2015

A Case Study Tested Framework for Multivariate Analyses of Microbiomes: Software for Microbial Community Comparisons

Eric M. Spaulding
University of Montana - Missoula

A CASE STUDY TESTED FRAMEWORK FOR MULTIVARIATE ANALYSES OF MICROBIOMES: SOFTWARE FOR MICROBIAL COMMUNITY COMPARISONS

By

Eric Marcus Spaulding

Associate of Science, The University of Montana, Helena, MT, 2007

Bachelor of Science, The University of Montana, Missoula, MT, 2012

Thesis

presented in partial fulfillment of the requirements
for the degree of

Master of Science
in Computer Science

The University of Montana
Missoula, MT

Summer 2015

Approved by:

J.B. Alexander S. Ross Ph.D., Dean
Graduate School

Douglas W. Raiford Ph.D., Chair
Computer Science

Alden H. Wright Ph.D.
Computer Science

William E. Holben Ph.D.
Biological Sciences

Spaulding, Eric M., M.S., August 2015 Computer Science

A Case Study Tested Framework For Multivariate Analyses Of Microbiomes: Software For Microbial Community Comparisons

Chairperson: Douglas W. Raiford

The study of microbiomes is important because our understanding of microbial communities is providing insight into human health and many other areas of interest. Researchers often use genomic data to study microbial organisms, demonstrating differences from one organism to the next. Metagenomic data is utilized to study communities of microbial organisms. The research described herein involved the development of a collection of computational methods.

This suite of computational methods and tools (written in the R and Perl languages) has become a framework used for metagenomic data analysis and result visualization. Multivariate analyses such as Linear Discriminate Analysis (LDA) are used to determine which microbial organisms are useful in distinguishing between microbial communities. The differences between communities are visualized in two or three dimensions using dimensional reduction techniques. Other analyses provided by the framework include, but are not limited to, feature selection, cross-validation, multi-objective optimization, side-by-side comparisons of communities, and identification of core members in a microbial community.

The effectiveness of these methods and techniques was verified in multiple real world case studies such as body fat classification of elk using a fecal microbiome, identification of important changes in community composition when permafrost is thawed, and longitudinal classification of intestinal locations. The fecal microbiome may be used in the future to assist in assessing the health of animal populations using non-invasive samples. Additionally, the analysis of thawing permafrost may yield insight into the release of greenhouse gases into the atmosphere, furthering our understanding of global warming. Our understanding of the intestinal microbiome may someday grant us understanding and control of our intestinal wellbeing, which plays a significant factor in immune system response and overall health.

ACKNOWLEDGMENTS

First, I'm grateful for the Montgomery GI Bill, which was instrumental to my financial ability to attend college. Then, Mike Rosulek's funding through NSF award #1149647 because that opportunity caused me to attempt graduate school, and also to Bill Holben and Douglas Raiford for funding metagenomic research with an NIH R15 grant that allowed me to finish graduate school.

I would also like to take this opportunity to express my thanks to the following people. Everyone at the Holben Lab and Bill Holben himself for the opportunity to work with world class researchers like Ellen Lark, Sam Pannoni, Frances Gilman who never fail to bring me fascinating problems. I want to thank Mike Rosulek for teaching me the theory behind computational algorithms and getting me excited to learn more about the math that drives our world. I really appreciate learning about Artificial Intelligence from Alden Wright because this allowed me consider the infinite possibility our computers really have. I especially need to express my gratitude to Douglas Raiford for his years of steady mentoring and guidance. He's always been there for me when school or life got me down. None of this would have been possible without him, and finally I want to thank the countless others who added to my research and knowledge to make this thesis possible.

TABLE OF CONTENTS

CODE LISTINGS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

When developing a large software project, core functionality is often identified and developed into a code library or suite of libraries. The libraries created during this project will be referred to as the 'framework' in this document. While the framework's purpose is data analysis and visualization, it can be separated into two distinct sections. The first part is the data pipeline (Perl) where data source integration modules allow data from multiple different data sources to be reformatted into the standard data format used by the framework. The rest of the framework (R) is focused on the automation of data analysis and visualization, using algorithms that are not currently part of other approaches.

To date, the main kind of data analyzed by this framework has been partial 16S ribosomal RNA gene sequences that have been PCR amplified. This data comes from environmental samples, which can be anything from feces in the forest to a q-tip swabbed in someone's mouth. These biological samples have to be processed in a laboratory, so that bacterial DNA can be extracted for amplification. The amplified DNA (i.e. amplicons) are purified in gels and then sent to a high-throughput DNA sequencing facility to obtain reads from the amplicons. This process generates thousands or millions of DNA reads per sample. These DNA fragments are parts of the genomes of the bacterial consortium in the original environmental sample; therefore, metagenomic data.

The information contained within metagenomic data is being used to solve and understand many problems. What if we knew precisely which micro organisms in our intestines caused us to be resistant to one disease and vulnerable to another? People are taking advantage of this information by using probiotics and avoiding certain foods. One benefit of understanding the bacterial communities inside of us is being able to make healthier decisions.

1.1 Motivation

Metagenomic data analysis commonly starts with software packages like Quantitative Insights Into Microbial Ecology (QIIME) [1]. Software that combines many other software projects together are commonly referred to as software pipelines. The QIIME pipeline provides a multitude of important data analyses for 16S metagenomic data. For example, one software module used by QIIME, UniFrac, provides beta diversity analysis and visualizations [2]. Beta diversity is a measurement of how different two communities are from each other [3]. Software in the QIIME pipeline can also classify the DNA sequences, allowing us to know which bacterial taxon that DNA sequence is most likely to be associated with. Information like this allows researchers to see the composition of the microbial communities represented by their samples. These communities are referred to as microbiomes [4].

While a current data analysis approach like beta diversity tells us that microbiomes are different from one another, it doesn't tell us which specific bacterial taxon differs from one community to the next.

1.2 Goal

The goal of this framework is to provide data analyses and visualizations that not only show that microbial communities differ but also show how they differ. This will be accomplished through a variety of independent approaches, allowing researchers to have more confidence in the framework's results. First, the framework will identify sets of genera that can specifically be used to tell microbiomes apart. Next, it will identify and visualize the core members (genera) of a microbiome, showing the genera that are the most present in the microbiome being analyzed. Then, the system will directly visualize two microbiomes side-by-side, allowing researchers to see a bird's-eye view of microbiome composition. Finally, these side-by-side microbiome visualizations will be converted to visualizations that show only the differences between the two microbiomes.

1.3 Benefits

Biologists and other scientists will be able to use these tools and libraries to analyze and visualize many different kinds of data. This will help them verify hypothesis-es and publish results that visualize their work.

1.4 Thesis Organization

The rest of this thesis is organized as follows:

- **Chapter 2** Literature review and overview of the framework
- **Chapter 3** Data acquisition and computational methods
- **Chapter 4** Presentation of case study results
- **Chapter 5** Discussion of results, conclusions, and future directions

CHAPTER 2 LITERATURE REVIEW

The field of metagenomic analysis uses a wide variety of tools and algorithms. These can be segregated into categories describing the various problems being solved, which include anything from sampling animals or environments to statistical analysis. One of the key ideas in metagenomic data analysis is the idea of a microbiome [5] [6]. A microbiome can be described as the composition of a microbial community in some location or environment. For example, the skin on a human knee has a microbiome and the inside of a human mouth has yet another microbiome. The main challenege presented by these microbiomes is the speed at which they can change and the variations that can be found between subjects from the same category [7] [8] [9]. Existing software pipelines provide many different kinds of analyses on metagenomic data such as beta diversity, OTU classification, and chimera detection.

2.1 Pipelines

The biggest benefit of a pipeline approach (such as that employed by QIIME) is a workflow that treats many other software projects as swappable modules in a larger computational process. This results in a flexible framework that supports many different algorithms and performs valuable analyses such as denoising and chimera detection. An example of another pipeline that might be supported by this framework in the future is MetAMOS [10]. The framework would only require an integration mod-

ule for MetAMOS output in the data pipeline, which would reformat the MetAMOS data output into the form expected by the framework's R libraries. Some examples of analyses supported by pipelines like these include:

- **Krona charts** - Web based dynamic pie chart visualizing all or part of the microbiome in a dataset at differing degrees of taxonomic resolution [11].
- **Alpha/Beta diversity plots** - Alpha diversity refers to a visualization of species richness in an ecosystem. By contrast, beta diversity is the difference in diversity from one ecosystem to another [12].
- **Heatmaps** - Visual plot showing the density of reads by phylogeny [13].

Pipelines like QIIME provide good flexibility and analysis, but support is more limited for experimenting with new algorithms and research methods. In order to add additional analysis to QIIME a script would have to be written so that it could be added to QIIME's primary python script invoking the new data analysis software. This means that it is possible for this framework to be someday integrated with pipelines like QIIME or MetAMOS, but there is no guarantee which software in use today will be the best or most widely used in the future. For this reason, it was decided that the developed framework should remain independent from pipeline approaches and is more likely to gain a web based graphical user interface than a command line interface.

2.2 Diversity

At the time of this writing, researchers have access to tools like UniFrac, Emperor, phyloseq, HMP, and others to perform data analysis [2] [14] [15] [16]. UniFrac's beta diversity analysis can be performed with either weighted or unweighted phylogenetic distances. Weighted distances take the phylogenetic distances between communities and weight each individual distance by the proportions of the community members. Unweighted distances on the other hand use the phylogenetic distances without taking into account the abundance of the populations in the community. Newer versions of UniFrac unify the weighted and unweighed approaches because the weighted and unweighted distance puts too much emphasis on either rare or abundant lineages [17].

In addition, the phylogenetic trees built in the QIIME pipeline prior to comparative analyses like UniFrac or Emperor can be generated with FastTree [18]. The phylogenetic trees are not necessarily used by the comparative analysis, but can also be used to create heatmap visualizations that describe community composition from a bird's-eye view.

Emperor is another powerful tool supported by QIIME [14]. It provides interactive 3D ordination plots that can be viewed in a web browser. These plots use the Principal Coordinates Analysis (PCoA) algorithm (also called Multi Dimensional Scaling 'MDS') in order to create visualizations of metagenomic data [19]. Emperor is a flexible tool that allows researchers to engage with complex multivariate data in a quick and visual way.

La Rosa describes another approach for determining whether microbiomes are different or not by using the Dirichlet-multinomial distribution [16]. This distribution allows the analyst to perform tests of hypotheses, testing microbiomes across groups. These mathematical techniques are available in the R-Package HMP and can be used to independently verify beta diversity results, showing that sub groups of data are different enough from each other for that difference to be statistically significant [20].

Another relevant framework example is phyloseq [15]. It is also built on the R Language using existing algorithms and tools for data analysis and visualization. Some examples of phyloseq analysis include visualizations of alpha/beta diversity, phylogeny, heatmaps, and microbiome networks. These analyses are useful bioinformatics-oriented extensions to the functionality already provided by R.

What these tools/approaches don't accomplish; however, is a way to describe how microbial communities are different, specifically which bacterial organisms can be used to differentiate one microbial community from another. This is a very hard problem because inter-subject variation (each subject being a single animal or set of environmental samples) in microbial data often overwhelms the differences found between microbial communities [7] [8] [9].

2.3 The Framework

The phylogenetic distances used by other tools like UniFrac are ignored because this framework focuses entirely on community composition. Therefore, in its attempt to overcome inter-subject variation, this tool is focused on statistical models and machine learning algorithms that allow researchers to seek answers to the following types of questions.

- Identify which members of a microbial community are the most important for differentiating some predefined grouping of the samples. For example, what's the difference between microbiomes sampled from two geographically distinct populations of North American Elk (Voting Process Section 3.4.4).
- Visualize the data's groups using only the more discriminatory community members (Scatter Plot Section 3.4.4.5).
- Build a model that is able to predict the labels of future data (Classifier Section 3.4.3).
- Identify and visualize core microbiomes (Core Pie Charts Section 3.5.1).
- Visualize bacterial presence in one microbiome vs. another (Log Ratio Bar Plots Section 3.5.2).

CHAPTER 3 METHODS

The methods described below are organized in roughly the same order they were performed during the case studies. Some methods are independent of the rest, so these methods are arranged at the end.

3.1 Methods Syntax And Conventions

Some of the special syntax conventions used in this document are as follows.

- *Scientific Names* - capitalized and italicized.
- **`Scripts`** - bold and Courier font.
- *Variables* - italicized
- **Functions** - bold
- *System Commands* - italicized and underlined

3.2 QIIME

While the framework can be used on nearly any multivariate data, the origin of the data used during development and testing was QIIME's biom format [21] Operational Taxonomic Unit (OTU) matrices generated by **`pick_otus.py`** [22]. At the bacterial level (metagenomic data) OTUs (clusters of organisms) for phylogentic categories like species or genus are defined by similarity in DNA [23].

Specifically, once samples are taken from an environment, the next step is to recover the DNA by following a protocol like that found in the Qiagen DNA extraction kit [24]. Sample preparation and sequencing involves many steps including DNA recovery, Polymerase Chain Reaction (PCR), gel purification, multiplexing amplicons, sequencing by synthesis, and then demultiplexing to a FASTA format before bioinformatic analysis can begin. The framework currently uses sequence files in the FASTA/Pearson format [25], but could easily be adapted to other formats as needed.

QIIME was used to determine which OTU reference sequence most closely matches each sequence in the FASTA files using PyNAST [26] and Greengenes [27]. From that point, chimera detection was accomplished by UCHIME and USEARCH [28]. Once QIIME has made sequence matches, it assigns a lineage or taxonomy to each sequence using the Ribosomal Database Project (RDP) classifier [29]. The RDP classifier uses the techniques and algorithms described by Greengenes for 16S data [30] [31].

At the end of the QIIME pipeline, the FASTA files are identified to the genus level based on each sequence's similarity to the reference sequence in the database that most closely matched. Having this genus level OTU data from QIIME allowed this framework to focus on statistical analysis with data that has already been processed into OTUs, showing the proportion between the number of reads for each taxon and the number of reads in that sample.

Because the framework acquires data from multiple sources in many different formats, it makes sense to establish a gateway for new data to be fed into the framework in a standard way. I met this need by developing a series of data source integration scripts (modules) written in the Perl language.

3.3 Data Pipeline

Currently, I have developed an integration module written for a few variations of QIIME's OTU table output and RDP output via the Simple Object Access Protocol (SOAP) [32] web-service function seqmatch. Having separate integration scripts for each data source allows the framework to establish a uniform data standard in the main data analysis framework. Additionally, this flexibility leaves open the possibility of integrating other data sources in the future.

The Perl module **`DataPipeline.pm`** was developed in support of the framework described herein so that core functionality like uniform data output, formatting lineage string headers, and normalizing counts data to proportions could be maintained in a single place rather than scattered across data integration scripts. An example of Perl's flexible array usage is shown in Code Listing 3.1 where lineage strings are formated to display the most specific known taxonomy. Additionally, the core module supports functionality for generating the union and/or intersection of data matrices in other Perl scripts. This kind of data manipulation allows multiple QIIME or other data sources to be merged into a single data set.

Code Listing 3.1: Format Lineage Strings

```
sub cleanTaxa{
  my $args = shift;
  #adjust the level because arrays are zero based
  my $level = $args->{'level'} - 1;
  my $data = deleteLineageSymbols({hash => $args->{'hash'}});
  my %cleanedData = ();
  my @phyloLevels =
    ("kingdom","phylum","class","order","family","genus","species");
  my $othercount = 0;
  foreach my $taxon (keys(%$data)){
    my @phylo = split(";",$taxon);

    #if there aren't enough elements inject some blank others
    while(@phylo < ($level + 1)){ push(@phylo, "Other"); }
    my $cleanTaxon = $phylo[$level];
    if($cleanTaxon eq "Other"){
      $othercount++;
      for(my $i = $level; $i >= 0; $i--){
        if($phylo[$i] ne "Other"){
          $cleanTaxon = $phyloLevels[$i] . "_" . $phylo[$i] . "_" .
    $cleanTaxon . "_" . $othercount;
          $i = -1; #stop the loop here because
                   #found the most specific known phylogeny
        }
      }
    }
    $cleanedData{$cleanTaxon} = $data->{$taxon};
  }
  print "there were ".keys(%$data)." taxa\n";
  print "of which $othercount were others\n";
  return \%cleanedData;
}
```

3.3.1 Integration With QIIME's OTU Data

In order to use QIIME's OTU ouput, the **`combineFormatOTUData.pl`** script was developed as the integration module used before data analysis is done in the framework's R libraries. To support a couple variations of QIIME output, this script has five option flags, a target depth, and a target data folder. The target data folder determines the root folder that will be searched for OTU data. Recursively searching the sub folders within this root folder can be disabled or enabled. The target depth

determines what level of OTU data the script is looking for. For example, L6 data would be genus and L2 data would be phylum. The possible levels in order are domain (1), phylum (2), class (3), order (4), family (5), genus (6), and species (7).

The five boolean option flags are *samplerows, docounts, propfromcounts, badsamples,* and *printwithoutothers*. The *samplerows* flag determines the orientation of the matrix, which determines whether the data samples will be the rows or the columns. The *docounts* flag requires that the script is able to find the tab separated file (tsv) generated from the biom format matrix. This tsv file must also have lineage strings available, which can be created with the bash command in Code Listing 3.2.

Code Listing 3.2: Convert Biom Matrix To TSV

```
biom convert -i infile -o outfile --to-tsv --header-key taxonomy
```

This process is necessary because QIIME's OTU matrix output is in the biom format [21]. In order to convert a biom format matrix into a tsv matrix, a UNIX or Linux style operating system (OS) is recommended. Ubuntu 15.2 run from VirtualBox [33] was used to test the *biom* command successfully at the time of this writing. From a Linux-based OS the *biom* command can be installed in a terminal window as seen in Code Listing 3.3. In order for these commands to work, the *pip* command needs to be available. Pip is the PyPA recommended tool for installing python packages, which makes it a good way to install many useful and important bioinformatics tools [34].

Code Listing 3.3: Install PIP

```
pip install numpy
pip install biom-format
```

The *propfromcounts* flag will stop the script from searching for proportions data and instead generate the proportions matrix from counts information. This means that *propfromcounts* can't be true when *docounts* is false because the script would be searching for nothing at all. Next, the *badsamples* flag will cause the script to keyword search for files that list sample names to be removed from the final matrix. This is most likely to be necessary for samples with so few counts that the microbiome is not well represented [4] [35]. Finally, the *printwithoutothers* flag allows a second set of matrices to be printed with all taxa not known to the specified depth removed. For example, a taxon might be known only to the family level, making it an unknown taxon in a genus-level matrix.

The QIIME pipeline currently provides information for bacterial lineages no deeper than the genus level so researchers need to turn to other sources for the sequences that merit higher level resolution. At the time of this writing, the only database providing species level classification of bacterial lineages is the Seqmatch Ribosomal Database Project (RDP) database [36].

3.3.2 Integration With Seqmatch RDP Data

In order to access the Seqmatch RDP database and classify bacteria to the species level, the database has to be accessed over the web. At this time, access the Seqmatch RDP database is only available via the Internet using SOAP (automatable) or a web browser (manual). Because this framework is focused on automation, the **`soapSubmit.pl`** script was developed to submit sequences to this database. The sequences in the FASTA [25] file are sent fifty at a time and the responses from Seqmatch RDP are saved into a file. The current web service connection information can be seen in Code Listing 3.4.

Next, the **processRDPOutput.pl** script I developed represents the framework's integration module with the seqmatch RDP web database. The raw output from the seqmatch RDP database is first collapsed into a smaller file where only the best response for each sequence is recorded. This collapsed information is then processed into proportions and counts OTU matrices following the data pipeline's standard data format.

Code Listing 3.4: Seqmatch RDP SOAP Settings

```
my $ws_path        = 'http://rdp.cme.msu.edu/services/seqmatch';
my $ws_proxy       = 'http://rdp.cme.msu.edu/services/seqmatch';
my $ws_attr        = {'xmlns:n1' =>
    'http://rdp.cme.msu.edu/services/seqmatch'};
my $ws_function    = 'n1:seqmatchWithOptions';
my $ws_wsdl        = 'http://rdp.cme.msu.edu/services/seqmatch?wsdl';
my $ws_xml_schema  = 'http://rdp.cme.msu.edu/services/seqmatch?xsd=1';
```

3.3.3 Filtering A FASTA By OTUs

With some DNA recovery techniques, biological samples may yield millions of sequences. In a case like this the seqmatch RDP process is too slow because it processes sequences at a rate of roughly 50,000 sequences per 24 hours. Large FASTA files with millions of sequences can be sent through a QIIME pipeline, resulting in genus-level OTU classification. The **filterFastaByClusteredTaxonomy.pl** script was developed as a three stage process to filter a large FASTA file full of raw sequences down to smaller FASTA files containing only the sequences associated with a specific OTU lineage.

First, the function **getReferenceIds** searches through a folder and keyword searches the lineage strings in the OTU files. This allows the script to search for a genus, class, or phylum all at the same time with the benefit that each search will be considered independent from any other search. By allowing simultaneous searches, the script

saves researchers computational time and effort. At the end of this process, the OTU IDs (reference ids) are known for each keyword search being done. Next, the function **getSequenceIds** uses a bunch of cluster files (created by QIIME) and builds a list of the original sequence ids that are associated with the OTU IDs found by **getReferenceIds**. For the last step of the process, the function **filterFasta** opens up the large FASTA file and uses the lists of sequence ids found by **getSequenceIds** to build smaller *filtered* FASTA files for each search being done. These smaller FASTA files can then be sent to the seqmatch RDP database, allowing researchers to gain species level resolution for the OTUs they are most interested in.

Once the Perl scripts in the data pipeline have formatted the data received from QIIME or Seqmatch RDP, the framework's real work begins. The R language is used to perform statistical analysis and machine learning to visualize trends in the data.

3.4 Data Analysis Portion

Below is a complete listing of R Packages required by the framework. The reason for choosing the R language for this task is CRAN's large support for statistical and graphical libraries [37].

- **R** - The language used for analysis and visualization [38].
- **rgl** - Allows for openGL graphics in R [39].
- **MASS** - Implements LDA and much more [40].
- **Rcpp** - Lets R use and communicate with c++ code [41].
- **biom** - API allowing access to the biom format [42].
- **HMP** - Provides several functions to perform formal hypothesis testing [20].
- **gridBase** - Generate a list of grid viewports which correspond to the current inner, figure, and plot regions of the current base plot [43].
- **grid** - Adds an nx by ny rectangular grid to an existing plot [43].
- **ellipse** - Routines for drawing ellipses and ellipse-like confidence regions [44].
- **Cairo** - Allows for the output of graphics with anti-aliasing in Windows [45].
- **digest** - Enables SHA-1 encoding [46].

3.4.1 Floating Search

One of the most common problems in multivariate data analysis is feature selection [47]. The reason it's so important is the need to identify what parts of a given dataset are the most important within the context of a specific research question. For example, a researcher with microbial data might want to know specifically which intestinal bacteria differentiate a sick host from a healthy host. The framework attempts to enable researchers to ask questions like this by using the Floating Search algorithm [48] in novel ways to achieve Feature Selection. The floating search algorithm is used to identify a set of dimensions (features) in a dataset in order to optimize separation and tight grouping of predefined groups (clusters). The J3 score algorithm [49] is used by the floating search algorithm as a scoring function to assign scores to subsets of dimensions.

Once the Floating Search algorithm has identified a good set of features, there is still a need to further reduce the data set to two or three dimensions because humans can not visualize data in more than three dimensions. In order to accomplish this task without losing critical information, the framework employs dimensional reduction techniques.

3.4.2 Dimensional Reduction

Visualization of data to check for grouping and separation is often done using two or three dimensional scatter plots. Dimensional reduction techniques are an effective means for taking a larger dataset and projecting it into a newly generated vector space such that the large majority of the dataset's critical information is present in a few dimensions.

The framework supports three forms of dimensional reduction: Linear Discriminate Analysis (LDA) [50], Principal Component Analysis (PCA) [51], and Multi Dimensional Scaling (MDS) [19]. Multi Dimensional Scaling is also referred to as Principal Components Analysis (PCoA). While all three of these mathematical techniques facilitate visualization, LDA is used most often in this framework because, in addition to visualization (Scatter Plots), LDA is also used as a predictor of future data (Classifier [52]).

3.4.3 Supervised Learning

In order to build an LDA model (Classifier), data with known answers (labels) is required. Once an LDA model has been constructed, it can be used to label new data without using a known answer. Before trusting these answers, mathematical verification of the model's quality should be performed so that researchers have an understanding of the model's reliability. When testing the performance (reliability) of a classifier (model), cross-validation is a common approach [53].

Cross-fold-validation is accomplished by splitting a dataset up into pieces (groups). For each fold of the cross-validation process, one group of the data will be withheld from the classifier. All of the other groups will then be used to train the classifier (build a model). This model is then used to predict the labels of the withheld data. The model's frequency of correct predications provides the researcher with a classifier accuracy (confidence) for his or her data.

Similar to the concept of withholding data when testing a classifier, this information should also be withheld when choosing features. In order to have more confidence in the features chosen by the framework, a novel ranking algorithm has been developed by moving the feature selection step into a cross-validation process.

3.4.4 Ranking Features

In order to identify which features best describe subgroups within a data set, the framework uses statistical information gathered while building LDA classifier models inside of cross-validation. That information is used to 'vote' on or rank the features (dimensions) of the data.

3.4.4.1 Feature Selection Inside of Cross-Validation

The framework collects statistical information about the data using a two-step process for each fold of cross-validation. First, feature selection is performed to select a subset of features based on only the training data for the fold. Then, and LDA model is constructed from the training data using only the chosen features.

This process can be configured to use either leave-one-out cross-validation or leave-group-out cross-validation, which determines the number of cross-validation folds and the data withheld during each fold. Since each fold can potentially select different features, there is a problem knowing which features to use when visualizing the data. Therefore, the features that *are* chosen need to be recorded in a statistics database. For example, a feature chosen in seven folds of a ten-fold cross-validation would have a value of seven in the database. The process of collecting statistical information is described by the pseudo code in Code Listing 3.5.

There is also an independent cross-validation process for each number of features to be found by feature selection. Therefore, the statistics database tracks all of the information about dozens of cross-validation events. This allows the framework to consider not only which features are best, but also how many features should be used.

Pseudo Code Listing 3.5: Leave-Group-Out Cross-Fold-Validation

```
Function LGO-CV( matrix, n_genera ):
  n_samples = number of rows of matrix
  n_correct = 0
  for each group g (LGO fold):
    The test samples are those from group g
    The training samples are all remaining samples
    Do floating search on training samples with n_genera genera
    Build LDA model with training data filtered to selected genera
    Predict test data with the LDA model
    n_correct += number samples correctly classified
  return n_correct/n_samples
```

3.4.4.2 The Statistics Database

The database of statistical information is then analyzed to identify which features best discriminate between the dataset's groups, resulting in a separate analysis for each number of features being considered as seen in Table 4.1. This analysis is visualized in a box and whisker plot (Figure 4.1). The box plot analysis determines each set of features by counting the frequency that each feature is chosen during the cross-validation process and then normalizing that frequency by the number of folds and overall performance of the LDA classifier. This process results a ranking of features that favors features that are chosen the most often by the floating search and weights those features by their actual performance with the classifier.

3.4.4.3 Picking the Number of Features

In order to assist the researcher in choosing the best number of features to use, the framework visualizes the possible choices in a box and whisker plot [43] [54] [55] where the y-axis represents LDA classifier accuracy and the x-axis represents the number of features used. Each box has multiple LDA classifier accuracies dependent on the number of noise reduction levels applied to the data. For example, a data set might

have five levels of noise reduction applied, resulting in five independent data sets that would go through the entire data analysis process until they are all compiled into the box and whisker plot. Because microbiome data tends to be very sparse, noise reduction in this case is represented by the removal of exceptionally sparse taxa, specifically any taxa with nonzero values in fewer than 1, 3%, 5%, 8%, and 16% of all relevant samples are filtered out of the base matrix. This 'pruning' of sparse data results in up to five separate levels of data used in each box of the box and whisker plot.

Additionally, the researcher can look at the box and whisker plot to identify peaks in the classifier's performance that indicate possible over fitting of the data [56]. An example of over fitting can be seen in the top panel of Figure 4.5 at around 17 features. The pseudo code in Code Listing 3.6 further describes the process by which the framework chooses each set of features associated with one of the boxes on the box and whisker plot.

Pseudo Code Listing 3.6: Voting For Genera By Weighted Frequency

```
Function choose_genera( matrix, n_genera, info generated by LGO-CV ):
  for each LGO-CV:
    cv_accuracy = accuracy returned by LGO-CV
    for each genus g:
      count_folds[g] = 0
      for each fold of CV:
        if fold chooses g then count_folds[g] += 1
      accuracy[g] += count_folds[g]*cv_accuracy
  Sort genera into descending order by accuracy[g]
  return set of the best n_genera genera
```

3.4.4.4 Multi-Objective Optimization

The process of choosing the correct dimensions requires choosing a number of features that exhibits high average accuracy and low variation in those achieved accuracies, while giving preference to higher feature counts. The framework approaches this task by using a multi-objective optimization algorithm called Pareto Optimal Analysis [57]. This algorithm is represented in R by the functions seen in Code Listing 3.7 and 3.8. This technique allows the framework to balance several objectives against one another, giving the researcher several optimal solutions to work with when visualizing his or her data (Figure 4.5). Additionallly, the dominant points on the frontier are also written to a Comma Separated Values (CSV) file, so that the framework and researcher can look at the Pareto information in more detail.

Code Listing 3.7: Find N-Dimensional Pareto Frontier

```
#this function will return the rows of the matrix
#that make up the pareto frontier
findParetoBoundary = function(m){
  rows = nrow(m)
  cols = ncol(m)
  pareto_frontier = c()
  for(point in 1:rows){
    domination = apply(m[-point,], 1, function(d)
                  paretoDomination(m[point,], d))
    if(sum(domination) == 0){ #no point dominates this point
      pareto_frontier = c(pareto_frontier, point)
    }
  }
  o = order(m[pareto_frontier,1])
  return(pareto_frontier[o])
}
```

Code Listing 3.8: Check For Pareto Domination

```
paretoDomination = function(point, dominator){
  numdims = length(point)
  if(numdims != length(dominator)){
    print("points of varying lengths can't exist in the same
    vectorspace")
    return(NULL)
  }
  greater = FALSE
  peer    = TRUE
  for(d in 1:numdims){
    if(dominator[d] > point[d]){ greater = TRUE; }
    if(dominator[d] < point[d]){ peer = FALSE; }
  }
  return(greater && peer)
}
```

3.4.4.5 Scatter Plots

Once the researcher has determined the number of features, the final step of the ranking features process is visualizing the results (Figure 4.8). The framework supports a variety of different visual options for the scatter plots listed below.

- **Color** - Groups within the data can be colored using either a global color palette or local choices.
- **Symbols** - Points on a scatter plot can have different symbols. These symbols can be applied to the same groups as the colors or a different grouping of the data.
- **Centroids** - Black dots can be added to the center of each group.
- **Ellipses** - To assist in visualizing groups within the data ellipses can be added at 1 standard deviation.
- **Lines of Demarcation** - Lines can be drawn to visualize the intersection of two groups (distributions).

Scatter plot visualizations are useful for showing separation between groups of data. This framework supports other visualizations as well. Some examples are pie charts and bar plots of microbiome distributions.

3.5 Additional Analysis For Metagenomic Data

In order to further explore and visualize aspects of microbiome data, the framework allows researchers to quickly see their data in pie charts and bar plots.

3.5.1 Core Microbiome

Visualizing the core of a microbiome gives researchers insight into the most important or at least most consistent bacteria in their samples. These bacteria are likely to be the most significant micro organisms in a set of samples often warranting further investigation.

The definition of a core microbiome with respect to a data set is a list of the bacteria that are present in every sample in a predefined group of the data (Figure 4.9). The **taxa_filter** function seen in Code Listing 3.9 can then be applied twice to a dataset a single group at a time in order to identify the core of the samples in question. By default the function will return statistical information about the core. When the names parameter is set to true, the function will return a listing of the names that are members of the core rather than the statistical information associated with those members.

Code Listing 3.9: Filter Taxa To Find Core Microbiome

```
taxa_filter = function(taxon, threshold, names=FALSE){
  filter = taxon > threshold
  if(sum(filter) == length(filter)){
    if(names){
      return(1)
    } else {
      stats = list(median=median(taxon),
                   min=min(taxon),max=max(taxon))
      return(stats);
    }
  }
  else { return(NULL); }
}
```

Another way of comparing microbiomes is a side-by-side comparison of each microbiome's members, which allows researchers to visually see the microbiomes that their OTU tables describe.

3.5.2 Comparing Taxa Presence Between Microbiomes

The concept behind these bar plot visualizations was the desire to compare different DNA extraction techniques against each other. Once the data has been segregated into groups (microbiomes), the R code works by tallying each group by one of four possible quantification methods.

- **Count** - This method counts 1 for each sample a given taxon is found in.
- **Sum** - A straight forward sum of each taxon's presence within the microbiome.
- **Arithmetic Mean** - The average presence for each taxon.
- **Geometric Mean** - The multiplicative rate of each taxon's change across samples.

The most useful method is an arithmetic mean because this method does not favor microbiome data represented by more samples. The other methods do bring their own unique strengths and weaknesses, allowing microbiomes to be compared in slightly different ways. The count method for example results in a low resolution view of the microbiomes but doesn't require logarithmic smoothing to compare results across bacteria. All of these methods result in a bird's-eye view of the microbiomes, showing researchers where bacterial taxa are (or aren't) present in their data. Additionally, for visualization purposes, the presence of each taxon is logarithmically smoothed allowing taxa with high microbiome representation (50% or more) to be seen alongside taxa with low representation (.1% or less).

First, the bar plot is rendered after picking a quantification method (Figure 4.38). This plot shows averaged presence for each sample site in the rat gut with bacterial taxa (y-axis) and 0 to 1 normalized presence (x-axis).

In order to tell which bacteria are more represented in one microbiome or the other, the next step is to generate a log ratio [58] bar plot (Figure 4.39). A log ratio can be described as logarithms of ratios, which allows the bar plot to grow proportionally to either the left or right as the ratio changes. This technique is like putting the two sides of the bar plot into a tug-of-war, causing the differences in bacterial presence between the two microbiomes to become very noticeable. For example, this was used to determine that more *Lactobacillaceae* was found by Qiagen extraction than MoBio.

CHAPTER 4 RESULTS

The Data Analysis and Visualization for Bioinformatics framework has been used in several independent microbiome-based case studies. Not every study used every analysis provided by this framework.

4.1 Mouse Longitudinal Case Study

In the longitudinal mouse study, samples were taken from two cohorts of mice in six locations: ileum, cecum, tip of cecum, proximal colon, mid colon, and distal colon. This was done on two separate occasions a year apart (i.e. cohorts). From this information, six subsets of the data were analyzed and visualized. These subsets of the data were comprised of strain C57B1/6 and strain CD-1, which were designated strains B and C, respectively. Separate mirrored analyses were performed for each mouse strain. The subsets were represented by four, three, and two chamber comparisons. The ileum, cecum, proximal colon, and distal colon make up the four chamber analysis, which found 14 genera (strain B) and 15 genera (strain C). The three chamber analysis consisted of the proximal, mid, and distal colon, while the cecum and tip of cecum was used for the two chamber analysis. These analyses found 13 genera (strain B) and 12 genera (strain C) for the two chamber sets, and they resulted in 13 genera (strain B) and 16 genera (strain C) for the three chamber sets.

For each of the six analyses, the data was further split into five matrices with each matrix imposing increasingly strict criteria on the genera. In the framework this is referred to as the pruning levels of a dataset in which genera are only kept if they have nonzero values in at least 1, 3%, 5%, 8%, or 16% of the samples. Next for each matrix, LDA cross-fold-validation was performed with feature selection done on each training set (fold) of the process. Information about which features were used, how often they were used, and how they performed was then collected into a statistics database. This database was then visualized in box and whisker plots for each of the six analyses (Figures 4.1, 4.2, and 4.3) with strain B (top) and strain C (bottom). To further assist the researcher in the difficult task of choosing the best number of features (best box from the box and whisker plot), the framework also performed multi-objective optimization on the box plot's boxes. This information was then saved to a CSV and visualized in 3D Pareto frontiers (Figures 4.4, 4.5, and 4.6) with strain B (left) and strain C (right). Using the Pareto frontier, an optimal number of features was chosen for each analysis (Tables 4.1, 4.2, 4.3, 4.4, 4.5, and 4.6). With the six sets of features in hand, the framework visualized each set of genera using LDA (Figures 4.7 and 4.8) and performed LDA cross-validation to ascertain its confidence in the genera chosen.

Additonally, core microbiomes (Figures 4.10 and 4.9) were identified for each intestinal chamber in order to find the most prominent members (genera) of each microbiome. These prominent genera were then compared to the discriminatory genera found earlier.

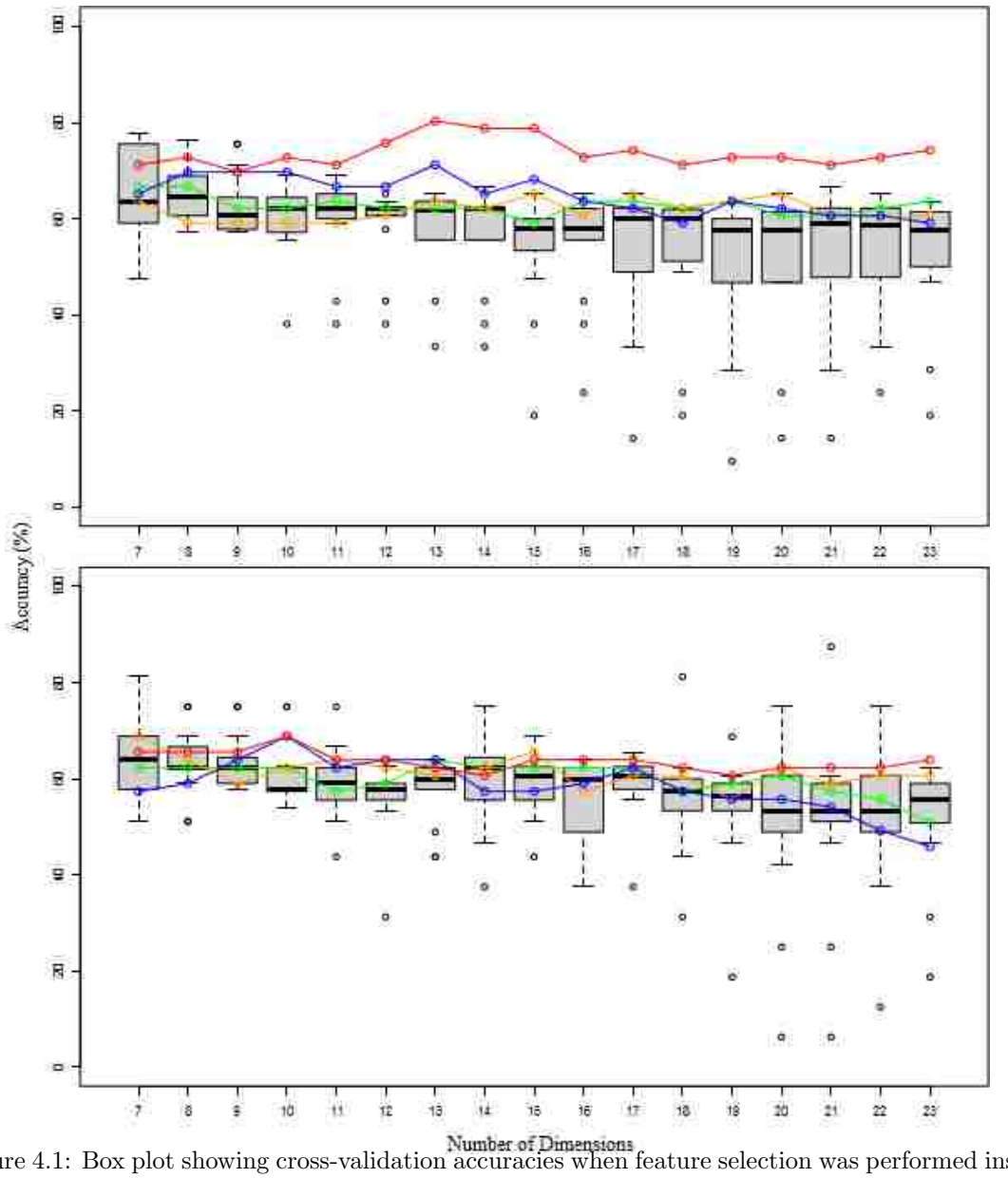

Figure 4.1: Box plot showing cross-validation accuracies when feature selection was performed inside of cross-validation to different numbers of dimensions. The base dataset used was Cohort1&2 with no unknown genera with Strain B (top) and Strain C (bottom) filtered to 4 chambers: Ileum, Cecum, Proximal Colon, and Distal Colon. The green line shows the accuracy found when using the P1 or complete dataset. The orange line shows the accuracy found when using the P16% dataset. The blue line shows the accuracy found when feature selection was performed outside (before) LDA cross-validation. The red line represents the cross-validation accuracy achieved when a set of taxa chosen from the 3D-Pareto frontier are used to perform LDA cross-validation.

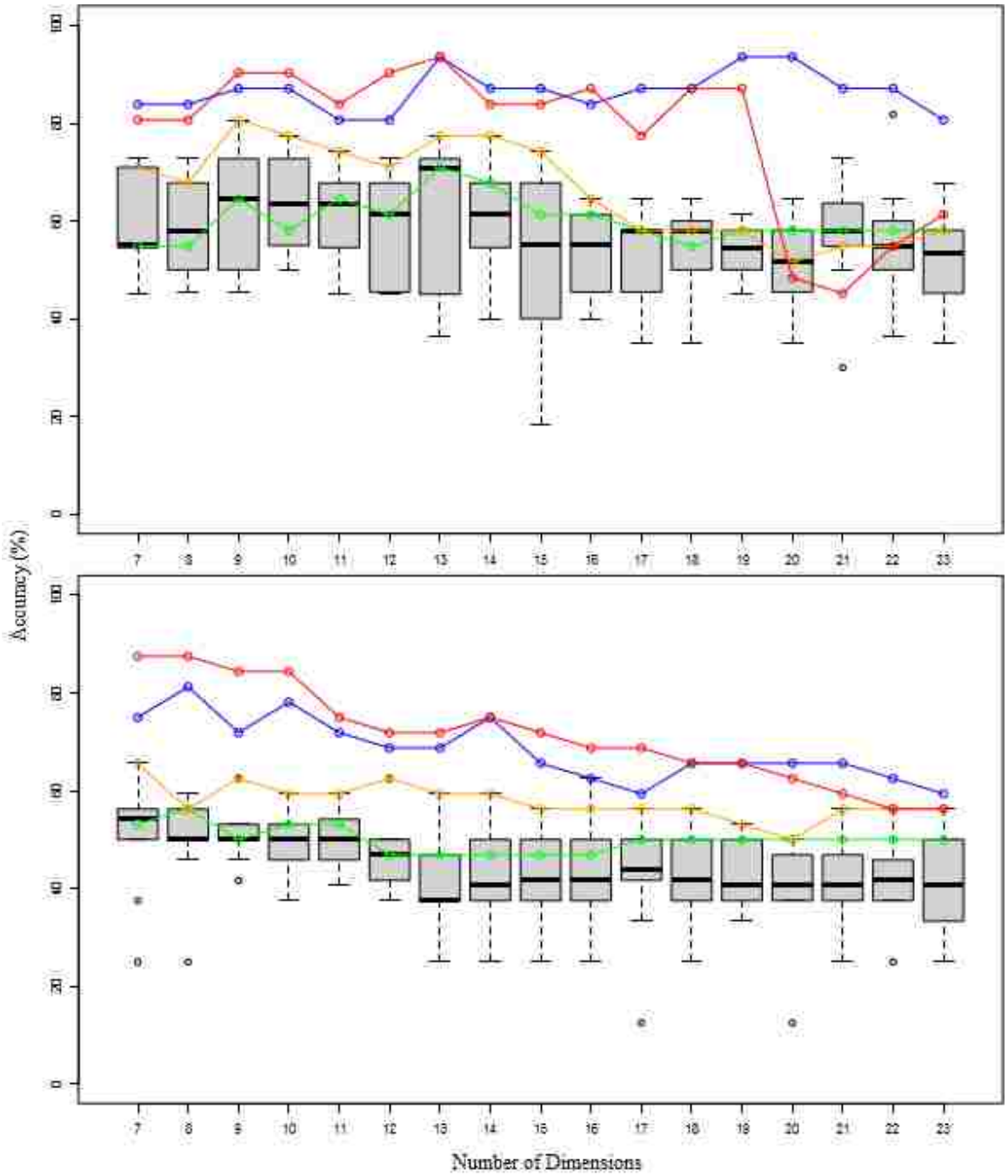

Figure 4.2: Box plot showing cross-validation accuracies when feature selection was performed inside of cross-validation to different numbers of dimensions. The base dataset used was Cohort1&2 with no unknown genera with Strain B (top) and Strain C (bottom) filtered to 2 chambers: Cecum and Tip of Cecum. See Figure 4.1 for more information.

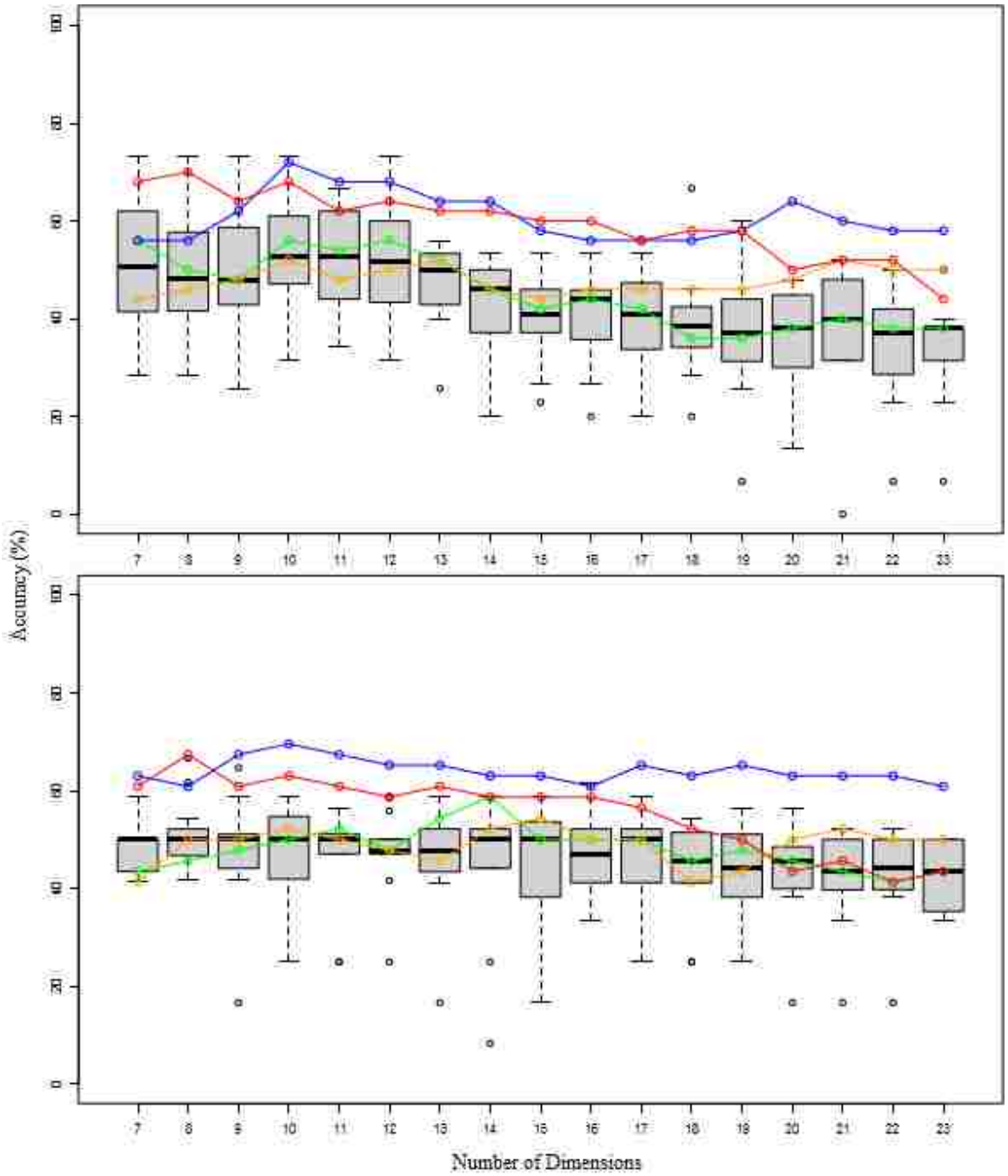

Figure 4.3: Box plot showing cross-validation accuracies when feature selection was performed inside of cross-validation to different numbers of dimensions. The base dataset used was Cohort1&2 with no unknown genera with Strain B (top) and Strain C (bottom) filtered to 3 chambers: Proximal Colon, Mid Colon, and Distal Colon. See Figure 4.1 for more information.

To assist in selecting the number of features, the 3D Pareto frontier scatter plots visualize a multi objective optimization of the longitudinal mouse intestine case study's box plots. The dominant points on the Pareto frontier have mathematically demonstrated an ideal balance between good median accuracy and low variance while favoring higher numbers of features.

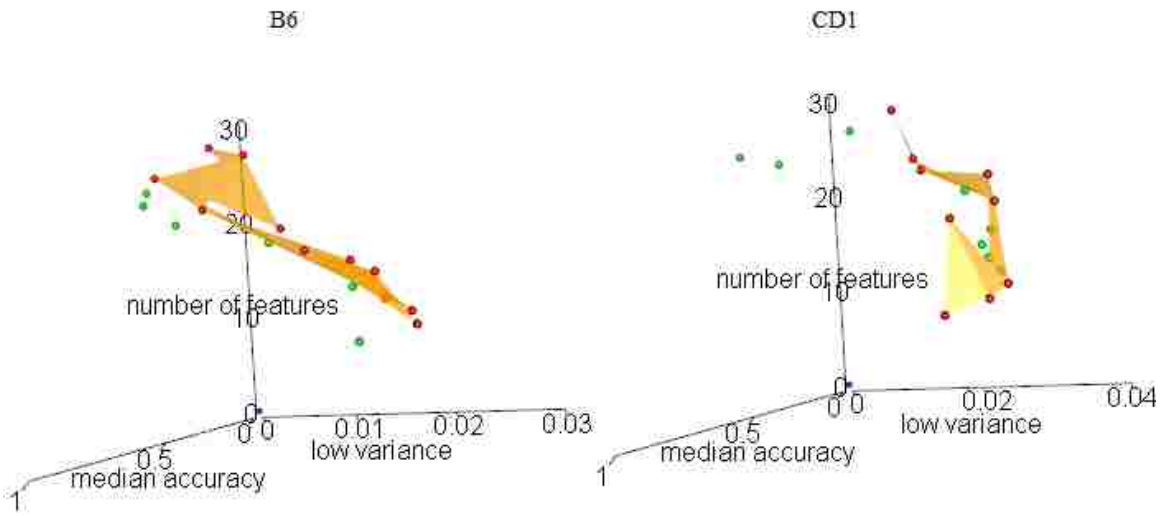

Figure 4.4: Scatter plot visualizing a 3D-Pareto Frontier, optimizing median cross-validation accuracy, low variance, and higher numbers of features. The base dataset used was Cohort1&2 with no unknown genera with Strain B (left) and Strain C (right) filtered to 4 chambers: Ileum, Cecum, Proximal Colon, and Distal Colon. Scatter plot of the 3D-Pareto frontier when the above box plot boxes are optimized by median accuracy, lowest variance, and number of dimensions. The green points represent boxes that are dominated. The red points represent boxes that are dominated by no other box, showing equally optimal solutions. The orange border is a series of triangles drawn when the red points are sorted by median accuracy and sets of 3 points are taken using a sliding window to draw n - 2 triangles. The blue point in the background represents the origin (0,0,0) as a frame of reference.

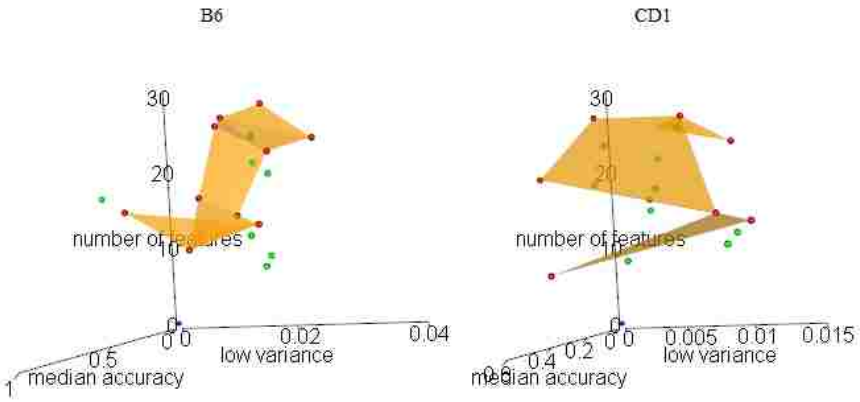

Figure 4.5: Scatter plot visualizing a 3D-Pareto Frontier, optimizing median cross-validation accuracy, low variance, and higher numbers of features. The base dataset used was Cohort1&2 with no unknown genera with Strain B (left) and Strain C (right) filtered to 2 chambers: Cecum and Tip of Cecum. See Figure 4.4 for more information.

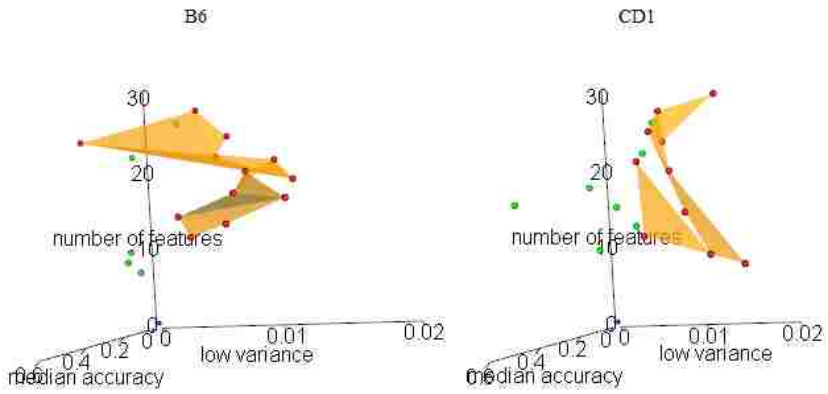

Figure 4.6: Scatter plot visualizing a 3D-Pareto Frontier, optimizing median cross-validation accuracy, low variance, and higher numbers of features. The base dataset used was Cohort1&2 with no unknown genera with Strain B (left) and Strain C (right) filtered to 2 chambers: Proximal Colon, Mid Colon, and Distal Colon. See Figure 4.4 for more information.

Tables 4.1 and 4.2 represent the genera identified using the voting process for the four chamber LDA plot visualized in Figure 4.7.

Table 4.1: Strain B - 14 genera identified using the voting process

Genera	*Rank*
Oscillibacter	*6.75*
Lactobacillus	*6.38*
Robinsoniella	*6.24*
Ruminococcus	*5.93*
Barnesiella	*5.65*
Dorea	*5.63*
Coprobacillus	*4.97*
Coprococcus	*4.76*
Butyricimonas	*4.64*
Blautia	*4.33*
Turicibacter	*4.28*
Mucispirillum	*4.03*
Anaerotruncus	*3.57*
Parabacteroides	*3.46*

Table 4.2: Strain C - 15 genera identified using the voting process

Genera	*Rank*
Lactobacillus	*6.80*
Dorea	*6.54*
Turicibacter	*6.49*
Oscillibacter	*6.31*
Sporacetigenium	*6.23*
Robinsoniella	*6.01*
Akkermansia	*5.80*
Marvinbryantia	*5.63*
Asaccharobacter	*5.45*
Anaerotruncus	*5.35*
Bacteroides	*5.17*
Butyricicoccus	*5.09*
Coprobacillus	*4.89*
Papillibacter	*3.98*
Sporobacter	*3.60*

Tables 4.3 and 4.4 represent the genera identified using the voting process for the two chamber LDA plot visualized in Figure 4.8 (top).

Table 4.3: Strain B - 13 genera identified using the voting process

Genera	*Rank*
Lactobacillus	*4.76*
Parabacteroides	*4.65*
Bacteroides	*4.10*
Turicibacter	*3.70*
Robinsoniella	*3.48*
Butyricimonas	*3.38*
Mucispirillum	*3.27*
Barnesiella	*3.17*
Holdemania	*3.04*
Lactonifactor	*3.03*
Anaerovorax	*2.82*
Marvinbryantia	*2.72*
Sporobacter	*2.40*

Table 4.4: Strain C - 12 genera identified using the voting process

Genera	*Rank*
Sporacetigenium	*2.85*
Lactobacillus	*2.53*
Butyricicoccus	*2.46*
Ruminococcus	*2.27*
Coprobacillus	*2.25*
Oscillibacter	*2.21*
Parabacteroides	*2.09*
Asaccharobacter	*2.04*
Johnsonella	*1.93*
Turicibacter	*1.93*
Roseburia	*1.91*
Dorea	*1.83*

Tables 4.5 and 4.6 represent the genera identified using the voting process for the three chamber LDA plot visualized in Figure 4.8 (bottom).

Table 4.5: Strain B - 13 genera identified using the voting process

Genera	*Rank*
Oscillibacter	*5.09*
Robinsoniella	*4.00*
Dorea	*3.76*
Butyricimonas	*3.28*
Alistipes	*3.12*
Ruminococcus	*3.10*
Barnesiella	*3.07*
Bacteroides	*3.00*
Coprobacillus	*2.99*
Enterorhabdus	*2.92*
Blautia	*2.81*
Holdemania	*2.65*
Parabacteroides	*2.55*

Table 4.6: Strain C - 16 genera identified using the voting process

Genera	*Rank*
Bifidobacterium	*4.44*
Dorea	*4.42*
Parasutterella	*4.01*
Turicibacter	*3.99*
Anaerotruncus	*3.75*
Akkermansia	*3.61*
Lactobacillus	*3.53*
Sporobacter	*3.39*
Coprobacillus	*3.32*
Bacteroides	*3.31*
Robinsoniella	*3.29*
Parabacteroides	*3.12*
Johnsonella	*3.07*
Holdemania	*2.99*
Allobaculum	*2.48*
Marvinbryantia	*2.40*

The LDA scatter plots presented are the final product of the analysis performed in the longitudinal mouse study. These plots visualize the longitudinal separation between bacterial communities found in intestinal chambers (Figures 4.7 and 4.8) for each of the six sub-sets of the data, using bacteria with genus level resolution.

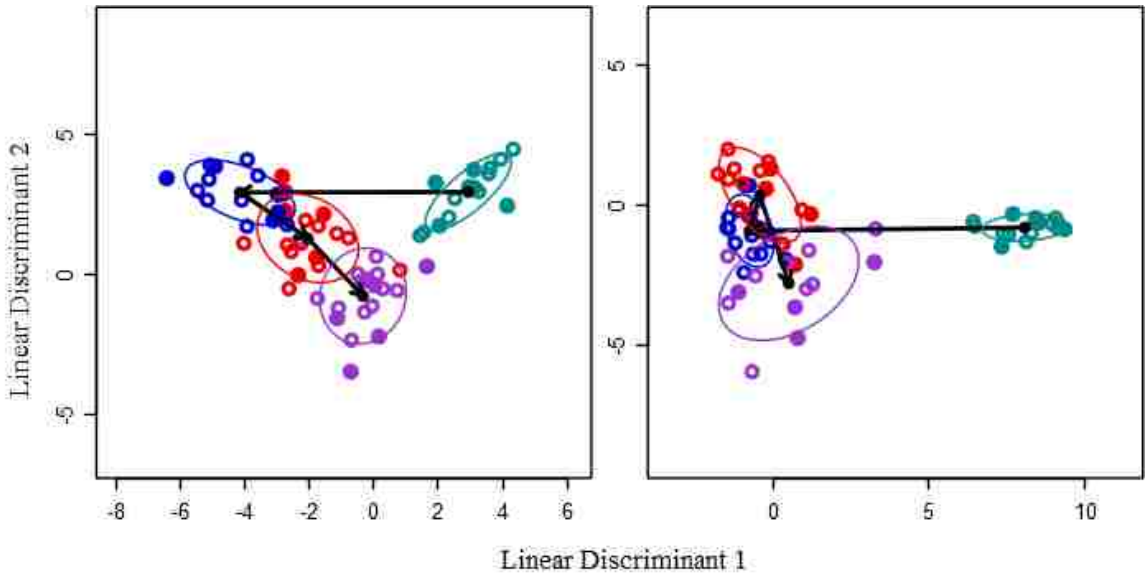

Figure 4.7: Linear Discriminant Analysis (LDA) of the four main compartments sampled from C57B1/6 strain mice (left panel) and CD-1 strain mice (right panel). Filled circles and open circles represent cohorts 1 and 2, respectively. Black dots represent the centroid for each cluster and ellipses indicate 1 standard deviation. The arrows show the flow of digesta between chambers. The plots were made using vote-determined genera shown in Tables 4.1 and 4.2. The accuracies were 78.79% (62.12%)(left panel) and 63.93% (65.57%)(right panel). The first accuracies listed used the vote-determined genera, while the right side accuracies were for genera identified using 'floating search within each cross-validation fold'.

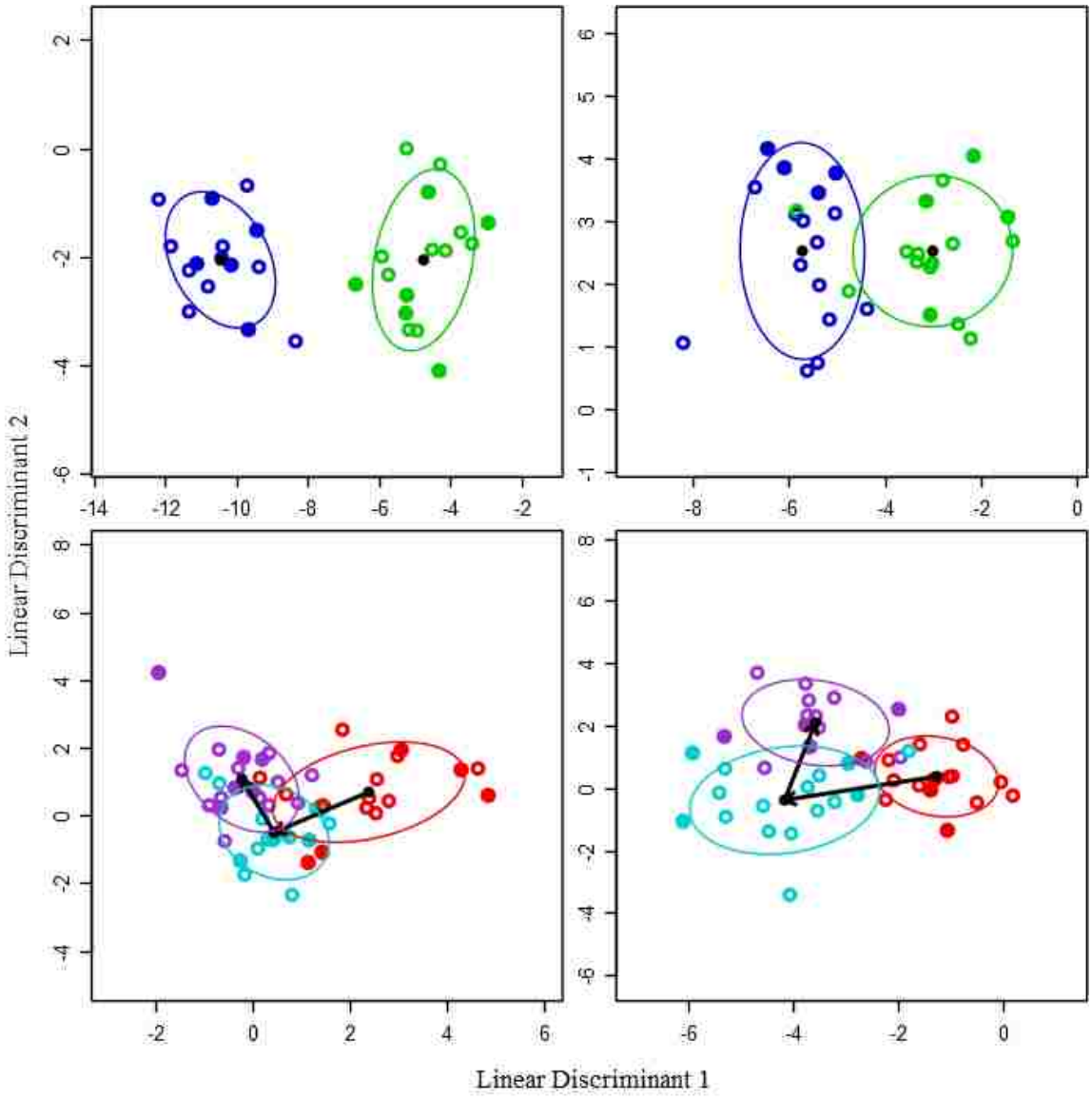

Figure 4.8: LDA of the Tip of the Cecum and Cecum (top panels) and Proximal, Mid, and Distal Colon (bottom panels) for C57B1/6 strain mice (left) and CD-1 strain mice (right). Filled circles and open circles represent cohorts 1 and 2, respectively. Black dots represent the centroid for each cluster and ellipses indicate 1 standard deviation. The arrows show the flow of digesta between chambers. The plots were made using vote-determined genera shown in Tables 4.3, 4.4, 4.5, and 4.6. The accuracies were 93.55% (77.42%)(top left), 71.88% (62.50%)(top right), 62.00% (52.00%)(bottom left), 58.70% (50.00%)(bottom right). The first accuracies listed used the vote-determined genera, while the right side accuracies were for genera identified using 'floating search within each cross-validation fold'.

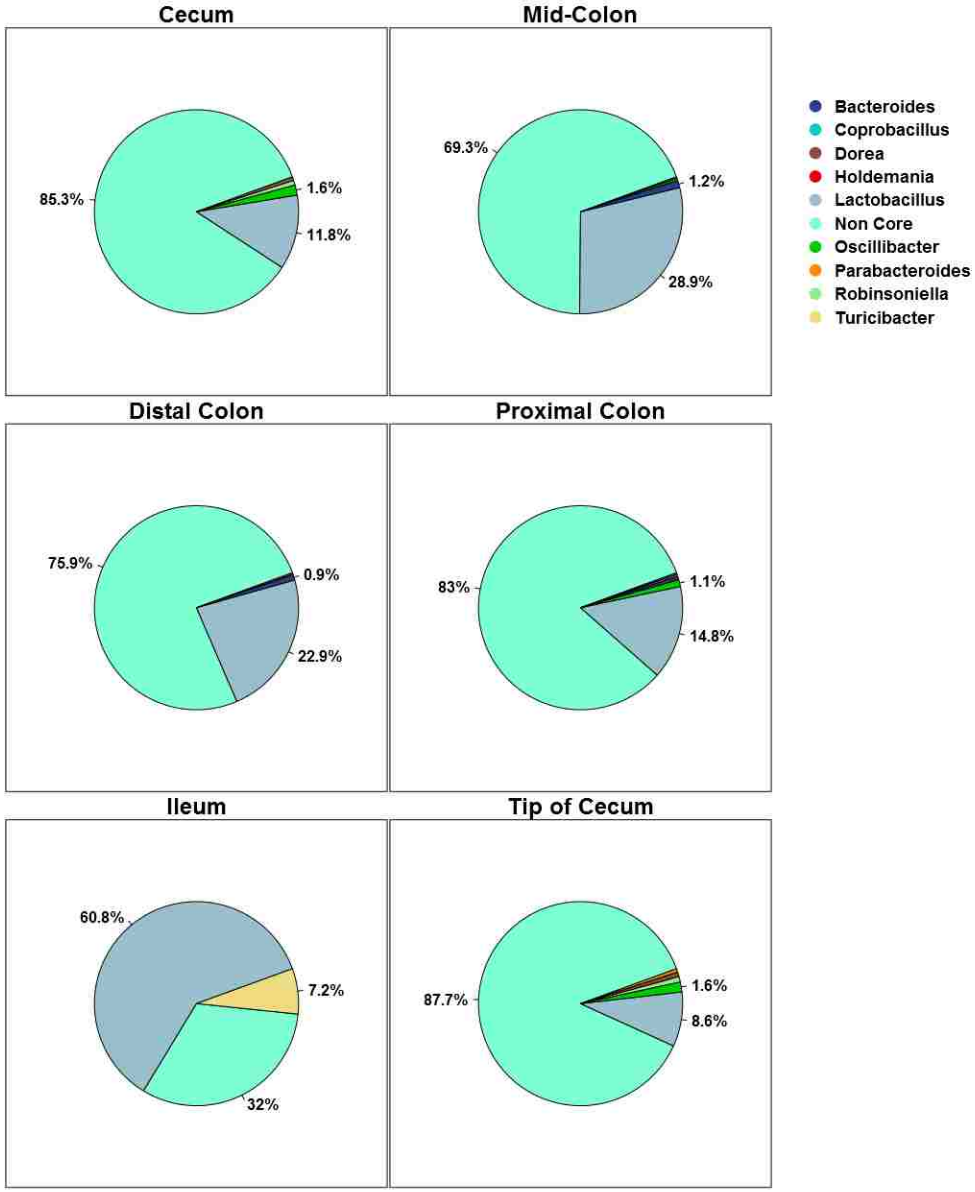

Figure 4.9: Pie charts visualizing the core and noncore microbiome for each intestinal chamber of the mice. Each plot represents the core genera (as described in Section 3.5.1) across both strains of mice for that chamber shown relative to the noncore genera.

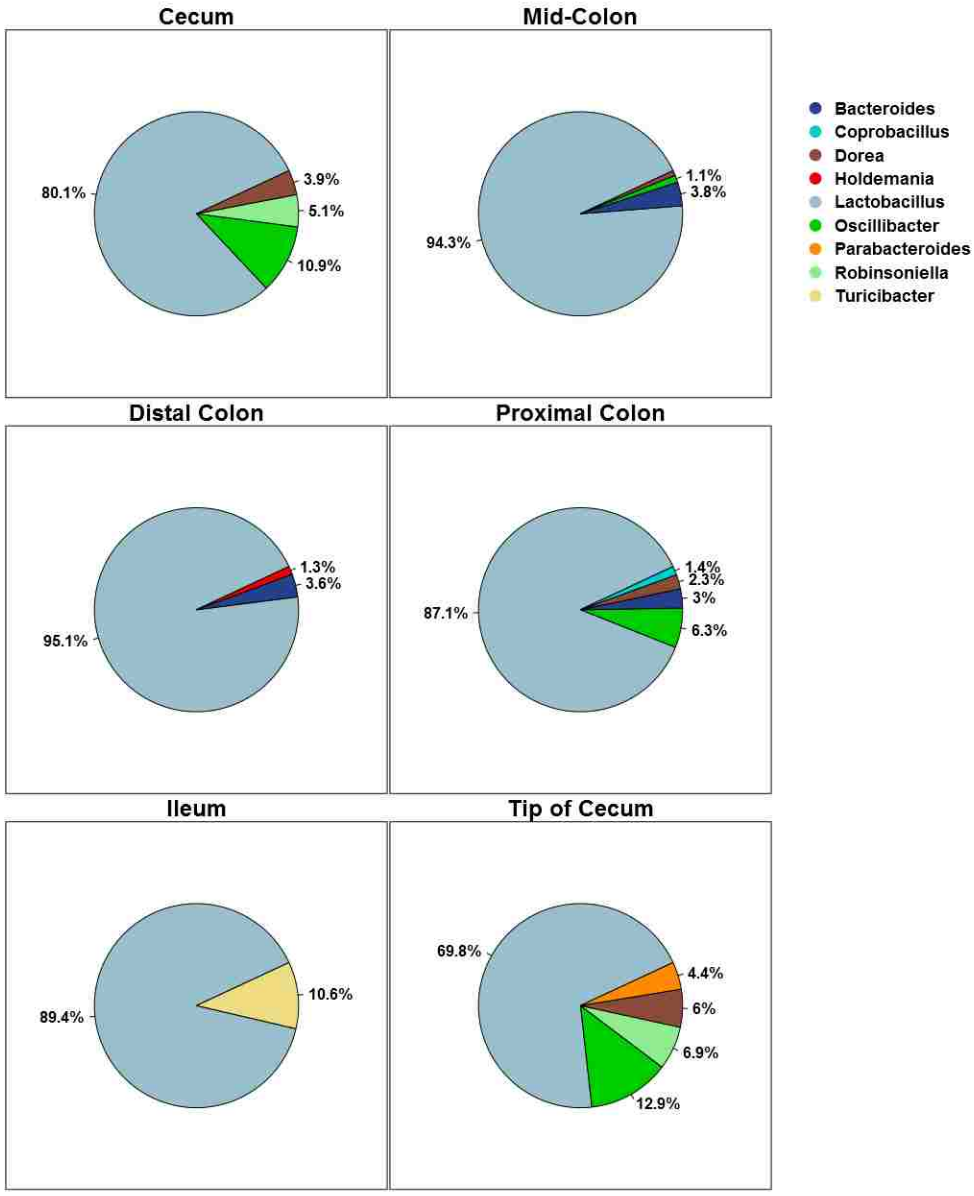

Figure 4.10: Pie charts visualizing the core microbiome for each intestinal chamber of the mice. Each plot represents the core genera (as described in Section 3.5.1) across both strains of mice for that chamber.

4.2 Elk Fecal Microbiome Study

For the elk fecal microbiome study, elk were tagged and fecal samples were collected from four separate Elk populations in the Missoula, Montana area: Bitterroot, Blacks Ford, Sapphire, and Tobacco Root. When the elk were tagged, basic information like age, gender, body fat content, and thyroid hormone levels was collected. Unfortunately, body fat information was not collected for the Bitterroot elk, so those elk were excluded from the body fat analysis. Two sets of analyses were performed in which the elk were first grouped by region and then body fat. The body fat groupings were comprised of the following categories: (6 to 7%] 20 samples, (7 to 8%] 21 samples, (8 to 9%] 15 samples, (9 to 10%] 11 samples, and (10%+] 5 samples.

For both analyses, the data was further split into five matrices with each matrix imposing increasingly strict criteria on the genera. In the framework this is referred to as the pruning levels of a dataset in which genera are only kept if they have nonzero values in at least 1, 3%, 5%, 8%, or 16% of the samples. Next for each matrix, LDA cross-fold-validation was performed with feature selection done on each training set (fold) of the process. Information about which features were used, how often they were used, and how they performed was then collected into a statistics database. This database was then visualized in box and whisker plots for each of the analyses (Figure 4.11) with body fat (top) and region (bottom). To further assist the researcher in the difficult task of choosing the best number of features (best box from the box and whisker plot), the framework also performed multi-objective optimization on the box plot's boxes. This information was then saved to a CSV and visualized in 3D Pareto frontiers (Figure 4.12) with body fat (left) and region (right). Using the Pareto frontier an optimal number of features was chosen for each analysis as seen in Tables 4.7 and 4.8. With both sets of features in hand, the framework visualized each

set of genera using LDA (Figures 4.13 and 4.14) and performed LDA cross-validation to ascertain its confidence in the genera chosen.

Additonally, core microbiomes (Figures 4.16, 4.15, 4.18, and 4.17) were identified for each region and body fat category in order to find the most prominent members (genera) of each microbiome. These prominent genera were then compared to the discriminatory genera found earlier. Even though the elk's fecal core microbiomes accounted for only about 15% of the whole, they were too diverse to analyze in full. Therefore, a threshold of 0.005% used, requiring genera to have at least that much presence in every sample in order to qualify for that microbiome's core.

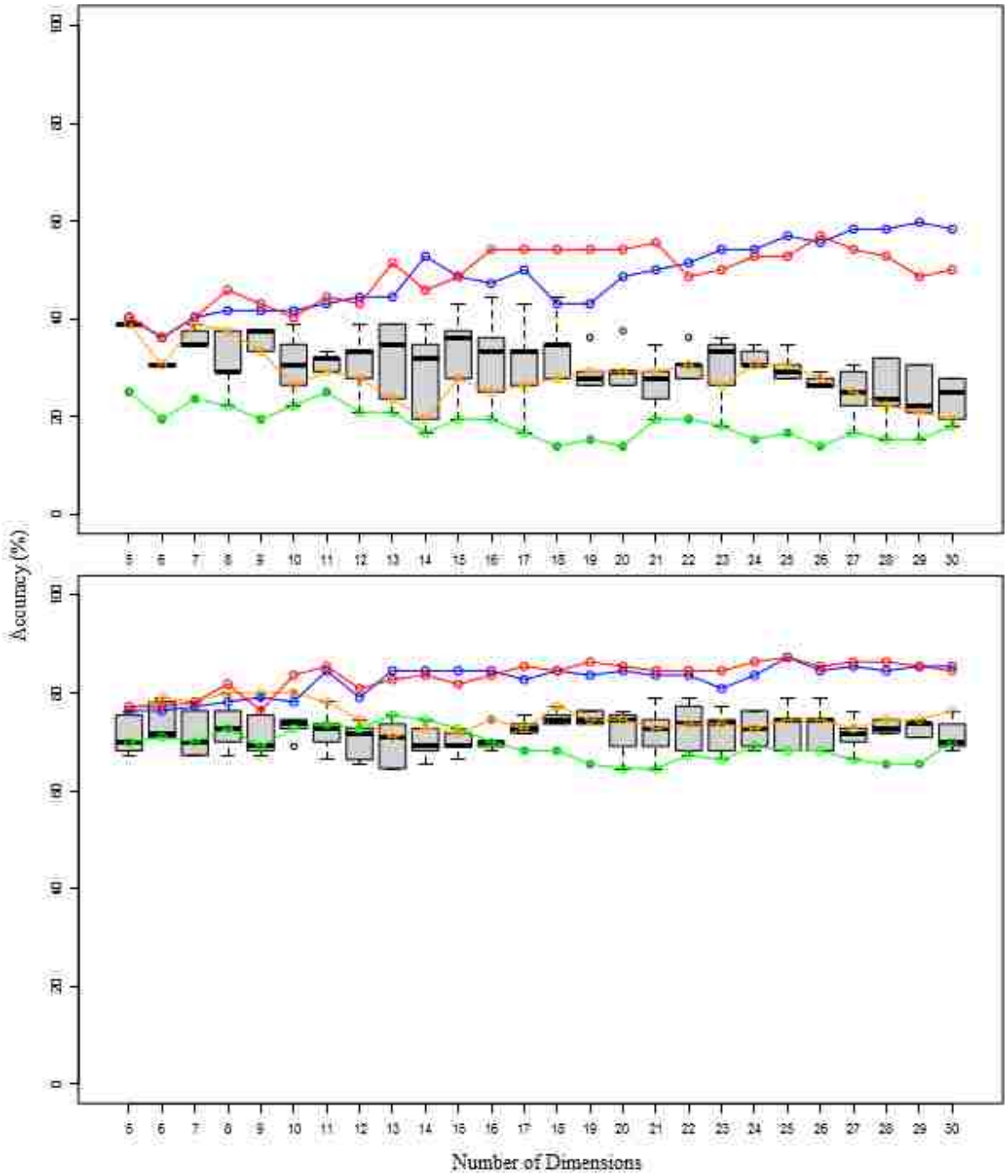

Figure 4.11: Box plot showing cross-validation accuracies when feature selection was performed inside of cross-validation to different numbers of dimensions. The base dataset used was population 1 through 4 with no unknown genera with body fat (top) and region (bottom). See Figure 4.1 for more information.

To assist in selecting the number of features, the 3D Pareto frontier scatter plots visualize a multi objective optimization of the elk fecal microbiome case study's box plots. The dominant points on the Pareto frontier have mathematically demonstrated an ideal balance between good median accuracy and low variance while favoring higher numbers of features.

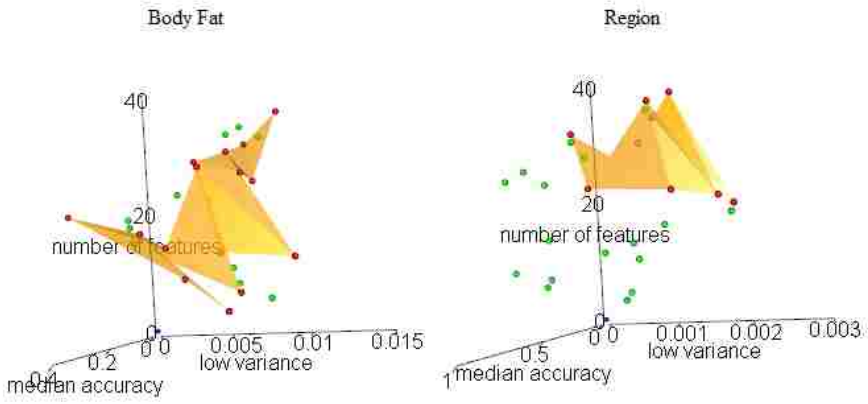

Figure 4.12: Scatter plot visualizing a 3D-Pareto Frontier, optimizing median cross-validation accuracy, low variance, and higher numbers of features. The base dataset used was population 1 through 4 with no unknown genera with body fat (left) and region (right). See Figure 4.4 for more information.

Tables 4.7 and 4.8 represent the genera identified using the voting process for the LDA plots visualized in Figures 4.13 and 4.14.

Table 4.7: Body Fat - 21 genera identified using the voting process

Genera	*Rank*
02d06	*1.27*
Adlercreutzia	*1.27*
Odoribacter	*1.27*
Oscillospira	*1.27*
Sporobacter	*1.24*
Sutterella	*1.23*
Coprobacillus	*1.16*
L7A E11	*1.14*
Mogibacterium	*1.10*
Dorea	*1.05*
Slackia	*1.01*
rc4-4	*0.95*
Roseburia	*0.80*
Methanimicrococcus	*0.79*
Paraprevotella	*0.66*
Nitrosomonas	*0.57*
CF231	*0.56*
BF311	*0.54*
Treponema	*0.53*
Faecalibacterium	*0.53*
Elusimicrobium	*0.53*

Table 4.8: Region - 17 genera identified using the voting process

Genera	*Rank*
CF231	*3.55*
Victivallis	*3.46*
Papillibacter	*2.92*
Slackia	*2.83*
Blautia	*2.62*
Anaerorhabdus	*2.52*
Lactobacillus	*2.40*
Friedmanniella	*2.11*
Sporosarcina	*2.01*
Streptococcus	*1.99*
Mogibacterium	*1.98*
Salinibacterium	*1.96*
Butyricicoccus	*1.94*
Butyrivibrio	*1.84*
SMB53	*1.78*
Roseburia	*1.56*
Bacteroides	*1.39*

The LDA scatter plots presented below are the final product of the analysis performed in the elk fecal microbiome study. These plots visualize the separation of fecal microbiomes by region (Figure 4.13) and then body fat (Figure 4.14), using bacteria with genus level resolution.

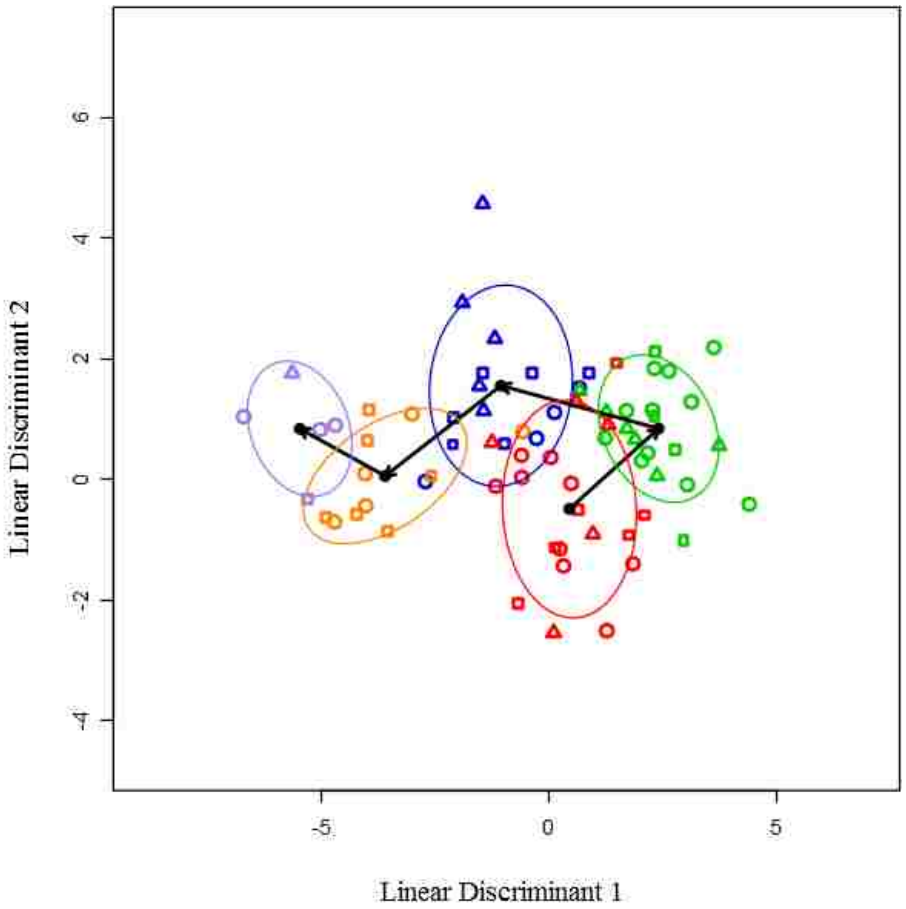

Figure 4.13: Linear Discriminant Analysis (LDA) of the body fat groups sampled from Sapphire, Blacks Ford, and Tobacco Root elk. Circles, triangles, and squares represent Sapphire, Blacks Ford, and Tobacco Root, respectively. Black dots represent the centroid for each cluster and ellipses indicate 1 standard deviation. The arrows show progression from less to more body fat. The accuracies were 55.56% (29.17%). The first accuracy listed used the vote-determined genera (Table 4.7), while the right side accuracy was for genera identified using 'floating search within each cross-validation fold'.

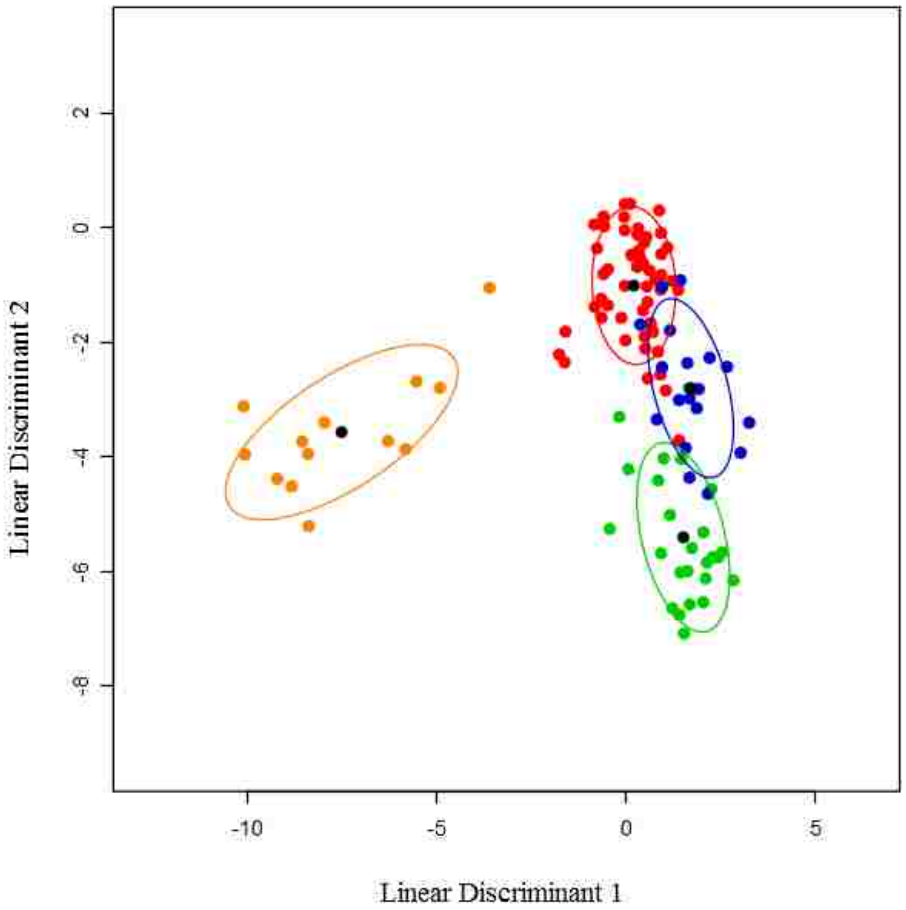

Figure 4.14: Linear Discriminant Analysis (LDA) of the regional groups sampled from Sapphire, Blacks Ford, Bitterroot, and Tobacco Root elk. Black dots represent the centroid for each cluster and ellipses indicate 1 standard deviation. The accuracies were 85.45% (72.73%). The first accuracy listed used the vote-determined genera (Table 4.8), while the right side accuracy was for genera identified using 'floating search within each cross-validation fold'.

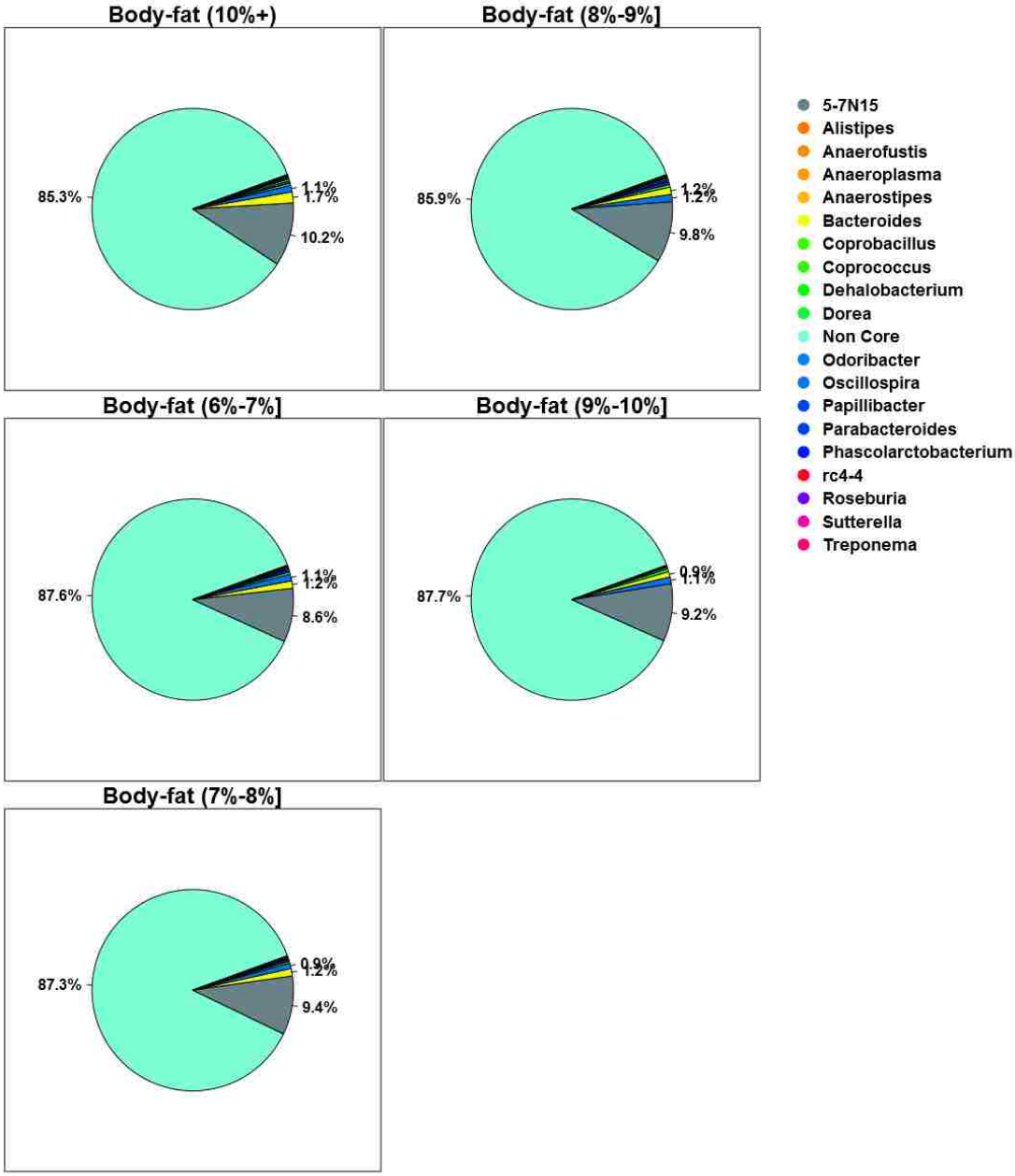

Figure 4.15: Pie charts visualizing the core and noncore microbiome for each body fat group of the elk. Each plot represents the core genera (as described in Section 3.5.1) across all elk for that group using a .005% threshold.

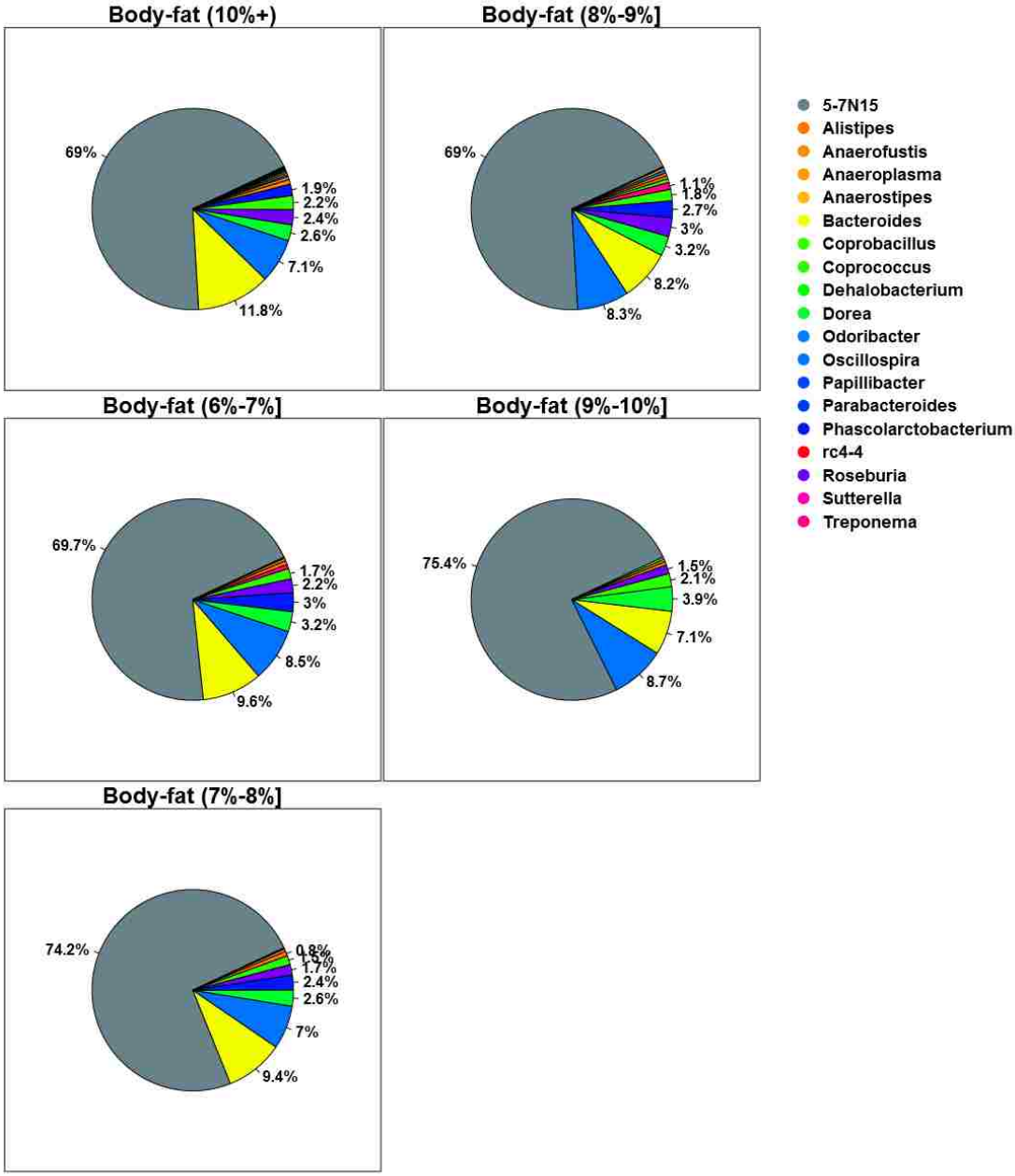

Figure 4.16: Pie charts visualizing the core microbiome for each body fat group of the elk. Each plot represents the core genera (as described in Section 3.5.1) across all elk for that group using a .005% threshold.

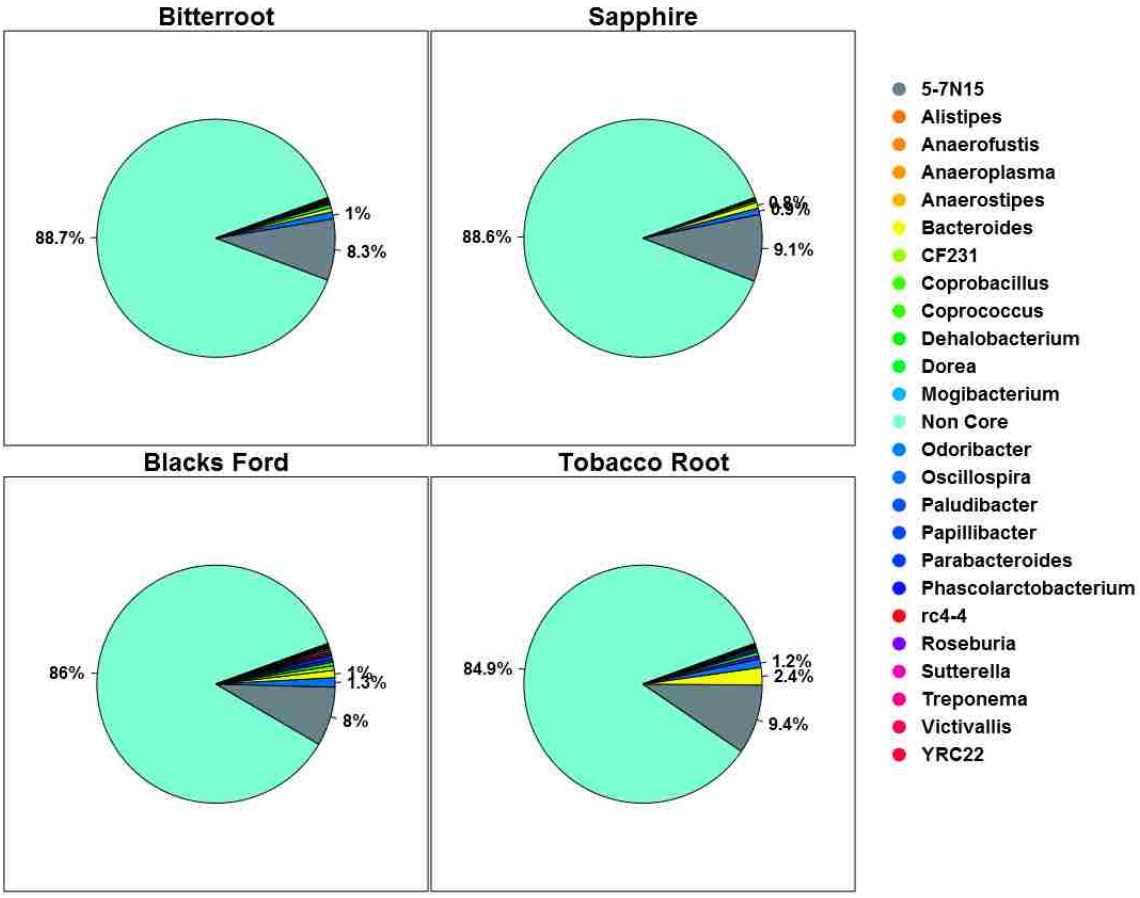

Figure 4.17: Pie charts visualizing the core and noncore microbiome for each regional group of the elk. Each plot represents the core genera (as described in Section 3.5.1) across all elk for that group using a .005% threshold.

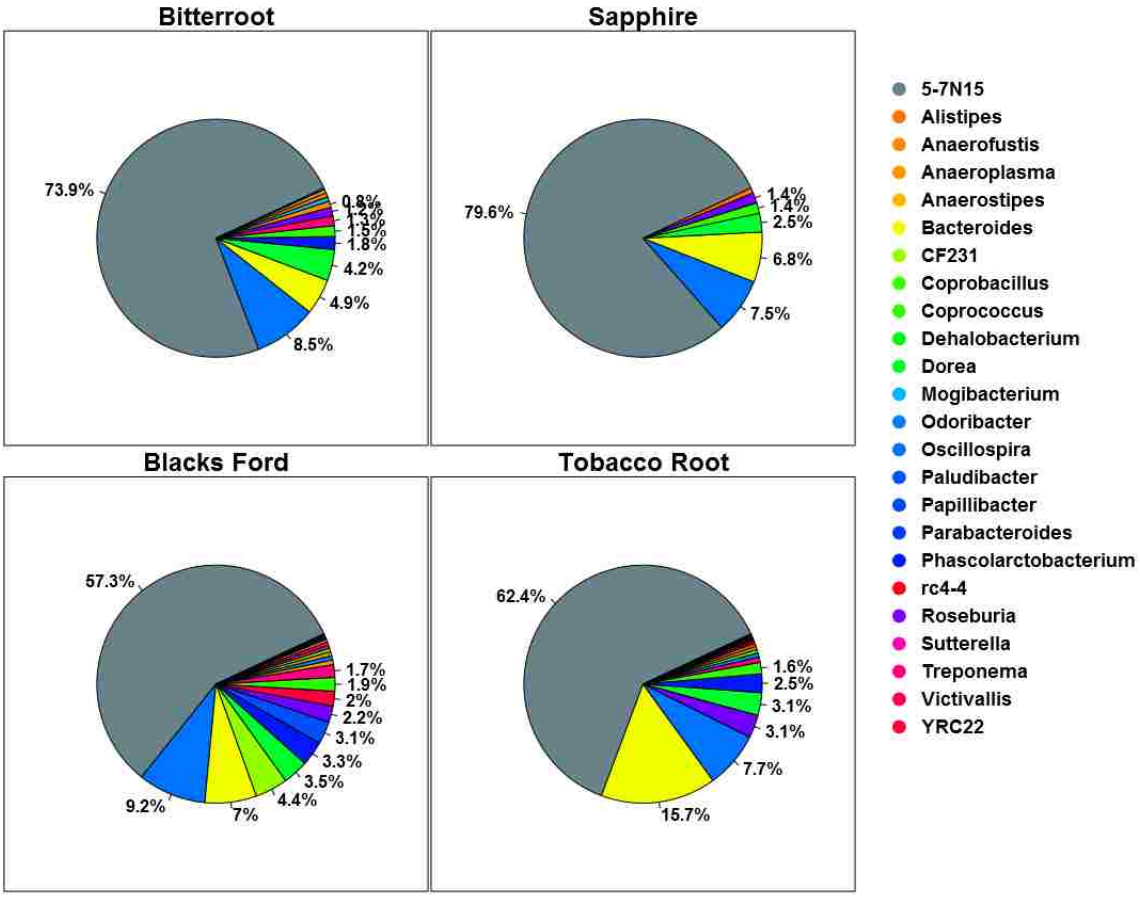

Figure 4.18: Pie charts visualizing the core microbiome for each regional group of the elk. Each plot represents the core genera (as described in Section 3.5.1) across all elk for that group using a .005% threshold.

4.3 Effects Of Warming On Permafrost Microbiomes

For the last decade, the significance of interactions between microbial communities and climate change have exceeded the expectations of the scientific community. These interactions are not yet fully understood because of the large microbial diversity in soil and permafrost. A further complication to understanding these systems is the constant changes in climate currently affecting these soil-based communities around the world. It is currently believed that as permafrost thaws it does release an unanticipated abundance of greenhouse gases into the atmosphere. Research in this area is actively progressing in studies being done around the world [59].

For this case study, six permafrost soil areas were divided into eight plots each on Disko Island, Greenland during the spring of 2012. Samples were then collected from the permafrost active layer in June, July, and late August of 2013. These eight plots can be described as being snow-side vs. protected, warmed vs. natural, and shrub removal vs. normal. The control plot can then be described as natural and normal, allowing the study to focus on the microbiome differences caused by enhanced snow accumulation, temperature, and vegetation. The samples were then grouped by season collected and treatment. Only four of the eight treatments were analyzed: control, snow-side control, warmed, and snow-side with warming, allowing this case study to focus on seasonal differences vs. constant temperature differences.

For both analyses, the data was further split into five matrices with each matrix imposing increasingly strict criteria on the genera. In the framework this is referred to as the pruning levels of a dataset in which genera are only kept if they have nonzero values in at least 1, 3%, 5%, 8%, or 16% of the samples. Next for each matrix, LDA cross-fold-validation was performed with feature selection done on each training set (fold) of the process. Information about which features were used, how

often they were used, and how they performed was then collected into a statistics database. This database was then visualized in box and whisker plots for each of the analyses (Figure 4.19) with season (top) and treatment (bottom). To further assist the researcher in the difficult task of choosing the best number of features (best box from the box and whisker plot), the framework also performed multi-objective optimization on the box plot's boxes. This information was then saved to a CSV and visualized in 3D Pareto frontiers (Figure 4.20) with season (left) and treatment (right). Using the Pareto frontier an optimal number of features was chosen for each analysis as seen in Tables 4.9 and 4.10. With both sets of features in hand, the framework visualized each set of genera using LDA (Figures 4.21 and 4.22) and performed LDA cross-validation to ascertain its confidence in the genera chosen.

Additonally, core microbiomes (Figures 4.24, 4.23, 4.26, and 4.25) were identified for each season and permafrost treatment in order to find the most prominent members (genera) of each microbiome. These prominent genera were then compared to the discriminatory genera found earlier. Similar to the elk study's fecal microbiome, the permafrost microbiomes were very diverse, but not so diverse as to require a special threshold during core microbiome identification.

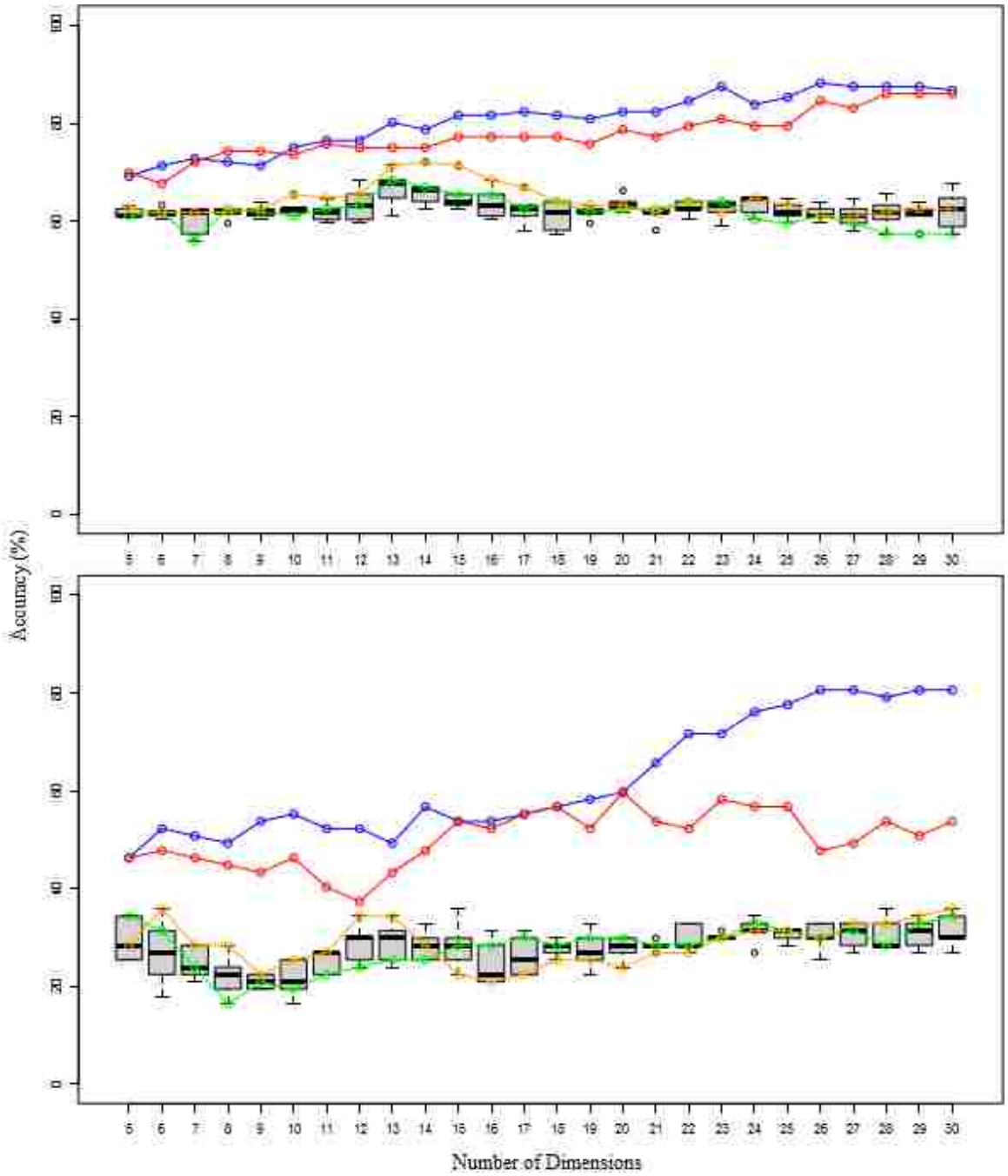

Figure 4.19: Box plot showing cross-validation accuracies when feature selection was performed inside of cross-validation to different numbers of dimensions. The base dataset used was sample set 1 through 3 with no unknown genera with season (top) and treatment (bottom). See Figure 4.1 for more information.

To assist in selecting the number of features, the 3D Pareto frontier scatter plots visualize a multi objective optimization of the warmed vs. natural permafrost microbiome case study's box plots. The dominant points on the Pareto frontier have mathematically demonstrated an ideal balance between good median accuracy and low variance while favoring higher numbers of features.

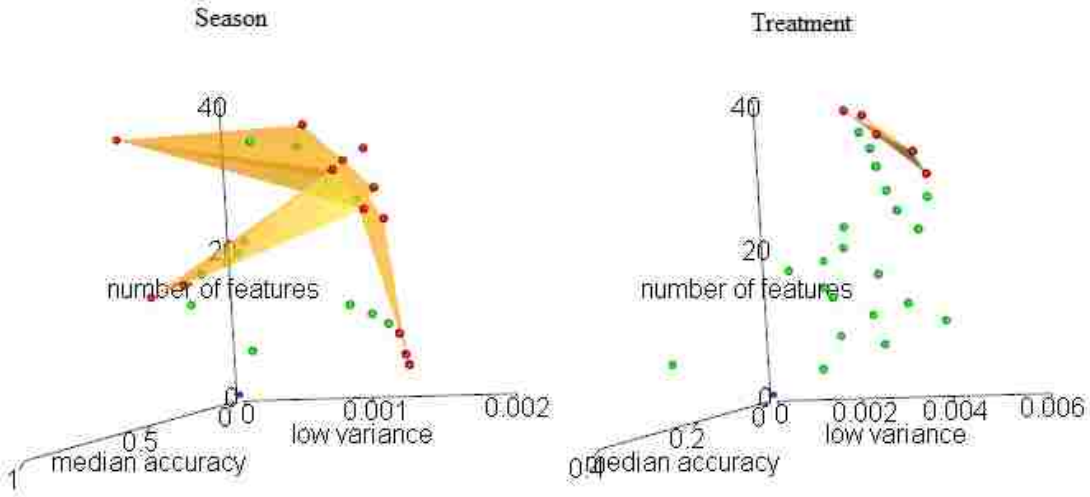

Figure 4.20: Scatter plot visualizing a 3D-Pareto Frontier, optimizing median cross-validation accuracy, low variance, and higher numbers of features. The base dataset used was sample set 1 through 3 with no unknown genera with season (top) and treatment (bottom). See Figure 4.4 for more information.

Tables 4.9 and 4.10 represent the genera identified using the voting process for the LDA plots visualized in Figures 4.21 and 4.22.

Table 4.9: Season - 13 genera identified using the voting process

Genera	*Rank*
Conexibacter	*3.33*
Mucilaginibacter	*3.33*
Burkholderia	*2.49*
Caedibacter	*2.42*
Singulisphaera	*2.41*
Planctomyces	*2.39*
Variovorax	*2.36*
Ferruginibacter	*2.27*
Phaselicystis	*2.04*
Sediminibacterium	*1.97*
Luteibacter	*1.90*
Asticcacaulis	*1.70*
Flavisolibacter	*1.56*

Table 4.10: Treatment - 23 genera identified using the voting process

Genera	*Rank*
Acidovorax	*1.22*
Flavisolibacter	*0.96*
Schlesneria	*0.86*
Streptacidiphilus	*0.82*
Humicoccus	*0.80*
Paenibacillus	*0.77*
Burkholderia	*0.74*
Herbaspirillum	*0.71*
Cystobacter	*0.66*
Phenylobacterium	*0.63*
Streptomyces	*0.59*
Caedibacter	*0.59*
Planctomyces	*0.58*
Gemmatimonas	*0.57*
Chondromyces	*0.55*
Rhodanobacter	*0.52*
Iamia	*0.51*
Gemmata	*0.50*
Pirellula	*0.49*
Actinoallomurus	*0.46*
Ferruginibacter	*0.45*
Pedomicrobium	*0.44*
Luteolibacter	*0.44*

The LDA scatter plots presented below are the final product of the analysis performed in the permafrost microbiome case study. These plots visualize the separation of permafrost microbiomes by season (Figure 4.21) and then treatment (Figure 4.22), using bacteria with genus level resolution.

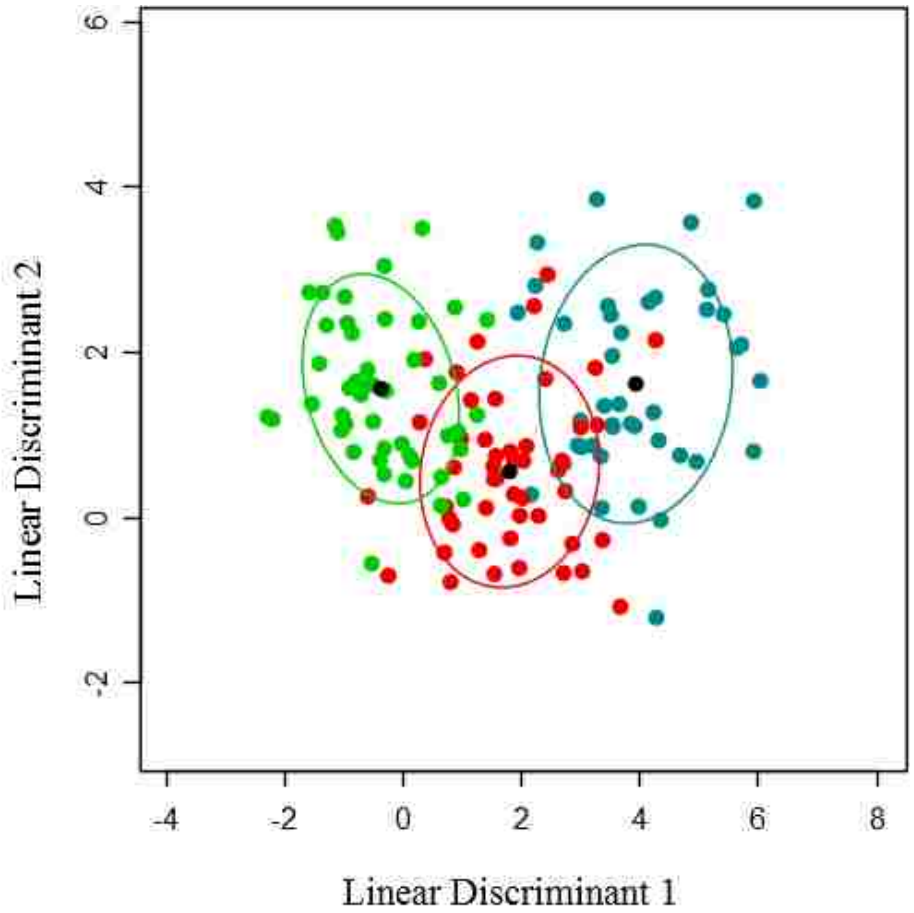

Figure 4.21: Linear Discriminant Analysis (LDA) of the seasonal groups sampled in June, July, and August. Black dots represent the centroid for each cluster and ellipses indicate 1 standard deviation. The accuracies were 75.00% (71.32%). The first accuracy listed used the vote-determined genera (Table 4.9), while the right side accuracy was for genera identified using 'floating search within each cross-validation fold'.

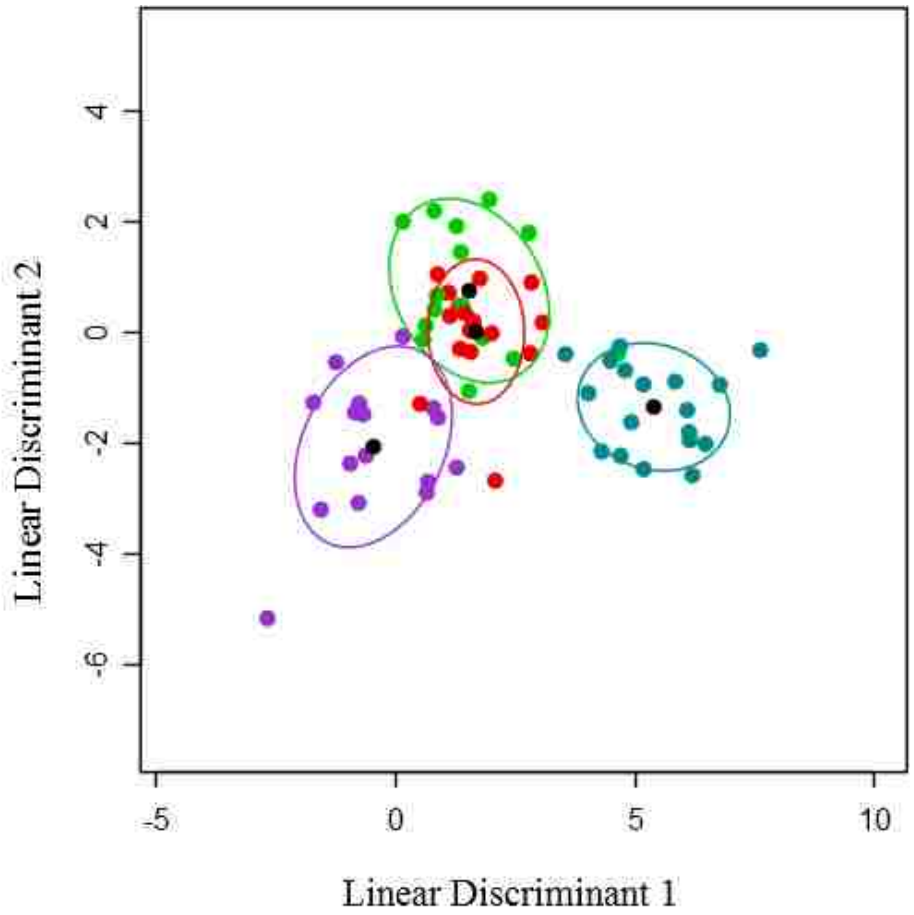

Figure 4.22: Linear Discriminant Analysis (LDA) of the treatment groups: Control, Snow-side Control, Warmed, and Snow-side Warmed. Black dots represent the centroid for each cluster and ellipses indicate 1 standard deviation. The accuracies were 58.21% (29.85%). The first accuracy listed used the vote-determined genera (Table 4.10), while the right side accuracy was for genera identified using 'floating search within each cross-validation fold'.

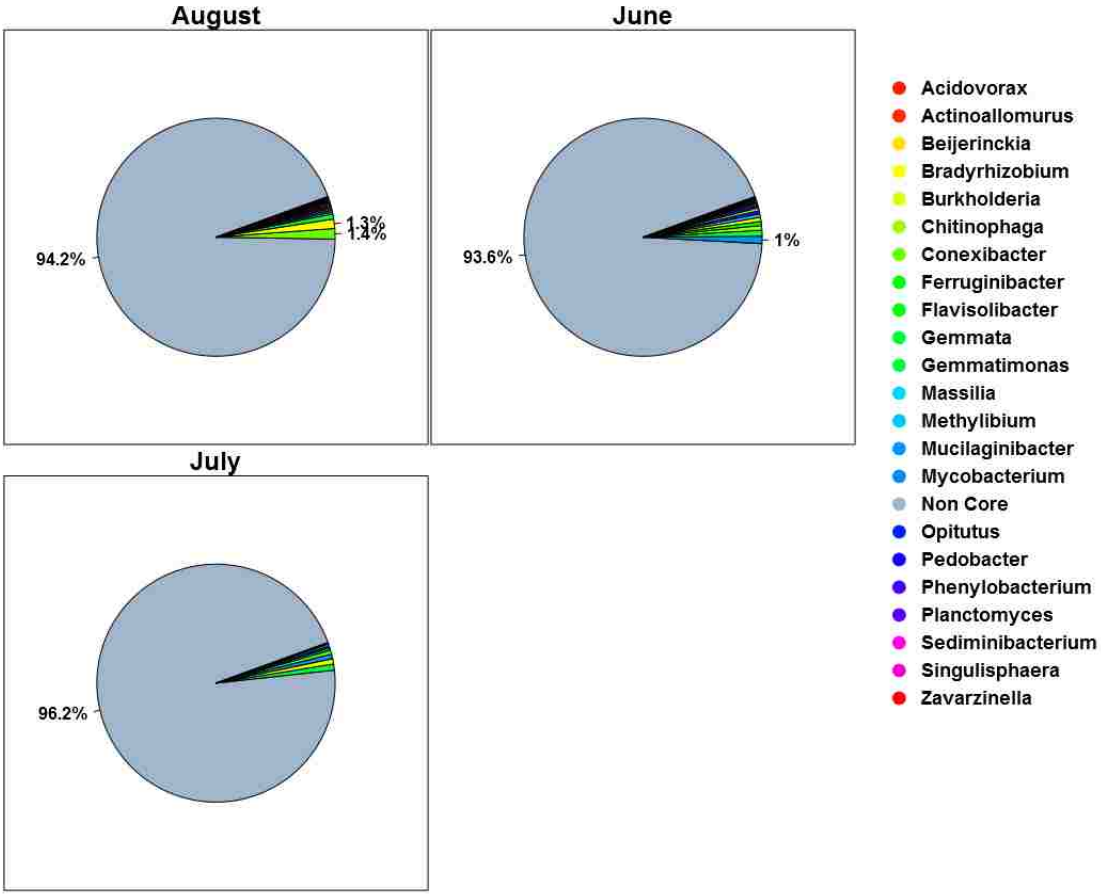

Figure 4.23: Pie charts visualizing the core and non-core microbiome for each seasonal group of the permafrost. Each plot represents the core genera (as described in Section 3.5.1) across all samples for that group using a 0.0% threshold.

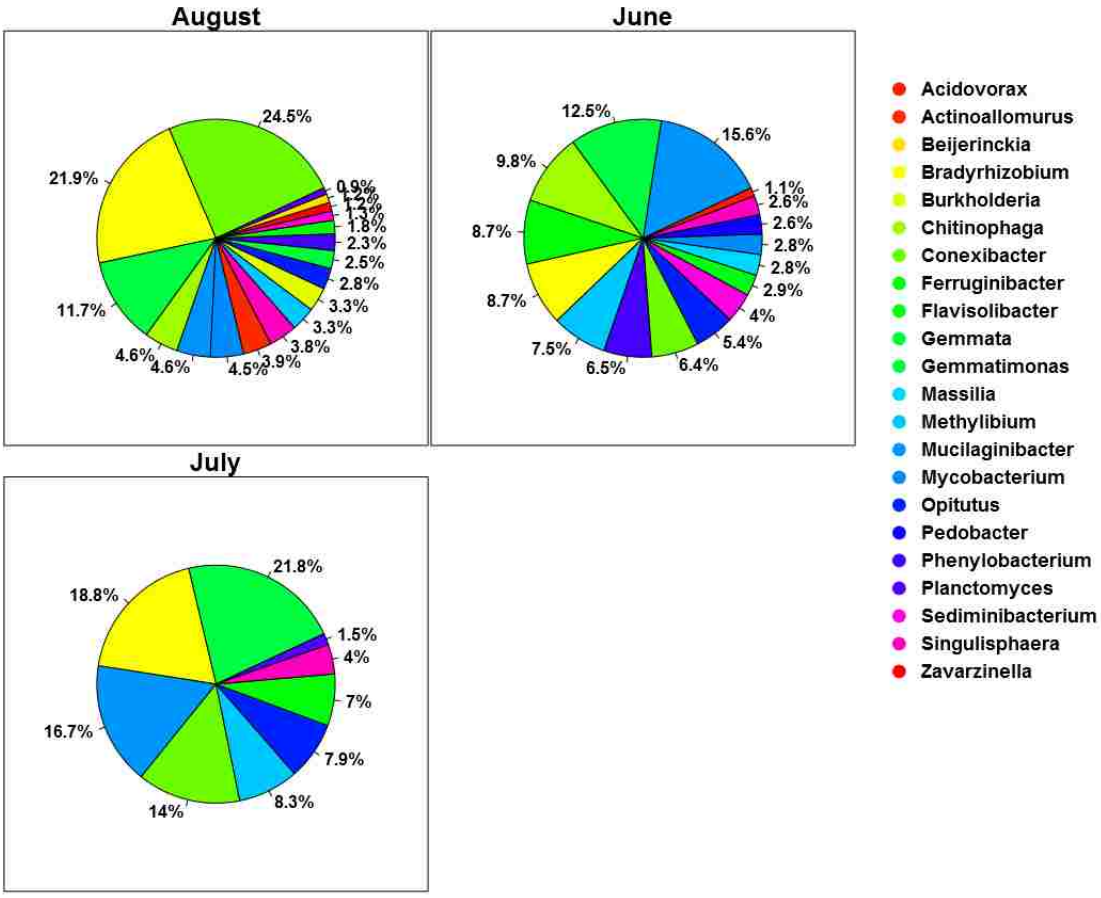

Figure 4.24: Pie charts visualizing the core microbiome for each seasonal group of the permafrost. Each plot represents the core genera (as described in Section 3.5.1) across all samples for that group using a 0.0% threshold.

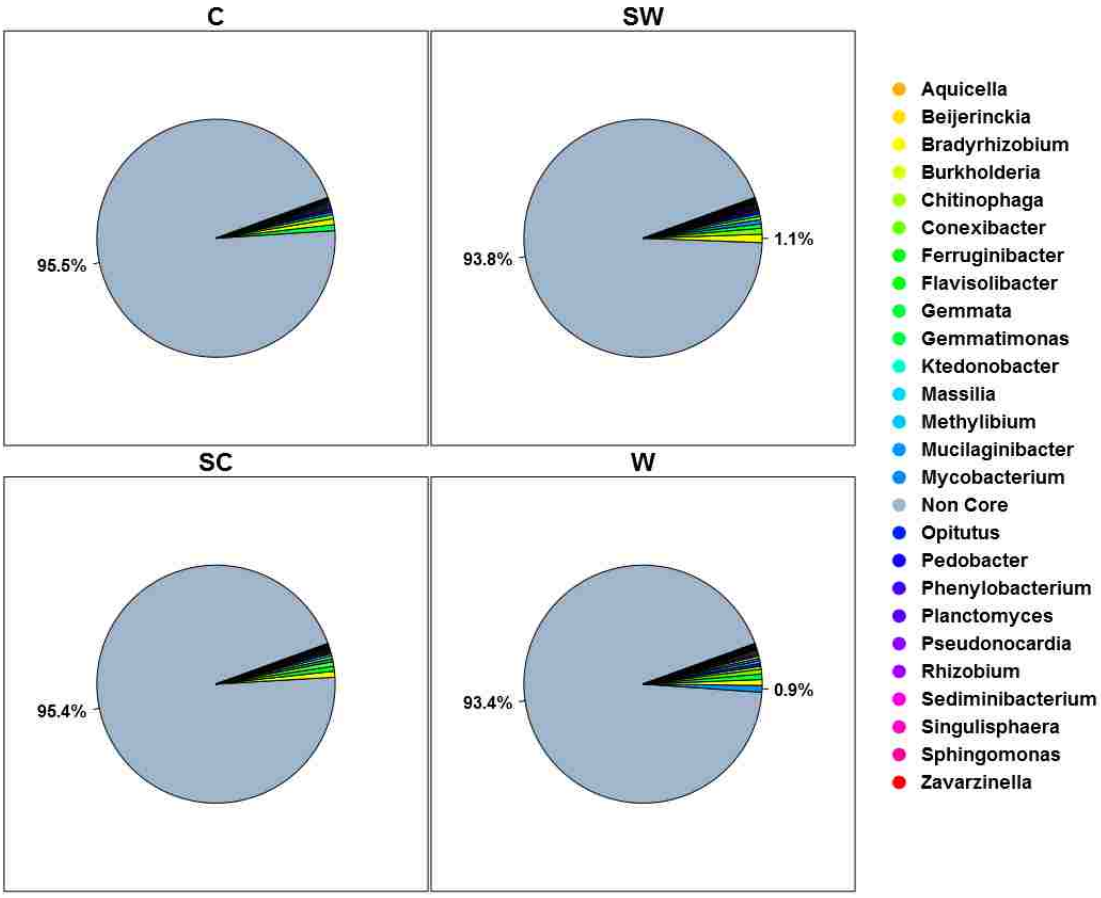

Figure 4.25: Pie charts visualizing the core and non-core microbiome for each treatment of permafrost. Each plot represents the core genera (as described in Section 3.5.1) across all samples for that group using a 0.0% threshold.

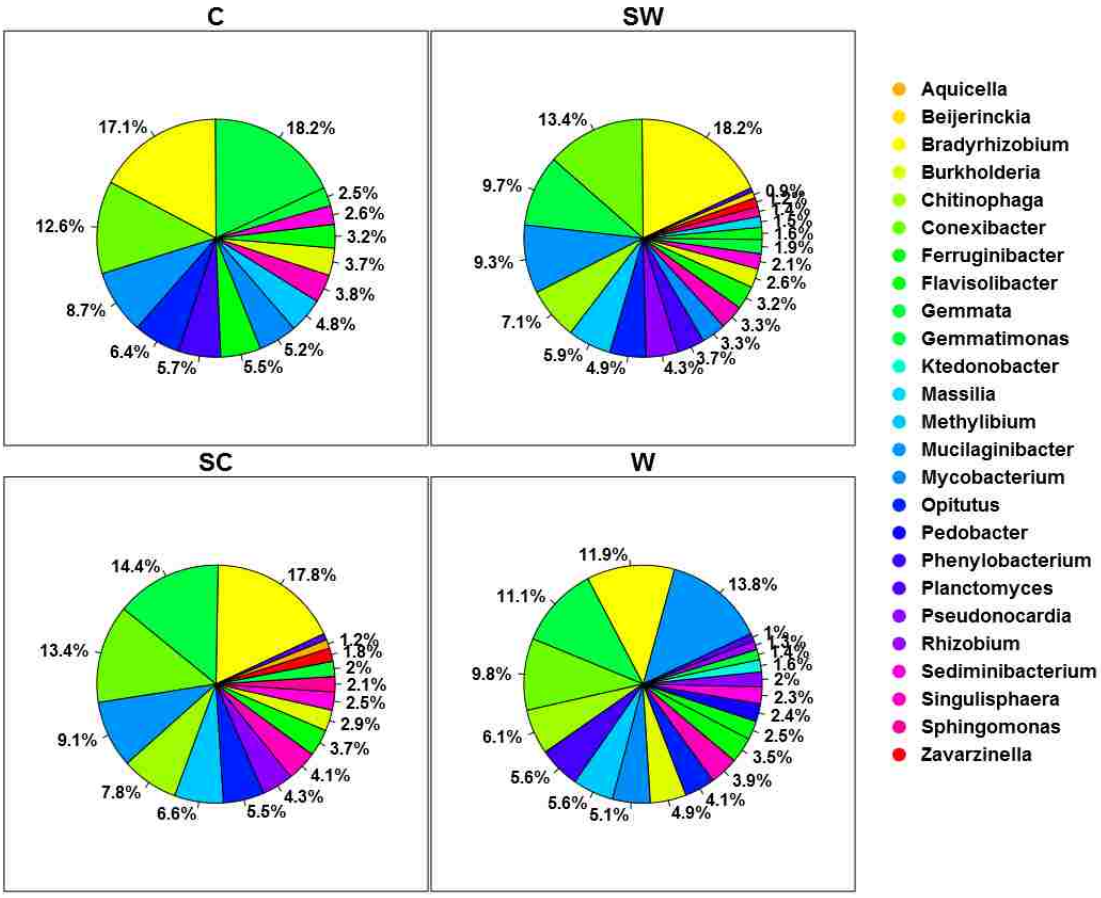

Figure 4.26: Pie charts visualizing the core microbiome for each treatment of permafrost. Each plot represents the core genera (as described in Section 3.5.1) across all samples for that group using a 0.0% threshold.

4.4 Effects Of Yogurt On Mouse Intestinal Microbiomes

The food industry has seen an increase in the sales of probiotic foods because people believe that such products can enhance their health. This case study aimed to show that probiotic bacteria don't actually replace a host's intestinal bacteria, but rather that they perturb the intestinal environment in a way that modifies the total microbiome composition. This is believed to cause shifts in the distribution of bacteria across the intestinal microbiome. Toward this end, ten mice were dissected and sampled from six intestinal locations: ileum, cecum, tip of cecum, proximal colon, mid colon, and distal colon. Five of the mice had been fed yogurt in addition to their normal diet and the other five mice were not fed yogurt.

For this analysis, the data was further split into five matrices with each matrix imposing increasingly strict criteria on the genera. In the framework this is referred to as the pruning levels of a dataset in which genera are only kept if they have nonzero values in at least 1, 3%, 5%, 8%, or 16% of the samples. Next for each matrix, LDA cross-fold-validation was performed with feature selection done on each training set (fold) of the process. Information about which features were used, how often they were used, and how they performed was then collected into a statistics database. This database was then visualized in box and whisker plot for the analysis (Figure 4.27). To further assist the researcher in the difficult task of choosing the best number of features (best box from the box and whisker plot), the framework also performed multi-objective optimization on the box plot's boxes. This information was then saved to a CSV and visualized in a 3D Pareto frontier (Figure 4.28). Using the Pareto frontier an optimal number of features was chosen for the analysis (Table 4.11). With both sets of features in hand, the framework visualized the vote determined genera using LDA (Figure 4.29) and performed LDA cross-validation to ascertain its

confidence in the genera chosen.

Additionally, core microbiomes (Figures 4.31 and 4.30) were identified for each mouse treatment in order to find the most consistently present members (genera) of each microbiome (sampling site). These prominent genera were then compared to the discriminatory genera found earlier. Then, in order to visualize the entire control microbiome vs. the yogurt fed microbiome, a bar plot analysis was done at the Phylum level (Figures 4.32, 4.33) and Family level (Figures 4.34, 4.35) of taxonomic resolution. This analysis was intended to identify specific bacteria or groups of bacteria that warranted further investigation.

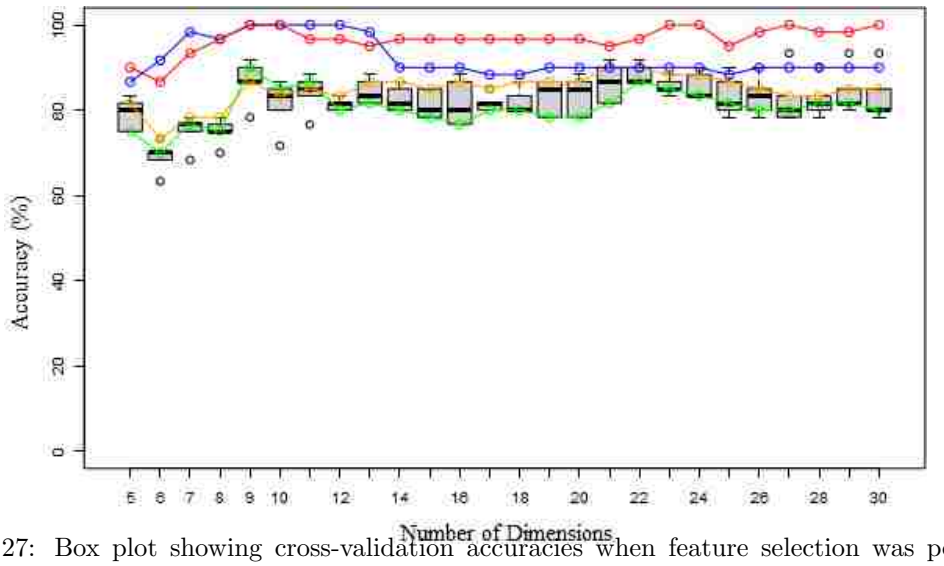

Figure 4.27: Box plot showing cross-validation accuracies when feature selection was performed inside of cross-validation to different numbers of dimensions. The base dataset used was yogurt mice cohort 1 with no unknown genera. See Figure 4.1 for more information.

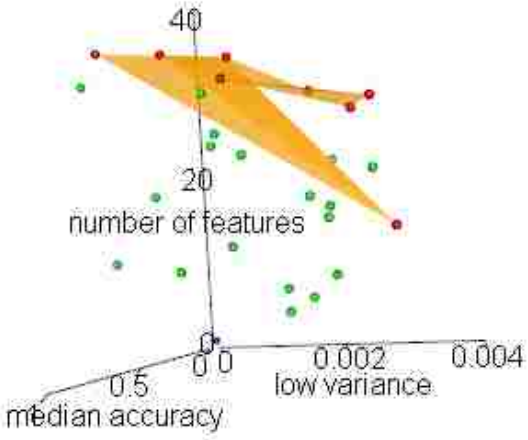

Figure 4.28: Scatter plot visualizing a 3D-Pareto Frontier, optimizing median cross-validation accuracy, low variance, and higher numbers of features. The base dataset used was yogurt mice cohort 1 with no unknown genera. See Figure 4.4 for more information.

Table 4.11 represents the genera identified using the voting process for the LDA plot visualized in Figure 4.29.

Table 4.11: Yogurt Mice - 12 genera identified using the voting process

Genera	*Rank*
Sporacetigenium	*3.99*
Oscillibacter	*3.58*
Dorea	*3.42*
Coprococcus	*3.10*
Paralactobacillus	*2.85*
Lactobacillus	*2.78*
Hydrogenoanaerobacterium	*2.77*
Staphylococcus	*2.77*
Parasutterella	*2.45*
Clostridium	*2.11*
Acetitomaculum	*2.04*
Turicibacter	*1.71*

The LDA scatter plot (Figure 4.29) was created using voted genera to visualize differentiation between control mice and the yogurt fed mice.

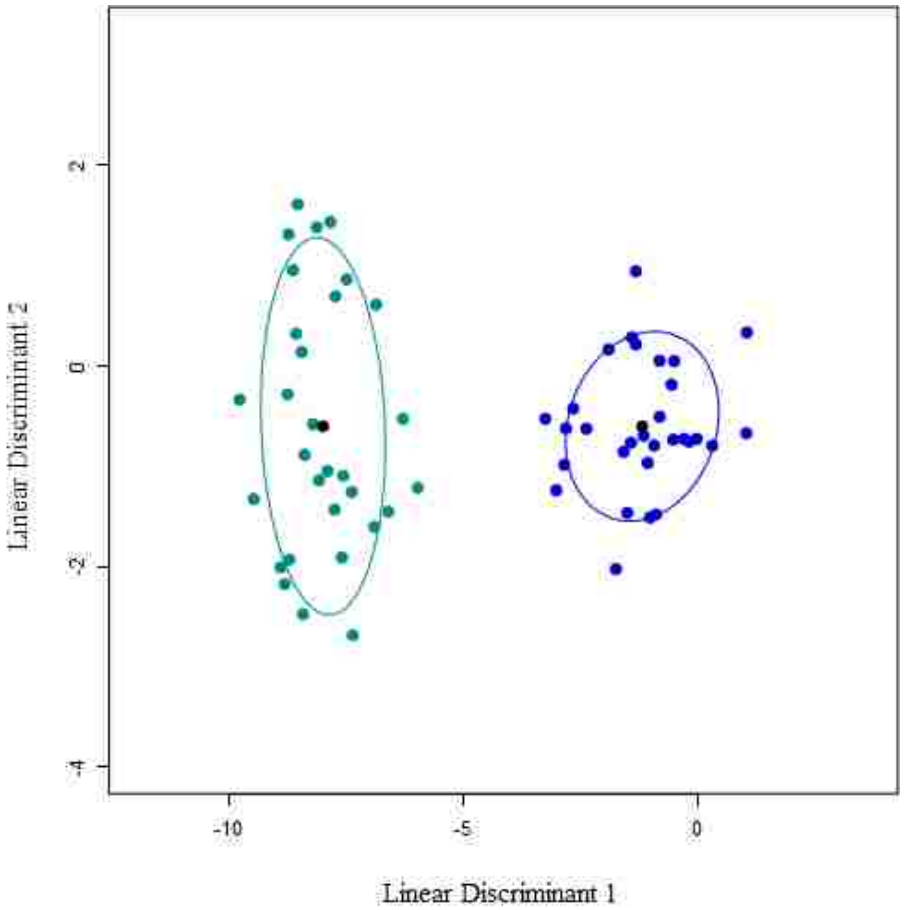

Figure 4.29: Linear Discriminant Analysis (LDA) of the yogurt fed and control mice. Black dots represent the centroid for each cluster and ellipses indicate 1 standard deviation. The accuracies were 96.67% (83.33%). The first accuracy listed used the vote-determined genera (Table 4.11), while the right side accuracy was for genera identified using 'floating search within each cross-validation fold'.

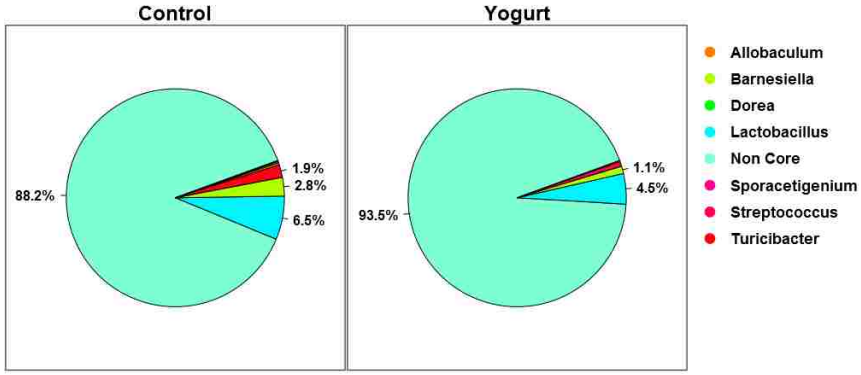

Figure 4.30: Pie chart showing median presence of bacterial genera that are present in every sample of each group. The group of mice labeled control (left) did not receive yogurt. The group of mice labeled yogurt (right) did receive yogurt.

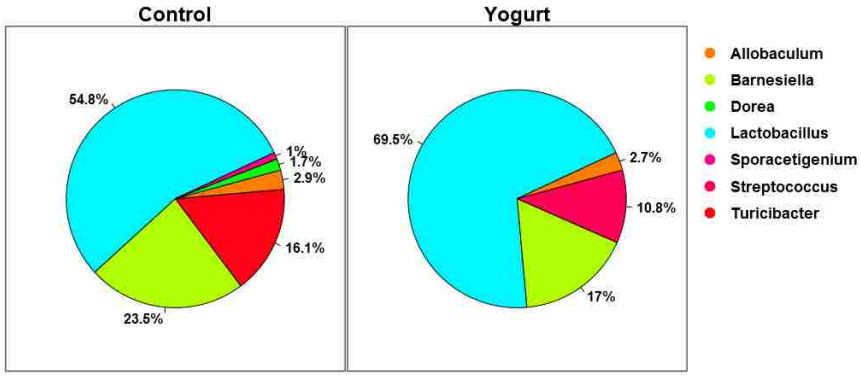

Figure 4.31: Pie chart showing median presence of bacterial genera that are present in every sample of each group. The group of mice labeled control (left) did not receive yogurt. The group of mice labeled yogurt (right) did receive yogurt.

Bar plots visualizing the microbiome of control mice vs. yogurt fed mice at first at Phylum (Figures 4.32, 4.33) and then Family (Figures 4.34, 4.35) level resolution. The bacteria with the greatest presence across all samples are arranged towards the top. Additionally for visualization purposes, the presence of each bacteria is logarithmically smoothed allowing bacteria with high microbiome representation (50% or more) to be seen alongside bacteria with low representation (.1% or less).

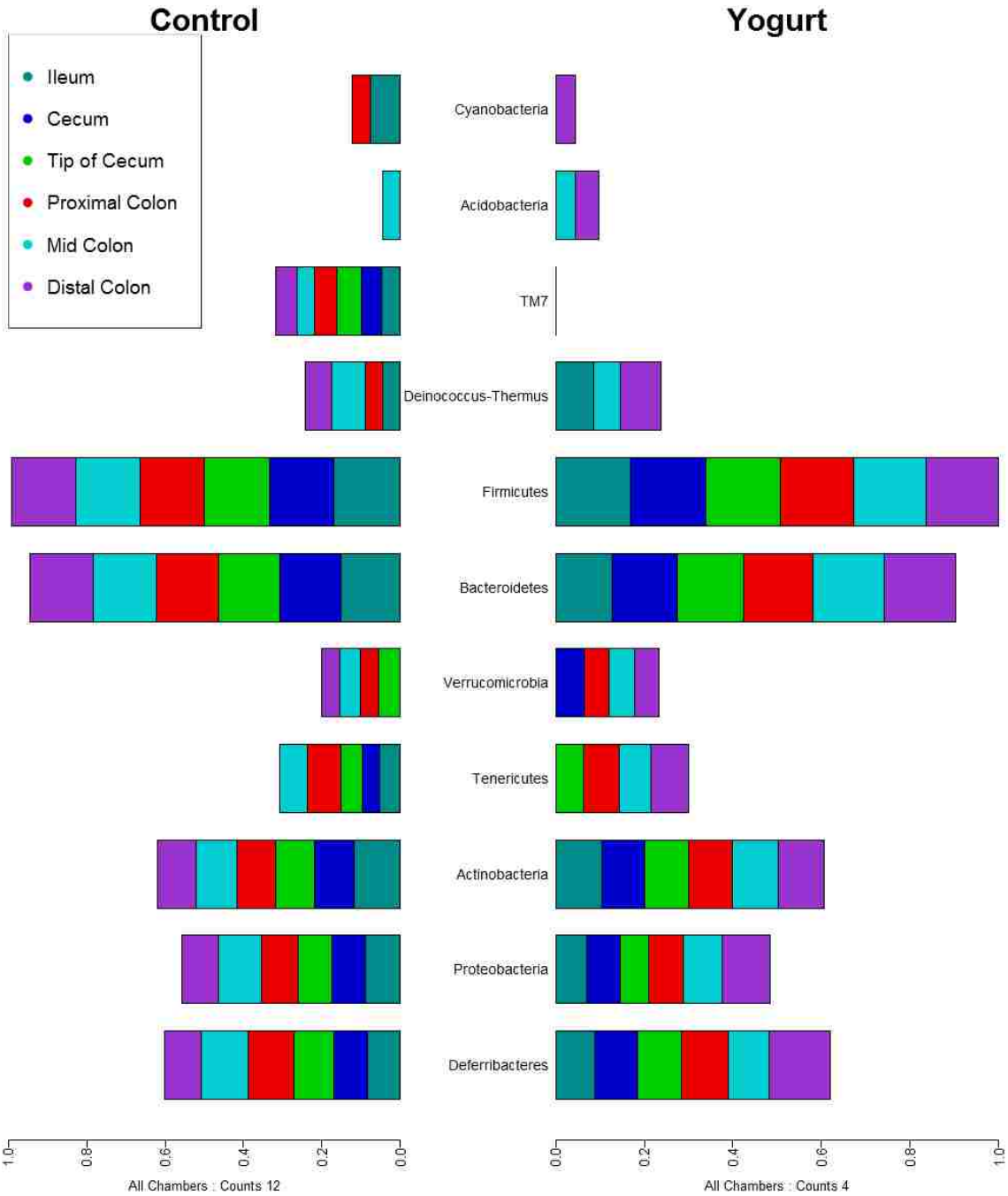

Figure 4.32: Side-by-side bar plot showing average bacterial presence by intestinal chamber (with Phylum level resolution) found in the control (left) mice and the yogurt (right) mice.

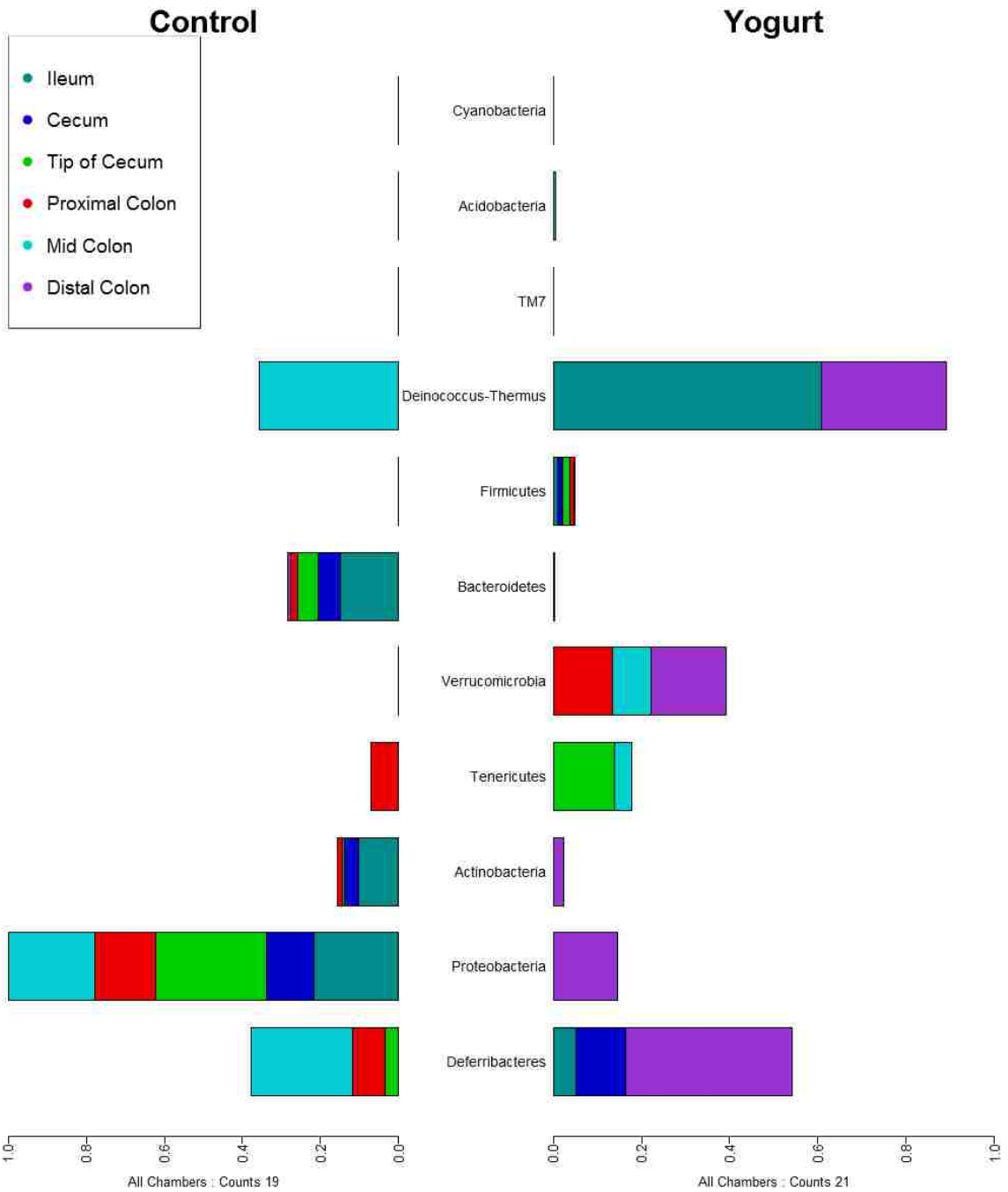

Figure 4.33: Side-by-side bar plot showing a log ratio of average bacterial presence by intestinal chamber (with Phylum level resolution) found in the control (left) mice and the yogurt (right) mice.

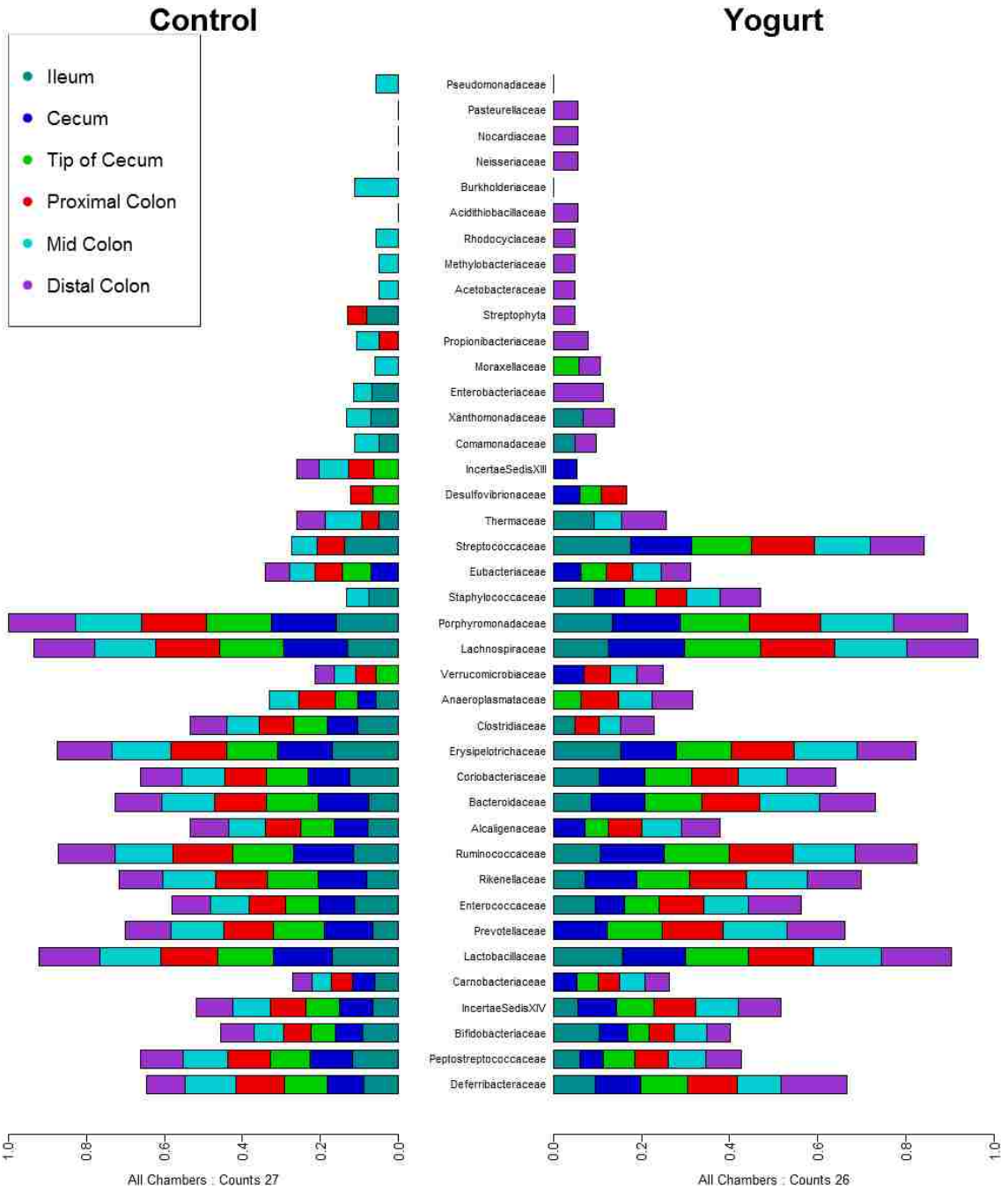

Figure 4.34: Side-by-side bar plot showing average presence by chamber (with Family level resolution) found in the control (left) mice and the yogurt (right) mice.

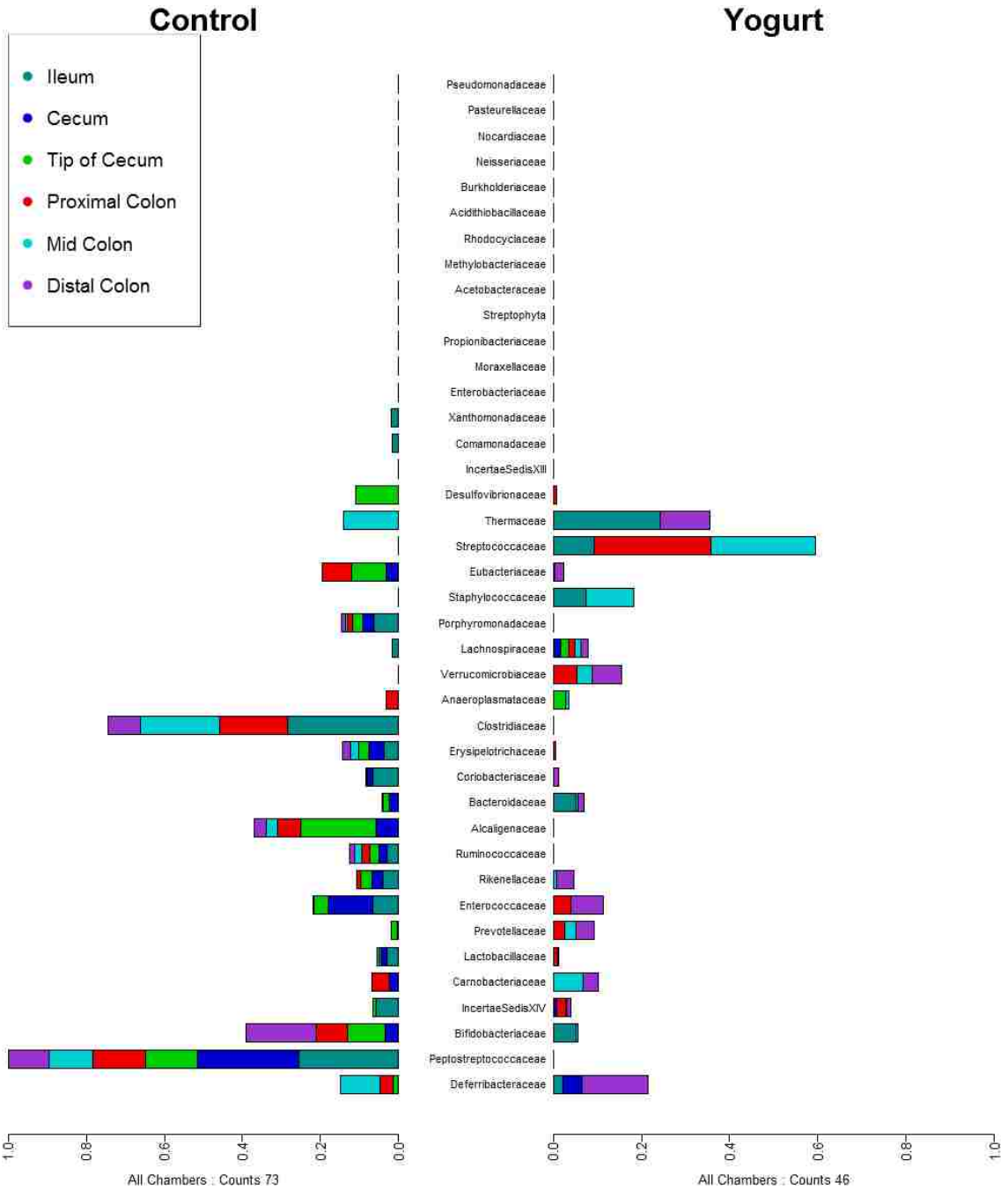

Figure 4.35: Side-by-side bar plot showing a log ratio of average presence by chamber (with Family level resolution) found in the control (left) mice and the yogurt (right) mice.

4.5 DNA Recovery Qiagen vs. MoBio

Researchers today have many different options when selecting methods for the recovery of DNA. In order to compare two of these methods against each other, eight rats were dissected and sampled at three intestinal locations: cecum, proximal colon, and distal colon. This set of 24 samples was then processed using Qiagen and then MoBio DNA extraction kits in turn. The side-by-side bar plots (Figures 4.38 and 4.39) were then created to visualize bacteria identified uniquely by one method or the other. Then in order to visualize the microbiomes from a different perspective, a core microbiome analysis (Figures 4.37 and 4.36) was done to determine which core bacteria are found more readily by one method or the other.

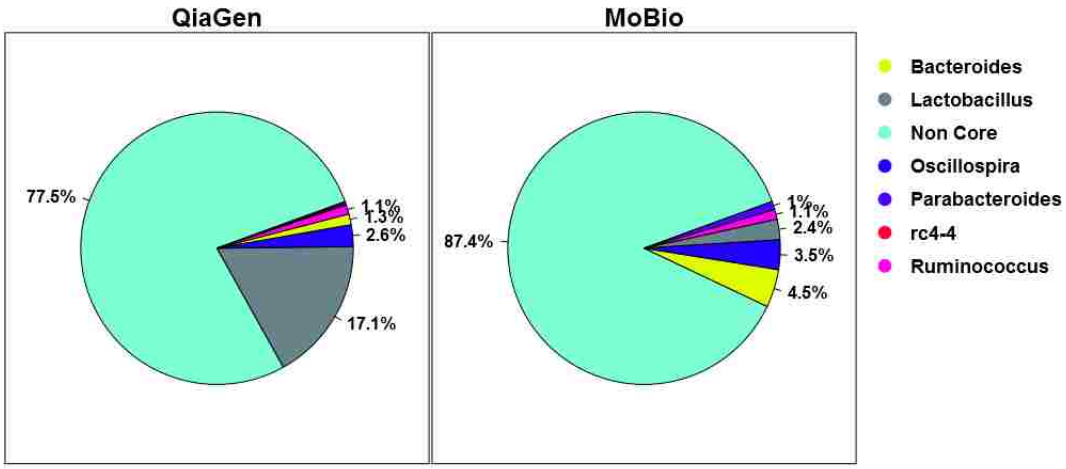

Figure 4.36: Pie chart showing median presence of bacterial genera that are present in every sample of each group. The two DNA recovery methods are Qiagen (left) and MoBio (Right).

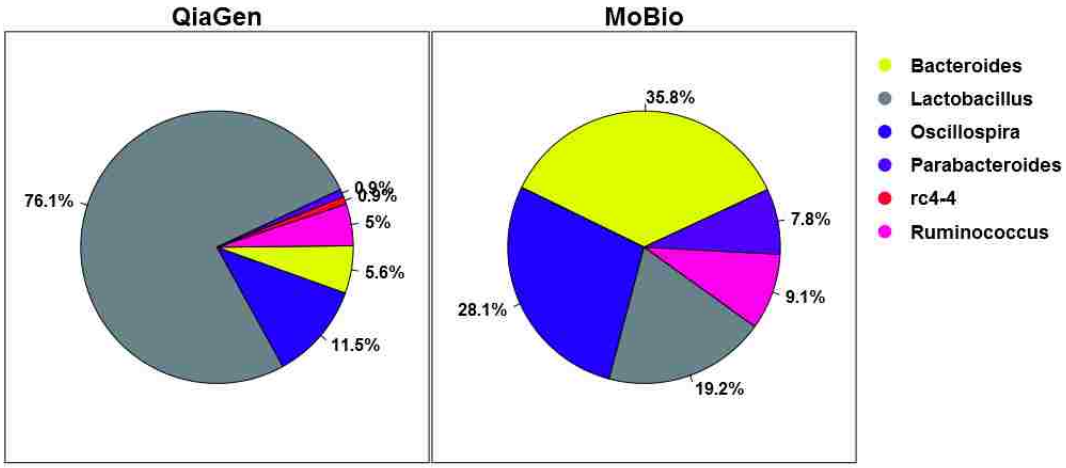

Figure 4.37: Pie chart showing median presence of bacterial genera that are present in every sample of each group. The two DNA recovery methods are Qiagen (left) and MoBio (Right).

Bar plots visualizing rat microbiomes found by Qiagen and MoBio DNA recovery techniques at Family level resolution (Figures 4.38, 4.39). The bacteria with the most presence across all samples are arranged towards the top. Additionally for visualization purposes, the presence of each bacteria is logarithmically smoothed allowing bacteria with high microbiome representation (50% or more) to be seen alongside bacteria with low representation (.1% or less).

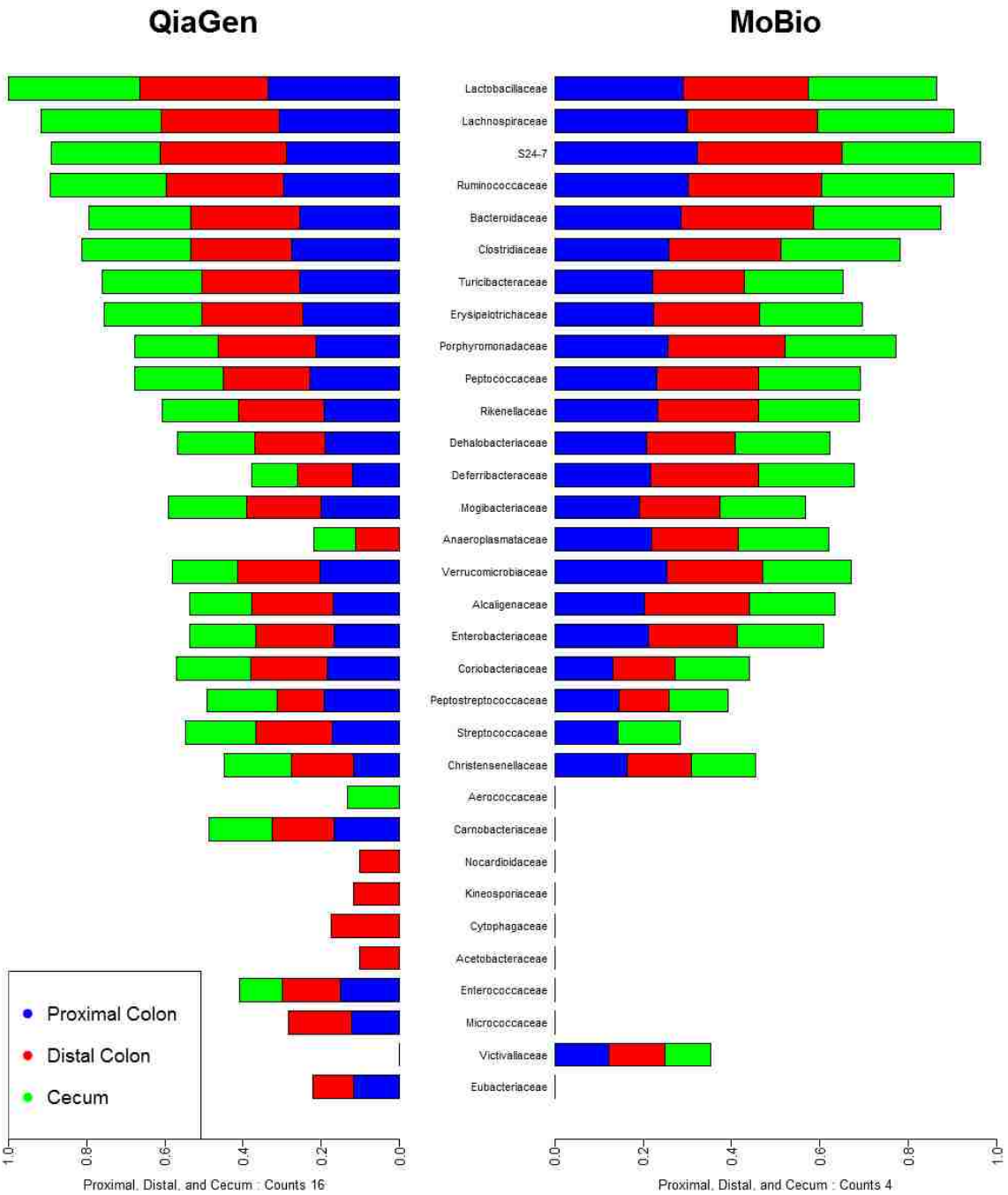

Figure 4.38: Side-by-side bar plot showing average presence by intestinal chamber (with Family level resolution) found by the Qiagen (left) and MoBio (right) methods.

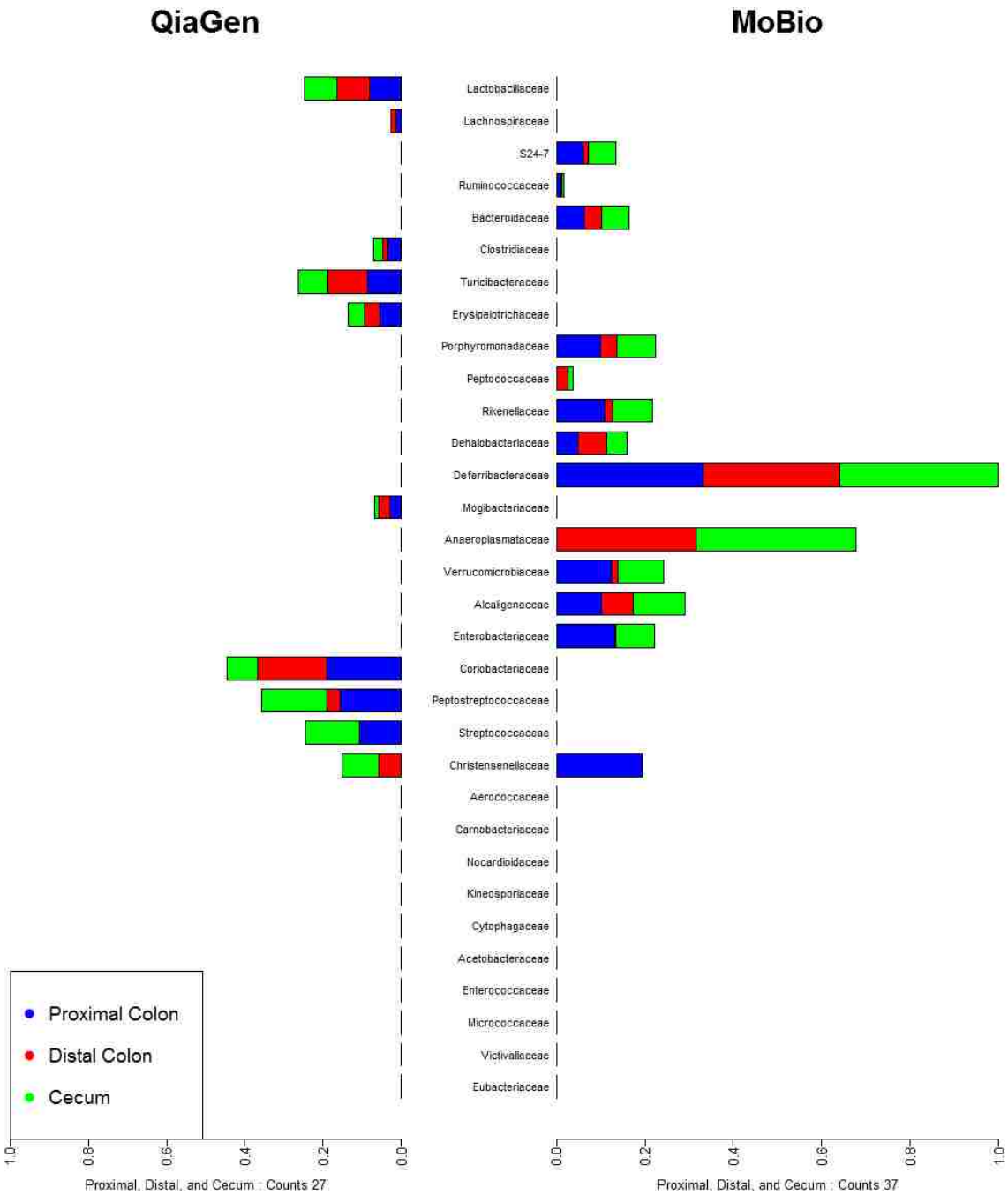

Figure 4.39: Side-by-side bar plot showing a log ratio of average presence by intestinal chamber (with Family level resolution) found by the Qiagen (left) and MoBio (right) methods.

CHAPTER 5 DISCUSSION

The Data Analysis and Visualization for Bioinformatics Framework allows researchers to ask and answer questions about the bacterial communities that other studies have been unable to address due to the complicating inter-subject variation present in microbiome analyses [7] [8] [9].

5.1 Discriminating Between Microbiomes

In several case studies, the framework was successful in identifying bacterial genera that discriminate between microbiomes. These tables can be seen in chapter 4 in the appropriate case study section (Tables 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, and 4.11).

In order to find the discriminatory genera, each case study went through a similar set of analyses using the framework developed for these studies. This involved performing feature selection inside of cross-validation, and then visualizing this information in box and whisker plots (Figures 4.1, 4.2, 4.3, 4.11, 4.19, and 4.27). From the box plots, multi-objective analysis was done and visualized in 3D-Pareto frontiers, which allowed the researcher to consider only the numbers of features that performed the best (Figures 4.4, 4.5, 4.6, 4.12, 4.20, and 4.28). Then using these discriminatory genera, the separation between the microbiomes was visualized in LDA scatter-plots (Figures 4.7, 4.8, 4.13, 4.14, 4.21, 4.22, and 4.29).

The cross-validation numbers presented in the LDA scatter plot captions are representative of the framework's confidence in its ability to assign future samples to the correct microbiome. The two cross-validation accuracies displayed on the LDA scatter plot images refer to first the LDA classifier's performance using the most discriminatory taxa identified by the box plot and Pareto frontier analysis. The second accuracy is the cross-validation accuracy achieved when a floating search was used on each fold to identify a number of genera inside of LDA cross-fold-validation.

5.2 Visualizing Core Microbiome Members

In each case study, the core members (genera) of each microbiome were identified using the method described in Section 3.5.1 (Figures 4.10, 4.9, 4.18, 4.17, 4.16, 4.15, 4.24, 4.23, 4.26, 4.25, 4.31, 4.30, 4.37, and 4.36). The core microbiome is comprised of the most consistently present members of a microbiome at each sampling site, which allows researchers to identify which bacterial genera may warrant further study. These core members were also found to often overlap with the discriminatory genera used to visualize microbiome separation (LDA scatter plots). This a good double check of the significance of these bacteria because the core analysis is completely independent from the discriminatory genera analysis.

5.3 Visualizing Microbiomes Side-by-side

In some studies it was useful to visually compare one microbiome to another (Figures 4.32, 4.34, and 4.38). This bird's-eye view allowed the researchers to easily see which bacteria were most present and where, but did not clearly display where one microbiome had more of a given bacterium than the other. In order to address this

problem, log ratio bar plots were created, showing only the differences between the microbiomes (Figures 4.33, 4.35, and 4.39).

This kind of bar plot visualization can be done at multiple levels phylogenetic resolution. For example, QIIME produces OTU tables for phylum, class, order, family, and genus level resolution. Additionally, the Seqmatch RDP database provides OTU resolution to the species level.

If a microbiome has too many members to be clearly visualized with this method, side-by-side bar plots could be combined with core microbiome analysis. Combining algorithms like this would allow researchers to get a picture of the distribution of the most significant organisms in the microbiome.

5.4 Biological Conclusions From Case Studies

Each case study done with this framework had different goals. Surprisingly, the same set of algorithms was able to address these varied research questions.

5.4.1 Separation Between Mouse Intestinal Microbiomes

During the longitudinal mouse study, the framework was able to show clear separation between the microbiomes of the six intestinal sample sites. As expected, the strain B (inbred) mice showed less inter-subject variation than the strain C (outbred) mice. This was an anticipated result, and it was nice to see the results meet the expectations in this way (Figures 4.7 and 4.8).

The reason for performing analysis on three sub-sets of the six sampling locations was to observe not just the microbial interaction on the system as a whole, but to observe the microbial interactions in more nuanced scenarios. For example, in the early intestinal chambers *Lactobacillus* was one of the most significant bacterium,

but later in the colon *Dorea* and *Bacteroides* became more important for telling chambers apart. Observations like this may someday lead to a greater understanding of intestinal microbiology and its impact on our health.

5.4.2 How Elk Fecal Microbiomes Vary For Body Fat

Recently there have been many studies of the fecal microbiome concerning obesity and diversity [60] [61] [62]. At the Holben Lab, the framework was used to analyze data recovered from elk fecal pellets in four Montana areas near Missoula: Sapphire, Blacks Ford, Tobacco Root, and Bitterroot. The majority of the elk sampled were pregnant females, but there were also males and non-pregnant females. Due to the lack of body fat information for the Bitterroot elk, those elk were excluded from the body-fat analysis.

The framework was able to determine discriminatory bacterial genera separating the elk's fecal microbiomes by region with a high degree of confidence (Figure 4.14). Additionally, it was able to discriminate between the microbiomes across all regions as a function of body-fat (Figure 4.13); however, even with moderately significant results the framework's confidence was much lower for the body-fat than the regional results.

These results suggest a novel method for identifying animal condition through the host fecal microbiome. This bioinformatics approach could presumably be conducted on non-invasive samples in the future, representing a relatively inexpensive information source. Wildlife managers could then use this information when facing the challenges of protecting threatened and endangered species or maintaining game species. Therefore, the potential applications of this research are far reaching and may, with further refinement, represent a significant tool for conservation in the future.

5.4.3 How Warming And Season Effect The Soil Microbiome

The microbiome of permafrost samples is particularly interesting in its effect on our understanding of climate change [59]. Even recognizing that soil and permafrost microbiomes are particularly difficult to analyze due to their massive diversity, the framework was used at the Holben Lab in an attempt to identify discriminatory bacterial genera in permafrost soil samples between different treatments (environmental conditions) (Figure 4.22) in Greenland. I found that the seasonal changes in the permafrost microbiome were so overwhelming that discrimination between permafrost treatments was only moderately successful. On the other hand, the framework had much more success discriminating between seasons (Figure 4.21). This leads me to believe that a significant global warming event could change the microbial communities in the worlds soil and permafrost in a profound way.

5.4.4 Effects Of Yogurt On The Mouse Intestinal Microbiome

In one of the framework's more confident results, the microbiome of the yogurt fed mice was shown in Figure 4.29 to be to be almost completely differentiable from the microbiome of the control mice. In this study, ten mice were sampled in six intestinal locations each. Five of the mice were fed yogurt and five were not.

Microbiome presence was also visualized for the yogurt mice case study (Figures 4.34, 4.35, 4.32, and 4.33). This was done to visually identify bacteria there were present or more present in either the control mice or the yogurt fed mice. This analysis led to a list of genera that warranted further investigation. Being a preliminary experiment, further analyses were beyond the scope of this work, but could someday be incorporated in the framework's libraries.

5.4.5 DNA Recovery Qiagen Vs MoBio

In the DNA recovery methods case study two different procedures were used on the same set of samples. These results were then compared (MoBio vs. Qiagen) across their microbiomes (Figures 4.38 4.39). When looking across the family resolution microbiomes as a whole, the Qiagen extraction method allowed detection of genera in 12 sample locations where they were not found by MoBio. Meanwhile, MoBio allowed detection of genera in 1 sample location where it wasn't detected by the Qiagen method.

5.5 Conclusions

This framework provides many functions for computational and visual data analysis which can be used by researchers to better understand metagenomic and other multivariate data. I have proven, in several case studies, that bacterial taxa can be found and used to discriminate one microbiome from another with reasonable confidence. In addition, these results were corroborated by identifying core microbiome members and explored by visualizing differences between microbiomes.

5.6 Future Directions

In the future, this framework could become much more accessible and useful to researchers given a web-based graphical user interface. A feature like this would allow researchers to apply these methods to their data without having to program in the Perl and R languages. Furthermore, a Graphical User Interface (GUI) would be a convenience even to veteran programmers because they would have no need to learn the inner workings of this framework.

Furthermore, a significant computational bottleneck in the framework's methodology is the the R language implementation of the Floating Search algorithm. The computational process could be made considerably faster by refactoring this function and others into C or C++. These faster and more efficient implementations could then be brought into the framework's R code as external libraries via the Rcpp package. Potentially, this could change computational time (in some cases) from weeks to hours.

In order to improve the maintainability and assist with the implementation of future functionality, the framework as a whole could benefit from a set of test modules. These modules would confirm the correctness of various algorithms and functions throughout the framework, allowing developers to know when new functionality might not be well integrated with the framework as a whole.

Lastly with newer versions of the R language, better programming styles are being supported. If this framework was to receive significant work in the future, much of it would benefit from being rewritten to take advantage of the language changes provided by newer implementations of the R language.

BIBLIOGRAPHY

[1] J. G. Caporaso, J. Kuczynski, J. Stombaugh, K. Bittinger, F. D. Bushman, E. K. Costello, N. Fierer, A. G. Pena, J. K. Goodrich, J. I. Gordon, G. A. Huttley, S. T. Kelley, D. Knights, J. E. Koenig, R. E. Ley, C. A. Lozupone, D. McDonald, B. D. Muegge, M. Pirrung, J. Reeder, J. R. Sevinsky, P. J. Turnbaugh, W. A. Walters, J. Widmann, T. Yatsunenko, J. Zaneveld, and R. Knight, "QIIME allows analysis of high-throughput community sequencing data," *Nat Meth*, vol. 7, no. 5, pp. 335–336, May 2010. [Online]. Available: http://dx.doi.org/10.1038/nmeth.f.303

[2] C. Lozupone and R. Knight, "Unifrac: a new phylogenetic method for comparing microbial communities," *Applied and environmental microbiology*, vol. 71, no. 12, pp. 8228–8235, 2005.

[3] I. Harrison, M. Laverty, and E. Sterling, "Alpha, beta, and gamma diversity," *Connexions*, 2004.

[4] X. C. Morgan and C. Huttenhower, "Chapter 12: Human microbiome analysis," *PLoS Comput Biol*, vol. 8, no. 12, p. e1002808, 12 2012. [Online]. Available: http://dx.doi.org/10.1371%2Fjournal.pcbi.1002808

[5] L. V. Hooper, D. R. Littman, and A. J. Macpherson, "Interactions between the

microbiota and the immune system," *Science*, vol. 336, no. 6086, pp. 1268–1273, 2012.

[6] I. Cho and M. J. Blaser, "The human microbiome: at the interface of health and disease," *Nature Reviews Genetics*, vol. 13, no. 4, pp. 260–270, 2012.

[7] A. Lavelle, G. Lennon, N. Docherty, A. Balfe, H. E. Mulcahy, G. Doherty, O. Diarmuid, J. M. Hyland, F. Shanahan, K. Sheahan *et al.*, "Depth-dependent differences in community structure of the human colonic microbiota in health," 2013.

[8] G. Rogers, J. Kozlowska, J. Keeble, K. Metcalfe, M. Fao, S. Dowd, A. Mason, M. McGuckin, and K. Bruce, "Functional divergence in gastrointestinal microbiota in physically-separated genetically identical mice," *Scientific reports*, vol. 4, 2014.

[9] P. B. Eckburg, E. M. Bik, C. N. Bernstein, E. Purdom, L. Dethlefsen, M. Sargent, S. R. Gill, K. E. Nelson, and D. A. Relman, "Diversity of the human intestinal microbial flora," *science*, vol. 308, no. 5728, pp. 1635–1638, 2005.

[10] T. J. Treangen, S. Koren, D. D. Sommer, B. Liu, I. Astrovskaya, B. Ondov, A. E. Darling, A. M. Phillippy, and M. Pop, "Metamos: a modular and open source metagenomic assembly and analysis pipeline," *Genome Biol*, vol. 14, no. 1, p. R2, 2013.

[11] B. D. Ondov, N. H. Bergman, and A. M. Phillippy, "Interactive metagenomic visualization in a web browser," *BMC bioinformatics*, vol. 12, no. 1, p. 385, 2011.

[12] R. H. Whittaker, "Vegetation of the siskiyou mountains, oregon and california," *Ecological monographs*, vol. 30, no. 3, pp. 279–338, 1960.

[13] F. Meyer, D. Paarmann, M. D'Souza, R. Olson, E. M. Glass, M. Kubal, T. Paczian, A. Rodriguez, R. Stevens, A. Wilke *et al.*, "The metagenomics rast server–a public resource for the automatic phylogenetic and functional analysis of metagenomes," *BMC bioinformatics*, vol. 9, no. 1, p. 386, 2008.

[14] Y. Vázquez-Baeza, M. Pirrung, A. Gonzalez, and R. Knight, "Emperor: a tool for visualizing high-throughput microbial community data," *Structure*, vol. 585, p. 20, 2013.

[15] P. J. McMurdie and S. Holmes, "phyloseq: an r package for reproducible interactive analysis and graphics of microbiome census data," 2013.

[16] P. S. La Rosa, J. P. Brooks, E. Deych, E. L. Boone, D. J. Edwards, Q. Wang, E. Sodergren, G. Weinstock, and W. D. Shannon, "Hypothesis testing and power calculations for taxonomic-based human microbiome data," 2012.

[17] J. Chen, K. Bittinger, E. S. Charlson, C. Hoffmann, J. Lewis, G. D. Wu, R. G. Collman, F. D. Bushman, and H. Li, "Associating microbiome composition with environmental covariates using generalized unifrac distances," *Bioinformatics*, vol. 28, no. 16, pp. 2106–2113, 2012.

[18] M. N. Price, P. S. Dehal, A. P. Arkin *et al.*, "Fasttree 2–approximately maximum-likelihood trees for large alignments," *PloS one*, vol. 5, no. 3, p. e9490, 2010.

[19] W. S. Torgerson, *Theory and Methods of Scaling*. Wiley, Jan. 1958, published: Hardcover. [Online]. Available: http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0471879452

[20] P. La Rosa, E. Deych, B. Shands, and W. Shannon, "Hmp: Hypothesis testing and power calculations for comparing metagenomic samples from hmp," 2011.

[21] D. McDonald, J. C. Clemente, J. Kuczynski, J. R. Rideout, J. Stombaugh, D. Wendel, A. Wilke, S. Huse, J. Hufnagle, F. Meyer *et al.*, "The biological observation matrix (biom) format or: how i learned to stop worrying and love the ome-ome," *GigaScience*, vol. 1, no. 1, p. 7, 2012.

[22] R. C. Edgar, "Search and clustering orders of magnitude faster than blast," *Bioinformatics*, vol. 26, no. 19, pp. 2460–2461, 2010.

[23] K. T. Konstantinidis and J. M. Tiedje, "Genomic insights that advance the species definition for prokaryotes," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 7, pp. 2567–2572, 2005.

[24] Qiagen, "The QIAGEN Guide to GoodMicrobiological Practice," *QIAGEN News*, no. 5, pp. 21–23, 1998. [Online]. Available: http://www.qiagen.com/literature/qiagennews/0598/985theqi.pdf

[25] W. R. Pearson and D. J. Lipman, "Improved tools for biological sequence comparison," *Proceedings of the National Academy of Sciences*, vol. 85, no. 8, pp. 2444–2448, 1988.

[26] J. G. Caporaso, K. Bittinger, F. D. Bushman, T. Z. DeSantis, G. L. Andersen, and R. Knight, "Pynast: a flexible tool for aligning sequences to a template alignment," *Bioinformatics*, vol. 26, no. 2, pp. 266–267, 2010.

[27] T. Z. DeSantis, P. Hugenholtz, N. Larsen, M. Rojas, E. L. Brodie, K. Keller, T. Huber, D. Dalevi, P. Hu, and G. L. Andersen, "Greengenes, a chimera-checked 16s rrna gene database and workbench compatible with arb," *Applied and environmental microbiology*, vol. 72, no. 7, pp. 5069–5072, 2006.

[28] R. C. Edgar, B. J. Haas, J. C. Clemente, C. Quince, and R. Knight, "Uchime improves sensitivity and speed of chimera detection," *Bioinformatics*, vol. 27, no. 16, pp. 2194–2200, 2011.

[29] Q. Wang, G. M. Garrity, J. M. Tiedje, and J. R. Cole, "Naive bayesian classifier for rapid assignment of rrna sequences into the new bacterial taxonomy," *Applied and environmental microbiology*, vol. 73, no. 16, pp. 5261–5267, 2007.

[30] D. McDonald, M. N. Price, J. Goodrich, E. P. Nawrocki, T. Z. DeSantis, A. Probst, G. L. Andersen, R. Knight, and P. Hugenholtz, "An improved greengenes taxonomy with explicit ranks for ecological and evolutionary analyses of bacteria and archaea," *The ISME journal*, vol. 6, no. 3, pp. 610–618, 2012.

[31] J. J. Werner, O. Koren, P. Hugenholtz, T. Z. DeSantis, W. A. Walters, J. G. Caporaso, L. T. Angenent, R. Knight, and R. E. Ley, "Impact of training sets on classification of high-throughput bacterial 16s rrna gene surveys," *The ISME journal*, vol. 6, no. 1, pp. 94–103, 2012.

[32] D. Box, D. Ehnebuske, G. Kakivaya, A. Layman, N. Mendelsohn, H. F. Nielsen, S. Thatte, and D. Winer, "Simple object access protocol (soap) 1.1," 2000.

[33] J. Watson, "Virtualbox: bits and bytes masquerading as machines," *Linux Journal*, vol. 2008, no. 166, p. 1, 2008.

[34] J. L. C. M. Donald Stufft, Marcus Smith *et al.* (2008, october) The pypa recommended tool for installing python packages. [Online]. Available: https://pypi.python.org/pypi/pip

[35] J. Walter and R. Ley, "The human gut microbiome: ecology and recent evolutionary changes," *Annual review of microbiology*, vol. 65, pp. 411–429, 2011.

[36] J. R. Cole, Q. Wang, J. A. Fish, B. Chai, D. M. McGarrell, Y. Sun, C. T. Brown, A. Porras-Alfaro, C. R. Kuske, and J. M. Tiedje, "Ribosomal database project: data and tools for high throughput rrna analysis," *Nucleic acids research*, p. gkt1244, 2013.

[37] K. Hornik, "R FAQ," 2015. [Online]. Available: http://CRAN.R-project.org/doc/FAQ/R-FAQ.html

[38] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2015. [Online]. Available: http://www.R-project.org

[39] D. Adler, O. Nenadic, and W. Zucchini, "Rgl: A r-library for 3d visualization with opengl," in *Proceedings of the 35th Symposium of the Interface: Computing Science and Statistics*, 2003.

[40] W. N. Venables and B. D. Ripley, *Modern Applied Statistics with S*, 4th ed. New York: Springer, 2002, iSBN 0-387-95457-0. [Online]. Available: http://www.stats.ox.ac.uk/pub/MASS4

[41] D. Eddelbuettel and R. François, "Rcpp: Seamless R and C++ integration," *Journal of Statistical Software*, vol. 40, no. 8, pp. 1–18, 2011. [Online]. Available: http://www.jstatsoft.org/v40/i08/

[42] P. J. McMurdie and the biom-format team http://biom format.org/, "biom: An interface package (beta) for the biom file format," 2014.

[43] P. Murrell, *R graphics*. CRC Press, 2011.

[44] D. Murdoch, E. Chow, and J. F. Celayeta, "ellipse: Functions for drawing ellipses and ellipse-like confidence regions," *R package version 0.3-5*, 2007.

[45] S. Urbanek and J. Horner, "Cairo: R graphics device using cairo graphics library for creating high-quality bitmap (png, jpeg, tiff), vector (pdf, svg, postscript) and display (x11 and win32) output," *R package version*, pp. 1–4, 2012.

[46] A. Lucas, J. Tuszynski, H. Bengtsson, and N. Yes, "Package digest," 2013.

[47] G. H. J. Ron Kohavi, "Wrappers for feature subset selection," 1997.

[48] P. Pudil, J. Novovičová, and J. Kittler, "Floating search methods in feature selection," *Pattern recognition letters*, vol. 15, no. 11, pp. 1119–1125, 1994.

[49] K. Fukunaga, *Introduction to statistical pattern recognition*. Academic press, 2013.

[50] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 7, pp. 179–188, 1936.

[51] K. Pearson, "On lines and planes of closest fit to systems of points in space," vol. 2, no. 6, p. 559572, 1901.

[52] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*. MIT press, 2012.

[53] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics Springer, Berlin, 2001, vol. 1.

[54] J. M. Chambers, *Graphical methods for data analysis*, 1983.

[55] R. A. Becker, J. M. Chambers, and A. R. Wilks, "The new s language," *Pacific Grove, Ca.: Wadsworth & Brooks, 1988*, vol. 1, 1988.

[56] G. Trunk, "A problem of dimensionality: A simple example," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, no. 3, pp. 306–307, 1979.

[57] C.-L. Hwang and A. S. M. Masud, *Multiple objective decision makingmethods and applications*, 1979.

[58] A. Ultsch, *Is Log Radio a Good Value for Identifying Differential Expressed Genes in Microarray Experiments?* Fachbereich Mathematik und Informatik, 2003.

[59] C. Luo, L. M. Rodriguez-R, E. R. Johnston, L. Wu, L. Cheng, K. Xue, Q. Tu, Y. Deng, Z. He, J. Z. Shi *et al.*, "Soil microbial community responses to a decade of warming as revealed by comparative metagenomics," *Applied and environmental microbiology*, vol. 80, no. 5, pp. 1777–1786, 2014.

[60] M. Raman, I. Ahmed, P. M. Gillevet, C. S. Probert, N. M. Ratcliffe, S. Smith, R. Greenwood, M. Sikaroodi, V. Lam, P. Crotty *et al.*, "Fecal microbiome and volatile organic compound metabolome in obese humans with nonalcoholic fatty liver disease," *Clinical Gastroenterology and Hepatology*, vol. 11, no. 7, pp. 868–875, 2013.

[61] S. Yildirim, C. J. Yeoman, M. Sipos, M. Torralba, B. A. Wilson, T. L. Goldberg, R. M. Stumpf, S. R. Leigh, B. A. White, and K. E. Nelson, "Characterization of the fecal microbiome from non-human wild primates reveals species specific microbial communities," *PloS one*, vol. 5, no. 11, p. e13963, 2010.

[62] S. Greenblum, P. J. Turnbaugh, and E. Borenstein, "Metagenomic systems biology of the human gut microbiome reveals topological shifts associated with obesity and inflammatory bowel disease," *Proceedings of the National Academy of Sciences*, vol. 109, no. 2, pp. 594–599, 2012.