Kernel Methods for Data Integration in Microbiome-Omics Studies

Nanxun Ma

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2022

Reading Committee:

Michael C. Wu, Chair

Timothy A. Thornton

Wei Sun

Program Authorized to Offer Degree:
Biostatistics

University of Washington

Abstract

Kernel Methods for Data Integration in Microbiome-Omics Studies

Nanxun Ma

Chair of the Supervisory Committee:

Michael C. Wu

Department of Biostatistics

The human microbiome plays an important role for maintaining the external and internal environment of human health and is associated with many different health conditions and diseases. Meanwhile, other sources of omics data are usually collected simultaneously. However, it is remained a scientific objective to integrate microbiome data with other types of omics data, considering a sequence of challenges including high dimensionality, compositional structures, non-linear effects, and missing data. Facing these challenges, we focus on development of novel statistical approaches for integrative analysis using kernel method, with particular emphasis on integrating microbiome and other types of omics data. Kernel methods are popular for high-dimensional data due to their ability to accommodate nonlinear effects and have been tailored to capture important data-type specific effects. Within this context, we will use kernel approaches to improve understanding of the relationship between data types and to improve the analyses of the individual data types in relation to others.

In the first part of this dissertation, we propose to use a sparse kernel RV (KRV) coefficient to facilitate the identification of genomic features associated with overall microbiome

composition (beta-diversity). The KRV is a generalized measure of multivariate correlation between two data sets, in this case microbiome and genomics, that are embedded as kernel matrices. For microbiome data, we construct fixed, ecologically relevant kernels incorporating important ecological structure. For genomic data, we construct kernels which include feature-specific weights. Sparse estimation of the weights enables selection of genomic markers.

The difficulties of integrating microbiome data with metabolites data remain unanswered for classification problems, when we have both types of data for training with labels, but only metabolites data for prediction. In the second part of the dissertation, we develop classification models using multiple data types that can be applied to future data sets in which only one category of data is collected. Hence, we introduce kernel structures into discriminant analysis, and develop the kernel linear discriminant analysis (KLDA), which can leverage the prediction accuracy utilizing data that are only partially exist. The general KLDA can handle high dimensionality of microbiome data but not the omics data. We then propose a penalized version of KLDA, which can incorporate different types of penalty terms per request of different types of omics data, for example L1 or L2 penalties, to handle the situation that both datasets are high-dimensional with as a classification method.

We evaluate the performance of these methods through extensive simulation studies and apply them to studies investigating the association of an inflammatory bowel disease and women menopause strategies with microbiome data.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank my advisor, Michael Wu, for his mentorship and support over the past several years. His enthusiasm and responsibilities on biostatistics and academic work have been the main inspiration to my development as a biostatistician. This dissertation would not have been possible without him. I am also grateful to my committee members Tim Thornton, Wei Sun, and Amanda Phipps for their valuable insight on this work.

I would like to extend my sincere thanks to Robyn McClelland who I am lucky to have worked with as a research assistant for two years, for broadening my research to various clinical trials data and showing me how to be a good collaborator.

I have benefited greatly from the guidance of many members of the University of Washington Department of Biostatistics. Jon Wakefield, Katie Kerr, Lyndia Brumback, Ali Shojaie, James Hughes, Tim Randolph, Lurdes Inoue, and Scott Emerson have all provided invaluable support and mentorship in research and teaching. I am also deeply grateful to Gitana Garofalo for her advocacy for students and her support in both academic and personal difficulties.

I would like to express my sincere appreciation to the Department of Biostatistics, Collaborative Health Studies Coordinating Center (CHSCC) and Public Health Sciences Division of Fred Hutch Cancer Research Center. I am extremely grateful to have a chance to meet so many brilliant people and experience the warm community culture there.

DEDICATION

To my parents and husband, who have loved, supported, and encouraged me every step of the way

Chapter 1

INTRODUCTION

Microbes are tiny living things that are only visible with the help of a microscope in the traditional definition. Microbes can live in water, soil, in the air, and on or inside of living creatures, like our human body. The human body is home to microbes with estimated counts over 100 trillion. The microbiome is the genetic material of all the microbes, which includes bacteria, fungi, protozoa and viruses. The number of genes in all the microbes in one person's microbiome is estimated to be 200 times the number of genes in the human genome. And the microbiome on or inside a human body may weigh as much as five pounds. The microbiome consists of microbes that are both helpful and potentially harmful. Most of them are symbiotic and some are pathogenic that can promote deceases.

The Human Microbiome Project discovers the composition of the microbiome variation based on diet, health, and environment. Some large scale microbiome profiling studies which have resulted in discoveries relate the microbiome to many different conditions such as cancer, HIV, menopause, blood pressure (Schwabe and Jobin, 2013; Hensley-McBain et al., 2019; Mitchell et al., 2018; Sun et al., 2019). Sine the microbiome is also essential for human development, immunity and nutrition, some autoimmune diseases are also associated with dysfunction in the microbiome or microbiome communities, such as diabetes, rheumatoid arthritis, allergies, and fibromyalgia. Microbiome may also influence human's susceptibility to infectious diseases and contribute to chronic illnesses of the gastrointestinal system such as irritable bowel syndrome. For example, Type I diabetes is an autoimmune disease associated with a less diverse gut microbiome, shown by a sequence of animal studies. Clinicians and

researchers are developing clinical procedures to utilize microbiome to help human maintain in health or even cure deceases. For example, Fecal microbiota transplantation can restore healthy gut microbiota, and potentially work as treatment of colitis, constipation, irritable bowel syndrome and Clostridium difficile infections.

Omics is a rapidly evolving field that encompasses genomics, epigenomics, transcriptomics, proteomics, and metabolomics. Researchers study omics data to receive better comprehensive knowledge on biological sciences. For humans, the omics is of tremendous interest to basic science researchers and clinicians alike in the pursuit of a deeper understanding of human health, especially in extraordinarily detailed molecular level.

Increasingly, scientists are simultaneously collecting microbiome data with other sources of Omics data such as metabolomics McHardy et al. (2013), gene expression Morgan et al. (2015), and DNA methylation Cortese et al. (2016). Conversely, many large scale genomic studies, such as large genome wide association study (GWAS) cohorts, are also collecting microbiome data Igartua et al. (2017); Wang et al. (2018). The ability to integrate microbiome data with other types of genomic data promises comprehensive achievement of many scientific objectives.

Despite the promises of these data, joint analysis presents a number of difficulties. Central challenges include standard problems for analyzing omics data including high-dimensionality, nonlinear effects, interactions among data features, modest effect sizes, and limited availability of samples. Furthermore, in analyzing multiple data types, it is also necessary to accommodate the nature of the individual data types. For example, microbiome data are subject to zero inflation, over-dispersion, compositionality, and structural (e.g. phylogenetic and functional) constraints Li (2015) while other data types have their own characteristics (e.g. spatial relationships for methylation and epistasis for genetics). Those challenges become more sophisticated when they are combined with different statistical challenges in both methodology

and application, such as missingness, factorizational data, clustered data structures and sparsity.

In this introduction, we first state the data integration problem with motivating examples with co-informative datasets, then provide an overview of several broad classes of statistical analyses for microbiome data, where we build the medhos for microbiome data upon. We end by outlining the aims of this dissertation.

1.1 Microbiome Profiling and Omics Techniques

Traditional studies of the human microbiome data collection and sampling involve microbes from samples such as skin swabs, endoscopic biopsies, and cultivation which takes weeks or months to collect, with possibility of cultivation failure or biased culture results. With the development of next generation sequencing technologies, scientists are able to collect and quantify the microbiome composition by direct DNA and RNA sequencing without laborious isolation and cultivation of individual microorganisms. The advanced metagenomic sequencing approaches to microbiome profiling is efficient, affordable and effective in measuring microbiome taxa. There are several types of sequencing approaches to summarize the taxa in a microbial community. We introduce the 16S approaches related to data generation in this dissertation.

The 16S ribosomal RNA (rRNA) gene is often sequenced to study bacterial composition. This gene is present in all bacterial species, and contains highly conserved regions that enable PCR amplification, as well as highly variable regions that enable taxonomic classification. sequences of this gene can serve as unique markers for different types of bacteria. The 16S rRNA sequencing can reveal the phylogenetic structure of microbiome compositions. Phylogenetic structures are important in building kernel spaces for microbial communities, and in this dissertation, we will consider several kernels adapting phylogenetic tree information,

such as kernels based on UniFrac distances. After preprocessing the raw sequences, the 16S sequence reads are mapped to an existing phylogenetic tree in a taxonomic-dependent way (Matsen et al., 2010). On the other hand, 16S sequence can also be clustered into operational taxonomic units (OTUs) at a certain similarity level in a taxonomic-independent way (Schloss et al., 2009), usually at a 97% similarity level. This approach is an effective way of clustering the sequencing reads based on the pairwise Hamming distances, and hence scientists can characterize OTUs by the representative DNA sequence and then assign them with taxonomic lineage, through comparison with existing microbiome databases.

However, 16S rRNA data have their own limitations. For example, they do not provide any information about bacterial gene inventory and functionality. In addition, the data do not have a property with high sensitivity in identifying bacteria at the species level. The set of taxon counts from 16S data therefore contains relative information about taxon abundance, but not absolute information. Usually we will make further steps to process the OTUs relative abundance data with some transformations and/or centralization. There are other methods such as quantitative PCR, which can provide absolute abundance data for individual bacterial species. However it is expensive given the current technologies to generate absolute abundances for the entire microbiome. Both absolute and relative abundances are essential for bacterial community health, and in fact absolute abundances contain more information than relative abundances. Given the current technological limitations and difficulties to collect absolute abundance data, most existing data only provide relative abundances. In this dissertation, we also consider relative abundance data.

In terms of Omics data, with the development of technologies such as protein microarrays, Gel-based proteomics, mass spectrometry, and high-throughput cell assays, which are among commonly employed omics techniques such as in metabolomics, interactomics, genomics, and transcriptomics, researchers are able to collect large amounts of raw data as well as summaries

in the form of lists of sequences, genes, proteins, metabolites, or SNPs.

1.2 Motivating Examples

The first motivation example is a clinical study conducted by Morgan et al. (2015) which evaluates the relationship between microbiome composition and host (human) gene expression within the context of IBD (Inflammatory Bowel Disease). Collecting both gene expression and gut microbiome profiles on IBD patients, the scientists were able to identify global associations between microbiome community profiles and the global transcriptome, and noted possible associations between microbiome profiles and the groups of genes in the interleukin 12 (IL12) pathway. However, a limitation of these analyses is that they only provide insight into associations between groups of features and no individual genes are implicated as driving the associations. Since the dimension of gene expression data is large, even with an identified association between microbiome and global transcriptome, scientists and clinicians are still facing millions or more combinations of gene expressions that are potentially associated with IBD and microbiome community.

The second motivation example is from a randomized clinical trial on women menopause symptoms and health (Mitchell et al., 2021) study (MsFLASH study). DNA extraction and polymerase chain reaction (PCR) amplification and sequencing of the 16S rRNA gene were performed at enrollment of the study as well as at 12-weeks of visits. However, only at the beginning of the exam do we have both microbiome and genomic data without any missing. In the 12-week visit, the genomics data are the same with the first visit since human genome is mostly stable, while the microbial community and composition can differ much along with time. For the 12-week visit, most of samples did not take 16S rRNA sequencing. If we only take use of the genomics data for women vaginal pH analysis, it is a waste of resources and ablations, while if we only keep the samples with full microbiome records for both visits, then

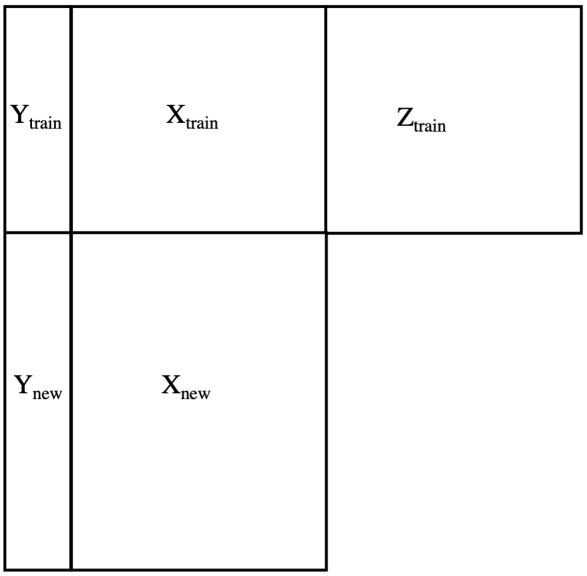

Figure 1.1: A Motivation Example Figure of How Missingness of Data are Presented in the Real Cases.

we end up with small part of samples left, which is under selection bias concerns and may violates data completeness assumption. To simplify the problem, we show the example Figure 1.1 to illustrate the data integration challenge. In this figure, both data $\mathbf{X}$ and auxillary data $\mathbf{Z}$ exist for training data corresponding to $\mathbf{Y}_{train}$, while for another new data, we nolonger have auxiliary data available and hope to classify $\mathbf{Y}_{new}$ based on known $\mathbf{X}_{new}$. Meanwhile, the existing data include repeated sampling for same participant at the same visit, which add additional complexity to the data with potential clustered correlations. Furthermore, microbiome and genomic data are co-informative in the sense that microbial community in human body can be influenced with genetic factors. It is a challenge if we can bring in microbiome data, with non negligible missing for the second half of the study, to help the analysis and classification problem reach higher accuracy.

1.3 Statistical Approaches to Microbiome-Omics Data Integration

With the questions we raise from the motivation examples, it is challenging to carry out proper analysis with microbiome data integrating with omics data, while both data have their own properties added to the statistical difficulties. For microbiome data, at the most basic level, issues come from data normalization to taxonomic abundance estimation. First, with the normalization of microbiome data to proportions, it will result in compositional data, and there are correlations among taxa. Not accounting for the data compositions and correlations using traditional statistical methodology may lead to low power and biased results. Second, microbiome data are often high-dimensional, with more taxa observed than subjects. This is due to the extraordinary amount, diversity and gene mutants of microbiomes in the human body. There are a lot of existing statistical methods developed for high-dimensional data, but rarely are they targeted and applied to compositional data. New statistical methods for analyzing high-dimensional compositional data are thus required. Furthermore, microbiome data are also subject to zero inflation and over-dispersion. Kernel methods are good choices that can be leveraged to accomplish the task to overcome these difficulties with semi-parametric factors accounted for. Finally, microbiome data sometimes are accompanied with structural information, such as bacterial phylogeny or functions (Li, 2015), so it is often more powerful and accurate across a range of statistical methods if we correctly account for those structures.

The growing availability of omics data is providing researchers with unprecedented, large-scale views of biological systems. Omics data, on the other hand, also has their statistically challenging characteristic, such that we need to treat them differently with microbiome data. First, omics data are often high-dimensional. For example, transcriptomics dataset collected from RNA-sequencing technologies can generate hundreds to thousands of transcripts (Joyce and Palsson, 2006), with modest effect sizes. Secondly, omics data are suffering from

non-linear effects accompany with network structures. An example is the gene-regulator networks uncovered with transcriptomics and genomic information and that functional-states data reveal the overall behaviour, or phenotype, of the cell or system (Yamada et al., 2021). Meanwhile, it is also important and necessary to accommodate the nature of the individual types of omics data. For genomics data, the simple genome sequencing data should be treated differently with genome annotation data, since the former one is usually noisy while later one defines the complement of proteins and functional RNAs that are available to the cell, as well as their associated regulatory elements.

Scientific questions raised from microbiome-omics data integration are posted, especially to the microbiome side, at the level of the entire microbial communities. Facing the unique challenges and opportunities, we dig into scientific questions focusing on the following centers (1) identifying single or multiple omics data that are associated with microbial communities (2) for supervised classification problem, utilizing co-informative microbial data to boost the classification accuracy (3) the methods of data integration should be adapt to multiple data characteristics and high-dimension features.

First, distance-based methods provide a flexible choice for microbial community analyses. The use of between-sample diversity to compare overall taxonomic profiles leverages the overall effect as a whole for microbiome data, given the nature of microbiome data with their sparsity and composition. Identification of genomic features related to microbiome composition can provide important biological insights into the relationship between microbiome, genomics, and diseases. However, how to identify genomic features related to overall microbial community composition, called beta-diversity analysis, is unclear. Marginally screening for associations between individual genomic markers and overall microbiome composition does not accommodate correlation among genomic markers. Variable selection is a natural strategy, yet existing approaches designed for multivariate outcomes fail for microbiome data due to

higher dimensionality than the methods accommodate and the necessity of incorporating a phylogenetic or ecologically relevant structure. We can construct distance-based analyses by computing pairwise dissimilarities among samples, where the measures of dissimilarity are ecologically relevant and may incorporate phylogenetic structure. Two types of distance-based kernels are available, where the UniFrac distance based kernel successfully incorporates phylogenetic structures while Bray-Curtis distance based kernel does not require phylogenetic structures provided. The matrix of pairwise dissimilarities is summarized. One the other hand, by adding a sequence of non-constant moving parameters to the general kernels representing omics data, we can integrate both microbiome and omics data with a kernel RV coefficient with biological meaning. With optimizing the coefficient utilizing the sequence of non-constant parameters, we can propagate the single association for single omics variables to the overall effects such that those omics variables are selected.

Second, a kernel machine framework is useful not only for testing or regression framework, but also for representing bases of the expanded Hilbert space. The representation of kernel bases is thus used to train the weights combined in the discriminant analysis to derive kernel based discriminant coordinates. The matrix of kernel discriminant coordinates based on kernel constructed with pairwise dissimilarities can be summarized by its top principal coordinates for visualization. When facing the challenges that microbiome data are missing in testing samples, the kernel discriminant coordinates are used for re-weighting omics data to achieve higher classification accuracy.

Finally, microbiome and omics data analysis are also compatible for machine learning methods. Statistically, both microbiome and omics data tend to be high-dimensional, and microbial communities are accompanied by extrinsic phylogenetic or functional structure. Scientifically, knowing which genome or metabolite input is associated with microbial composition is important for diagnostic and prognostic purposes. Statistical learning methods, in

particular penalized methods, which are broadly available in optimal scoring and quadratic programming, are attractive because they are designed for high-dimensional settings. With different choices of penalty functions, we can achieve the goal of induce sparsity by identifying variables with high signals or improving classification.

1.4 Organization of the dissertation

In this dissertation, we focus on development of statistical approaches for integrative analysis using kernel methods (Cristianini and Shawe-Taylor, 2000), with particular emphasis on integrating microbiome and other genomic data. Kernel methods are popular for high-dimensional data due to their ability to accommodate nonlinear effects and have been tailored to capture important data-type specific effects. Within this context, we will use kernel approaches to improve understanding of the relationship between data types and to improve the analyses of the individual data types in relation to others. To these ends, the dissertation is organized as follows:

In Chapter 2, We consider the problem of understanding the relationship between microbiome and other omics. We propose to use kernels to capture structure in both microbiome and genomic data. We propose to use a sparse kernel RV (KRV) coefficient to facilitate simultaneous identification of genomic features associated with overall microbiome composition. The KRV is a generalized measure of multivariate correlation between two data sets, in this case microbiome and genomics, that are embedded as kernel matrices. For microbiome data, we construct fixed, ecologically relevant kernels incorporating important ecological structure. For genomic data, we construct kernels which include feature-specific weights. Sparse estimation of the weights enables selection of genomic markers. Results show that sparse KRV can accurately select genomic features associated with microbiome composition, while accommodating ecological structure in the microbiome. We apply the approach to

identify host gene transcripts related to microbiome composition.

In Chapter 3, we develop the kernel linear discriminant analysis methods for building omics-based classification models that borrow information from other microbiome studies in construction but that can be applied to future studies that collect only omics data. The kernel linear discriminant analysis is available for the situation that omics data are of low-dimension and without requirements on the microbial data, and the methods can be achieved equally through optimal scoring or Rayleigh quotient approaches. In Chapter 4, we extend the methods of linear kernel discriminant analysis to penalized linear kernel discriminant analysis, such that high-dimensional omics data are accepted by the methods. It is flexible to choose between L1 and L2 penalties, where L1 penalties can induce sparsity while L2 penalty can be applied with broader types of omics data without sparsity assumptions.

Finally, in Chapter 5 we discuss directions for future research.

Chapter 2

SPARSE KERNEL RV COEFFICIENT[1]

2.1 Introduction

Integration of microbiome data with other types of genomic data has the potential to elucidate the mechanisms underlying the relationship between the microbiome and conditions such as cancer, HIV, menopause, and hypertension (Schwabe and Jobin, 2013; Hensley-McBain et al., 2019; Mitchell et al., 2018; Sun et al., 2019). Consequently, many microbiome studies are simultaneously collecting other omics data such as metabolomics (McHardy et al., 2013), gene expression (Morgan et al., 2015), and DNA methylation (Cortese et al., 2016). Conversely, many large scale genomic studies, such as genome wide association studies, are also collecting microbiome data (Igartua et al., 2017; Wang et al., 2018). However, although integration of these data offers more comprehensive understanding of the biology of important conditions, fully achieving the promises of these data is stymied by analytical challenges.

Central statistical challenges for integrating microbiome and other genomics include the usual problems encountered for analyzing any individual data type: high-dimensionality, nonlinear effects, interactions among data features, modest effect sizes, and limited availability of samples. Furthermore, in analyzing multiple data types, it is also necessary to accommodate the nature of the individual data types. For example, microbiome data are subject to zero inflation, over-dispersion, compositionality, and structural (such as phylogenetic and functional) constraints (Li, 2015). Other data types offer their own quirks and have

[1]The contents of this chapter are based on the paper Identifying Genomic Features Related to Microbiome Community Composition Using a Sparse Kernel RV Coefficient

characteristics that need to be accommodated as well, such as epistatic effects for genetics or spatial characteristics of DNA methylation. We propose to address these challenges and develop an approach for identifying individual genomic features that are related to overall microbiome composition at the community level.

Community level analysis, also called beta-diversity analysis, is a powerful and commonly used strategy in microbiome studies (Huttenhower et al., 2012; Li, 2015). Instead of focusing on individual bacterial taxa, this mode of analysis looks for overall, global shifts in the microbiome in relation to variables of interest. This mode of analysis is often more powerful than individual taxon analysis, particularly when individual taxa show modest yet concerted shifts, and emphasizes differences in microbial community structure. Operationally, community level analysis typically involves computing a distance (between samples) matrix based on the microbiome data. The distances are ecologically informed and may capture important aspects of microbiome data including phylogenetic information or qualitative effects (Lozupone and Knight, 2005; Lozupone et al., 2011; Chen et al., 2012). Classically, these distances matrices were then regressed on particular variables of interest to assess associations (Anderson, 2001). More recently, kernel based approaches have been developed for assessing associations between microbiome beta-diversity and individual variables (Chen and Li, 2013; Zhao et al., 2015; Wu et al., 2016; Plantinga et al., 2017), and these approaches have been extended to examine multivariate variables (Zhan et al., 2017b), including high-dimensional genomic features (Zhan et al., 2017a).

A limitation of current beta-diversity analysis approaches is that they are restricted to testing, and outside of marginal analysis, no methods exist for identifying individual genomic features associated with overall microbiome composition. Marginal analysis of the genomic features is feasible, but fails to consider correlation and does not reflect the importance of each feature in the presence of other genomic features. An alternative strategy

is to conduct variable selection treating the microbiome as a usual multivariate outcome. Possible approaches include sparse multivariate regression approaches such as sparse linear mixed models and generalized estimating equations (Bondell et al., 2010; Fu, 2003), or sparse multivariate correlation measures such as sparse canonical correlation analysis (sCCA) (Witten et al., 2009; Iaci et al., 2010). However, sparse regression approaches often cannot handle the large number of microbial taxa, and neither sparse regression nor sCCA directly incorporate the microbiome structure that is embedded within distances/kernels. Finally, most existing sparse multivariate regression or multivariate correlation approaches implicitly assume linearity in the genomic features. Given these limitations, it remains unclear, in practice, how to select individual features related to microbiome composition.

Our work is motivated by a study conducted by Morgan et al. (2015) which evaluates the relationship between microbiome composition and host (human) gene expression within the context of IBD. Collecting both gene expression and gut microbiome profiles on IBD patients, the investigators were able to identify global associations between microbiome community profiles and the global transcriptome, and noted possible associations between microbiome profiles and the groups of genes in the interleukin 12 (IL12) pathway. We later reproduced the global associations and formally evaluated the associations with the IL12 pathway (Zhan et al., 2017a). However, a limitation of these analyses is that they only provide insight into associations between groups of features and no individual genes are implicated as driving the associations.

To identify individual genomic features related to microbiome community composition, we propose to introduce sparsity into the kernel RV coefficient (KRV) (Zhan et al., 2017a) by using the previously developed KNIFE framework (Allen, 2013). A generalization of the classical RV coefficient, the KRV assesses the overall correlation between microbiome community composition and another set of genomic features after embedding both data types

in kernel matrices. For microbiome data, as in previous kernel methods for beta-diversity, we construct fixed, ecologically relevant kernels which allows for capture of structure and data characteristics. For genomic data, we can similarly tailor kernels, but variable selection is difficult as there are no explicit coefficients or feature-specific parameters. Therefore, following Allen (2013), we construct modified kernels by adding a new vector of feature-specific weights. Using the full microbiome kernels and the modified genomic kernels, we can then maximize the KRV with an L_1 penalty on the feature-specific weights, leading to sparse estimation and selection of the genomic features.

As a practice paper, we emphasize that the novelty of this project lies in the particular application which is an important applied problem for which no satisfactory approach currently exists. Our contribution represents a combination and translation of different statistical tools with aim towards utility rather than a *de novo* statistical method. As noted earlier, alternative frameworks for assessing sparse multivariate correlation are possible, but the importance of utilizing the kernel framework is due to the ability to capture community structure in a fashion that is meaningful and natural to microbiome scientists, i.e. the use of ecologically informed distances and kernels. Conversely, outside of the KNIFE approach, few generally applicable strategies are available for variable selection with kernel methods.

The remainder of this article is organized as follows. In the next section, we present our proposed sparse KRV strategy, first briefly reviewing the KRV and then describing the incorporation of feature-specific weights according to different kernel choices. We then construct a penalized objective function, and corresponding computational approach, to sparsely estimate the weights. In Section 3, we present simulation studies across several realistic scenarios to validate and study the empirical properties of our approach. In Section 4, we apply the sparse KRV to the motivating study to identify genomic features driving associations with microbiome composition. Using these data, we further present additional

validation using sample splitting. We conclude with a brief discussion in Section 5.

2.2 Sparse KRV for Selecting Genomic Features

We consider a study in which we have microbiome and genomic data collected on the same individuals. Let $\mathbf{Y}_i = [y_{i1}, y_{i2}, \ldots, y_{ir}]'$ denote the vector of r microbiome features for the i^{th} individual ($i = 1, \ldots, n$) and $\mathbf{Z}_i = [z_{i1}, z_{i2}, \ldots, z_{ip}]'$ denote the vector of p genomic features for the i^{th} individual. The genomic features may represent all the measured features or may be restricted to a subset of interest, e.g. genes within a pathway or SNPs within a gene. The objective is to identify a subset of $\mathbf{z}_{\cdot j}$'s associated with $\mathbf{Y}$, accommodating ecological or functional structure in $\mathbf{Y}$.

In this section, we first review the existing KRV approach to capturing generalized multivariate correlation between structured data, then describe how to conduct feature selection.

2.2.1 Kernel RV coefficient

The classical RV coefficient was developed as a measure of linear correlation between two sets of multivariate measurements collected on the same samples (Escoufier, 1973). The kernel RV (KRV) coefficient (Zhan et al., 2017a) is a generalization of the RV coefficient that facilitates capture of more complex structure in the data as well as nonlinear relationships among the measurements. In particular, to accommodate structure, we define $\mathbf{K}_y$ and $\mathbf{K}_z$ to be kernel matrices constructed based on the microbiome profiles and genomic features, respectively. Then the KRV statistic is

$$R = \frac{tr(\mathbf{K}_y \mathbf{K}_z)}{\sqrt{tr(\mathbf{K}_y^2) tr(\mathbf{K}_z^2)}}, \tag{2.1}$$

which is a generalized measure of correlation on the kernelized data, with large KRV indicating stronger dependency and zero KRV indicating uncorrelatedness.

For microbiome data, following prior work (Zhao et al., 2015), we define $\mathbf{K}_y$ by transforming existing distance and dissimilarity measures that are commonly used in microbiome analysis, such as the UniFrac distance and Bray-Curtis dissimilarity. Specifically, if $\mathbf{D}$ is an $n \times n$ matrix of pairwise distances between microbiome community profiles, then we set $\mathbf{K}_y = -\frac{1}{2}(\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{n})\mathbf{D}^2(\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{n})^T$. Importantly, depending on the choice of distance used, $\mathbf{D}$ captures structural aspects of the data and potential forms of association. For example, the UniFrac family of distances directly incorporates phylogenetic information, with unweighted UniFrac focusing on qualitatitive differences in community structure (i.e., taxon presence/absence), weighted UniFrac focusing on relative abundance, and generalized UniFrac as a hybrid. The variety of distances and dissimilarities has been thoroughly characterized, with different choices emphasizing different aspects of the community composition.

For genomic data, popular choices include commonly used positive semi-definite kernels such as the linear kernel ($\mathbf{K}_{ij}(\mathbf{Z}) = \mathcal{K}_z(\mathbf{Z}_i, \mathbf{Z}_{i'}) = \sum_{j=1}^{p} z_{ij}z_{i'j}$), which is the default and most commonly used kernel for analyzing genetic variants. Other kernels include the generic Gaussian radial basis function (RBF) kernel ($\mathbf{K}_{ij}(\mathbf{Z}) = \mathcal{K}_z(\mathbf{Z}_i, \mathbf{Z}_{i'}) = \exp\left\{-\sum_{j=1}^{p}(z_{ij} - z_{i'j})^2/\rho\right\}$), and the genetic identity-by-state kernel ($\mathbf{K}_{ij}(\mathbf{Z}, \mathbf{c}) = \mathcal{K}_z(\mathbf{Z}_i, \mathbf{Z}_{i'};) = (2p)^{-1}\sum_{j=1}^{p}|z_{ij} - z_{i'j}|$).

Zhan et al. (2017a) was not the first to derive R. In particular, R is also called the kernel alignment statistic (Cristianini et al., 2002) in the statistical learning literature and is closely related to the Hilbert-Schmidt independence criterion (Gretton et al., 2005) and distance covariance (Székely et al., 2009). Starting from the RV perspective primarily enables transfer of results from the RV framework to the KRV in order to facilitate hypothesis testing. Regardless of the background from which one approaches the statistic, calculation of R requires utilization of all variables in $\mathbf{Z}$ with no explicit selection or prioritization of individual

features.

2.2.2 Sparse kernel RV coefficient

Our objective is to select genomic features rather than assess global correlation. To achieve selection, we would ideally set the contribution of some genomic features to be zero. However, in contrast to sparse penalized regression approaches, no explicit regression coefficients exist; nor are there any weights/loadings to shrink as in canonical correlation analysis and principal component analysis. Thus, following the general framework of the KNIFE approach (Allen, 2013), we redefine kernels for the genomic features and incorporate weights for each specific feature. Examples of redefined genomic kernels with an added weight vector, $\mathbf{c}$, are the linear kernel, $\mathbf{K}_{ij}(\mathbf{Z}, \mathbf{c}) = \mathcal{K}_z(\mathbf{Z}_i, \mathbf{Z}_{i'}; \mathbf{c}) = \sum_{j=1}^{p} c_j z_{ij} z_{i'j}$; the Gaussian RBF kernel, $\mathbf{K}_{ij}(\mathbf{Z}, \mathbf{c}) = \mathcal{K}_z(\mathbf{Z}_i, \mathbf{Z}_{i'}; \mathbf{c}) = \exp\left\{-\sum_{j=1}^{p} c_j (z_{ij} - z_{i'j})^2/\rho\right\}$; and the identity-by-state (IBS) kernel, $\mathbf{K}_{ij}(\mathbf{Z}, \mathbf{c}) = \mathcal{K}_z(\mathbf{Z}_i, \mathbf{Z}_{i'}; \mathbf{c}) = (2p)^{-1} \sum_{j=1}^{p} c_j IBS(z_{ij}, z_{i'j})$. In each case, the c_j's are feature specific weights which index the contribution of the corresponding variables to the kernel function and matrix. Thus, to achieve variable selection, we would like to shrink some of the c_j to exactly zero such that the corresponding feature no longer contributes.

To enable sparse estimation of $\mathbf{c}$, we can maximize the KRV, substituting the usual kernels with corresponding modified kernels, subject to an additional L_1 constraint:

$$\widehat{\mathbf{c}} = \underset{\mathbf{c}}{\operatorname{argmax}}\, R(c) = \underset{\mathbf{c}}{\operatorname{argmax}} \frac{tr(\mathbf{K}_y \mathbf{K}_z(\mathbf{c}))}{\sqrt{tr(\mathbf{K}_y^2) tr(\mathbf{K}_z(\mathbf{c})^2)}}, \quad \text{s.t.} \sum_{j=1}^{p} |c_j| \leq t. \tag{2.2}$$

The form (2.2) appears challenging to approach, but we can invert the denominator and numerator of equation (2.2), fix the components that do not contain c_j's, and subject the component of the previous denominator containing $\mathbf{c}$ to an equality constraint. Then after simplification as noted below, the solution to (2.2) can be obtained via the equivalent objective

function

$$\widehat{\mathbf{c}} = \arg\min_{\mathbf{c}} f(\mathbf{c}) + s\|\mathbf{c}\|_1 \qquad \text{s.t. } h(\mathbf{c}) = M \tag{2.3}$$

where $f(\mathbf{c}) = tr(\mathbf{K}_z(\mathbf{c})^2)$, $h(\mathbf{c}) = tr(\mathbf{K}_y\mathbf{K}_z(\mathbf{c}))$, and $M = w\sqrt{tr(\mathbf{K}_y^2)}$. Intuitively, the component of the objective that we directly optimize is the term in the denominator involving $\mathbf{c}$, denoted $f(\mathbf{c})$. We scale the coefficients appropriately by penalizing so that the numerator, denoted $h(\mathbf{c})$, is equal to the remaining component of the denominator (M, not dependent on $\mathbf{c}$). The tuning parameter s controls the degree of sparsity and can be selected by a grid search to maximize the cross-validated KRV coefficient. For derivation details, please refer to the supplementary materials of this article in Web Appendix B.

For fixed s, (2.3) is a standard optimization problem with a linear inequality (the L_1 term) and linear equality constraints which are solvable by standard optimization approaches such as the alternating direction method of multipliers algorithm (ADMM) (Boyd et al., 2011). In the following section, we outline the optimization procedure for specific cases.

2.2.3 Modified ADMM algorithm for different choices of kernels

We solve the problem of equation (2.3) using a modified alternating direction method of multipliers (ADMM) algorithm (Boyd et al., 2011). The original ADMM algorithm converts penalized optimization problems to equivalent optimization problems with penalties re-expressed as linear constraints. As (2.3) already has a linear constraint, we modify the ADMM algorithm for sparse KRV to accommodate the additional constraint. We therefore solve (2.3) by optimizing

$$\widehat{\mathbf{c}} = \arg\min_{\mathbf{c}} f(\mathbf{c}) + s\|\mathbf{z}\|_1 \qquad \text{s.t. } h(\mathbf{c}) = M,\ \mathbf{c} = \mathbf{z} \tag{2.4}$$

The augmented Lagrangian for this problem is

$$L_w(\mathbf{c}, \mathbf{z}, \mu, \lambda) = f(\mathbf{c}) + s\|\mathbf{z}\|_1 + \mu(h(\mathbf{c}) - M) + \frac{w}{2}(h(\mathbf{c}) - M)^2 \\ +\lambda^T(\mathbf{c} - \mathbf{z}) + \frac{w}{2}\|\mathbf{c} - \mathbf{z}\|^2 \tag{2.5}$$

where μ and λ are Lagrangian multipliers for the two linear constraints and w is the augmented Lagrangian parameter. The general approach we take is to iteratively update our targets, $\mathbf{c}$ and $\mathbf{z}$, and the Lagrangian multipliers until convergence, indicating that the global minimum of the augmented Lagrangian has been reached. The details of the algorithm are presented below.

Multiple-constraint ADMM for solving sparse KRV problem:

1. Initialize $\mathbf{c}^{(0)}$ and $\mathbf{z}^{(0)}$ with $\mathbf{0}$ or warm starts, $\mu^{(0)} = 0$, $\lambda^{(0)} = 0$

2. Update each parameter according to:

   - $\mathbf{c}^{(k+1)} \leftarrow \arg\min_{\mathbf{c}} L_w(\mathbf{c}, \mathbf{z}^{(k)}, \mu^{(k)}, \lambda^{(k)})$
   - $\mathbf{z}^{(k+1)} \leftarrow \arg\min_{\mathbf{z}} L_w(\mathbf{c}^{(k+1)}, \mathbf{z}, \mu^{(k)}, \lambda^{(k)})$
   - $\mu^{(k+1)} \leftarrow \mu^{(k)} + w(h(\mathbf{c}^{(k+1)}) - M)$
   - $\lambda^{(k+1)} \leftarrow \lambda^{(k)} + w(\mathbf{c}^{(k+1)} - \mathbf{z}^{(k+1)})$

3. Update $k \leftarrow k + 1$; repeat (2) until convergence.

This algorithm is guaranteed to converge to the global minimal of function 2.4 under Section 3.2 of Boyd et al. (2011), as long as the kernel matrices for both the genomic data and microbiome data are positive semi-definite. While UniFrac distances and Bray-Curtis dissimilarities may yield uncorrected kernels with small negative eigenvalues, the standard transformation of distance/dissimilarity matrices to kernel matrices includes a correction to

guarantee positive semi-definiteness. For the linear kernel, optimization speed can be improved by using an explicit formulation for the parameter updates at each iteration. However, for the Gaussian kernel, there is no explicit solution at each step, so we use the restricted Newton's method (Byrd et al., 1995) to optimize. Implementation and programming details are shown in the Appendix.

2.3 Simulation Studies

We now assess the feature selection performance of sparse KRV across a range of simulation scenarios.

2.3.1 Simulation methods

We generate microbiome data using the approach described in Zhao et al. (2015). Specifically, OTU counts for each individual are generated from a Dirichlet-Multinomial distribution with parameters estimated from a real throat microbiome dataset consisting of 856 OTUs measured on 60 samples (Charlson et al., 2010). We generate 1000 reads, distributed among these 856 OTUs, for each individual.

For genomic data, we begin by considering a linear association with the microbiome. We generate genomic data based on the abundance of OTUs in an index set $\mathcal{A}$ via

$$\mathbf{Z}_{ij} = \sum_{k \in \mathcal{A}_j} \alpha_k \frac{Y_{i(k)}}{\bar{Y}_{(k)}} + \epsilon_{ij} \tag{2.6}$$

where α_k is an indicator for whether $k \in \mathcal{A}_j$ and $\bar{Y}_{(k)}$ denotes the average read count of the (k)-th OTU across samples.

The choice of associated taxa $\mathcal{A}_j$ permits several true forms of association between the microbiome and genomic features. In particular, we vary the level of phylogenetic relationships

among associated taxa and the abundance of taxa in the associated set. The true form of association (including its abundance and dependence on phylogeny) will affect which microbiome kernels perform best.

Under simulation scenario 1, the index set $\mathcal{A}$ includes the ten most abundant OTUs among all simulated samples, without taking phylogenetic information into account. Under simulation scenario 2, we perform the partitioning around medoids algorithm to partition the 856 OTUs into 20 clusters, then choose the index set as one of the most highly abundant clusters, which generally includes 25-39 OTUs comprising 8.4%-11.0% of total read counts. In this way we simulate outcome that are related to a cluster of taxa depending on a phylogenetic tree. Finally, under simulation scenario 3, we partition taxa as in scenario 2 and select a phylogenetic cluster with low abundance. The low abundance cluster consist of 18-40 OTUs, with 1.2%-2.1% of total counts. For all three scenarios, error terms ϵ_{ij} are independently generated from random standard normal distribution $\mathcal{N}(0, 1)$.

For all simulation scenarios, we use the weighted UniFrac, unweighted UniFrac, generalized UniFrac with $\alpha = 0.5$, and Bray-Curtis distances to construct microbiome kernels using the transformation $\mathbf{K} = -\frac{1}{2}(\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{n})\mathbf{D}^2(\mathbf{I} - \frac{\mathbf{1}\mathbf{1}^T}{n})^T$, where D_{ij} is the distance or dissimilarity between two microbiome samples. We choose these four kernels because they represent a variety of ecologically relevent features of microbiome data. Specifically, the UniFrac kernels account for phylogenetic information as well as OTU abundance (weighted, generalized) and presence/absence (unweighted, generalized), and the Bray-Curtis kernel relies only on abundance, ignoring phylogeny. Positive semi-definiteness is enforced by performing an eigendecomposition of $\mathbf{K}$ and replacing all negative eigenvalues with their absolute values, as described in Zhao et al. (2015).

We simulate data with sample size $n = 50$, number of OTUs $p = 856$, and number of genomic features $q = 50$. The sparsity of the association between the genomic features and

the microbiome is varied by choosing between 1 and 30 features to have a true association with the microbiome. Performance is evaluated by three measurements: the number of truly associated features that were selected by the method (true discovery number, or TRUE), the number of unassociated features that were selected by the method (false discovery number, or FALSE), and the Hamming distance, which is the sum of false discoveries (unassociated features that were selected) and false non-discoveries (associated features that were not selected). All metrics are averaged across 100 simulations. Tuning parameters of sparse KRV simulations are chosen by five-fold cross validation. We compare the sparse KRV approach to sparse canonical correlation (sparse CCA, Witten et al. (2009)), with tuning parameters selected by 100 permutations.

2.3.2 Simulation results

Table 2.1 presents the true discovery number, false discovery number, and Hamming distance under simulation scenario 1 with 5, 10, 20, and 30 associated genomic features. For sparse KRV, we consider UniFrac distances (weighted, generalized, and unweighted) and the Bray-Curtis dissimilarity to capture microbial information. For comparison, we also include corresponding results from sparse CCA.

We find that when 5 genomic features are truly associated with microbiome composition, the sparse KRV method selects an average of 4.83 to 4.93 of the associated features across different microbial kernels. Among different choices of kernels, the Bray-Curtis kernel has the smallest average number of false discoveries and the highest average true discoveries (0.01 and 4.93, respectively), whereas the unweighted UniFrac kernel has the highest false discovery number and smallest true discovery number (0.11 and 4.83). The weighted UniFrac and generalized UniFrac kernels have intermediate values for both, yielding false discovery numbers below 0.1 and average true discovery numbers of 4.89. Meanwhile, sparse CCA

provides an average of 3.03 true discoveries and 6.92 false discoveries.

With increasing numbers of associated genomic features, the true discovery number also increases such that the true discovery rate is always higher than 94%. False discovery numbers for all four kernels remain relatively low, almost always below 10% (the only exception in sparse KRV is the unweighted UniFrac with 30 truly associated features). Comparing different kernels, the Bray-Curtis kernel uniformly outperforms the three UniFrac kernels in both true discovery rate and false discovery rate (and thus also in Hamming distance) in scenario 1. This is to be expected, since the main strength of the UniFrac kernels is their incorporation of phylogenetic information, and phylogeny is irrelevant in simulation scenario 1.

In sum, although the true discovery rate is not perfect for higher numbers of associated genomic features, we find that the results are indeed acceptable as a feature selection method, especially considering the low false discovery rate. Although sparse canonical correlation works well for multivariate normal data, our results demonstrate that sparse CCA has much lower true positive rates and much higher false positive rates than sparse KRV in our data setting due to the correlation structure and non-normal distribution of microbiome OTU data.

The corresponding results from scenario 2, in which an abundant phylogenetic cluster of taxa is associated with the outcome, are displayed in Table 2.2. We again find that the sparse KRV method with a variety of microbiome kernels has a high true discovery rate when we simulate 5 genomic features associated with microbiome composition, averaging over 4.8 true positives. Bray-Curtis has an average true positive discovery number of 4.90 and an average false discovery number of 0.10; the weighted UniFrac kernel also has an average true discovery number of 4.90 and false discovery number of 0.13; unweighted UniFrac has an average true positive discovery number of 4.88 and an average false discovery number of 0.07; and generalized UniFrac kernel has an average true discovery number of 4.98 and false

Method	Metric	5	10	20	30
$\mathbf{K}_{BC}$	TRUE	4.93	9.78	19.32	28.78
	FALSE	0.01	0.27	0.62	1.40
	HAM	0.08	0.49	1.30	2.62
$\mathbf{K}_{w}$	TRUE	4.89	9.66	18.99	28.29
	FALSE	0.08	0.43	0.82	1.80
	HAM	0.19	0.77	1.83	3.51
$\mathbf{K}_{u}$	TRUE	4.83	9.59	18.89	28.39
	FALSE	0.11	0.42	0.86	3.84
	HAM	0.28	0.83	1.97	5.45
$\mathbf{K}_{50}$	TRUE	4.89	9.67	18.96	28.33
	FALSE	0.05	0.42	0.62	2.28
	HAM	0.16	0.75	1.66	3.95
CCA	TRUE	3.03	5.36	6.97	8.93
	FALSE	6.92	6.98	1.97	0
	HAM	8.89	11.62	15.00	21.07

Table 2.1: Scenario 1 results: 10 most abundant OTUs

discovery number of 0.28. Based on the degree of similarity in the performance across kernel choices (true discovery number 4.88-4.90, false discovery number 0.07-0.17), in the case of 5 associated features, no single kernel is clearly superior.

As the number of related features increases, all four kernels retain high true discovery rates and low numbers of false discoveries. All four kernels again generally perform comparably by both accuracy measures, and the differences among methods are much smaller than in simulation scenario 1. In contrast to scenario 1, in which phylogenetic information was irrelevant, the true association is between the genomic features and phylogenetically clustered taxa. Though none of the kernels in this scenario performs substantially better than the others, the discrepancy between scenarios 1 and 2 displays the importance of kernel choice for selection accuracy: the UniFrac kernels show better relative performance when phylogeny plays a role in the association (i.e., comparing scenario 2 to scenario 1). Depending on the phylogenetic structure of the associated taxa, either the Bray-Curtis or the UniFrac kernels may be superior in particular cases.

Of note, in both simulation scenarios 1 and 2, the false discovery number does not increase linearly with the number of features that are truly associated with microbiome data. This is because our method uses cross validation to choose tuning parameters that maximize the KRV coefficient. When the true signal size to be distinguished is small (such as when many genomic features are associated with the outcome), the region surrounding the optimal choice of tuning parameter is relatively flat, which results in more deviation from the optimal parameter and a higher number of false discoveries.

For sparse CCA in scenario 2, we again see lower average true positive discovery rates and substantially more false positive discoveries with sparse CCA than with sparse KRV, regardless of which kernel is used in sparse KRV.

Method	Metric	5	10	20	30
$\mathbf{K}_{BC}$	TRUE	4.90	9.72	19.23	28.81
	FALSE	0.10	0.32	0.87	3.02
	HAM	0.20	0.60	1.64	4.21
$\mathbf{K}_{w}$	TRUE	4.90	9.71	19.19	28.84
	FALSE	0.13	0.35	0.73	3.27
	HAM	0.23	0.64	1.54	4.43
$\mathbf{K}_{u}$	TRUE	4.88	9.69	19.16	28.76
	FALSE	0.07	0.44	0.68	3.10
	HAM	0.19	0.75	1.52	4.34
$\mathbf{K}_{50}$	TRUE	4.89	9.69	19.11	28.78
	FALSE	0.17	0.47	0.88	3.79
	HAM	0.28	0.78	1.77	5.01
CCA	TRUE	3.33	5.87	11.29	12.5
	FALSE	12.21	12.29	4.89	0
	HAM	13.88	16.42	13.6	17.5

Table 2.2: Scenario 2 results: A dense cluster

Finally, we present results under simulation scenario 3 in Table 2.3. When 5 genomic features are truly associated with microbiome composition, the sparse KRV method selects, on average, 3.58 to 3.83 of the associated features across the different microbial kernels. Comparing different choices of kernels, the weighted UniFrac kernel performs the best in

terms of Hamming distance. It has the largest average number of true discoveries at 5, 10, 20 and smallest average false discovery at 5, 10, and 30. The other kernels (Bray-Curtis, unweighted UniFrac, and generalized UniFrac) can also distinguish more than half of the associated features.

Compared with Table 2.1 and Table 2.2, the true discovery number is universally lower in Table 2.3. Especially when more than ten genomic features are associated with the microbiome, the sparse KRV method is able to correctly select roughly half of them, although the average number of false discoveries remains similar to other scenarios. This is a result of the low abundance of OTUs that were selected to be associated with the genomic features. When we introduce random errors of the same size in simulations with lower signal strength, a loss of power is a natural result. Despite poorer ability to detect associated features compared to other scenarios, the sparse KRV method still outperforms sparse CCA in both measures of detection accuracy under this setting. Hence we find that across a range of ecologically relevant scenarios for the microbiome, sparse KRV is much better able to distinguish genomic features that are associated with microbiome composition than existing methods.

We also conduct simulations with non-linear associations between microbiome and genomic features using the Gaussian kernel for genomic features. The simulation results for these scenarios are included in the Supplement (Web Appendix A), and broadly show that when we use a Gaussian kernel for genomic features under non-linear simulation settings, the true discovery number remains high, but with a higher false discovery number compared to the linear settings. More details are presented in the Web Appendix A as supplementary materials to this article.

Method	Metric	5	10	20	30
$\mathbf{K}_{BC}$	TRUE	3.59	6.61	10.80	16.00
	FALSE	0.15	0.39	0.87	2.80
	HAM	1.56	3.78	10.07	16.80
$\mathbf{K}_{w}$	TRUE	3.83	6.84	11.41	16.42
	FALSE	0.10	0.36	0.96	2.61
	HAM	1.27	3.52	9.55	16.19
$\mathbf{K}_{u}$	TRUE	3.60	6.47	10.88	16.11
	FALSE	0.10	0.39	0.83	3.06
	HAM	1.50	3.92	9.95	16.95
$\mathbf{K}_{50}$	TRUE	3.58	6.53	10.36	17.21
	FALSE	0.14	0.42	0.81	4.48
	HAM	1.56	3.89	10.45	17.27
CCA	TRUE	3.27	5.79	9.95	12.30
	FALSE	9.64	11.58	5.29	0
	HAM	11.37	15.79	15.34	17.7

Table 2.3: Scenario 3 results: A rare cluster

2.4 Analysis of Inflammatory Bowel Disease (IBD) Data

We apply the proposed sparse KRV method to select gene expression features that are associated with microbiome composition in an inflammatory bowel disease (IBD) study, evaluating the selection results by split samples. The goal of the study was to understand how the microbiome and host gene expression may interact and what role they jointly play in the development of pouchitis, or inflammation of the ileal pouch constructed after removal of the colon and rectum (Morgan et al., 2015). After quality control, host gene expression data and microbiome data are available for a total of 255 pouch and pre-pouch ileum (PPI) samples. Morgan et al. (2015) mentions the enrichment of microbiome associated host transcript patterns within the interleukin-12 (IL12) pathway, though no formal statistical results were reported. In our application, we focus on selecting features from the IL12 pathway that are associated with microbiome data. After excluding singleton and doubleton OTUs, the set of features used in our analysis includes 5153 OTUs. For genomic data, we include 21 host transcripts in the IL12 pathway.

We choose a linear kernel structure for the gene expression data and perform the analysis using sparse KRV with Bray-Curtis, weighted UniFrac, unweighted UniFrac, and generalized UniFrac kernels for the microbiome data. Feature selection results are presented in Table 2.4. For each selected gene (identified by NCBI ID, abbreviation, and chromosome), we report its weight coefficient for all four kernels (i.e., the value in $\mathbf{c}$ corresponding to that gene), where a weight of 0 indicates that the gene was not selected using that kernel.

When we apply linear structure for gene expression data, among 21 features in the IL12 pathway, sparse KRV with the Bray-Curtis kernel selects 6 features; with the weighted UniFrac kernel, 5 features; with the unweighted UniFrac kernel, 8 features; and with the generalized UniFrac kernel, 5 features. Though different subsets of features were selected using each of the four kernels, there are five genes – interferon gamma (IFNG), interleukin-18 receptor 1 (IL18R1), tyrosine kinase 2 (TYK2), mitogen-activated protein kinase 8 (MAPK8), and JUN – that are selected using all four kernels. IL12RB1 is selected using two out of four kernels and ETV5, CD3D and CD3G are each selected using one out of four kernels.

Table 2.4 also includes results using the Gaussian kernel for the IL12 pathway features. Among 21 features in IL12 pathway, there are 4 features selected by sparse KRV with the Bray-Curtis kernel; 8 features with weighted UniFrac; 5 features with unweighted UniFrac; and 6 features with generalized UniFrac. Among the selected genes, three (IFNG, IL18R1, TYK2) are selected using all four kernels; two genes (JUN, MAPK8) are selected using three kernels; IL12RB2 is selected using two kernels, and three genes are selected by one models. These results are broadly consistent with the results of sparse KRV with the linear genomic kernel; the same 5 features are selected by most of the models, and both methods include a few genes that are selected only once or twice.

Comparing these real data results with the simulation results, we find that though our simulations show high true positive and relatively low false positive results for all four

kernels, with relatively similar performance across kernels, the behaviors of different kernels on real data may vary more substantially. There are several possible underlying causes of this variability. For example, an exact linear association almost never holds in the real world, and it could be that these associations are driven at least in part by rare taxa, for which the true discovery rate tends to be lower (so that different kernels reveal different facets of the association). Also, there are more complicated phylogenetic and functional relationships among taxa to be explored. As new ecological relationships are discovered and new measures of dissimilarity continue to be introduced, the additional information can be easily incorporated into the sparse KRV analysis through choice of an appropriate microbiome kernel.

Kernel	Gene ID	Gene	Chr.	$\mathbf{K}_{BC}$	$\mathbf{K}_{w}$	$\mathbf{K}_{u}$	$\mathbf{K}_{50}$
	3458	IFNG	12	94.01	135.24	150.28	110.23
	8809	IL18R1	2	61.91	52.61	127.10	111.07
	7297	TYK2	19	43.22	28.42	53.47	56.12
	5599	MAPK8	10	38.57	38.59	48.75	55.51
Linear	3725	JUN	1	11.76	8.86	3.93	19.83
	2119	ETV5	3	0	0	43.05	0
	3594	IL12RB1	19	0	19.33	36.97	0
	915	CD3D	11	0	0	17.50	0
	917	CD3G	11	6.39	0	0	0
	3458	IFNG	12	41.81	148.22	18.80	101.33
	8809	IL18R1	2	75.61	135.06	18.86	23.58
	7297	TYK2	19	3.76	12.41	16.33	7.54
	3725	JUN	1	0	12.23	4.01	1.48
Gaussian	5599	MAPK8	10	6.27	5.10	0	1.09
	3595	IL12RB2	1	0	0	20.05	7.40
	6775	STAT4	2	0	40.98	0	0
	3594	IL12RB1	19	0	22.18	0	0
	919	CD247	1	0	3.12	0	0

Table 2.4: Sparse KRV results for IL12 pathway in IBD data, incorporating Gaussian kernel for genomic data.

To further illustrate our proposed method and compare different kernel choices, we perform

the KRV test (Zhan et al., 2017a) on the full data set as well as selected features to compare statistical power. KRV is a fast, small-sample kernel test of independence between microbiome composition and genomic data with accurate type I error control (Zhan et al., 2017a). Here, we use KRV to test the association between (1) the full set of 21 genomic features and the microbiome, and (2) subsets of features selected by sparse KRV. The results of this analysis using all subjects are presented in the first two columns of Table 2.5. Comparing the KRV test using all genomic features to using only those selected by sparse KRV, we find higher p-values for KRV with all features in every scenario except a Gaussian genomic kernel with a Bray-Curtis microbiome kernel. That is, the p-value is almost always more significant after selection. These results indicate that feature selection has the potential to improve power for testing the association between genomic data and the microbiome. Comparing the linear kernel to the Gaussian kernel, in this dataset, the Gaussian kernel had higher power when paired with all UniFrac microbiome kernels (but not the Bray-Curtis kernel).

However, these results could be compromised due to use of the same data for feature selection and testing. We therefore validate our results by randomly splitting the IBD subjects into two groups ($n_1 = 128$, $n_2 = 127$). We then apply the sparse KRV method on set 1 (training data) to obtain a feature list, and evaluate the genomic features selected by set 1 using the KRV test on set 2 (validation data, with the selected feature list or the full set of features). These results are shown in the last two columns of Table 2.5. Comparing the p-values of set 2 before and after selection of a feature list from set 1, we find higher power on the validation set post-selection for all kernels except the unweighted UniFrac when the genomic features are summarized by a linear kernel. With the Gaussian kernel, the p-values are lower post-selection for the Bray-Curtis and generalized UniFrac kernels, but higher for weighted and unweighted UniFrac. Meanwhile, because the unweighted UniFrac metric is focused on taxon presence or absence and therefore places the most weight on rare taxa, it

can be highly variable with data splitting (particularly given that there are 5153 OTUs in this dataset, mostly rare). Therefore in a real data analysis, we suggest choosing the kernel that best matches expected microbiome data structure, which will improve power.

Data		Full data		Train Data		Vali. Data	
Features		All	Sparse KRV	All	Sparse KRV	All	Validated
Linear	$\mathbf{K}_{BC}$	0.0070	0.0053	0.3957	0.2158	0.0028	0.0002
	$\mathbf{K}_{w}$	0.0603	0.0321	0.0832	0.0239	0.1253	0.0209
	$\mathbf{K}_{u}$	0.0289	0.0265	0.1171	0.0611	0.0111	0.4483
	$\mathbf{K}_{50}$	0.0195	0.0161	0.0964	0.0992	0.0404	0.0018
Gaussian	$\mathbf{K}_{BC}$	0.0029	0.0579	0.5248	0.3268	0.0613	0.0354
	$\mathbf{K}_{w}$	0.0087	0.0009	0.0435	0.0364	0.0166	0.0959
	$\mathbf{K}_{u}$	0.0257	0.0077	0.1074	0.0578	0.0368	0.4494
	$\mathbf{K}_{50}$	0.0041	0.0008	0.1260	0.0423	0.0079	0.0048

Table 2.5: KRV test P-values of IBD data with two sample splitting

2.5 Discussion

This is a practice paper focused on borrowing ideas from different areas and translating them to the microbiome framework. Specifically, this work combines the existing KNIFE and KRV frameworks to enable powerful selection of genomic features related to microbiome community profiles while accommodating key structures in the microbiome data, a problem for which alternative methods have not been developed. Our simulation results as well as the real data application suggest that the proposed approach can often correctly identify the genomic features driving associations.

In this article, we propose a sparse KRV feature selection method to identify genomic features that are associated with microbiome composition. The kernel matrices for microbiome data are constructed to incorporate ecologically relevant information such as phylogenetic relationships among taxa. For genomic data, options of linear and Gaussian kernels are

provided. By providing not just a list of features, but also a weight associated with each feature, the sparse KRV method gives more insight into how each feature may be involved in the association. Simulations and real data analyses indicate that the proposed method is able to select features associated with microbiome composition while keeping false discovery rates reasonably low. Additionally, the selected list of features potentially increase the power of the KRV test and could be used to help inform further laboratory experiments and regression analyses.

Because feature-specific weights are added within the genomic kernel and selection is performed by penalizing the weights, it is easy to consider multiple choices of kernel for microbiome composition. In this article, we mainly focus on Bray-Curtis, weighted UniFrac, unweighted UniFrac, and generalized UniFrac kernels for microbiome data because they capture ecologically relevant features of microbiome compositions, including phylogeny, and are commonly used in practice. However, our method is not restricted to those kernels. For example, sparse KRV also works for feature selection when both datasets are normally distributed, in which case applying linear kernels on both datasets works well. Other, more complicated kernels could also be applied to correspond to particular data types, as long as they satisfy the positive semi-definite requirement. Furthermore, our method could be extended to a convex combination of kernels by assigning each kernel a scalar coefficient and defining the new kernel as a weighted sum of individual kernels: $\mathbf{K}_{new} = \alpha_u \mathbf{K}_u + \alpha_w \mathbf{K}_w + \alpha_{50} \mathbf{K}_{50}$. A penalty could be applied to select some subset of α's, producing an optimal kernel selection method (for the microbiome) combined with a feature selection method (for genomic features). This would provide robustness by avoiding the need to choose a particular microbiome kernel.

Turning to the genomic kernels, our algorithms accommodate either a linear kernel or a Gaussian kernel. Since the genomic kernel is closely related to the optimization procedure, it cannot be as flexible as the fixed kernels used for the microbiome. The linear kernel algorithm

is computationally faster than the Gaussian kernel, although when there are non-linear associations, the Gaussian kernel method is potentially more powerful. In real data analysis at a large scale, we recommend beginning with the linear kernel if researchers are primarily interested in subsequent statistical analysis due to the efficiency gains. If researchers are more interested in further laboratory examinations, we suggest using both algorithms and combining the feature sets for experiments, since the superset will increase the true discovery rate as much as possible, especially when the microbiome taxa are rare.

Sparse KRV is mainly intended to facilitate the selection of genomic features associated with microbiome composition. However, applications of sparse KRV can be extended to multiple inference methods. By optimizing the KRV test statistic, sparse KRV is maintains a natural relationship with the KRV test and provides a list of features that are more powerful for subsequent hypothesis testing. Features that are selected by sparse KRV are also prime candidates for community level exploratory or inferential analysis such as variance component analysis and the sequence kernel association test (SKAT) (Wu et al., 2011) or optimal and robust variants such as SKAT-O (Lee et al., 2012). Furthermore, the absolute value of feature specific weights can be used to construct a new variable as a linear combination of multiple omics features. Similar to a principal component, this variable could be used in place of the full feature set for further analysis. Finally, other penalties such as the fused lasso or graphical lasso could be used in place of the L_1 penalty to incorporate more information about the structure of the genomic features.

Chapter 3

KERNEL LINEAR DISCRIMINANT ANALYSIS[1]

3.1 Introduction

The low cost and ease of sample collection has spurred interest in omics data based prediction and classification models. Many of these studies also collect additional types of microbiome data (Zackular et al., 2014; Yu et al., 2017; Berry et al., 2015; Xiao et al., 2014; Fukuda and Fujita, 2014) that are co-informative with the omics in subsets of individuals, but these data are sometimes more difficult or expensive to obtain. The difficulties of integrating microbiome data with genomic data remain unanswered for classification problems, when we have both types of data for training with labels, but only omics data for prediction. In this paper, we develop classification models using multiple data types that can be applied to future data sets in which only genomics data are collected. Hence, we propose to introduce kernel structures into discriminant analysis.

Discriminant analysis is a classical statistical technique which is broadly used in classification problems. Linear discriminant analysis (LDA) is a simple case of discriminant analysis by assuming each class of data is normally distributed with the same covariance structure. LDA is commonly used to analyze omics data including microbiome compositional profiles (Pawlowsky-Glahn and Buccianti, 2011) after log-center transformation. Recently, many papers are focused on sparse version of discriminant analysis, such as sparse LDA (Wu et al., 2009; Clemmensen et al., 2011; Shao et al., 2011) and sparse quadratic discriminant analysis

[1]The contents of this chapter are based on the paper Kernel Linear Discriminant Analysis and Sparse Kernel Linear Discriminant Analysis

(QDA) (Cai and Zhang, 2019). To be noticed, there are existing literature introducing a kernel discriminant analysis which build discriminant function on a Hilbert space to be a general case (KDA or GDA) (Baudat and Anouar, 2000) and other transformation forms of discriminant analysis like modified discriminant analysis (MDA) (Xu et al., 2009). However, none of these approaches directly allow for incorporation of co-informative data to leverage the classification accuracy.

Meanwhile, microbiome and omics data like genomics data often encounter statistical challenges. The central one is high-dimensionality and nonlinear effects which may fail the traditional classification methods like logistic regression. Furthermore, it is not appropriate to treat different data types as equal because their hidden structures are inherited differently.

The motivation example is from a randomized clinical trials on women menopause symptoms and health (Mitchell et al., 2021) (MsFLASH study). DNA extraction and polymerase chain reaction (PCR) amplification and sequencing of the 16S rRNA gene were performed at enrollment of the study as well as at 12-weeks of visits. However, only at the beginning of the exam do we have both microbiome and genomic data without any missing. In the 12-week visit, the genomics data are the same as the first visit since the human genome is mostly stable, while the microbial community and composition can differ much along with time. For the 12-week visit, most samples did not take 16S rRNA sequencing. If we only take use of the genomics data for women vaginal pH analysis, it is a waste of resources and ablations, while if we only keep the samples with full microbiome records for both visits, then we end up with small part of samples left, which is under selection bias concerns and may violates data completeness assumption. We hope to answer the question in this and the following chapters: can we utilize microbiome data to help another data type to classify health conditions or biomarkers?

Facing those challenges, we propose a new method called kernel linear discriminant analysis

(Kernel LDA, or KLDA, which is not to be confused with KDA). Kernel LDA is a method that could replace the classical LDA's Gaussian prior by any pre-assumed positive definite prior by changing the distance definition. It is flexible in terms of sensitively capturing complicated covariance structures or hidden functiontional structures that may be available from complementary or auxiliary data and from other types of data, for example, microbiome data. This framework offers several advantages in that the model building incorporates additional information to potentially improve classification accuracy, but only omics data need to be available in future data. The proposed method provide a linear decision boundary in a transformed discriminant space, hence providing interpretive decision boundary and discriminant coordinates, while maintaining its flexibility on model training and testing procedures. Furthermore, kernel LDA, as compared with KDA, can be transformed into an equivalent form of a generalized least squares form plus a quadratic penalty, which avoids optimizing difficulties and non-convex problems potentially caused by KDA or GDA. We can also extend kernel LDA to more complicated discriminant analysis by introducing the penalty terms as high-dimensional kernel LDA so that it incorporates microbiome and omics data with both high-dimension variables. This extension will be introduced in the following chapter.

The chapter is organized as follows. In Section 3.2, we review the traditional LDA in both Raileigh quotient forms as well as Optimal scoring forms, then introduce our proposed kernel LDA method and its extension to high-dimensional data as penalized kernel LDA method. For kernel LDA, we present the approach in both Rayleigh quotient and in optimal scoring ways, and for penalized kernel LDA, the availability of approach is determined by the form of penalties. For L2 penalty, both forms of approach exist, while for L1 penalty and non-differential penalty, only optimal scoring are available. In Section 3.3, we present the results of simulation study, which show the classification accuracy of our proposed methods

under different situations. In Section 3.4, we discuss the limitation of the methods and provide insights of extensions.

3.2 Methods

3.2.1 Linear Discriminant Analysis

Linear discriminant analysis is a method for classification using a linear combination of features as the classifying boundary. LDA (Fisher, 1936; Friedman et al., 2001) has been shown to perform well and have good asymptotic property as the sample size tends to infinity in classical low-dimensional settings. Let $\mathbf{X}_{N\times p}$ be the matrix of N samples with p-dimensional variables, and each individual sample $\mathbf{x}_i$ uniquely belongs to one of classes with the number of total classes to be K. Here for future calculation simplicity, we assume matrix $\mathbf{X}$ is centered. $\mathbf{Y}_{N\times K}$ is a matrix of $0, 1$ where $Y_{i,k}$ means whether sample i is in class C_k $(k = 1, \ldots, K)$. For classical modeling, we assume that samples in k-th class C_k are independently distributed as normal distribution with fixed mean and variance as $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_w)$ where we call $\boldsymbol{\mu}_k$ as the mean of samples in class k and $\boldsymbol{\Sigma}_w$ as within-class covariance throughout all classes. LDA can be solved by finding discriminant coordinate vectors from optimizing a Rayleigh quotient (Parlett, 1974) form and applying the prior probability with Bayesian rule. It has been shown that Rayleigh Quotient enjoys a good mathematical property to serve as an loss function for optimizing problems (Fan et al., 2015). In order to achieve that, let's define a between-class covariance matrix as $\boldsymbol{\Sigma}_b = \frac{1}{N}\sum_{k=1}^{K} n_k \boldsymbol{\mu}_k \boldsymbol{\mu}_k^T$, where we estimate $\boldsymbol{\mu}_k$ by its empirical mean as $\boldsymbol{\mu}_k = \frac{1}{n_k}\sum_{i\in C_k} \mathbf{x}_i$. The empirical estimate of within-class covariance is then as $\boldsymbol{\Sigma}_w = \frac{1}{N}\sum_{k=1}^{K}\sum_{i\in C_k}(\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T$. To simplify the notation, we can rewrite the between-class and within-class covariance matrices into forms of matrix operations as $\boldsymbol{\Sigma}_b = \frac{1}{N}\mathbf{X}^T(\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T - \frac{1}{N}\mathbf{1}\mathbf{1}^T)\mathbf{X} = \frac{1}{N}\mathbf{X}^T(\mathbf{P}_\mathbf{Y} - \frac{1}{n}\mathbf{1}\mathbf{1}^T)\mathbf{X}$ and $\boldsymbol{\Sigma}_w = \frac{1}{N}\mathbf{X}^T(\mathbf{I}_n - \mathbf{P}_\mathbf{Y})\mathbf{X}$, where $\mathbf{P}_\mathbf{Y} = \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T$ is the projection matrix of $\mathbf{Y}$. It is obvious to see the summation of

within and between class covariance matrices $\mathbf{\Sigma}_b + \mathbf{\Sigma}_w = \frac{1}{N}\mathbf{X}^T(\mathbf{I}_n - \frac{1}{N}\mathbf{1}\mathbf{1}^T)\mathbf{X} = \frac{1}{N}\mathbf{X}^T\mathbf{X} = \mathbf{\Sigma}_{tot}$ is exactly the total covairance matrix $\mathbf{\Sigma}_{tot}$ since $\mathbf{X}$ is centered. For the between and covariance matrices summation, we borrow the term of total variance from ANOVA as the decomposition of the total variance when the design matrix is centered (St et al., 1989).

For multi-class linear discriminant analysis, we optimize the direction matrix as combination of discriminant coordinates $\mathbf{B} = \left(\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_L\right) \in \mathcal{R}^{p \times L}$ such that

$$\mathbf{B} = \underset{\mathcal{R}^{p\times L}}{\arg\max}\, tr(\mathbf{B}^T\mathbf{\Sigma}_b\mathbf{B}) \qquad \text{s.t. } \mathbf{B}^T\mathbf{\Sigma}_w\mathbf{B} = \mathbf{I}_L \tag{3.1}$$

When $N = 2$, the above LDA Equation 3.1 is degenerated to optimize the form of Rayleigh quotient

$$\boldsymbol{\beta} = \underset{\mathcal{R}^p}{\arg\max}\frac{\boldsymbol{\beta}^T\mathbf{\Sigma}_b\boldsymbol{\beta}}{\boldsymbol{\beta}\mathbf{\Sigma}_w\boldsymbol{\beta}} \tag{3.2}$$

For the above two classes problem, the only discriminant coordinate is the eigenvector corresponding to the largest eigenvalue of $\mathbf{\Sigma}_w^{-1}\mathbf{\Sigma}_b$.

Meanwhile, there is an equivalent method for linear discriminant analysis developed by (Hastie et al., 1994, 1995) through Optimal Scoring. Optimal scoring relates the discriminant functions with the average squared residuals of constrained linear regression by using a sequence of scores $\boldsymbol{\theta}$ such that we can directly optimize the loss function which is similar as a loss function of linear regression to get the discriminant coordinates with corresponding scores simultaneously. Using same notations as in previous text, the discriminant direction with optimal score for the i-th class is as follows,

$$(\boldsymbol{\beta}_i, \boldsymbol{\theta}_i) = \underset{\boldsymbol{\beta}\in\mathcal{R}^p, \boldsymbol{\theta}\in\mathcal{R}^K}{\arg\min}\frac{1}{N}\|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|^2 \qquad \text{s.t. } \frac{1}{N}\|\mathbf{Y}\boldsymbol{\theta}_i\|^2 = 1 \tag{3.3}$$

The optimal scoring provide an interesting view of expressing traditional formula of discriminant analysis into squared residual forms, providing the insight of introducing kernels into it.

3.2.2 Kernel LDA

Since classical LDA only provides linear decision boundaries in Euclidean spaces among classes, we propose a kernel LDA method which has linear decision boundaries in a Hilbert space induced by a kernel and can be transformed back into Euclidean spaces with non-linear boundaries, maintaining its original association and interoperability of discriminant coordinates. Similar to classical LDA, we present equivalent kernel LDA expressions in both Rayleigh quotient and Optimal scoring forms. In this section, we keep the notations the same with previous. For the design matrix $\mathbf{X}$, we refer to the set of data collected with low cost techniques and without missing throughout training, testing and future unlabeled data for classifying.

3.2.2.1 Kernel LDA by Rayleigh Quotient

Kernel LDA through Rayleigh quotient for two classes problem is defined as follows,

$$\boldsymbol{\beta} = \arg\max_{\mathcal{R}^p} \frac{\boldsymbol{\beta}^T \boldsymbol{\Sigma}_{\mathbf{K},b} \boldsymbol{\beta}}{\boldsymbol{\beta}^T \boldsymbol{\Sigma}_{\mathbf{K},w} \boldsymbol{\beta}} \tag{3.4}$$

The kernel between-class covariance matrix is $\boldsymbol{\Sigma}_{\mathbf{K},b} = \frac{1}{N}\mathbf{X}^T\mathbf{K}\mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{K}\mathbf{X}$, and within covariance matrix is $\boldsymbol{\Sigma}_{\mathbf{K},w} = \frac{1}{N}\mathbf{X}^T\mathbf{K}\mathbf{X} - \boldsymbol{\Sigma}_{\mathbf{K},b}$. The optimal $\boldsymbol{\beta}$ is solved by the eigenvector corresponding to the largest eigenvalue of $\boldsymbol{\Sigma}_{\mathbf{K},w}^{-1}\boldsymbol{\Sigma}_{\mathbf{K},b}$. For multi-class cases, we define the Kernel LDA as $\mathbf{B} = \arg\max_{\mathcal{R}^{p\times L}} tr(\mathbf{B}^T\boldsymbol{\Sigma}_{\mathbf{K},b}\mathbf{B})$ s.t. $\mathbf{B}^T\boldsymbol{\Sigma}_{\mathbf{K},w}\mathbf{B} = \mathbf{I}_L$.

Here, we introduce the kernel $\mathbf{K}$, which is an $N \times N$ matrix which transforms the common space expanded by original design matrix $\mathbf{X}$ into kernel expanded linear space. The kernel $\mathbf{K}$

is constructed by additional expensive or auxiliary data that are only collected for training, but not for future prediction as we stated in the introduction. We define the auxiliary data for training as a matrix $\mathbf{Z}_{N\times r}$, with sample size N same with training data, and number of parameters as r, which could be either larger or smaller than the sample size.

There are multiple choices of kernels fitting the settings. For genomic data, popular choices include commonly used positive semi-definite kernels such as the linear kernel ($\mathbf{K}_{ii'}(\mathbf{Z}) = \mathcal{K}_z(\mathbf{Z}_i, \mathbf{Z}_{i'}) = \sum_{j=1}^{p} z_{ij} z_{i'j}$), which is the default and most commonly used kernel for analyzing genetic variants. Other kernels include the generic Gaussian radial basis function (RBF) kernel ($\mathbf{K}_{ii'}(\mathbf{Z}) = \mathcal{K}_z(\mathbf{Z}_i, \mathbf{Z}_{i'}) = \exp\left\{-\sum_{j=1}^{p}(z_{ij} - z_{i'j})^2/\rho\right\}$), and the genetic identity-by-state kernel ($\mathbf{K}_{ii'}(\mathbf{Z}, \mathbf{c}) = \mathcal{K}_z(\mathbf{Z}_i, \mathbf{Z}_{i'};) = (2p)^{-1}\sum_{j=1}^{p}|z_{ij} - z_{i'j}|$). One advantage of those kernel is that they do not require any assumptions of dimensions or sparsity for genomic data, which often to be high-dimensional.

In the Rayleigh quotient form, we do not express the between-covarience matrix in general empirical forms, but actually in a space expanded by $\langle\cdot,\cdot\rangle_{\mathbf{K}}$, where $\mathbf{K}$ is a reproducing kernel. Given that we have a finite observations of kernel matrix $\mathbf{K}$, and we assume it is well constructed with positive semi-definite and symmetric property, we can write the singular value decomposition as $\mathbf{K} = \mathbf{U}^T\mathbf{S}^2\mathbf{U}$, where the decomposition has good property as $\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}$ and $\mathbf{S}$ is diaganal matrix. Hence, we can again write the between-covariance matrix as $\mathbf{B}_{\mathbf{K},b} = \mathbf{X}^T\mathbf{U}^T\mathbf{S}^2\mathbf{U}\mathbf{Y}(\mathbf{Y}^T\mathbf{U}^T\mathbf{U}\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{U}^T\mathbf{S}^2\mathbf{U}\mathbf{X} = \widetilde{\mathbf{X}}^T\mathbf{S}^2\widetilde{\mathbf{Y}}(\widetilde{\mathbf{Y}}^T\widetilde{\mathbf{Y}})^{-1}\widetilde{\mathbf{Y}}\mathbf{S}^2\widetilde{\mathbf{X}}$, with both $\widetilde{\mathbf{X}}$ and $\widetilde{\mathbf{Y}}$ be the transformed matrix of $\mathbf{X}$ and $\mathbf{Y}$. Then the kernel between-covariance matrix can be understood as we put different weights for each rows for transformed design matrix, and the weights are captured by the eigenvalues of $\mathbf{K}$. In this way, the empirical covariance matrices are able to integrate high-dimensional data as auxiliary information contributing as kernel structures, to provide better classification accuracy.

3.2.2.2 Kernel LDA by Optimal Scoring

Equivalent with the format in Rayleigh Quotient, the Kernel LDA through Optimal Scoring is defined as follows, for each $i = 1, \ldots, K$

$$(\boldsymbol{\beta}_{\mathbf{K},i}, \boldsymbol{\theta}_i) = \underset{\boldsymbol{\beta}\in\mathcal{R}^p, \boldsymbol{\theta}\in\mathcal{R}^K}{\arg\min} \frac{1}{N}\|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|^2_{\mathbf{K}} \qquad \text{s.t. } \frac{1}{N}\|\mathbf{Y}\boldsymbol{\theta}_i\|^2 = 1 \tag{3.5}$$

In Equation 3.5, we find the kernel discriminant coordinates for each class with a score by optimizing directly on summation of residuals in kernel space induced by $\mathbf{K}$.

By utilizing kernel framework for training data, we optimize the kernel discriminant coordinates $\boldsymbol{\beta}_{\mathbf{K},i}$ in a Hilbert space expanded by kernel $\mathbf{K}$ with its inner product $\langle\cdot,\cdot\rangle_{\mathbf{K}}$ such that the distance of residuals are minimized. It can be understood by assigning weights to samples. We are able to assign weights for samples when we transform the calculations back into Euclidean space. By choosing an appropriate kernel, we are able to utilize the additional data $\mathbf{Z}$ to increase the classification accuracy.

We develop the kernel LDA in optimal scoring because the Rayleigh quotient form is not directly compatible to incorporate some penalty terms, and it is not straightforward to implement in programming. In the following section, we show the algorithm of Kernel LDA by optimal scoring, and in Chapter 4 we will go a step further to incorporate penalty terms with Kernel LDA so that not only high-dimensional omics data but also high-diemsonal microbiome data can be integrated into one model.

3.2.3 Kernel LDA Algorithm

We present the algorithm set-up as follows. First, we randomly initialize $\boldsymbol{\theta}_*$ as a K-vector, and K be the number of classes that we will classify on, and let $Q = K - 1$ be the actual rank of label matrix. The label matrix $\mathbf{Y}$ is a $N \times K$ matrix of 0 and 1 with $\mathbf{Y}_{ij}$ indicates whether

sample i is in the class j if $\mathbf{Y}_{ij} = 1$ and not in the class with $\mathbf{Y}_{ij} = 0$. For any $i \in (1, \ldots, Q)$, set $\boldsymbol{\Theta}_i = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{i-1})$ be a $K \times (i-1)$ matrix of all calculated score $\boldsymbol{\theta}_i$ before each class i. Let $\mathbf{M}_\pi = \frac{1}{n}\mathbf{Y}^T\mathbf{Y}$ be a constant normalization for label prior probability. The first $\boldsymbol{\Theta}_i$ is assigned to identity as $\boldsymbol{\Theta}_1 = \mathbf{1}_K$. We obtain the kernel matrix by utilizing a properly chosen kernel function ϕ operating on data $\mathbf{Z}$ so that $\mathbf{K} = \langle \phi(\mathbf{Z})^T, \phi(\mathbf{Z}) \rangle$. Usually the kernel function satisfies the Riesz representation theorem (Rudin, 1973).

Then, for any $i \in (1, \ldots, Q)$, we perform the following algorithm in sequence:

1. Initialize $\boldsymbol{\theta}_i^{(0)} = (\mathbf{I}_K - \boldsymbol{\Theta}_i\boldsymbol{\Theta}_i^T\mathbf{M}_\pi)\boldsymbol{\theta}_*$. Normalize $\boldsymbol{\theta}_i^{(0)} \leftarrow \frac{\boldsymbol{\theta}_i^{(0)}}{\sqrt{\boldsymbol{\theta}_i^{(0)T}\mathbf{M}_\pi\boldsymbol{\theta}_i^{(0)}}}$ such that $\boldsymbol{\theta}_i^{(0)T}\mathbf{M}_\pi\boldsymbol{\theta}_i^{(0)} = 1$.

2. For iteration m, we update $\boldsymbol{\beta}_i^{(m)}$ and $\boldsymbol{\theta}_i^{(m)}$ until convergence criteria reached. There are steps within iteration as follows:

   (a) For a fixed $\boldsymbol{\theta}_i^{(m-1)}$, update $\widehat{\boldsymbol{\beta}}_i^{(m)} \leftarrow \arg\min \left\{ \frac{1}{N} \|\mathbf{Y}\boldsymbol{\theta}_i^{(m-1)} - \mathbf{X}\boldsymbol{\beta}\|_{\mathbf{K}}^2 \right\}$.

   (b) For a fixed $\boldsymbol{\beta}_i^{(m)}$, update $\boldsymbol{\theta}_i^{(m)} \leftarrow \arg\min \left\{ \frac{1}{N} \|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}_i^{(m)}\|_{\mathbf{K}}^2 \right\}$ such that $\boldsymbol{\theta}_i^{(m)T}\mathbf{M}_\pi\boldsymbol{\theta}_i^{(m)} = 1$ and $\boldsymbol{\theta}_i^{(m)T}\mathbf{M}_\pi\boldsymbol{\theta}_l^{(m)} = 0$ for all $i \neq l$.

3. The classification rule is to assign new data $(\mathbf{X}_{new}\boldsymbol{\beta}_1, \ldots, \mathbf{X}_{new}\boldsymbol{\beta}_Q)$ to the closest centroid of training data in $(\mathbf{X}\boldsymbol{\beta}_1, \ldots, \mathbf{X}\boldsymbol{\beta}_Q)$.

In details of step 2, we update $\boldsymbol{\beta}$ by $\widehat{\boldsymbol{\beta}}^{(m)} \leftarrow (\mathbf{X}^T\mathbf{K}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{K}\mathbf{Y}\boldsymbol{\theta}_i^{(m-1)}$ and update $\boldsymbol{\theta}$ by $\boldsymbol{\theta}_i^{(m)} \leftarrow (\mathbf{I}_K - \boldsymbol{\Theta}_i\boldsymbol{\Theta}_i^T\mathbf{M}_\pi)\mathbf{M}_\pi^{-1}\mathbf{Y}^T\mathbf{K}\mathbf{X}\boldsymbol{\beta}_i^{(m)}$, then normalize $\boldsymbol{\theta}_i^{(m)} \leftarrow \frac{\boldsymbol{\theta}_i^{(m)}}{\sqrt{\boldsymbol{\theta}_i^{(m)T}\mathbf{M}_\pi\boldsymbol{\theta}}}$. Here, we call the part of $\mathbf{M}_\pi^{-1}\mathbf{Y}^T\mathbf{K}\mathbf{X}\boldsymbol{\beta}_i^{(m)}$ as the original solver of $\boldsymbol{\theta}$, and the part of $(\mathbf{I}_K - \boldsymbol{\Theta}_i\boldsymbol{\Theta}_i^T\mathbf{M}_\pi)$ as the normalizing constant such that the projection condition satisfied. To be specific, the normalization works as the following illustration, with first part as normalize projection factor

and the second part as original solver of the optimising goal.

$$\boldsymbol{\theta}_i^{(m)} \leftarrow \underbrace{(\mathbf{I}_K - \boldsymbol{\Theta}_i \boldsymbol{\Theta}_i^T \mathbf{D}_\pi)}_{normalize/projection} \underbrace{(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{K}\mathbf{X}\boldsymbol{\beta}_i^{(m)}}_{\text{original solver of } \theta_y^{(m-1)}} \tag{3.6}$$

From Step 2, we have obtained the results of Kernel LDA discriminant coordinates. Recall that two-class LDA's decision boundary takes the form of $\widehat{\boldsymbol{\beta}}\mathbf{x} = \frac{1}{2}\widehat{\boldsymbol{\beta}}(\widetilde{\boldsymbol{\mu}}_1 + \widetilde{\boldsymbol{\mu}}_2) + const$, we can view step 3 as similar strategy to classify a new input of data. For Kernel LDA, the step 3 of algorithm is to classify a new sample $\mathbf{X}_{new}$, then find the inter product of the projected matrix with determinant coordinate $\boldsymbol{\beta}$ from $\boldsymbol{\beta}_1$ to $\boldsymbol{\beta}_Q$, performing binary classification on each exclude-one labels until we find finish classify new samples to Q classes.

In summary, the classification rule is to first find the prior of classes with decision boundary using training data $\widehat{\boldsymbol{\beta}}\mathbf{K}\mathbf{X}$ with decision boundary r, then for a new testing input we similarly projected it onto the same lines to compare with decision r, then classify new samples correspondingly.

3.2.4 Simulation strategy

In this section, we conducted simulations to evaluate the classification performance of our proposed methods and to compare them with existing methods across a range of simulation scenarios. For each scenario, we have both training data and test data. The training data has both data forms of metabolomics data and microbial auxiliary data for each sample. The testing data only has the metabolomics data, without any auxiliary data provided. We perform simulations in the setting that we have special types of auxiliary data to extrapolate the potential clustered structures of samples. To test the effect of our proposed methods, we proposed both data generating mechanisms in linear models and logistic models. In the main context, we present the simulation design and results under the data generation for normal

settings, and we include the results for logistic settings in the simulation section of the next chapter.

For training data we fix sample size N. We first create a labeling column for N samples to generate a simulation, and the labeling column has 2 distinct classes, with half (or around half for different choices of sample size setting) of them in the first class, and the rest of them in the second class. We assign each 4 samples into one cluster so in total there are $N/4$ clusters. The clusters can be viewed as the smallest units with correlated structures, such as repeated measurements of same subjects, or different clinical visit records of the same person and samples from the same cluster won't fall into different classes. So, the cluster contains the information related to classes, while not exactly as classes assigned. We generate a design matrix $\mathbf{X}$ from random normal distribution from $N(\theta, \mathbf{\Sigma})$. Because only a few metabolites are strongly associated with the outcome, we let $\theta_j = f(r * L)$ for $j = 1, 2, \ldots, m$, where m is the number of covariates associated with outcome and we could change it in different settings, and $f(\cdot)$ is a function to provide different simulation settings, for example identity function as $f(x) = x$ or squared function as $f(x) = x^2$. We consider $\mathbf{\Sigma} = \mathbf{I}_p$ for the simplest setting. And L is the true label for the classes, and for two class problems it is a n-vector with 0 and 1. r is a parameter to control the main effect for the simulation to test the power of the classification methods. The auxiliary data $\mathbf{Z}$ are simulated by $\mathbf{Z} = \mathbf{XB} + \mathbf{E}$ where $\mathbf{E}$ is err terms generated from Cholesky decomposition of a randomly created compound symmetric covariance matrix with within cluster the correlation as $\mathbf{\Sigma}_i = \sigma^2\mathbf{1} + \sigma_1^2\mathbf{I}$ with $\sigma^2 = 0.3$, $\sigma_1^2 = 0.2$. $\mathbf{B}$ is a $p \times N$ matrix generate from independent standard normal distribution to represent potential linear association between design and auxiliary data.

We set the main information matrix $\mathbf{X}$ with dimension of the covariates $p = 20$, where $m = 2$ of the covariates are simulated with a labeling column with standard random errors. The sample sizes N for training data are chosen from $(36, 100, 200)$ to test the adaptability of

our proposed methods, and the sample sizes for testing data are fixed as 500. This simulation setting is to test model performance under general low-dimension parameters setting with sparse design matrices.

Throughout the simulation study, three kernel methods are used. The first one is Linear kernel defined as $K(x_1, x_2) = x_1^T x_2$. The second is Gaussian kernel as $K(x_1, x_2) = \exp(-\rho^{-1}\|x_1 - x_2\|^2)$. with a hyper-parameter ρ to control the non-linearly of the kernel. The third one is kernel function based on Bray-Curtis dissimilarity, constructed by $\mathbf{K} = -\frac{1}{2}(\mathbf{I}_N - \frac{\mathbf{1}_N\mathbf{1}_N^T}{N})\mathbf{D}^2(\mathbf{I}_N - \frac{\mathbf{1}_N\mathbf{1}_N^T}{N})$, where $\mathbf{D}$ is a $N \times N$ distance matrix from Bray-Curtis dissimilarity. To handle different simulation scenario for moderate and high-dimensional cases, our proposed methods include kernel LDA, high-dimensional kernel LDA with L1 penalty and L2 penalty, and each of proposed methods are applied with different choices of kernels as stated before. There are some methods involve parameter selection, for example tuning parameters for SDA, logistic lasso and high-dimensional kernel LDA with L1 and L2 penalties, and another example such as Gaussian kernel hyper-parameters. All those parameters are chosen from leave-one-out cross-validation on training data.

For each simulation setup, we further set the parameter r in $(0, 0.1, 0.2, 0.5, 1)$ to test the model performance when the signal are ranging from no-signal ($r = 0$), weak signal ($r = 0.1, 0.2$) to moderate and strong signal ($r = 0.5, 1$). All simulations are performed from 1,000 replications.

The baseline comparison methods include linear discriminant analysis (LDA) (Cohen et al., 2014), sparse discriminant analysis (SDA) (Clemmensen et al., 2011), kernel discriminant analysis with different choices of kernels (Duong et al., 2007), and logistic lasso (Tibshirani, 1996).

3.3 Results

We summarize the simulation results with empirical classification accuracy in the Figure 3.1. For detailed numerical values, Table 3.1, 3.2, and 3.3 present the accuracy of testing data for simulation, for sample size chosen from $(36, 100, 200)$. Methods for comparisons include LDA, SDA, KDA with linear kernel, and KDA with Gaussian kernel. Among our proposed methods, if the accuracy is higher than highest baseline methods for comparison, we highlight the result in bold. To compare among kernels, the Gaussian kernel had the best performance over linear kernel and Bray-Curtis kernels, so in the results we only presented the KLDA with Gaussian kernels (the other kernel results are in the appendix). When the sample size is increased, the testing accuracy of KLDA is improved from 0.922 for 36 samples to 0.974 for 100 samples and 0.985 for 200 samples reaching almost similar performance with penalized KLDA. Since KLDA did not have tuning parameters, it is more efficient when sample size is large to reach an acceptable accuracy. For the Gaussian kernel, compared among sample sizes, there was higher accuracy associated with larger sample sizes. Since the accuracy was pretty saturated, close to 1, the trend was not obvious. Throughout the simulation study, our proposed methods had better performance than LDA, SDA and KDA, which are all methods for discriminant analysis.

To test the ability to distinguish signals, we set r in $(1, 0.5, 0.2, 0.1, 0)$ in decreasing order to see if our methods work with weak signals. We oberve that all the methods listed in the table have decreasing trend as the signal r decreasing, and when the signal is 0, the classification accuracy is around 0.5 i.e as similar to randomly generated results. When the signal is as strong as 1 and 0.5, the kernel LDA are uniformly better than the other baseline models. When the signal is smaller, as 0.2 and 0.1, our proposed methods don't stand out and actually all methods had about similar accuracy.

We further present simulation results for non-linear function $f = x^2$ as additional results,

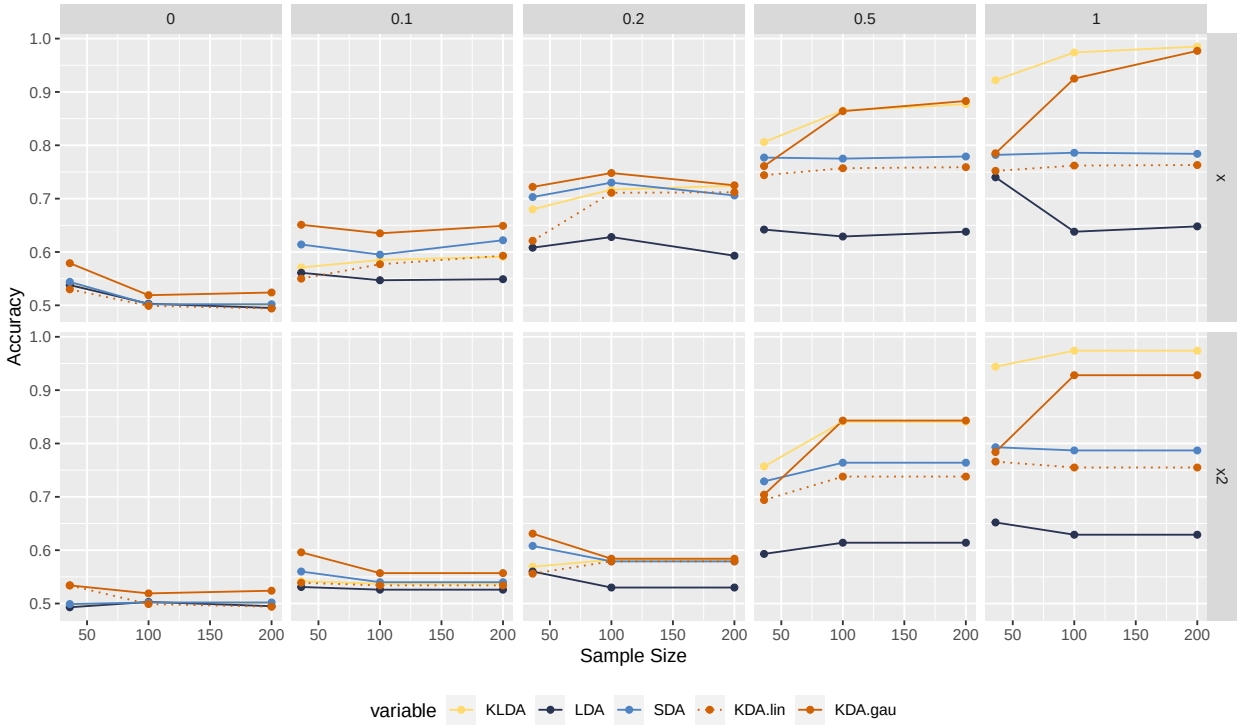

Figure 3.1: Simulation Results under Normal Data Generation Setting for Low Dimensional Cases

with different signals similar to linear function. The results are similar as in linear setting, that the KLDA and penalized KLDA had better results for strong signals.

3.4 Discussion

In this chapter, we propose kernel LDA as a method by integrating auxiliary data that exists for training but missing for testing to achieve higher discriminant analysis accuracy. The way to incorporate the partially existing data is to put them into the kernel machine so that it can help find better weighted discriminant coordinates. The proposed method can apply to high-dimensional data in the partially existing part. For the main data, if it is also high-dimensional, we proposed the penalized version of the kernel LDA to overcome the difficulties. Kernel LDA provides a flexible approach to model the microbial data, genomic data and metabolic data with traditional clinical records at the same time and nested structure of correlations.

While many kernel methods such as KDA has been used as an extension of linear discriminant analysis, this chapter specifically defines a discriminant analysis in a transformed space expanded by the kernels, providing not only an approach with higher accuracy to utilize auxiliary data, but also interpretive and meaningful discriminant coordinates. The methods can be applied to a broad range of classification problems and data types, with a variety of kernels as long as they are satisfied with reproducible properties.

We are focused on three popular kernel machines, the linear kernel and the Gaussian kernel and Bray-Curtis kernel. The linear kernel performs adequately good accuracy with less resources taken while iterating the algorithm to solve the optimization problem. The Gaussian kernel, on the other hand, provides broader acceptance of non-linear association between data types. Meanwhile, the kernel LDA method can be extended to more complicated kernel structures. For example, bacterial phylogeny is usually associated with presence and function,

Simulation Setting	KLDA-Gaussian	LDA	SDA	KDA-Linear	KDA-Gaussian
$f = x$ with r =1	**0.922**	0.740	0.782	0.785	0.785
$f = x$ with r =0.5	**0.806**	0.642	0.777	0.761	0.761
$f = x$ with r =0.2	0.680	0.608	0.703	0.722	0.722
$f = x$ with r =0.1	0.571	0.561	0.614	0.651	0.651
$f = x$ with r =0	0.498	0.493	0.499	0.534	0.534
$f = x^2$ with r =1	**0.944**	0.652	0.793	0.766	0.784
$f = x^2$ with r =0.5	**0.757**	0.593	0.729	0.694	0.704
$f = x^2$ with r =0.2	0.569	0.560	0.608	0.631	0.631
$f = x^2$ with r =0.1	0.542	0.531	0.560	0.596	0.596
$f = x^2$ with r =0	0.498	0.493	0.499	0.534	0.534

Table 3.1: Classification accuracy for simulation in normal setting with sample size 36 and evaluation sample size 500

Simulation Setting	KLDA-Gaussian	LDA	SDA	KDA-Linear	KDA-Gaussian
$f = x$ with r =1	**0.974**	0.638	0.786	0.762	0.925
$f = x$ with r =0.5	**0.865**	0.629	0.775	0.757	0.864
$f = x$ with r =0.2	0.717	0.628	0.730	0.748	0.748
$f = x$ with r =0.1	0.585	0.547	0.595	0.635	0.635
$f = x$ with r =0	0.500	0.501	0.500	0.517	0.517
$f = x^2$ with r =1	**0.974**	0.629	0.787	0.755	0.928
$f = x^2$ with r =0.5	0.841	0.614	0.764	0.738	0.843
$f = x^2$ with r =0.2	0.582	0.530	0.579	0.584	0.584
$f = x^2$ with r =0.1	0.537	0.526	0.540	0.557	0.557
$f = x^2$ with r =0	0.500	0.501	0.500	0.517	0.517

Table 3.2: Classification accuracy for simulation in normal setting with sample size 100 and evaluation sample size 500

Simulation Setting	KLDA	LDA	SDA	KDA-linear	KDA-Gaussian
$f = x$ with r =1	**0.985**	0.648	0.784	0.763	0.977
$f = x$ with r =0.5	**0.877**	0.638	0.779	0.759	0.883
$f = x$ with r =0.2	0.724	0.593	0.706	0.725	0.712
$f = x$ with r = 0.1	0.591	0.549	0.622	0.593	0.649
$f = x$ with r =0	0.501	0.495	0.502	0.494	0.524
$f = x^2$ with r =1	**0.985**	0.635	0.783	0.759	0.979
$f = x^2$ with r =0.5	0.842	0.596	0.741	0.718	0.848
$f = x^2$ with r =0.2	0.555	0.520	0.576	0.552	0.560
$f = x^2$ with r =0.1	0.522	0.516	0.540	0.518	0.546

Table 3.3: Classification accuracy for simulation in normal setting with sample size 200 and evaluation sample size 500

so leveraging phylogenetic information may provide higher power and accuracy for the kernel LDA method. Potential future work includes UniFrac distances to incorporate phylogenetic information, and a hybrid kernel as a linear combination of different kernel machines.

Chapter 4

PENALIZED KERNEL LINEAR DISCRIMINANT ANALYSIS

4.1 Introduction

Omics is a rapidly evolving field that encompasses genomics, epigenomics, transcriptomics, proteomics, and metabolomics. Researchers study omics data to receive better comprehensive knowledge on biological sciences. For humans, the omics is of tremendous interest to basic science researchers and clinicians alike in the pursuit of a deeper understanding of human health, especially at an extraordinarily detailed molecular level.

With the development of Omics technologies such as protein microarrays, Gel-based proteomics, mass spectrometry, and high-throughput cell assays, which are among commonly employed techniques in metabolomics, interactomics, genomics, and transcriptomics, researchers are able to collect large amounts of raw data as well as summaries in the form of lists of sequences, genes, proteins, metabolites, or SNPs. With big promise arising in this field, there are increasing demands in data processing, integration, analysis ,and interpretation of omics data.

In the last chapter, our proposed kernel LDA methods answer the question of how to integrate omices data with auxiliary microbiome data and provide interpretation of discriminant cocordinates for each omics input. However, the kernel LDA have more constraints on omics data compared to the microbiome data. One of the central challenges of analysing omics data using kernel LDA method, is the high dimensionality. The kernel linear discriminant analysis methods that can increase the classification accuracy by integrating the microbiome data which only exist in training data. This method does not require the microbiome data to

be low-dimension, since the microbial parts are transferred into the kernel spaces. However, it does have requirements as the classical linear discriminant analysis on the general design matrix, that the number of parameters (here omics covariates) to be lower than the sample sizes.

In this chapter, we extend our previous work of kernel linear discriminant analysis to a penalized kernel linear discriminant analysis (pKLDA or penalized KLDA) that can be applied to multiple types of omics data, with flexibility of choosing different penalty terms. Different with the previous methods for kernel LDA in last chapter, where we present the approach in both Rayleigh quotient and in optimal scoring ways, for penalized kernel LDA, the availability of approach is determined by the form of penalties. For L2 penalty, both forms of approach exist, while for L1 penalty and non-differential penalty, only optimal scoring are available.

The chapter is organized as follows. In Section 4.2, we introduce our proposed method, penalized KLDA, in both Rayleigh quotient forms as well as Optimal scoring forms, to accommodate different demands of omics data. We also present the algorithm for penalized kernel LDA accounting for different choices of penalty terms. In Section 4.3, we present the setting of simulation study and results in logistic settings and normal settings, showing the classification accuracy of our proposed methods under different situations. In Section 4.4, we apply our proposed methods to real data coming from randomized clinical trials. In Section 4.5, we discuss the limitation of the methods and potential extensions.

4.2 Methods

4.2.1 Penalized Kernel LDA

By introducing the kernel framework, we are able to process high-dimensional microbiome data using kernel techniques. On the other hand, some omics data that can be collected with

less cost, and they are also suffering from statistical difficulties caused by their high-dimension properties. Here, we propose a penalized kernel LDA method, which is an extension of our kernel LDA, but can control for high-dimensional omics data as the main input of the design matrix $\mathbf{X}$.

We first introduce the generalized version of penalized Kernel LDA by optimal scoring, which has flexibility to accommodate penalty terms as popular ones like L1 penalty, L2 penalty, elastic net, and any quadratic penalty with matrix $\mathbf{\Omega}$.

Let $\mathbf{X}_{N\times p}$ be the matrix of N samples with p-dimensional variables, and each individual sample $\mathbf{x}_i$ uniquely belongs to one of classes with the number of total classes to be K. Here for future calculation simplicity, we assume matrix $\mathbf{X}$ is centered. $\mathbf{Y}_{N\times K}$ is a matrix of $0, 1$ where $Y_{i,k}$ means whether sample i is in class C_k $(k = 1, \ldots, K)$. With some abuse of notations, the kernel $\mathbf{K}$ (to be distinguished with K, number of classes, is written in bold math symbol), which is an $N \times N$ matrix which transforms the common space expanded by original design matrix $\mathbf{X}$ into kernel expanded linear space. The kernel $\mathbf{K}$ is constructed by additional expensive or auxiliary data that are only collected for training, but not for future prediction as we stated in the introduction. We define the auxiliary data for training as a matrix $\mathbf{Z}_{N\times r}$, with sample size N same with training data, and number of parameters as r, which could be either larger or smaller than the sample size.

For each i in $(1, ..., K)$, the penalized kernel LDA is defined as,

$$(\boldsymbol{\beta}_{\mathbf{K},i}, \boldsymbol{\theta}_i) = \underset{\boldsymbol{\beta}\in\mathcal{R}^p, \boldsymbol{\theta}\in\mathcal{R}^K}{\arg\min} \frac{1}{N}\Big\{\|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|^2_{\mathbf{K}} + \text{penalty}\Big\} \tag{4.1}$$

$$\text{s.t. } \frac{1}{N}\|\mathbf{Y}\boldsymbol{\theta}_i\|^2 = 1 \tag{4.2}$$

The penalty terms can take the forms as (1) L1 penalty $\lambda\|\boldsymbol{\beta}\|_1$, (2) L2 penalty $\lambda\|\boldsymbol{\beta}\|_2^2$, (3) penalty induced by a positive semi-definite matrix $\mathbf{\Omega}$ $|\boldsymbol{\beta}\|^2_{\mathbf{\Omega}}$, and etc.

However, only with some specific type of penalty can we have a corresponding expression of penalized kernel LDA by Rayleigh quotient matched with optimal scoring. For example the if we take the form of (3) $\mathbf{\Omega}$-norm penalty, then the optimal scoring approach in equation 4.1 is equivalent to the following form as

$$\boldsymbol{\beta} = \arg\max_{\mathcal{R}^p} \frac{\boldsymbol{\beta}^T \mathbf{\Sigma}_{\mathbf{K},b} \boldsymbol{\beta}}{\boldsymbol{\beta}^T (\mathbf{\Sigma}_{\mathbf{K},w} + \mathbf{\Omega}) \boldsymbol{\beta}} \tag{4.3}$$

The choice (2) L2 penalty from the Rayleigh quotient is a special case of equation 4.3 with $\mathbf{\Omega} = \lambda \mathbf{I}$. The discriminant coordinates of those high-dimensional kernel LDA methods are not hard to get, since we have analytical solutions to the optimizing functions. We show the algorithm of Penalized Kernel LDA with $\mathbf{\Omega}$ penalty by optimal scoring approach in the appendix. Meanwhile, since loss function is not derivative with L1 penalty, nor exists an equivalent form of Rayleigh quotient, we can still find the discriminant coordinates from optimal scoring approach.

In equation 4.5, we define the penalized kernel LDA with L1 penalty as, for each i in $(1, \ldots, K)$, the kernel discriminant coordinates and its corresponding scores are taking the form from the optimal solution to the following equation, ponding scores are taking the form from the optimal solution to the following equation,

$$(\boldsymbol{\beta}_{\mathbf{K},i}, \boldsymbol{\theta}_i) = \arg\min_{\boldsymbol{\beta} \in \mathcal{R}^p, \boldsymbol{\theta} \in \mathcal{R}^K} \frac{1}{N} \Big\{ \|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|_{\mathbf{K}}^2 + \lambda \|\boldsymbol{\beta}\|_1 \Big\} \tag{4.4}$$

$$\text{s.t. } \frac{1}{N} \|\mathbf{Y}\boldsymbol{\theta}_i\|^2 = 1 \tag{4.5}$$

4.2.2 A general algorithm for penalized KLDA

In this section, we are focused on the algorithm development of penalized KLDA through optimal scoring.

The algorithm to find the discriminant coordinates with generalized penalty terms for kernel LDA is as follows. The initial parameter setting for penalized kernel LDA is similar with the kernel LDA. First, we randomly initialize $\boldsymbol{\theta}_*$ as a K-vector, and K be the number of classes that we will classify on, and let $Q = K - 1$ be the actual rank of label matrix. The label matrix $\mathbf{Y}$ is a $N \times K$ matrix of 0 and 1 with $\mathbf{Y}_{ij}$ indicates whether sample i is in the class j if $\mathbf{Y}_{ij} = 1$ and not in the class with $\mathbf{Y}_{ij} = 0$. For any $i \in (1, \ldots, Q)$, set $\boldsymbol{\Theta}_i = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{i-1})$ be a $K \times (i-1)$ matrix of all calculated score $\boldsymbol{\theta}_i$ before each class i. Let $\mathbf{M}_\pi = \frac{1}{n}\mathbf{Y}^T\mathbf{Y}$ be a constant normalization for label prior probability. The first $\boldsymbol{\Theta}_i$ is assigned to identity as $\boldsymbol{\Theta}_1 = \mathbf{1}_K$. We obtain the kernel matrix by utilizing a properly chosen kernel function ϕ operating on data $\mathbf{Z}$ so that $\mathbf{K} = \langle \phi(\mathbf{Z})^T, \phi(\mathbf{Z}) \rangle$. Usually the kernel function satisfies the Riesz representation theorem.

1. Initialize $\boldsymbol{\theta}_i^{(0)} = (\mathbf{I}_K - \boldsymbol{\Theta}_i \boldsymbol{\Theta}_i^T \mathbf{M}_\pi)\boldsymbol{\theta}_*$. Normalize $\boldsymbol{\theta}_i^{(0)} \leftarrow \frac{\boldsymbol{\theta}_i^{(0)}}{\sqrt{\boldsymbol{\theta}_i^{(0)T} \mathbf{M}_\pi \boldsymbol{\theta}_i^{(0)}}}$ such that $\boldsymbol{\theta}_i^{(0)T} \mathbf{M}_\pi \boldsymbol{\theta}_i^{(0)} = 1$.

   (a) For a fixed $\boldsymbol{\theta}_i^{(m-1)}$, update $\widehat{\boldsymbol{\beta}}_i^{(m)} \leftarrow \arg\min \left\{ \frac{1}{N} \|\mathbf{Y}\boldsymbol{\theta}_i^{(m-1)} - \mathbf{X}\boldsymbol{\beta}\|_{\mathbf{K}}^2 + \lambda \mathcal{P}(\boldsymbol{\beta}) \right\}$. This step could be performed by quadratic optimization through augmented Lagrangian method.

   (b) For a fixed $\boldsymbol{\beta}_i^{(m)}$, update $\boldsymbol{\theta}_i^{(m)} \leftarrow \arg\min \left\{ \frac{1}{N} \|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}_i^{(m)}\|_{\mathbf{K}}^2 \right\}$ such that $\boldsymbol{\theta}_i^{(m)T} \mathbf{M}_\pi \boldsymbol{\theta}_i^{(m)} = 1$ and $\boldsymbol{\theta}_i^{(m)T} \mathbf{M}_\pi \boldsymbol{\theta}_l^{(m)} = 0$ for all $i \neq l$.

2. The classification rule is to assign new data $(\mathbf{X}_{new}\boldsymbol{\beta}_1, \ldots, \mathbf{X}_{new}\boldsymbol{\beta}_Q)$ to the closest centroid of training data in $(\mathbf{X}\boldsymbol{\beta}_1, \ldots, \mathbf{X}\boldsymbol{\beta}_Q)$.

When the penalty term is L2 norms or any quadratic norms as $\boldsymbol{\Omega}(\boldsymbol{\beta})$, there are explicit solutions for the step 2.(a) in the above algorithm. However, when the L2 penalty is replaced by a non-differential penalty, for example L1 penalty, the step 2.(a) is not trivial. We introduce

the alternating direction method of multipliers algorithm (Boyd et al., 2011) with application to L1 penalty in the following section.

4.2.3 Penalized KLDA with L1 penalty

The algorithm to find the discriminant coordinates for L1-penalized kernel LDA is as follows, with the same initial parameters as the general algorithm for penalized KLDA.

1. Initialize $\boldsymbol{\theta}_i^{(0)} = (\mathbf{I}_K - \boldsymbol{\Theta}_i\boldsymbol{\Theta}_i^T\mathbf{M}_\pi)\boldsymbol{\theta}_*$. Normalize $\boldsymbol{\theta}_i^{(0)} \leftarrow \frac{\boldsymbol{\theta}_i^{(0)}}{\sqrt{\boldsymbol{\theta}_i^{(0)T}\mathbf{M}_\pi\boldsymbol{\theta}_i^{(0)}}}$ such that $\boldsymbol{\theta}_i^{(0)T}\mathbf{M}_\pi\boldsymbol{\theta}_i^{(0)} = 1$.

2. For iteration m, we update $\boldsymbol{\beta}_i^{(m)}$ and $\boldsymbol{\theta}_i^{(m)}$ until convergence criteria reached. There are steps within iteration as follows:

   (a) For a fixed $\boldsymbol{\theta}_i^{(m-1)}$, update $\widehat{\boldsymbol{\beta}}_i^{(m)} \leftarrow \arg\min\left\{\frac{1}{N}\|\mathbf{Y}\boldsymbol{\theta}_i^{(m-1)} - \mathbf{X}\boldsymbol{\beta}\|_{\mathbf{K}}^2 + \lambda\|\boldsymbol{\beta}\|_1\right\}$. This step could be performed by quadratic optimization through alternating direction method of multipliers methods.

      i. Define augmented Largrangian function $\mathcal{L} = \|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}\|_{\mathbf{K}}^2 + \lambda\|\mathbf{z}\| + s^T(\boldsymbol{\beta} - \mathbf{z}) + \frac{w}{2}(\boldsymbol{\beta} - \mathbf{z})^2$

      ii. Randomly initialize $\boldsymbol{\beta}$, $\mathbf{z}$ and initialize $s = 0$.

      iii. Update $\boldsymbol{\beta} \leftarrow (\mathbf{X}^T\mathbf{K}\mathbf{X} + w/2\mathbf{I})^{-1}(\mathbf{X}^T\mathbf{K}\mathbf{Y}\boldsymbol{\theta} + w/2\mathbf{z} - s)$. Then update $\mathbf{z} \leftarrow \mathcal{S}_{\lambda/w}(\boldsymbol{\beta} + s/w)$ where $\mathcal{S}$ is the soft-threshold function. Finally update $s \leftarrow s + w(\boldsymbol{\beta} - \mathbf{z})$

      iv. Repeat iii. until convergence.

   (b) For a fixed $\boldsymbol{\beta}_i^{(m)}$, update $\boldsymbol{\theta}_i^{(m)} \leftarrow \arg\min\left\{\frac{1}{N}\|\mathbf{Y}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\beta}_i^{(m)}\|_{\mathbf{K}}^2\right\}$ such that $\boldsymbol{\theta}_i^{(m)T}\mathbf{M}_\pi\boldsymbol{\theta}_i^{(m)} = 1$ and $\boldsymbol{\theta}_i^{(m)T}\mathbf{M}_\pi\boldsymbol{\theta}_l^{(m)} = 0$ for all $i \neq l$.

3. The classification rule is to assign new data $(\mathbf{X}_{new}\boldsymbol{\beta}_1, \ldots, \mathbf{X}_{new}\boldsymbol{\beta}_Q)$ to the closest centroid of training data in $(\mathbf{X}\boldsymbol{\beta}_1, \ldots, \mathbf{X}\boldsymbol{\beta}_Q)$.

4.2.4 Summary of Algorithms

To briefly compare Penalized KLDA through Rayleigh quotient, Penalized KLDA in optimal scoring, and Penalized KLDA with L1 penalty, we summarize the pros and cons for each method. (1) Penalized KLDA through Rayleigh quotient is computationally efficient, but only available for quadratic penalty. (2) Penalized KLDA in optimal scoring can be applied to a broader range of penalty functions, but takes more resources to iteratively find the local optimal of loss functions. (3) Penalized KLDA with L1 penalty is a special case of (2), with non-differentiable L1 penalty. It has one more layer of computational complexity compared with (2) with L2 penalty or quadratic penalty, and hence is more computationally expensive.

Different penalty terms are designed for different omics data. For example, L2 penalty is more appropriate for metabolic analysis when the data are moderately high-dimensional but not sparse. L1 penalty is more applicable to high-dimensional large profiling genomics data. With better understanding of omics data, more penalties or combined multiple penalty terms is interesting to be discovered.

4.3 Simulation Experiments

4.3.1 Simulation Setup

In this section, we carry out simulations in three scenarios to test the performance of penalized kernel linear discriminant analysis, with both L1 and L2 penalties available. We evaluate the classification performance of our proposed methods and to compare them with existing methods, including the kernel LDA if possible. The training data has both data forms of metabolomics data and microbial auxiliary data for each sample. The testing data only has

the metabolomics data, without any auxiliary data provided. We perform simulations in the setting that we have special types of auxiliary data to extrapolate the potential clustered structures of samples. To test the effect of our proposed methods, we proposed both data generating mechanisms in linear models and logistic models. The normal settings are similar to what we have proposed in the Chapter 3, with extension to the new methods. And in this section, we spend more space on logistic settings.

The baseline comparison methods include linear discriminant analysis (LDA) (Cohen et al., 2014), sparse discriminant analysis (SDA) (Clemmensen et al., 2011), kernel discriminant analysis with different choices of kernels (Duong et al., 2007), and logistic lasso (Tibshirani, 1996).

4.3.2 Simulation Strategy for Logistic Generated Data

For training data we fix sample size N. We first create a labeling column for N samples for generating simulation, and the labeling column has 2 distinct classes, with half (or around half for different choices of sample size setting) of them in the first class, and the rest of them in the second class. We assign each 4 samples into one cluster so in total there are $N/4$ clusters. The clusters can be viewed as the smallest units with correlated structures, such as repeated measurements of same subjects, or different clinical visit records of the same person. We first generate a linear model by $l = \mathbf{X}\mathbf{b}_1 + \mathbf{Z}\mathbf{b}_2$. The responses are simulated by random binomial distributions based on logistic transformation as $pr = l/(1 + \exp(l))$.

For the first simulation scenario, we set the main information matrix $\mathbf{X}$ with dimension of the covariates $p = 30$, where $r = 2$ of the covariates are simulated with a labeling column with standard random errors. We design the covariances and then use it to generate a matrix $\mathbf{Z}$. The covariance matrix $\mathbf{\Sigma}$ is generated as a block diagonal matrix with $N/4$ blocks, and each block is a compound symmetric matrix with $\mathbf{\Sigma}_i = \sigma^2\mathbf{1} + \sigma_1^2\mathbf{I}$ with $\sigma^2 = 0.3$, $\sigma_1^2 = 0.2$. The

auxiliary matrix $\mathbf{Z}$ is then simulated by $\mathbf{Z} = \mathbf{XB} + \mathbf{W}$ where $\mathbf{W}$ is generated from first $r = 30$ columns of Cholesky decomposition of covariance matrix, and $\mathbf{B}$ is a $p \times N$ matrix generate from independent standard normal distribution to represent potential linear association between design and auxiliary data. The sample sizes N are chosen from $(36, 100, 200)$ to test the adaptability of our proposed methods. This simulation setting is to test model performance under moderate high-dimension (for sample size 36) with sparse design matrices.

The second simulation scenario similar with the first simulation scenario, while the difference is that we increased the number of covariates and the number of informative covariates, set to be $p = 30$, $r = 10$ for 36 samples, $p = 80$, $r = 30$ for 100 samples and $p = 160$, $r = 30$ for 200 samples. This simulation setting is designed to test difference behavior under non-sparse settings, with changes of covariates sizes.

The third simulation scenario, we set the dimension of covariates p to be 1.25 times of the sample size N, which provides a high-dimension setting throughout different choices of sample sizes, while keeping the true related number of covariates to be fixed as 10. The third simulation setting is a true high-dimension scenario.

4.3.3 Simulation Strategy for Normally Generated Data

The data generating mechanism for normal data setting is the same with the simulation strategy in Chapter 3. For the first simulation scenario, we set the main information matrix $\mathbf{X}$ with dimension of the covariates $p = 20$, where $m = 2$ of the covariates are simulated with a labeling column with standard random errors. The sample sizes N for training data are chosen from $(36, 100, 200)$ to test the adaptability of our proposed methods, and the sample sizes for testing data are fixed as 500. In this simulation setting, both KLDA and penalized KLDA are available for a low dimension case.

We additionally design the second simulation scenario, we set the dimension of covariates

p to be 1.25 times of the sample size N, i.e. $p = 45$ for 36 samples, $p = 125$ for 100 samples and $p = 250$ for 200 samples. This scenario provides a true high-dimension setting throughout different choices of sample sizes, while keeping the true related number of covariates to be fixed as 2. At that time, we only test SDA and penalized KLDA performance since LDA and KLDA are no-longer applicable for high-dimensional design matrix.

4.3.4 Simulation Results for Logistic Settings

Table 4.1 presents the accuracy of testing data for simulation scenario one, for sample size chosen from $(36, 100, 200)$. Methods for comparisons include LDA, SDA, KDA with linear kernel, and KDA with Gaussian kernel. Among our proposed methods, if the accuracies are higher than the best performance of the baseline methods, we mark the result in bold to highlight them. For each kernel choice, we have Kernel LDA suitable for low dimensional cases, high-dim kernel LDA with L1 penalty and high-dim kernel LDA with L2 penalty. To compare among kernels, we include results for linear kernel, Gaussian kernel and kernel based on Bray-Curtis distance dissimilarity. Among three kernels, the Gaussian kernel methods have the best performance over linear kernels and Bray-Curtis kernels, especially when the sample size is 36. When the sample size increased, the testing accuracy of the linear kernel gets improved the most at 200 samples, reaching almost similar performance for Gaussian kernels. Since the linear kernel did not have hyper parameters, it is more efficient when sample size is large to reach an acceptable accuracy. But for smaller sample sizes, the Gaussian kernels are a lot more accurate with testing accuracy 0.966, 0.987 and 0.974 for three proposed methods. Bray-Curtis kernel had lower accuracy performance than linear kernels and Gaussian kernels, but still better than the other baseline methods. To compare among three proposed methods within the same kernel choices, for linear kernel, the high-dim kernel LDA had better results across sample sizes, but the performance did not differ much when the sample sizes were

100 and 200 given 1000 simulation replications. For Gaussian kernel, when sample size were 36, high-dim kernel LDA with L1 penalty had the highest accuracy with 0.987; when the sample sizes were 100 the kernel LDA had the best performance with 0.979 accuracy; and when the sample sizes were 200 the high-dim kernel LDA with L1 penalty won again with 0.984 accuracy. Compared among sample sizes, there was higher accuracy associated with larger sample sizes, though when the accuracy was near 1, the trend was not obvious. In summary, for low dimensional cases, penalized KLDA methods perform slightly better than kernel LDA, but not much. Gaussian kernel is also slighly better than linear and Bray-Curtis, while taking more computational resources to tun additional hyperparameter.

Table 4.2 shows the simulation results under the second scenario. In this setting, we test the model performance under moderate non-sparse settings in moderate high dimensional cases, with an increasing number of parameters with sample sizes. For baseline methods for comparisons, the KDA with Gaussian kernel has the highest testing accuracy for a variety of sample sizes. We use the same choices as in the first scenario to present our proposed methods. Among linear, Gaussian and Bray-Curtis kernel choices, kernel LDA with Gaussian kernel has higher accuracy in all sample sizes than the logistic lasso methods. When the sample size is 36, high-dim Kernel LDA with Gaussian kernel with either L1 and L2 kernels both have higher accuracy than baseline methods, and the one with L2 penalty model has the highest accuracy. When the sample size increased to 100 and 200, high-dim kernel LDA methods performance tended to be less accurate since in this setting we increased the true number of true parameters, however they are still better than LDA and SDA methods, and the model with L2 penalty had better performance than L1 model. Meanwhile, we notice that for kernel LDA with linear kernel, the performance under moderate high dimensional settings decreases much compared with simulation scenario 1, indicating that when the number of parameters is large, Gaussian or Bray-Curtis kernels are better choices than linear kernels.

Methods for Comparison / Sample Size	36	100	200
LDA	0.570	0.541	0.542
SDA	0.780	0.753	0.746
KDA linear kernel	0.564	0.537	0.541
KDA Gaussian kernel	0.706	0.855	0.852
Kernel LDA linear kernel	0.959	**0.968**	**0.974**
penalized kernel LDA Gaussian kernel with L1 penalty	0.959	**0.974**	**0.976**
penalized kernel LDA linear kernel with L2 penalty	0.962	**0.974**	**0.977**
Kernel LDA Gaussian kernel	**0.966**	**0.979**	**0.974**
penalized kernel LDA Gaussian kernel with L1 penalty	**0.987**	**0.968**	**0.978**
penalized kernel LDA Gaussian kernel with L2 penalty	**0.974**	**0.975**	**0.984**
Kernel LDA Bray-Curtis kernel	0.858	0.958	0.952
penalized kernel LDA Bray-Curtis kernel with L1 penalty	0.861	**0.964**	0.958
penalized kernel LDA Bray-Curtis kernel with L2 penalty	0.783	**0.964**	0.959

Table 4.1: Simulation Setting 1 with testing accuracy. Data generating mechanism is from logistic linear models.

Methods for Comparison / Sample Size	36	100	200
LDA	0.572	0.561	0.531
SDA	0.785	0.778	0.750
KDA linear kernel	0.567	0.580	0.528
KDA Gaussian kernel	0.807	0.790	0.892
Kernel LDA linear kernel	0.562	0.569	0.576
penalized kernel LDA linear kernel with L1 penalty	0.562	0.540	0.676
penalized kernel LDA linear kernel with L2 penalty	0.810	0.799	0.839
Kernel LDA Gaussian kernel	**0.963**	**0.959**	**0.940**
penalized kernel LDA Gaussian kernel with L1 penalty	**0.978**	0.879	0.839
penalized kernel LDA Gaussian kernel with L2 penalty	**0.985**	**0.942**	0.923
Kernel LDA Bray-Curtis kernel	0.861	0.878	0.823
penalized kernel LDA Bray-Curtis kernel with L1 penalty	0.871	0.778	0.744
penalized kernel LDA Bray-Curtis kernel with L2 penalty	0.780	0.799	0.742

Table 4.2: Simulation Setting 2 with testing accuracy. Data generating mechanism is from logistic linear models.

Methods for Comparison / Sample Size	36	100	200
SDA	0.769	0.763	0.753
KDA linear kernel	0.583	0.537	0.528
KDA Gaussian kernel	0.769	0.867	0.880
penalized kernel LDA linear kernel with L1 penalty	0.571	0.663	0.653
penalized kernel LDA linear kernel with L2 penalty	0.664	0.836	**0.854**
penalized kernel LDA Gaussian kernel with L1 penalty	**0.862**	**0.907**	**0.922**
penalized kernel LDA Gaussian kernel with L2 penalty	**0.885**	**0.923**	**0.939**
penalized kernel LDA Bray-Curtis kernel with L1 penalty	0.724	0.759	0.763
penalized kernel LDA Bray-Curtis kernel with L2 penalty	0.724	0.808	0.843

Table 4.3: Simulation Setting 3 with testing accuracy. Data generating mechanism is from logistic linear models.

The simulation results of scenario 3 were shown as in Table 4.3. In this setting, we applied the methods to a real high-dimensional case, where the number of parameters p was larger than sample size N, we excluded the comparisons for the methods only applicable for low dimensional data, such as LDA and kernel LDA. Among the baseline methods, when sample size is 36 and 100, logistic lasso had the best performance, while KDA with Gaussian kernel reached the highest accuracy when sample size was 200. The high-dimensional kernel LDA with Gaussian kernels had the highest accuracy among our proposed methods, and both methods with L1 and L2 penalties performed better than baseline. Meanwhile, the L2 penalties are consistently better than the L1 penalized version.

4.3.5 Simulation Results for Normal Generated Data

For low dimensional settings, we summarize the simulation results with empirical classification accuracy for normally simulated data in the Figure 4.1. For detailed numerical values, Table 4.4, 4.5, and 4.6 present the accuracy of testing data for simulation, for sample size chosen from $(36, 100, 200)$. When the sample size is increased, the testing accuracy of KLDA is improved from 0.922 for 36 samples to 0.974 for 100 samples and 0.985 for 200 samples reaching almost similar performance with penalized KLDA. Since KLDA did not have tuning parameters, it is more efficient when sample size is large to reach an acceptable accuracy. To compare among three proposed methods within the same kernel choices, the high-dim kernel LDA with L2 penalty have better results for smaller sample sizes under scenario 1, and the performance did not differ much when the sample sizes increased.

For a true high-dimensional setting, we summarize the results in Figure 4.2. Numerical results are in Table 4.7, 4.8, and 4.9 present the accuracy of testing data for simulation, for sample size chosen from $(36, 100, 200)$. Compared with SDA with large signal 1, the pLDA with Gaussian kernel and L2 penalty has the best performance in either linear and nonlinear

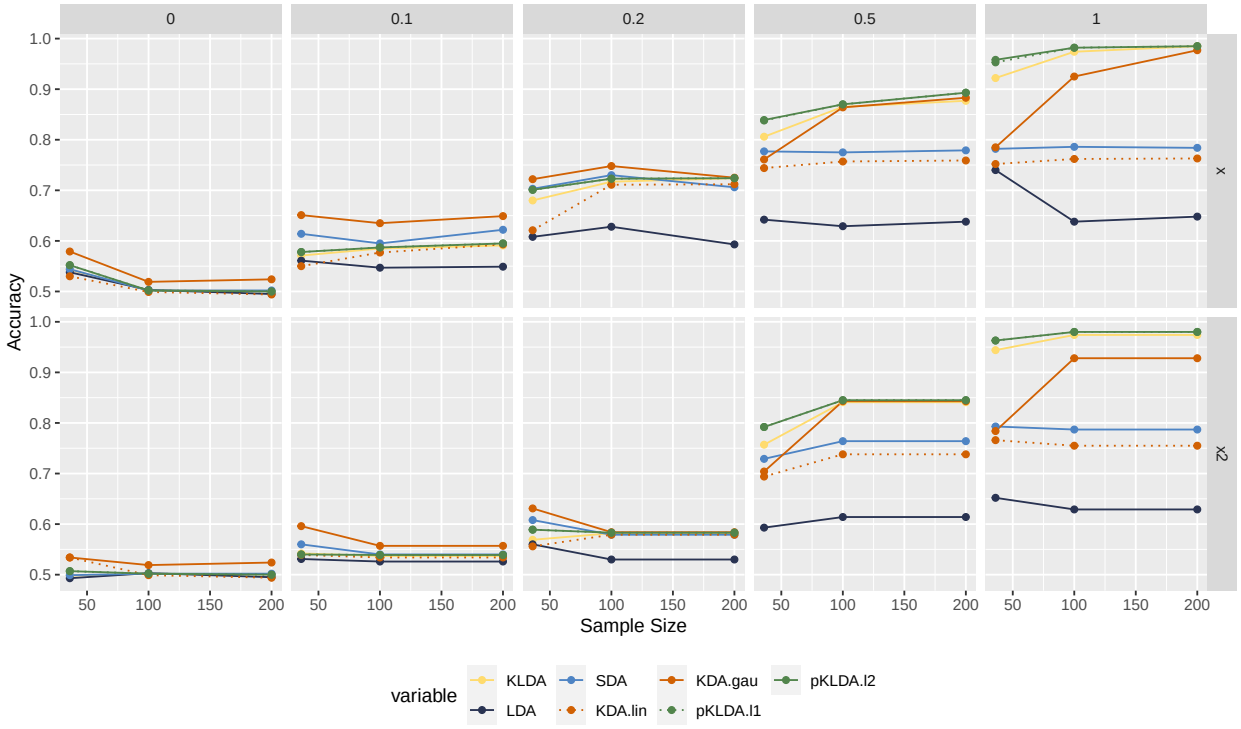

Figure 4.1: Simulation Results under Normal Data Generation Setting Scenario 1

Simulation Setting	KLDA	LDA	SDA	KDA.lin	KDA.gau	pKLDA.l1	pKLDA.l2
$f = x$ with r =1	0.922	0.740	0.782	0.785	0.785	**0.953**	**0.958**
$f = x$ with $r = 0.5$	0.806	0.642	0.777	0.761	0.761	**0.838**	**0.839**
$f = x$ with r =0.2	0.680	0.608	0.703	0.722	0.722	0.701	0.701
$f = x$ with r =0.1	0.571	0.561	0.614	0.651	0.651	0.578	0.578
$f = x$ with r =0	0.498	0.493	0.499	0.534	0.534	0.507	0.507
$f = x^2$ with r =1	0.944	0.652	0.793	0.766	0.784	**0.963**	**0.963**
$f = x^2$ with r =0.5	0.757	0.593	0.729	0.694	0.704	**0.792**	**0.792**
$f = x^2$ with $r = 0.2$	0.569	0.560	0.608	0.631	0.631	0.589	0.589
$f = x^2$ with r =0.1	0.542	0.531	0.560	0.596	0.596	0.540	0.540

Table 4.4: Classification accuracy for simulation scenario 1 in normal setting with sample size 36 and evaluation sample size 500, with penalized kerenel LDA added

Simulation Setting	KLDA	LDA	SDA	KDA.lin	KDA.gau	pKLDA.l1	pKLDA.l2
$f = x$ with r =1	**0.974**	0.638	0.786	0.762	0.925	**0.982**	**0.982**
$f = x$ with r =0.5	0.865	0.629	0.775	0.757	0.864	**0.870**	**0.870**
$f = x$ with r =0.2	0.717	0.628	0.730	0.711	0.748	0.723	0.723
$f = x$ with r =0.1	0.585	0.547	0.595	0.577	0.635	0.587	0.587
$f = x$ with r =0	0.502	0.503	0.502	0.499	0.519	0.502	0.502
$f = x^2$ with r =1	**0.974**	0.629	0.787	0.755	0.928	**0.980**	**0.980**
$f = x^2$ with r =0.5	0.841	0.614	0.764	0.738	0.843	0.845	0.845
$f = x^2$ with r =0.2	0.582	0.530	0.579	0.579	0.584	0.583	0.583
$f = x^2$ with r =0.1	0.537	0.526	0.540	0.534	0.557	0.539	0.539

Table 4.5: Classification accuracy for simulation scenario 1 in normal setting with sample size 100 and evaluation sample size 500, with penalized kerenel LDA added

Simulation Setting	KLDA	LDA	SDA	KDA.lin	KDA.gau	pKLDA.l1	pKLDA.l2
$f = x$ with r =1	**0.985**	0.648	0.784	0.763	0.977	**0.985**	**0.985**
$f = x$ with r =0.5	**0.877**	0.638	0.779	0.759	0.883	**0.893**	**0.893**
$f = x$ with r =0.2	0.724	0.593	0.706	0.725	0.712	0.724	0.724
$f = x$ with r = 0.1	0.591	0.549	0.622	0.593	0.649	0.595	0.595
$f = x$ with r =0	0.501	0.495	0.502	0.494	0.524	0.500	0.500
$f = x^2$ with r =1	**0.985**	0.635	0.783	0.759	0.979	**0.987**	**0.987**
$f = x^2$ with r =0.5	0.842	0.596	0.741	0.718	0.848	**0.854**	**0.854**
$f = x^2$ with r =0.2	0.555	0.520	0.576	0.552	0.560	0.560	0.560
$f = x^2$ with r =0.1	0.522	0.516	0.540	0.518	0.546	0.523	0.523

Table 4.6: Classification accuracy for simulation scenario 1 in normal setting with sample size 200 and evaluation sample size 500, with penalized kerenel LDA added

setup. However, when the signal decreases, as low as between 0 and 0.5, the penalized KLDA methods work similarly with SDA, with slightly better performance at signal 0.5 and almost same accuracy for weaker signals.

4.3.6 Summary of Simulation Results

At the end of the section, we would like to provide suggestions to choose among kernel LDA and kernels. When the omics data are low dimensional, we suggest applying kernel LDA with linear kernel, which achieves relatively good accuracy with low cost of computation resources. When the data are moderately high-dimensional, the KLDA with linear kernel no longer retains its good performance, we instead recommend KLDA with Gaussian kernel, with or without penalties. When the number of parameters is larger than the number of samples, we then recommend the penalized KLDA with properly chosen penalty terms given the consideration of omics data. When the microbial auxiliary data are with specific types such as relative abundance, we recommend adding Bray-Curtis kernel in consideration.

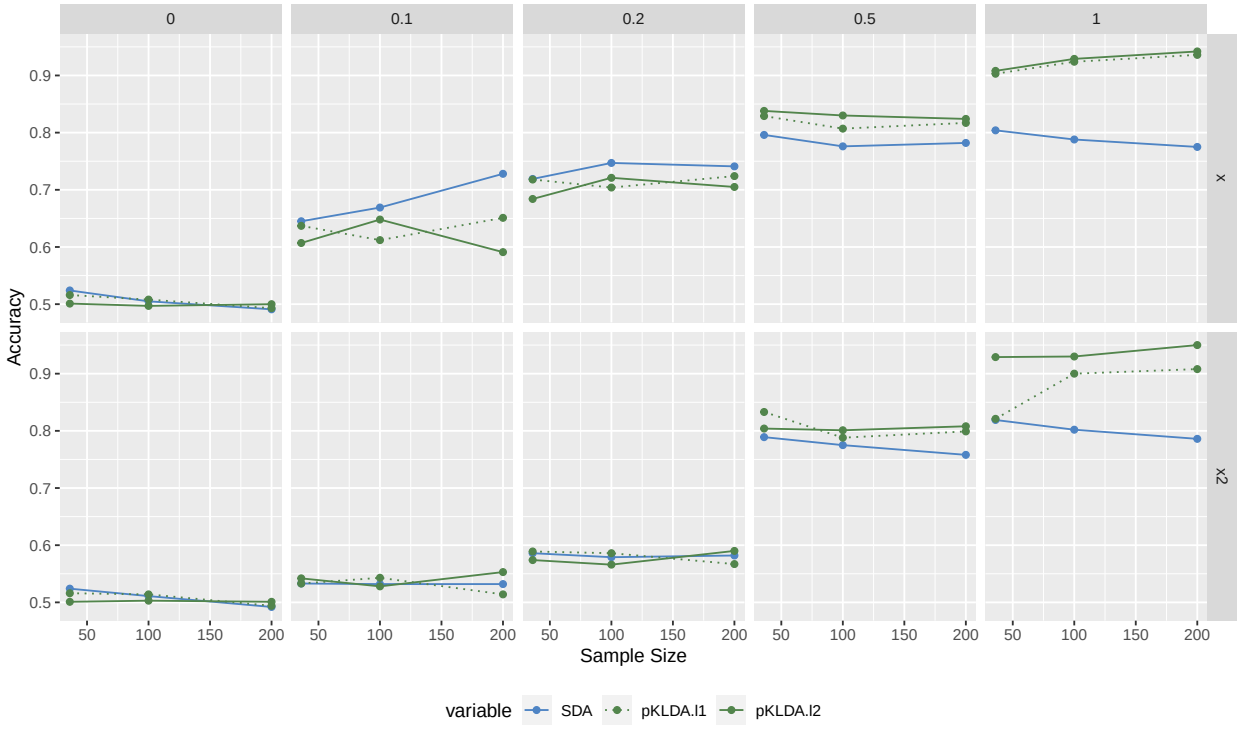

Figure 4.2: Simulation Results under Normal Data Generation Setting Scenario 2

Simulation Setting	SDA	pKLDA.l1	pKLDA.l2
$f = x$ with r =1	0.804	0.903	0.908
$f = x$ with r =0.5	0.796	0.829	0.838
$f = x$ with r =0.2	0.719	0.718	0.684
$f = x$ with r =0.1	0.645	0.637	0.607
$f = x$ with r =0	0.524	0.516	0.501
$f = x$ with r =1	0.819	0.821	0.929
$f = x$ with r =0.5	0.789	0.833	0.804
$f = x$ with r =0.2	0.586	0.589	0.574
$f = x$ with r =0.1	0.533	0.533	0.542

Table 4.7: Classification accuracy for simulation scenario 2 in normal setting with sample size 36 and evaluation sample size 500

Simulation Setting	SDA	pKLDA.l1	pKLDA.l2
$f = x$ with r =1	0.788	0.924	0.929
$f = x$ with r =0.5	0.776	0.807	0.830
$f = x$ with r =0.2	0.747	0.704	0.721
$f = x$ with r =0.1	0.669	0.612	0.648
$f = x$ with r =0	0.505	0.508	0.497
$f = x$ with r =1	0.802	0.900	0.930
$f = x$ with r =0.5	0.775	0.788	0.801
$f = x$ with r =0.2	0.579	0.586	0.566
$f = x$ with r =0.1	0.532	0.543	0.528

Table 4.8: Classification accuracy for simulation scenario 2 in normal setting with sample size 100 and evaluation sample size 500

Simulation Setting	SDA	pKLDA.l1	pKLDA.l2
$f = x$ with r =1	0.775	0.936	0.942
$f = x$ with r =0.5	0.782	0.817	0.824
$f = x$ with r =0.2	0.741	0.724	0.705
$f = x$ with r =0.1	0.728	0.651	0.591
$f = x$ with r =0	0.491	0.493	0.500
$f = x$ with r =1	0.786	0.908	0.950
$f = x$ with r =0.5	0.758	0.799	0.808
$f = x$ with r =0.2	0.582	0.567	0.590
$f = x$ with r =0.1	0.532	0.514	0.553

Table 4.9: Classification accuracy for simulation scenario 2 in normal setting with sample size 200 and evaluation sample size 500

4.4 Real Data Analysis

Our dataset comes from MsFLASH (Menopause Strategies: Finding Lasting Answers for Symptoms and Health) Trial (Mitchell et al., 2021), which is a randomized clinical trial for women post-menopausal with vaginal discomfort between June 2016 and APril 2017 while. DNA extraction and polymerase chain reaction (PCR) amplification and sequencing of the 16S rRNA gene were performed at enrollment of the study as well as at 4th and 12th weeks of visits. We assume microbiome data are expensive to collect and misses for one sample each time, then we take use of the metabolites as the core design matrices for the model input. After removing taxa presenting less than 10% of samples, disparsifying taxon and log-ratio transforming the data, we successfully collected 43 samples with 199 metabolic covariates and 43 samples with 380 bacteria taxon and at baseline. Samples are classified based on the pH value, whether smaller, equal or larger than 7, based on the records at week 12. For model selection, we use five folds of cross-validation to tune hyper parameters. We also use another fold of leave-one-out cross validation to calculate the testing accuracy.

We first explore the data by dividing samples into three classes based on their pH acidity, neutrality or basicity. To visualize the effect penalized LDA method, we draw two by two discriminant plots with classes different colored. As shown in Figure 4.3, the samples with pH$<$ 7 (colored in red) and pH$>$ 7 (colored in green) are almost separated in the discriminant plot of SDA, but the samples with pH$=$ 7 mix the other two classes such that SDA fails to classfy all three classes. In Figure 4.4, three classes are better separated in the transformed discriminant spaces of high-dim kernel LDA with Gaussian kernel and L2 penalty. Though some samples of acidity cannot be fully distinguished from the others, we observe a huge improvement from SDA to Kernel LDA.

We compare our proposed kernel LDA in high-dimensional cases with linear, Gaussian, and Bray-Curtis kernels with L1 and L2 penalty choices with baseline methods including

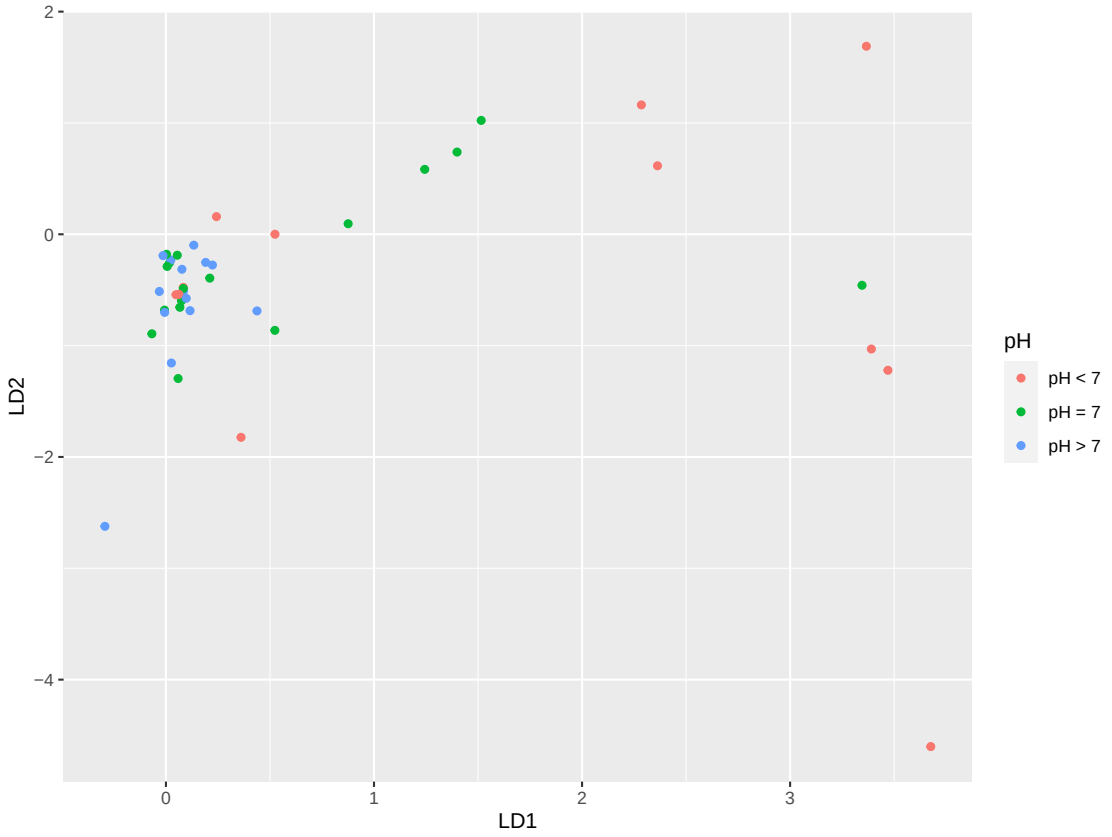

Figure 4.3: Discriminant Plot for Sparse Discriminant Analysis

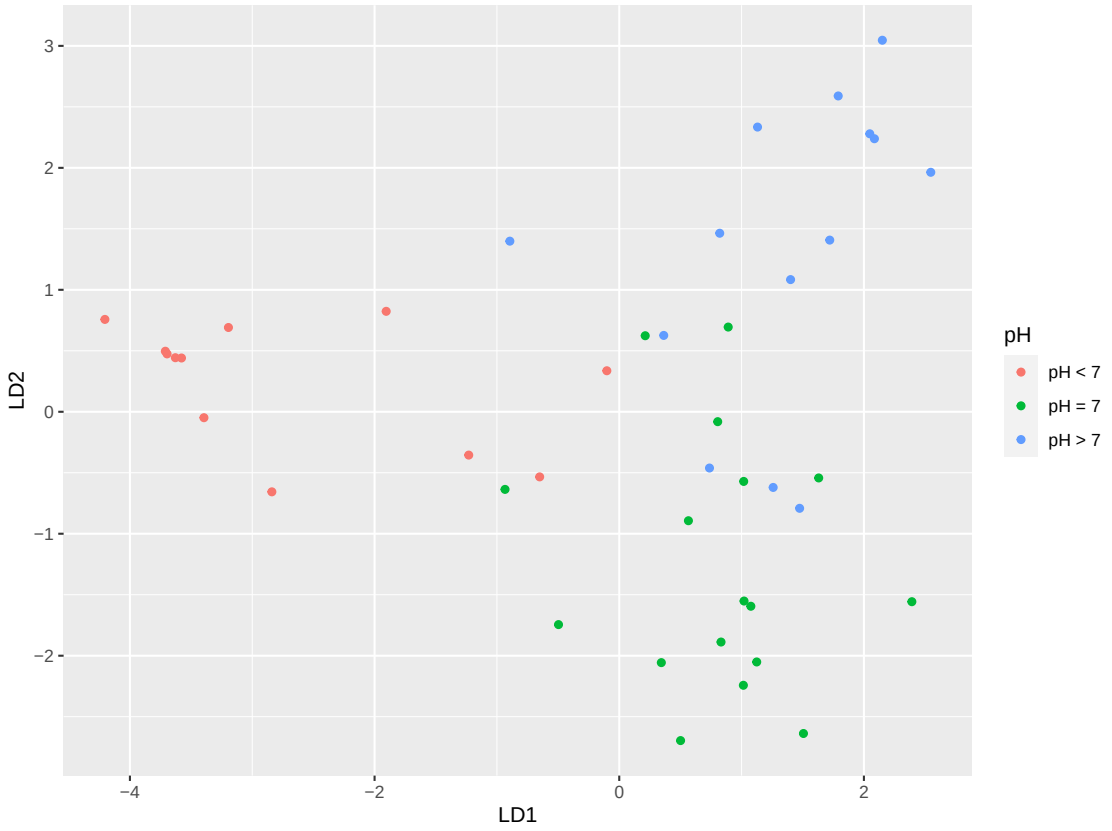

Figure 4.4: Discriminant Plot for Penalized Kernel LDA Gaussian Kernel with L2 Penalty

SDA, KDA with linear and Gaussian kernel and logistic lasso. Table 4.10 presents the results of testing accuracy for all methods of comparisons. Among the baseline methods, KDA with Gaussian kernel

Methods for Comparison	Testing Accuracy
SDA	0.581
KDA linear kernel	0.628
KDA Gaussian kernel	0.744
penalized kernel LDA linear kernel with L1 penalty	0.837
penalized kernel LDA linear kernel with L2 penalty	0.860
penalized kernel LDA Gaussian kernel with L1 penalty	0.860
penalized kernel LDA Gaussian kernel with L2 penalty	**0.884**
penalized kernel LDA Bray-Curtis kernel with L1 penalty	0.720
penalized kernel LDA Bray-Curtis kernel with L2 penalty	0.790

Table 4.10: Real Data Analysis, using metabolite data with auxiliary microbial data to categorize samples by class 1 (pH < 7), class 2 (pH > 7), and class 3 (pH $= 7$), with leave-one-out cross validation. The model with best performance is highlighted in bold.

Testing accuracy are shown in the Table 4.10. Among the baseline methods, KDA with Gaussian kernel has the highest testing accuracy as 0.744 (32/43). Among our proposed methods, high-dim kernel LDA with Gaussian kernel and L2 penalty, High-dim kernel LDA Gaussian kernel with L2 penalty has highest testing accuracy as 0.884 (38/43). Our proposed methods have better classification accuracy than baseline methods.

4.5 Discussion

In this chapter, we propose penalized kernel LDA as a method by integrating auxiliary data that exists for training but missing for testing to achieve higher discriminant analysis accuracy, that can be extended to high-dimensional omics data. The way to incorporate the partially existing data is to put them into the kernel machine so that it can help find better weighted discriminant coordinates. Penalized Kernel LDA provides a flexible approach to model the omics data with co-informative microbial data with traditional clinical records at the same time considering nested structure of correlations.

By fitting the sparse kernel linear discriminant analysis model using metabolic data as main input while integrating the microbiome data as co-informative auxiliary data with kernel machines, penalized kernel LDA model provides flexibility in the face of unknown microbiome data in the evaluation part but still achieve higher accuracy compared with existing methods.

Simulations show that, when the data are low dimensions or moderate high-dimensions, penalized KLDA has classification accuracy similar to or slightly higher than the KLDA. When the data are high-dimensional, it provides an achievable solution to the classification problem by providing multiple choices of penalties.

Facing the real data in high-dimensional settings, penalized KLDA with L2 penalty and Gaussian kernel performs the best to classify samples by predicting their pH values, and it even out performs the logistic lasso, which is a common choice of high-dimension classification problem that we seen similar performance in the simulation settings.

The choices among kernels for the microbiome data among linear, Gaussian and Bray-Curtis kernels are all available. Our suggestion is to first apply the methods with linear kernel and Bray-Curtis because they do not have additional hyper parameters as with Gaussian kernel, such that the former two kernels take less computation effort to get the results. When there are potential non-linear effects or functional structures involved in the microbiome data,

we would recommend a Gaussian kernel.

Similarly when we face the choices between L1 and L2 penalties, we recommend the L2 penalty as the first solution since it is more interpretative for omics data without induce sparsity, as well as would take less computing resources. Recommendations of choosing proper kernels are also included in this dissertation. If the dimension of omics data are highly sparse and highly depersonalized, we recommend L1 penalty. And if more complicated data structures show up, for example the omics data comes with tree based structures, we can further apply a fused lasso as the penalty to better incorporate the statistical characteristic of the data.

Chapter 5

DISCUSSIONS AND FUTURE WORK

The investigation of the microbiome-omics data integration has become a popular topic of scientific and clinical investigation into health, disease, and treatment strategies. With the development of microbiome methods, there is a growing demand to accommodate modern study designs from a biostatician perspective to incorporate structural information at the early stage of clinical or biomedical studies. Identification of omics covariates that are associated with whole microbial taxa communities and higher classification or prediction accuracy can be achieved based on the better understanding of microbiome data. The methods proposed in this dissertation can add valuable tools to a microbiologist's or bioinformatician's toolbox in both of these areas. They also point towards new avenues for further statistical methods development.

One potential future work is to incorporate more phylogenetic information in the kernel LDA models. In the analysis of MsFlash Data, the microbiome data are only collected with microbial taxa abundance, without phylogenetic structures, and we are able to compare the KLDA with Linear, Gaussian and Bray-Curtis kernels with existing methods. As an extension, it will be valuable if there is data available with coefficients assigned to branches of a phylogenetic tree. The existing phylogenetic trees will allow us to apply the kernel machines that account for phylogenetic structures, for example, kernels based on UniFrac distance of similarities.

Another important area of further development is to apply the KVV methods into genetic data, such that we are able to use microbiome data to facilitate gene-based analysis of GWAS.

Although environmental influences dominate the gene expressions, there is considerable evidence that microbial diversity as well as individual microbes are associated with human genetics. Since there is no direct, easy mapping from SNPs with microbes, it is a future direction to apply the sparse KRV method to harness microbiome data to increase power to identify genetic variants associated with health markers. Genetics data are sharing some joint properties with omics data that can potentially benefit from the KRV methods. We hope that the data integration will be conducted in a first stem, in which the samples with both genomic and microbiome data are defined. For each gene along the genome, we apply the sparse KRV method and find the weights vector for variants of the gene. We can further use the linear combination of weights and variants to be a new variable which can be assessed for association with a complex trait in a new data set.

Meanwhile, throughout this dissertation, we do not examine the direction of causality among microbiome, omics and disease. Although the DNA sequencing of humans rarely changes along with time, it is challenging to answer the question if there are potential associations between diseases and gene expression or metabolisms mediated by the microbial composition, or vice versa. Recently, there are scientists developing new methods to analyze microbiome data using graphical models and with causal justifications, and we hope our methods can be extended with those approaches. In methodological aspect, both KRV and KLDA will be full-filled with deep understanding in a causal relationship, and in practice, the penalized KLDA can be extended with other penalties such as graphical lasso such that the methods can be applied to more types of situations.

BIBLIOGRAPHY

Genevera I Allen. Automatic feature selection via weighted kernels and regularization. *Journal of Computational and Graphical Statistics*, 22(2):284–299, 2013.

Marti J Anderson. A new method for non-parametric multivariate analysis of variance. *Austral ecology*, 26(1):32–46, 2001.

Gaston Baudat and Fatiha Anouar. Generalized discriminant analysis using a kernel approach. *Neural computation*, 12(10):2385–2404, 2000.

David Berry, Orest Kuzyk, Isabella Rauch, Susanne Heider, Clarissa Schwab, Eva Hainzl, Thomas Decker, Mathias Müller, Birgit Strobl, Christa Schleper, et al. Intestinal microbiota signatures associated with inflammation history in mice experiencing recurring colitis. *Frontiers in microbiology*, 6:1408, 2015.

Howard D Bondell, Arun Krishna, and Sujit K Ghosh. Joint variable selection for fixed and random effects in linear mixed-effects models. *Biometrics*, 66(4):1069–1077, 2010.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

Richard H Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on scientific computing*, 16(5): 1190–1208, 1995.

T Tony Cai and Linjun Zhang. A convex optimization approach to high-dimensional sparse quadratic discriminant analysis. *arXiv preprint arXiv:1912.02872*, 2019.

Emily S Charlson, Jun Chen, Rebecca Custers-Allen, Kyle Bittinger, Hongzhe Li, Rohini Sinha, Jennifer Hwang, Frederic D Bushman, and Ronald G Collman. Disordered microbial communities in the upper respiratory tract of cigarette smokers. *PloS one*, 5(12): e15216, 2010.

Jun Chen and Hongzhe Li. Kernel methods for regression analysis of microbiome compositional data. In *Topics in Applied Statistics*, pages 191–201. Springer, 2013.

Jun Chen, Kyle Bittinger, Emily S Charlson, Christian Hoffmann, James Lewis, Gary D Wu, Ronald G Collman, Frederic D Bushman, and Hongzhe Li. Associating microbiome composition with environmental covariates using generalized unifrac distances. *Bioinformatics*, 28(16):2106–2113, 2012.

Line Clemmensen, Trevor Hastie, Daniela Witten, and Bjarne Ersbøll. Sparse discriminant analysis. *Technometrics*, 53(4):406–413, 2011.

Patricia Cohen, Stephen G West, and Leona S Aiken. *Applied multiple regression/correlation analysis for the behavioral sciences*. Psychology press, 2014.

Rene Cortese, Lei Lu, Yueyue Yu, Douglas Ruden, and Erika C Claud. Epigenome-microbiome crosstalk: a potential new paradigm influencing neonatal susceptibility to disease. *Epigenetics*, 11(3):205–215, 2016.

Nello Cristianini and John Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, 2000.

Nello Cristianini, John Shawe-Taylor, Andre Elisseeff, and Jaz S Kandola. On kernel-target alignment. In *Advances in neural information processing systems*, pages 367–373, 2002.

Tarn Duong et al. ks: Kernel density estimation and kernel discriminant analysis for multivariate data in r. *Journal of Statistical Software*, 21(7):1–16, 2007.

Yves Escoufier. Le traitement des variables vectorielles. *Biometrics*, pages 751–760, 1973.

Jianqing Fan, Zheng Tracy Ke, Han Liu, and Lucy Xia. Quadro: A supervised dimension reduction method via rayleigh quotient optimization. *Annals of statistics*, 43(4):1498, 2015.

Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

Wenjiang J Fu. Penalized estimating equations. *Biometrics*, 59(1):126–132, 2003.

Katsuyuki Fukuda and Yoshiyuki Fujita. Determination of the discriminant score of intestinal microbiota as a biomarker of disease activity in patients with ulcerative colitis. *BMC gastroenterology*, 14(1):49, 2014.

Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*, pages 63–77. Springer, 2005.

Trevor Hastie, Robert Tibshirani, and Andreas Buja. Flexible discriminant analysis by optimal scoring. *Journal of the American statistical association*, 89(428):1255–1270, 1994.

Trevor Hastie, Andreas Buja, and Robert Tibshirani. Penalized discriminant analysis. *The Annals of Statistics*, 23(1):73–102, 1995.

Tiffany Hensley-McBain, Michael C Wu, Jennifer A Manuzak, Ryan K Cheu, Andrew Gustin, Connor B Driscoll, Alexander S Zevin, Charlene J Miller, Ernesto Coronado, Elise Smith, et al. Increased mucosal neutrophil survival is associated with altered microbiota in hiv infection. *PLoS pathogens*, 15(4):e1007672, 2019.

Curtis Huttenhower, Dirk Gevers, Rob Knight, Sahar Abubucker, Jonathan H Badger, Asif T Chinwalla, Heather H Creasy, Ashlee M Earl, Michael G FitzGerald, Robert S Fulton, et al. Structure, function and diversity of the healthy human microbiome. *nature*, 486(7402):207, 2012.

Ross Iaci, TN Sriram, and Xiangrong Yin. Multivariate association and dimension reduction: A generalization of canonical correlation analysis. *Biometrics*, 66(4):1107–1118, 2010.

Catherine Igartua, Emily R Davenport, Yoav Gilad, Dan L Nicolae, Jayant Pinto, and Carole Ober. Host genetic variation in mucosal immunity pathways influences the upper airway microbiome. *Microbiome*, 5(1):16, 2017.

Andrew R Joyce and Bernhard Ø Palsson. The model organism as a system: integrating'omics' data sets. *Nature reviews Molecular cell biology*, 7(3):198–210, 2006.

Seunggeun Lee, Michael C Wu, and Xihong Lin. Optimal tests for rare variant effects in sequencing association studies. *Biostatistics*, 13(4):762–775, 2012.

Hongzhe Li. Microbiome, metagenomics, and high-dimensional compositional data analysis. *Annual Review of Statistics and Its Application*, 2:73–94, 2015.

Catherine Lozupone and Rob Knight. UniFrac: a new phylogenetic method for comparing microbial communities. *Applied and Environmental Microbiology*, 71(12):8228–8235, 2005.

Catherine Lozupone, Manuel E Lladser, Dan Knights, Jesse Stombaugh, and Rob Knight. Unifrac: an effective distance metric for microbial community comparison. *The ISME journal*, 5(2):169, 2011.

Frederick A Matsen, Robin B Kodner, and E Virginia Armbrust. pplacer: linear time maximum-likelihood and bayesian phylogenetic placement of sequences onto a fixed reference tree. *BMC bioinformatics*, 11(1):1–16, 2010.

Ian H McHardy, Maryam Goudarzi, Maomeng Tong, Paul M Ruegger, Emma Schwager, John R Weger, Thomas G Graeber, Justin L Sonnenburg, Steve Horvath, Curtis Huttenhower, et al. Integrative analysis of the microbiome and metabolome of the human intestinal mucosal surface reveals exquisite inter-relationships. *Microbiome*, 1(1):17, 2013.

Caroline M Mitchell, Sujatha Srinivasan, Anna Plantinga, Michael C Wu, Susan D Reed, Katherine A Guthrie, Andrea Z LaCroix, Tina Fiedler, Matthew Munch, Congzhou Liu, et al. Associations between improvement in genitourinary symptoms of menopause and changes in the vaginal ecosystem. *Menopause (New York, NY)*, 25(5):500–507, 2018.

Caroline M Mitchell, Nanxun Ma, Alissa J Mitchell, Michael C Wu, DJ Valint, Sean Proll, Susan D Reed, Katherine A Guthrie, Andrea Z Lacroix, Joseph C Larson, et al. Association between postmenopausal vulvovaginal discomfort, vaginal microbiota, and mucosal inflammation. *American Journal of Obstetrics and Gynecology*, 2021.

Xochitl C Morgan, Boyko Kabakchiev, Levi Waldron, Andrea D Tyler, Timothy L Tickle, Raquel Milgrom, Joanne M Stempak, Dirk Gevers, Ramnik J Xavier, Mark S Silverberg, et al. Associations between host gene expression, the mucosal microbiome, and clinical outcome in the pelvic pouch of patients with inflammatory bowel disease. *Genome biology*, 16(1):67, 2015.

Beresford N Parlett. The rayleigh quotient iteration and some generalizations for nonnormal matrices. *Mathematics of Computation*, 28(127):679–693, 1974.

Vera Pawlowsky-Glahn and Antonella Buccianti. *Compositional data analysis*. Wiley Online Library, 2011.

Anna Plantinga, Xiang Zhan, Ni Zhao, Jun Chen, Robert R Jenq, and Michael C Wu. MiRKAT-S: a community-level test of association between the microbiota and survival times. *Microbiome*, 5(1):17, 2017.

Walter Rudin. Functional analysis, 1973.

Patrick D Schloss, Sarah L Westcott, Thomas Ryabin, Justine R Hall, Martin Hartmann, Emily B Hollister, Ryan A Lesniewski, Brian B Oakley, Donovan H Parks, Courtney J Robinson, et al. Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. *Applied and environmental microbiology*, 75(23):7537–7541, 2009.

Robert F Schwabe and Christian Jobin. The microbiome and cancer. *Nature Reviews Cancer*, 13(11):800–812, 2013.

Jun Shao, Yazhen Wang, Xinwei Deng, Sijian Wang, et al. Sparse linear discriminant analysis by thresholding for high dimensional data. *The Annals of statistics*, 39(2):1241–1265, 2011.

Lars St, Svante Wold, et al. Analysis of variance (anova). *Chemometrics and intelligent laboratory systems*, 6(4):259–272, 1989.

Shan Sun, Anju Lulla, Michael Sioda, Kathryn Winglee, Michael C Wu, David R Jacobs Jr, James M Shikany, Donald M Lloyd-Jones, Lenore J Launer, Anthony A Fodor, et al. Gut

microbiota composition and blood pressure: The cardia study. *Hypertension*, 73(5): 998–1006, 2019.

Gábor J Székely, Maria L Rizzo, et al. Brownian distance covariance. *The annals of applied statistics*, 3(4):1236–1265, 2009.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

Jun Wang, Alexander Kurilshikov, Djawad Radjabzadeh, Williams Turpin, Kenneth Croitoru, Marc Jan Bonder, Matthew A Jackson, Carolina Medina-Gomez, Fabian Frost, Georg Homuth, et al. Meta-analysis of human genome-microbiome association studies: the mibiogen consortium initiative. *Microbiome*, 6(1):101, 2018.

Daniela M Witten, Robert Tibshirani, and Trevor Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10(3):515–534, 2009.

Chong Wu, Jun Chen, Junghi Kim, and Wei Pan. An adaptive association test for microbiome data. *Genome medicine*, 8(1):56, 2016.

Michael C Wu, Lingsong Zhang, Zhaoxi Wang, David C Christiani, and Xihong Lin. Sparse linear discriminant analysis for simultaneous testing for the significance of a gene set/pathway and gene selection. *Bioinformatics*, 25(9):1145–1151, 2009.

Michael C Wu, Seunggeun Lee, Tianxi Cai, Yun Li, Michael Boehnke, and Xihong Lin. Rare-variant association testing for sequencing data with the sequence kernel association test. *The American Journal of Human Genetics*, 89(1):82–93, 2011.

Shuiming Xiao, Na Fei, Xiaoyan Pang, Jian Shen, Linghua Wang, Baorang Zhang, Menghui Zhang, Xiaojun Zhang, Chenhong Zhang, Min Li, et al. A gut microbiota-targeted dietary intervention for amelioration of chronic inflammation underlying metabolic syndrome. *FEMS microbiology ecology*, 87(2):357–367, 2014.

Ping Xu, Guy N Brock, and Rudolph S Parrish. Modified linear discriminant analysis approaches for classification of high-dimensional microarray data. *Computational Statistics & Data Analysis*, 53(5):1674–1687, 2009.

Ryo Yamada, Daigo Okada, Juan Wang, Tapati Basak, and Satoshi Koyama. Interpretation of omics data analyses. *Journal of human genetics*, 66(1):93–102, 2021.

Jun Yu, Qiang Feng, Sunny Hei Wong, Dongya Zhang, Qiao yi Liang, Youwen Qin, Longqing Tang, Hui Zhao, Jan Stenvang, Yanli Li, et al. Metagenomic analysis of faecal microbiome as a tool towards targeted non-invasive biomarkers for colorectal cancer. *Gut*, 66(1):70–78, 2017.

Joseph P Zackular, Mary AM Rogers, Mack T Ruffin, and Patrick D Schloss. The human gut microbiome as a screening tool for colorectal cancer. *Cancer prevention research*, 7(11): 1112–1121, 2014.

Xiang Zhan, Anna Plantinga, Ni Zhao, and Michael C Wu. A fast small-sample kernel independence test for microbiome community-level association analysis. *Biometrics*, 73(4): 1453–1463, 2017a.

Xiang Zhan, Xingwei Tong, Ni Zhao, Arnab Maity, Michael C Wu, and Jun Chen. A small-sample multivariate kernel machine test for microbiome association studies. *Genetic Epidemiology*, 41(3):210–220, 2017b.

Ni Zhao, Jun Chen, Ian M Carroll, Tamar Ringel-Kulka, Michael P Epstein, Hua Zhou, Jin J Zhou, Yehuda Ringel, Hongzhe Li, and Michael C Wu. Testing in microbiome-profiling studies with MiRKAT, the microbiome regression-based kernel association test. *The American Journal of Human Genetics*, 96(5):797–807, 2015.

Appendix A

APPENDIX FOR CHAPTER 2

A.1 Alternating Direction Method of Multipliers for Linear Kernel

Let

$$L_w(\mathbf{c}, \mathbf{z}, \mu, \lambda) = \mathbf{c}^T\mathbf{B}\mathbf{c} + s\|\mathbf{z}\|_1 + \mu(\mathbf{c}^T\boldsymbol{g} - 1) + \frac{w}{2}(\mathbf{c}^T\boldsymbol{g} - 1)^2 + \lambda^T(\mathbf{c} - \mathbf{z}) + \frac{w}{2}\|\mathbf{c} - \mathbf{z}\|^2 \tag{A.1}$$

Then the ADMM algorithm for a linear kernel is

$$\mathbf{c}^{k+1} \leftarrow (2\mathbf{B} + w\boldsymbol{g}\boldsymbol{g}^T + w\mathbf{I})^{-1}\big((wM - \mu^k)\boldsymbol{g} + w\mathbf{z}^k - \lambda^k\big) \tag{A.2}$$

$$\mathbf{z}^{k+1} \leftarrow \mathcal{S}_{s/w}(\mathbf{c}^{k+1} + \frac{1}{w}\lambda^k) = \begin{cases} c_i + \frac{1}{w}\lambda_i - \frac{s}{w} & \text{if } c_i + \frac{1}{w}\lambda_i > \frac{s}{w} \\ 0 & \text{if } |c_i + \frac{1}{w}\lambda_i| \leq \frac{s}{w} \\ c_i + \frac{1}{w}\lambda_i + \frac{s}{w} & \text{if } c_i + \frac{1}{w}\lambda_i < -\frac{s}{w} \end{cases} \tag{A.3}$$

$$\mu^{k+1} \leftarrow \mu^k + w(\mathbf{c}^{k+1^T}\boldsymbol{g} - M) \tag{A.4}$$

$$\lambda^{k+1} \leftarrow \lambda^k + w(\mathbf{c}^{k+1} - \mathbf{z}^{k+1}) \tag{A.5}$$

A.2 Additional Simulation Results for Gaussian Kernel

I this section, we simulate genomic data with non-linear association with microbiome data according to

$$\mathbf{Z}_{ij} = \psi(\sum_{k \in \mathcal{A}_j} \alpha_k \frac{Y_{i(k)}}{\bar{Y}_{(k)}}) + \epsilon_{ij} \tag{A.6}$$

where $\psi(\cdot)$ is a exponential function. To present the results, we assign Gaussian kernel with hyperparameter ρ to genomic data $\mathbf{Z}$. For microbiome data, we use same kernels as in linear settings. Similarly, we also perform simulations in three scenarios as stated in the main text. For hyperparameter choices, we use a grid of ρ to be chosen in 20 grid between 0.01 and 10, presenting the best results among these rho's. For scenario 1 and 2, we choose $\rho = 0.4$, and for scenario 3, we choose $\rho = 0.1$. We just choose among 20 due to limitation of resources. Actually, as increasing of ρ, we obtains a larger False discovery with a larger True discovery. Results are shown in the following tables.

For scenario 1, the Bray-Curtis performs the best compared with other kernels. For scenario 2, weighted Unifrac and unweighted Unifrac win over Bray-Curtis and generalized Unifrac. Comparing between the best two kernels for scenario 2, weighted UniFrac tends to have higher true positive but also slightly higher false positive. For scenario 3, unweighted Unifrac performs the best, though the true positive is much lower than the other scenario because there are rare signals to detect in this setting.

Comparing Gaussian cases with linear cases, results are shown that Gaussian kernels can be used when there are non-linear association between data. However in practice, we suggest use linear method first since Gaussian took more computing resources and memories to run, and need to find a proper hyperparameter ρ to make the model works better.

Method	Metric	5	10	20	30
$\mathbf{K}_{BC}$	TRUE	4.62	9.41	17.59	26.76
	FALSE	0.82	2.04	2.04	1.32
	HAM	1.2	2.63	4.45	4.56
$\mathbf{K}_{w}$	TRUE	4.18	8.25	16.47	24.62
	FALSE	1.27	2.99	3.83	2.63
	HAM	2.09	4.74	7.36	8.01
$\mathbf{K}_{u}$	TRUE	4.11	8.2	16.33	24.5
	FALSE	1.72	3.03	3.41	2.38
	HAM	2.61	4.83	7.08	7.88
$\mathbf{K}_{50}$	TRUE	4.14	8.27	16.43	25.61
	FALSE	1.53	2.81	3.69	2.87
	HAM	2.39	4.54	7.26	7.26
CCA	TRUE	2.79	1.7	1.63	2.66
	FALSE	13.92	1.07	0	0.28
	HAM	16.13	9.37	18.37	27.62

Table A.1: Scenario 1 results, Gaussian Kernel, $\rho = 1$

Method	Metric	5	10	20	30
$\mathbf{K}_{BC}$	TRUE	4.23	8.38	16.51	24.71
	FALSE	1.38	2.85	2.69	1.78
	HAM	2.15	4.47	6.18	7.07
$\mathbf{K}_{w}$	TRUE	4.63	9.22	18.38	27.58
	FALSE	0.25	1.87	1.84	2.75
	HAM	0.62	2.65	3.46	5.17
$\mathbf{K}_{u}$	TRUE	4.15	8.71	17.35	26.49
	FALSE	0.34	1.12	1.83	2.25
	HAM	1.19	2.41	4.48	5.76
$\mathbf{K}_{50}$	TRUE	4.18	8.28	16.43	24.62
	FALSE	1.31	1.66	3.67	2.89
	HAM	2.13	3.38	7.24	8.27
CCA	TRUE	2.05	1.81	3.04	3.27
	FALSE	8.05	2.04	1.57	0.28
	HAM	11	10.23	18.53	27.01

Table A.2: Scenario 2 results, Gaussian Kernel

Method	Metric	5	10	20	30
$\mathbf{K}_{BC}$	TRUE	2.86	5.66	10.97	14.4
	FALSE	2.86	5.66	10.97	14.4
	HAM	3.22	5.88	12.46	18.46
$\mathbf{K}_{w}$	TRUE	2.93	5.63	11.09	15.64
	FALSE	1.58	2.11	4.29	3.71
	HAM	3.65	6.48	13.2	18.07
$\mathbf{K}_{u}$	TRUE	3.12	6.06	12.06	18.06
	FALSE	1.21	1.76	4.09	3.5
	HAM	3.09	5.7	12.03	15.44
$\mathbf{K}_{50}$	TRUE	3.06	5.92	11.6	17.31
	FALSE	1.65	1.28	4.73	3.88
	HAM	3.59	5.36	13.13	16.57
CCA	TRUE	3.12	3.9	6.61	7.67
	FALSE	10.47	5.99	3.95	1.76
	HAM	12.35	12.09	17.34	24.09

Table A.3: Scenario 3 results, Gaussian Kernel