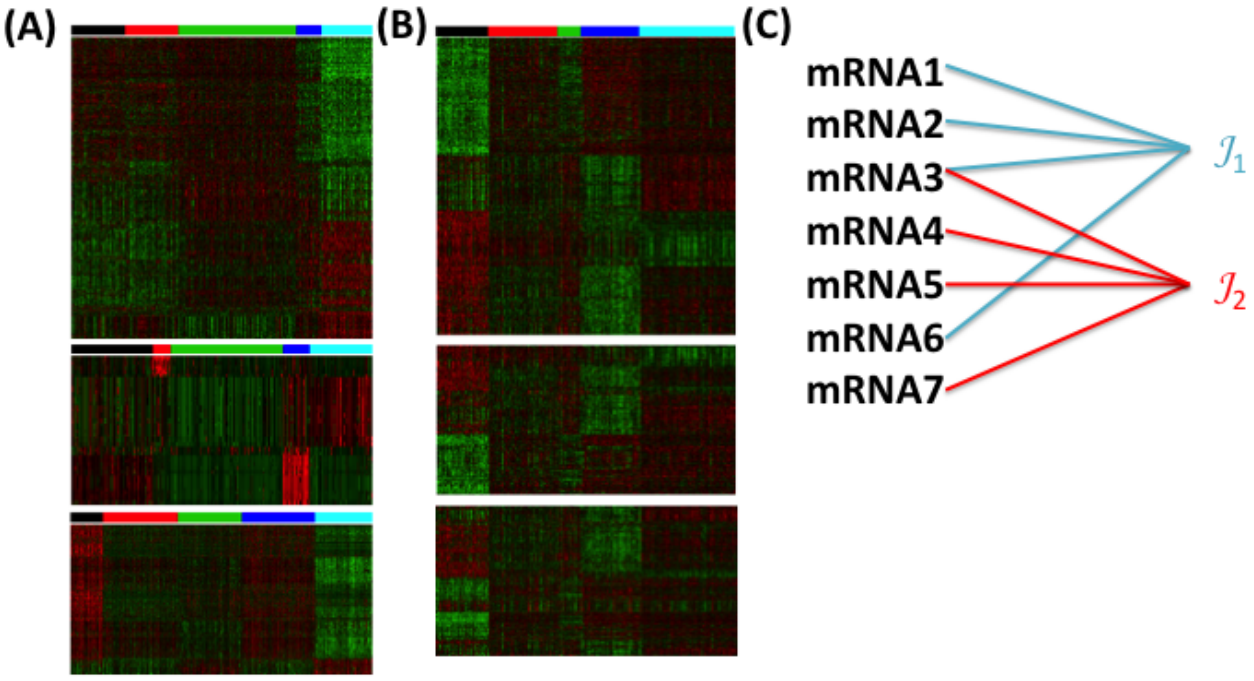


Figure 10: Illustration of IS-Kmeans.

(A) Clustering of mRNA (upper heatmap) CNV (middle heatmap) and methylation (lower heatmap) profiles separately results in different five clusters of breast cancer subtypes (represented by color bars of five colors). (B) IS-Kmeans merges mRNA (upper heatmap) CNV (middle heatmap) and methylation (lower heatmap) and perform sample clustering. Inter-omics biological knowledge is also taken into account by overlapping group lasso. (C) An illustrating example of design of overlapping group lasso penalty term $\Omega(\mathbf{z})$ to incorporate prior knowledge of pathway information. Here $\Omega(\mathbf{z}) = \sqrt{1+1+1/2+1}\sqrt{z_1^2+z_2^2+1/2\times z_3^2+z_6^2}+\sqrt{1/2+1+1+1}\sqrt{1/2\times z_3^2+z_4^2+z_5^2+z_7^2}$.

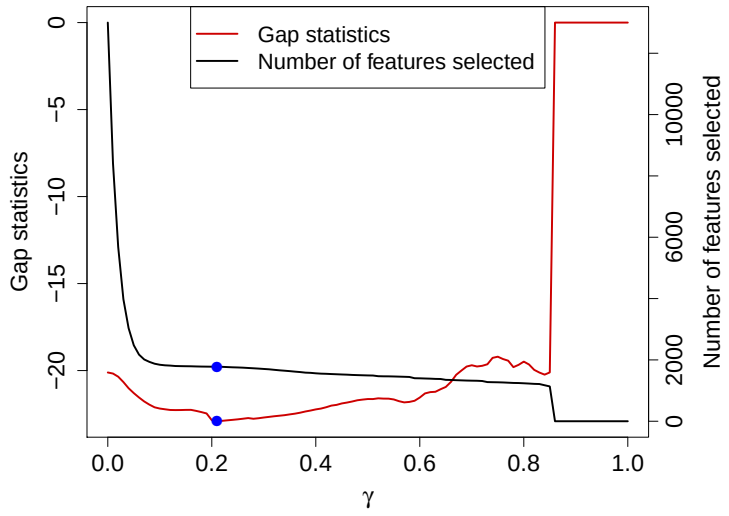


Figure 11: Selection of tuning parameter γ.

This figure was from the simulated dataset in Section 3.4.1 with $\alpha = 0.5$. X-axis represents tuning parameter γ. Red curve and left y-axis denote the corresponding gap statistics. Black curve and right y-axis denote the corresponding number of selected features. The blue dots ($\gamma = 0.21$) represent where the gap statistics is minimized, and the corresponding number of selected feature is 1778.

Table 8: Comparison table of simulation with relative effect size $f = 0.6$.

θ	method	α	ARI	Jaccard index	AUC	# features	time [mins]
1	IS-Kmeans	1	0.940 (0.239)	0.781 (0.202)	0.943 (0.138)	1465	0.44
		0.95	0.940 (0.239)	0.791 (0.204)	0.945 (0.136)	1483	0.52
		0.5	0.940 (0.239)	0.779 (0.202)	0.971 (0.084)	1420	0.56
		0.05	0.940 (0.239)	0.946 (0.214)	0.997 (0.012)	1723	0.67
0.2	IS-Kmeans	1	0.940 (0.239)	0.781 (0.202)	0.943 (0.138)	1465	0.44
		0.95	0.940 (0.239)	0.783 (0.202)	0.943 (0.138)	1469	0.57
		0.5	0.940 (0.239)	0.602 (0.159)	0.943 (0.134)	1105	0.57
		0.05	0.940 (0.239)	0.467 (0.096)	0.888 (0.111)	2824	1.2
	iCluster		0.374 (0.323)	0.383 (0.274)		1239	26
	sparse Kmeans 1		0.312 (0.370)	0.105 (0.101)		896	0.12
	sparse Kmeans 2		0.361 (0.424)	0.204 (0.124)		2137	0.13

We simulated $B = 100$ times and calculated mean and standard deviation of each quantity. θ denotes the probability grouping information is correct for each feature inside groups. α is the tuning parameter balancing the emphasis between individual penalty and group penalty. For each method, we allow its own tuning parameter selection method to optimize its performance.

Table 9: Comparison of different methods using TCGA breast cancer (K=5).

method	ARI	nfeature	G1	G2	G3	time
ISKmeans	0.379	2066	843	538	49	12.1 mins
SparseKmeans	0.332	2034	1466	284	0	6.85 mins
iCluster	0.272	2475	1725	375	0	3.91 hours

G3 represents feature groups (gene symbol) where all three types of features are selected. Similarly, G2 represents feature groups (gene symbol) where only two types of features are selected; G1 represents feature groups (gene symbol) where only one type of feature is selected; We also compared the clustering result with PAM50 subtype definition in terms of ARI.

Table 10: Comparison of different methods using metabric breast cancer (K=5).

method	ARI	nfeature	G1	G2	p value	time
ISKmeans	0.233	1882	1494	194	8.29×10^{-17}	38.4 mins
SparseKmeans	0.22	2004	2004	0	3.04×10^{-13}	34.3 mins
iCluster	0.0572	2471	2471	0	0.143	11.8 hours

G2 represents feature groups (gene symbol) where all two types of features are selected; G1 represents feature groups (gene symbol) where only one type of feature is selected; Clustering result is compared with PAM50 subtype definition in terms of ARI. Survival p-value obtained from log rank test are given for clustering assignment for each method.

Table 11: Comparison of different methods by ARI for IS-Kmeans

		Verhaak		Kohlmann		Balgobind	
method	pathway	# features	ARI	# features	ARI	# features	ARI
IS-Kmeans	Biocarta	1009	0.932	1000	0.948	999	0.792
	KEGG	1002	0.901	1013	0.948	990	0.792
	Reactome	993	0.932	994	0.948	1008	0.792
iCluster		982	0.733	1233	0.504	1020	0.214
sparse K-means		992	0.932	998	0.948	1014	0.792

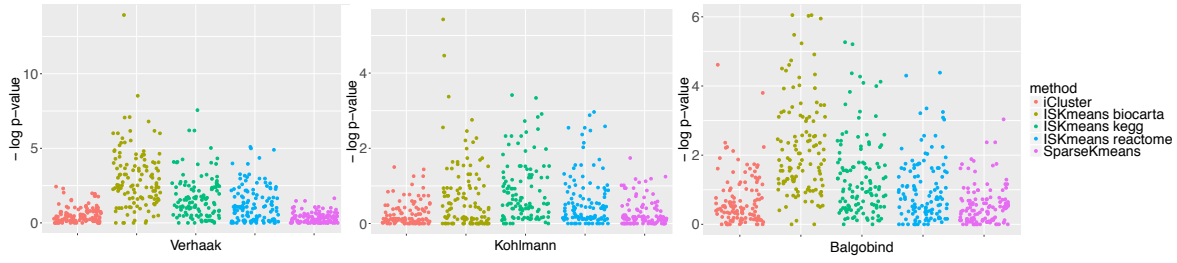


Figure 12: Pathway enrichment analysis result for Leukemia BioCarta

4.0 DISCUSSION AND FUTURE WORK

4.1 DISCUSSION

Clustering analysis is essential to disease subtype discovery, which is a first step toward personalized medicine. With the accumulation of large amount of genomic data, it is urgent and practical to combine multiple cohorts/omics types to increase statistical power and reproducibility. It is also challenging because there are many statistical difficulties. In this these, we proposed both meta-analysis sparse Kmeans and integrative sparse Kmeans to tackle this problem. Simulation and real data application both showed promising result. These works are nice contribution to both statistical and biological community. They are not only innovative statistical methodologies, but also practice tools for real data applications.

4.2 INTEGRATIVE META SPARSE KMEANS

In this section we want to propose a unified framework extending both meta Sparse Kmeans and integrative sparse Kmeans. This part is unfinished and left as a future work. This method will integrate multiple types of omics data and combine multiple cohorts. Several issues have to be considered simultaneously. 1, combining multiple cohorts. 2, integrating different omics data types. 3, allowing missing datatype in several cohorts. 4, allowing overlapping group information. 5, achieving sparse clustering. We propose tentative objective function to tackle this problem. Evaluation will be performed in the future.

$$\min -\sum_j z_j \left(\frac{1}{S} \sum_{s=1}^{S} \frac{BCSS_j^{(s)}(C^{(s)}(K))}{TSS_j^{(s)}} + \lambda \times f_j^{match}(M) \right) + \gamma_1 \|\mathbf{z}\|_1 + \gamma_2 \Omega(\mathbf{z})$$

s.t. $\|\mathbf{z}\|_2 \leq 1$, $z_j \geq 0$, $\forall j$.

$$\min -\sum z_j R_j + \gamma_1 \|\mathbf{z}\|_1 + \gamma_2 \Omega(\mathbf{z})$$

s.t. $\|\mathbf{z}\|_2 \leq 1$, $z_j \geq 0$, $\forall j$, $R_j = \frac{1}{S} \sum_{s=1}^{S} \frac{BCSS_j^{(s)}(C^{(s)}(K))}{TSS_j^{(s)}} + \lambda \times f_j^{match}(M)$

APPENDIX A

APPENDIX FOR META SPARSE KMEANS

A.1 ALGORITHMS FOR SIMULATED ANNEALING

When the number of studies is large, the space to search for matching clusters across studies is not viable with exhaustive search. To maximize the matching objective Equation 2.3.4b, denoted as $\pi(M)$, we applied simulated annealing, a stochastic optimization algorithm for non-convex function (Kirkpatrick et al., 1983). Our configuration space is defined as a matching matrix, where the columns correspond to the studies, and the rows correspond to the matched clusters. For example, if the first row of 3 studies is (1,2,1), that means the first cluster of 1^{st} study, second cluster of 2^{nd} study and first cluster of 3^{rd} study are matched as one disease subtype. Also denote $0 \leq \beta \leq 1$ as the temperature cooling coefficient and α_i as the acceptance rate at temperature T_i, which is defined as:

$$\alpha_i = \frac{\text{total number of acceptance}}{\text{total number of simulated annealing steps}}$$

at each temperature. β will decide how slow the temperature T decreases and balance between the accuracy of the result and computation speed. η is the acceptance threshold which decides when the algorithm stops.

The simulated annealing is conducted in the following steps:

1. Start with a high temperature $T_i(i = 1)$.

2. At temperature T_i (one simulated annealing step), we perturb the configuration space by randomly choosing two elements in the cluster matching enumeration M from two studies and switch their positions, then calculate the new target value $\pi(M^{new})$. Accept the new configuration with probability:

$$P_{acc} = \min\left(1, \exp\big(-\frac{\pi(M^{new}) - \pi(M^{old})}{T}\big)\right)$$

   This procedure will be repeated N times (MC steps).
3. Set $T_{i+1} = T_i \times \beta$
4. Repeat Step 2-3 until $\alpha_i < \eta$,

In our analysis, we used the MC steps $N = 300$ at each temperature T_i. The temperature decreasing rate β is 0.9. The simulated annealing stops when the acceptance ratio drops below $\eta = 0.1$ or the total simulated annealing steps exceed $10,000$. The initial temperature T_1 is set as the objective function value of the initial configuration. In case the initial temperature is too high which result in a high acceptance ratio, we multiply the temperature with $\beta = 0.7$ whenever the acceptance rate $\alpha_i > 0.5$. This will accelerate the convergent rate at initial steps when the acceptance rate is high.

A.2 COMPARING METASPARSEKMEANS AND PAM50 CLUSTERS ON METABRIC

PAM50 is currently the most popular transcriptomic subtype definition of breast cancer. The model consists of 50 intrinsic genes to predict the five subtypes of breast cancer. Among these 50 genes, 42 appeared in the METABRIC dataset and among these 42 genes, 22 overlapped with 194 genes selected by our MetaSparse$K$means result (Fisher's exact tests p-value for overlap enrichment $< 2.2 \times 10^{-16}$). supplementary Table 12 shows a full comparison of the two clustering results by PAM50 and MetaSparse$K$means. There are significant similarity but also discrepancy between the two. Since no underlying truth is known in such a real application, it is difficult to judge which one is better (although MetaSparse$K$means generated smaller p-value of survival difference of the subtypes). Conceptually, PAM50 is a

supervised machine learning result that utilizes class labels determines by many past studies with prior biological knowledge. On the other hand, MetaSparse$K$means is a pure in silico clustering approach.

Table 12: Comparison of MetaSparse$K$means clustering and PAM50 clustering results on METABRIC dataset.

	1	2	3	4	5
Basal	8	122	8	10	180
Her2	9	95	67	60	7
LumA	354	1	34	330	0
LumB	16	3	261	205	5
Normal	122	11	10	57	0

Columns: 5 clusters defined by MetaSparse$K$means. Rows: 5 clusters defined by PAM50.

APPENDIX B

APPENDIX FOR INTEGRATIVE SPARSE KMEANS

B.1 PROOF FOR THEOREM OF IS-KMEANS

Proof of Theorem 3.3.1. Given equal separation ability for each feature $R_1 = \ldots = R_j = \ldots = R_J = R$ and the proposed design of overlapping group lasso penalty, Equation 3.3.1 becomes

$$\min_{C,\mathbf{z}} -\sum_{j=1}^{J} z_j R + \gamma\alpha\|\mathbf{z}\|_1 + \gamma(1-\alpha)\sum_{1\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)}\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)\times z_j^2}\right),$$
$$\text{subject to } \|\mathbf{z}\|_2 \le 1, z_j \ge 0, \forall j.$$

First we can take away the constraint $z_j \ge 0, \forall j$. It is easy to see that if any $z_j < 0$, we can always use $-z_j$ to replace the solution and the objective function will decrease. We can write down the Lagrange function of Equation 3.3.1 after dropping the constraint $z_j \ge 0, \forall j$:

$$L(\mathbf{z},\lambda) = -\sum_{j=1}^{J} z_j R + \gamma\alpha\|\mathbf{z}\|_1 + \gamma(1-\alpha)\sum_{1\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)}\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)\times z_j^2}\right) + \lambda(\|\mathbf{z}\|_2^2 - 1)$$

Partial derivative of the Lagrange is:

$$\frac{\partial L(\mathbf{z})}{\partial z_j} = -R + \gamma\alpha\frac{\partial|z_j|}{\partial z_j} + \gamma(1-\alpha)\sum_{1\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j'\in\mathcal{J}_g} 1/h(j')}\frac{\mathbb{I}\{j\in\mathcal{J}_g\}\times 1/h(j)\times z_j}{\sqrt{\sum_{j'\in\mathcal{J}_g} 1/h(j')\times z_{j'}^2}}\right) + 2\lambda z_j$$

It is easy to verify that $z_1 = z_2 = \ldots = z_J = 1/\sqrt{J}$, $\lambda = \frac{\sqrt{J}(R-\gamma)}{2}$ will make $\frac{\partial L(\mathbf{z})}{\partial z_j} = 0, \forall j$. Since the object function is a convex function, according to sufficiency of KKT condition, the

proposed penalty design will lead to the solution of "Unbiased Feature Selection" principle.

□

Proof of Theorem 3.3.2. For intrinsic gene set $\mathcal{I}$, we have $R_j = R > 0$ for $j \in \mathcal{I}$. For non-intrinsic gene set $\bar{\mathcal{I}}$, we have $R_j = 0$ for $j \in \bar{\mathcal{I}}$. Given the proposed design of overlapping group lasso penalty, Equation 3.3.1 becomes

$$\min_{C,\mathbf{z}} -\sum_{j=1}^{J} z_j R\mathbb{I}(j \in \mathcal{I}) + \gamma\alpha\|\mathbf{z}\|_1 + \gamma(1-\alpha)\sum_{1\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j\in(\mathcal{J}_g\cap\mathcal{I})} 1/h(j)}\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)\times z_j^2}\right),$$
subject to $\|\mathbf{z}\|_2 \le 1, z_j \ge 0, \forall j$.

First we can similarly take away the constraint $z_j \ge 0, \forall j$. We can write down the Lagrange function of Equation 3.3.1 after dropping the constraint $z_j \ge 0, \forall j$:

$$\begin{aligned} L(\mathbf{z},\lambda) = &-\sum_{j=1}^{J} z_j R\mathbb{I}(j \in \mathcal{I}) + \gamma\alpha\|\mathbf{z}\|_1 + \\ &\gamma(1-\alpha)\sum_{1\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j\in(\mathcal{J}_g\cap\mathcal{I})} 1/h(j)}\sqrt{\sum_{j\in\mathcal{J}_g} 1/h(j)\times z_j^2}\right) + \lambda(\|\mathbf{z}\|_2^2 - 1) \end{aligned}$$

Partial derivative of the Lagrange is:

$$\begin{aligned} \frac{\partial L(\mathbf{z})}{\partial z_j} = &- R\mathbb{I}(j \in \mathcal{I}) + \gamma\alpha\frac{\partial|z_j|}{\partial z_j} + \\ &\gamma(1-\alpha)\sum_{1\le g\le \mathcal{G}_0}\left(\sqrt{\sum_{j'\in(\mathcal{J}_g\cap\mathcal{I})} 1/h(j')}\frac{\mathbb{I}\{j\in\mathcal{J}_g\}\times 1/h(j)\times z_j}{\sqrt{\sum_{j'\in\mathcal{J}_g} 1/h(j')\times z_{j'}^2}}\right) + 2\lambda z_j \end{aligned}$$

It is easy to verify that if for $j \in \mathcal{I}$, $z_j = 1/\sqrt{J}$, $j \in \bar{\mathcal{I}}$, $z_j = 0$ and $\lambda = \frac{\sqrt{J}(R-\gamma)}{2}$ is a zero solution to the partial derivative of the Lagrange function. Note here we set the subgradient $\frac{\partial|z_j|}{\partial z_j} = 0$ at $z_j = 0$. Since the object function is a convex function, according to sufficiency of KKT condition, the proposed penalty design leads to "Unbiased Feature Selection" principle.

□

B.2 OPTIMIZATION BY KKT CONDITION

There are two optimization problems.

$$\begin{cases} \mathbf{x}_g^+ = \arg\min_{\mathbf{x}_g} \|\mathbf{x}_g\|_2 + \mathbf{y}_g^\top \mathbf{x}_g + \frac{\rho}{2}\|\mathbf{x}_g - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2 \\ \mathbf{z}^+ = \arg\min_{\mathbf{z}} -\sum z_j R_j - \sum_{1\le g\le \mathcal{G}} \mathbf{y}_g^\top (\boldsymbol{\beta}_g \circ \mathbf{z}) + \frac{\rho}{2}\|\mathbf{x}_g^+ - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2 \\ \quad \text{subject to } \|\mathbf{z}\|_2 \le 1, z_j \ge 0. \end{cases}$$

It is a convex optimization problem for $\mathbf{x}_g^+$ with no constraint. The stationarity condition states that the sub-gradient of the objective function will be 0 at the optimum solution. Therefore we have:

$$S(\mathbf{x}_g^+) + \mathbf{y}_g + \rho(\mathbf{x}_g^+ - \boldsymbol{\beta}_g \circ \mathbf{z}) = 0,$$

where $S(\mathbf{v})$ is the sub-gradient of $\|\mathbf{v}\|_2$ and

$$S(\mathbf{v}) \in \begin{cases} \frac{\mathbf{v}}{\|\mathbf{v}\|_2}, \text{if } \|\mathbf{v}\|_2 \ge 1 \\ \mathbf{0}, \text{otherwise} \end{cases}$$

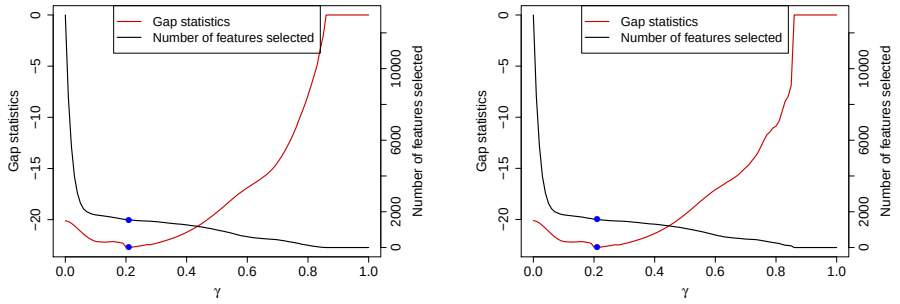
If we define $\mathbf{a}_g = \boldsymbol{\beta}_g \circ \mathbf{z} - \frac{\mathbf{y}_g}{\rho}$, it can be derived that $\mathbf{x}_g^+ = (1 - \frac{1}{\rho\|\mathbf{a}_g\|_2})_+ \mathbf{a}_g$, where $(\cdot)_+ = \max(0, \cdot)$.

The optimization problem for $\mathbf{z}^+$ is a convex optimization problem with two constraints. We first write down the Lagrange function and convert the constrained optimization problem into an un-constrained optimization problem:

$$\arg\min_{\mathbf{z}} -\sum_j z_j R_j - \sum_{1\le g\le \mathcal{G}} \mathbf{y}_g^\top (\boldsymbol{\beta}_g \circ \mathbf{z}) + \frac{\rho}{2}\|\mathbf{x}_g^+ - \boldsymbol{\beta}_g \circ \mathbf{z}\|_2^2 + u(\|\mathbf{z}\|_2 - 1) - \sum_j v_j z_j$$

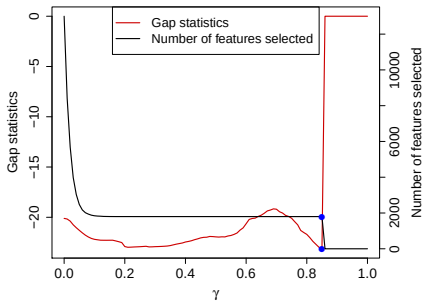
such that $u \in \mathbb{R}$, $u \ge 0$, $v_j \in \mathbb{R}$ and $v_j \ge 0\ \forall j$. Taking gradient of the Lagrange function with respect to $\mathbf{z}$ and use the constraints, we can derive the solution to this problem. Define $b_j = \sum_{1\le g\le \mathcal{G}} \rho \boldsymbol{\beta}_{gj}^2$ and $c_j = \sum_{1\le g\le \mathcal{G}} \left(\rho \mathbf{x}_{gj}^+ + \mathbf{y}_{gj}\right) \circ \mathbf{m}_{gj}$, where $\boldsymbol{\beta}_g = (\boldsymbol{\beta}_{g1}, \boldsymbol{\beta}_{g2}, \ldots, \boldsymbol{\beta}_{gJ})^\top$, $\mathbf{x}_g = (\mathbf{x}_{g1}, \mathbf{x}_{g2}, \ldots, \mathbf{x}_{gJ})^\top$, $\mathbf{y}_g = (\mathbf{y}_{g1}, \mathbf{y}_{g2}, \ldots, \mathbf{y}_{gJ})^\top$, and $\mathbf{m}_g = (\mathbf{m}_{g1}, \mathbf{m}_{g2}, \ldots, \mathbf{m}_{gJ})^\top$. The solution is given as following: we define $f_j(u) = (\frac{R_j + c_j}{b_j + 2u})_+$. If $\sum_j f_j(u)^2 < 1$, $z_j^+ = f_j(0)$. Otherwise $z_j^+ = f_j(u)$ and u is selected s.t. $\|\mathbf{z}^+\|_2 = 1$.

B.3 SUPPLEMENTARY MATERIALS FOR IS-KMEANS



(a) $\alpha = 1$

(b) $\alpha = 0.95$



(c) $\alpha = 0.05$

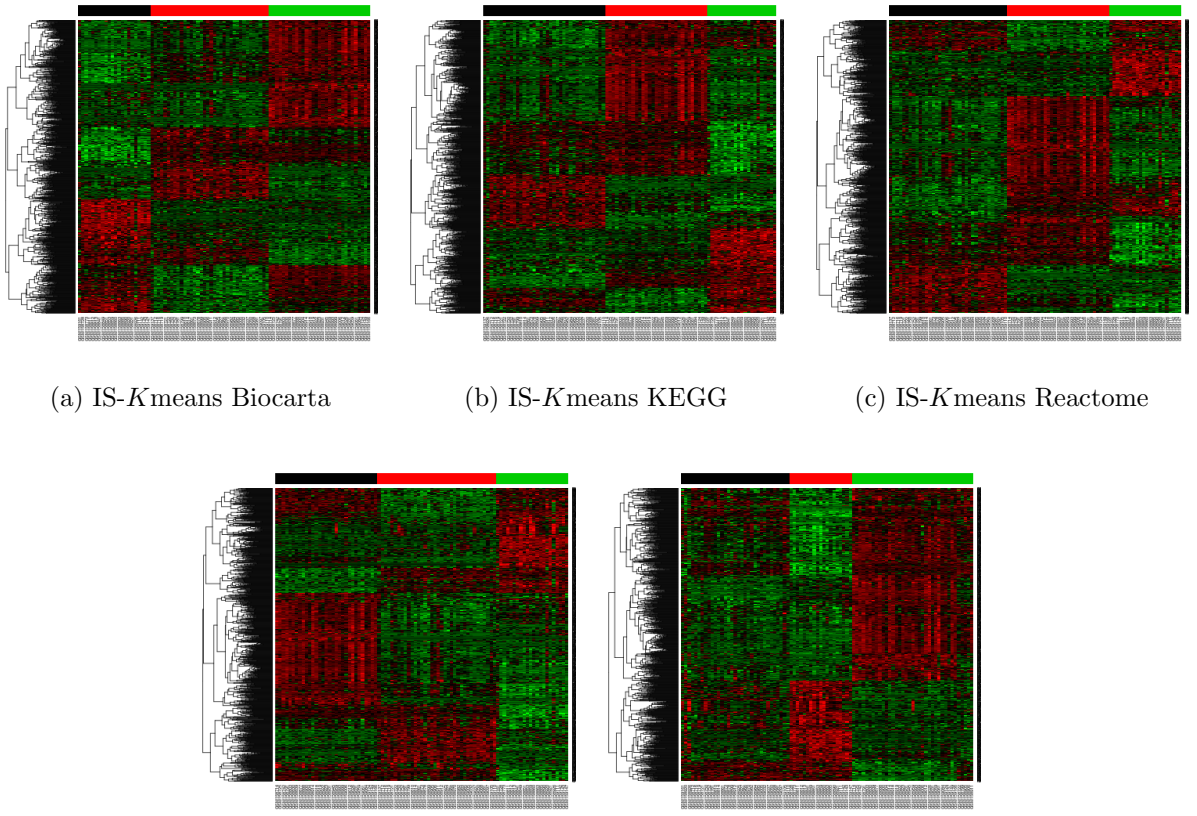
Figure 13: Selection of tuning parameter γ.

This figure is from the simulated dataset in Section 4.1 with relative effect size $f = 0.6$ and $\theta = 1$. Complementary to Figure 2, simulation setting with $\alpha = 1, 0.95, 0.05$ are evaluated. X-axis is tuning parameter γ, red curve and left y-axis denote the corresponding gap statistics, black curve and right y-axis denote the corresponding number of selected features. The blue dots represent where the gap statistics is minimized, and the corresponding number of selected features are 1548, 1566, 1800.

Table 13: Comparison table of simulation with relative effect size $f = 0.8$.

θ	method	α	ARI	Jaccard index	AUC	# features	time [mins]
1	IS-Kmeans	1	1.000 (0.000)	0.812 (0.042)	0.995 (0.004)	1461	0.38
		0.95	1.000 (0.000)	0.826 (0.041)	0.996 (0.004)	1486	0.49
		0.5	1.000 (0.000)	0.904 (0.032)	0.999 (0.001)	1627	0.52
		0.05	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	1800	0.57
0.2	IS-Kmeans	1	1.000 (0.000)	0.812 (0.042)	0.995 (0.004)	1461	0.38
		0.95	1.000 (0.000)	0.771 (0.045)	0.995 (0.004)	1388	0.55
		0.5	1.000 (0.000)	0.835 (0.038)	0.994 (0.004)	1512	0.49
		0.05	1.000 (0.000)	0.495 (0.021)	0.938 (0.004)	2815	1
	iCluster		0.722 (0.325)	0.672 (0.198)		1906	26
	sparse Kmeans 1		0.931 (0.209)	0.159 (0.073)		5777	0.13
	sparse Kmeans 2		0.898 (0.253)	0.070 (0.035)		506	0.11

We simulated $B = 100$ times and calculated mean and standard deviation of each quantity. θ denotes the probability grouping information is correct for each feature inside groups. α is the tuning parameter balancing the emphasis between individual penalty and group penalty. For each method, we allow its own tuning parameter selection method to optimize its performance.

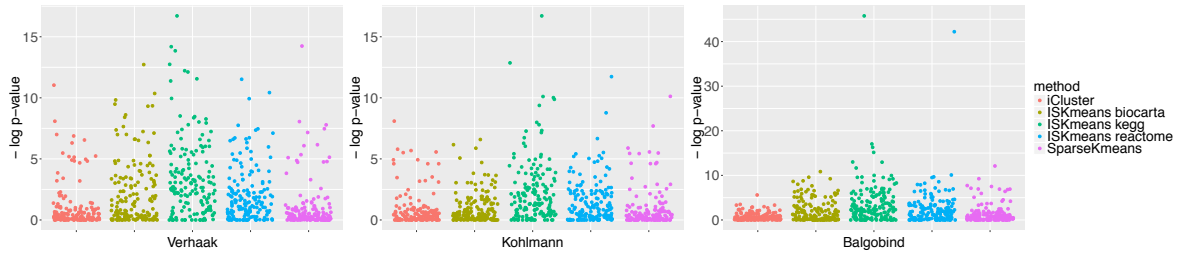


(a) IS-Kmeans Biocarta (b) IS-Kmeans KEGG (c) IS-Kmeans Reactome

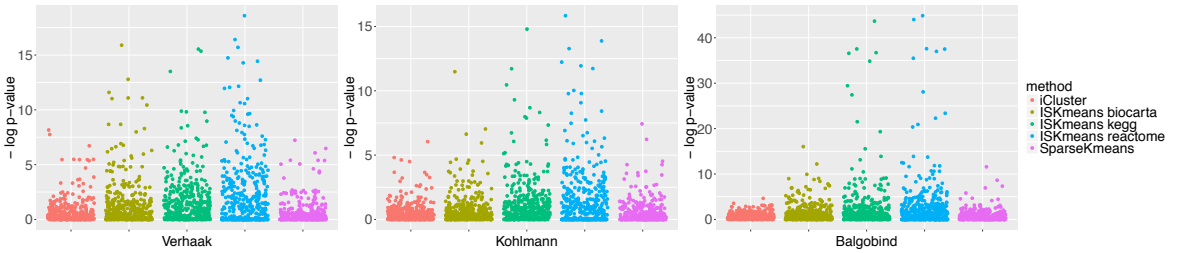
(d) Sparse K-means (e) iCluster

Figure 14: Heatmap of Verhaak by IS-Kmeans.

Heatmap of Verhaak by IS-Kmeans (using BioCarta, KEGG and Reactome pathway databases as prior knowledge), Sparse K-means and iCluster. Number of selected features are: 1,009 for Figure S14(a), 1,002 for Figure S14(b), 993 for Figure S14(c), 982 for Figure S14(d), 992 for Figure S14(e).



(a) test KEGG pathway



(b) test Reactome pathway

Figure 15: Pathway enrichment analysis result for Leukemia using KEGG and Reactome as testing database.

Table 14: Comparison of IS-Kmeans and PAM50 clustering results on TCGA multi-omics dataset.

	1	2	3	4	5
Basal	0	4	8	135	13
Her2	4	54	4	0	42
LumA	59	25	153	0	0
LumB	86	89	13	0	1
Normal	1	6	68	0	3

Columns: five clusters defined by IS-Kmeans. Rows: five clusters defined by PAM50.

Table 15: Comparison of IS-Kmeans and PAM50 clustering results on METABRIC multi-omics dataset.

	1	2	3	4	5
Basal	0	69	24	0	235
Her2	8	133	78	19	0
LumA	343	2	146	228	0
LumB	112	5	147	226	0
Normal	45	29	106	9	11

Columns: five clusters defined by IS-Kmeans. Rows: five clusters defined by PAM50.

Table 16: Comparison table of perturbation analysis for IS-Kmeans with $f = 0.8$.

θ	method	α	ARI	Jaccard index	AUC	# features	time [mins]
1	IS-Kmeans	1	0.980 (0.078)	0.817 (0.066)	0.973 (3e-02)	1701	0.53
		0.95	0.980 (0.078)	0.825 (0.065)	0.975 (3e-02)	1715	0.64
		0.5	0.980 (0.078)	0.905 (0.063)	0.990 (1e-02)	1652	0.59
		0.05	0.984 (0.068)	0.993 (0.005)	1.000 (1e-04)	1812	0.67
0.2	IS-Kmeans	1	0.980 (0.078)	0.817 (0.066)	0.973 (3e-02)	1701	0.52
		0.95	0.980 (0.078)	0.817 (0.065)	0.973 (3e-02)	1708	0.7
		0.5	0.980 (0.078)	0.777 (0.058)	0.973 (3e-02)	1863	0.68
		0.05	0.980 (0.078)	0.496 (0.032)	0.911 (2e-02)	3002	1.3
	iCluster		0.717 (0.291)	0.679 (0.186)		1657	27
	sparse Kmeans 1		0.824 (0.285)	0.096 (0.026)		2065	0.13
	sparse Kmeans 2		0.826 (0.291)	0.119 (0.043)		6139	0.16

In the simulation setting Step a3, the template gene expression is simulated from a t-distribution with degree of freedom 3, location parameter 9 and scale parameter 2. In Step a4, we set relative effect size $f = 0.8$. In Step a5, X'_{skmi} is simulated from a t-distribution with degree of freedom 3, location parameter μ'_{skm} and scale parameter σ_1^2. We simulated $B = 100$ times and calculated mean and standard deviation of each quantity. θ denotes the probability grouping information is correct for each feature inside groups. α is the tuning parameter balancing the emphasis between individual penalty and group penalty. For each method, we allow its own tuning parameter selection method to optimize its performance.

Table 17: Comparison table of perturbation analysis for IS-Kmeans with $f = 0.6$.

θ	method	α	ARI	Jaccard index	AUC	# features	time [mins]
1	IS-Kmeans	1	0.798 (0.368)	0.592 (0.260)	0.811 (0.204)	1455	0.55
		0.95	0.798 (0.368)	0.601 (0.262)	0.817 (0.200)	1474	0.68
		0.5	0.793 (0.368)	0.710 (0.292)	0.899 (0.126)	1426	0.68
		0.05	0.801 (0.358)	0.914 (0.150)	0.992 (0.013)	1760	0.91
0.2	IS-Kmeans	1	0.809 (0.352)	0.600 (0.249)	0.829 (0.190)	1468	0.55
		0.95	0.809 (0.352)	0.600 (0.248)	0.829 (0.189)	1478	0.74
		0.5	0.809 (0.352)	0.573 (0.227)	0.832 (0.184)	1678	0.74
		0.05	0.809 (0.352)	0.397 (0.122)	0.791 (0.147)	2900	1.5
	iCluster		0.300 (0.313)	0.261 (0.196)		790	27
	sparse Kmeans 1		0.334 (0.377)	0.243 (0.202)		4755	0.17
	sparse Kmeans 2		0.187 (0.304)	0.022 (0.036)		725	0.13

In the simulation setting Step a3, the template gene expression is simulated from a t-distribution with degree of freedom 3, location parameter 9 and scale parameter 2. In Step a4, we set relative effect size $f = 0.6$. In Step a5, X'_{skmi} is simulated from a t-distribution with degree of freedom 3, location parameter μ'_{skm} and scale parameter σ_1^2. We simulated $B = 100$ times and calculated mean and standard deviation of each quantity. θ denotes the probability grouping information is correct for each feature inside groups. α is the tuning parameter balancing the emphasis between individual penalty and group penalty. For each method, we allow its own tuning parameter selection method to optimize its performance.

BIBLIOGRAPHY

Abramson, V. G., Lehmann, B. D., Ballinger, T. J., and Pietenpol, J. A. (2015). Subtyping of triple-negative breast cancer: Implications for therapy. *Cancer*, 121(1):8–16.

Akalin, A., Kormaksson, M., Li, S., Garrett-Bakelman, F. E., Figueroa, M. E., Melnick, A., Mason, C. E., et al. (2012). methylkit: a comprehensive r package for the analysis of genome-wide dna methylation profiles. *Genome Biol*, 13(10):R87.

Balgobind, B. V., Van den Heuvel-Eibrink, M. M., De Menezes, R. X., Reinhardt, D., Hollink, I. H., Arentsen-Peters, S. T., van Wering, E. R., Kaspers, G. J., Cloos, J., de Bont, E. S., et al. (2011). Evaluation of gene expression signatures predictive of cytogenetic and molecular subtypes of pediatric acute myeloid leukemia. *haematologica*, 96(2):221–230.

Bass, A., Thorsson, V., Shmulevich, I., Reynolds, S., Miller, M., Bernard, B., Hinoue, T., Laird, P., Curtis, C., Shen, H., et al. (2014). Comprehensive molecular characterization of gastric adenocarcinoma. *Nature*, 513(7517):202–209.

Begum, F., Ghosh, D., Tseng, G. C., and Feingold, E. (2012). Comprehensive literature review and statistical considerations for gwas meta-analysis. *Nucleic acids research*, page gkr1255.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 289–300.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122.

Boyd, S. P. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

Chang, L.-C., Lin, H.-M., Sibille, E., and Tseng, G. C. (2013). Meta-analysis methods for combining multiple expression profiles: comparisons, statistical characterization and an application guideline. *BMC bioinformatics*, 14(1):368.

Cheng, C., Shen, K., Song, C., Luo, J., and Tseng, G. C. (2009). Ratio adjustment and calibration scheme for gene-wise normalization to enhance microarray inter-study prediction. *Bioinformatics*, 25(13):1655–1661.

Cheng, Y. (1995). Mean shift, mode seeking, and clustering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(8):790–799.

Curtis, C., Shah, S. P., Chin, S.-F., Turashvili, G., Rueda, O. M., Dunning, M. J., Speed, D., Lynch, A. G., Samarajiwa, S., Yuan, Y., et al. (2012). The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups. *Nature*, 486(7403):346–352.

Desmedt, C., Piette, F., Loi, S., Wang, Y., Lallemand, F., Haibe-Kains, B., Viale, G., Delorenzi, M., Zhang, Y., d'Assignies, M. S., et al. (2007). Strong time dependence of the 76-gene prognostic signature for node-negative breast cancer patients in the transbig multicenter independent validation series. *Clinical cancer research*, 13(11):3207–3214.

Domany, E. (2014). Using high-throughput transcriptomic data for prognosis: A critical overview and perspectives. *Cancer Research*, 74(17):4612–4621.

Dudoit, S. and Fridlyand, J. (2002). A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome biology*, 3(7):research0036.

Eisen, M. B., Spellman, P. T., Brown, P. O., and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868.

Esteller, M. (2007). Cancer epigenomics: Dna methylomes and histone-modification maps. *Nature Reviews Genetics*, 8(4):286–298.

Fan, X. and Kurgan, L. (2015). Comprehensive overview and assessment of computational prediction of microrna targets in animals. *Briefings in bioinformatics*, 16(5):780–794.

Fisher, R. A. (1925). *Statistical methods for research workers*. Genesis Publishing Pvt Ltd.

Fraga, M. F., Ballestar, E., Paz, M. F., Ropero, S., Setien, F., Ballestar, M. L., Heine-Suñer, D., Cigudosa, J. C., Urioste, M., Benitez, J., et al. (2005). Epigenetic differences arise during the lifetime of monozygotic twins. *Proceedings of the National Academy of Sciences of the United States of America*, 102(30):10604–10609.

Gentleman, R., Carey, V., Huber, W., Irizarry, R., and Dudoit, S. (2005). *Bioinformatics and computational biology solutions using R and Bioconductor*, volume 746718470. Springer New York.

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., et al. (1999). Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *science*, 286(5439):531–537.

Hansen, K. D., Langmead, B., Irizarry, R. A., et al. (2012). Bsmooth: from whole genome bisulfite sequencing reads to differentially methylated regions. *Genome Biol*, 13(10):R83.

Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

He, B., Yang, H., and Wang, S. (2000). Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities. *Journal of Optimization Theory and applications*, 106(2):337–356.

Hochberg, Y. and Tamhane, A. C. (2009). Multiple comparison procedures.

Huang, J., Horowitz, J. L., Wei, F., et al. (2010). Variable selection in nonparametric additive models. *The Annals of Statistics*, 38(4):2282–2313.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of classification*, 2(1):193–218.

Huo, Z., Ding, Y., Liu, S., Oesterreich, S., and Tseng, G. (2016). Meta-analytic framework for sparse k-means to identify disease subtypes in multiple transcriptomic studies. *Journal of the American Statistical Association*, 111(513):27–42.

Huo, Z. and Tseng, G. C. (2017). Integrative sparse k-means with overlapping group lasso in genomic applications for disease subtype discovery. *The Annals of Applied Statistics*, In press.

Ioannidis, J. P., Allison, D. B., Ball, C. A., Coulibaly, I., Cui, X., Culhane, A. C., Falchi, M., Furlanello, C., Game, L., Jurman, G., et al. (2008). Repeatability of published microarray gene expression analyses. *Nature genetics*, 41(2):149–155.

Ivshina, A. V., George, J., Senko, O., Mow, B., Putti, T. C., Smeds, J., Lindahl, T., Pawitan, Y., Hall, P., Nordgren, H., et al. (2006). Genetic reclassification of histologic grade delineates new clinical subtypes of breast cancer. *Cancer research*, 66(21):10292–10301.

Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin del la Société Vaudoise des Sciences Naturelles*, 37:547–579.

Jacob, L., Obozinski, G., and Vert, J. P. (2009). Group lasso with overlap and graph lasso. In *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*, pages 433–440. ACM.

Kaufman, L. and Rousseeuw, P. (1987). *Clustering by means of medoids*. North-Holland.

Kaufman, L. and Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*, volume 344. Wiley. com.

Kim, E.-Y., Kim, S.-Y., Ashlock, D., and Nam, D. (2009). Multi-k: accurate classification of microarray subtypes using ensemble k-means clustering. *BMC bioinformatics*, 10(1):260.

Kirkpatrick, S., Jr., D. G., and Vecchi, M. P. (1983). Optimization by simmulated annealing. *science*, 220(4598):671–680.

Koboldt, D. C., Fulton, R. S., McLellan, M. D., Schmidt, H., Kalicki-Veizer, J., McMichael, J. F., Fulton, L. L., Dooling, D. J., Ding, L., Mardis, E. R., et al. (2012). Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70.

Kohlmann, A., Kipps, T. J., Rassenti, L. Z., Downing, J. R., Shurtleff, S. A., Mills, K. I., Gilkes, A. F., Hofmann, W.-K., Basso, G., Dell?Orto, M. C., et al. (2008). An international standardization programme towards the application of gene expression profiling in routine leukaemia diagnostics: the microarray innovations in leukemia study prephase. *British journal of haematology*, 142(5):802–807.

Kohonen, T. (1998). The self-organizing map. *Neurocomputing*, 21(1):1–6.

Kulis, M. and Esteller, M. (2010). Dna methylation and cancer. *Adv Genet*, 70:27–56.

Lehmann, B. D., Bauer, J. A., Chen, X., Sanders, M. E., Chakravarthy, A. B., Shyr, Y., and Pietenpol, J. A. (2011). Identification of human triple-negative breast cancer subtypes and preclinical models for selection of targeted therapies. *The Journal of clinical investigation*, 121(7):2750.

Li, J., Tseng, G. C., et al. (2011). An adaptively weighted statistic for detecting differential gene expression when combining multiple transcriptomic studies. *The Annals of Applied Statistics*, 5(2A):994–1019.

Lock, E. F. and Dunson, D. B. (2013). Bayesian consensus clustering. *Bioinformatics*, 29(20):2610–2616.

Loi, S., Haibe-Kains, B., Desmedt, C., Lallemand, F., Tutt, A. M., Gillet, C., Ellis, P., Harris, A., Bergh, J., Foekens, J. A., et al. (2007). Definition of clinically distinct molecular subtypes in estrogen receptor–positive breast carcinomas through genomic grade. *Journal of clinical oncology*, 25(10):1239–1246.

Lu, S., Li, J., Song, C., Shen, K., and Tseng, G. C. (2010). Biomarker detection in the integration of multiple multi-class genomic studies. *Bioinformatics*, 26(3):333–340.

Mackay, A., Weigelt, B., Grigoriadis, A., Kreike, B., Natrajan, R., A?Hern, R., Tan, D. S., Dowsett, M., Ashworth, A., and Reis-Filho, J. S. (2011). Microarray-based class discovery for molecular classification of breast cancer: analysis of interobserver agreement. *Journal of the National Cancer Institute*, 103(8):662–673.

MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

Maitra, R. and Ramler, I. P. (2009). Clustering in the presence of scatter. *Biometrics*, 65(2):341–352.

McLachlan, G. J., Bean, R., and Peel, D. (2002). A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, 18(3):413–422.

Milligan, G. W. and Cooper, M. C. (1985). An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179.

Network, T. C. G. A. (2012). Comprehensive molecular portraits of human breast tumours.

Park, Y., Figueroa, M. E., Rozek, L. S., and Sartor, M. A. (2014). Methylsig: a whole genome dna methylation analysis pipeline. *Bioinformatics*, page btu339.

Parker, J. S., Mullins, M., Cheang, M. C., Leung, S., Voduc, D., Vickery, T., Davies, S., Fauron, C., He, X., Hu, Z., et al. (2009). Supervised risk predictor of breast cancer based on intrinsic subtypes. *Journal of clinical oncology*, 27(8):1160–1167.

Parsons, D. W., Jones, S., Zhang, X., Lin, J. C.-H., Leary, R. J., Angenendt, P., Mankoo, P., Carter, H., Siu, I.-M., Gallia, G. L., et al. (2008). An integrated genomic analysis of human glioblastoma multiforme. *Science*, 321(5897):1807–1812.

Perou, C. M., Sørlie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., Rees, C. A., Pollack, J. R., Ross, D. T., Johnsen, H., Akslen, L. A., et al. (2000). Molecular portraits of human breast tumours. *Nature*, 406(6797):747–752.

Qin, Z. S. (2006). Clustering microarray gene expression data using weighted chinese restaurant process. *Bioinformatics*, 22(16):1988–1997.

Ramasamy, A., Mondry, A., Holmes, C. C., and Altman, D. G. (2008). Key issues in conducting a meta-analysis of gene expression microarray datasets. *PLoS Medicine*, 5(9):1320–1333.

Richardson, S., Tseng, G. C., and Sun, W. (2016). Statistical methods in integrative genomics. *Annual Review of Statistics and Its Application*, 3:181–209.

Robertson, K. D. (2005). Dna methylation and human disease. *Nature Reviews Genetics*, 6(8):597–610.

Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010). edger: a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1):139–140.

Rosenwald, A., Wright, G., Chan, W. C., Connors, J. M., Campo, E., Fisher, R. I., Gascoyne, R. D., Muller-Hermelink, H. K., Smeland, E. B., Giltnane, J. M., et al. (2002). The use of molecular profiling to predict survival after chemotherapy for diffuse large-b-cell lymphoma. *New England Journal of Medicine*, 346(25):1937–1947.

Sadanandam, A., Lyssiotis, C. A., Homicsko, K., Collisson, E. A., Gibb, W. J., Wullschleger, S., Ostos, L. C. G., Lannon, W. A., Grotzinger, C., Del Rio, M., et al. (2013). A colorectal

cancer classification system that associates cellular phenotype and responses to therapy. *Nature medicine*, 19(5):619–625.

Scharpf, R. B., Tjelmeland, H., Parmigiani, G., and Nobel, A. B. (2009). A bayesian model for cross-study differential gene expression. *Journal of the American Statistical Association*, 104(488).

Shen, K. and Tseng, G. C. (2010). Meta-analysis for pathway enrichment analysis when combining multiple genomic studies. *Bioinformatics*, 26(10):1316–1323.

Shen, R., Olshen, A. B., and Ladanyi, M. (2009). Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis. *Bioinformatics*, 25(22):2906–2912.

Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2013). A sparse-group lasso. *Journal of Computational and Graphical Statistics*, 22(2):231–245.

Simon, R. (2005). Development and validation of therapeutically relevant multi-gene biomarker classifiers. *Journal of the National Cancer Institute*, 97(12):866–867.

Simon, R., Radmacher, M. D., Dobbin, K., and McShane, L. M. (2003). Pitfalls in the use of dna microarray data for diagnostic and prognostic classification. *Journal of the National Cancer Institute*, 95(1):14–18.

Smyth, G. K. (2005). Limma: linear models for microarray data. In *Bioinformatics and computational biology solutions using R and Bioconductor*, pages 397–420. Springer.

Song, C. and Tseng, G. C. (2014). Hypothesis setting and order statistic for robust genomic meta-analysis. *The annals of applied statistics*, 8(2):777.

Sørlie, T., Perou, C. M., Tibshirani, R., Aas, T., Geisler, S., Johnsen, H., Hastie, T., Eisen, M. B., van de Rijn, M., Jeffrey, S. S., et al. (2001). Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proceedings of the National Academy of Sciences*, 98(19):10869–10874.

Sørlie, T., Tibshirani, R., Parker, J., Hastie, T., Marron, J., Nobel, A., Deng, S., Johnsen, H., Pesich, R., Geisler, S., et al. (2003). Repeated observation of breast tumor subtypes in independent gene expression data sets. *Proceedings of the National Academy of Sciences*, 100(14):8418–8423.

Stouffer, S. A. (1949). A study of attitudes. *Scientific American*, 180(5):11.

Sugar, C. A. and James, G. M. (2003). Finding the number of clusters in a dataset. *Journal of the American Statistical Association*, 98(463).

Swift, S., Tucker, A., Vinciotti, V., Martin, N., Orengo, C., Liu, X., and Kellam, P. (2004). Consensus clustering and functional interpretation of gene-expression data. *Genome biology*, 5(11):R94.

Tibshirani, R., Hastie, T., Narasimhan, B., and Chu, G. (2002). Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences*, 99(10):6567–6572.

Tibshirani, R. and Walther, G. (2005). Cluster validation by prediction strength. *Journal of Computational and Graphical Statistics*, 14(3):511–528.

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423.

Tothill, R. W., Tinker, A. V., George, J., Brown, R., Fox, S. B., Lade, S., Johnson, D. S., Trivett, M. K., Etemadmoghadam, D., Locandro, B., et al. (2008). Novel molecular subtypes of serous and endometrioid ovarian cancer linked to clinical outcome. *Clinical Cancer Research*, 14(16):5198–5208.

Tseng, G. C. (2007). Penalized and weighted k-means for clustering with scattered objects and prior information in high-throughput biological data. *Bioinformatics*, 23(17):2247–2255.

Tseng, G. C., Ghosh, D., and Feingold, E. (2012). Comprehensive literature review and statistical considerations for microarray meta-analysis. *Nucleic acids research*, 40(9):3785–3799.

Tseng, G. C. and Wong, W. H. (2005). Tight clustering: A resampling-based approach for identifying stable and tight patterns in data. *Biometrics*, 61(1):10–16.

Tusher, V. G., Tibshirani, R., and Chu, G. (2001). Significance analysis of microarrays applied to the ionizing radiation response. *Proceedings of the National Academy of Sciences*, 98(9):5116–5121.

van't Veer, L. J., Dai, H., Van De Vijver, M. J., He, Y. D., Hart, A. A., Mao, M., Peterse, H. L., van der Kooy, K., Marton, M. J., Witteveen, A. T., et al. (2002). Gene expression profiling predicts clinical outcome of breast cancer. *nature*, 415(6871):530–536.

Verhaak, R. G., Hoadley, K. A., Purdom, E., Wang, V., Qi, Y., Wilkerson, M. D., Miller, C. R., Ding, L., Golub, T., Mesirov, J. P., et al. (2010). Integrated genomic analysis identifies clinically relevant subtypes of glioblastoma characterized by abnormalities in *PDGFRA*, *IDH1*, *EGFR*, and *NF1*. *Cancer cell*, 17(1):98–110.

Verhaak, R. G., Wouters, B. J., Erpelinck, C. A., Abbas, S., Beverloo, H. B., Lugthart, S., Löwenberg, B., Delwel, R., and Valk, P. J. (2009). Prediction of molecular subtypes in acute myeloid leukemia based on gene expression profiling. *haematologica*, 94(1):131–134.

Wang, S. and Liao, L. (2001). Decomposition method with a variable parameter for a class of monotone variational inequality problems. *Journal of optimization theory and applications*, 109(2):415–429.

Wang, W., Baladandayuthapani, V., Morris, J. S., Broom, B. M., Manyam, G., and Do, K.-A. (2013). ibag: integrative bayesian analysis of high-dimensional multiplatform genomics data. *Bioinformatics*, 29(2):149–159.

Wang, X., Kang, D. D., Shen, K., Song, C., Lu, S., Chang, L.-C., Liao, S. G., Huo, Z., Tang, S., Ding, Y., et al. (2012). An r package suite for microarray meta-analysis in quality control, differentially expressed gene analysis and pathway enrichment detection. *Bioinformatics*, 28(19):2534–2536.

Wang, Y., Klijn, J. G., Zhang, Y., Sieuwerts, A. M., Look, M. P., Yang, F., Talantov, D., Timmermans, M., Meijer-van Gelder, M. E., Yu, J., et al. (2005). Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. *The Lancet*, 365(9460):671–679.

Ward Jr, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, 58(301):236–244.

Wirapati, P., Sotiriou, C., Kunkel, S., Farmer, P., Pradervand, S., Haibe-Kains, B., Desmedt, C., Ignatiadis, M., Sengstag, T., Schutz, F., et al. (2008). Meta-analysis of gene expression profiles in breast cancer: toward a unified understanding of breast cancer subtyping and prognosis signatures. *Breast Cancer Res*, 10(4):R65.

Witkos, T., Koscianska, E., and Krzyzosiak, W. (2011). Practical aspects of microrna target prediction. *Current molecular medicine*, 11(2):93–109.

Witten, D. M. and Tibshirani, R. (2010). A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490).

Xie, B., Pan, W., and Shen, X. (2008). Penalized model-based clustering with cluster-specific diagonal covariance matrices and grouped variables. *Electronic journal of statistics*, 2:168.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67.

Zhu, L., Ding, Y., Chen, C.-Y., Wang, L., Huo, Z., Kim, S., Sotiriou, C., Oesterreich, S., and Tseng, G. C. (2016). Metadcn: meta-analysis framework for differential co-expression network detection with an application in breast cancer. *Bioinformatics*, page btw788.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476):1418–1429.