ON COMPOSITIONALLY AWARE AND NAÏVE APPROACHES TO NORMALIZATION OF 16S MICROBIOME DATA

by

Aaron Matthew Yerke

A dissertation submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in
Bioinformatics and Computational Biology

Charlotte

2022

Approved by:

_____
Dr. Anthony Fodor

_____
Dr. Elizabeth Cooper

_____
Dr. ZhengChang Su

_____
Dr. Alex Dornburg

_____
Dr. Gabriel Terejanu

Abstract

AARON MATTHEW YERKE. On Compositionally Aware and Naïve Approaches to Normalization of 16S Microbiome Data.
(Under the direction of DR. ANTHONY FODOR)

Compositional data refers to any data that represents parts of a whole, and DNA sequencing data is compositional in nature. This is due to the constraint on our current sequencing technologies that allow us to record a sample of the sequences rather than recording all the sequences. This means that sequencing data breaks the assumption of independence (Gloor et al., 2017). It has been long known that analysis of compositional data is challenging and can lead to spurious correlations. However, DNA sequencing data is inherently noisy due to both limitations of sequencing technology and its biological nature. Read depth, the number of sequencing reads from each sample, is known to be a confounding factor in many studies also plays a role in creating artifacts in this type of data. In this work, we demonstrate that read depth drives variance in four different datasets and propose a method for quantifying artifacts generated by read depth. We use this new method to compare untransformed data, several compositionally aware transformations, and a transformation which we call “lognorm” that normalizes samples by read depth in log space. Ultimately, we find that lognorm consistently had less read depth artifacts than the other transformations.

One way to determine the value of a data transformation is to show that it improves the performance of a machine learning classifier. We compared several common transformations to see if they improve the accuracy of a random forest and found that lognorm consistently significantly improves the accuracy of random forest. We believe that lognorm improves

accuracy by reducing read depth artifacts and allows the machine learning algorithm to learn from smaller signals within the data.

Dedication

I dedicate this dissertation to my wonderful and supportive family, especially my wife, Yogini, for her encouragement and support, and my mother, who helped provide me the time to finish my dissertation.

Acknowledgments

I would like to thank my advisor, Dr. Anthony Fodor, for his patience, guidance, and expertise while supervising this dissertation. I very much appreciate the time and resources that he expended to help me graduate.

My thanks to Ivory Blakely, Malcolm Zapata, and the recently minted Dr. James Johnson, for providing help on technical issues throughout my time as a PhD student.

I want to acknowledge fellow Fodor laboratory members: Dr. Shan Sun, Dr. Farnaz Faloudi, Dr. Matthew Tsilimigras, Jack Young, Ke Zeng, Anh Moss, Dr. Ali Sorgen, and Michael Sioda. They provided excellent scientific discourse and served as great role models.

I am appreciative of my committee members, Dr. Elizabeth Cooper, Dr. ZhengChang Su, Dr. Alex Dornburg, Dr. Gabriel Terejanu, and Dr. Aura Young for their time, feedback, and insight.

Finally, I would like to express my appreciation towards the UNC Charlotte Bioinformatics department - to faculty who help prepare me for the dissertation through their coursework and to staff members, such as Lauren Slane who kept me on track to graduate.

Table of Contents

List of Tables

List of Figures

List of abbreviations:

alr – additive log ratio

clr – centered log ratio

ilr – isometric log ratio

ASV – amplicon sequence variant

AUC – area under the curve

MLA – machine learning algorithm

PCA – principal components analysis

Chapter 1: Research overview

During my doctoral studies, I have two published papers on which I am co-author (described below) and one paper on which I am co-first author. In addition, this dissertation describes one project (chapter 3) which is close to publication ready and which we anticipate submission to https://www.biorxiv.org and for peer review shortly. Finally, we describe three projects which are more exploratory and will likely need additional coding or analyses before submission. This includes that last chapter of the dissertation (chapter 4) which describes an interesting artifact where sequencing depth is well correlated with many important meta-data variables as well as two projects briefly described below including a pipeline development project (BiolockJ) and an intriguing initial observation where different machine libraries can have substantial differences in performance on identical data.

Published

1. *A substitute variety for agronomically and medicinally important Serenoa repens (saw palmetto)*

Background: *Serenoa repens* (saw palmetto) is a small shrub that is native to Southeastern USA. The berries of uncultivated "green" saw palmetto are harvested and used for medicinal and health supplemental purposes, however it faces the challenge of being endangered due to overexploitation and its natural habitat loss. Further, the berries are an important part of the local

ecology. Besides the wild green variety of saw palmetto, there is a “silver” variety that is a cultivated commercially as a decorative landscaping plant.

Study aim: The aim of this study was to determine if berries of silver saw palmetto are a suitable substitute for the berries of wild green saw palmetto.

- **My contributions:** My role in the project was to perform the statistical analysis such as principal component analysis of tissue specific metabolites identified through multiple imaging and mass-spectrometry to show that the metabolite profile was very similar between the two berry varieties.

  I compared each metabolite from each imaging/chromatography/spectrometry technique to determine if they were different using statistical methods such as Student’s t-test and found that most of the biologically relevant ones were not statistically different. R scripts used for data processing and figure creation can be found at https://github.com/palomnyk/comparison_of_two_species. I am a co-author on this publication.

Study conclusions: A few metabolites differed greatly between the two types of berries, but they had little medical relevance. Our findings indicated that differences in the plants were only morphological (color and shape) and that their bioactive phytochemical constituents’ profiles were similar enough to support substitution for similar medicinal/therapeutic purposes.

Full citation: Jaiswal, Yogini, Daniel Weber, **Aaron Yerke**, Yanling Xue, Danielle Lehman, Taufika Williams, Tiqiao Xiao, Daniel Haddad, and Leonard Williams. "A substitute variety for agronomically and medicinally important Serenoa repens (saw palmetto)." Scientific reports 9, no. 1 (2019): 1-12.

2. ***3D Imaging and metabolomic profiling reveal higher neuroactive kavalactone contents in lateral roots and crown root peels of Piper methysticum (kava)***

Background: Kava is a small shrub whose peeled roots are consumed for their medicinal and recreational neuroactive properties. Kava is native to the Pacific islands where it has a long history of use in traditional medicine as a sedative, anesthetic, emetic, and euphoriant. Due to these properties, its use is being explored in the treatment of a wide range of neurological conditions including (but not limited to) alcoholism, depression, anxiety, psychosis, and sleep disorders. Despite its usefulness, there have been reports of kava causing hepatic toxicity.

Study aim: The aim of this study was to determine if specific parts of the kava plant have hepatotoxic metabolites. Traditionally, only the roots consumed, therefore we hypothesized that other parts of the plant may have some toxic properties and that if used as adulterants would have the potential to cause the hepatotoxic effects.

- **My contributions:** I provided the statistical analyses such as principal component analysis and mixed linear modeling of tissue specific metabolites identified through mass-spectrometry techniques. These data were multidimensional and cofounded, providing an interesting challenge. My work helped show that hepatotoxic metabolites were found in the stems and not the root cores or peels. R scripts used for data processing and figure creation can be found at https://github.com/palomnyk/root_stem_crown_comparison . I am a co-first author on this publication (full disclosure, the author co-first author is my spouse).

Study conclusions: This work concluded that the hepatoxic metabolites are found in the stems of the plants and challenged the practice of excluding peels from kava products.

Full citation: Jaiswal, Yogini S., **Aaron M. Yerke**, M. Caleb Bagley, Måns Ekelöf, Daniel Weber, Daniel Haddad, Anthony Fodor, David C. Muddiman, and Leonard L. Williams. "3D Imaging and metabolomic profiling reveal higher neuroactive kavalactone contents in lateral roots and crown root peels of Piper methysticum (kava)." GigaScience 9, no. 9 (2020): giaa096.

3. Avenanthramide Metabotype from Whole-Grain Oat Intake is Influenced by *Faecalibacterium prausnitzii* in Healthy Adults

Background: Avenanthramides (AVAs) are a type of polyphenols that are consumed only from oats and provide many health benefits through their anti-inflammatory effects. The metabolism of these molecules is initiated by the gut microflora rather than enzymes endemic to humans or mice, however until this project, the identity of this bacterium was not known. There are many AVAs, but they are structurally very similar. Our study focused on the AVA-C, also known as 2c, which is one of the most abundant AVAs in oats.

Study aim: My portion of the study focused on determining which bacteria was responsible for the initial steps in AVA metabolism based on metagenomic data and individual variations in response to whole-grain oats.

My contributions: Before joining UNCC's Bioinformatics Department as a PhD student, I worked on this project as a technician and organized the stool collection to make the metagenomic dataset. I also processed and stored the stool samples. After joining Dr. Fodor's

group, I continued to work on the dataset. I provided statistical analysis that demonstrated that *F. prausnitzii* was the best candidate for initiating AVA-C metabolism.

Study conclusions: Ultimately, a sample of *F. prausnitzii* was purchased and it was shown to perform the reaction *in vitro*. Stool of specific pathogen–free mice that were inoculated with *F. prausnitzii* but not in germ free mice was shown to perform the required reaction, thus allowing us to conclude that the bacteria is able to perform the required step of AVA-C metabolism. I am a co-author on this publication.

- **Full citation:** Pei Wang, Shuwei Zhang, **Aaron Yerke**, Christina L Ohland, Raad Z Gharaibeh, Farnaz Fouladi, Anthony A Fodor, Christian Jobin, Shengmin Sang, Avenanthramide Metabotype from Whole-Grain Oat Intake is Influenced by *Faecalibacterium prausnitzii* in Healthy Adults, *The Journal of Nutrition*, 2021;, nxab006, https://doi.org/10.1093/jn/nxab006

In Preparation

1. *Introducing BiolockJ – A unique tool for managing bioinformatics pipelines*

Background: The Fodor lab is developing their own unique platform that helps address the reproducibility crisis, BioLockJ. BioLockJ consists of a lightweight Java based framework that executes BASH scripts to call external applications called BioModules. The BioModules perform the heavy lifting, so to speak, by handling sequence processing, taxonomy assignment, univariate modelling, and report generations.

BioLockJ is meant to help a user go from raw data to publishable figures with minimal documentation overhead - all the information needed by BioLockJ should fit on a single page - the BioLockJ configuration file. To have replicable research, a user could publish their raw data, along with a single BioLockJ configuration file, which would indicate which BioModules would be run and basic input for them. BioLockJ will then be able to regenerate the publish-quality output.

The BioLockJ team has leveraged Docker, software that is used for making containers, for batch processing. This makes it very scalable and flexible in terms of operating systems. Thus with Docker integration, BioModules will have their own container with their dependencies preloaded. The BioLockJ team has created over 36 BioModules for running common statistical, genome assembly, or metagenomic packages such as QIIME, Kraken, or Metaphlan, as well as some custom modules for different statistical applications.

There are also customizable BioModules for Java, python, or R, so that community members can build their own modules. This is important because it allows users to expand the scope of BioLockJ's application.

My contributions: My initial contribution was to build a simple graphic user interface (GUI) using web development tools such as JavaScript, HTML, and CSS. However, as BiolockJ developed, this GUI became obsolete. As BiolockJ grew, I helped the overall application design, such as developing the Docker containers and porting data from one container to another. I also helped conceptualize the new GUI that has been recently developed by others in the group. My

programing, documentation, and other contributions to this project can be found at:

https://github.com/BioLockJ-Dev-Team/BioLockJ

Study conclusions: The Fodor group is already relying on BiolockJ to run our own pipelines.

This is a robust and flexible platform to process raw data to publishable figures.

Chapter 2: Compositional transformations for sequencing data

Overview

Microbial communities are complex biological entities that are crucial in virtually all natural and human-associated ecosystems. Advances in DNA sequencing techniques have made study of microbiomes possible on a scale that was previously impossible. This dissertation asks what the best way is to optimize the preprocessing and transformation steps in microbial DNA sequencing analysis to study the interaction of microbial communities with human hosts and other environments such as soil or water (**Figure 2**).

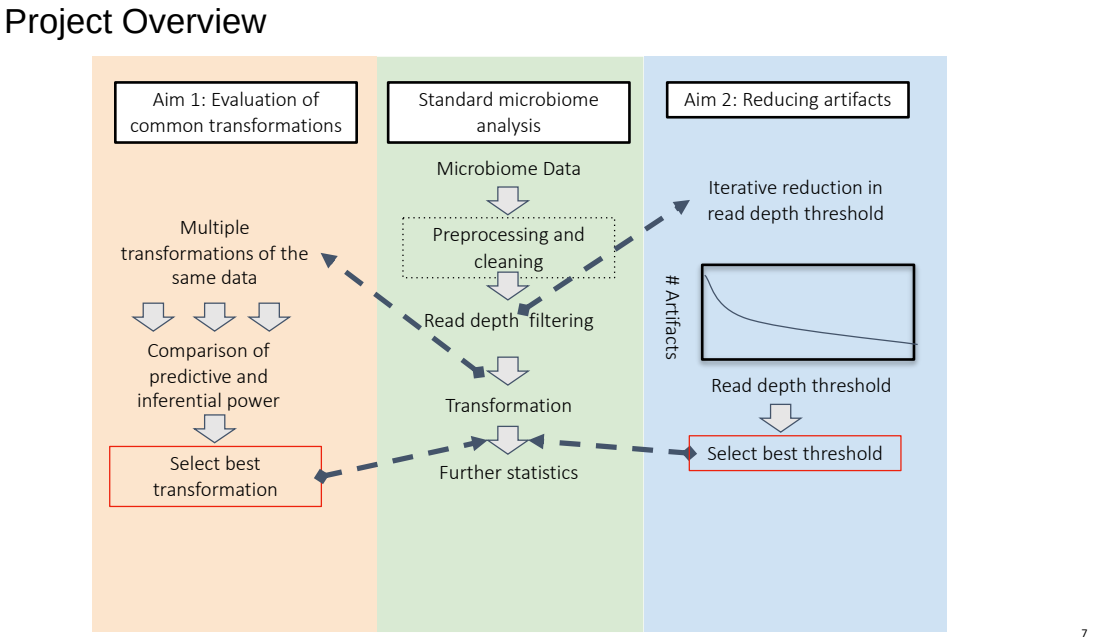

Figure 1: Improvements to microbial DNA sequencing analysis.
The standard analysis involves many steps (middle flowchart). Aim 1 describes our comparison of multiple transformations on the same dataset to find the best transformation for that dataset. Aim 2 describes an iterative process by which we find the best filtering threshold based on the number of artifacts.

Two of the biggest inherent challenges with DNA sequencing datasets are sparsity and compositionality (Gloor et al. 2017). Sparsity simply means that many taxa are only observed in

a few samples. Compositionality arises when the elements of a dataset are constrained to be parts of a whole. DNA sequences, like many other measurements in science, are inherently compositional data. In one sample, for example, 80% of the sequences might be assigned to phyla Firmicutes, 20% to Bacteroidetes, 15% Proteobacteria and 5% other. The population is constrained to 100%, so if one group changes, for example, Firmicutes increases, then the percentage of one or more different taxa must decrease. This means that assumption of independence in statistical modeling cannot reliably be assumed for compositional data and different techniques must therefore be used.

Current DNA sequencing technology samples DNA sequences with each sequence's absolute abundance as an unknown value. The resulting data is compositional because the values represent a proportion of an arbitrary and generally inconsistent number of reads known as read depth, rather than absolute counts. The read depth varies from sample to sample.
Using well-known, large, and publicly available microbiome datasets, this dissertation attempts to overcome some of the inherent challenges to analyzing this type of data.

Objective 1: Rigorous evaluation of commonly used compositional data transformations

Compositional data analysis has been an active research area for over 120 years (Pearson 1897). The standard methods for treating it is to transform it in such a way as to remove the constraining value. This process, however, makes the data more abstract and interpretation of results more difficult, but allows for the ability to use standard statistical methods that assume independence.

The most common treatments are variants of a log ratio transformation, as there is no constraint on the size of a log ratio. Each transformation has its pros and cons. One recently introduced transformation, Phylogenetic Isometric Log Ratio (PhILR) allows the use of phylogenetic trees in selecting and placing the elements into the ilr equation, offering possible advantages over others in using biologically relevant information as part of the transformation (Silverman et al. 2017).

With this aim, we evaluate these data transformations to identify any advantages that they offer, including use of various phylogenetic trees. This aim will reanalyze public datasets to determine if the new techniques can find new relationships in the data or validate previous results. Additionally, we will process the same dataset using multiple transformations to determine if any perform better at training machine learning models (MLM). The final step in this aim will be to assess if any of the transformations increase the statistical power of datasets for inference.

Objective 2: Develop statistical analysis method for understanding and reducing artifacts associated with samples that have different amounts of read depth.

In routine sequencing data processing, samples that have little or no sequences in them are removed, by setting a read depth threshold. This removes samples that might have been problematic for the sequencer at the expense of removing true samples with low abundance taxa. In the literature, we have often observed that an arbitrary threshold is often used rather than one that arises from rigorous evaluation. We find that the choice of which samples to include can

introduce subtle artifacts into all stages of the analysis, despite the use of compositionally aware transformations.

The discovery of these artifacts and their use in setting the filtering threshold provide researchers the opportunity to see how these choices affect the shape of their data and may indicate that compositional awareness is not the only factor to evaluate in a data transformation.

In summary, DNA sequencing analysis plays an important role in identifying genomic delineations in various diseases, diagnostics, and selection of the line of treatment. The research performed in this dissertation will provide insights into critical issues in an important analysis method that can ultimately improve the understanding of genetic factors related to human and animal diseases and crop improvements.

Background

The compositional nature of sequencing data

Compositional data arises when the elements of a dataset are constrained to be parts of a whole. Sequencing data is compositional in nature due to the fact that sequence counts are not counts of material input, but rather a proportion of an arbitrary and generally inconsistent read depth (Gloor *et al.*, 2017). To understand why compositional data requires different statistical approaches than standard count data, we should first look at the shape of the sample space. Each sample can be presented as a point in the sample space and the dimensions of the sample space are determined by the features of the data. In the case of DNA microbiome data, the features are the sequences or taxonomical classifications.

A two-part composition can be represented on two orthogonal axes as a straight line segment connecting each axis where the axis is 1, forming an isosceles right triangle with these axes. This straight line, which makes a hypotenuse with the axes, represents all possible values of the composition making it the sample space of the composition (**Figure 3**).

A

Sample sequencing data:

	Taxa 1	Taxa 2	Total
Counts	25443	2350	27793
% of total	92%	8%	100%

B

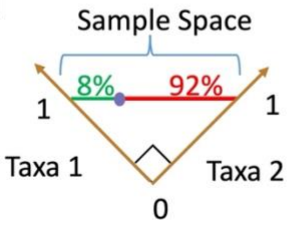

*Figure 2: Graphical representation of the sample space of a 2-dimensional sequencing data. (**A**) Example data of a single sample with counts in two taxa. Below the counts are the conversion of the numbers to percentages. The data are shown plotted on the brown simplex (**B**). The sample space (represented by red and green line segments) spans points 0,1 to 1,0. The purple dot represent the example data and is plotted at 0.92, 0.08. Since there are a limited number of sequences that sequencers can pick-up at a time, the sample space is finite, if the purple dot moved and the red line segment decreased, the green area would increase. If instead, the purple dot moved to the right, the red area would increase and the green area would decrease.*

The triangle that is made by the sample spaces and axes is called a unit simplex, which is the simplest object that can be made in a given space. As dimensions are added to the sample data, the unit simplex will increase in axes. A three dimensional dataset will be represented on a simplex three orthogonal axis and the sample space will be a plane that touches each axis at length 1 (John Aitchison 1994) (**Figure 4**).

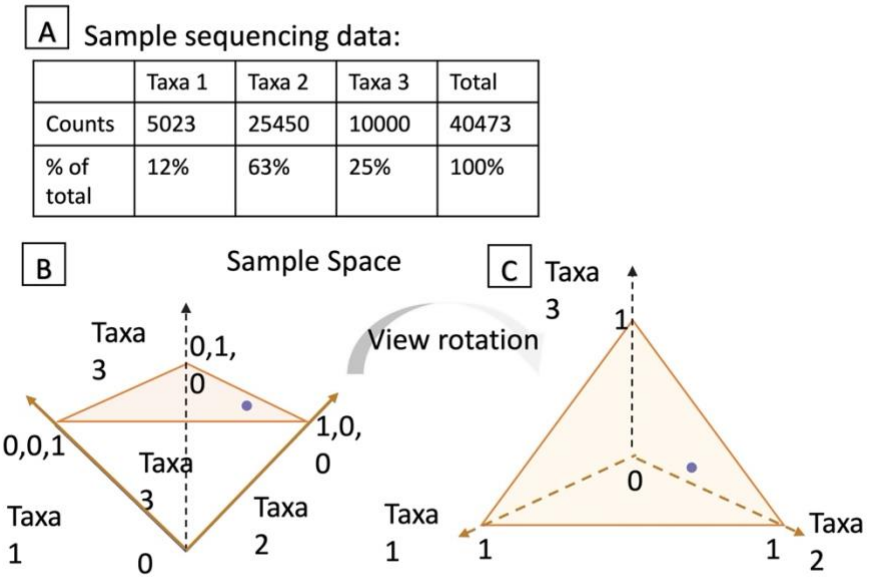

Figure 3: Graphical representation of the sample space of a 3-dimensional sequencing data. (A) Example data of a single sample with counts in three taxa. Below the counts are the conversion of the numbers to percentages. The data are shown plotted on the orange triangular plane (B). The sample spans points 0,1,0, 1,0,0, to 0,0,1. Taxa 3 (dashed black line) is partially hidden and runs away from the other two axis (brown lines). The purple dot represents the example data and is plotted at approximately 0.12, 0.63, 0.25. If we rotate our view of the plot in such that we are looking at the equilateral triangular plane of the samples space with the origin centered behind the plane, we can see more clearly, the approximate location of the purple sample data (C). From this view, this plot is equivalent to a ternary or simplex plot.

For each new dimension that is added to the data, a new orthogonal axis is added to the simplex, and the hyper-plane of the sample space grows to include it, however it is impossible to draw. A microbiome dataset containing k taxa could be modeled as a hyperplane on a k-dimensional simplex. A unit simplex where the maximum value for any point is 1 and none of the points are negative, and this is known as the Aitchison simplex (Aitchison, 1982) (**Table 1**). The constraint on the sum of the points helps the Aitchison simplex to represent the lack of independence in the points - if one taxa increases, another must decrease to keep the sum constant. It is also possible

to think of read depth as the constraining number for each axis rather than 1 since 1 is arbitrarily chosen to simplify calculations, however changing this number changes only the scale, but not the shape of the sample space.

Table 1: Data transformation and methods used and their formulas and references

Name	Formula
D-part simplex	$S^D = \left\{ \boldsymbol{x} = [x_{r1}, x_{r2}, x_{s1}, x_{s2}, \dots x_D];\ x_i > 0, i = 1, 2, \dots, D; \sum_{i=1}^{D} x_i = k \right\}$
Geometric mean	$G(x) = \sqrt[D]{x_1, x_2, \dots x_D}$
Centered log ratio (Aitchison, 1982)	$clr(x) = \ln[\frac{x_1}{g_m(x)}, \dots, \frac{x_D}{g_m(x)}]$
Additive log ratio (Aitchison, 1982)	$alr(x) = [log(\frac{x_1}{x_D}) \dots . log(\frac{x_{D-1}}{x_D})]$
Isometric log ratio (Egozcue *et al.*, 2003)	$ilr(r,s) = \sqrt{\frac{rs}{r+s}}\ ln\left[\frac{g_m(x_1,\dots,x_r)}{g_m(x_{r+1},\dots,x_{r+s})}\right]$

Number of possible combinations of ilr (Greenacre and Grunsky, 2018)	$\frac{(2x_D - 2)!}{(2^{x_{D-1}})(x_D - 1)!}$
Relative abundance/naïve proportion	RC/n
Logged relative abundance (Logged)	$log\left(\frac{RC}{n} + PC\right)$
Log normalization (lognorm) (Fodor et al. 2012)	$log\left(\frac{RC}{n} \times \frac{\sum x}{N} + PC\right)$
Margalef richness index	(RC - 1) / Log (n)
Shannon Wiener diversity index (Shannon 1948)	$H = -\Sigma (RC/N)_D * ln((RC/N)_D)$

Euclidean norm	$\sqrt{\lvert a\rvert^{2} + \lvert b\rvert^{2}}$
Aitchison norm or closure (*C*)	$C(x_{\{1\}}, x_{\{2\}}, \ldots, x_{\{D\}}) = [\frac{x_{\{1\}}}{\sum_{i=1}^{D} xi}, \frac{x_{\{2\}}}{\sum_{i=2}^{D} xi}, \ldots, \frac{x_{\{S\}}}{\sum_{i=S}^{D} xS}]$
For these formulas: $g_m(x)$is the geometric mean of x and D is the dimensions of the matrix x. S^D represents a simplex of D dimension with individual dimensions represented as x_D. In the case of a metagenomic dataset, the D dimensions could be taxa or OTUs. The variables *r* and *s* represent arbitrary subsets of S^D and x are compositional counts of *r* and *s* - in a metagenomic datasets, they would represent different taxa. RC = raw counts in a cell, n = number of sequences in a sample, Σx = total number of counts in the table, N = total number of samples, PC = pseudo-count, usually taken to be equal to "1". Table is modified from Matthew Brown's dissertation (Brown n.d.).	

Compositional data transformations

Common statistical methods assume that data are not bound by this summation constraint, and thus problems may arise if they are used on compositional data. The assumption of independence in parametric statistics means that they assume that the data may occupy any point in real space, rather than being limited to a narrow hyperplane. In order to use parametric techniques, the data must first be transformed in such a way that the sample space is in real space. An ideal transformation would also preserve the metrics of compositional data (Egozcue et al. 2003) and the resulting data should be easily interpretable.

The center-log ratio transform and the additive log ratio transform are standard techniques

In the 1980s, Aitchison realized that sample space for a logarithm of ratios is real space and came up with two such methods (**Table 1**), the center-log ratio transform (clr), and the additive

log ratio transform (alr) (**Figure 5**) that transform compositional data from Aitchison's simplex to real space (J Aitchison 1982).

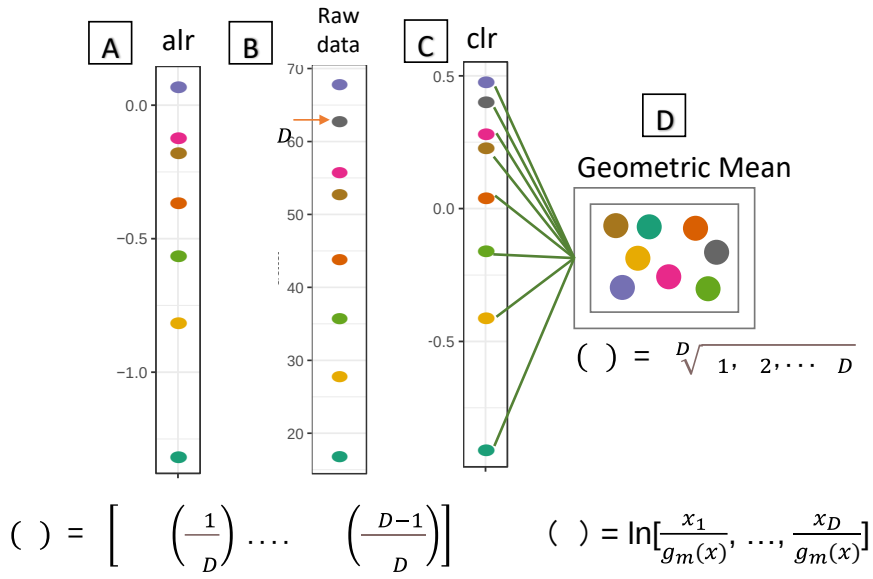

Figure 4 : Graphical representation of alr and clr transforms of a single sample. The raw data (B) holds 8 taxa. The grey point at 63 was chosen for the value of xD for the alr transform and its formula is shown below the plot. The alr transform has one less point than the raw data as the x_D point is used only in the denominator (A). The clr transformed data (C) has the same number of points as the raw data. The geometric mean is the denominator for the clr (D). It should be noted that the relative distance between the points is not preserved in either of these transforms.

Though these transforms are commonly used, they have some shortcomings when working with microbiome sequencing. The alr transformation is simply the log of each ratio over an arbitrary point (x_D) from the set. In the case of metagenomic data, this would be choosing a reference taxon (such as Firmicutes) and dividing each datapoint for each sample by the value of Firmicutes of that sample. Choosing x_D, the denominator, can be arbitrary and depends on the taxa that are available in the dataset.

In microbial ecology the most often used solution is to choose a ubiquitous taxon as x_D. However, due to the stochastic nature of sampling in sequencing data, even technical replicants may have very different abundances of taxa in different samples. The chosen taxa must not have zero counts any of the samples, as dividing by zero is undefined, though this can be remedied by adding a small number to all counts, so that there are no zeros in the dataset. This addition, called a pseudo count, can create bias if it treats all the zeros the same, because not all zeros in a metagenomic dataset are the same. Some zeros represent true biological absence of a species and some represent sampling or technical errors (Silverman et al. 2020).

Comparing across datasets may also be problematic, especially for diverse datasets, as the value of $x_D$can change the distance between the transformed values and may cause arbitrary differences in interpreting experimental results (Egozcue *et al.*, 2003).

The drawbacks of the clr relate to the fact that it transforms the simplex data into a subspace of real space, such that there are still limitations on the shape of the sample space. The sum of clr transformed data is zero, which leads to a single covariate matrix. This leads to problems in certain types of downstream analysis and a sensitivity to outliers (Gloor et al. 2017). In addition, the clr and alr make the biological significances of the results difficult to interpret.

<u>The isometric log ratio improves on the clr</u>

The isometric log ratio (ilr) builds upon the clr, as is evident by the similarity of their formulas (**Table 1**). The ilr creates a contrast between subsets of the compositional parts. The resulting transformed data are isometric - meaning that the angles and scalers of the untransformed data are preserved after the data transformation (Egozcue *et al.*, 2003). Like the alr and clr, the

biological significance of the ilr transformation is difficult, but not impossible, to interpret. However, when transforming data to log ratios, this is inevitable.

Another shortcoming of the ilr is that it is not a point-for-point transformation, but rather a transformation of ratios of subsets of the original data. For example, the number of possible ratios or balances, as they are called in some literature, increases exponentially with the number of features in the dataset (**Table 1**). The number of balances possible balances in a sample with 5, 20, or 53 features is 21, 4.100e+20, and 5.34e+80, respectively, with the latter being close to the number of atoms in the known universe. In one of the datasets that we will examine (the Jones dataset, Table 2) for example, there are 28,026 Amplicon Sequence Variants (ASVs) after preprocessing or 303 genus level taxa. Using all the possible ilr balances at either the ASVs or the genus level is computationally impossible, thus we need to find a way to use only selective ratios.

Fortunately, multiple groups (Morton *et al.*, 2017; Silverman *et al.*, 2017) realized that microbial sequencing data can be used to build bifurcating trees with nodes, edges, and tips that will give a natural subsets to the data for use in the ilr (**Figure 6**). Nodes of the trees become the subsets, or "balances" and the tips become the counts for the geometric means. Thus, we can run statistics on the ilr transformed nodes of phylogenetic trees such as those created by the unweighted pair group method with arithmetic mean clustering UPGMA method (Morton *et al.*, 2017). The resulting data are called sequential binary partitions or balance trees.

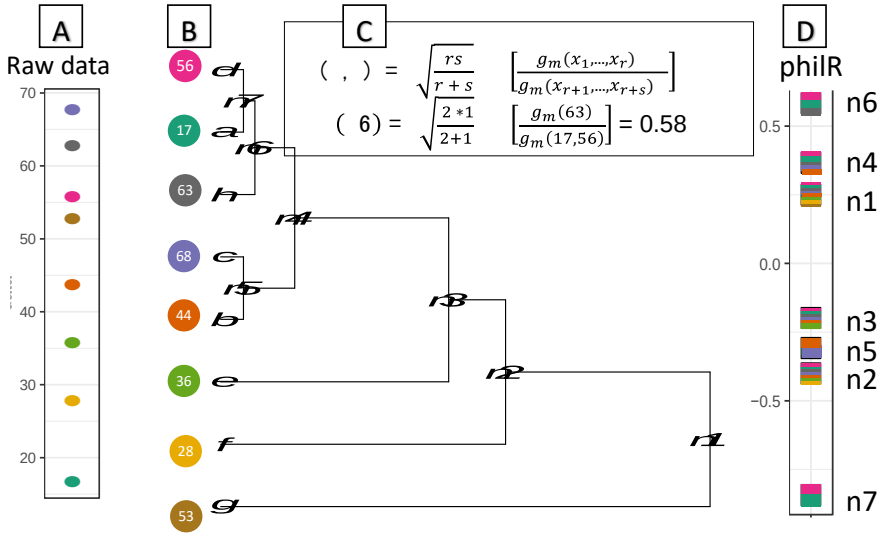

Figure 5: Graphical representation of a single node in a balance tree.
The raw data count data (A) from Figure 4 were used as the tree tips of the phylogenetic tree (B). The nodes of the tree are labeled n1 through n7. The balances for each node are calculated using the ilr formula and the with the values for node 6 used as an example for the usage of the formula (C). The resulting balances (D) are plotted to show which features were used to make them.

Balance trees add arbitrariness to the pipeline

While bifurcating phylogenetic trees work as a guide for the ilr, there are some seemingly arbitrary decisions that must be made. For example, we must determine which clade goes into the denominator and which goes into the numerator of the ilr formula. Both the R and the Python packages make these decisions arbitrarily. For example, gneiss, a python package, arbitrarily puts the left clade in the denominator and the right clade in the numerator, thus relying on the tree building algorithm to consistently put the sister clades on the correct side (Morton *et al.*, 2017).

An R package, PhILR uses allows for the user to direct the numerators and denominators with a sign matrix of the tree to decide which node does into the numerator (Silverman *et al.*, 2017; *Plotting a Sequential Binary Partition on a Tree in R · Statistics @ Home*, no date). This is a technique these authors borrowed from a tree building algorithm named ggtree ((Silverman *et al.*, 2017; *Plotting a Sequential Binary Partition on a Tree in R · Statistics @ Home*, no date)). If the sign matrix is not provided by the user, PhILR will arbitrarily pick in a similar way to gneiss. Given the arbitrary nature of this choice and no real way to make it circumspect, one might ask how important it is. The theory be behind the ilr, from which the balance trees are based, gives broad leeway in deciding how to partition the compositions into the ilr formula. Theoretically, an ilr with an equal number of random balances as a balance tree are both equally valid (Silverman et al. 2017). However, if this is true then this means that balance trees have quite a bit of randomness in-built and more rigorous testing seems warranted.

The Shannon-Weiner diversity index

When comparing metagenomic datasets, we often want to compare some aspect related to the number of classified sequences (taxa) or the sequences themselves between samples of a dataset – such a measure is broadly called the diversity index. There are many aspects of diversity that one might measure including, but not limited to, *richness* and *evenness*. Richness relates to the total number of features and evenness relates to the relative abundance of the samples within a dataset (Cameron et al. 2021).

One measure that combines both richness and evenness is the Shannon-Weiner diversity index (Shannon index). This equation provides a value that fits with an intuitive understanding of diversity (**Figure 7**). The Shannon index equation originally described entropy in statistical

mechanics; however, it was found useful for describing probability to predict a taxa randomly pulled from a sample (**Table 1**). The lowest possible Shannon index is 0 and this arises when a sample has only a single taxon and there is zero probability of a wrong prediction. The upper limit to the Shannon index depends on the size of the sample, which in the case of a metagenomic dataset is the read depth or the number of taxa in a sample (Shannon 1948).

A

Counts table	Taxa A	Taxa B	Taxa C	Total	Margalef index	Shannon index
Sample 1	4	4	4	12	0.8049	1.1
Sample 2	10	1	1	12	0.8049	0.566
Sample 3	6	6	0	12	0.5581	0.693

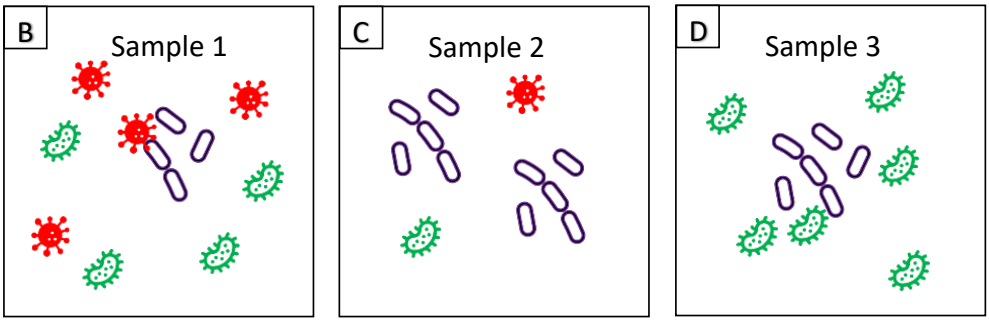

Figure 6: The Shannon-Weiner index gives an intuitive score of diversity.
An example count table of a dataset containing three samples that illustrate the advantage of the intuitive Shannon index (A). For each sample, the Shannon index is calculated, as well as the Margalef index which measures richness only. Each sample has the 12 counts; however, Sample 1 has them spread evenly between the three taxa. In Sample 2, Taxa A has a has ten times as many counts as either Taxa B or Taxa C drastically reducing its evenness. Even though its richness is the same as Sample 1, Sample 2 (C) has 50% of Sample 1's Shannon index. Sample 3 (D) has less taxa and therefore less richness than either other sample yet has a slightly higher Shannon index than Sample 2. The Shannon index provides a very intuitive measure for determining the diversity of a sample than the measure of richness alone.

Because the Shannon index measures richness, and metagenomic samples within the same dataset are unlikely to have the same read depth, researchers often seek to control richness through the process of rarefication, or rarefying. Rarefication simply means that all samples will be subsampled to a specific read depth before further analysis such as measuring the Shannon index. This means that some taxa from samples with high read depth will not be used and samples below the rarefication threshold will be discarded. Thus, it is likely that much of the dataset will be discarded before the analysis has even started, leading some researchers to argue that rarefication is "inadmissible" (McMurdie and Holmes 2014). Lately however, the idea of rarefication has been revisited and it is argued that repeatedly rarefying samples can ensure that more data is used (Cameron et al. 2021).

Conclusion

This dissertation will consist of two projects – one that examine the usage of phylogenetic balance trees and one that will examine how read depth shapes sequencing data. As we examine the usage of the balance trees, we will compare them to other transformations mentioned in this chapter. We will be determining if they can improve the accuracy of a random forest machine learning model. We will also be examining the role of read depth in metagenomic datasets and its effect on data transformations.

Chapter 3: Log-normalizing to read depth outperforms compositional data transformations

Abstract

Balance trees, also known as sequential binary partitions or CODA dendrograms, are a recent evolution in compositional data transforms. This transform combines the relative abundance counts with phylogenetics to create transformation which are ratios of subsets of the original data. As they are still a nascent technique in the field of microbiome analysis their utility has yet to be fully probed.

In this project, we rigorously examine the consequences for the ilr transform by comparing balance trees to well established compositional and non-compositional transformations. We first sought to optimize our ilr transformations by comparing different phylogenetic trees for inputs as implemented in the PhILR package, a popular R package. To achieve this aim, we compared two *de novo* tree building algorithms, UPGMA (hierarchical clustering) and IQTREE, and use of the Silva reference phylogeny to build trees. We also examined two different types of weighting as offered by PhILR. Combined, these weighting schemes have 10 possible options and combine to yield 24 combinations. We then compared the accuracy of our custom balance trees to the raw counts table, an inhouse transformation that normalizes samples to the read depth called "lognorm", and two compositionally aware transformations (the additive log ratio and the centered log ration) with the random forest machine learning algorithm. For four publicly available datasets, all these different data transformations were used to train random forest algorithms and the performance was measured by repeatedly shuffling the datasets with four-fold cross validation. We found that the compositional data transformations such as alr, clr, and ilr (PhILR) not only generally failed to improve upon raw counts tables, but often performed worse.

Our trials show that lognorm outperforms all the compositional transformations by a small but statistically significant margin.

We conclude that while the reasoning behind compositional transformation is compelling, sequencing data may have other artifacts whose effect is greater. We recommend the lognorm data transformation as preprocessing sequencing data for random forest as it had the highest accuracy among the transformations that we tested.

Introduction

As reviewed in the first chapter, compositionality is a well know problem in sequence experiments. Compositional data is any data where the data are considered to be parts of a whole (J Aitchison 1982). As such, independence between values cannot be assumed because if one part increases in value, another must decrease due to the constraint of all parts summing to a fixed amount. In the case of current DNA sequencing technologies such as Illumina Miseq, the fixed amount that sequences sum to is the read depth (Gloor et al. 2017). This is a problem because parametric statistics assumes independence and vectors of features that sum to a constant are, by definition, not independent. In order to circumvent this problem, many transformations to mathematically endow the data with independence have been proposed including isometric log ratio (ilr) which, as reviewed above, has the disadvantage that there are an astronomical number of possible variations (see **Chapter 2**).

A solution to the near infinite number of ilr transformations was independently proposed by two groups who utilized phylogenetic data to guide the ilr transformations (Morton et al. 2017; Silverman et al. 2017). While being mathematically no better than any other ilr transformation with the same number of balances, this approach provides an intuitive way of building balances.

In addition to offering a method of creating the balance trees, PhILR offers two weighting schemes to address the problem of the over representation of zeros in the counts tables and to allow the PhILR algorithm to incorporate the phylogenetic input into the output. However, the benefits of these weighting schemes to machine learning, if any exists, remain unclear.

In this chapter we consider the question of whether compositional corrections, such as those mentioned in the previous chapter, can aid in training machine learning methods. It is not uncommon to compositionally correct data before analysis with an MLA (Randolph et al. 2018; Zeller et al. 2014; Sisk-Hackworth et al. 2021; Lin, Salleb-Aouissi, and Hooven 2022; Maltecca et al. 2019).

On the one hand, one might anticipate that using the phylogenetic data to build balance trees, as is done with PhILR, could yield better results as the phylogenetic tree seems to add data to the study. On the other hand, some aspects of a dataset are distorted when converted to balances and it is unclear if this would impact the accuracy machine learning algorithms. We examine what impact that the choice in phylogenetic trees has on the PhILR transform, and the benefits of PhILR's weighting schemes to see if any of these tree-based transformations effect the accuracy of random forest machine learning.

In summary, with this project, we intend to explore the advantages and consequences of using balance trees with sequencing data. If balance trees are advantageous, we wish to explore the situations for which the various versions of the balance trees are most useful for microbiome data. We will show that no weighting scheme or combination of weighting schemes is more effective than another. We find, somewhat surprisingly, that raw, unnormalized counts tables and lognorm work better than compositional data transformations with the random forest algorithm.

Materials and Methods

We choose publicly available 16S microbiome sequencing datasets for our analysis with enough samples for which there were inference categories such as case/control (**Table 2**). We arbitrarily set this threshold at 200 samples. From each dataset, we arbitrarily dropped sample metadata features that was sparse (<1/4 total samples) to create some uniformity between the datasets.

Table 2: List of datasets used in chapter 3

Name	Number of Samples	Metadata Categories
Vangay (Vangay et al. 2018)	634	Recruitment.Location, Researcher, Sub.Study, Birth.Year, Age, Highest.Education, Ethnicity, Religion, Birth.Location, Type.Birth.Location, Arrival.in.US, Years.in.US, Location.before.US, Type.location.before.US, Years.lived.in.Location.before.US, Tobacco.Use, Alcohol.Use, Height Weight, Waist, BMI, BMI.Class, Breastfed, Age.at.Arrival, Sample.Group, Waist.Height.Ratio
Jones (Jones et al. 2018)	233	Age, BMI, Genotype, sex, Treatment, Visit, type
Zeller (Zeller et al. 2014)	226	Age, host_subject_id, geographic_location_(country_and/or_sea region)

Noguera-Julian (Noguera-Julian et al. 2016)	700	Host_Age, ETHNICITY, geo_loc_name_country, HIV_RiskGroup, HIV_serostatus, host_other_gender, host_sex, HIV_Profile, PCR_human_papilloma_virus, host_allergy, host_deposition_frequency_per_day, host_abdominal_transit_alterations, host_Residency_Area, HCV_coinfection, Anal_cytology, host_sexual_orientation, Syphilis_serology, HBV_coinfection, PCR_Neisseria_gonorrhoeae, PCR_Chlamydia_trachomatis, HIV_viral_load, CD4+_Tcell_counts, leukocytes, stool_consistency, lymphocytes, host_body_mass_index

Sequence processing

For 16S sequencing, we only used the forward reads, as the reverse reads tend to have a higher error rate (Schirmer *et al.*, 2015). We filtered, trimmed, removed bimeras, and assigned taxonomy to the 16S sequences with version 1.0.3 of the R package DADA2 (Callahan *et al.*, 2016). The resulting ASVs were aligned using version 2.0.2 of the R package DECIPHER (Wright, Yilmaz and Noguera, 2012).

Tree building

There were many steps taken to create all the transformations used in this project. We have provided a sketch of how each transformation was generated (**Figure 8**).

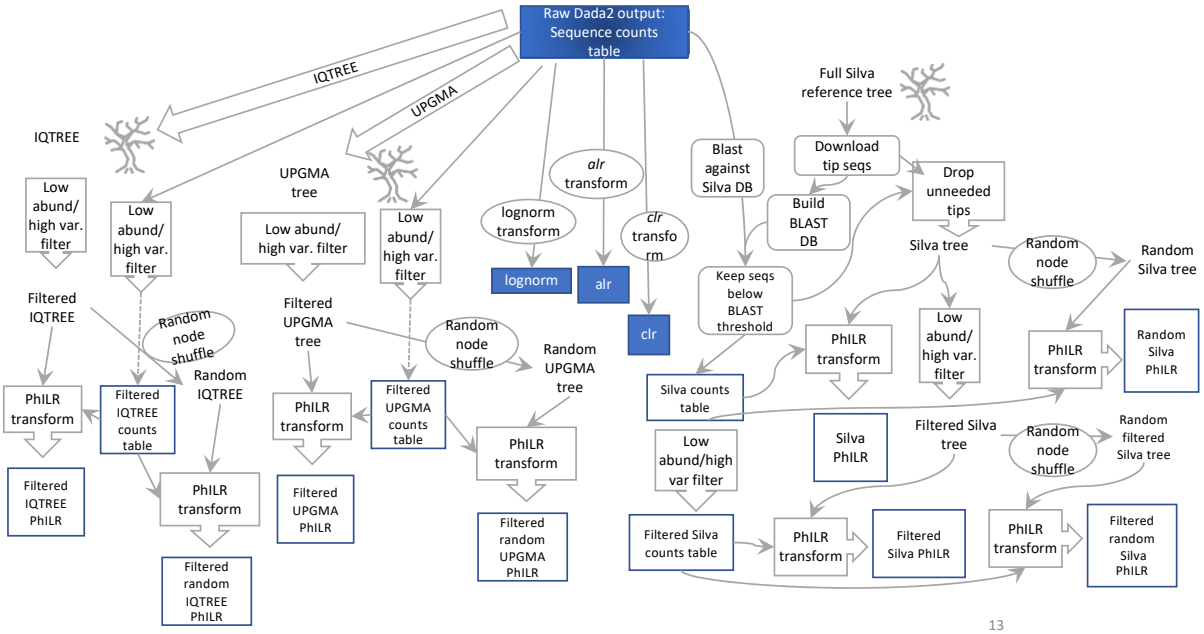

Figure 7, Schematic representation of workflow for the creation of each one of our data transformations.
This schematic starts with the raw Dada2 counts tables and each step in the workflow (grey arrows with white bubbles) lead to trees (grey blocks) or final datasets (blue boxes). Note that the datasets with random node shuffles were recreated at least 5 times.

To determine how the choice of phylogenetic tree impacts the ability of the transformed data to train a random forest algorithm, we chose two *de novo* trees and one reference tree, the Silva's Living Tree Project (Munoz et al. 2011), which tries to create the most accurate phylogeny as possible. We also modified the Silva reference tree for each dataset in our study by removing the taxa that were not present in that dataset from the Silva tree. For our *de novo* trees, we utilized another type of phylogenetic tree using the unweighted pair group method with arithmetic mean clustering (UPGMA). UPGMA is a method that clusters the sequences based on distance matrices (Weiß and Göker 2011). Hierarchical clustering is often considered to be an overly simple an approach, but we felt that it would be useful as a *de novo* control. The other *de novo* method is IQTREE, which infers trees by maximum likelihood. A disadvantage of *de novo*

methods, compared to the Silva reference tree, is that it will use all of the sequences available to it and will force branches between nodes based on this limited data. Since this method uses all the sequences available to it, it will contain more nodes than the trees we made through subtraction of the Silva tree. However, though it contains far more information, it will likely be less accurate than the Silva tree because the sequences are not carefully curated.

As a further control in this experiment, we used randomly generated trees that have the same number of nodes as other trees. In a randomly generated tree, each node is placed in a random position irrespective of true phylogenetic distance. The random trees are meant to test if PhILR requires a true tree to function. We also compare each PhILR transformed dataset to raw counts tables, alr, clr, and lognorm.

UPGMA Tree

To build the UPGMA trees *de novo* from the sequencing data, we used version 2.9.0 of the phangorn R package (Schliep, 2011). Our pipeline for mapping the sequencing data of each of our datasets to a reference tree relies on several tools and R packages.

Silva Living Tree Project

The reference tree comes from Silva's Living Tree Project, 16S rRNA-based LTP release 132 (Munoz *et al.*, 2011). The reference tree lists the GenBank locus at each tip, so we used this information to download the sequences from GenBank using the ape package. We then built a blast database out of the sequences and blasted the sequences from our study datasets using custom BASH scripts. If the resulting matches had e-value greater or equal to 10^{-10}, we culled them from the tips of the reference tree using custom R scripts to get a customized reference tree (Silva tree). Though the exact value of 10^{-10} was chosen arbitrarily, we believe that it is a

conservative threshold. We hypothesized that though the PhILR transformations made from Silva trees were smaller, they would perform better due to a filtering effect.

IQTREE

The alignment file from the DADA2 sequence processing was processed through the IQ-TREE version 2.1.2 for Linux 64-bit. We allowed the modelFinder to run for 48 hours on the Jones dataset using 1 core and then selected the highest scoring model for subsequent runs. The highest scoring model was "GTR+F+R5", which is a combination of a general time reversible model with unequal rates and unequal base, empirical amino acid frequencies, and the R5 free-rate model.

Images of all non-random trees used in this project can be found in **Supplementary A**.

The resulting customized reference trees and the *de novo* trees were then available for building phyloseq objects using version 1.16.2 of the phyloseq R package. A phyloseq object consists of a single object that holds sequencing data, sequence metadata, a taxonomy, and a tree. The phyloseq objects were later used for the philr transform using version 3.15 of the PhILR R package.

Variance and low abundance filtration

At this point, corresponding pairs of trees and counts tables were low abundance filtered using the following criteria:

sum(x > 3) > (0.2*length(x))

And then high variance samples meeting the following criteria were filtered:

sd(x)/mean(x) > 3.0

where x is the sequence in the counts table, sd is standard deviation, and the length is the number samples – the sequences were also dropped from the tips of tree corresponding tree (**Figure 8**). The sample counts were then given a pseudo count of 1 to eliminate the zeros that would impede PhILR. Finally, the phyloseq objects were processed through PhILR to create the ILR transform of our sequencing data tables. Node names were voted based on PhILR's voting function.
The PhILR vignette provided a low abundance/high variance filter- without this, the UPGMA and IQTREE trees were too large for PhILR, thus we only used the filtered version of each of these trees in our experiments.

Random trees

For each tree that we created with our workflow, we created 3 corresponding trees that had the same number of nodes and tips, but the nodes were randomly shuffled using the rtree function from version 5.6-2 of the ape R package (Paradis, Claude, and Strimmer 2004) (Data not shown).

Non-tree transformations

The *alr* and *clr* transformations of the ASV tables were done using the alr and clr functions, respectively, of version 1.1.15 of the R package called "rgr" (Garrett 2013). For the denominator of that *alr*, the taxa that was present in the most samples, was selected.
The lognorm function as per the formula in **Table 1** was written using custom R scripts.

Statistical tests

Kendall's correlations were calculated with R's inbuilt *cor.test* function.
ANOVA was calculated using R's inbuilt *anova* function. The Wilcoxon test was calculated using the *wilcoxon* function from version 1.9.1 of the Python library SciPy (Virtanen et al. 2020).

Machine learning algorithm selection

From version 1.1.2 of the scikit-learn Python library we selected the following MLAs to compare: logistic regression, linear discriminant analysis, k-nearest neighbors, decision tree, random forest classifier, Gaussian naive Bayes, and support vector (Pedregosa et al., n.d.). The Silva reference tree transformation with each of PhILR's 24 weighting scheme combination was tested with 10-fold cross-validation for each of the selected metadata features of each of the datasets.

Random forest comparisons

To create training and testing datasets we randomly assigned ¾ of our data to training and ¼ to testing. We felt that this was an acceptable split as we had chosen large datasets. It was then processed by the random forest algorithms and then the data were shuffled and split and processed again. We employed this shuffle/analysis cycle 20 times.

The random forest models were created using version 1.1.2 of the scikit-learn Python library and the in-built scoring methods were used to record accuracy (Pedregosa et al., n.d.). For categorical features, the random forest classifier was used and the recorded accuracy score was generated by dividing the correct number of predictions over the total number of predictions. For numeric data, the random forest regressor was employed and the recorded accuracy score was the coefficient of determination R^2, defined as $(1-uv)$, where u is the residual sum of squares and v is the total sum of squares.

There are 24 possible weighting combinations offered by PhILR

As mentioned earlier, PhILR offers two weighting schemes – one with four options and one with six options. Thus, there are a total of 24 combinations of weights offered. These weighting schemes provide a method to transform the count data before it enters the PhILR algorithm. PhIlR therefore allows for weighting based on branch length and/or the values of the tree tips. Philr offers a variety of options for both the values (part weights) and the branch lengths (ilr weights). For the part weights, there are 6 options: no weight, the geometric mean, the Aitchison norm, the Euclidean norm (**Table 1**), and the geometric mean multiplied by either the Aitchison norm or the Euclidean norm. The authors of PhILR prefer the geometric mean multiplied by the Euclidian norm, as this performed well in their preliminary benchmarks (Silverman et al. 2017). Each balance can either be weighted by the sum of its children's branch lengths, the square root of the sum of the children's branch lengths, or the sum of children's branch lengths plus the mean descendants each child's mean distance to its descendent tips. These weights enable the PhILR transform to differentiate itself from a pure ilr transform (Silverman et al. 2017).

Data sharing:

All BASH, Python, and R code and dataset metadata used for this project is available in the Git repository: https://github.com/amyerke/dissertation_lognorm_vs_CODA.

Results

Trees accepted by PhILR can have very different properties

As outlined in the introduction, compositionality can have a profound impact on the analysis of microbiome sequencing data. In this chapter, we examine the performance of different algorithms correcting for compositional artifacts in machine learning applications. We examine many different balance-tree based strategies for producing tables describing metagenomic datasets in which PhILR produces a weighted balance for an input binary phylogenetic tree. This pipeline has many arbitrary choices for optimization such as the weighting scheme or the phylogenetic tree used as input. In order to best explore the consequences of these choices for machine learning, we made several different trees to use as input. Our rationale for this was to determine if factors such as the weighting schemes take branch length, and the number of node descendants would improve the quality of the transform for machine learning. We choose to test UPGMA and IQTREE for building trees which use all available sequences as nodes as well as the Silva Tree of Life as the basis for a reference-tree based algorithm that does not use all sequences in a dataset. After building all the trees, we observed that that for every dataset, the Silva trees were an order of magnitude smaller than the UPGMA and IQTEE trees (**Table 3**). This is because the Silva trees were made through a subtractive method where only sequences in the intersection between our datasets and the Silva Tree of Life were used, whereas the UPGMA and IQTREE algorithms use all of the sequences to build a de-novo tree.

Table 3: Tree descriptions.

	tree name	num. nodes	num. tips	ave. branch length	variance branch length	ultrametric

Jones	Silva	1132	1133	0.0311	0.0012	FALSE
	Filtered_Silva	75	76	0.0556	0.0032	FALSE
	Filtered_UPGMA	227	228	0.0307	0.0015	TRUE
	UPGMA	28025	28026	0.0384	0.0104	TRUE
	IQTREE	28024	28026	0.1205	0.1798	FALSE
	Filtered_IQTREE	227	228	0.0840	0.0191	FALSE
Vangay	Silva	906	907	0.0344	0.0012	FALSE
	Filtered_Silva	35	36	0.1038	0.0130	FALSE
	Filtered_UPGMA	70	71	0.1300	0.0642	FALSE
	UPGMA	6821	6823	0.0521	0.3193	FALSE
	IQTREE	6821	6823	0.0202	0.0299	FALSE
	Filtered_IQTREE	70	71	0.0813	0.0144	FALSE
Zeller	Silva	1490	1491	0.0309	0.0009	FALSE
	Filtered_Silva	121	122	0.0606	0.0044	FALSE
	Filtered_UPGMA	207	208	0.0332	0.0012	TRUE
	UPGMA	11077	11078	0.0154	0.0007	TRUE
	IQTREE	11076	11078	0.0298	0.0409	FALSE
	Filtered_IQTREE	207	208	0.0792	0.0136	FALSE
Noguera-Julian	Silva	1233	1234	0.0330	0.0011	FALSE
	Filtered_Silva	52	53	0.0840	0.0089	FALSE
	Filtered_UPGMA	122	123	0.0273	0.0025	TRUE
	UPGMA	20365	20366	0.0509	0.0180	TRUE
	IQTREE	20364	20366	0.1204	0.1501	FALSE
	Filtered_IQTREE	122	123	0.0668	0.0295	FALSE

For each phylogenetic tree that we used for PhILR, we made 5 random shuffles of the nodes and included them as controls. The shuffled trees have the same number of nodes, but their nodes do not match to the same tips as the true trees and their branch-lengths will be incorrect. This tests how important this information is to weighting schemes that use it.

As we would expect given the very different algorithms used to produce them, despite coming from the same sequencing data, the UPGMA, IQTREE, and Silva trees have very different phylogenetic structures as can been by visual application (**Figure 9**).

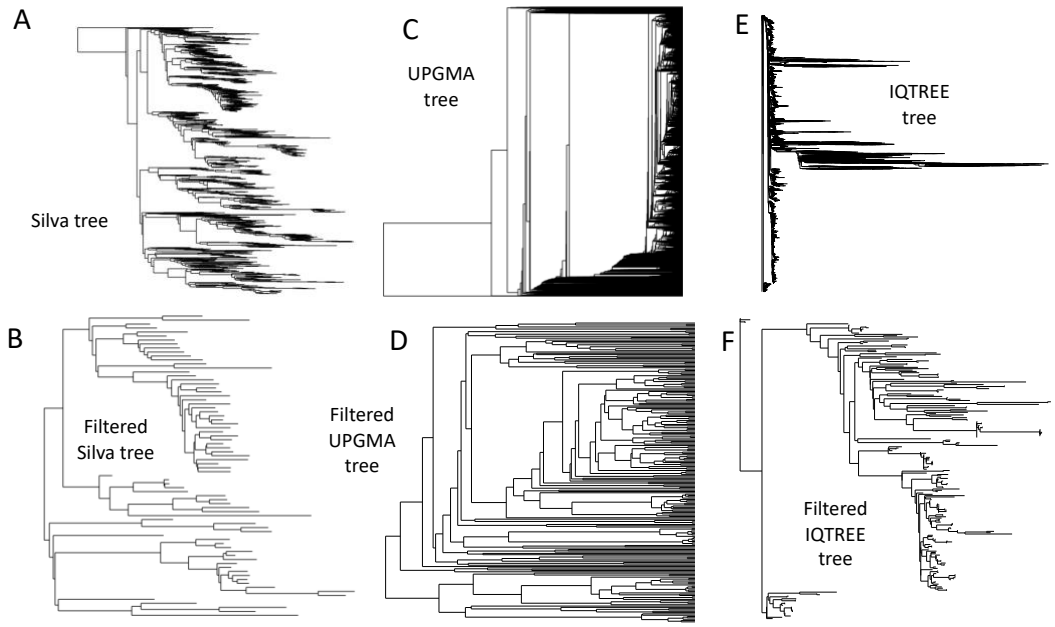

Figure 8: The low abundance/high variance filtered trees are less cluttered than the originals in the Jones dataset.
The original Silva reference tree (A) shows more taxa than the filtered version (B). The UPGMA (C) and IQTREE (E) algorithms are both greedy and thus the trees are too dense and were only used after low abundance/high variance filtering (D and F, respectively).

Random forest is the most effective untuned MLA for our data across weighting schemes

Our next goal was to find the best "out-of-the-box" MLA for our data. We selected seven common MLAs that were available in the Sci-Kit learn Python package: logistic regression, linear discriminant analysis, k-nearest neighbors, decision tree, random forest classifier, Gaussian naive Bayes, and support vector machines. We tested all seven MLAs and each weighting scheme on each metadata feature for each dataset. Testing on the Silva tree shows that there is a great variation in performance between MLA's however random forest is one of the best performers (**Figure 10** for Noguera-Julian's ETHNICTY) and similar results were seen for other trees (**Supplementary B** for all datasets, features, and trees).

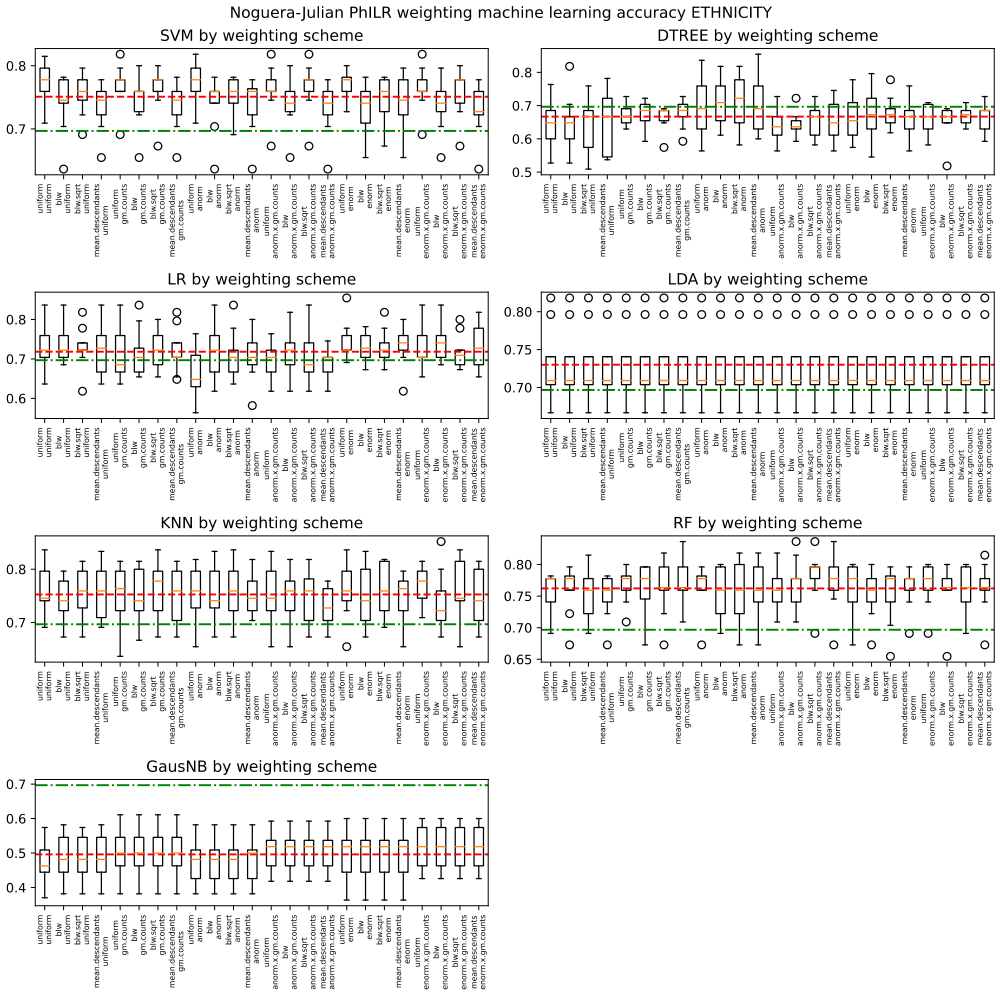

Figure 9: Boxplots of MLA results for Noguera-Julian's ETHNICITY show random forest (RF), K-nearest neighbor (KNN), and support vector machines (SVM) as the best performers. We recorded the accuracy of each MLA with 10 shuffles of 10-fold-cross validation. For this metadata feature, logistic regression (LR), linear discriminate analysis (LDA), decision tree (DTREE), and Gaussian naïve Bayes (GausNB) were the MLAs tested. Red dashed line indicates average MLA score for the specific MLA and the green dot-dashed line represents the average for the metadata feature ETHNICITY. Shown is the result for the Silva PhILR. Remaining plots for the Noguera-Julian dataset and other datasets available in ***Supplementary B.*** *The y-axis for each plot is the accuracy score for categorical meta-data or r-squared values for numerical data*

For each MLA and weighting scheme, we summarized the results of the dataset by taking the average of each metadata feature for each MLA. (**Figure 11** for the Jones dataset and others are in **Supplementary C**). We found that random forest consistently performed well – it was not always the highest performer, but at least always in the top three. When paired with random

forest, most weighting schemes (shown on the x-axis in Fig. 10) performed about the same, therefore picking the best one was difficult. In order to limit the scope of downstream analysis, we chose to use random forest and the combination of square-root of branch length (blw.sqrt) and Euclidean norm (enorm) for the PhILR weighting schemes.

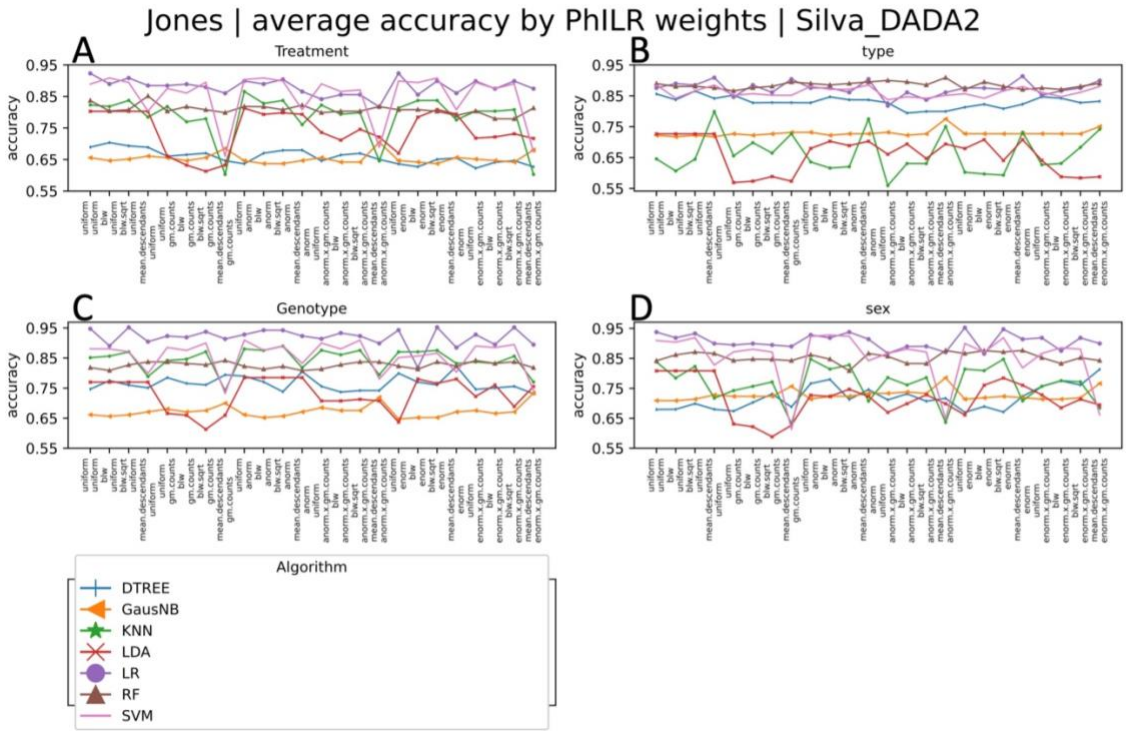

Figure 10: The Jones dataset shows random forest consistently performs well, as do all the PhILR weighting schemes.
*Here are treatment (A), type (stool vs swab), geneotype (C), and sex (D) of the Jones dataset showing the accuracy of each of our selected MLAs and PhILR transformations (**Other datasets in Supplementary C**). Shown are random forest (RF), K-nearest neighbor (KNN), and support vector machines (SVM) logistic regression (LR), linear discriminate analysis (LDA), decision tree (DTREE), and Gaussian naïve Bayes (GausNB). Each point represents an average of accuracy of each MLA with 10 shuffles of 10-fold-cross validation across a metadata feature.*

Log-norm has the highest average accuracy across all metadata categories

Having selected random forest as our model and blw.sqrt and enorm as our weighting

scheme for downstream analysis, we tested each transformation against each metadata feature of each dataset. We measured accuracy for 9 binary categories (such as "stool vs. swab" in the Jones dataset), 35 categories with multiple levels (such as ethnicity in the Vangay dataset) or 18 r-squared for quantitative variables (such as BMI in Jones dataset). We created 20 iterations out-of-bootstrap using 75% of the data for training and 25% for testing. This created 20 accuracy scores for each transformation and each metadata feature. As controls to our PhILR transformations, we created 5 random shuffles of the nodes of each tree and made PhILR transformations with them. This created 62 sets of boxplots, each with 24 transformations (All 62 random forest boxplots are in **Supplementary D**). As an example, we consider r-squared for a BMI and accuracy from "stool vs swab" from the Jones datasets (**Figure 12**). We see that lognorm in this example (gold bar) has a higher r-squared than most of the other transformations.

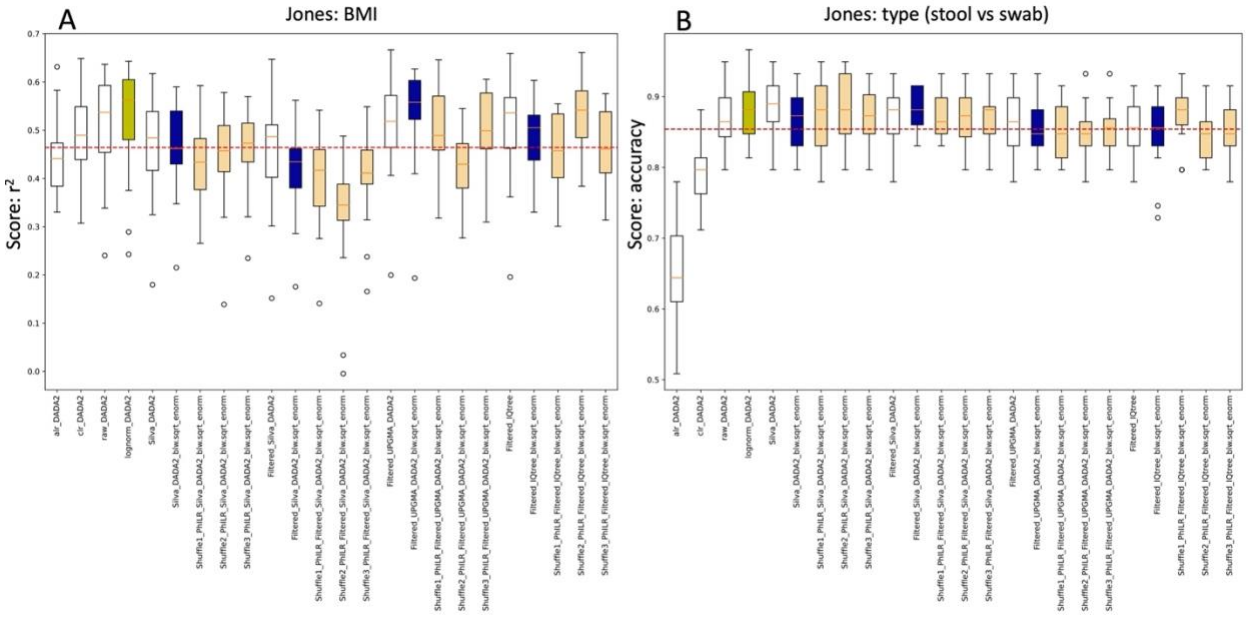

Figure 11*: Typical box and whiskers plots of random forest accuracy.*

This plot shows the scores of the random forest regressors on BMI in the Jones dataset (A). Since BMI is a numeric feature, the score for this feature approximates R^2, but for categorical data such as the "stool vs swab" feature of the same dataset (B) , the score will be the accuracy. The x-axis shows each transformation. Transformations that end in "blw.sqrt_enorm" are PhILR transformations where unshuffled (true) trees are blue and shuffled trees are orange. Lognorm is labeled in yellow and counts tables, alr, and clr are white. A similar plot was made for each metadata category. The red dashed line shows the mean for all the scores for this feature.

To summarize performance across all these transformations we averaged each of the 62 metadata features feature across each transformation. Examining the average of all of these 62 plots for each transformation (**Figure 13**) shows that log-norm on average yields a small improvement when compared to every other transformation and that, surprisingly, unnormalized data also out-performs nearly every other transformation but does not have an average accuracy as high as log-norm. None of the input options for PhILR, such as choice of tree, had much apparent effect on machine learning performance, except for alr and clr which, surprisingly, appeared to have noticeably worse scores.

To begin to address the statistical significance of the differences between different transformations, we compared specific transformations to each other by creating pairwise accuracy vs accuracy plots. To do this, we plotted the average score for each metadata feature for a given transformation against the average score of each metadata for another transformation – for this, we included both the accuracy and r^2 (**Supplementary D)**. In general, transformations yielded highly similar performance. For example, among the 276 separate pairwise comparisons, most comparisons have extremely high r^2 values such as the three pairwise comparisons examples shown in **Figure 14**. In general, across different choices of transformations including PhILR with shuffled trees, and the different methods for making phylogenetic trees to feed into PhILR, these transformations made little difference to performance and yielded scatter plots of

accuracy with high r-squared values (**Figure 14; Supplementary D**). However, we noticed that scatter plots involving the log-normalization transformation showed a small but consistent improvement across most of the 62 metadata categories when compared to other transformations (**Figure 15**).

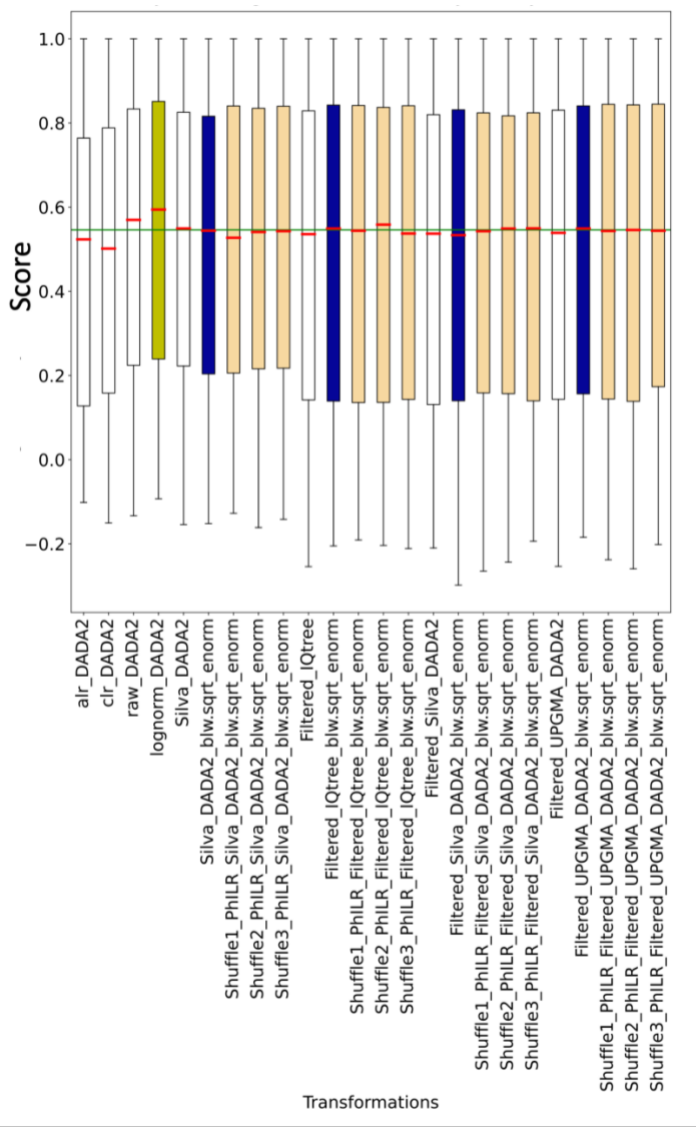

Figure 12: Lognorm has the highest accuracy.
These box and whisker plots show the average of all the points in each metadata feature for each transformation (62 points for each transformation). The red bars represent the median for each transformation and the green line represents the median of the entire dataset. Transformations that end in "blw.sqrt_enorm" are PhILR transformations where unshuffled (true) trees are blue and shuffled trees are orange. Lognorm is labeled in yellow and counts tables, alr, and clr are white. A similar plot was made for each metadata category

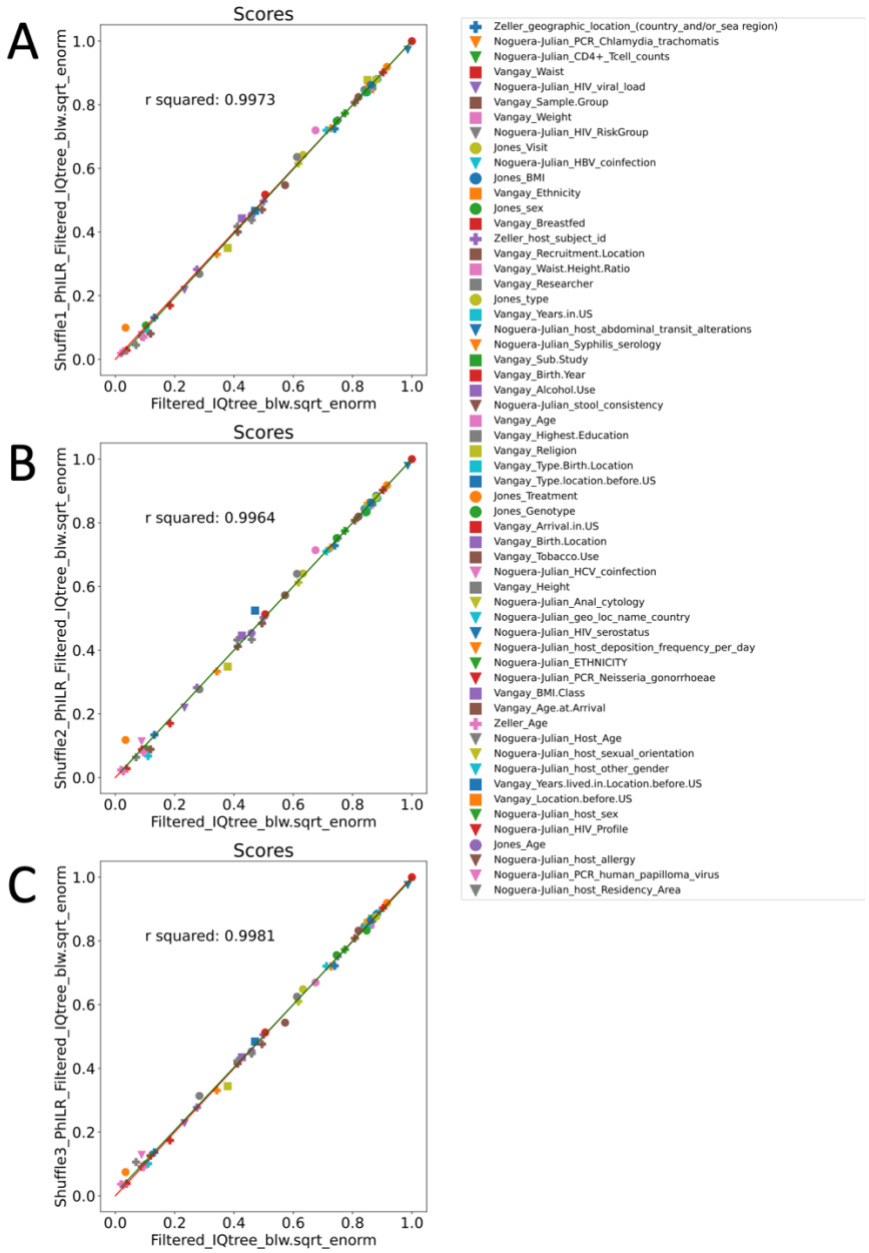

Figure 13: PhILR transforms with shuffled trees are as accurate as PhILR transforms made with "true" trees.
*The red line represents the sample space where both transformations have the same score. The green line is the best fit of the points. Shuffle1 of IQTREE PhILR (**A**), Shuffle2 of IQTREE PhILR (**B**), Shuffle3 of IQTREE PhILR(**C**), all show agreement with IQTREE PhILR. This legend (**D**) is the same for these plots and all the following accuracy vs. accuracy plots.*

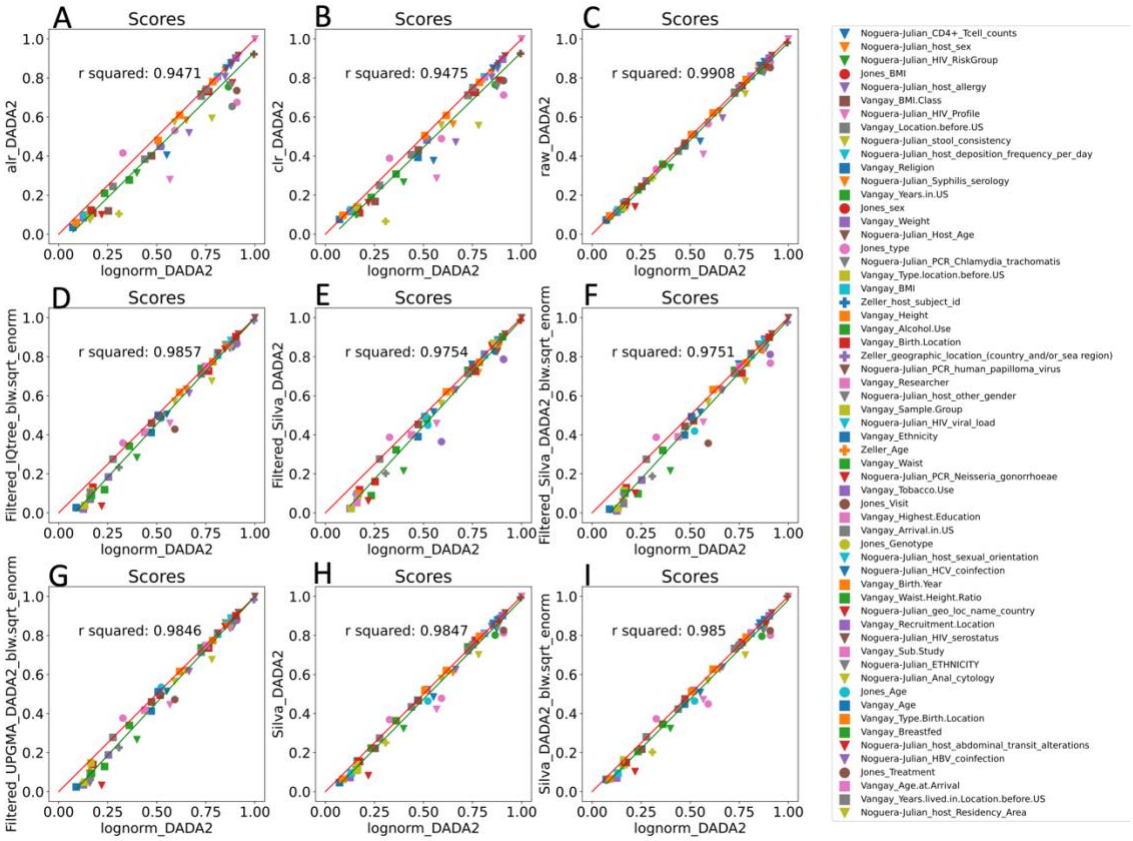

Figure 14: Accuracy vs accuracy plots indicate that lognorm provides the most accuracy for the random forest.
Points for each plot represent to scores of random forest classifier (accuracy) and random forest regressor (r^2). Lognorm performs favorably compared to filtered alr (A), clr(B), raw DADA2(C), Filtered IQTREE PhILR (D), filtered Silva DADA2 counts table (E), filtered Silva DADA2 PhILR (F) filtered UPGMA PhILR (G), Silva DADA2 counts table (H), Silva DADA2 PhILR (I).

In order to assess patterns of statistical significance, we next calculated pairwise Wilcoxon p-value for each of these plots comparing each normalization scheme against every other possible normalization scheme. For example, for log-norm (gold boxplot in Fig. 15), we report the results of the paired Wilcoxon test against every other normalization scheme. For each transformation we plotted the log of the p-value multiplied by the sign of the mean difference in score. This gives a visualization for our p-values that can indicate if the transformation is favorable or

unfavorable (**Figure 16**). From this, we can see that lognorm is clearly outperforming every other transformation. We can also see that alr is among the worst performance.

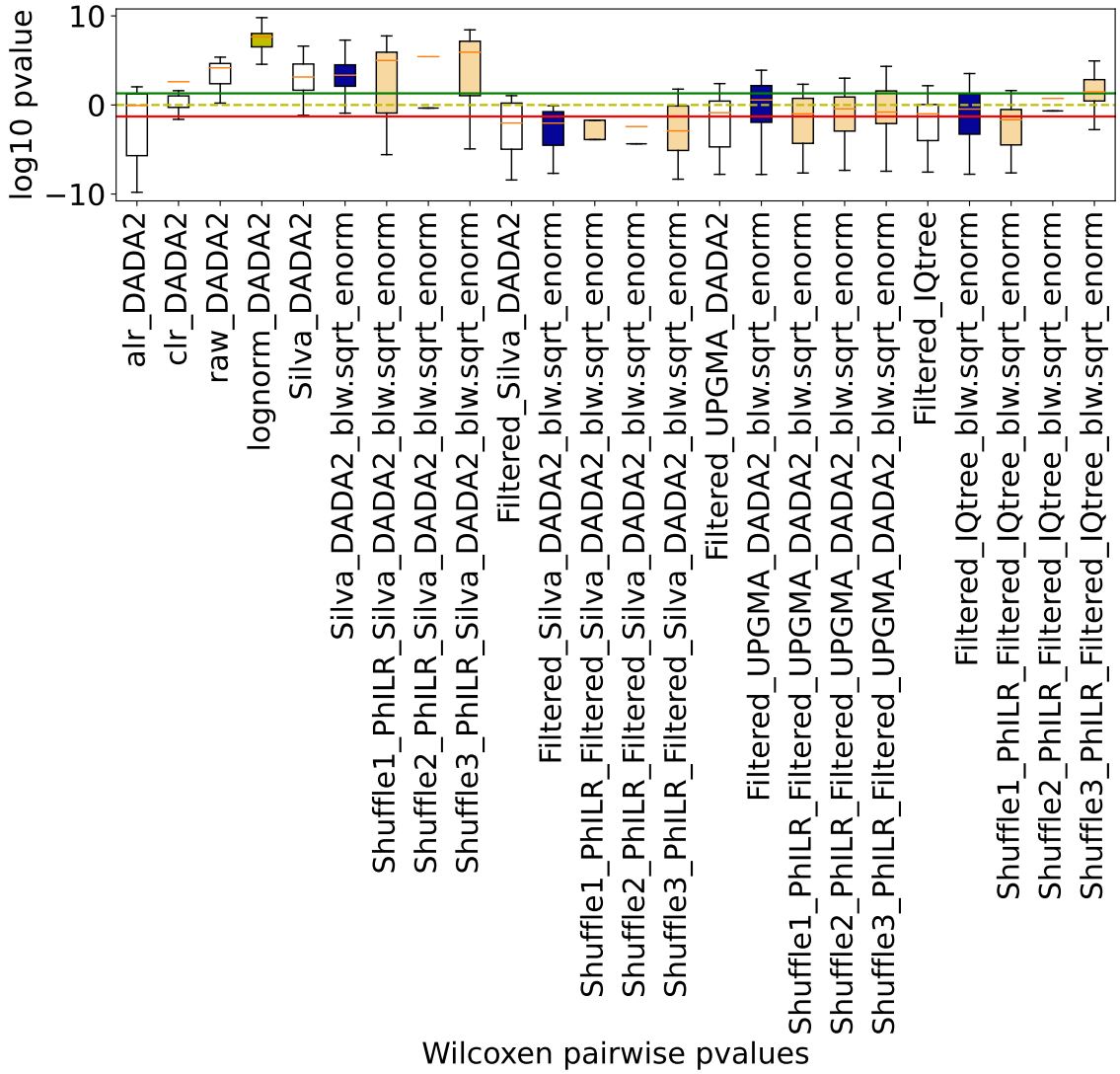

Figure 15: Pairwise p-values indicate that lognorm performs significantly better than every other transformation.
The y-axis shows log10 of the p-value and the x axis shows how well each transformation performed against all the others. A positive value indicates that the average accuracy of the given transformation is higher than others and a negative p-value indicates that the average accuracy is lower than the others. The area above the green line represents the sample space where the transformation is significantly better and the area below the red line represents the area where points are significantly worse.

These pairwise comparisons highlight the marked decrease in performance after low abundance/high variance filtering, as seen in the Silva DADA2 and filtered Silva DADA2 datasets. Finally, we can also see that filtering the DADA2 reads through the Silva reference database to make the Silva reference trees may have been beneficial.

Discussion

We tested 24 transformations 20 times against 62 metadata categories for a total of 29,280 random forest trials. This allowed us to make inferences on patterns with very small effect sizes. With this, we sought to answer a few questions:

- whether the ILR is an improvement over the simpler and more straightforward alr and clr,
- whether the arbitrary choices for PhILR input can improve the PhILR transform,
- and whether the compositionally aware transformations are an improvement over no transformation or a non-compositionally aware transformation.

Our results suggest that compositionally aware transformations may not be appropriate for training a random forest with ASV data, but that PhILR may have some advantages over clr and alr if you must transform (**Figure 12**). To be fair, in PhILR's debut article, Silverman et. al. acknowledge that random forest seems robust to the PhILR transform (Silverman et al. 2017). We tested all the available weighting schemes in PhILR and found them all to have relatively similar performance for most of our selected MLAs. This also allowed us to pick an effective weighting scheme and helped to limit the scope of our experiments, as our intention was to focus on the performance of transformations rather than optimizing our chosen MLA. We also felt that as out-of-the-box MLAs would provide an equal chance to each transformation, rather than one that optimized to a particular transformation.

We expected that the PhILR made from the IQTREE phylogenetic tree would be more accurate than the one that made from the UPGMA phylogenetic tree, as the hierarchical clustering used in UPGMA is much less sophisticated, however our results indicate that the choice for *de novo* tree made no difference, also evidenced by the PhILR transforms with randomly shuffled trees performing as well as those with true trees. The tree that improved PhILR was mapping our sequences to the Silva reference tree. This tree comes from a carefully curated database, so we believe that it improved our ASV count tables by filtering them, thus we saw the same improved accuracy in both the Silva PhILR and the Silva PhILR counts table. This leads us to believe that the quality of the tree used for PhILR is much less important than the quality of the counts table. We used high variance/low abundance filters to reduce the tree size but found that this was detrimental to accuracy in the case of the Silva reference tree PhILR and the counts tables. For future pipelines, alternative filters should be explored.

There are as many alr transformations as there are taxonomic features or ASV in each dataset, thus there were many more alr's available than the one that we tested. Thus, when we find that alr was our worst performing transformation, we also believe that the alr could be tuned better. It may be considered a short coming of this study that we did not test every possible version of alr for each dataset. However, we felt that we do our best to pick the best version of each transformation and limit the scope of this project, just as we did with choosing the MLA and the PhILR weighting scheme. We picked the version of the alr that we felt was best.

Comparing each compositional transformation may be moot, as both no transformation and lognorm provided as accurate as, if not more, results than any of the compositionally aware transformations (**Figure 15**). Thus, we believe that though the math behind compositionality is

compelling, that compositionality is not the biggest source of noise in sequencing data. In fact, as lognorm normalizes to read depth, we suspect that this is a likely source of artifacts and will study this further the next chapter.

Finally, although our trials indicate that lognorm provides better accuracy to the random forest, we believe that compositional transformations, especially balance trees, will still be useful for other purposes. It is possible that there are situations where compositional transformations can provide better classification than either lognorm nor raw counts tables can.

Conclusions

Thus far, our results indicate that the PhILR transform is an improvement on the alr and clr, but we do not believe that it is consistently better than the counts table or normalizing to read depth for use with a random forest classifier. Our results indicate that the quality of trees used for the PhILR transform do not matter for either the ilr transformation or the weighting scheme, as no "real" tree outperformed the series of random trees. We believe that the best course for obtaining accurate results from a random forest classifier is to use the untransformed counts table, the lognorm, or filter the counts tables using a curated database.

Further, we found that branch weighting worked no better for the accurate trees than it did for the random tree. No part weighting consistently performed better than any other nor did they consistently perform better than non-CODA transformations like the lognorm function or no transformation.

We believe that the Silva counts tables likely performs so well because its sequences have been filtered through the Silva database when it was created. This creates a dataset that is likely free of chimeras and "primer gunk" that can adulterate DNA sequencing datasets.

Additionally, we have come to suspect that the low abundance and high variance filters that we used on the “filtered” transformations may be reducing their performance. In fact, since the unfiltered transformations seem to be performing better, we believe that alternatives should be found and used.

Attributions

My role in this project was the conceptualization and experimental design, compilation of transformation methods from existing software packages, and evaluation of the algorithms. Dr. Fodor oversaw the work and aided in the conceptualization and experimental design.

Chapter 4: Reducing read depth artifacts

Abstract

It has long been known that read depth artifacts can be a confounding factor in inference. Here we examine 4 different publicly available datasets and find many important metadata variables that are significantly associated with read depth. We examine the results of 6 different normalization schemes on these data sets and evaluate the extent to which these schemes eliminate associations between read depth and the first 5 PCA axes. We find dramatic differences in the degree to which different normalization schemes produce PCA axes that are correlated with read depth. For the first, and presumably most important PCA axis, the lognorm transformation nearly eliminates associations with read depths and the r-squared associations between PCA1 and read-depth are consistent across different filter thresholds of sample exclusion. However, the results for other PCA axes vary widely with no clear single best normalization scheme and different datasets and normalization schemes showing different patterns in relationship to filter thresholds. We conclude that for the 1st PCA axis, the log-normalization scheme is clearly preferable, but investigators might need to consider different normalization algorithms in a case-by-case basis before performing inference on different datasets. In the near term, we will expand this research by (i) examining different distance metrics such as Bray-Curtis, (ii) determining to what extent variation in the first PCA and PCoA axes are driven by differences in Shannon diversity, (iii) determine how the commonly used

normalization scheme rarefication interacts with read depth artifacts and (iv) considering explicitly how different normalization schemes change inference on key metadata variables.

Introduction

As outlined in chapter 2, compositionality plays a big role in affecting the "shape" of microbiome sequencing data. For 16S sequencing data, the count tables are not true count data. This is because for each sample the taxa add up to the read depth. Read depth is the number of reads that are found in a DNA sequencing sample (Gloor et al. 2017). All the samples in the dataset will have a read depth, but it is unlikely that any read depth will be repeated, except in problematic samples where the depth is zero due to sequencing errors. This constraint creates the condition for compositionality in the data. Compositional data must be handled with particular care, otherwise spurious correlations will result (Pearson 1897). So, clearly the read depth is also an important aspect in shaping the data. However, the role that the read depth threshold plays in an applied bioinformatics analysis of microbiome data is often overlooked.

Microbiome datasets are sparse

Lately, the compositionality of sequencing count data is widely investigated and methods of transforming the data to remove compositional artifacts are continuously discovered and discussed (Tsilimigras and Fodor 2016; Fernandes et al. 2013; Love, Huber, and Anders 2014; Silverman et al. 2017). In addition to its compositionality, Sequencing count data has another inherent characteristic that makes analysis difficult - the data are sparse, meaning that the count tables are populated mostly by zeros. For example, using the raw Jones *et al* dataset as a typical example, we can see that zeros are the most common count for any given taxa (**Figure 17**). In

human metagenomic datasets, this sparsity is due to the absence of many taxa across samples. With this sparsity, the geometric mean of most taxa can be zero or close to zero. If it is zero, a transformation such as the clr or ilr that requires division by the geometric mean will be undefined. A common solution to this is to add a small number, or pseudo count to each element (Tsilimigras and Fodor, 2016). However, compositionality is not the only problem of SEQUENCING count data.

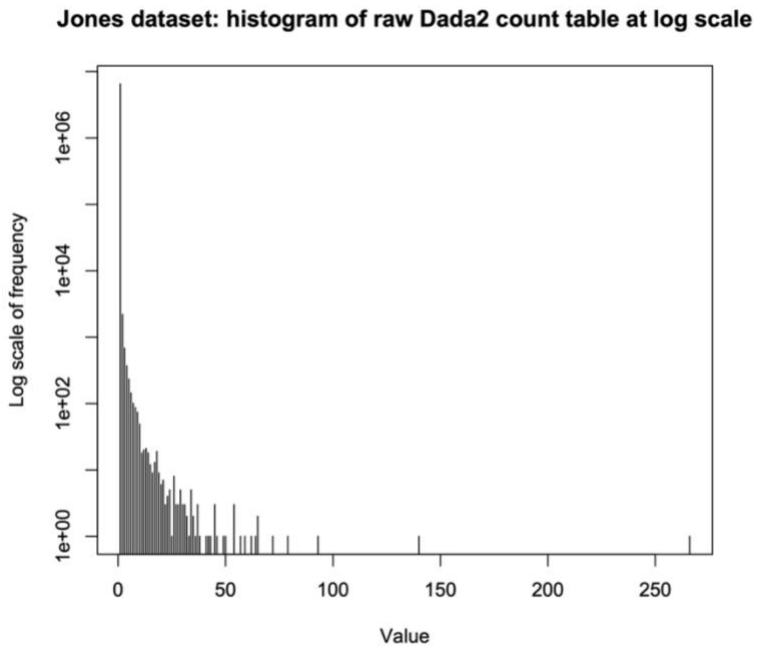

Figure 16: Microbiome datasets are sparse. The majority of the counts are zeros or ones in the Jones et al dataset.
This histogram shows the frequency of counts on the y axis and the log of the value of counts on the x axis. The zeros are by far the most frequent values.

Sparcity is related to read depth

Different samples will have different total amounts of sequences, and this creates a problem for differential analysis (such as comparison of treatment to control). Due to the fact SEQUENCING data reads suffer from coverage bias, meaning that they do not have a uniform distribution for all

the taxa that might be present in the sample. This is due to the variance in the binding affinity of PCR primers to taxa with different levels of GC content or specific motifs during PCR amplification (Ross *et al.*, 2013). Thus, different taxa may not amplify as well as others for this reason or an unknown reason, biasing the count tables. Further, specific samples may have a lower read depth due to the presence of compounds that interfere with PCR or sequencing, a lower quality of DNA, damage to the sample at any point before it was sequenced, or biological variability. Any of these, or a random chance could reduce the samples read depth.

Some transformations attempt to correct for sparsity and composition

Given the problems with sparsity and composition, a transformation may try to elucidate whether a given zero was due to its absence in its sample or if it was present in too small amounts for it to be sampled and treat each type of zero differently. The ALDEx2 R package creates such predictions using Bayesian methods (Fernandes et al. 2013). In our experiments, we use a function from this package that does this and then transforms the resulting data with the clr transformation.

Another relatively recent technique is to use the phylogenetic data encoded in the raw sequences to inform the ratios in the ilr, such as the one performed by the R package PhILR (Silverman et al. 2017) or the python package gniess (Morton et al. 2017). ILR transformations were the topic of the previous dissertation chapter.

Samples with low read depth are typically removed

New data transformations constantly evolve to deal with the complex issues of composition and sparsity (Fernandes et al. 2013). Often new transformations increase in complexity and

computational burden, attempting to shape our data in such a way that it is free of artifacts. Each transformation will give the data a different shape. The way these transformations change our data has been the focus of considerable attention in the literature (Fernandes et al. 2013; Silverman et al. 2017; Egozcue et al. 2003; Knight et al. 2018). In this chapter we will argue that a seemingly minor decision, where to set the thresholds for removing samples, can impact the data in surprisingly ways. Since PCR, which is the part of the sequencing process, is known to have some chance of failing (Hansen *et al.*, 1998), samples with low read depth are often suspected of having damage to the samples or other problems with the sequencing. To deal with this, data analysts often remove samples with the least number of raw sequences. But currently there is no standard for what read depth to use as a threshold.

Materials and Methods

We choose publicly available 16S microbiome SEQUENCING datasets for our analysis as these datasets are comparatively small and easily available. We looked for datasets with enough samples that outliers would not overly influence the results (**Table 4**).

Table 4: List of datasets in this project with their sample number, mean read depth, and highest read depth (in a single sample).

Short Name	Number of Samples	Mean read depth ± standard deviation	Highest read depth
Vangay (Vangay et al. 2018)	634	11,903 ± 15,601	267,454

Jones (Jones et al. 2018)	233	103,848 ± 143,025	1,734,468
Zeller (Zeller et al. 2014)	226	218,538 ± 142,363	752,310
Noguera-Julian (Noguera-Julian et al. 2016)	700	865,289 ± 105,301	1,360,527

Sequence processing

For 16S sequencing, we only used the forward reads. We filtered, trimmed, removed bimeras, and assigned taxonomy to the 16S sequences with the R package, DADA2 (Callahan *et al.*, 2016).

Data transformations

Several data transformations were used in this project. All of the transformations are freely available as R packages (**Table 5**). We used the raw output from Dada2 as our starting point for transformations.

Table 5: Data transformations that are compared in chapter 4

Transformation	R packages	Version	Description
raw sequences	DADA2 (Callahan et al. 2016)	1.14.1	raw output of dada2 with no transformations

Alr	rgr (Garrett 2013)	1.1.15	additive log ratio
Clr	rgr (Garrett 2013)	1.1.15	center log ratio (see table 1)
log normalization	None	None	(See table 1)
PhILR transform	PhILR (Silverman et al. 2017)	1.12.0	Phylogenetic Isometric Log-Ratio Transform
aldex2.clr	ALDEx2 (Fernandes et al. 2013, 2)	1.18.0	Generate Monte Carlo samples of the Dirichlet distribution for each sample. Convert each instance using the centered log-ratio transform

To build the phylogenetic trees for the PhiLR transformation, the resulting ASVs were aligned using version 2.0.2 of the R package DECIPHER (Wright, Yilmaz and Noguera, 2012).
Our pipeline for mapping the sequencing data of each of our datasets to a reference tree relies on several tools and R packages. The reference tree comes from Silva's Living tree project, 16S rRNA-based LTP release 132 (Munoz *et al.*, 2011). The reference tree lists the GenBank locus at each tip, so we used this information to download the sequences from GenBank using the ape package. We then built a blast database out of the sequences and blasted the sequences from our study datasets using custom BASH scripts. If the resulting matches had e-value greater or equal to 10^{-10}, we removed them from the tips of the reference tree using custom R scripts to get a

customized reference tree. This version of PhILR was used because it had performed well in the previous chapter.

Statistical tests

Spearman's correlations were calculated with R's inbuilt cor.test function.

P-values were calculated as specified using either an ANOVA R's inbuilt anova function or a Student's t-test using R's inbuilt t.test function.

The principal components analysis (PCA) was calculated using R's built-in prcomp function from the stats package. Area under the curve (AUC) was approximated using the trapezoidal method with the trapz function from the pracma R library (version 1.9.9).

All R code and taxonomic tables used for this project are available in the GitHub repository and can be found here: https://github.com/palomnyk/read_depth_artifacts.

Results

Read depth significantly correlates to sample features

In the publicly available datasets that we used for this study, we tested for a relationship between the dataset features and read depth. To do this we compared categorical features with read depth using ANOVA (**Table 7**) and compared continuous features with read depth using Spearman's correlation (**Table 8**). We found that in each dataset, the read depth had a strong relationship to many of the traits of the samples, such as age, geographic location, ethnicity, and many more. Interestingly, features do not necessarily have the same relationship with read depth between datasets. For example, in the Jones and Vangay datasets, the correlation between age and read depth was -0.1099, and -0.0989, respectively, and only -0.0248 in the Zeller dataset. In the

Vangay dataset, the p-value for ethnicity and read depth was 0.4164, whereas it was 0.0114 in the Noguera-Julian dataset.

Some traits should not have a significant relationship with read depth but do. For example, in the Jones dataset, whether the patient was given a treatment or a placebo should be randomized yet it has a significant relationship with read depth.

Table 6: Relationship between dataset categorical features and sample read depth. P-values calculated using ANOVA.

Dataset	Feature	p-value	pval < 0.05
Jones	Genotype	0.1778	FALSE
Jones	Sex	0.1754	FALSE
Jones	Treatment	**0.025**	TRUE
Jones	Visit	0.2672	FALSE
Jones	Type (Stool vs. Swab)	0.756	FALSE
Zeller	host_subject_id	**<0.0001**	TRUE
Zeller	geographic_location_.country_and.or_sea.region.	**<0.0001**	TRUE
Vangay	Recruitment.Location	**<0.0001**	TRUE
Vangay	Researcher	0.1079	FALSE
Vangay	Sub.Study	**0.0056**	TRUE
Vangay	Highest.Education	**0.036**	TRUE
Vangay	Ethnicity	0.4164	FALSE
Vangay	Religion	0.9707	FALSE
Vangay	Birth.Location	**<0.0001**	TRUE
Vangay	Type.Birth.Location	0.348	FALSE
Vangay	Arrival.in.US	**<0.0001**	TRUE
Vangay	Location.before.US	**<0.0001**	TRUE
Vangay	Type.location.before.US	0.1548	FALSE
Vangay	Years.lived.in.Location.before.US	0.9978	FALSE
Vangay	Tobacco.Use	0.3712	FALSE
Vangay	Alcohol.Use	0.1753	FALSE
Vangay	BMI.Class	0.9455	FALSE
Vangay	Breastfed	0.9974	FALSE

Vangay	Sample.Group	**0.027**	TRUE
Noguera-Julian	ETHNICITY	**0.0114**	TRUE
Noguera-Julian	geo_loc_name_country	**<0.0001**	TRUE
Noguera-Julian	HIV_RiskGroup	**0.0356**	TRUE
Noguera-Julian	HIV_serostatus	0.5242	FALSE
Noguera-Julian	host_other_gender	0.1223	FALSE
Noguera-Julian	host_sex	0.0519	FALSE
Noguera-Julian	HIV_Profile	**<0.0001**	TRUE
Noguera-Julian	PCR_human_papilloma_virus	**<0.0001**	TRUE
Noguera-Julian	host_allergy	**<0.0001**	TRUE
Noguera-Julian	host_abdominal_transit_alterations	**<0.0001**	TRUE
Noguera-Julian	host_Residency_Area	**<0.0001**	TRUE
Noguera-Julian	HCV_coinfection	**<0.0001**	TRUE
Noguera-Julian	Anal_cytology	**<0.0001**	TRUE
Noguera-Julian	host_sexual_orientation	**<0.0001**	TRUE
Noguera-Julian	Syphilis_serology	**<0.0001**	TRUE
Noguera-Julian	HBV_coinfection	**2E-04**	TRUE
Noguera-Julian	PCR_Neisseria_gonorrhoeae	**<0.0001**	TRUE
Noguera-Julian	PCR_Chlamydia_trachomatis	**<0.0001**	TRUE
Noguera-Julian	HIV_viral_load	**<0.0001**	TRUE
Noguera-Julian	stool_consistency	**<0.0001**	TRUE

Table 7: Relationship between dataset numeric features and sample read depth. P-values calculated using Spearman's correlation.

datatset	feature	Spearman's R
Jones	Age	-0.1099
Jones	BMI	0.0428
Zeller	Age	-0.0248
Vangay	Birth.Year	0.0989
Vangay	Age	-0.0989
Vangay	Years.in.US	0.2788
Vangay	Height	0.1548
Vangay	Weight	0.0907

Vangay	Waist	-0.0115
Vangay	BMI	0.0186
Vangay	Age.at.Arrival	-0.1599
Vangay	Waist.Height.Ratio	-0.0667
Noguera-Julian	Host_Age	0.0053
Noguera-Julian	host_deposition_frequency_per_day	-0.0049
Noguera-Julian	CD4._Tcell_counts	0.1834
Noguera-Julian	leukocytes	-0.1213
Noguera-Julian	lymphocytes	-0.0856
Noguera-Julian	host_body_mass_index	-0.0511

Read depth correlates with PCA axes

Having established that the read depth interacts with the features of the samples, we wanted to see how read depth correlated to the first five components of a principal component analysis (PCA). A PCA creates single dimensional slices of multidimensional data and is used for dimensional reduction. We performed a PCA analysis using the raw DADA2 output. For PCA axes 1-5, we calculated the Spearman correlation to read depth. We found that, for each dataset, nearly every axis had a strong correlation to read depth and that PCA1 had the strongest correlation (**Table 8**). The first component (PCA1), or slice, accounts for the greatest possible variance in the data, thus, a strong correlation to read depth would indicate that read depth was having a strong effect on the data. We consider the high correlation between read depth and PCA1 to be a "read depth artifact".

Table 8: Correlation between PCA axes and Read depth of raw DADA2 output.

dataset	PCA1	PCA2	PCA3	PCA4	PCA5
Jones	-0.748	0.660	0.400	-0.357	0.414

Vangay	-0.751	-0.192	-0.307	-0.447	-0.587
Noguera-Julian	0.760	-0.522	-0.301	0.668	0.358
Zeller	0.494	0.441	-0.345	-0.246	0.449

Read depth thresholds and transformations alter correlation between PCA1 and read depth

Since it appears that read depth is driving variance, we sought to see if we could reduce the read depth artifacts though read depth thresholds and common data transformations.

To do this, we set several thresholds for read depth on our raw data. We selected these thresholds as proportions of the dataset's median read depth because each dataset has a unique read depth. We selected proportions median rather than the mean read depth because some of the datasets have a very high variance and the median is known to be more robust to outliers than the mean. We selected 0.55 of the median as our highest threshold because we felt that it was still low enough for the dataset to have practical uses

We selected several popular transformations (**Table 5**) to test at various read depths. At each threshold, we discarded samples that did not meet that threshold and transformed the remaining samples using each of our selected transformations. For each transformation, we performed a PCA and plotted the Spearman correlation between PCA1 and the read depth (**Figure 18**). This was repeated for our various datasets.

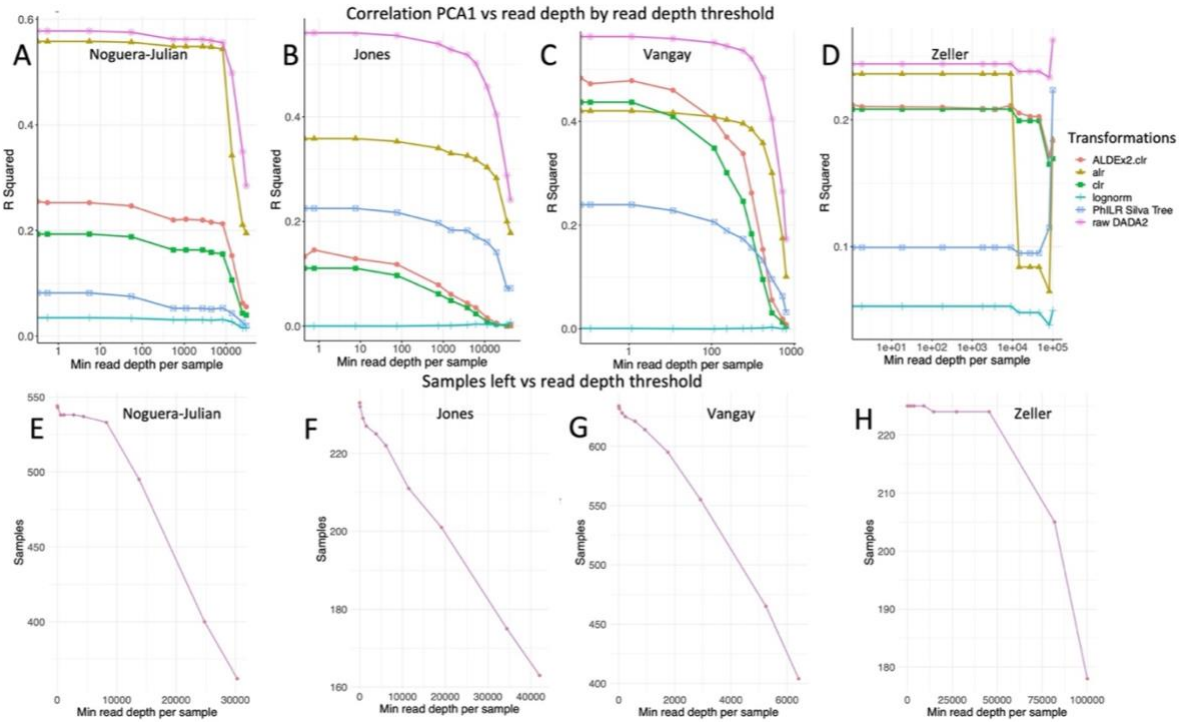

Figure 17: The shape of the data after transformation is dependent on the threshold for read depth and choice of transformation.
The *top row of* plots show the Spearman R2 on the Y axis and the *read depth thresholds* on the X axis *for the Noguera-Julian (A), Jones (B), Vangay (C), and Zeller (D) datasets. For each dataset, the points were selected as 0, 0.00001, 0.0001, 0.001, 0.01, 0.02, 0.05, 0.08, 0.15, 0.25, 0.45, and 0.55 of the median read depth of each dataset. The bottom row plots the number of samples in the dataset at a given read depth threshold for the Noguera-Julian (E), Jones (F), Vangay (G), and Zeller (H) datasets.*

For most transformations and datasets, there was a noticeable decrease in the correlation between read depth and PCA1 of that transformation as the read depth threshold was increased. A notable exception to this rule was the Zeller dataset, for which the threshold removed very few samples (**Figure 18H**). Another exception was the lognorm transform has almost no correlation with PCA1 at any read depth in any dataset. We found that raw DADA2 output had the most read depth artifacts and alr often had the second most read depth artifacts with PCA1. ALDEx2, clr, and PhILR had the intermediate number of read depth artifacts and their order changed based on the dataset.

PCA2 shows lognorm as having a strong correlation with read depth for all the datasets except Jones, where the correlation remains low (**Figure 19**). On the other hand, at this PCA axis, there is no clear best or worse choice for a transformation that reduces read depth artifacts that is not overfit for a single dataset. PCA3 seems to show that alr and raw DADA2 have more read depth artifacts and compared to the other transformations, which seem to reduce it.

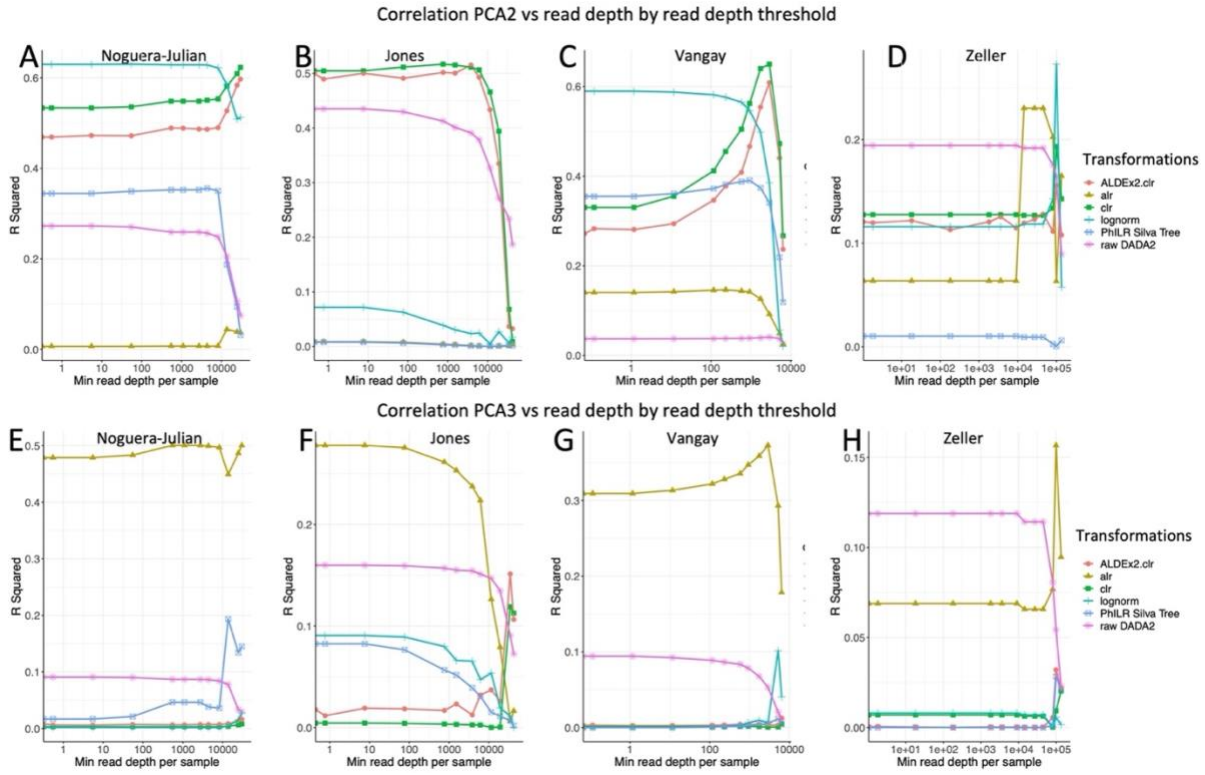

Figure 18: The correlation between PCA2 and PCA3 and read transformation is dependent on the threshold for read depth and the transformation.
The *top row of* plots show the Spearman R2 *between read depth and PCA2* on the Y axis and the *read depth thresholds* on the X axis *for the Noguera-Julian (A), Jones (B), Vangay (C), and Zeller (D) datasets.* The *bottom row of* plots show the Spearman R2 *between read depth and PCA3* on the Y axis and the *read depth thresholds* on the X axis *for the Noguera-Julian (E), Jones (F), Vangay (G), and Zeller (H) datasets. For each dataset, the points were selected as 0, 0.00001, 0.0001, 0.001, 0.01, 0.02, 0.05, 0.08, 0.15, 0.25, 0.45, and 0.55 of the median read depth of each dataset.*

Raw DADA2 consistently has the most or second most read depth artifacts at PCA4 and PCA5 and for most of the datasets, the other transformations have very few read depth artifacts overall (**Figure 21**).

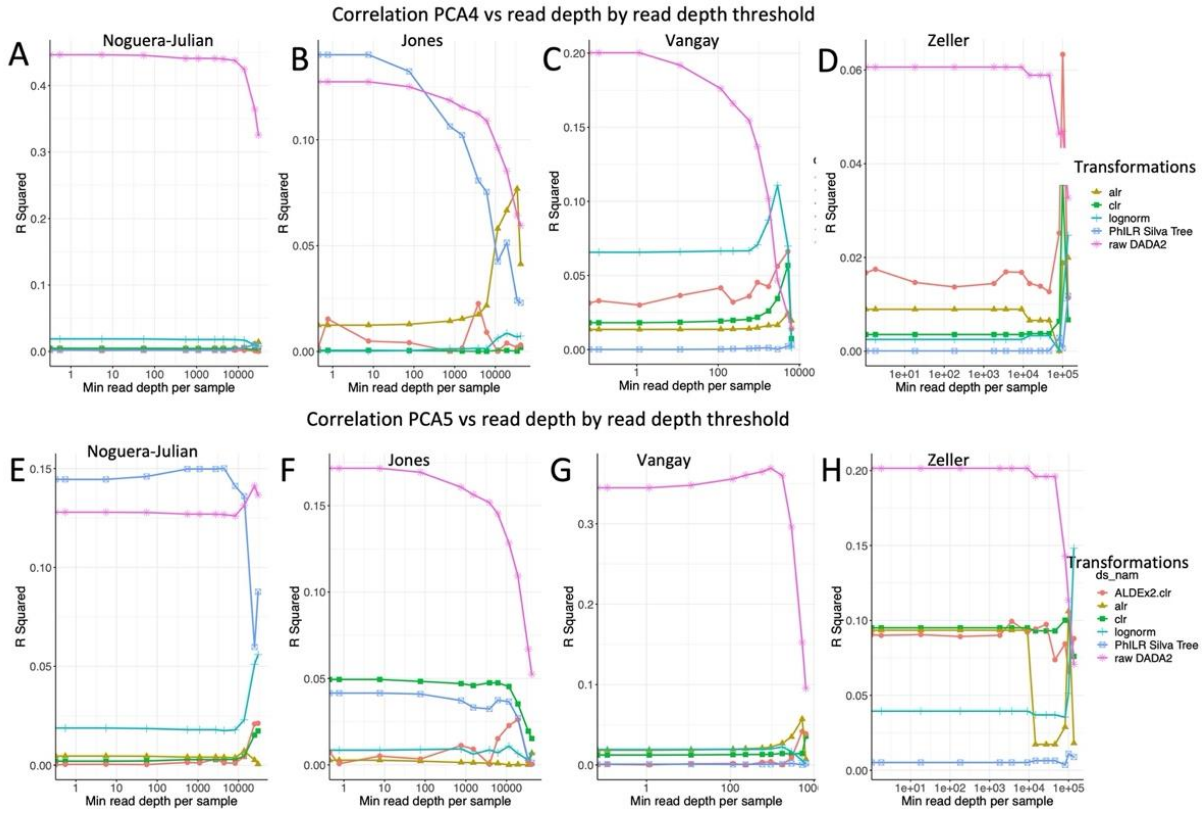

Figure 19: The correlation between PCA4 and PCA5 and read transformation is dependent on the threshold for read depth and the transformation.
The *top row of* plots show the Spearman R2 *between read depth and PCA4* on the Y axis and the *read depth thresholds* on the X axis *for the Noguera-Julian (A), Jones (B), Vangay (C), and Zeller (D) datasets.* The *bottom row of* plots show the Spearman R2 *between read depth and PCA5* on the Y axis and the *read depth thresholds* on the X axis *for the Noguera-Julian (E), Jones (F), Vangay (G), and Zeller (H) datasets. For each dataset, the points were selected as 0, 0.00001, 0.0001, 0.001, 0.01, 0.02, 0.05, 0.08, 0.15, 0.25, 0.45, and 0.55 of the median read depth of each dataset.*

To summarize all 5 PCA axes, we calculated the area under the curve (AUC) for each transformation at each PCA axis and plotted this for each dataset (**Figure 21**). The AUC describes area under the line segment created by R^2 of the given transformation against the read

depth thresholds of the datasets. We found that lognorm may have more correlation to read depth on other axes than PCA1. Our results show that no transformation is the best or worst for each axis on each of our datasets. Thus, we conclude that none of our selected transformations reduce read depth artifacts at every PCA axis.

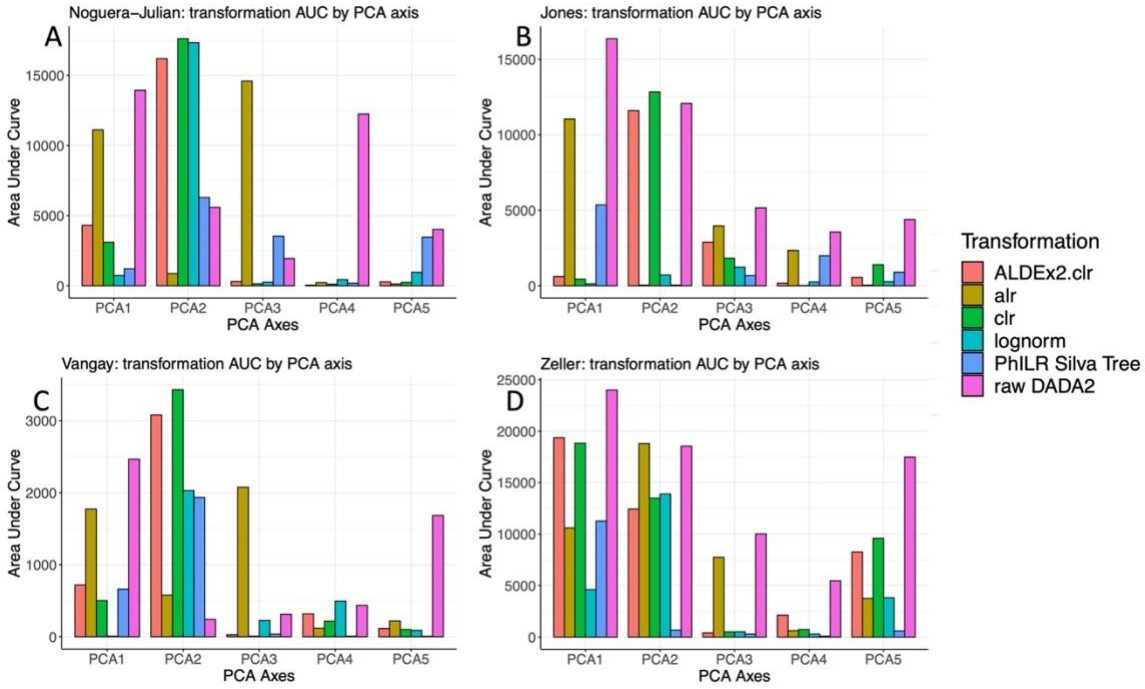

Figure 20: Lognorm has the least AUC at most, but not all PCA axes.
For each plot, the bars represent the area under the curve (AUC) of each transformation at a given PCA axis. The AUC describes area under the line created by R^2 of the given transformation against the read depth thresholds of the datasets. The read depth thresholds were 0, 0.00001, 0.0001, 0.001, 0.01, 0.02, 0.05, 0.08, 0.15, 0.25, 0.45, and 0.55 of the median read depth of each dataset.

Discussion

Read depth threshold greatly impacts the shape of the data

In the case of some of the traits, such as age, one could argue that during one's life the number and composition of bacteria in our gut shifts as we age (Yatsunenko et al. 2012), thus, this is not unexpected as read depth can be influenced by change in bacterial composition as primers can be biased towards one type of community.

However, it is also known that there is an arbitrary aspect to read depth (Gloor *et al.*, 2017), thus it may be useful to remove it as a driver of variance and a source of noise, as doing so may help uncover more subtle relationships.

Lognorm may reduce read depth artifacts

In the previous chapter, we found that lognorm outperformed compositionally aware transformations on machine learning applications. One of the reasons for this performance may be the reduction of read depth artifacts. In this chapter, we show that lognorm reduces read depth artifacts in PCA1. As PCA1 explains the most variance, this may lead to less noise in the data. The reason that lognorm is likely doing this, is that read depth is a term in lognorm's formula that is simply divided away (**Table 1**)**.** In addition to being a very simple transformation, lognorm is likely reducing a noise that is common for all 16S datasets.

Though read depth as an artifact is not eliminated from all PCA axes by lognorm, there is still merit to in respect to PCA1, as it is the axis that most describes the variance in the dataset. Eliminating it here can greatly reduce the noise in the dataset and allow researchers to find

relationships in the data that had be previously masked by the noise. To our knowledge, no other transformation tries to solve the read depth noise issue.

Read depth artifact analysis provides a new tool for evaluating new transformations and for selecting a threshold to remove low abundance samples

In this chapter, we have provided a novel tool for visualizing both the effects of transformations/normalizations and read depth threshold in microbiome studies. Thus, when evaluating tools in the future, it may be helpful to know if they reduce the effects of read depth on PCA1.

Read depth artifact analysis provides a new tool for evaluating thresholding for samples. One could use the data from Figure 18 to choose a read depth threshold in such a way that maximizes the number of samples used in a study while also reducing low abundance samples.

An alternative way to reduce read depth artifacts is rarefaction. Rarefication is a technique to equalize the number of reads in each sample. To do this, a read depth is selected, often arbitrarily, and, for each sample, reads are randomly sampled until that read depth is reached. If a sample does not meet the required read depth, it is removed from the dataset. For the price of losing some data, and possibly samples, the dataset will have an equal read depth for all samples. When rarefaction is done there is an understood bias of samples with higher read depths having more taxa (Cameron et al. 2021). This is thought to help in calculating diversity measures such as Shannon diversity. Future studies will involve a comparison of our current methods to rarefaction.

Conclusions

We find that the read depth filter plays a large role in the microbiome analysis that is not currently being discussed. We find that read depth strongly drives the variance in PCA1. The lognorm data transformation may help reduce the effect of read depth on PCA1. We recommend that analysts of similar data check their data for read depth artifacts to see how their read depth threshold and transformation affects their data.

Attributions

My role in this project was the conceptualization and experimental design, compilation of transformation methods from existing software packages, and evaluation of the algorithms. Dr. Fodor oversaw the work and aided in the conceptualization and experimental design

References

Aitchison, J. 1982. “The Statistical Analysis of Compositional Data,” no. 2: 40.

Aitchison, John. 1994. “Principles of Compositional Data Analysis.” *Lecture Notes-Monograph Series* 24: 73–81.

Brown, Matthew C. n.d. “The Microbiota Multiverse: From Gut to Brain and Beyond.” Ph.D., United States -- North Carolina: The University of North Carolina at Charlotte. Accessed August 31, 2021. https://www.proquest.com/docview/2212995092/abstract/30D79B9AC3AB4BA8PQ/1.

Callahan, Benjamin J., Paul J. McMurdie, Michael J. Rosen, Andrew W. Han, Amy Jo A. Johnson, and Susan P. Holmes. 2016. “DADA2: High-Resolution Sample Inference from Illumina Amplicon Data.” *Nature Methods* 13 (7): 581–83. https://doi.org/10.1038/nmeth.3869.

Cameron, Ellen S., Philip J. Schmidt, Benjamin J.-M. Tremblay, Monica B. Emelko, and Kirsten M. Müller. 2021. “Enhancing Diversity Analysis by Repeatedly Rarefying next Generation Sequencing Data Describing Microbial Communities.” *Scientific Reports* 11 (1): 22302. https://doi.org/10.1038/s41598-021-01636-1.

Egozcue, J. J., V. Pawlowsky-Glahn, G. Mateu-Figueras, and C. Barceló-Vidal. 2003. “Isometric Logratio Transformations for Compositional Data Analysis.” *Mathematical Geology* 35 (3): 279–300. https://doi.org/10.1023/A:1023818214614.

Fernandes, Andrew D., Jean M. Macklaim, Thomas G. Linn, Gregor Reid, and Gregory B. Gloor. 2013. “ANOVA-Like Differential Expression (ALDEx) Analysis for Mixed Population RNA-Seq.” *PLOS ONE* 8 (7): e67019. https://doi.org/10.1371/journal.pone.0067019.

Fodor, Anthony A., Erich R. Klem, Deirdre F. Gilpin, J. Stuart Elborn, Richard C. Boucher, Michael M. Tunney, and Matthew C. Wolfgang. 2012. “The Adult Cystic Fibrosis Airway Microbiota Is Stable over Time and Infection Type, and Highly Resilient to Antibiotic Treatment of Exacerbations.” *PLOS ONE* 7 (9): e45001. https://doi.org/10.1371/journal.pone.0045001.

Garrett, Robert G. 2013. “The ‘Rgr’ Package for the R Open Source Statistical Computing and Graphics Environment - a Tool to Support Geochemical Data Interpretation.” *Geochemistry: Exploration, Environment, Analysis* 13 (4): 355–78. https://doi.org/10.1144/geochem2011-106.

Gloor, Gregory B., Jean M. Macklaim, Vera Pawlowsky-Glahn, and Juan J. Egozcue. 2017. “Microbiome Datasets Are Compositional: And This Is Not Optional.” *Frontiers in Microbiology* 8. https://doi.org/10.3389/fmicb.2017.02224.

Jones, Roshonda B., Xiangzhu Zhu, Emili Moan, Harvey J. Murff, Reid M. Ness, Douglas L. Seidner, Shan Sun, et al. 2018. “Inter-Niche and Inter-Individual Variation in Gut Microbial Community Assessment Using Stool, Rectal Swab, and Mucosal Samples.” *Scientific Reports* 8 (1): 4139. https://doi.org/10.1038/s41598-018-22408-4.

Knight, Rob, Alison Vrbanac, Bryn C. Taylor, Alexander Aksenov, Chris Callewaert, Justine Debelius, Antonio Gonzalez, et al. 2018. “Best Practices for Analysing Microbiomes.” *Nature Reviews Microbiology* 16 (7): 410–22. https://doi.org/10.1038/s41579-018-0029-9.

Lin, Yun Chao, Ansaf Salleb-Aouissi, and Thomas A. Hooven. 2022. “Interpretable Prediction of Necrotizing Enterocolitis from Machine Learning Analysis of Premature Infant Stool Microbiota.” *BMC Bioinformatics* 23 (1): 104. https://doi.org/10.1186/s12859-022-04618-w.

Love, Michael I., Wolfgang Huber, and Simon Anders. 2014. “Moderated Estimation of Fold Change and Dispersion for RNA-Seq Data with DESeq2.” *Genome Biology* 15 (12): 550. https://doi.org/10.1186/s13059-014-0550-8.

Maltecca, Christian, Duc Lu, Constantino Schillebeeckx, Nathan P. McNulty, Clint Schwab, Caleb Shull, and Francesco Tiezzi. 2019. “Predicting Growth and Carcass Traits in Swine Using Microbiome Data and Machine Learning Algorithms.” *Scientific Reports* 9 (1): 6574. https://doi.org/10.1038/s41598-019-43031-x.

McMurdie, Paul J., and Susan Holmes. 2014. “Waste Not, Want Not: Why Rarefying Microbiome Data Is Inadmissible.” *PLOS Computational Biology* 10 (4): e1003531. https://doi.org/10.1371/journal.pcbi.1003531.

Morton, James T., Jon Sanders, Robert A. Quinn, Daniel McDonald, Antonio Gonzalez, Yoshiki Vázquez-Baeza, Jose A. Navas-Molina, et al. 2017. “Balance Trees Reveal Microbial Niche Differentiation.” *MSystems* 2 (1). https://doi.org/10.1128/mSystems.00162-16.

Munoz, Raúl, Pablo Yarza, Wolfgang Ludwig, Jean Euzéby, Rudolf Amann, Karl-Heinz Schleifer, Frank Oliver Glöckner, and Ramon Rosselló-Móra. 2011. “Release LTPs104 of the All-Species Living Tree.” *Systematic and Applied Microbiology* 34 (3): 169–70. https://doi.org/10.1016/j.syapm.2011.03.001.

Noguera-Julian, Marc, Muntsa Rocafort, Yolanda Guillén, Javier Rivera, Maria Casadellà, Piotr Nowak, Falk Hildebrand, et al. 2016. “Gut Microbiota Linked to Sexual Preference and HIV Infection.” *EBioMedicine* 5 (March): 135–46. https://doi.org/10.1016/j.ebiom.2016.01.032.

Paradis, Emmanuel, Julien Claude, and Korbinian Strimmer. 2004. “APE: Analyses of Phylogenetics and Evolution in R Language.” *Bioinformatics* 20 (2): 289–90. https://doi.org/10.1093/bioinformatics/btg412.

Pearson, Karl. 1897. “Mathematical Contributions to the Theory of Evolution.—On a Form of Spurious Correlation Which May Arise When Indices Are Used in the Measurement of Organs.” *Proceedings of the Royal Society of London* 60 (359–367): 489–98. https://doi.org/10.1098/rspl.1896.0076.

Pedregosa, Fabian, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, et al. n.d. “Scikit-Learn: Machine Learning in Python.” *MACHINE LEARNING IN PYTHON*, 6.

Randolph, Timothy W., Sen Zhao, Wade Copeland, Meredith Hullar, and Ali Shojaie. 2018. “KERNEL-PENALIZED REGRESSION FOR ANALYSIS OF MICROBIOME DATA.” *The Annals of Applied Statistics* 12 (1): 540–66. https://doi.org/10.1214/17-AOAS1102.

Shannon, C. E. 1948. “A Mathematical Theory of Communication.” *The Bell System Technical Journal* 27 (3): 379–423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x.

Silverman, Justin D., Kimberly Roche, Sayan Mukherjee, and Lawrence A. David. 2020. “Naught All Zeros in Sequence Count Data Are the Same.” *Computational and*

Structural Biotechnology Journal 18 (September): 2789–98. https://doi.org/10.1016/j.csbj.2020.09.014.

Silverman, Justin D, Alex D Washburne, Sayan Mukherjee, and Lawrence A David. 2017. "A Phylogenetic Transform Enhances Analysis of Compositional Microbiota Data." Edited by Anthony Fodor. *ELife* 6 (February): e21887. https://doi.org/10.7554/eLife.21887.

Sisk-Hackworth, Laura, Adrian Ortiz-Velez, Micheal B. Reed, and Scott T. Kelley. 2021. "Compositional Data Analysis of Periodontal Disease Microbial Communities." *Frontiers in Microbiology* 12. https://www.frontiersin.org/article/10.3389/fmicb.2021.617949.

Tsilimigras, Matthew C. B., and Anthony A. Fodor. 2016. "Compositional Data Analysis of the Microbiome: Fundamentals, Tools, and Challenges." *Annals of Epidemiology*, The Microbiome and Epidemiology, 26 (5): 330–35. https://doi.org/10.1016/j.annepidem.2016.03.002.

Vangay, Pajau, Abigail J. Johnson, Tonya L. Ward, Gabriel A. Al-Ghalith, Robin R. Shields-Cutler, Benjamin M. Hillmann, Sarah K. Lucas, et al. 2018. "US Immigration Westernizes the Human Gut Microbiome." *Cell* 175 (4): 962-972.e10. https://doi.org/10.1016/j.cell.2018.10.029.

Virtanen, Pauli, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, et al. 2020. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python." *Nature Methods* 17 (3): 261–72. https://doi.org/10.1038/s41592-019-0686-2.

Weiß, Michael, and Markus Göker. 2011. "Chapter 12 - Molecular Phylogenetic Reconstruction." In *The Yeasts (Fifth Edition)*, edited by Cletus P. Kurtzman, Jack W. Fell, and Teun Boekhout, 159–74. London: Elsevier. https://doi.org/10.1016/B978-0-444-52149-1.00012-4.

Yatsunenko, Tanya, Federico E. Rey, Mark J. Manary, Indi Trehan, Maria Gloria Dominguez-Bello, Monica Contreras, Magda Magris, et al. 2012. "Human Gut Microbiome Viewed across Age and Geography." *Nature* 486 (7402): 222–27. https://doi.org/10.1038/nature11053.

Zeller, Georg, Julien Tap, Anita Y. Voigt, Shinichi Sunagawa, Jens Roat Kultima, Paul I. Costea, Aurélien Amiot, Jürgen Böhm, Francesco Brunetti, and Nina Habermann. 2014. "Potential of Fecal Microbiota for Early-stage Detection of Colorectal Cancer." *Molecular Systems Biology* 10 (11): 766.