Winter 2022

Compositional Datasets and the Nested Dirichlet Distribution

Bianca Luedeker
biancaluedeker@gmail.com

COMPOSITIONAL DATASETS AND THE

NESTED DIRICHLET DISTRIBUTION

Approved by:

Dr. Monnie McGee
Associate Professor in Department of Statistical Science, Southern Methodist University

---

Dr. Ian Harris
Associate Professor in Department of Statistical
Science, Southern Methodist University

Dr. Charles South
Professor or Practice in Department of Statistical Science, Southern Methodist University

---

Dr. Jacob Turner (External)
Assistant Professor in Department of Mathematics and Statistics, Stephen F. Austin State University

COMPOSITIONAL DATASETS AND THE NESTED DIRICHLET DISTRIBUTION

A Dissertation Presented to the Graduate Faculty of the

Dedman College

Southern Methodist University

in

Partial Fulfillment of the Requirements

for the degree of

Doctor of Philosophy

with a

Major in Statistical Science

by

Bianca Luedeker

B.S. Ed., Northern Arizona University
M.S., Northern Arizona University

December 17, 2022

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Monnie McGee for her counseling services and statistical expertise. When I arrived at SMU after losing my job, I was filled with existential dread that haunted me during my four years here. Dr. McGee was there to pick up the pieces of my shattered soul and rearrange them into a being that was able to function long enough to complete this degree. I would also like express my gratitude to Dr. Ian Harris whose office I would visit when I was in a state of deep depression. I would like to thank Dr. Charles South for his encouragement along the way. I would like to thank the student who came before me, Dr. Jacob Turner, whose work was the road map for this dissertation. Lastly, I want to thank Sheila Crain for her help during my time at SMU and ensuring that I would be able to finish this degree while teaching at another university. I greatly appreciate everyone's patience with this host of a ghost I have become.

I am very grateful to my partner Zach Brutsche. I know I kept you up at night with panic attacks and melt downs. I would have expected you to leave by now, but hey, you're still here. Wow! *In Owen's Wilson's voice*

I would like to thank my parents for instilling me with blind ambition. I will continue to strive for the highest goals. Achieve at all costs! Thanks guys. *slow clap*

Luedeker, Bianca

B.S. Ed., Northern Arizona University
M.S., Northern Arizona University

Compositional Datasets and the
Nested Dirichlet Distribution

Advisor: Dr. Monnie McGee
Doctor of Philosophy degree conferred December 17, 2022
Dissertation completed November 18, 2022

Compositional data is a special type of multivariate data where each component of the data vector is sandwiched between 0 and 1 and the sum of the components must be 1. For example, the proportion of time that each of 7 mice spend in one of four quadrants of a circular water maze is between 0 and 1, and the total proportion of time spent in the maze is 1. In this case, the proportion of time spent in each quadrant of the maze is a "component". If there are two sets of mice, one set of normal mice and one set of cognitively impaired mice, the experiment has a two-sample design. Such data is frequently analyzed incorrectly by comparing the two samples via a t-test (or ANOVA for multiple samples) on one component of the vector at a time.

In this dissertation, this problem is corrected by analyzing compositional datasets using nested Dirichlet distributions, generalized versions of Dirichlet distributions that allow for positive correlations among components. Specifically, we extend a previous result of two-sample comparisons using Dirichlet distributions and nested Dirichlet distributions to multi-sample comparisons. The performance of the new test in terms of type I error rates and power is established using simulation studies. In addition, to use a nested model, an appropriate tree which describes the relationship between components must first be found. An existing data driven tree finding algorithm is improved upon by including an extra step that prunes unnecessary nodes using confidence intervals for the differences between parameters at each level of the tree. The tree finding algorithm and multi-sample

test are demonstrated on two datasets. The first dataset measures the proportion of home runs, triples, doubles, singles, outs, and other events for batters on professional baseball teams from three age groups. The second dataset compares the composition of job types in the United States by region.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

This thesis is dedicated to Dr. Monnie McGee.

Chapter 1
Compositional Data and Its Analysis

A compositional dataset of size n is any dataset composed of n k-dimensional vectors where the components of the vector measure parts of a whole. For example, the components may measure the amount of each type of mineral in a rock or the proportion of time spent on different daily activities for every person in a sample. The components of each vector must sum to 1. These vectors lie on the S^{k-1} simplex defined as

$$S^{k-1} = \left\{ (x_1, x_2, \ldots, x_k) : \min(x_j) \geq 0, \ \sum_{j=1}^{k} x_j = 1 \right\}$$

(Zhang and Dao, 2020). A characteristic of compositional data is that once the values of $k-1$ components are known, the k^{th} component is determined; therefore, components are not independent. Count data of mutually exclusive and exhaustive categories can be easily transformed into compositional data by dividing the counts in each category by the total count taken over all categories. For example, suppose we wanted to measure the proportion of students attending 50 universities broken down by class: freshmen, sophomores, juniors, seniors, and graduate students. The five components of each vector would be the proportion of students in each class. Thus, this would be an example of compositional data with $k = 5$ (the type of student) and $n = 50$ (the number of universities).

There are five different models that are commonly used in the literature when analyzing compositional data. These are the multinomial, Dirichlet, Dirichlet-multinomial, Dirichlet-tree distribution (nested Dirichlet), and the Dirichlet-multinomial tree distribution (nested Dirichlet-multinomial). Each of these will be described in turn in the remainder of this chapter.

1.1. Multinomial Distribution

Multinomial distributions are routinely used to model count compositional data; data that could be re-scaled as proportions but is left in count form (van den Boogaart and Tolosana-Delgado, 2013). An example of this type of data is typically seen in microbiome studies. Microbiome studies involve determining the microbial organisms present in a collection of biological material from an environment, such as the human body or seawater. Genomic fragments, called "reads", are extracted from biological material via high-throughput sequencing and these fragments are used to identify organisms, or taxa to which the organisms belong, within the material. The result is a vector of counts of each type of organism or taxa in the material, and these counts are assumed to make up the entire population of organisms within the material. Gao et al. (2021) explains various methods of extracting the genomic material, determining its origin, and analyzing the resulting data. There are many applications of microbiome studies. The most relevant to this dissertation is the use of reads as operational taxonomic units (OTUs) to classify the organisms in the sample by taxonomic groups. Microbiome datasets are compositional datasets because there is a total number of OTUs imposed by the sequencer; therefore, the proportion of each taxa within a given experiment should sum to 1 (Gloor et al., 2017).

Let n be the sample size and k be the number of unique operational taxonomic units (OTUs) (e.g. k different species, k different genera, etc.). Let $\boldsymbol{x_i} = (x_{i1}, x_{i2}, \ldots x_{ik})$ for $i = 1, 2, \ldots, n$ be the vector of taxa counts for sample i. Let $N_{i.} = \sum_{j=1}^{k} x_{ij}$ be the total count for sample i. For simplicity, we will assume that $N = N_{1.} = N_{2.} = \cdots = N_{n.}$. This is not an unreasonable assumption to make as usually the number of reads is fixed before the experiment is conducted (Yang et al., 2019). Let $\boldsymbol{\pi} = (\pi_1, \ldots \pi_k)$ be the actual proportions for each OTU. The multinomial model assumes

$$\boldsymbol{x_i} \overset{\text{i.i.d}}{\sim} \text{Multinomial}(N, \boldsymbol{\pi}). \tag{1.1}$$

The density is

$$f(\boldsymbol{x_i}|\boldsymbol{\pi}) = \binom{N}{\boldsymbol{x_i}} \prod_{j=1}^{k} \pi_j^{x_{ij}} \text{ where } x_{ij} \in \{0, \ldots N\},\ j \in \{1 \ldots k\},\ \sum_j x_{ij} = N. \tag{1.2}$$

There are two problems that can arise when using the multinomial model. The first is that the multinomial model assumes that $N_{i.}$ are equal for all i. While this assumption is often reasonable, in micorbiome studies, for example, it is possible that the number of reads, $N_{i.}$, varies from sample to sample. This can occur due to missing values or other errors when the number of reads is fixed, or it can be designed into the study, as when the number of reads is not fixed. The second problem is that a multinomial model does not accurately model overdispersion. In microbiome studies, taxa proportions vary from sample to sample and there is no parameter in the multinomial model to capture this variability (La Rosa et al., 2012). To address this issue, a hierarchical model, the Dirichlet-Multinomial distribution can be used. Before discussing the Dirichlet-Multinomial model, first the Dirichlet Distribution is introduced.

1.2. Dirichlet Distribution

The Dirichlet distribution is a common distribution that was used in the earliest studies of compositional data analysis (Ng et al., 2011). It is is often used as the conjugate prior to the multinomial distribution in Bayesian analysis (Gelman et al., 2003). Suppose a composition is made up of k variables. We will also refer to the k variables as components. The appropriate Dirichlet distribution will have k parameters $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_k)$. Denote this distribution as DD$(\alpha_1, \ldots, \alpha_k)$ and let $\boldsymbol{X} = [X_1, \ldots X_k]^T$ be a random vector distributed as DD$(\alpha_1, \ldots, \alpha_k)$. Let $A = \sum_{j=1}^{k} \alpha_j$. A is known as the precision. The α_j can be thought of as counts from a prior or a current study, depending on the context.

The density function is:

$$f(\boldsymbol{x}|\boldsymbol{\alpha}) = \frac{\Gamma(A)}{\Gamma(\alpha_1)\ldots\Gamma(\alpha_k)} \prod_{j=1}^{k} x_j^{\alpha_j - 1} \quad 0 \leq x_j \leq 1 \text{ for } j = 1, \ldots, k. \tag{1.3}$$

The expected value of a component is

$$\mathrm{E}[X_j] = \frac{\alpha_j}{A} = \pi_j \tag{1.4}$$

and the variance of a component is

$$\mathrm{Var}[X_j] = \frac{\alpha_j(A - \alpha_j)}{A^2(1+A)} = \frac{\pi_j(1-\pi_j)}{A+1} \tag{1.5}$$

as described in Turner (2013) and Ng et al. (2011).

Turner (2013) notes two drawbacks when using the Dirichlet distribution to model compositional data. One drawback is that Equation 1.5 implies that components with the same mean must also have the same variance. Another limitation is that the covariance between any two components is non-positive (Minka, 1999). Specifically:

$$\mathrm{Cov}[X_i, X_j] = -\frac{\alpha_i \alpha_j}{A^2(1+A)} = -\frac{\pi_i \pi_j}{A+1}. \tag{1.6}$$

In general, compositional datasets seen in practice do not follow these constraints. A more flexible model is needed.

1.3. Dirichlet-Multinomial Distribution

The Dirichlet-multinomial distribution is a hierarchical model that can be used to account for the heterogeneity present in microbiome samples and other compositional datasets.

This extra source of variation can be attributed to the fact that every person is a unique environment for microorganisms and proportions of OTUs will naturally vary from person to person. For example, studies have shown diet and disease significantly impact the type and proportions of OTUs that occur in the human gut (see references in McDonald et al. (2015)). Heterogeneity can be further explained by levels of one or more covariates. Wadsworth et al. (2017) uses Dirichlet-multinomial regression to relate microbiome counts to environmental and genetic covariates selected using a Bayesian method.

Let π denote the mean taxa proportions. The Dirichlet-multinomial (DM) model assumes

$$\boldsymbol{q_i} \overset{\text{i.i.d}}{\sim} \text{Dirichlet}(A\boldsymbol{\pi}), \quad \boldsymbol{x_i}|\boldsymbol{q_i} \sim \text{Multinomial}(N_{i.}, \boldsymbol{q_i}) \tag{1.7}$$

where $\sum_{j=1}^{k} \pi_j = 1$, $\pi_j > 0$, and A is the precision constant. $\boldsymbol{q_i}$ is a vector of hyperparameters drawn from a Dirichlet distribution.

Integrating the joint density with respect to $\boldsymbol{q_i}$ yields the marginal density

$$f(\boldsymbol{x_i}) = \binom{Ni.}{\boldsymbol{x_i}} \frac{\Gamma(A)}{\Gamma(Ni.+A)} \prod_{j=1}^{K} \frac{\Gamma(x_{ij}+A\pi_j)}{\Gamma(A\pi_j)}. \tag{1.8}$$

As $A \to \infty$, the DM distribution approaches a Multinomial$(Ni., \boldsymbol{\pi})$.

The drawbacks of using the DM distribution are that power can be reduced when there are many taxa and that signal can not be localized to any subgroup of taxa (Tang et al., 2017). To alleviate these problems, the underlying structure of the relationships between taxa (or components in general) can be described with a tree.

1.4. Dirichlet-Tree Distribution (Nested Dirichlet Distribution)

Turner (2013) and Null (2009) refer to the Dirichlet-tree distribution as the nested Dirichlet distribution (NDD), while Tang et al. (2017) and Minka (1999) call this distribution the Dirichlet-tree distribution. We will use both names interchangeably throughout this dissertation. Both Minka (1999) and Dennis (1991) derive basic properties of the nested Dirichlet distribution. The nested Dirichlet distribution is a more general form of the Dirichlet distribution. Using the nested Dirichlet distribution relaxes the constraints that variables with the same mean must have the same variance and that the covariance between variables is always negative (Null, 2009). The correlation structure between variables is determined by how the variables are nested.

1.4.1. An Illustrative Example

The structure of a nested Dirichlet distribution can be described visually with a tree diagram as in Figure 1.1. Suppose that we have a random vector $(X1, X2, X3, X4, X5)$ of proportions that sum to 1. These variables are represented as terminal nodes in Figure 1.1. There are three nesting variables in the diagram: $X6, X7$, and Root. We say that $X1, X2$ and $X3$ are nested under $X6$ while $X4$ and $X5$ are nested under $X7$. Note that $X6 = X1 + X2 + X3$ and $X7 = X4 + X5$. Each branch is labeled with an α parameter that is used to specify the Dirichlet distributions.

There are three subtrees of importance in Figure 1.1: the tree with $X6$ as the parent, the tree with $X7$ as the parent, and the tree with $X6$ and $X7$ as children. Each of these subtrees is conditionally independent of the others. From the tree, we have that $(X6, X7) \sim \text{DD}(\alpha_6, \alpha_7)$, $(X1, X2, X3)|X6 \sim \text{DD}(\alpha_1, \alpha_2, \alpha_3)$, and $(X4, X5)|X7 \sim \text{DD}(\alpha_4, \alpha_5)$. Since each sub-tree is described by a conditionally independent Dirichlet distribution, the numerical methods used to estimate Dirichlet parameters in Minka (2000) can be used

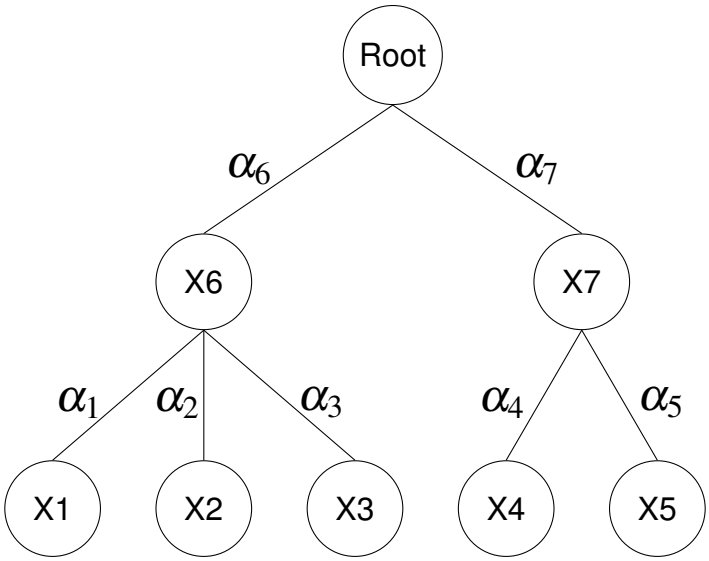


Figure 1.1: Tree diagram used to illustrate the nested Dirichlet distribution.

in each subtree of the nested Dirichlet distribution. The Dirichlet distribution can also be parameterized via means and precisions. In order to discuss expected value, the tree for the example with the α parameters swapped out for the mean parameters is shown in Figure 1.2.

The expected value $E[X1] = \pi_{11}\pi_1 = p_1$ via the assumption of two independent Dirichlet distributions, that of $(X1, X2, X3)|X6$ and $(X6, X7)$. The expected value of the remaining variables are computed in Equation 1.9.

$$\begin{aligned} E[X2] = \pi_{12}\pi_1 = p_2 \quad E[X3] = \pi_{13}\pi_1 = p_3 \\ E[X4] = \pi_{21}\pi_2 = p_4 \quad E[X5] = \pi_{22}\pi_2 = p_5 \end{aligned} \tag{1.9}$$

The proportions π_{11}, π_{12} and π_{13} are conditioned on $X6$. Similarly, π_{21} and π_{22} are conditioned on $X7$. To get the proportion at a terminal node, multiply the proportions along the branches leading to the terminal node. Equation 1.10 lists further relationships between the parameters.

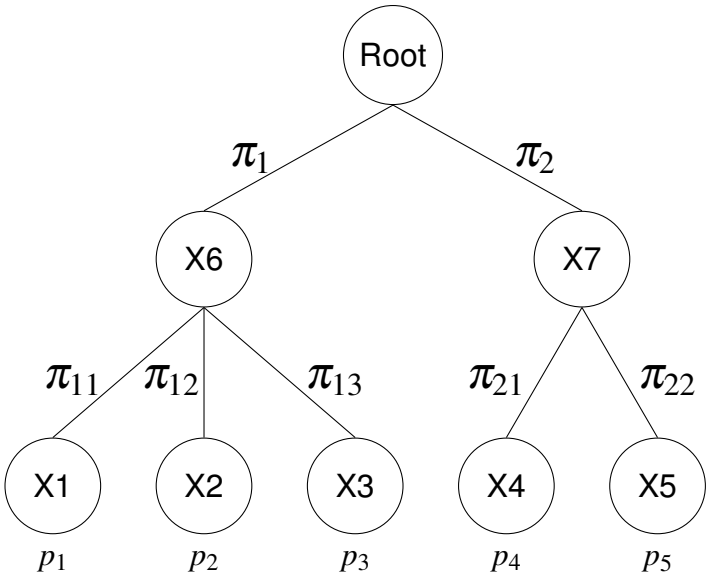


Figure 1.2: Tree diagram for the example relabeled with mean parameters.

$$\begin{array}{ll} \pi_1 = p_1 + p_2 + p_3 & \pi_2 = p_4 + p_5 \\ \pi_{11} + \pi_{12} + \pi_{13} = 1 & \pi_{21} + \pi_{22} = 1 \\ \pi_{1j} = \frac{p_j}{\pi_1} \text{ for } j = 1,2,3 & \pi_{21} = \frac{p_4}{\pi_2},\ \pi_{22} = \frac{p_5}{\pi_2} \end{array} \tag{1.10}$$

In many studies, the nesting tree arises naturally. For instance, in micorbiome studies the nesting tree used is typically the the phylogenetic tree that describes how microorganisms are related. In Turner (2013), the tree mirrors the process of how flow cytometry counts cells of each type. First, the cells are classified and counted as members of large groups that act as the nesting parent nodes. Subsequent classification (also known as *gating*) creates finer classifications that are represented as children nodes.

In other cases there is no obvious nesting between the variables. Finding an appropriate nesting tree can be difficult and computationally infeasible when the number of variables is large. In Null (2009), the dataset consists of vectors whose components are counts of fourteen different plate appearance outcomes. Even with just fourteen variables

(terminal nodes), it is impossible for a computer to check the fit of all possible nesting trees because there are too many. To get around this problem, Null (2009) constrains his search to binary trees with all fly ball outcomes nested under the same internal node and all ground ball outcomes nested under a different internal node. This is a different approach than those seen in microbiome studies since only binary trees are considered.

The purpose of the tree is to capture the correlation structure of the variables. In cases where no natural tree exists, tree finding algorithms can be used to find the best fitting tree. The tree finding algorithms may also be used in cases where natural trees do exist to find a different tree that provides a better fit. The existence of a phylogenetic tree or the gating in a flow process suggests one tree. However, this does not mean that a better fitting tree in terms of correlation structure does not exist. Tree finding algorithms, including a method to prune unnecessary internal nodes, are discussed in Chapter 3.

1.4.2. Notation for the Nested Dirichlet Distribution

The ordinary Dirichlet distribution is a conjugate prior for the multinomial distribution. The nested Dirichlet is a conjugate prior for a hierarchical multinomial distribution where the nesting tree describes the structure for both distributions. In this section, we will be describing the nested Dirichlet distribution in terms of how it relates to the associated multinomial distribution. Notation and formulas that describe the distribution are necessary. The notation used for the nested Dirichlet distribution is nonstandard. To avoid a large volume of clunky notation, we will follow the example set by Turner (2013) and describe the notation presented in Wang and Zhao (2017) in terms of our toy example. For convenience, we will use the labeling in Figure 1.3. Let $\Omega = \{1,2,3,4,5\}$ be the set of five terminal nodes. There are three internal nodes. The set of internal nodes, $\mathscr{I}$ will be represented by subsets of terminal nodes. For the tree in Figure 1.3, the internal nodes are $\mathscr{I} = \{\{1,2,3\},\{4,5\},\{1,2,3,4,5\}\}$ corresponding to the internal nodes labeled 6, 7,

and 8, respectively.

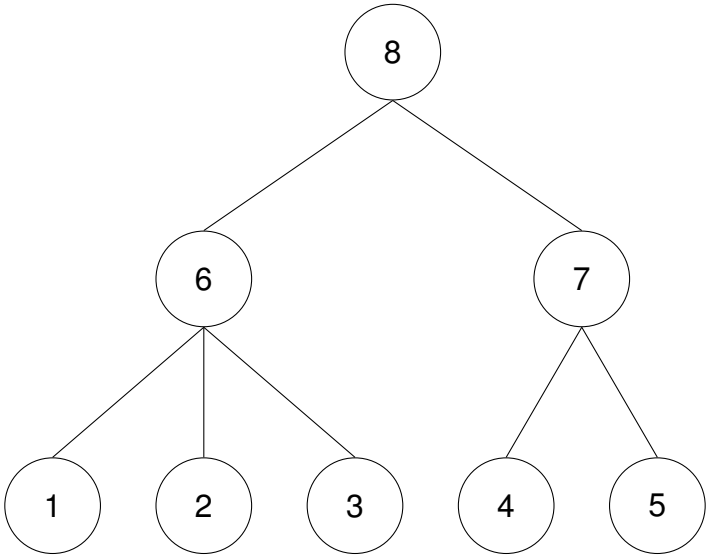


Figure 1.3: Tree diagram used to illustrate the nested Dirichlet distribution with the internal nodes labeled.

For each $A \in \mathscr{I}$, let $\mathscr{C}(A)$ be the set of child nodes of A. For example, $\mathscr{C}(\{1,2,3,4,5\}) = \{\{1,2,3\},\{4,5\}\}$ and $\mathscr{C}(\{1,2,3\}) = \{\{1\},\{2\},\{3\}\}$. Define $m(A) = |\mathscr{C}(A)|$ to be the number of children of node A. Rewrite $\mathscr{C}(A) = \{\mathscr{C}(A)_1, \ldots, \mathscr{C}(A)_{m(A)}\}$. Let x_j be the count of the items in any terminal node j. Define $x(A) = (x_1(A), \ldots, x_{m(A)}(A))$ where each component $x_j(A) = \sum_{\omega \in \mathscr{C}(A)_j} x_\omega$. For example, let $A = \{1,2,3,4,5\}$. Then $x(A) = (x_1 + x_2 + x_3,\ x_4 + x_5)$. The set $x(A)$ gives the number of observations in each child node. Let $N(A) = \sum_{j=1}^{m(A)} x_j(A)$, the total number of observations under node A. Let N be the total number of observations. The nested Dirichlet distribution is the prior of the multinomial model where each count vector $x(A)$ is conditioned on $N(A)$ for each A.

Let π_j for $j = 1, \ldots, 8$ be the probability of being classified in any of the 8 nodes. Note that $\pi_8 = 1$, $\pi_6 + \pi_7 = 1$, $\pi_6 = \pi_1 + \pi_2 + \pi_3$, and $\pi_7 = \pi_4 + \pi_5$. Let $\pi = (\pi_1, \ldots \pi_5)$ be the five free parameters. First, each item is classified as belonging to either node 6 or node 7.

The mass function at this level in the tree is the binomial

$$f(N(6),N(7)|\pi_6,\pi_7) \propto \prod_{j=6}^{7} \pi_j^{N(j)} \tag{1.11}$$

of which the Dirichlet distribution for the topmost subtree is the prior.

At the next level in the tree, the classification of items is made finer by separating those in node 6 into node 1, node 2, or node 3. Node 7 splits into node 4 and node 5. Each of these distributions is multinomial conditioned on $N(6)$ and $N(7)$ respectively. By Bayes' theorem, the multinomial distribution represented by this tree is

$$f(\boldsymbol{x}|\boldsymbol{\pi},\pi_6,\pi_7) \propto \prod_{j=6}^{7} \pi_j^{N(j)} \prod_{j=1}^{5} \pi_j^{x_j} \tag{1.12}$$

where $N(j)$ is the total number of observations nested under node j and is and π_j is the probability of being classified under node j. The conjugate prior of this product of multinomials is the nested Dirichlet distribution for the tree in Figure 1.3. The conjugate prior is a 5-variate compositional random vector denoted $\boldsymbol{\pi} = (\pi_1,\pi_2,\pi_3,\pi_4,\pi_5)$ since π_6 and pi_7 are simply functions of $\boldsymbol{\pi}$.

1.4.3. Dirichlet-Multinomial Tree Distribution

The Dirichlet-multinomial tree distribution is a nested distribution with $i = 1,\ldots,n$ samples. Let x_{ij} be the count for sample i and terminal node j. The count vector for internal node A in sample i is $\boldsymbol{x}_i(A) = \left(x_{i1}(A),\ldots x_{ik(A)}(A)\right)$ and $N_i(A) = \sum_{j=1}^{k(A)} x_{ij}(A)$. The Dirichlet-tree Multinomial (DTM) models each count vector $\boldsymbol{x}_i(A)$ conditional on the count classified under each internal node $N(A)$. The model is

$$\boldsymbol{q}_{A,i} \overset{iid}{\sim} \text{Dir}(\nu_A\pi_A), \quad \boldsymbol{x}_i(A)|N_i(A),\boldsymbol{q}_{A,i} \sim \text{Multinomial}(N_i(A),\boldsymbol{q}_{A,i}) \tag{1.13}$$

where the $q_{A,i}$ are hyperparameters and v_A are dispersion parameters. This model is more appropriate when the data is originally count data instead of proportions, for example, a dataset composed of the counts of each taxa in a study of gut microbiomes.

1.5. Literature Review: Compositional Data Applications and Methods

Compositional data occurs naturally in a variety of applications from sports analytics (Null, 2009) to genomic studies (Koslovsky and Vannucci, 2020; La Rosa et al., 2012; Tang et al., 2017; Zhang and Dao, 2020). Unfortunately, compositional data has proven difficult to analyze due to the constant sum constraint and such data are often analyzed incorrectly. A common strategy is to analyze a single component of a composition vector using a normal distribution as the underlying model (Vorhees and Williams, 2006; Barnhart et al., 2015; Tian et al., 2019). The first paper to address both the vast applications of compositional data and the difficulty in analysis was Aitchison (1982). In the paper, it is noted that the ordinary Dirichlet distribution has limited applications because it does not reflect the actual patterns of variability seen in real compositional datasets. Instead of using a nested Dirichlet model, which did not make an appearance until 1991 (Dennis, 1991), Aitchison handles compositional data by transforming the data from the simplex to the real space after which point, classical methods can be used.

Compositional data can be the response variable, explanatory variable, or both the response and the explanatory variables in a linear model. When exploring whether the mean vector is different among G groups in a compositional data setting, the compositional data is essentially the response variable in a linear model where group membership is the single categorical explanatory variable. Aitchison (1982) proposed three transformations in this case. The additive log-ratio transformation is given by

$$\text{alr}(\boldsymbol{x}) = [\ln(x_1/x_k), \ldots, \ln(x_{k-1}/x_k)] \tag{1.14}$$

which maps the vector of proportions onto $\mathbb{R}^{k-1}$. The components are treated asymmetrically with this method as x_k plays a special role. The centered log-ratio transformation is given by

$$\text{clr}(\boldsymbol{x}) = \left[\ln\left(\frac{x_1}{\prod_{i=1}^{k} x_i^{1/k}}\right), \ldots, \ln\left(\frac{x_k}{\prod_{i=1}^{k} x_i^{1/k}}\right)\right] \tag{1.15}$$

whose symmetric treatment of the components aid in interpretation. The drawback is that the components must satisfy a constant sum constraint. The last of the Aitchison's transformations is the isometric log-ratio transformation given by

$$clr(\boldsymbol{x}) \cdot H^t \tag{1.16}$$

where H is the Helmert matrix originally defined by Egozcue et al. (2003). An in-depth description of a suite of log-normal transformations and how to analyze compositional data in R can be found in the textbook by van den Boogaart and Tolosana-Delgado (2013).

When the explanatory variables are compositional or both the response and explanatory variables are compositional, then a transformation must be used before creating a linear regression model, or else any inference made from the model can not be trusted (Hron et al., 2012). When the response variable is real valued and the explanatory variable is compositional, Hron et al. (2012) suggests using the isometric log-ratio transformation on the components. Define $z_i \; i = 1, \ldots k-1$ as the transformed components of the original compositional data vector $[x_1, \ldots, x_k]^T$. Then the regression model is simply $\text{E}(Y|\boldsymbol{z}) = \gamma_0 + \gamma_1 z_1 + \cdots + \gamma_{k-1} z_{k-1}$ where there are no constraints placed on the parameters. This transformation is not able to handle the case when some of the components are zero. In this case, Alenazi (2021) uses what is known as an α-transformation on the data. When both the response and explanatory variables are compositional, Wang et al. (2013) used an isometric log-ratio transformation on both sets of variables.

Recently, there have been many studies that have moved away from the techniques introduced by Aitchison and have instead focused on novel methods using the Dirichlet

and Dirichlet-multinomial distributions. Hijazi and Jernigan (2009) use Dirichlet regression to study the composition of sand, silt, and clay at different depths in an Arctic lake. This is a classic dataset used by Aitchison. The lake depth is the only covarariate in the model. In a study by Hickey et al. (2015), the importance for forest managers to be able to predict the proportion of grades of wood products: sawlog, pallet, stake, and pulp, in a forest compartment is addressed. A current method of predicting the proportion of each is to use if-then statements in an Excel spreadsheet, in other words, a decision-tree method. Another method is to use multivariate linear regression models and ignore the fact the data is constrained between 0 and 1 (Soares et al., 1995). Using these methods, it is possible to get outputs that sum to more than one. Hickey et al. (2015) provides a much needed update by using Dirichlet regression to forecast the percentages. The compositional regression model generated predicts the proportions of each product more accurately than the Excel based decision-tree method. In this case, nesting of the variables is not explored.

The goal in Stewart et al. (2014) is to determine if two groups of predators have the same diet by examining fatty acid profiles found in their adipose tissue. A predator “absorbs" the fatty acid signatures of the prey it eats. There are about 65 different fatty acids, but only 40 are used in this study. This yields compositional data with 40 components. The standard technique for analyzing compositional data by performing a log transformation on the data so that it can be treated as multivariate normal can not be used in this case for two reasons. The first is that the data contain many zeros and the log of zero is undefined. The second is that the sample size is small compared with the number of components in a fatty acid signature. The sample size from the application dataset in this case is 10. To address these two problems, the authors use a permutation test with the null hypothesis being no difference between the two groups. They construct two different measurements of distance, one that cannot handle absolute zeros and one that can. Absolute zero means that fatty acid proportion is actually zero. There is another type of zero called “rounded zero”. A rounded zero is one where the value is rounded down to

zero or that fatty acid is below the detection limit. In the case of rounded zero, the zeros can be replaced by a small value before the analysis is done. Hypothesis test procedures are created for two independent samples and two paired samples by comparing distances between components in the full composition to distances between components in subcompositions. As in the next study we will see, no nesting is done.

Maugard et al. (2019) analyze the proportion of time mice spend in different quadrants of a Morris water maze (See Chapter 2). The current standard is to analyze this type of data using ANOVA and two-sample t-tests. In the water maze setting, Maugard et al. is the first to acknowledge that the data should be analyzed as compositional data. The authors fit a Dirichlet distribution to the data, but as we see in Chapter 2, it is possible that a nested Dirichlet distribution is a better fit.

High-throughput biological and genomic research has generated many compositional datasets that have a natural nesting structure. For example, in flow cytometry, cells in a sample are sorted into groups, such as specific types of T-cells, via wavelengths of colored dye that are attached to antibodies (McKinnon, 2018). The resulting data is compositional since each observation can be transformed into a vector of proportions of each type of T-cell, and the total proportions for each type of T-cell should sum to 1. Historically, t-tests and ANOVA have been used to test for differences in proportions of cell types between groups in the field of flow cytometry. Turner (2013) presents a new way of analyzing the data, called layered Dirichlet modeling (LDM), by modeling the data with a nested Dirichlet distribution and applying a new two-sample test to detect differences between the mean vectors between groups.

Microbiome studies such as the Human Microbiome Project (Human Microbiome Project Consortium, 2012) and the American Gut Project (McDonald et al., 2015) produce compositional datasets where the researchers seek to determine the proportions of species of microorganisms in multiple groups of individuals. La Rosa et al. (2012) is a seminal paper where the focus is to develop a statistic to test for differences in bacterial taxa composi-

tion between G groups where $G \geq 2$. The authors give the formula for a Wald-type test statistic that does just that when the underlying distribution is Dirichlet-multinomial but do not provide any method for comparing G groups when the data is modeled as a nested Dirichlet distribution or a nested Dirichlet-multinomial distribution. The package `HMP` has functions and datasets related to this paper (La Rosa et al., 2019).

Tang et al. (2017) presents a test for the equality of mean proportion vectors for $G \geq 2$ groups, when the data is assumed to come from a nested Dirichlet-multinomial model. The authors apply their test to the American Gut dataset (McDonald et al., 2015). The goal is to determine if the microbiome compositions of people with varying diets are different. Although the authors claim that their test works for $G > 2$ groups, they perform simulations with just two groups and the application they chose has two groups. Therefore, it is unclear how this test would perform on datasets composed of more than two groups.

Zhang and Dao (2020) list three issues with other tests for differences between groups of compositional data: only two groups can be tested at a time, regularity conditions on the covariance matrix must be met, and there is a sparsity assumption that states only a small portion of the components in the composition may be different across groups. Zhang and Dao (2020) outline a non-parametric test to compare the distribution of microbiome data for G groups that addresses these issues. The test statistic is the sample distance covariance between two random vectors X and Y where taxa counts are modeled with the ordinary multinomial model. If the composition is independent of the categorical variable, then the value of the distance covariance is 0. This is not a test to determine whether mean vectors are different, but a test to check if the distributions are different. Thus, this test can capture non-linear associations. The distribution of the distance covariance is unknown and the authors estimate p-values using a permutation test. Under the null hypothesis that the distribution of the components are equal for all groups, group labels can be permuted.

Another approach to test for differences between groups is to use a regression model with categorical variables indicating group membership. Wang and Zhao (2017) look at the effect of diet on gut microbiome using a regression model. There are over 100 covariates based on the answers to a food questionnaire. The regression model is novel as it is the first of its kind to use the nested Dirichlet-multinomial as the response distribution. The response vector is a vector of the counts of 28 genera-level operational taxonomic units of human gut microorganisms.

The R package `MicroBVS` (Koslovsky and Vannucci, 2020) was created as an extension to the work in Wang and Zhao (2017). `MicroBVS` takes as inputs a count compositional dataset, a set of covariates for each subject, and a nesting tree. Bayesian variable selection is used to generate a regression model with selected covariates as the explanatory variables and a Dirichlet-tree multinomial model as the response. The dataset from Wang and Zhao (2017) is used to demonstrate how the package works. This example uses a phylogenetic tree describing the relationships between genera as the nesting tree. This highlights a drawback of using the `MicroBVS` package: in order to run, it requires a nesting tree as one of the inputs. Although `MicroBVS` was developed for human microbiome studies, it can be used for any application involving compositional data given as counts. For many applications, a nesting tree will have to be found before using the package. Hence, it is important to have a data driven algorithm that can provide a nesting tree in any setting.

1.6. Finding a Nesting Tree

When a natural tree does not exist, finding the best nesting tree is a challenging problem, especially when the number of components in a composition is large. Null (2009) describes 14 non-overlapping events that can take place when a baseball player goes up to bat. Null (2009) analyzes the proportion that each event occurred for major league

baseball players using a nested Dirichlet distribution to model the data. There are far too many nesting trees to consider with 14 variables. To determine which nesting tree is the best, he uses two constraints to pare down the trees under consideration to a manageable subset. The first constraint is that all fly ball outcomes are nested under one node and all ground ball outcomes are nested under a different node. The second constraint is that he only considers generalized Dirichlet distributions. A generalized Dirichlet distribution is a special nested Dirichlet distribution where every branching is dichotomous (Connor and Mosimann, 1969). Instead of searching for the true nesting structure, a binary cascade is used. Null acknowledges that better fitting trees probably exist. Turner (2013) contributed to research in this area by developing a tree finding algorithm that gives the best fitting binary nested Dirichlet distribution. However, he also mentioned that finding the best fitting tree required a method to prune unnecessary nodes of a generalized Dirichlet distribution.

Yang et al. (2019) makes use of Dirichlet-multinomial models, trees, microbiome data, and regression in unexpected ways. What they present is a workaround to traditional parametric regression equations by constructing a tree. Suppose we have microbiome samples from n subjects with m covariates. Suppose also that there are k taxa. Instead of building a phylogenetic tree to describe how the taxa are related, the tree is built to describe how the covariates effect the microbiome composition. For each covariate, the set of all cutpoints is formed. The cutpoints are midpoints of the unique values seen in the dataset. The algorithm goes through all the cutpoints, forming two groups. The cutpoint that wins is the one that finds the most homogeneous two child groups, creating two child nodes. The process is repeated with each covariate and each child node until some stopping criterion is met. This fits the full tree to the data. Then the full tree is pruned using a cost complexity function. This balances the number of terminal nodes with goodness of fit to avoid overfitting.

1.7. Conclusion

In this dissertation, we present a likelihood ratio test for differences between more than two groups when the data comes from a nested Dirichlet distribution. This is an extension of the work in Turner (2013) which stopped at just two groups. The procedure presented in La Rosa et al. (2012) is a multiple group testing procedure, but the assumed distribution is not nested. The work in Tang et al. (2017) uses a nested Dirichlet-multinomial model instead of a nested Dirichlet model and never performs any simulations or applications for more than two groups of data. Wang and Zhao (2017) also assume a nested Dirichlet-multinomial model instead of a Dirichlet and use regression instead of a likelihood ratio test approach. Zhang and Dao (2020) look at the multiple group problem but do so from a non-parametric approach. Our work is important when the data is given strictly as proportion and not count data.

In terms of tree finding, we present a new method for pruning trees by constructing confidence intervals for the difference in α parameters between parent and child nodes. Simulation studies are done to establish the coverage probability for varying sample sizes and parameter values. The method is applied to Null's baseball data in a way that looks at batting averages and a dataset of counts of jobs in 20 industry types from 35 metro areas in the United States.

Chapter 2
A Re-Analysis of a Re-Analysis of a Morris Water Maze Experiment

In Chapter 1, five models for compositional data were presented. Two of these can be used to model compositional data where the form of the original dataset is a vector of proportions: the Dirichlet and the nested Dirichlet distributions. An example of data that can only be expressed as vector of proportion is that from the probe test of the Morris water maze experiment, explained in this next section. Each component is the proportion of time a rodent spends in a quadrant of a maze. This chapter describes the Morris water maze and re-analyzes a set of data presented in Maugard et al. (2019).

2.1. The Morris Water Maze

The Morris water maze test was initially conceived by Richard Morris with the purpose of assessing the spatial learning of rats (Morris, 1981). Prior to the inception of the Morris water maze, the spatial acuity of rats was evaluated using mazes where proximal cues were available, such as the smell of a tasty treat at the end. Examples of these types of mazes are the t-maze (Deacon and Rawlins, 2006), y-maze (Kraeuter et al., 2019), and radial maze (Connor and Mosimann, 1969), shown in Figure 2.1. Researchers wanted to know if rats could recognize and remember distal cues instead of proximal cues to escape a maze. Thus, the Morris water maze was born.

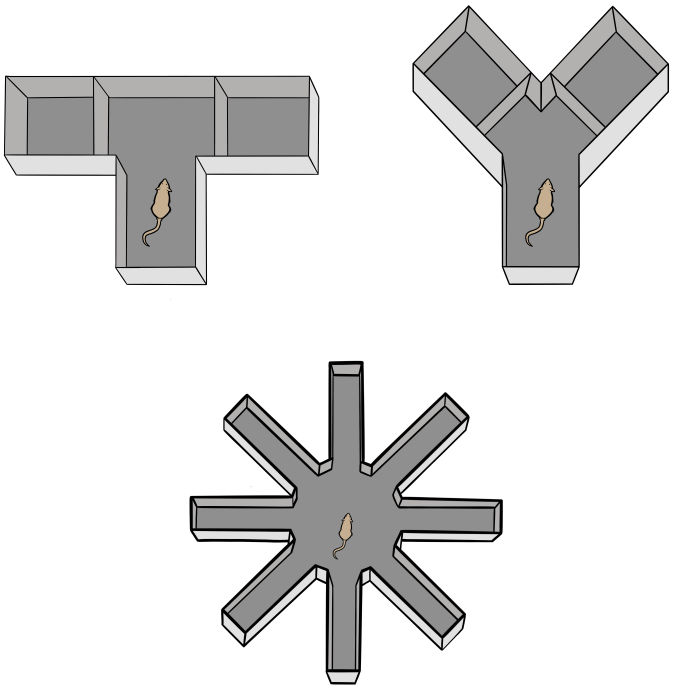


Figure 2.1: Three types of mazes typically used with mice and rats to assess spatial learning. These are the t-maze, y-maze, and radial maze. Although the rodent cannot see the goal, they are able to smell the treat at the end. This image was based on the graphics in Leising and Blaisdell (2009). Drawn by Calliope Luedeker.

Morris (1984), Vorhees and Williams (2006), and Tian et al. (2019) give detailed protocols on how to conduct the water maze test. Briefly, mice or rats are placed in a circular tank of warm water opacified by the addition of milk or tempera paint. Two imaginary perpendicular line segments divide the tank into quadrants as shown in Figure 2.2. The endpoints of the segments are labeled with the cardinal directions north, east, south, and west (these do not correspond to the actual directions and only serve to act as a convenient labeling system). In one quadrant, a hidden platform is located 1 cm below the surface of the water equidistant from the side and center of the tank and the two perpendicular lines. The platform is made from clear or white plastic so that it is not visible from the viewpoint of the rodent. The rodent can swim to the platform and use it to escape the water. Rodents are introduced into the maze over multiple trials conducted on multiple days. Experimenters perform four types of experiments using the water maze.

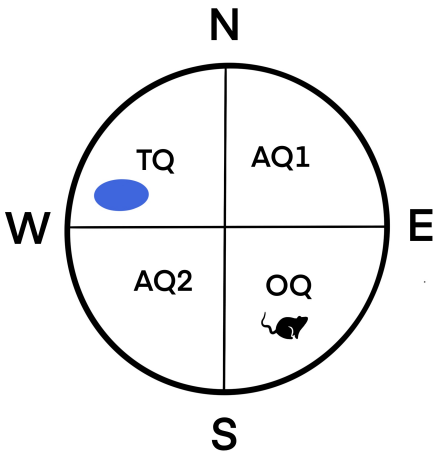


Figure 2.2: A diagram of a Morris water maze. The highlighted region is the hidden platform. The quadrants are labeled TQ (target quadrant containing the platform), AQ1 (adjacent quadrant 1), AQ2 (adjacent quadrant 2) and OQ (opposite quadrant, where the mouse is placed into the maze).

In the first type of experiment, called the cue test, either the hidden platform is marked with a flag or a visible platform sitting above the surface of the water is used. This test acts as a control. Rodents' swimming ability, visuomotor skills, and motivation are assessed. Time to reach the platform, known as escape latency, is recorded. This measurement often serves as a covariate in future analyses. At least four trials of the cue test are conducted with each rodent starting at each of the four cardinal directions (Vorhees and Williams, 2006).

The next set of trials are known as learning trials or the hidden platform test. In these trials, the platform is moved to a new quadrant and the flag is removed to make it invisible to the rodents. To match the diagram, we will say that the platform is in the NW position. During a trial, the rodent is released facing the wall of the tank in either the N, E, S, or W position. The rodent is given a fixed amount of time, 60 seconds for example, to swim to the platform. If it is unable to find the platform after 60 seconds, the researcher will lead the rodent to the platform and allow it to stand on the platform for 15 seconds before it is warmed and given another trial. Four trials are given per day, one at each starting

position, for a period of five days per rodent. Over the span of the trials, the rodents learn the location of the platform using distal extra-maze cues located in the room where the maze is housed, such as a bookcase on one side of the room. It is important for the distal cues in the regions of the room to be distinct.

After the learning trials are complete, a single probe trial, also known as a transfer test (Vorhees and Williams, 2006), is given. In the probe trial, the platform is removed from the tank. The rodent is given 60 seconds to swim around the tank freely to determine if it remembers the location of the hidden platform. The researcher can measure the number of times the rodent crosses over the location where the platform was located, the percent of time spent in each quadrant, or the length of the path in each quadrant. The probe test should be given at least 24 hours after the last learning trial. The purpose of the probe trial is to evaluate spatial memory.

A fourth test that is sometimes administered is a reversal test. In this test, the location of the hidden platform is moved to one of the three remaining positions. The rodent is then given multiple trials over multiple days to see how long it takes to forget the original platform location and learn a new one.

2.1.1. Recent Uses of the Morris Water Maze

The first paper by Morris (Morris, 1981) was used only to determine if rats could use distal cues to build a mental map of their surroundings to find a location. A follow-up paper by Morris (Morris, 1984) explores the effect of changes to the hippocampal and cortical lesions on the spatial learning of rats. Since then, the water maze has been used extensively to measure the effects of drugs, brain lesions, and disease on spatial memory and learning in rats, mice, and humans.

More recently, the water maze test has been used to investigate the effects of traumatic brain injury in rats (Andersen et al., 2020). The study compared the escape latency of two groups of rats in the hidden platform trial. One group had undergone a surgery that left them with a traumatic brain injury while the other group of rats had undergone a "sham" surgery that emulated the traumatic brain injury surgery without the injury step. Both studies found the impaired rodents tended to perform worse on the water maze test.

Alzheimer's research is another area where the Morris water maze has been employed. In Tian et al. (2019) the water maze test was used to compare the spatial memory of three groups of mice. One group was afflicted with an Alzheimer's type disease, another group was afflicted with the same disease and treated with manual acupuncture, and the remaining group was a control group composed of healthy mice. In Maugard et al. (2019) the data consisted of measurements taken from performance on a water maze test from seven wild-type mice and seven 3xTG-AD mice. The wild-type mice can be assumed to be healthy mice and the 3xTG-AD mice have three mutations related to Alzheimer's disease that affect the hippocampus and cerebral cortex. The studies found significant differences between healthy mice and the Alzheimer's type mice.

The Morris water maze test is no longer used only with rodents. Over 40 years since its inception, virtual water mazes have been created for humans to explore. These virtual water mazes are similar to the physical ones used with mice and rats. Zhong et al. (2017) uses a virtual water maze to determine the effects of age on spatial memory and learning in humans and finds three distinct groups with a significant difference in performance between young adults and a group of older poor performers. A follow up to this study (Reynolds et al., 2019) reaffirms these differences. A similar virtual water maze test is performed with groups of young adults, old good performers, and old poor performers. In this study, the subjects are placed into an fMRI machine as the test is being conducted to determine which parts of the brain were being activated. Young adults showed more activation in the anterior hippocampus while older adults showed activation of the pre-

frontal cortex. These two studies demonstrate a compensatory shift of spatial memory and learning as humans age. From the dates on these citations, one can see that the water maze experiment is still in use for various types of memory studies.

2.1.2. Data Analysis

Data from all four types of experiments has typically been analyzed using traditional methods. Of the 25 articles published using Morris water maze data since 2018, 24 of them have used classical tests such as t-tests, ANOVA, and ANCOVA (Maugard et al., 2019). When the measurements collected are escape latencies, Tian et al. (2019) states that ANOVA is the method to use. A newer approach has been to use corrected cumulative proximity to the target as the measurement of performance in all four experiment types (Zhong et al., 2017; Reynolds et al., 2019). To generate corrected cumulative proximity, the Euclidean straight line distance to the target is sampled every 200 ms. This data is summed and corrected for the differences in starting distances from the target with each starting position. ANOVA and ANCOVA are then used to detect differences between groups. Realizing that the data from the cue and hidden platform tests can be censored, Andersen et al. (2020) shows that ANOVA is not the correct approach when using escape latencies and instead uses a special type of survival model. The authors may have realized the limitations of ANOVA, but sadly, when it comes to probe test data, no new methods for analysis are provided.

In the probe test, the quadrant where the platform was previously located is labeled as the target quadrant. The quadrant opposite the target quadrant is known as the opposite quadrant. The two remaining quadrants are called adjacent quadrant 1 and adjacent quadrant 2. If a rodent has a good memory, we would expect the proportion of time spent in the target quadrant to be high and the proportion of time spent in the opposite quadrant to be low. The data produced is a vector of the four proportions of time spent in each

quadrant for each rodent.

It is clear that the data produced from the probe test is compositional. Maugard et al. (2019) notes that when it comes to analyzing the data generated in the probe test, the compositional nature of the data is largely ignored in the literature. A protocol on how to analyze the water maze test data can be found in Vorhees and Williams (2006). The analysis prescribed for the probe test is that a t-test or ANOVA be conducted on only the proportion of time in the target quadrant. This is particularly egregious since this method throws away all other components of the proportion vector and assumes that a proportion, which is constrained between 0 and 1, follows a normal distribution. Using time in only one of the quadrants does not account for the necessary dependence between time spent in the four quadrants of the maze. If a mouse spends a great deal of time in the target quadrant, then the mouse is not spending time in other quadrants, which means that the time spent in at least one other quadrant is negatively correlated with the time spent in the target quadrant.

Even when shiny new techniques are used in parts of the analysis, statistical analysis can revert back to the most comfortable methods. Vouros et al. (2018) uses machine learning to analyze raw data and in the process generates a different type of compositional data. In the experiment, the paths of the rodents are traced and broken into overlapping segments of a fixed length. Each segment is classified as exhibiting one of nine types of behaviors by an ensemble of classifiers using majority voting. The resulting dataset is compositional, describing the percent of segments classified as each behavior type. Once again, the compositional nature of the dataset is ignored. The authors compare a group of stressed and normal mice by conducting nine univariate Friedman tests, one for each strategy. The information regarding the correlation structure among strategies is not investigated.

Maugard et al. (2019) is the first paper to use data from all four quadrants to analyze probe test data. This study is also the first to analyze probe data by assuming the

data follows a Dirichlet distribution. However, this analysis is still problematic because the correlations between proportions of time spent in all four quadrants exhibit both positive and negative correlations. This indicates that a nested Dirichlet distribution may provide a better fit to the data. The motivation for writing this dissertation is to promote better ways of analyzing compositional data. In the rest of this chapter, we use the Maugard dataset, which is readily available online, to compare analyses with the Dirichlet and nested Dirichlet distribution.

2.2. Data Description and Exploratory Analysis

The experiment in Maugard et al. (2019) was designed to elucidate the effects of Alzheimer's disease on spatial memory. Seven wild-type mice (WT) and seven 3xTG-AD (3TG) mice completed one trial of the probe test after being given several trials of the hidden platform test. WT mice act as the control group and are considered cognitively healthy. The 3TG mice have three mutations related to Alzheimer's disease that affect the hippocampus and cerebral cortex; thus, it is expected that their memories are impaired. The percent of time each mouse spends in each quadrant during the probe test is recorded. The data is a set of 14 proportion vectors with four components each. Let TQ, AQ1, AQ2, OQ be the proportion of time spent in the target quadrant, adjacent quadrant 1, adjacent quadrant 2, and opposite quadrant, respectively. We would expect the 3TG mice to spend less time in the target quadrant than the WT mice.

To determine whether a Dirichlet distribution or a nested Dirichlet distribution is better to model the data, the two groups of mice were combined and the sample correlations between pairs of components were found. The sample correlations are in Table 2.1. Note that the correlation between the components AQ1 and OQ is positive. Recall that if the data follow a Dirichlet distribution, then the theoretical correlation between every pair of components is negative. Thus, this one positive sample correlation is an indication

that a nested Dirichlet distribution is a reasonable model for the data. However, a 95% confidence interval for the correlation coefficient yields $(-0.41, 0.63)$. Since this correlation is not statistically different than 0, a non-nested model may also provide an adequate fit for the data. To determine the nesting structure, the tree finding algorithm presented in Turner (2013) was used.

	TQ	AQ1	OQ	AQ2
TQ	1.00	-0.66	-0.51	-0.32
AQ1	-0.66	1.00	0.15	-0.25
OQ	-0.51	0.15	1.00	-0.27
AQ2	-0.32	-0.25	-0.27	1.00

Table 2.1: The correlation between pairs of components for the probe test in Maugard et al. (2019). Note that the correlation between AQ1 and OQ is positive. This indicates that a nested Dirichlet distribution should be used to model the data rather than a Dirichlet distribution.

The goal is to determine if the proportion of time spent in each of the four quadrants differs between the healthy mice and treatment mice. Ternary diagrams are especially created for compositional data, and can aid in visualizing differences between the two groups (van den Boogaart and Tolosana-Delgado, 2013). Figure 2.3 shows a matrix of all possible ternary diagrams for the sample data. Ternary diagrams plot the data within a triangle to show relationships among variables whose values sum to a constant. Each triangle represents a sub-composition with three components: the two components represented by the labeled corners and a third that is the sum of the proportions of the remaining two components. The closer a point is to one of the corners, the higher the proportion of that component for that observation. A point directly in the center would represent a value of $(1/3, 1/3, 1/3)^T$.

For example, consider the triangle in the first position of the second row of Figure 2.3. This triangle has the left point labeled AQ1, right point labeled TQ, and top point labeled with a star. The star represents the sum of OQ and AQ2. Let's examine the blue dot

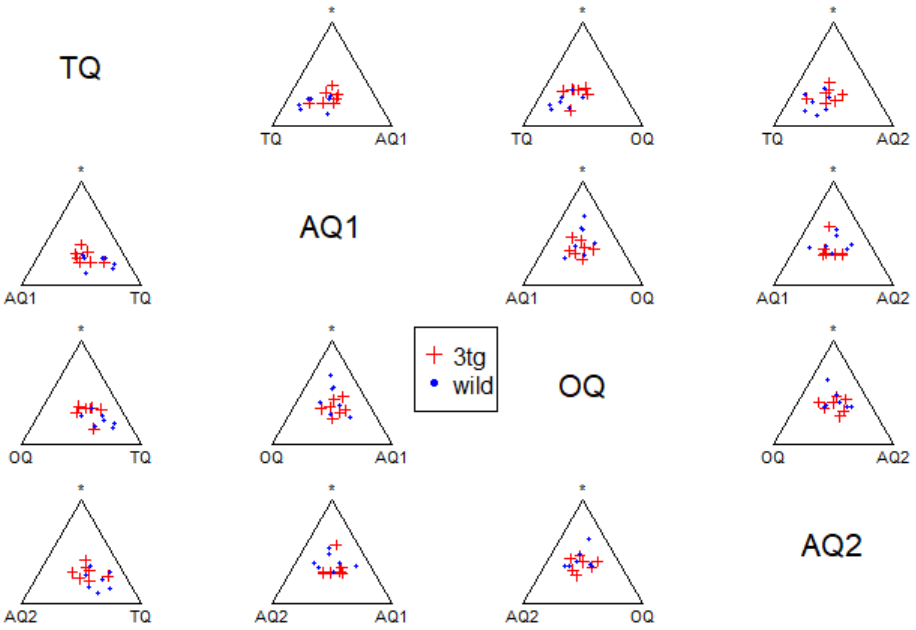


Figure 2.3: All possible ternary diagrams comparing the control and treatment mice. The star at the peak of each triangle is the sum of the two components not labeled. The blue dots represent the control group and the red crosses represent the treatment group. There does not appear to be two distinct groups in the diagrams.

near the bottom of the triangle. This point is far away from the star. Thus, the sum of the proportion of combined time spent in OQ and AQ2 is small for this rodent. Since the dot is equally spaced between AQ1 and TQ, the proportion of time spent in these quadrants is relatively equal. This dot corresponds to the observation $(\text{TQ} = 0.42,\ \text{AQ1} = 0.35,\ \text{OQ} = 0.15,\ \text{AQ2} = 0.08)$. The combined proportion of time spent in OQ and AQ2 is 0.23, which is the minimum combined time that any mouse spent in these two quadrants. Alternatively, the red cross that is almost in the center of the triangle corresponds to the observation $(0.21, 0.22, 0.24, 0.33)$. This mouse in the treatment group spent a combined 0.57 proportion of its time in OQ and AQ2, and approximately equal amounts of time in AQ1 and TQ.

Another way to think about an observation in a ternary diagram is to draw segments through the observation that are perpendicular to the sides of the triangle. Label the segments with a scale starting with 0 at the side of the triangle and ending with 1 at the point opposite that side. The proportion of the component whose label is on the point is given by the height of the observation relative to that point. This is shown in Figure 2.4.

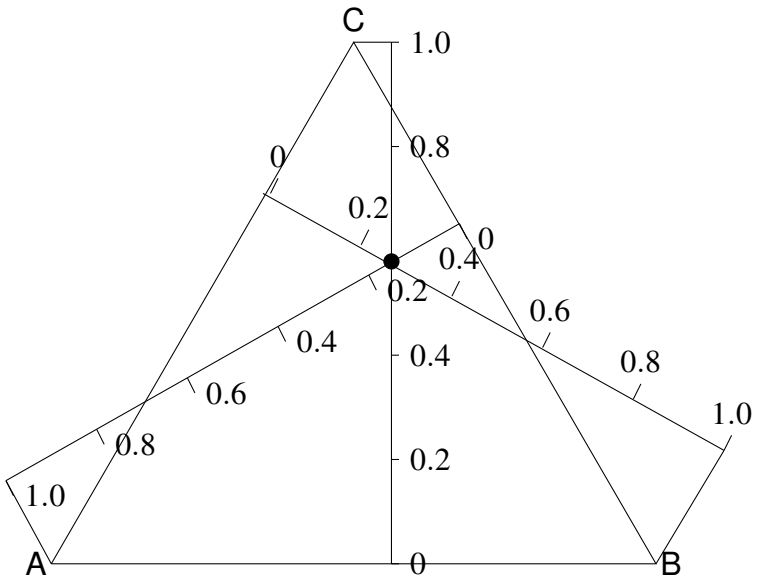


Figure 2.4: This diagram shows how proportions are calculated from ternary diagrams. This diagram is modeled after a figure in van den Boogaart and Tolosana-Delgado (2013).

Ternary diagrams were drawn in R using the plot function in the `compositions` package (van den Boogaart et al., 2021). In Figure 2.3, the blue dots represent the 7 sample compositions from WT mice and the 7 red crosses represent the sample compositions from 3TG mice. If there were a difference between the two groups, we would expect to see two distinct clusters in at least some of the ternary diagrams. In this case, there is not a clear separation between the control and treatment group in any of the diagrams. However, it does seem as though the wild group has some observations closer to the TQ corner while the 3TG group observations tend to stay clustered in the center of the diagrams. Although visually there is not a clear separation between groups, it is still possible that there could be a statistically significant difference between the groups since the diagrams don't display all facets of the relationships among the groups. We are limited to investigating sub-compositions with three components. This could mask important relationships.

Ternary diagrams can be difficult to interpret. We present another matrix of plots showing the same dataset in a traditional way. Figure 2.5 shows a matrix of scatter plots of the components of the proportion vectors taken two by two. Care must be taken when analyzing scatter plots for compositional data because they can be misleading (van den Boogaart and Tolosana-Delgado, 2013). When looking at pairwise relationships between components, there is no guarantee that the patterns seen in this subset of variables matches that in the complete dataset. We are looking at too small of a snippet of information. Hence, this is the only time we will display data in this format. In this matrix of plots, it is difficult to distinguish two distinct groups. To test our suspicion that there is not a difference between the proportion of time spent in each quadrant between the groups, we discuss the analysis presented in Maugard et al. (2019), then conduct hypothesis tests using a Dirichlet distribution model and a nested Dirichlet distribution model in the next sections.

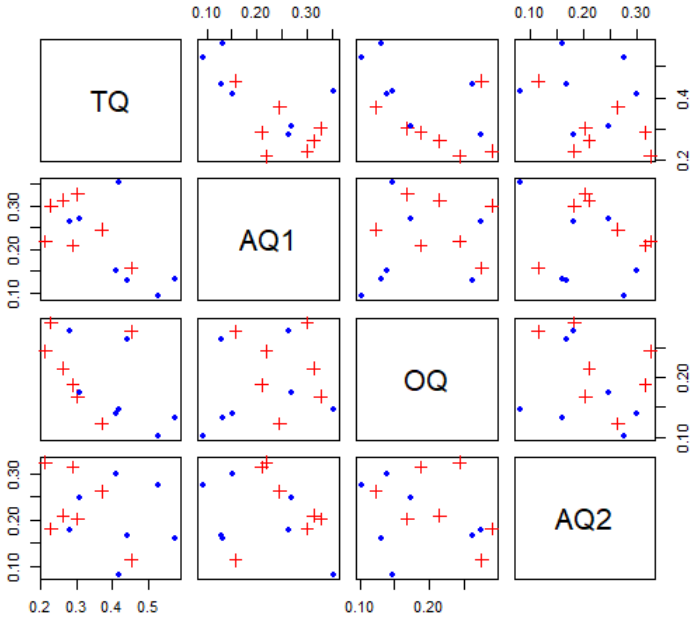


Figure 2.5: A scatter plot of the proportions of time the mice spent in each quadrant taken two at a time. The control group mice are shown with blue dots while the treatment group is shown with red crosses. There is not a clear separation between the two groups in any of the scatter plots.

2.3. Original Analysis

Maugard et al. (2019) determined that there was evidence of differences in the amount of time spent in each quadrant between the group of WT mice and the 3TG mice by modeling the data using a Dirichlet distribution. The use of the Dirichlet distribution is an improvement over previous analyses; however, the way the test was conducted is questionable. Let $\pi_1 = (\text{TQ}_1, \text{AQ1}_1, \text{OQ}_1, \text{AQ2}_1)^T$ be the population mean proportion of time spent in each quadrant for WT mice and π_2 be the same vector of parameters for the treatment population. The hypothesis test the authors wanted to conduct is:

$$\begin{aligned} \text{H}_0 &: \ \pi_1 = \pi_2 \\ \text{H}_1 &: \ \pi_1 \neq \pi_2. \end{aligned} \tag{2.1}$$

The authors were not aware of the test for equal means for two independent samples of Dirichlet distributions presented in Turner (2013). Instead, they conduct the following set of hypothesis tests:

$$\begin{aligned} &H_0: \ TQ_1 = AQ1_1 = AQ2_1 = OQ_1 = 0.25 \\ &H_1: \ \text{At least one of the above is not } 0.25 \end{aligned} \tag{2.2}$$

$$\begin{aligned} &H_0: \ TQ_2 = AQ1_2 = AQ2_2 = OQ_2 = 0.25 \\ &H_1: \ \text{At least one of the above is not } 0.25. \end{aligned} \tag{2.3}$$

The authors posit that if the mice cannot remember where the platform was, they will show no preference for any of the quadrants. They expect to reject the null hypothesis for the wild type mice and fail to reject the null hypothesis for the treatment mice. If the authors reject one of the above null hypotheses and fail to reject the other, they conclude that there is evidence of a difference in the amount of time spent in each quadrant between the control and treatment groups. Here is where a logical problem arises. If the authors fail to reject the null hypothesis for both groups it is unclear what conclusion they would draw. Similarly, it is unclear what the authors would conclude if they reject the null for both groups.

The authors fit separate $DD(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ to the control data and the treatment data using MLEs. They ultimately end up rejecting the null hypothesis for the wild-type mice and fail to reject the null for the treatment group. They conclude the mice in the treatment group have worse memories.

It is not explicitly stated, but we assume that the authors believe that Pr(type I) for the overarching test in the manner they conducted it in is 5% as long as the hypothesis tests for the individual groups were conducted at the 5% level. We conduct a simulation study to determine whether the overall Pr(type I) error is close to the nominal value of 5% in the next section.

2.3.1. Simulation Results for the Maugard et al. Procedure

Before performing any analysis on the data, we made sure that we were able to replicate the analysis that the authors had done. Following the procedure in the paper, we obtained the same p-values as the authors for the test they performed. For the test presented in Equation 2.1, where the distribution obtained for the healthy mice is compared to the uniform, the authors obtained a p-value of 0.0021. For the test comparing the distribution obtained for the 3xTg mice versus the uniform, the p-value was 0.26. Hence, the authors concluded the healthy mice had better spatial memories capabilities then the 3xTg mice.

The probability of a type I error using the method described by the authors is of major interest. To estimate the Pr(type I), we performed a simulation study. In the simulation study, we drew 14 observations from a DD$(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ distribution using five different sets of α parameters. The first seven compositions were assigned to a simulated 3xTG-AD group and the last seven were assigned to the wild-type group. The test outlined by the authors was then applied to the simulated sample data. We recorded the number times that the null hypotheses for both groups was rejected (Reject Both), the number of times both hypotheses were not rejected (Fail to Reject Both), and the number of times one null hypothesis was rejected and the other was not (Reject One). This simulation was repeated 10,000 times for each set of α parameters. Since Maugard et al. (2019) never specified what to do in the case where both null hypotheses are rejected or both are not rejected, we decided that we would only declare that there is evidence that the two groups have different means in the case where one is rejected and other is not. The Pr(type I) was calculated as the number of times only one null hypothesis was rejected divided by 10,000. The results are shown in Table 2.2.

From the table of results, it is clear that Pr(type I) is different from 5%. Instead, Pr(type I) ranges from less than 1% to 50%. Furthermore, the Pr(type I) is dependent upon the

$(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$	Reject Both	Fail to Reject Both	Reject One	Pr(type I)
(9.8, 6.1, 5.4, 5.9)	5501	663	3836	0.3836
(6.8, 6.8, 6.8, 6.8)	26	9033	941	0.0941
(9, 6, 6, 6)	2307	2675	5018	0.5018
(8, 7, 7, 5)	1812	3269	4919	0.4919
(12, 5, 5, 5)	9918	0	82	0.0082

Table 2.2: The simulation results using the method presented in the water maze paper.

parameter values of the Dirichlet distribution. The first set of parameters, (9.8, 6.1, 5.4, 5.9) are the maximum likelihood estimates from the set of 14 observations in the actual data rounded to one decimal place. These were calculated using the `dirichlet.mle` function in the `sirt` pacakge (Robitzsch, 2020). The type I error in this case represents declaring that the two groups have different mean vectors when in fact they do not. For the case given in the Maugard et al. (2019) paper, a type I error occurs almost 40% of the time. Pr(type I) is reduced in two cases. The first case is when the Dirichlet distribution is uniform, that is, all the α values are the same. In the simulation study, a uniform value of 6.8 was chosen to keep the precision $A = \sum \alpha$ comparable between sets of parameters. Since the distribution is uniform, we are more likely to fail to reject both null hypotheses. The other case where Pr(type I) is low is when the distribution deviates significantly from the uniform. This is represented in the simulation study with the parameter vector (12, 5, 5, 5). In this case, the probability of rejecting both null hypotheses is high.

In any case, this procedure should not be used to replace a two-sample test. The probability of a type I error can not be determined since it depends both on the parameter values and which of the hypotheses are rejected. In the next section, we compare the results in Maugard et al. (2019) to the results using the two independent samples test described in Turner (2013).

2.3.2. Difference in Means for Two Independent Samples Using a Dirichlet Distribution

We start our analysis of the data by assuming the data follows a DD$(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$. In the sample correlation matrix, there is a positive correlation between the variables OQ and AQ1. This is an indication that the nested Dirichlet model may provide a better fit for the data. We model the data as Dirichlet merely as a starting point.

Let A_1 be the precision parameter for the population of wild-type mice and A_2 be the precision parameter for the population of 3TG mice. Let n_1 and n_2 be the respective sample sizes. Under the null hypothesis, we assume a common mean vector π. The precision parameters for each population are allowed to vary. If not, we would be testing not just that the means were the same, but also that the two distributions were identical, which is more restrictive.

The log-likelihood function under null is:

$$\begin{aligned} L_0(A_1, A_2, \pi|\text{data}) = n_1 \log\Gamma(A_1) - n_1 \sum_{j=1}^{k} \log\Gamma(A_1\pi_j) + n_1 \sum_{j=1}^{k} (A_1\pi_j - 1)\overline{\log x_j} \\ + n_2 \log\Gamma(A_2) - n_2 \sum_{j=1}^{k} \log\Gamma(A_2\pi_j) + n_2 \sum_{j=1}^{k} (A_2\pi_j - 1)\overline{\log y_j} \end{aligned} \tag{2.4}$$

where $x_{ij},\ i = 1, \ldots, n_1,\ j = 1 \ldots k$ is the sample data for the wild mice and the quantity $\overline{\log x_j} = \frac{1}{n_1}\sum_{i=1}^{n_1} \log x_{ij}$ for $j = 1 \ldots 4$. The y_{ij} is the sample data for the treatment mice and the quantity $\overline{\log y_j}$ is defined similarly.

The log-likelihood function under the alternative hypothesis is:

$$\begin{aligned} L_1(A_1, A_2, \pi_1, \pi_2|\text{data}) = n_1 \log\Gamma(A_1) - n_1 \sum_{j=1}^{k} \log\Gamma(A_1\pi_{1j}) + n_1 \sum_{j=1}^{k} (A_1\pi_{1j} - 1)\overline{\log x_j} \\ + n_2 \log\Gamma(A_2) - n_2 \sum_{j=1}^{k} \log\Gamma(A_2\pi_{2j}) + n_2 \sum_{j=1}^{k} (A_2\pi_{2j} - 1)\overline{\log y_j}. \end{aligned} \tag{2.5}$$

Using these likelihoods, the likelihood ratio test statistic is

$$\Lambda = -2[\max L_0(A_1, A_2, \pi|\text{data}) - \max L_1(A_1, A_2, \pi_1, \pi_2|\text{data})]. \tag{2.6}$$

The number of free parameters under the null hypothesis is $k+1$ and the number of free parameters under the alternative is $2k$. Taking their difference yields $k-1$. Thus, the test statistic Λ follows an approximate χ^2 distribution with $k-1$ degrees of freedom (Casella and Berger, 2002).

There is no closed form for the maximum of the log-likelihood function. Instead, a numerical method must be used to maximize the function. We used the `optim` function in R version 4.1.2 to get maximum values (R Core Team, 2021). In order to get the most accurate values, the method L-BFGS-B should be selected (Byrd et al., 1995). L-BFGS-B is used when the parameter values are constrained. In particular, every element of the mean vectors must be bounded between 0 and 1 and the precisions must be bounded below by 0.

For this dataset, the computed test statistic was 7.223 with a corresponding p-value of 0.065. With this p-value, many researchers would not reject the null hypothesis. We come to the opposite conclusion of Maugard et al. (2019). We simply do not have enough evidence, with such a small sample size, to determine that the two groups spend different amounts of time in each quadrant.

2.4. Applying the Tree Finding Algorithm

Recall that the sample correlation matrix indicates that the data could be modeled using a nested Dirichlet distribution. The tree that describes how the four variables are nested is unknown. There is no intuitive hierarchy among the four quadrants. The nesting structure in this case is based entirely on the correlation structure among the variables,

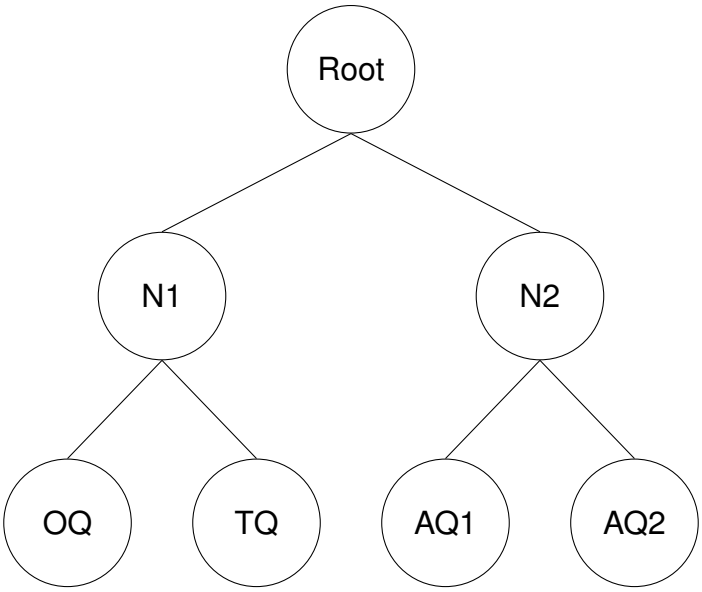


Figure 2.6: This tree does not fit the water maze data since the sample correlation between AQ1 and OQ is positive, the sample correlation between AQ2 and OQ is negative, yet AQ1 and AQ2 are nested under the same nesting variable. Either both correlations should be positive or both correlations should be negative.

not any physical real life relationship among the variables.

With four variables, there are a total of 20 nesting trees that can be built. To find the one that matches the correlation structure, Null (2008) notes that if a variable x is negatively (positively) correlated with a nesting variable y, then x will be negatively (positively) correlated with any variable nested under y. This rules out some nestings. For example, the nesting tree shown in Figure 2.6 will not work for this dataset. In Figure 2.6, N1 and N2 are the two nesting variables. If OQ is positively (negatively) correlated with N2, then it is also positively (negatively) correlated with the two variables nested underneath N2, AQ1 and AQ2. This is not the case, as OQ is positively correlated with AQ1 and negatively correlated with AQ2.

One way to determine a good nesting structure is to use a data driven algorithm. The sample data is input into the algorithm and all the possible nesting trees are fit to the data. The fit is evaluated by a user chosen criteria. Options are the maximum of

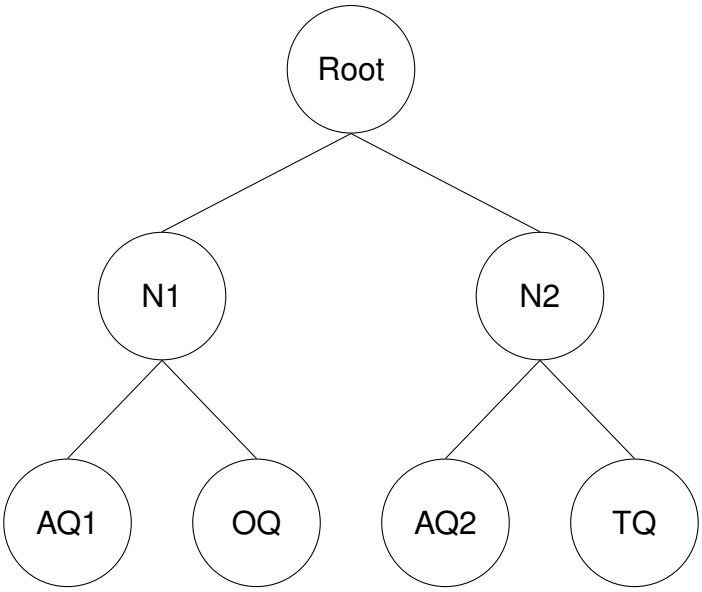


Figure 2.7: The tree with the largest maximum likelihood value for the water maze data. Note that the variables that are positively correlated are nested under the same nesting variable.

the log-likelihood, AIC, or BIC. The tree with the largest maximum criteria is selected Turner (2013). Allowing the data driven algorithm to generate the best fitting nesting tree with maximum likelihood yields the tree in Figure 2.7. The variables that are positively correlated with each other are nested under the same nesting variable. The variables that are negatively correlated with OQ are nested under a different nesting variable. This makes sense with regards to the sample correlation matrix in Table 2.1.

We fit this model to the data by finding the MLEs of the Dirichlet parameters for each of the three subtrees using the function `dirichlet.mle` from the R package `sirt` (Robitzsch, 2020). The subtree with children N1 and N2 is modeled as DD(8.1, 11.2). The subtree with children AQ1 and OQ is modeled by DD(11.6, 10.3). Lastly, the subtree with children AQ2 and TQ is modeled by DD(5.6, 9.2). The tree with the MLEs along each branch are shown in Figure 2.8. To check how well this model matches the sample data, we simulated 1000 draws using the function `rdirichlet` in the package `gtools` (Warnes et al., 2021). The sample correlation matrix from the simulated data is in Table 2.3. Compare this matrix

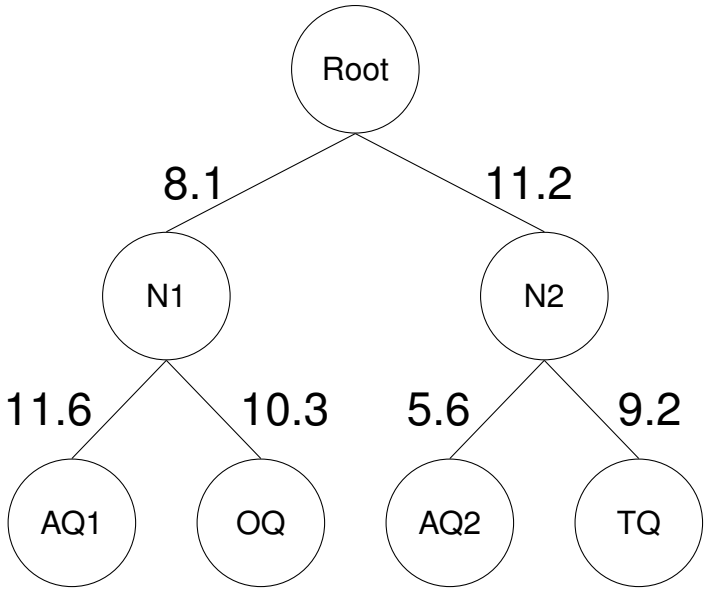


Figure 2.8: The best fitting tree with the branches labeled with the corresponding MLEs

of correlations with that in Table 2.1. All of the correlations from the simulated data are near those in Table 2.3. More importantly, the signs of the correlations from the simulated data match those of the signs of the correlations of the sample data. The nested Dirichlet model does a fine job of capturing the correlation structure.

	TQ	AQ1	OQ	AQ2
TQ	1.00	-0.55	-0.53	-0.31
AQ1	-0.55	1.00	0.20	-0.37
OQ	-0.33	0.20	1.00	-0.33
AQ2	-0.31	-0.37	-0.33	1.00

Table 2.3: The correlation between pairs of variables for the simulated water maze data. Compare this matrix with that in Table 2.1. Although the values of the correlations are not the same, they are all “in the ballpark". Specifically, the signs of the correlations for the simulated data match that of the sample data.

Another way we checked how well the model fit the data is by plotting the simulated data with the observed data in a matrix of ternary plots. If the simulated data was a poor fit, we would expect the observed data to fall outside of the cloud of points. The plot of the ternary diagrams is in Figure 2.9. The red dots represent the observed values from the

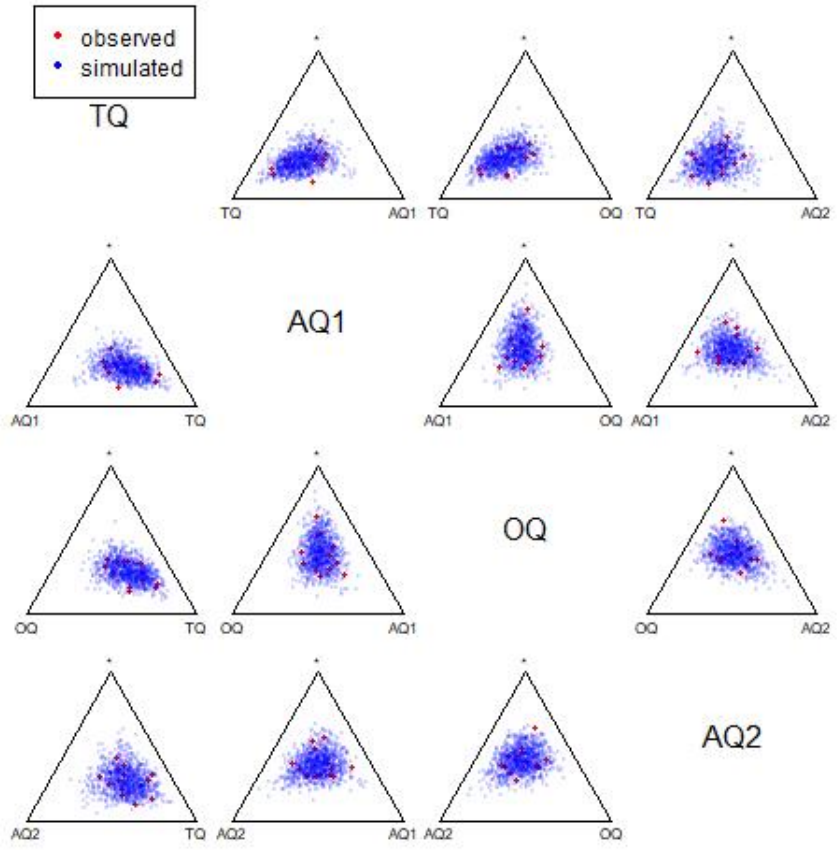


Figure 2.9: Ternary diagrams comparing the observed data to the data simulated through the nested Dirichlet distribution. The red dots represent the observed data points and the blue dots represent simulated data points. The observed data falls within the cloud of simulated data.

water maze experiment and the blue dots represent the simulated data. The observed values fall in the cloud of simulated data, indicating that this nested Dirichlet distribution is a good fit for the data. Now that we have a working model, we will conduct a hypothesis test for differences between the two groups of mice using the nested Dirichlet test.

2.5. Overall Test Using a Nested Dirichlet Model

Since we have settled on a nested Dirichlet model, the overall test presented in Turner (2013) can be applied. Let Λ_i be the likelihood ratio test statistic for the i^{th} subtree, $i = 1,2,3$. The overall test to determine if there are differences between the mean vectors for the two groups of mice is $\Lambda_{\text{overall}} = \sum_{i=1}^{3} \Lambda_i$. The test statistic, Λ_{overall} follows a χ^2 distribution with degrees of freedom $\sum_{i=1}^{3}(k_i - 1)$, where k_i is the number of variables in the i^{th} Dirichlet distribution. Note that the test statistic is just the sum of the test statistics for each of the independent Dirichlet distributions. In this case, $k_i = 2$ for all i. If the overall test is rejected, one-at-a-time tests can determine which components of the mean vectors differ.

Figure 2.10 gives the test statistic for each subtree in the example. Note that $\Lambda_{\text{overall}} = 2.79 + 0.16 + 3.18 = 6.13$. The test statistic follows a χ^2 distribution with 3 degrees of freedom. The p-value associated with this test statistic is 0.105. Compare this to the p-value using a non-nested Dirichlet distribution. The p-value in that case was 0.065. In either case, the conclusion remains the same: there is not enough evidence to conclude a difference between the two groups of mice.

2.6. Summary and Conclusion

The authors of Maugard et al. (2019) take a step in the right direction by analyzing the data as compositional data using a Dirichlet distribution rather than as a series of univariate observations using a normal distribution. The analysis needs to be taken forward two more steps. First, the two sets of data need to be compared to one another using a test like that presented in Turner (2013). Secondly, the correlation structure of the data suggests that a nested Dirichlet distribution is a more appropriate model than a Dirichlet distribution due to the positive correlation among one pair of variables.

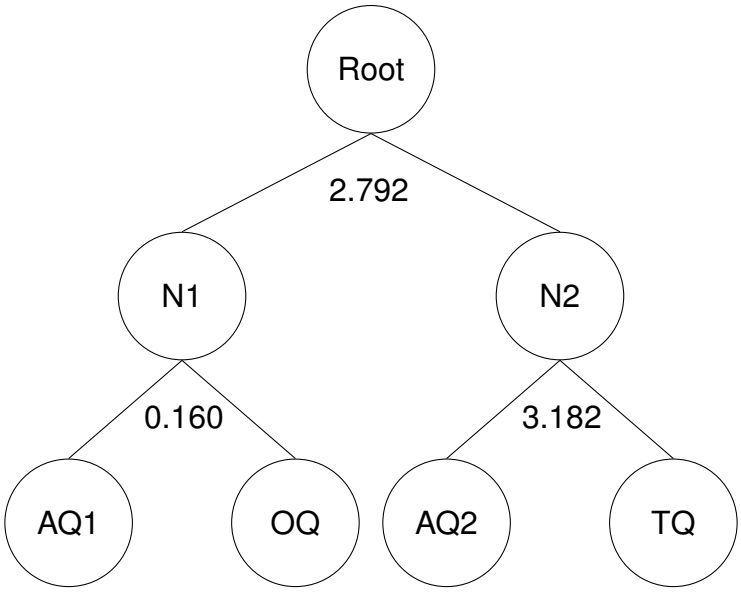


Figure 2.10: The tree structure used for the water maze data with the value of likelihood ratio test statistic for each subtree.

Using a data driven tree finding algorithm, a nesting structure was found that replicated the sample correlation structure. When analyzing the data using this nested Dirichlet distribution and a likelihood ratio test for two independent samples, no evidence of a difference between the healthy mice and the mice effected with symptoms resembling Alzheimer's disease was found in regards to the proportion of time they spent in the quadrants of the water maze. This is a reversal of the conclusion that was presented in Maugard et al. (2019). Since there was overlap between the control and treatment group seen in the ternary diagrams (Figure 2.3), we believe that the test fails to reject correctly. One major limitation of the study was the small sample size. Because the sample sizes were so small, the tests were underpowered. Perhaps with a larger sample size, a difference between the groups would be apparent.

Chapter 3
Confidence Intervals for Differences in Alpha Values at Different Levels

3.1. Introduction

Suppose we have a compositional dataset we would like to model using a nested Dirichlet distribution where the nesting structure of the data is unknown. Turner (2013) created an algorithm that outputs a binary tree of best fit for a given compositional dataset. This was used in Chapter 2 to determine the tree of best fit for the water maze data. The number of terminal nodes on each branch is either one or two. It is likely that more than two nodes should exist on any given level. An issue with this tree finding algorithm is that the tree produced is over-specified. There will be be more bifurcations in the tree than are necessary. This is problematic when conducting hypothesis tests because an overspecified tree leads to overestimation of the variance, leading to more conservative tests.

For example, suppose the sample data is generated from a population with a true nesting structure as seen at the top of Figure 3.1. The tree finding algorithm will continue to find bifurcations to get the nesting structure shown at the bottom of Figure 3.1. In this case, the parameter α_6 is not needed and estimating it inflates variance. Since the tree at the bottom collapses to the tree at the top, the parameters have a special relationship. The sum of the parameters on the lower branches must equal the parameter of the level above. In this case, $\alpha_6 = \alpha_1 + \alpha_2$.

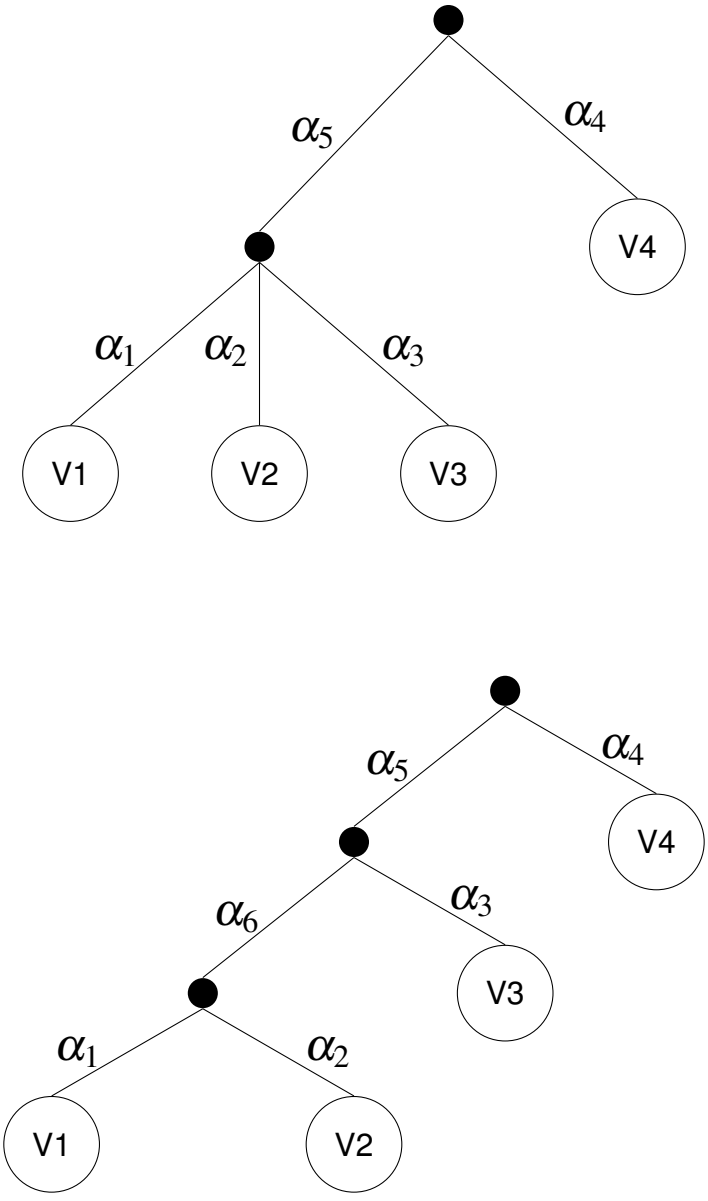


Figure 3.1: The top tree represents the actual nesting structure of the data. The bottom tree is the generalized tree produced by the tree finding algorithm. In order for the bottom tree to collapse to the top tree, $\alpha_6 = \alpha_1 + \alpha_2$.

In order for the tree algorithm to produce trees that more accurately reflect the true nesting structure, the generalized tree must be pruned to eliminate the unnecessary internal nodes. One way to do this is to create confidence intervals for differences in the parameter in the level above and the sum of the parameters in the level below. If this interval contains zero, the internal node may be removed. There are multiple ways to

generate confidence intervals for this difference parameter, but the simplest is to use MLEs as MLEs have desirable asymptotic properties.

3.2. Constructing MLE Confidence Intervals

The smallest non-trivial nested tree is given in Figure 3.2. The structure on the right side of the tree is irrelevant. The tree could have a much more complicated structure on the right and the mathematics would still work out the same. We are interested only in the internal node above V_1 and V_2.

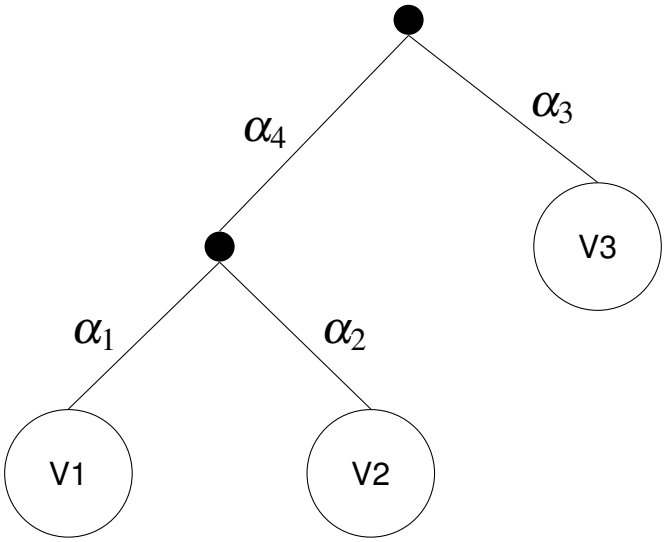


Figure 3.2: The smallest non-trivial nesting tree possible. If $\alpha_4 = \alpha_1 + \alpha_2$ the tree collapses.

To produce a confidence interval for the quantity $\alpha_4 - \alpha_1 - \alpha_2$, first the values of the individual parameters are estimated with their MLEs. Note that each layer of the tree is modeled by an independent Dirichlet distribution. The model for the top layer will be denoted DD(α_3, α_4) and the model for the bottom layer will be denoted DD(α_1, α_2). Denote the MLEs for each parameter as $\widehat{\alpha_k}$. There is no closed form expression for the MLEs. To obtain the values of the MLEs, a Newton-Raphson algorithm as presented in Minka (2000) is used. To obtain the MLEs for this dissertation, we used the function `dirichlet.mle`

from the R package `sirt` (Robitzsch, 2020). Since $\widehat{\alpha_4} - \widehat{\alpha_1} - \widehat{\alpha_2}$ is a linear combination of MLEs estimated from the same sample dataset, $\widehat{\alpha_4} - \widehat{\alpha_1} - \widehat{\alpha_2}$ is an MLE for the parameter $\alpha_4 - \alpha_1 - \alpha_2$.

MLEs are asymptotically unbiased and normally distributed with covariance matrix equal to the inverse of the Fisher information matrix. For this example,

$$\begin{bmatrix} \widehat{\alpha_1} \\ \widehat{\alpha_2} \\ \widehat{\alpha_3} \\ \widehat{\alpha_4} \end{bmatrix} \sim \text{MVN}\left(\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix}, \begin{bmatrix} \mathsf{I}(\alpha_1, \alpha_2)^{-1} & \mathbf{0} \\ \mathbf{0} & \mathsf{I}(\alpha_3, \alpha_4)^{-1} \end{bmatrix} \right) \tag{3.1}$$

where $\mathsf{I}(\alpha_i, \alpha_j)$ is the Fisher information matrix and $\mathbf{0}$ is the appropriately sized zero matrix. Since the Dirichlet distribution used to model the top level and the bottom level are independent, the pairs of parameter estimates $(\widehat{\alpha_1}, \widehat{\alpha_2})$ and $(\widehat{\alpha_3}, \widehat{\alpha_4})$ are independent from each other. Because of this independence, there are zero entries for the correlations between the estimates in the covariance matrix. To estimate the information matrices, $\mathsf{I}(\widehat{\alpha_i}, \widehat{\alpha_j})$ is used. The estimated variance of $\widehat{\alpha_4} - \widehat{\alpha_1} - \widehat{\alpha_2}$ is

$$\hat{\mathsf{V}}(\widehat{\alpha_4} - \widehat{\alpha_1} - \widehat{\alpha_2}) = \mathsf{V}(\widehat{\alpha_4}) + \mathsf{V}(\widehat{\alpha_1}) + \mathsf{V}(\widehat{\alpha_2}) + 2\mathsf{COV}(\widehat{\alpha_1}, \widehat{\alpha_2}) \tag{3.2}$$

where each element can be found in the appropriate spot in the covariance matrix. A C% confidence interval is

$$(\hat{\alpha_4} - \hat{\alpha_1} - \hat{\alpha_2}) \pm Z_{C/100} \hat{\mathsf{V}}(\hat{\alpha_4} - \hat{\alpha_1} - \hat{\alpha_2}) \tag{3.3}$$

where $Z_{C/100}$ is found from the inverse of the standard normal CDF. Note that we will never use α in this dissertation to denote a significance level. Instead, α will be reserved for the parameter values of Dirichlet distributions.

Before moving on to a simulation study of how well the interval performs with small sample sizes, we will take a peek under the hood of the CI and explicitly state what the log-likelihood and information matrix is for a Dirichlet distribution.

3.3. Theory for a Dirichlet Distribution

The following is reproduced from Minka (2000). Let $\{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n\}$ be a sample of n k-dimensional vectors that are i.i.d. DD$(\alpha_1,\ldots,\alpha_k)$. Let $\overline{\log x_j} = \frac{1}{n}\sum_{i=1}^{n}\log x_{ij}$ for $j = 1\ldots k$ and $\Gamma(x)$ be Euler's gamma function. Then the log-likelihood function parameterized in terms of $\boldsymbol{\alpha}$ is:

$$L(\boldsymbol{\alpha}|\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n) = n\log\Gamma\left(\sum_{j=1}^{k}\alpha_j\right) - n\sum_{j=1}^{k}\log\Gamma(\alpha_j) + n\sum_{j=1}^{k}(\alpha_j-1)\overline{\log x_j}. \tag{3.4}$$

This is slightly different than the mean-precision parameterization of the log-likelihood seen in Equation 2.4. The gradient of the log-likelihood is $\boldsymbol{g} = (g_1, g_2, \ldots, g_k)$ where

$$g_j = n\Psi\left(\sum_{j=1}^{k}\alpha_j\right) - n\Psi(\alpha_j) + n\overline{\log x_j} \tag{3.5}$$

for $j = 1,\ldots,k$. $\Psi(x) = \frac{d}{dx}\ln\Gamma(x)$ is the digamma function. The elements of the Hessian matrix are given by

$$\begin{aligned}\frac{d^2L(\boldsymbol{\alpha}|\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n)}{d\alpha_i^2} &= n\Psi'(A) - n\Psi'(\alpha_i)\\ \frac{d^2L(\boldsymbol{\alpha}|\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n)}{d\alpha_i\alpha_j} &= n\Psi'(A) \qquad i \neq j\end{aligned} \tag{3.6}$$

where Ψ' is the tri-gamma function. The Hessian matrix, $\boldsymbol{H}$ can be written as $\boldsymbol{H} = \boldsymbol{Q} + \mathbf{1}\mathbf{1}^T z$ where $\boldsymbol{Q}$ is the diagonal matrix with entries $q_{jj} = -n\Psi'(\alpha_j)$ for $j = 1,\ldots,k$ and

$z = n\Psi'(A)$.

Fisher's information matrix and the Hessian only differ by a sign: $\boldsymbol{I}(\boldsymbol{\alpha}) = -\boldsymbol{H}(\boldsymbol{\alpha})$. Thus, to obtain the covariance matrix, $\boldsymbol{H}^{-1}$ is needed. We have that

$$\boldsymbol{I}^{-1} = -\boldsymbol{H}^{-1} = \frac{\boldsymbol{Q}^{-1}\mathbf{1}^T\mathbf{1}\boldsymbol{Q}^{-1}}{1/z + \mathbf{1}^T\boldsymbol{Q}^{-1}\mathbf{1}} - \boldsymbol{Q}^{-1}. \tag{3.7}$$

We now have all the tools we need to build MLE confidence intervals for the difference in α parameters whose formula is given in Equation 3.3.

3.4. Simulation Studies

The purpose of the simulation study is to determine the type I error rate and power when testing the hypothesis

$$\mathrm{H}_0 : \alpha_4 - \alpha_1 - \alpha_2 = 0 \quad \text{vs} \quad \mathrm{H}_1 : \alpha_4 - \alpha_1 - \alpha_2 \neq 0 \tag{3.8}$$

when the sample size n and parameters α are varied and the tree structure is assumed to be that as in Figure 3.2.

3.4.1. Type I Error and Coverage Probability

The first set of simulations was used to determine coverage probability. Three random values between 0 and 10 were chosen for $\alpha_1, \alpha_2, \alpha_3$ with $\alpha_4 = \alpha_1 + \alpha_2$. This yielded a nested Dirichlet distribution with bottom level modeled by DD$(5,1)$ and top level modeled by DD$(2,6)$. For this set of parameter values, simulated datasets with varying sample sizes of $n = 5,\ 10,\ 20,\ 100,\ 500,\ 1000$ and 2000 were constructed. For dataset, we esti-

mated the parameters using MLEs and constructed a 95% confidence interval. For each confidence interval, it was noted whether the true parameter value of 0 was contained in the interval. The simulation was repeated 5000 times for each sample size setting. The entire process was then repeated with parameter values that were ten times as large (parameter set 2) and parameters that were 100 times as large (parameter set 3). The empirical coverage probabilities are shown in Table 3.1.

Sample Size	(5, 1, 2, 6)	(50, 10, 20, 60)	(500, 100, 200, 600)
5	1.000	1.000	1.000
10	0.995	0.996	0.996
20	0.974	0.973	0.977
30	0.968	0.968	0.965
100	0.961	0.952	0.956
500	0.950	0.951	0.951
1000	0.948	0.955	0.949
2000	0.948	0.948	0.951

Table 3.1: The coverage probabilities for varying sample sizes and parameter value sets after 5000 simulations.

As indicated by the table, the coverage probability is much larger than the stated nominal rate when the sample size is small. This is to be expected as asymptotic distributions are being employed. The coverage probability is near the nominal rate when sample size reaches 100. The difference in the magnitude of the parameter values did not have much effect on the coverage probability.

3.4.2. Power

To determine power for the method, α_4 was changed by one, two, and three units in both directions while the other parameters were held at $\alpha_1 = 5,\ \alpha_2 = 1\ \alpha_3 = 2$. The

simulation was run 5000 times with varying sample sizes of $n = 5, 10, 20, 30, 100, 500$. The proportion of time 0 was not in the interval was recorded. The results are shown in Table 3.2. The vector of parameter values is displayed at the top of the table.

Sample Size	(5, 1, 2, 7)	(5, 1, 2, 5)	(5, 1, 2, 8)	(5, 1, 2, 4)	(5, 1, 2, 9)	(5, 1, 2, 3)
5	0.00	0.00	0.00	0.00	0.00	0.00
10	0.00	0.01	0.00	0.01	0.01	0.02
20	0.03	0.03	0.05	0.08	0.07	0.21
30	0.05	0.05	0.08	0.15	0.13	0.39
100	0.10	0.14	0.27	0.49	0.49	0.93
500	0.39	0.53	0.89	1.00	0.99	1.00

Table 3.2: The probability of correctly rejecting the false null hypothesis that $\alpha_4 = \alpha_1 + \alpha_2$ for varying sample sizes and effect sizes.

The empirical power is horrendous for sample sizes of 5, 10, 20, and 30 regardless of effect size. Sample size has to be at least 100 for power to be acceptable. With an effect size of positive one, negative one, or positive two, a sample size of 500 is needed for the power to be acceptable. An interesting feature to note is that power is always greater when α_4 is smaller than $\alpha_1 + \alpha_2$ by d units than when α_4 is larger than the sum by d units. A sample size of $n = 100$ and a difference of $d = -3$ achieves a power of 0.93.

3.5. Conclusion

The confidence interval construction posed in this paper achieves a coverage probability that is near the nominal probability for sample sizes as small as 30 and varying magnitudes of the parameters. However, the power for the associated test is still small when sample size is 30 and the size of α_4 is 50% greater or less than that of the sum $\alpha_1 + \alpha_2$. To get an acceptable power, we need a sample size of at least 100 for the same effect size. Power is greater when the value of α_4 is less than the sum $\alpha_1 + \alpha_2$.

Adding a check as to whether the tree can be collapsed or not will be helpful in the case where the true tree is simpler than the one produced by the algorithm. However, we are bound to have problems when the true tree is the one with more bifurcations due to low power. A simple way to alleviate the issue of low power is to decrease the confidence level to get narrower confidence intervals. A type I error in this case is not a cataclysmic error to make. It means that a more complex tree will be used when a simpler one would have worked as well. Turner (2013) demonstrated that over specification of a tree has less of an impact on statistical inference then under specification and leads to more conservative tests. Thus, it is not a bad idea to sacrifice a small type I error rate for more power. A limitation of this procedure is that by pruning the tree, the implication is that the null hypothesis has been accepted. A method of pruning the tree that circumvents the acceptance of the null problem is to add a step to the tree finding algorithm that computes the maximum log likelihood where interior nodes are removed one at a time at each iteration. This method is not explored here, but left for future work.

Chapter 4
Addressing Bias When Using MLEs

Before proceeding any further, it is important to address the fact the MLEs of the α parameters of a Dirichlet distribution are positively biased (Gioia and Pagui, 2021; Null, 2008). The precision parameters, A, which are the sum of the α parameters, will naturally also be positively biased. The denominators of the estimated variances and covariances when constructing confidence intervals for pruning internal nodes are based on a function of A. Thus, if the bias is not corrected, the variance of the quantity $\widehat{\alpha_4} - \widehat{\alpha_1} - \widehat{\alpha_2}$ will be underestimated. This will lead to confidence intervals that are narrower than they would be if the parameter estimates were not biased. Furthermore, the test for equal mean vectors for two groups applied in Chapter 2 and the test for equal mean vectors for multiple groups presented in Chapter 5 are also based on α parameters. Thus, biased estimates could affect the outcome of these tests.

Consider the distribution Dir$(8, 6, 10, 13)$. We simulated taking a sample of size 25 from this distribution then computing the MLEs of the α parameters 10,000 times using the function `dirichlet.mle` in the `sirt` package (Robitzsch, 2020). The mean of the 10,000 MLEs is computed. Figure 4.1 shows a histogram of the MLEs and the mean of the MLEs for the four α parameters. The red line represents the actual parameter value and the blue dashed line is the mean of the parameter estimates. It is clear from Figure 4.1 that the MLEs of the α parameters are overestimates.

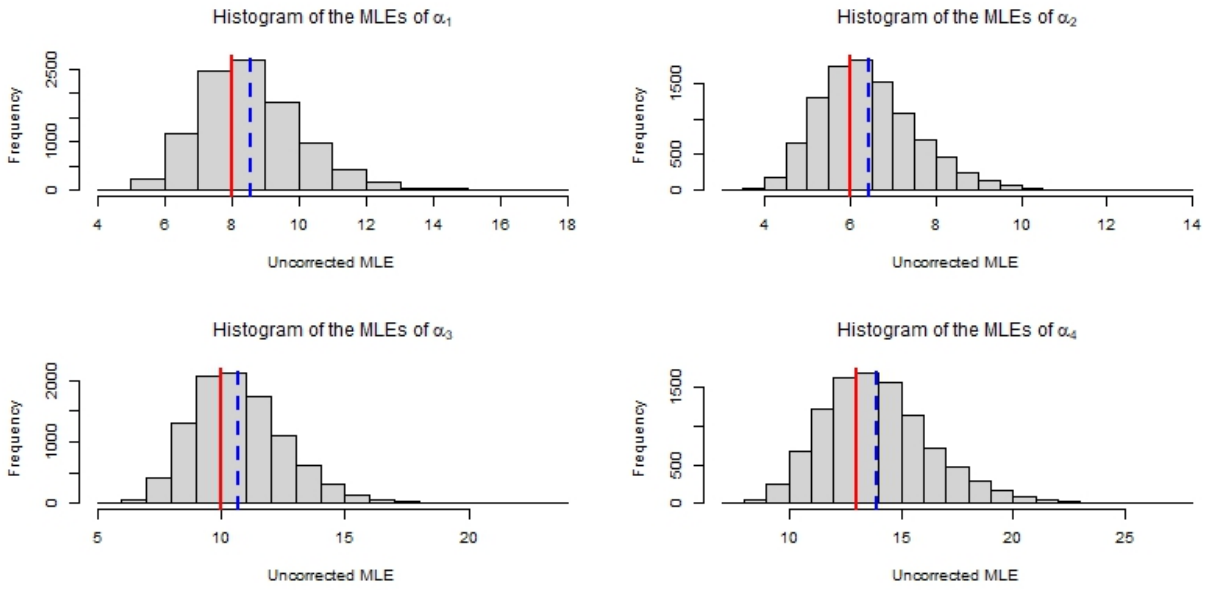


Figure 4.1: Histograms of the MLEs of the α parameters computed from 10,000 simulated datasets from Dir$(8,6,10,13)$ with sample size $n = 25$. The MLEs were not bias corrected. The red lines represent the actual parameter value and the blue lines represent the mean of the 10,000 MLEs.

4.1. Estimating Bias

A method of bias correction is to estimate the bias using a second-order bias approximation as presented in Hijazi and Jernigan (2009). The bias approximation is

$$\widehat{B(\alpha)} = \frac{1}{2} I(\alpha)^{-1} \frac{d[\text{vec}(H(\alpha)^T)]}{d\alpha} \text{vec}(I(\alpha)^{-1}) \tag{4.1}$$

where $I(\alpha) = -H(\alpha)$ is the Fisher information matrix. The function, vec$(\cdot)$ creates a vector from a matrix by columns. For example,

$$\text{vec} \begin{pmatrix} 1 & 2 & 3 \\ a & b & c \end{pmatrix} = (1, a, 2, b, 3, c).$$

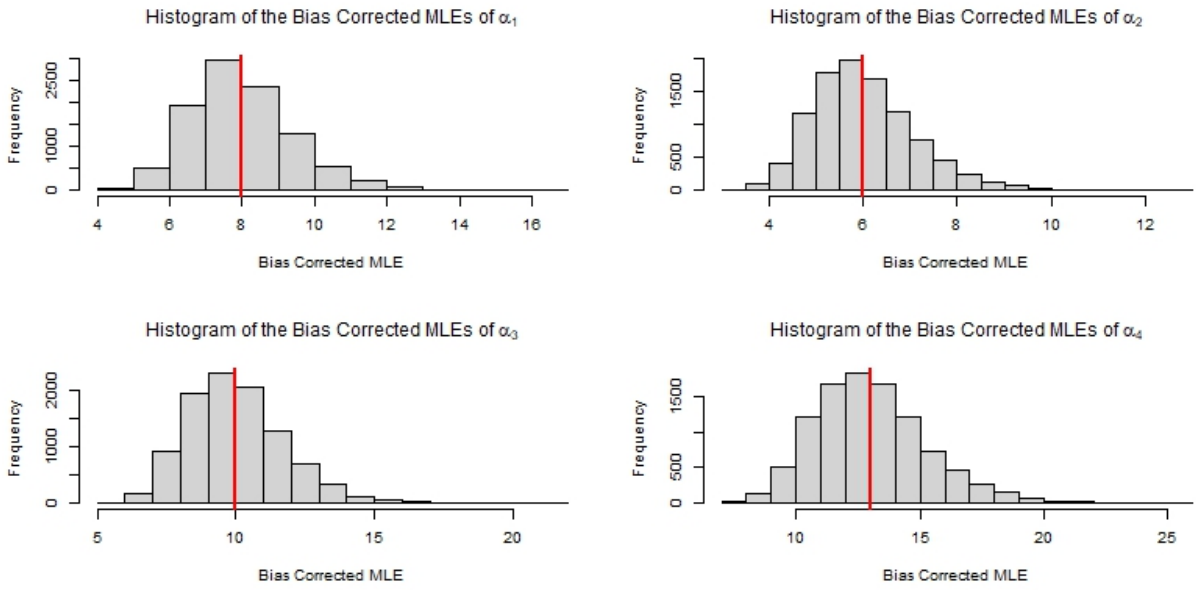


Figure 4.2: Histograms of the biased corrected MLEs of the α parameters computed from 10,000 simulated datasets from Dir$(8, 6, 10, 13)$. The red lines represent the actual parameter value and the blue lines represent the mean of the 10,000 MLEs. There is no visually discernible difference between the actual parameter values and the means of the bias corrected MLEs.

The estimated bias is then subtracted from the MLEs. When the values of the α parameters are known, the bias correction produces an unbiased estimate; however, in practice the values of α will be unknown. In this case, the MLEs of the α parameters, $\hat{\alpha}$, are used in place of α in the bias estimate. These bias corrected estimates are highly accurate. Figure 4.2 shows the actual parameter value as a red line and the mean of the bias corrected estimates as the blue dashed line for our example. It is impossible to see the difference based on visual inspection. Table 4.1 displays the mean of the MLEs of the α parameters and the means of the bias corrected MLEs for various sample sizes over 10,000 simulations. The bias of the uncorrected MLEs decreases with sample size. The bias corrected MLEs change little with sample size.

If we are estimating α parameters outside of the context of a likelihood ratio hypothesis test, then the bias corrected estimates are a good choice. However, if we are trying to

Sample Size	n=25		n=50		n=100		n=500	
Parameter	NO BC	BC	NO BC	BC	NO BC	BC	NO BC	BC
8	8.57	8.00	8.27	8.00	8.13	7.99	8.03	8.00
6	6.43	6.01	6.22	6.00	6.10	6.00	6.02	6.00
10	10.72	10.00	10.35	10.00	10.17	10.00	10.04	10.01
13	13.93	13.00	13.45	13.00	13.22	13.00	13.05	13.00

Table 4.1: The means of the MLEs and bias corrected MLEs are shown in the table for different sample sizes. The means were calculated based on 10,000 simulations. Note that the bias decreases with sample size. The mean of the the bias corrected MLES do not vary much with sample size.

estimate the α parameters and precisions under the null hypothesis of equal means when maximizing a log-likelihood for independent samples from two or more groups, problems arise. In the next section, we show that the bias corrected estimates of the α parameters are still biased in this scenario.

4.2. Bias of MLEs Under the Null Hypothesis of Equal Means

Suppose we have two independent samples of size n_1 and n_2 from Dirichlet distributions with common mean vector π and different precisions A_1 and A_2. Drawing upon the water maze data, let our example have four components $\pi = (\pi_1, \pi_2, \pi_3, \pi_4)$. Then our two Dirichlet distributions are Dir$(A_1\pi)$ and Dir$(A_2\pi)$. We seek to maximize

$$\begin{aligned} L_0(A_1, A_2, \pi|\text{data}) = n_1 \log\Gamma(A_1) - n_1 \sum_{j=1}^{K} \log\Gamma(A_1\pi_j) + n_1 \sum_{j=1}^{K} (A_1\pi_j - 1)\overline{\log x_{1j}} + \\ n_2 \log\Gamma(A_2) - n_2 \sum_{j=1}^{K} \log\Gamma(A_2\pi_j) + n_2 \sum_{j=1}^{K} (A_2\pi_j - 1)\overline{\log x_{2j}}. \end{aligned} \tag{4.2}$$

To examine the bias of the estimated parameters in this situation, we conducted a simulation where $\pi = (0.2, 0.1, 0.3, 0.4)$, $A_1 = 12$ and $A_2 = 8$. Then $\boldsymbol{\alpha_1} = 12(0.2, 0.1, 0.3, 0.4) =$

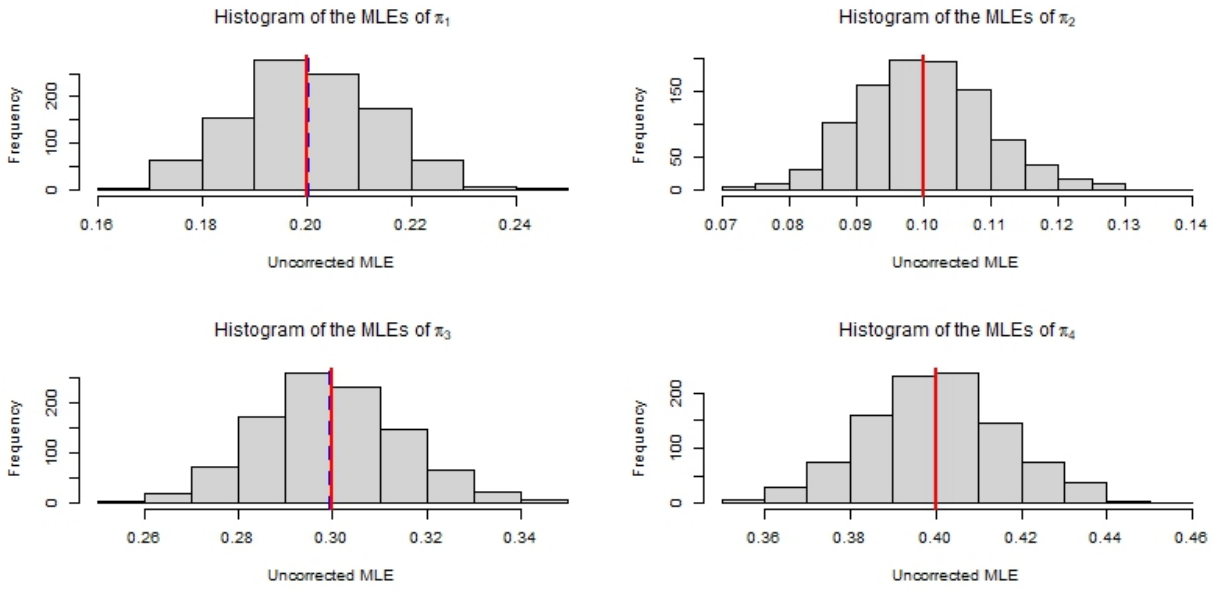


Figure 4.3: Histograms of the uncorrected estimates of the four components of the common mean vector. The red line marks the parameter value and the blue dashed line is mean over the 10,000 simulations. There is no visual discernible difference between the actual parameter and the mean of the estimates.

$(2.4, 1.2, 3.6, 4.8)$ and $\boldsymbol{\alpha_2} = 8(0.2, 0.1, 0.3, 0.4) = (1.6, 0.8, 2.4, 3.2)$. These values were chosen to be close to the values we saw in the water maze data. 10,000 simulations were run with unbalanced samples of sizes $n_1 = 30$ and $n_2 = 40$. The uncorrected MLEs of the π and α parameters were calculated for each simulation using the procedure presented in Chapter 5 to maximize the likelihood. This yielded the results presented in Figure 4.3, Figure 4.4 and Figure 4.5.

Figure 4.3 shows that the uncorrected MLEs of the components of the common mean vector are unbiased. It is known that the MLEs of the mean vector of a Dirichlet distribution are unbiased (Turner, 2013), but it was unclear if this property would hold when looking at the estimate of a common mean vector. Figure 4.4 and Figure 4.5 show that the uncorrected MLEs computed by maximizing Equation 4.2 are positively biased. The MLEs under the null hypothesis of equal means display the same properties as the MLEs when looking at a single group.

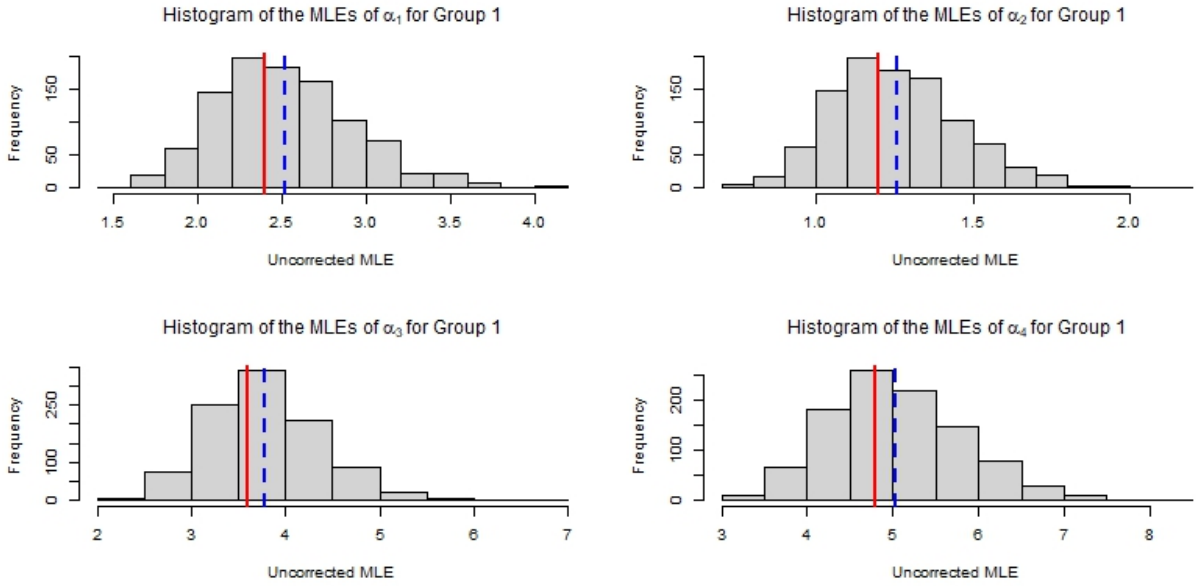


Figure 4.4: Histograms of the uncorrected MLEs of the α parameters for group 1 computed by maximizing Equation 4.2. The red lines represent the actual parameter value and the blue lines represent the mean of the 10,000 MLEs. The MLEs are positively biased.

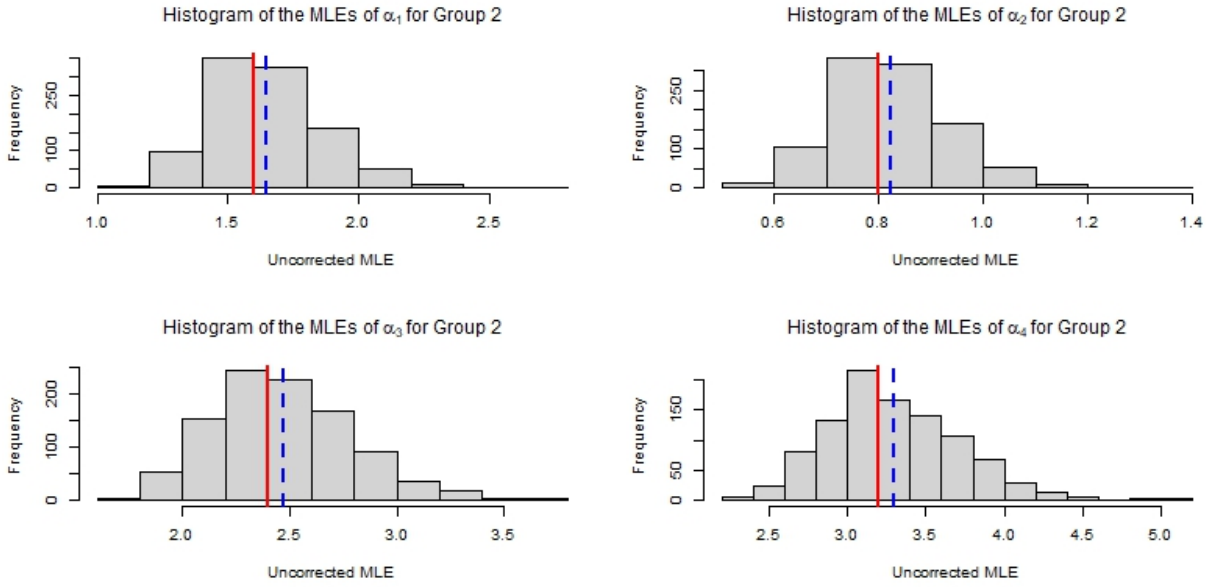


Figure 4.5: Histograms of the uncorrected MLEs of the α parameters for group 2 computed by maximizing Equation 4.2. The red lines represent the actual parameter value and the blue lines represent the mean of the 10,000 MLEs. The MLEs are positively biased.

Next we examine the bias corrected MLEs for the eight α parameters calculated by maximizing the likelihood in Equation 4.2. After subtracting the estimated bias from each α parameter, we get the results seen in Figure 4.6. The bias corrected estimates do not perform as well as they did with just one group. From Figure 4.6, it is clear that every α parameter is underestimated. The estimated bias is an overestimate when working under the hypothesis of an equal mean vector.

The current method for bias correction of the MLEs of the α parameters under the null hypothesis of equal mean vectors for two or more groups is not adequate. The estimated bias in this situation is too large, leading to over correction of the estimates. Instead of subtracting the bias estimate, a solution is to subtract a percentage of the bias estimate.

Let $\hat{\alpha}$ be the uncorrected MLE of α. Let $\hat{B}$ be the estimated bias. Let $\tilde{\alpha} = \hat{\alpha} - \hat{B}$. Consider a new estimate of α denoted as

$$\tilde{\alpha}_c = \hat{\alpha} - c\hat{B} \tag{4.3}$$

where $c \in (0,1)$. Note that if we take the mean of $\hat{\alpha}$ and $\tilde{\alpha}$ we have

$$\begin{aligned}\frac{1}{2}(\hat{\alpha}+\tilde{\alpha}) &= \frac{1}{2}(\hat{\alpha}+\hat{\alpha}-\hat{B}) \\ &= \frac{1}{2}(2\hat{\alpha}-\hat{B}) \\ &= \hat{\alpha} - \frac{1}{2}\hat{B}.\end{aligned}$$

Thus, the mean between the two estimates follows the form in Equation 4.3 with $c = 0.5$ Every estimate of the form in Equation 4.3 can be thought of as weighted mean between the uncorrected MLE and the bias corrected MLE.

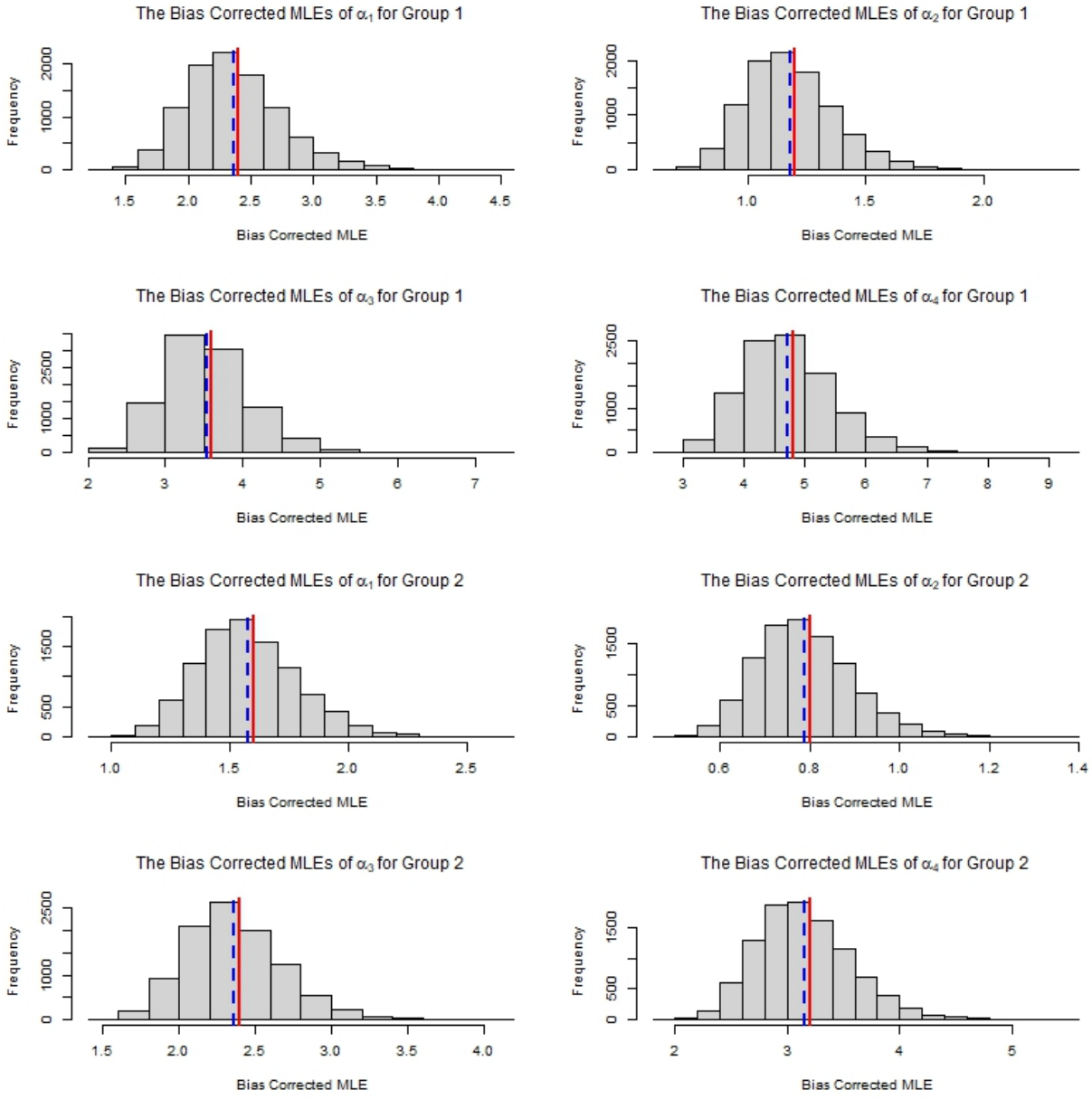


Figure 4.6: Histograms of the bias corrected MLEs of α parameters when assuming the mean vector of the two groups is the same. The red line marks the actual parameter value and the blue dashed line is the mean over all simulations. The bias corrected MLEs are underestimates.

4.3. Conclusion

A future avenue of research is to determine a value of c that minimizes the bias. This value of c is likely related to the number of groups G, the sample sizes, and the magnitude of each precision. The value of c should be chosen to minimize the difference between the estimated precision $\tilde{A}_c$ and the actual precision A since the values of individual α values can be obtained by multiplying the unbiased common mean vector estimate by the estimates of the precisions. Chapter 5 explores this bias correction method further in the multigroup setting.

Chapter 5
Hypothesis tests for $G > 2$ groups

In Chapter 2, a likelihood ratio test for equal means of two independent samples drawn from Dirichlet distributions was applied to a dataset. In this chapter, we extend this test to the multigroup setting.

5.1. Likelihood Ratio Test for G groups

Suppose we have G independent samples of compositional data with K components. Let $n_1, n_2, \ldots, n_G$ denote the sample sizes. Let $\{\boldsymbol{x}_{r1}, \ldots, \boldsymbol{x}_{rn_r}\}$ be the data from sample r, with $r = 1, 2, \ldots, G$. Assume each sample is generated by a Dirichlet distribution with mean vector $\boldsymbol{\pi_r}$ and precision A_r with $r = 1, \ldots G$. We can perform the hypothesis test:

$$H_0 : \boldsymbol{\pi_1} = \boldsymbol{\pi_2} = \cdots = \boldsymbol{\pi_G} \; vs \; H_1 : \boldsymbol{\pi_s} \neq \boldsymbol{\pi_t} \text{ for at least one pair} \tag{5.1}$$

using a likelihood ratio test.

The log-likelihood function under the null hypothesis is:

$$\begin{aligned} L_0(A_1, \ldots, A_G, \boldsymbol{\pi} | \text{data}) = {} & n_1 \log\Gamma(A_1) - n_1 \sum_{j=1}^{K} \log\Gamma(A_1\pi_j) + n_1 \sum_{j=1}^{K} (A_1\pi_j - 1)\overline{\log x_{1j}} + \\ & n_2 \log\Gamma(A_2) - n_2 \sum_{j=1}^{K} \log\Gamma(A_2\pi_j) + n_2 \sum_{j=1}^{K} (A_2\pi_j - 1)\overline{\log x_{2j}} + \\ & \ldots \\ & n_G \log\Gamma(A_G) - n_G \sum_{j=1}^{K} \log\Gamma(A_G\pi_j) + n_G \sum_{j=1}^{K} (A_G\pi_j - 1)\overline{\log x_{Gj}} \end{aligned} \tag{5.2}$$

where $\overline{\log x_{rj}} = \frac{1}{n_r}\sum_{i=1}^{n_r} \log x_{rji}$. To maximize the log-likelihood under the null in an efficient manner, we followed the steps in Turner (2013) to create a two step maximization procedure. In one step, we treat the mean as fixed and maximize over the precision and in the other step we treat the precision as fixed and maximize over the mean. This is done iteratively until a convergence criterion is reached.

If we treat the mean as fixed, the log-likelihood function is

$$L_0(A_r|\boldsymbol{\pi}, \text{data}) = n_r \log\Gamma(A_r) - n_r \sum_{j=1}^{K} \log\Gamma(A_r\pi_j) + n_r \sum_{j=1}^{K} (A_r\pi_j - 1)\overline{\log x_{rj}} \tag{5.3}$$

for each sample r where $\Gamma(\cdot)$ is the gamma function. To maximize the entire log-likelihood over all the r groups, we need only to maximize the log-likelihood for each piece and take the sum over all the pieces. To get the maximum, we use the Newton-Raphson method presented in Minka (2000). The first and second derivatives of the log-likelihood are required to use the Newton-Raphson method. These are:

$$\frac{dL(A_r|\boldsymbol{\pi}, \text{data})}{dA_r} = n_r\psi(A_r) - n_r \sum_{j=1}^{K} \pi_j \left(\psi(A_r\pi_j) + \overline{\log x_{rj}}\right) \tag{5.4}$$

$$\frac{d^2L(A_r|\boldsymbol{\pi}, \text{data})}{dA_r^2} = n_r\psi'(A_r) - n_r \sum_{j=1}^{K} \pi_j^2\psi'(A_r\pi_j) + \overline{\log x_{rj}} \tag{5.5}$$

where $\psi(\cdot)$ is the digamma function and $\psi'(\cdot)$ is the trigamma function. The second order, generalized Newton iteration is:

$$\frac{1}{A_{\text{new}}} = \frac{1}{A_{\text{old}}} + \frac{1}{A_{\text{old}}^2}\left(\frac{\psi(A_{\text{old}}) - \sum_{j=1}^{K}\pi_j(\psi(A_{\text{old}}\pi_j) + \overline{\log x_{rj}})}{\psi'(A_{\text{old}}) - \sum_{i=1}^{K}\pi_j^2\psi'(A_{\text{old}}\pi_j)}\right) \tag{5.6}$$

where A_{old} is the current precision and A_{new} is the precision in the next iteration. The iteration process is continued until $|A_{\text{new}} - A_{\text{old}}|$ is less than some convergence criteria. The convergence criteria we used was 10^{-5}. This process is repeated for each A_r individually.

The next step is to maximize the log likelihood over the common mean vector while holding each precision at A_{new}. The log-likelihood in this case is

$$L_0(\boldsymbol{\pi}|A_1,\ldots A_G,data) = \sum_{r=1}^{G}\left[n_r\sum_{j=1}^{K}(A_r\pi_j)\overline{\log x_{rj}} - n_r\sum_{j=1}^{K}\log\Gamma(A_r\pi_j)\right]. \tag{5.7}$$

Since all the groups share a single mean vector, the log-likelihood can not be maximized as the sum of G pieces. Instead, the mean vector is written using the unconstrained vector given by $\pi_j = \frac{z_j}{\sum_{j=1}^{K} z_j}$. The log-likelihood is maximized with respect to the vector z.

In order to maximize this function two competing methods in the `optim` function in R were used. Initially, the Nelder-Mead method was the method of choice. This worked well when all the parameters in the mean vector were far from the boundaries of 0 and 1 (see Table 5.2). However, this optimization method produced negative values of the mean vector when the components were ≤ 0.01. At this point, the constrained optimization method "L-FBGS-B" was used. The lower bound on the z values was set to 0.001. Even though the actual lower bound on z is 0, using 0 as a lower bound produced infinite values and caused the algorithm to stop functioning.

The L-FBGS-B can be used with or without explicitly inputting the gradient of the log-likelihood. If the gradient is not specified, `optim` computes the gradient numerically. Numerical computation proved to be inefficient and often would converge in 20 to 30 iterations. Once the gradient was specified, the algorithm converged in an average of 3 iterations. The K components of the gradient for the log-likelihood holding each precision fixed are given by:

$$\frac{dL(\boldsymbol{z}|A_1,\ldots,A_G,data)}{dz_k} = \sum_{r=1}^{G}\frac{n_rA_r}{\sum_{j=1}^{K} z_j}\left(\overline{\log x_{rk}} - \psi(A_r\pi_k) - \sum_{j=1}^{K}\pi_j\left[\,\overline{\log x_{rj}} - \psi(A_r\pi_j)\right]\right). \tag{5.8}$$

Minka (2000) suggested using method of moments estimates as the initial values for the common mean vector and each precision. The method of moments estimate for the

common mean vector was taken to be:

$$\pi_{j,int} = \frac{1}{n}\sum_{r=1}^{G}\sum_{i=1}^{n_r} x_{rji} \tag{5.9}$$

where $n = \sum_{r=1}^{G} n_r$. The initial value used for each precision was:

$$A_{r,int} = \frac{E[x_{r1}] - E[x_{r1}^2]}{E[x_{r1}^2] - E[x_{r1}]^2} \tag{5.10}$$

with the expected values replaced by the appropriate sample means. Using these initial values with the above method resulted in convergence in an average of three iterations.

5.2. Bias in Precision MLEs in the Multiple Groups Setting

In this section, we continue the discussion started in Chapter 4 by examining bias in the case of more than two groups. When testing the algorithm to find the MLEs under the null hypothesis we used an example with four samples from four different Dirichlet distributions. The four Dirichlet distributions had common mean vector $\pi = (0.2,\ 0.1,\ 0.7)$ and different precision values: $A_1 = 20,\ A_2 = 35,\ A_3 = 15,\ A_4 = 22$. For each distribution, the sample size was $n = 30$. The simulation was repeated 10,000 times. The histograms of the uncorrected MLEs of the precisions are shown in Figure 5.1. It is clear from the plots, that the MLEs are positively biased.

To get the bias corrected estimates of precision, we obtain the individual α estimates as $\hat{\alpha} = \hat{A}\hat{\pi}$, apply the bias approximation in Equation 4.1, then subtract the bias from each $\hat{\alpha}$. The bias corrected $\hat{\alpha}$ are then summed to get the bias corrected precision values. The bias correcting procedure was applied to the simulated dataset used to generate Figure 5.1. The results are shown in Figure 5.2. Just like in Chapter 4, this procedure yields underestimates of the precision parameters.

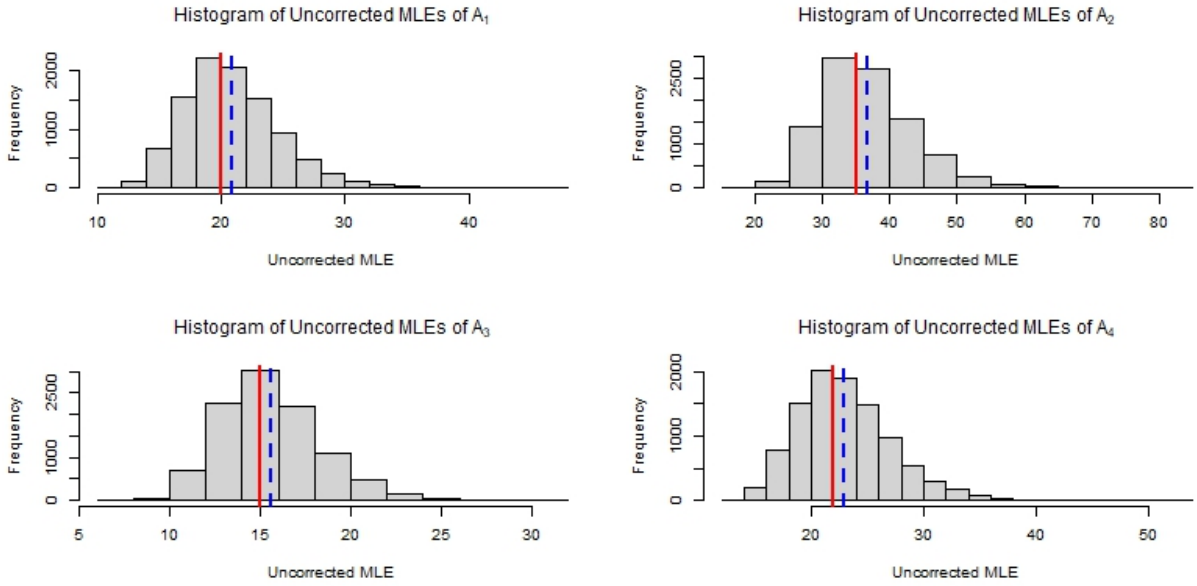


Figure 5.1: Histograms of the MLEs of the four precision parameters. The red line represents the actual parameter values of 20, 35, 15, and 22 respectively. The blue dashed lines are the mean of the MLEs over the 10,000 simulations. The MLEs of precision are positively biased.

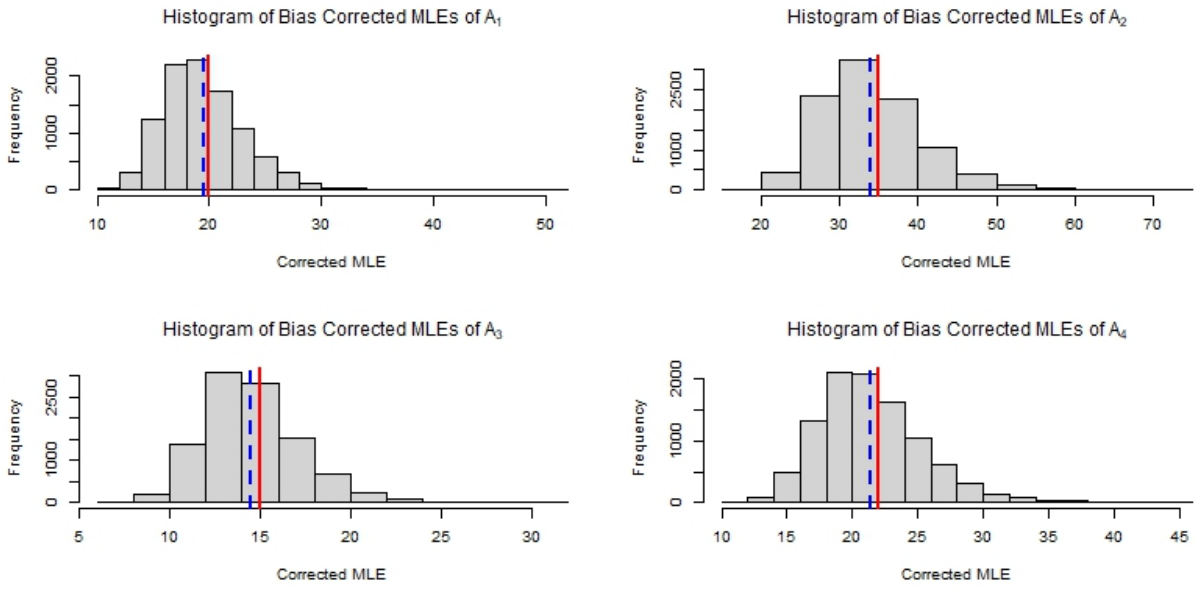


Figure 5.2: Histogram of the bias corrected estimates of precisions over 10,000 simulations. The red line is the actual precision value and the blue dashed line is the mean of the bias-corrected MLEs. Using bias corrected estimates yields underestimates of precision values.

c	A_1	A_2	A_3	A_4
0.8	0.258 (0.013)	0.473 (0.014)	0.277 (0.018)	0.294 (0.013)
0.7	0.117 (0.006)	0.228 (0.006)	0.171 (0.010)	0.139 (0.006)
0.65	0.047 (0.002)	0.105 (0.003)	0.119 (0.007)	0.061 (0.002)

Table 5.1: The absolute differences and relative differences (as a a proportion of the actual precision value) for three sets of bias corrected estimates of precision.

Let $\tilde{\alpha}$ be the bias corrected estimate of α. That is, $\tilde{\alpha} = \hat{\alpha} - \widehat{B(\hat{\alpha})}$. Since $\tilde{\alpha}$ is an underestimate of α, we look at estimators of the form $\tilde{\alpha}_c = \hat{\alpha} - c \cdot \widehat{B(\hat{\alpha})}$, where $0 \leq c \leq 1$. Table 5.1 lists the absolute differences and relative differences as a proportion of the actual precision value between the precisions and the bias corrected estimates for different values of c. Proportions are shown in parentheses. A value of $c = 0.65$ decreases the amount of relative difference between each precision and its estimate to less than 1% of the actual precision value. Figure 5.3 shows the histograms of the bias corrected estimates when $c = 0.65$ for 10,000 simulated datasets used to produce Figure 5.1. There is a good match between the parameter estimates and the actual parameter values.

5.3. Likelihood Under the Alternative Hypothesis

The log-likelihood under the alternative hypothesis in Equation 5.1 is

$$\begin{aligned} L_0(A_1,\ldots,A_G,\boldsymbol{\pi_1},\ldots,\boldsymbol{\pi_G}|\text{data}) = {} & n_1 \log\Gamma(A_1) - n_1 \sum_{j=1}^{K} \log\Gamma(A_1\pi_{1j}) + n_1 \sum_{j=1}^{K} (A_1\pi_{1j} - 1)\overline{\log x_{1j}} + \\ & n_2 \log\Gamma(A_2) - n_2 \sum_{j=1}^{K} \log\Gamma(A_2\pi_{2j}) + n_2 \sum_{j=1}^{K} (A_2\pi_{2j} - 1)\overline{\log x_{2j}} + \\ & \ldots \\ & n_G \log\Gamma(A_G) - n_G \sum_{j=1}^{K} \log\Gamma(A_G\pi_{Gj}) + n_G \sum_{j=1}^{K} (A_G\pi_{Gj} - 1)\overline{\log x_{Gj}}. \end{aligned} \tag{5.11}$$

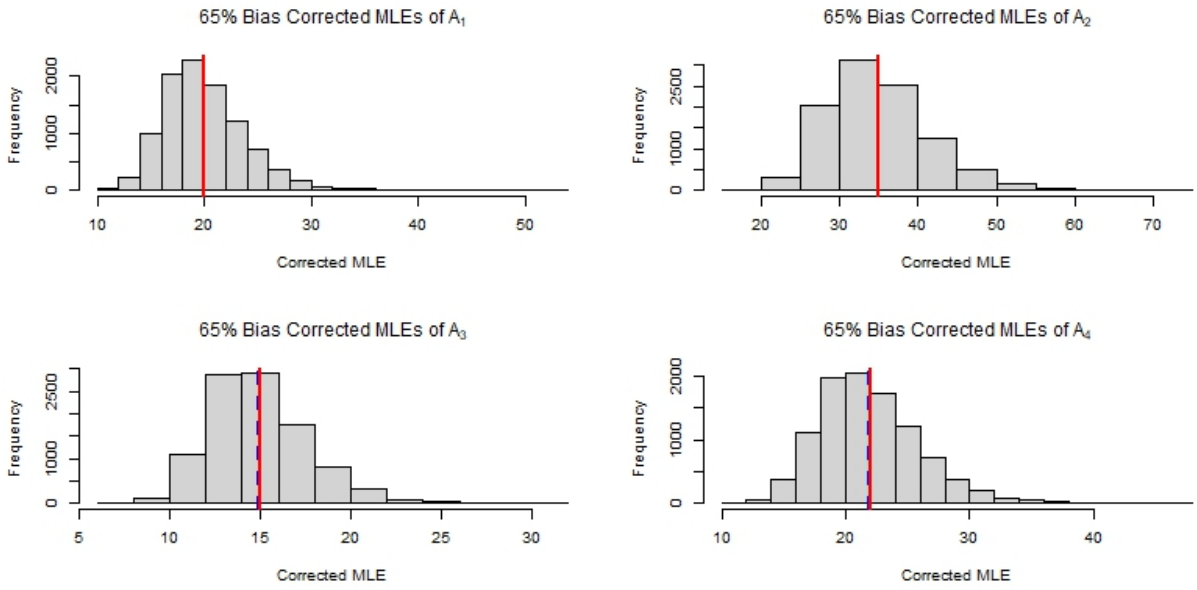


Figure 5.3: Histograms of the 65 percent bias corrected estimates of precisions over 10,000 simulated datasets. The red line is the actual precision value and the blue dashed line is the mean of the bias-corrected MLEs. The estimates and the precisions are close.

Maximizing the log-likelihood under the alternative hypothesis is actually easier than maximizing the log-likelihood under the null hypothesis since we can maximize the pieces for each group individually then take the sum of the maximums. To find where the maximums occur, we used the function `dirichlet.mle` in the package `sirt` (Robitzsch, 2020).

5.3.1. The Likelihood Ratio Test

The likelihood ratio test statistic is

$$\Lambda = -2[\max L_0(A_1,\ldots,A_G,\boldsymbol{\pi}|\text{data}) - \max L_1(A_1,\ldots,A_G,\boldsymbol{\pi_1},\ldots,\boldsymbol{\pi_G}|\text{data})]. \tag{5.12}$$

The number of free parameters in the unrestricted model is GK and the number of free parameters in the restricted model is $G+K-1$. Thus, $\Lambda \overset{approx}{\sim} \chi^2_{GK-G-K+1}$. To check that the test statistic was behaving correctly and to explore what sample sizes are necessary

to approach the nominal type I error rate, a simulation study is conducted in the next sections.

5.4. Simulation Study of a Single Layer Likelihood Ratio Test

In this chapter, the properties of the single layer likelihood ratio test for three, four, and five groups is investigated. There are two variations of this: the non-bias corrected test and the bias corrected test. We compare the performance of each. Since the original application was the water maze data, which had $K = 4$ components, the simulations are done with 4 components.

For a fixed number of groups, we have numerous simulation settings that we can alter, but we focus on three. These are:

1. The sample size for each group.
2. The value of the mean vector for each group.
3. The precision for each group.

5.5. Three Group Simulation Studies

Before doing simulation studies to determine type I error rates and power for the three group likelihood ratio test, a simulation was conducted to ensure that there was agreement between the theoretical distribution and empirical distribution of the test statistic. Because the likelihood ratio test is an asymptotic test, large sample sizes were used. The sample size vector chosen was $\boldsymbol{n} = (100, 100, 100)$. The precisions were chosen to be close to the sample values seen in the water maze data. The sample precisions from the real dataset were 27 for the wild rodents and 42 for the treatment rodents. We used 3 random

numbers between 20 and 45 for our precisions. This were 23, 32, and 42. A common mean vector of (0.25, 0.25, 0.25, 0.25) was chosen to avoid any problems that can occur when proportions are near zero or one. Within the `optim` function in R, there are multiple methods that can be selected to find the maximum of a function. 10,000 likelihood ratio test statistics were generated using the Nelder-Mead method and 10,000 likelihood ratio test statistics were generated using the constrained optimization method L-BFGS-B. The histograms of the likelihood ratio test statistics are plotted in Figure 5.4. They are overlaid with the χ^2_6 density. There appears to be a good match between the two. Ultimately, the L-BFGS-B method was used since it allows for constrained optimization.

5.5.1. Type I Error Simulations for Three Groups

The next batch of simulations was conducted to determine how close the empirical type I error rate was to the nominal type I error rate when sample sizes and mean vectors were altered. In this case, it is assumed that the mean vector for the three populations is the same while the precisions may be different. For each simulation setting, we repeated the simulation 10,000 times and counted the number of times the null hypothesis of equal means was incorrectly rejected. The precisions were held constant at $(23, 32, 42)$. The empirical probability of a type I error for the bias corrected procedure was also calculated. These are shown in Table 5.2 as BC. The entries with a NM denote that the Nelder-Mead method of optimization without bias correction was used. The L-BFGS-B optimization was used for all entries denoted LB or BC.

The empirical type I error rates are closest to the nominal error rates when the Nelder-Mead method of optimization is used. Using the constrained optimization method yields type I error rates that are too large. Unfortunately, as will be seen shortly, there are cases where the Nelder-Mead optimization will yield negative mean values and the and L-BFGS-B method must be used instead. Using bias corrected estimates of the precision with the

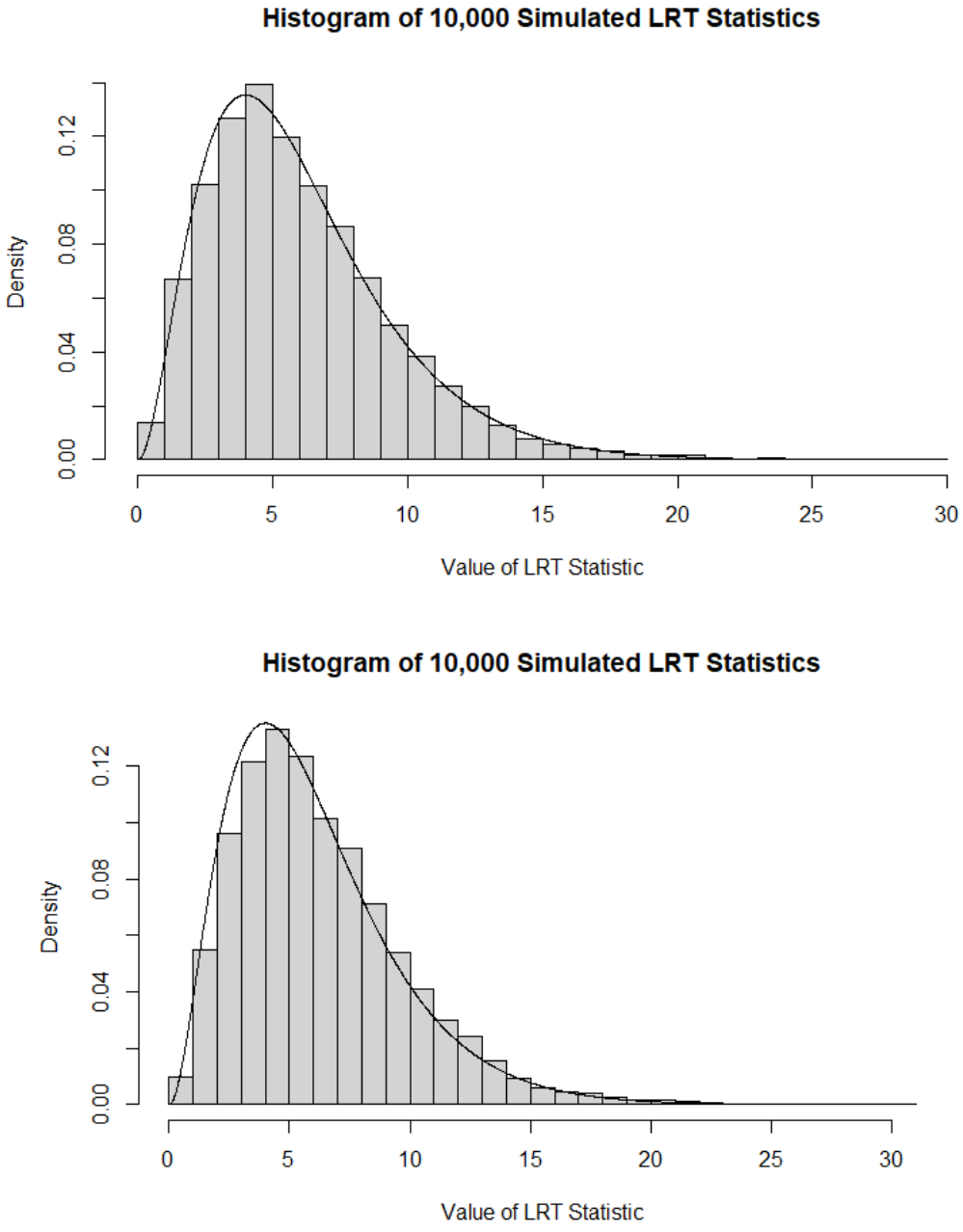


Figure 5.4: Histograms of the likelihood ratio test statistics for three groups with mean vector (0.25, 0.25, 0.25, 0.25), sample sizes of 100 per group, and precisions of 23, 32, and 42. The top histogram uses Nelder-Mead optimization method and the bottom histogram uses L-BFGS-B optimization method. The solid line represents the χ^2_6 density.

		Sample Sizes				
π	Method	$\mathbf{100}$	$\mathbf{20}$	$\mathbf{10}$	$\mathbf{5}$	$(20, 40, 60)$
(0.25, 0.25, 0.25, 0.25)	LB	0.0575	0.0735	0.0974	0.1475	0.0642
(0.25, 0.25, 0.25, 0.25)	BC	0.0575	0.0737	0.0975	0.1487	0.0642
(0.25, 0.25, 0.25, 0.25)	NM	0.0480	0.0599	0.0795	0.1131	0.0556
(0.1, 0.2, 0.3, 0.4)	LB	0.0618	0.0734	0.0942	0.1490	0.0701
(0.1, 0.2, 0.3, 0.4)	BC	0.0618	0.0738	0.0949	0.1510	0.0702
(0.1, 0.2, 0.3, 0.4)	NM	0.0513	0.0589	0.0746	0.1127	0.0600
(0.7, 0.1, 0.1, 0.1)	LB	0.0672	0.0752	0.0957	0.1497	0.0685
(0.7, 0.1, 0.1, 0.1)	BC	0.0672	0.0754	0.0964	0.1546	0.0687
(0.7, 0.1, 0.1, 0.1)	NM	0.0520	0.0605	0.0741	0.1149	0.0565
(0.97, 0.01, 0.01, 0.01)	LB	0.0072	0.1467	0.2239	0.3133	0.0876

Table 5.2: Type I errors for various mean vectors, sample sizes, estimation methods, and optimization methods for the three group likelihood ratio test for equal means. $\mathbf{100}$ is shorthand for $(100, 100, 100)$.

constrained optimization does little to correct the problem. Bias correction makes the test less conservative. The type I error rates are slightly larger when the bias correction is used than when it is not used.

With large sample sizes of 100 per group, the type I errors are still larger than the nominal value. Holding sample sizes fixed, the type I error rate increases as the mean vector moves farther away from all equal proportions. Sample sizes of size 20 give an empirical of type I error rate near 0.07 in most cases. Using sample sizes of size 10 is not advisable since the type I error rate jumps to over 0.09. And using sample sizes of size 5 is just foolish as the empirical type I error rate is near 0.15. One case where the sample sizes were unequal was investigated. In this case, all sample sizes were over 20. The empirical type I error rate was between the type I error rate for equal sample sizes of 100 and equal sample sizes of 20. The main driving force of type I error rates is the size of the samples, not whether they are equal in size.

The case where the components of the shared mean vector are near the boundaries of the parameter space, i.e. near 0 and near 1, creates difficulties. The example chosen to illustrate this was the mean vector (0.97, 0.01, 0.01, 0.01). In this case, the Nelder-Mead algorithm returned negative parameter estimates. Using the constrained optimization provided sensible parameter estimates. When we attempted to use the bias correction procedure, the function `dirichlet.mle` failed to produce results as part of the algorithm requires inverting a matrix and the entries were too near zero. The type I error probabilities generated were either nonsensical or useless. With sample sizes of 100, the type I error probability without bias correction was 0.0072. For the other sample sizes, the type I error probabilities explode except in the case where the sample sizes were 20, 40, and 60. Using a likelihood ratio test when the parameter values are near the boundaries of (0, 1) should only be used when sample sizes are large to avoid large type I error rates.

5.5.2. Power Analysis for Three Groups

Power simulations involving three groups and mean vectors with multiple components gets messy quickly. Just determining how the effect size should be measured is a problem in itself. To keep the complexity down, the following restrictions were used:

1. The mean vector consisted of only four components.
2. The three precisions for the groups were all different but held fixed at $(23, 32, 42)$.
3. The two groups with corresponding precisions of 23 and 32 had a mean vector fixed at $(0.25, 0.25, 0.25, 0.25)$. The mean vector of the group with a precision of 42 was altered. In other words, we want to know when the test detects that one of the three groups has a different mean vector.
4. Only the first two components of the mean vector were altered for the third group.

The effect size measures how far the first two components of the mean vector differ from $(0.25, 0.25, 0.25, 0.25)$. Let δ denote the effect size. We define the effect size to be the amount subtracted from π_2 and added to π_1. For instance, a mean vector of $(0.30, 0.2, 0.25, 0.25)$ would correspond to an effect size of $\delta = 0.05$ and $(0.20, 0.30, 0.25, 0.25)$ would correspond to $\delta = -0.05$.

A simulation study was conducted using 6 different sample size settings:

1. Large: (100, 100, 100)
2. Unequal: (20, 40, 60)
3. Comparison to unequal: (40, 40 , 40)
4. Medium: (20, 20, 20)
5. Small: (10, 10, 10)
6. Very Small: (5, 5, 5).

Each sample size setting was paired with an effect size $\delta \in \{-0.100, -0.095, \ldots, 0.095, 0.100\}$. This gives a total of $6 \times 41 = 246$ simulation settings. For each simulation setting, 10,000 datasets were simulated and the likelihood ratio test was applied to each. The probability of rejecting the null hypothesis of equal means among the groups using a 5% significance level was recorded.

Figure 5.5 shows the results of the simulation. These are the power curves for the different sample size settings. None of the curves reach the nominal type I error rate of 0.05 at $\delta = 0$. As the total sample size, $n = n_1 + n_2 + n_3$, decreases, the type I error rate increases. As the total sample size decreases, the power of the test also markedly decreases. The test works well for sample sizes of 100 each and detects differences with an effect size as small as 0.025 with high power. With smaller unequal sample sizes, the test performs nearly as well. Even at sample sizes of 20, the power of the test is adequate

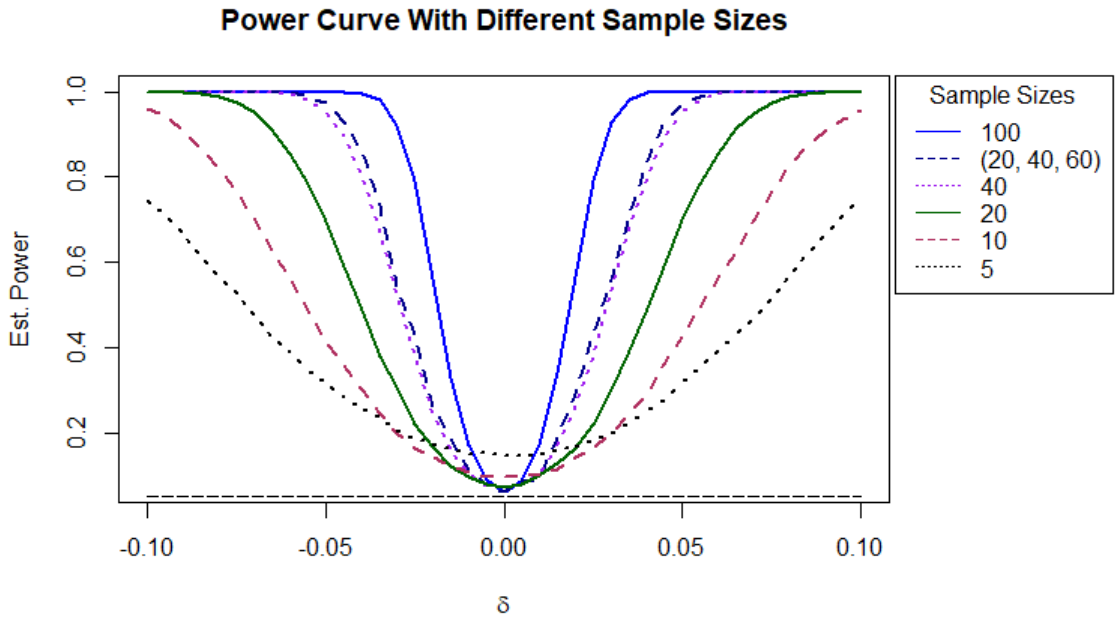


Figure 5.5: Power curves for the 3 group likelihood ratio test. Each curve represents a different sample size setting. The dashed line at the bottom represents the desired type I error rate of 5%. Note that when $\delta = 0$, the curves never reach the line.

for effect sizes of 0.05 or greater. When sample sizes dip below 10, power takes a major hit and type I errors are not where they should be. The power curves appear largely symmetric with respect to δ. We would expect to see similar curves for the four and five group cases.

We wanted to examine the effect of unequal sample sizes on power. To do this, we compared the power curve with unequal sample sizes $(20, 40, 60)$ to the equal sample size case of $(40, 40, 40)$. Both settings have a total sample size of 120. The power curves are distinguishable, but the vertical distances between them is small throughout the range of δ. The case with unequal sample sizes performs somewhat better than the case with equal sample sizes, which is surprising. This is probably due to the fact the group with the largest sample size of 60 is the group that has the different mean vector, making this difference more detectable than when only 40 units are assigned to group 3.

5.6. Four and Five Group Simulation Studies

Simulation studies for the likelihood ratio test were carried out for four groups for the common mean case to determine type I error rates. The precisions were randomly chosen from a uniform distribution with minimum 20 and maximum 50 then rounded to the nearest whole number. The four precisions were 46, 37, 43, and 33. These were held constant throughout the simulations while sample size and common mean vector were varied. The common mean vectors used were the same as before:

1. All equal: $(0.25, 0.25, 0.25, 0.25)$
2. All slightly different: $(0.1, 0.2, 0.3, 0.4)$
3. One large, while the rest are small: $(0.7, 0.1, 0.1, 0.1)$
4. All near the boundaries: $(0.97, 0.01, 0.01, 0.01)$.

The sample sizes used were also similar to those used with three groups:

1. Large and equal: $\mathbf{100} = (100, 100, 100, 100)$
2. Moderate and equal: $\mathbf{20} = (20, 20, 20, 20)$
3. Small and equal: $\mathbf{10} = (10, 10, 10, 10)$
4. Very small and equal: $\mathbf{5} = (5, 5, 5, 5)$
5. Unequal moderate sample sizes: $(20, 30, 40, 50)$.

Simulations for only the non-bias corrected L-BFGS-B method were conducted since this is the method we planned to use going forward and it was shown that bias correcting had little impact on type I errors. The results of the simulation are shown in Table 5.3. For samples of size 100, the type I error rate ranges from 0.0536 to 0.0605 for the cases where the proportions in the vector are not near the boundaries of 0 and 1. For the

π	$\mathbf{100}$	$\mathbf{20}$	$\mathbf{10}$	$\mathbf{5}$	$(20, 30, 40, 50)$
(0.25, 0.25, 0.25, 0.25)	0.0536	0.0725	0.0925	0.1430	0.0601
(0.1, 0.2, 0.3, 0.4)	0.0556	0.0722	0.0903	0.1497	0.0643
(0.7, 0.1, 0.1, 0.1)	0.0605	0.0726	0.0943	0.1522	0.0632
(0.97, 0.01, 0.01, 0.01)	0.0221	0.1431	0.1874	0.2664	0.0947

Table 5.3: Type I error rates for varying mean vectors and sample sizes for the four group case.

vector with proportions $(0.97, 0.01, 0.01, 0.01)$, the type I error is 0.221, which is not near the nominal rate of 0.05. As sample size decreases, the simulated type I error rate drifts away from the nominal rate. In the best case scenario, where the vector of proportions is a constant 0.25, the type I error rates range from 0.0725 for a sample of size 10 to 0.1430 for a sample of size 5. The type I error rate is reasonable for a sample of size 20, but is large for a sample of size 10 and a sample of size 5. Because the type I error rate is much larger than the nominal rate when sample size is smaller than 20, we would not recommend using this method for samples of size less than 20. The type I error rate increases for a fixed sample size as the proportions move closer to the boundaries of 0 and 1. The unequal sample size of $(20, 30, 40, 50)$ had little effect on the type I error rate as the smallest sample size was still larger than 20.

The set of simulations conducted for four groups was conducted with five groups. The precisions chosen were 35, 25, 26, 42, and 47. The unequal sample size case had samples of sizes 20, 25, 35, 40, 45. The results are shown in Table 5.4. The simulated type I error rates for the five group setting are slightly larger than the simulated type I error rate for four groups for the respective proportion vector and sample size combinations. Since there is a large discrepancy between the nominal rate and the simulated rate when samples sizes are smaller than 20, again, we would not recommend this approach for small sample sizes. Unequal sample sizes do not impact the type I error rate in the five group case as long as all sample sizes are greater than 20. The type I error rates for the

π	$\mathbf{100}$	$\mathbf{20}$	$\mathbf{10}$	$\mathbf{5}$	$(20, 25, 35, 40, 45)$
(0.25, 0.25, 0.25, 0.25)	0.0602	0.0771	0.0954	0.1651	0.0700
(0.1, 0.2, 0.3, 0.4)	0.0638	0.0737	0.0942	0.1670	0.0704
(0.7, 0.1, 0.1, 0.1)	0.0609	0.0771	0.1015	0.1735	0.0693
(0.97, 0.01, 0.01, 0.01)	0.0014	0.0794	0.1447	0.2407	0.0442

Table 5.4: Type I error rates for varying mean vectors and sample sizes for the five group case.

proportion vector $(0.97, 0.01, 0.01, 0.01)$ are anomalous. For both the four group case and the five group case, the error rates bounce from being much smaller than the nominal rate when sample sizes are 100 to being much larger than the nominal rate when sample sizes are small.

5.7. Summary

It is relatively painless to extend the single layer Dirichlet likelihood ratio test to more than two groups. Increasing the number of groups from three to four and then to five does not have a large impact on type I error rates. The type I error rates are largely driven by sample size and the common mean vector. For samples sizes of 100, type I error rates range from 5% - 6% for three to five groups. For samples sizes of 20, type I error rates increase to between 7% and 8%. For samples of size 10, this rate is between 9% and 10%, and for a sample of size 5, the error rate balloons to between 14% and 17%. Based on this result, the likelihood ratio test is not recommended for samples of size less than 20. The type I error rates were not affected by unbalanced sample sizes as long as the sample sizes were large enough. The test has high type I error rates when the values of the common mean vector are near the boundaries of (0,1). A different test needs to be developed in cases where the researcher suspects the mean vector to contain values near zero and values near one.

Chapter 6
Analyses for the NESTED DD for $G > 2$ groups

Near the end of Section 1.5, we saw that there were a few studies published that developed tests for differences in mean vectors for $G > 2$ groups when the data was generated from a nested Dirichlet-multinomial model. Tests for $G > 2$ do not exist when the data is generated from a nested Dirichlet model. We present such a test in this chapter.

6.1. The Overall Test

The test for equal means among G independent samples from a Dirichlet model can be extended to a test for equal means among G independent samples in a nested Dirichlet model. To demonstrate the mechanics of the test, the tree for the water maze data will be used. The tree, with branches labeled with the appropriate mean parameters, is shown in Figure 6.1. Each parameter needs an additional subscript to denote which group the parameter is associated with. To keep notation clear, we separate the group number from the other subscripts with a comma. For example, $\pi_{1,2}$ is the parameter π_1 pictured in the tree diagram for group 2 and $\pi_{12,3}$ is the parameter π_{12} for group 3.

Suppose that three independent samples have been taken.
Let $\boldsymbol{\pi_j} = (\pi_{1,j}, \pi_{2,j}, \pi_{11,j}, \pi_{12,j}, \pi_{21,j}, \pi_{22,j})$ for $j = 1, 2, 3$. The overall test is a test of the hypotheses:

$$H_0 : \boldsymbol{\pi_1 = \pi_2 = \pi_3} \qquad Ha : \text{At least one pair of mean vectors differs.} \tag{6.1}$$

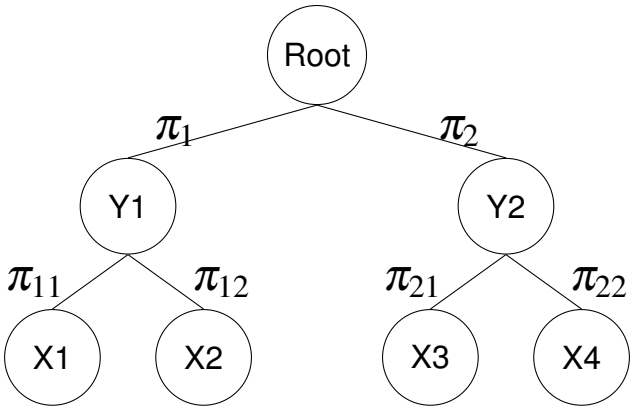


Figure 6.1: Tree diagram used to illustrate the overall test for equal means labeled with parameters.

An implicit assumption of this test is that the three groups being sampled from have the same nesting structure. Hence, it should be determined whether the nesting structure for the groups are similar before a test for equal means is carried out. The overall test for equal means does not assume that the the precision values for the groups are the same. Hence, the precisions values among the three groups at any level may differ.

In this example, there are three subtrees that are each modeled by a separate Dirichlet distribution: one with the root as the parent node, one with Y1 as the parent, and one with Y2 as the parent. We will call these subtrees layers. We will always use l to index the layers. Suppose a tree has L layers. Since each layer is conditionally independent of the other layers, the likelihood ratio test statistic in general is $\Lambda_{overall} = \sum_{l=1}^{L} \Lambda_l$ where each Λ_l is defined as in Equation 5.12 for each layer. The test statistic $\Lambda_{overall}$ follows a χ^2 distribution with degrees of freedom give by the sum $\sum_{l=1}^{L} GK_l - G - K_l + 1$ where G is the total number of groups and K_l is the number of components of the mean vector at each layer.

6.2. Simulation Study: Overall Test

To determine how well the overall test performs, a simulation study was conducted with three groups with the tree structure shown in Figure 6.1. The parameter values used for the simulated datasets were based on the water maze data. The mean values used were (0.42, 0.58) for the top layer, (053, 0.47) for the left bottom layer, and (0.38, 0.62) for the right bottom layer. The precision parameter values were chosen randomly from the range of 10 to 30 and rounded to the nearest whole number. For group 1, these were 13 for the top layer, 23 for the left bottom layer, and 29 for the right bottom layer. For group 2 the precision values were (22, 24, 18), and for group 3, they were (27, 19, 12) where the order is that specified for group 1.

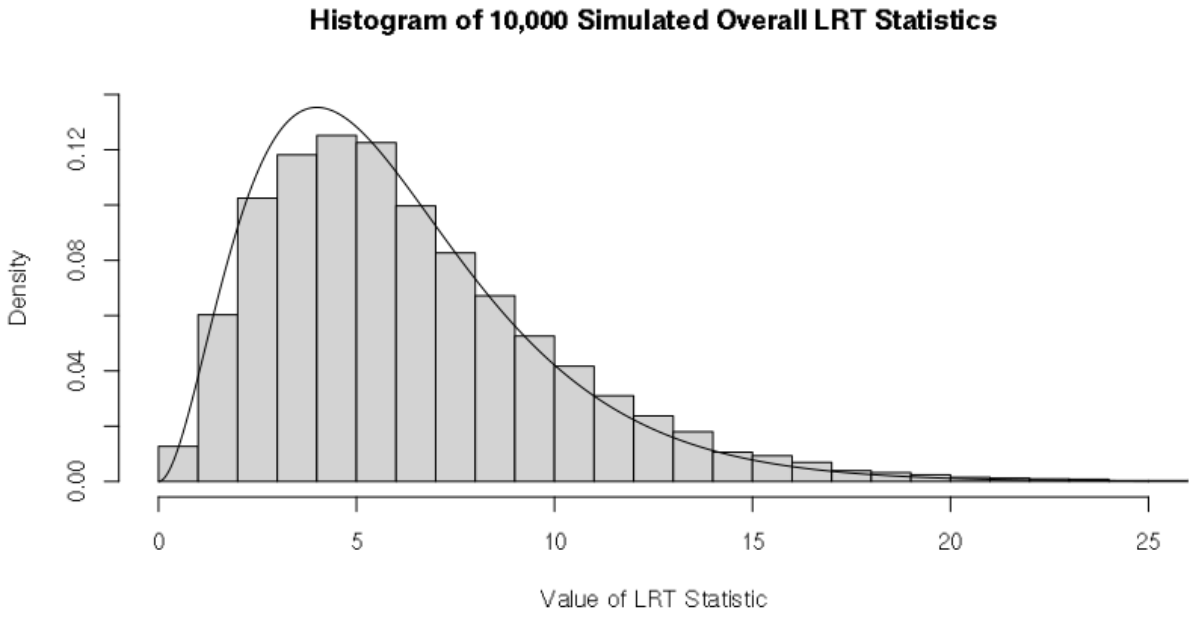


Figure 6.2: 10,000 simulated likelihood ratio test statistics for the nested Dirichlet case with three groups where the per group sample size is 100. The mean vectors at each layer for the three groups is assumed to be the same. The smooth curve is the χ^2 distribution with 6 degrees of freedom.

In the first simulation, the probability of a type I error for a large sample size was found. The simulation used three equal sample sizes of 100. A histogram of 10,000 simulated likelihood ratio test statistics as well as the χ^2 distribution with 6 degrees of freedom is shown in Figure 6.2. The theoretical distribution is more peaked than the histogram of the simulated data, but overall, the χ^2 distribution provides an adequate fit, in particular, the tail of the distribution does not deviate far from the histogram. The fit in the tails is important because the area in the tails is what gives us the p-value. The empirical probability of a type I error with these simulation settings is 0.0691. Just like in the situation without any nesting, we investigate the type I error rate for this nested model as sample size is varied.

6.3. Type I Error Rates and Power of the Overall Test for Nested Designs

As in the non-nested designs, we used six different sample size settings. If (n_1, n_2, n_3) is the vector of sample sizes for each group, then the sample size settings were $\mathbf{100} = (100, 100, 100)$, $\mathbf{40} = (40, 40, 40)$, unequal $= (20, 40, 60)$, $\mathbf{20} = (20, 20, 20)$, $\mathbf{10} = (10, 10, 10)$, $\mathbf{5} = (5, 5, 5)$. Using the precisions and mean vectors stated above, the empirical type I error rate was calculated using 10,000 simulations. The type I error rate is shown in Table 6.1.

As was seen with the non-nested Dirichlet models, the type I error rate is larger than 5% even with large sample sizes. This procedure should not be used with sample sizes of less than 20 as the type I error rate is close to 12% with a sample size of 10 and nearly a quarter with a sample size of 5. The error rate for the unequal sample size is slightly smaller than the error rate for the equal sample size with the same number of total observations. This same pattern was seen with the non-nested Dirichlet model.

sample size	$\mathbf{100}$	$\mathbf{40}$	unequal	$\mathbf{20}$	$\mathbf{10}$	$\mathbf{5}$
Type I	0.0638	0.0735	0.0717	0.0866	0.1229	0.2428

Table 6.1: Type I error rates for varying sample sizes when a nested Dirichlet likelihood ratio test is used. The error rates were calculated from 10,000 simulations.

6.4. Power Calculations

Now that the model being used is a nested Dirichlet distribution, there are even more parameters to change when investigating power. To keep things simple, the mean vector for the first two groups is set to be the same while the mean vector for the third group is different. The mean vector for the third group was only altered at the top level. Recall, the mean vector used for the top layer was $(0.42, 0.58)$. The mean vector for the top layer of the third group is $(0.42+\delta, 0.58-\delta)$ where $\delta \in (-0.10, 0.10)$. The precisions from the type I error rate simulations were used.

Figure 6.3 plots the power curves for the usual sample size settings. The power curves for the nested model resemble the power curves from the non-nested model with one clear exception. The power for the nested design is less than the power for the non-nested design for all sample size and effect size combinations. This makes sense as a more complicated model will require a larger sample size to correctly estimate the parameters.

Another aspect of Figure 6.3 that is notable is that the unequal sample size case has a greater power than the case where each group has a sample of size 40. Again, this is likely due to the fact that the largest sample, 60, is allocated to the group that has a different mean vector. None of the power curves reach the nominal type I error rate of 5% at $\delta = 0$. The power for a sample size of 5 each is low for every effect size (except where it is not supposed to be at $\delta = 0$). A sample size of 40 each gives adequate power when $|\delta| > 0.05$. The power curves are roughly symmetric.

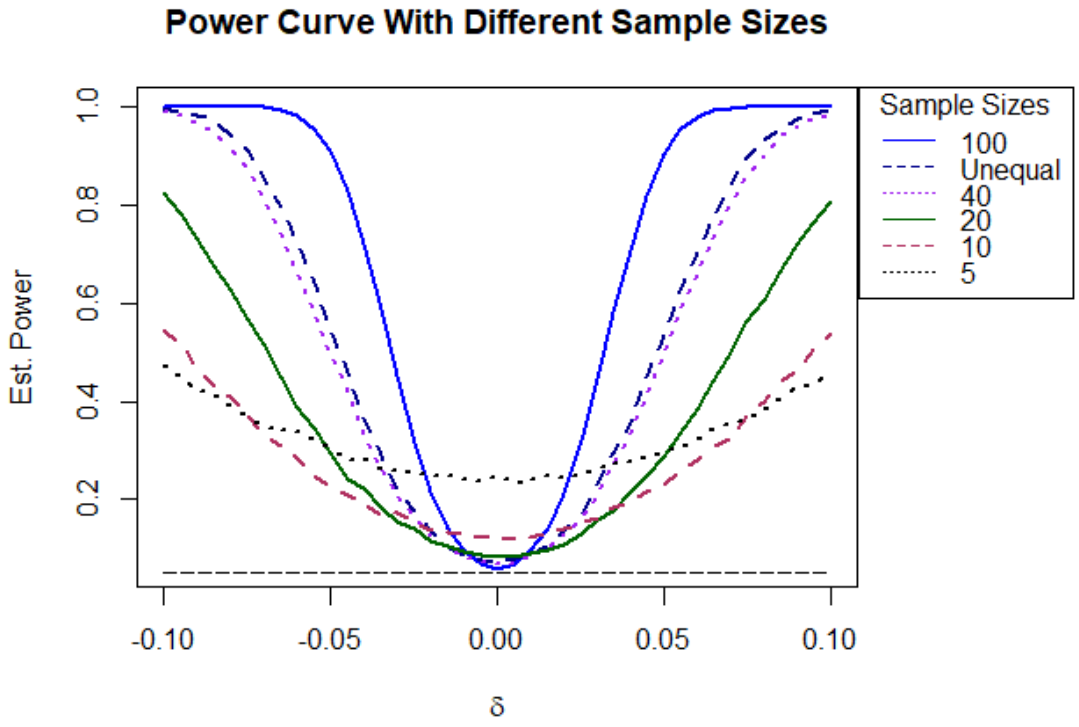


Figure 6.3: The power curves for the overall test for the nested Dirichlet distribution. The precisions are held fixed over all groups. The mean vector at the top layer is varied for the third group. The sample size for each group is varied.

6.5. Conclusion

The single layer likelihood ratio test for multiple groups developed in Chapter 5 was extended to a likelihood ratio test for a nested Dirichlet distribution for more than two groups. A simulation study was conducted to examine how well the test performs in the case when there are four components related through the tree shown in Figure 6.1. The test had a type I error rate of 0.0866 and adequate power when the sample size was 20. Power decreased substantially and type I error rates were much larger than the nominal rate with a sample size of 10 or below. Increasing the number of layers in the nesting tree will likely exacerbate these problems. We recommend using the nested Dirichlet likelihood ratio test when there are multiple groups as long as sample size is large.

Chapter 7
Application of Methods to the Baseball Dataset

In this chapter, we apply the multigroup likelihood ratio test for equal means, tree finding algorithm, and pruning procedure to a real life dataset.

7.1. Dataset Description

Null (2009) compiled a large dataset of outcomes that occur whenever a batter makes a plate appearance in baseball. The data tracks the performance of 1975 unique major league baseball players from the years 2000 to 2010. Since the paper was published in 2009 and the dataset contains observations from 2010, it is assumed that Dr. Null continued to collect data after the publication. The variables in the dataset include the batter's ID (a number between 1 and 3136), the year, the birth year of the batter, an indicator variable for the handiness of the batter, an indicator variable for the handiness of the pitcher, the total number of plate appearances, and 14 columns counting the number of times each of the 14 plate appearances outcomes occurred. The 14 plate appearances outcomes were:

1. Interference
2. Hit by Pitch
3. Base on Balls
4. Strikeout
5. Fly Ball Home Run
6. Fly Ball Triple
7. Fly Ball Double
8. Fly Ball Single
9. Fly Ball Out
10. Ground Ball Home Run

11. Ground Ball Triple
12. Ground Ball Double
13. Ground Ball Single
14. Ground Ball Out

One question posed by the author is does the composition of these 14 events depend on the batter's age. A similar question is whether a batter's performance changes with age.

A quantity used to measure a batter's performance is slugging percentage. The formula to calculate slugging percentage is:

$$\text{Slugging Percentage} = \frac{\text{Number of Singles} + 2 \cdot \text{Number of Doubles} + 3 \cdot \text{Number of Triples} + 4 \cdot \text{Number of Home Runs}}{\text{Number of At Bats}}. \tag{7.1}$$

Slugging percentage is a misnomer; a slugging percentage can take on any value between 0 and 4 and can be thought of as the average number of bases a batter gets per at bat. Since scoring a base hit in baseball is difficult and most of the time batters are struck out, slugging percentages in practice are less than one. For example, Babe Ruth was a spectacular batter and had a career slugging percentage of 0.6897.

To answer the question of how a batter's performance changes with age, slugging percentage was used as the basis of our analysis. Instead of having 14 different outcomes for plate appearances, the data was grouped into just six outcomes. These were

1. Home Runs
2. Singles
3. Doubles
4. Triples
5. Outs
6. Other

The Other category groups the outcomes interference, hit by pitch, and base on balls. Note that interference in this case means that the batter interfered with the catcher's attempt to retire a runner who is stealing a base during the batter's plate appearance. When interference is called, the batter is out. These three outcomes do not reflect on a batter's skill to hit the ball like the other five outcomes.

In addition to grouping the 14 outcomes into larger classes, two other adjustments were made before the data was analyzed. Firstly, each batter had two rows of data for each season. One row was for performance against right handed pitchers and the other row was for their performance against left handed pitchers. This data was simplified by combining the rows through the summing of the outcomes and ignoring the handedness of the pitcher. Secondly, the age of each batter each year had to be calculated. The season year was ignored and only age for that year was recorded. Lastly, the counts for each outcome were converted to proportions before analysis.

Once the ages of the batters were determined, the continuous age variable was binned into three groups. Although this binning allows the tests developed in previous chapters to be applied to the baseball dataset, the drawbacks include losing information and using a researcher degree of freedom to determine how the data should be binned (Simmons et al., 2011). If the data is binned in a different way, the results could potentially change. Every age from 20 to 49 years was represented in the data. The first group were youngsters aged 20 - 24. The second group were batters in the prime of their career aged 25 - 34. The last group were batters aged over 35. Younger batters, having less experience, will likely perform poorly in comparison to batters with more experience. Batters over the age of 35 are likely to have a decline in their athletic abilities. Each observation is a season for an individual batter. Thus, it is possible for a batter to be grouped into more than one age group since the data spans 11 years. In total, there were 8099 observations. There were 1063 observations in the first age group, 6021 in the second age group, and 1015 in the last age group. Although the sample sizes for each group varied widely, the

sample size of each is very large which means our test should perform well in terms of power and type I error rate.

There were some observations where the number of plate appearances was very small (less than 10 in a season). Out of the 8099 total observations, 812 of them had fewer than 10 plate appearances. We considered removing these from the dataset before the analysis, but ultimately decided there may be value in keeping the data intact.

7.2. Exploratory Data Analysis

The first thing examined to determine if a nested Dirichlet model would be necessary for this dataset was the correlation structure of the components. If the null hypothesis is true, then there is no difference in the batting compositions between the three groups, and all the data can be grouped together. With just a single group, the correlation matrix is given in Table 7.1.

	Home Run	Triple	Double	Single	Out	Other
Home Run	1.0000	-0.0217	0.0968	-0.0702	-0.2798	0.0764
Triple	-0.0217	1.0000	0.0138	0.0252	-0.0929	-0.0342
Double	0.0968	0.0138	1.0000	0.0034	-0.3239	-0.0042
Single	-0.0702	0.0252	0.0034	1.0000	-0.6772	-0.1180
Out	-0.2798	-0.0929	-0.3239	-0.6772	1.0000	-0.5171
Other	0.0764	-0.0342	-0.0042	-0.1180	-0.5171	1.0000

Table 7.1: The correlation matrix for the baseball data when a single group is constructed. The positive correlations between components are highlighted.

There are small positive correlations between outcomes likely to score points. In total, there are five pairs with positive correlations: (singles, doubles), (singles, triples), (doubles, triples), (doubles, home runs), and (home runs, other). The fact that there are positive correlations between event types indicates that a nested Dirichlet model is a bet-

ter fit than a non-nested Dirichlet model. With a sample size of 8099, a correlation of 0.0252 are larger is statistically different from 0. The 95% confidence interval with a value of $r = 0.0252$ is $(0.003, 0.047)$. The largest of the positive correlations is between home runs and other for the third age group and is 0.1751 (See Table 7.2). Many of the positive correlations are small (< 0.10). Hence, a non-nested Dirichlet may fit just as well. In the analysis stage, both models are fit to determine if using the more complex model is worth it. There are negative correlations between outs and all base hits. This is exactly what we would expect since a base hit and an out are “opposite" events, one helps to score points and another does not.

If we look at the correlation structure for the three groups separately, we get the three matrices shown in Table 7.2. The signs of the correlations for the youngest group resembles that of the combined group with one exception: the correlation between doubles and triples is a negative number with a small magnitude. Since this value is close to zero, this is not very concerning. Similarly, for age group 2, there is a negative correlation between doubles and singles, but the magnitude of the negative correlation is tiny. The correlation structure for the third age group has small magnitude negative correlations for the pairs (doubles, triples) and (singles, triples).

The next piece of exploratory data analysis is to look at the ternary diagrams for the groups to attempt to discern differences between the three. Since there are 8099 individual data points, the ternary diagrams for the three groups were plotted separately. These diagrams are shown in Figure 7.1. For an explanation on how to read and interpret the ternary diagrams, see Section 2.2.

The three sets of diagrams look fairly similar with the only real difference being that the third set is less densely packed since fewer data points belong to this group. Looking at the ternary diagrams in this case drives home the point that a formal test is necessary to determine if there are differences in the mean vectors between the three groups. Attempting to determine whether or not there are differences in compositions by examining

	Home Run	Triple	Double	Single	Out	Other
Home Run	1.0000	-0.0209	0.1248	-0.0388	-0.2047	0.0379
Triple	-0.0209	1.0000	-0.0287	0.0368	-0.1060	-0.0240
Double	0.1248	-0.0287	1.0000	0.0366	-0.3335	-0.0469
Single	-0.0388	0.0368	0.0366	1.0000	-0.7152	-0.1306
Out	-0.2047	-0.1060	-0.3335	-0.7152	1.0000	-0.4824
Other	0.0379	-0.0240	-0.0469	-0.1306	-0.4824	1.0000

	Home Run	Triple	Double	Single	Out	Other
Home Run	1.0000	-0.0130	0.0876	-0.0716	-0.2925	0.0701
Triple	-0.0130	1.0000	0.0377	0.0268	-0.1049	-0.0197
Double	0.0876	0.0377	1.0000	-0.0134	-0.3159	0.0033
Single	-0.0716	0.0268	-0.0134	1.0000	-0.6774	-0.1206
Out	-0.2925	-0.1049	-0.3159	-0.6774	1.0000	-0.5062
Other	0.0701	-0.0197	0.0033	-0.1206	-0.5062	1.0000

	Home Run	Triple	Double	Single	Out	Other
Home Run	1.0000	-0.0663	0.1744	-0.1268	-0.2945	0.1751
Triple	-0.0663	1.0000	-0.0513	-0.0073	-0.0343	-0.0932
Double	0.1744	-0.0513	1.0000	0.0907	-0.3803	0.0237
Single	-0.1268	-0.0073	0.0907	1.0000	-0.6037	-0.0739
Out	-0.2945	-0.0343	-0.3803	-0.6037	1.0000	-0.6478
Other	0.1751	-0.0932	0.0237	-0.0739	-0.6478	1.0000

Table 7.2: The correlations matrices for the baseball data for the three different age groups. The first matrix is the correlation matrix for the youngest, followed by the correlation matrix for baseball players aged 25-34, and then the correlation matrix for older players. The three age groups have similar correlation structures with slight variations. Positive correlations are highlighted.

45 unique diagrams, where some are so densely populated that the entire triangle is filled or nearly filled is no longer a useful method. Hence, the next step is to find an appropriate nesting tree and use our overall test.

7.3. Nested Dirichlet Analysis

In this section, a complete analysis is carried out on the baseball dataset using a nested Dirichlet model.

7.3.1. Finding and Evaluating the Nesting Tree

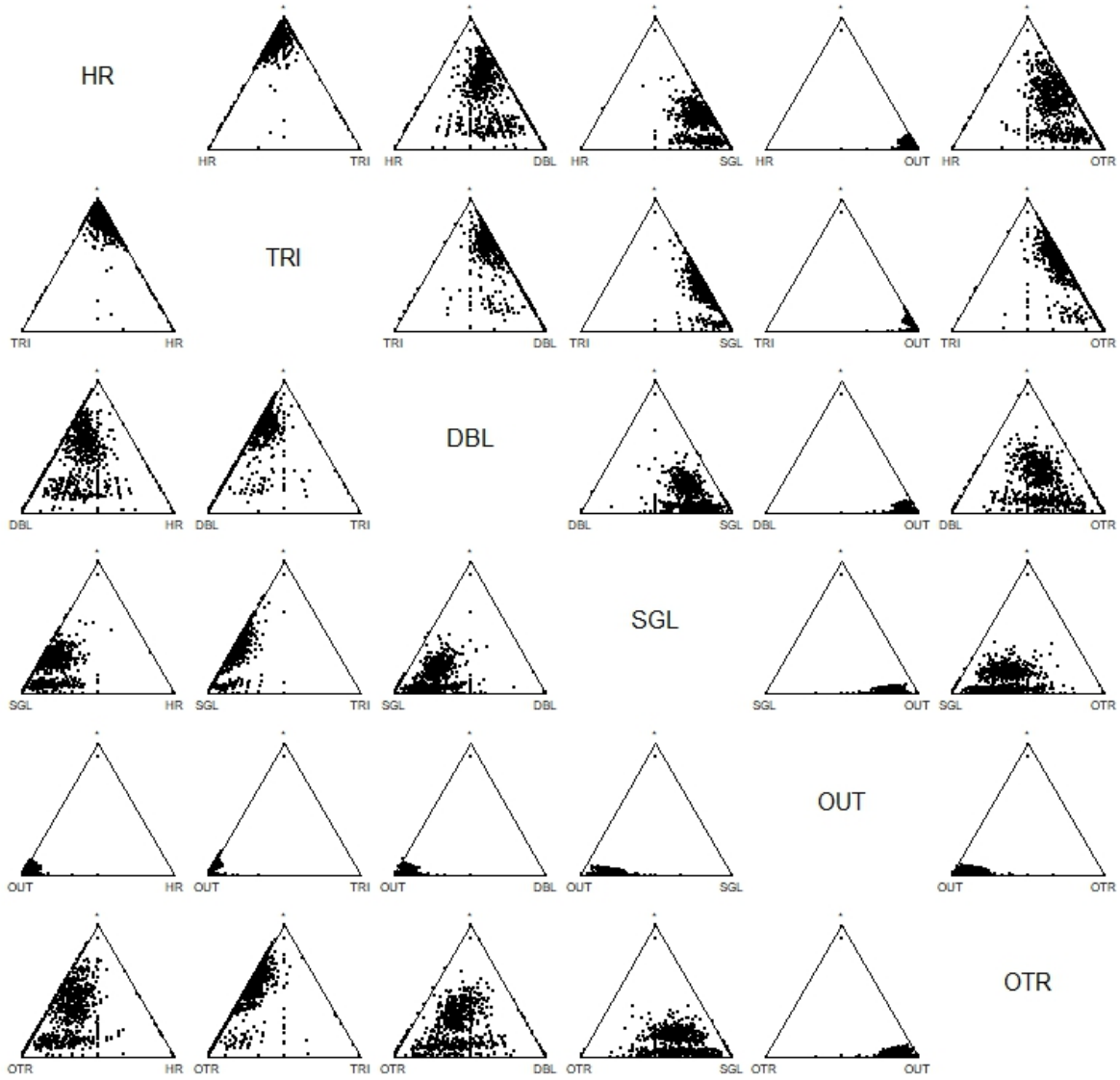
To determine which nesting tree to use, the data, as one big group, was fed into the tree finding algorithm presented in Chapter 7 of Turner (2013). Before the data was inputted into the tree finding algorithm, a procedure was performed to the dataset to deal with zero entries. Each value x in the dataset was transformed using the function:

$$x_{\text{new}} = \frac{x_{\text{old}} + \varepsilon}{1 + 2 \cdot \varepsilon} \tag{7.2}$$

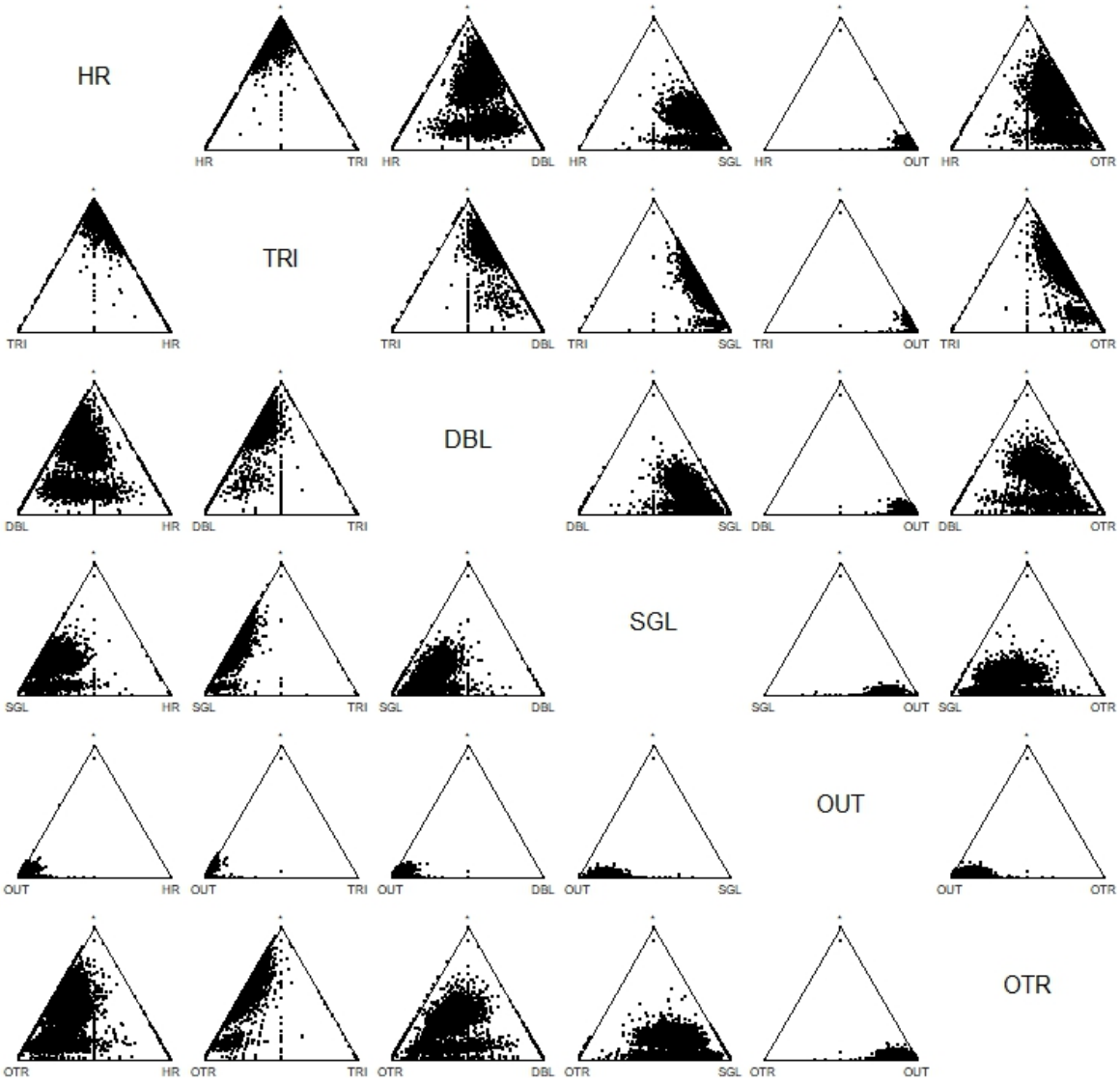
where ε was chosen to be 10^{-5}. The values were then re-scaled such that every row has a sum of one. This method of dealing with zeros was chosen since it is the same method as seen in the function `dirichlet.mle` in the `sirt` package (Robitzsch, 2020).

The tree with the maximum likelihood amongst all binary trees is shown in Figure 7.2. Since over-fitting has been shown to have little impact on the conclusions of the analysis, we did not check to see if the tree should be collapsed in the initial analysis (Turner, 2013). In Section 7.4 the analysis is completed using a tree with no nesting. In Section 7.5, we determine, using the test presented in Chapter 3, if a simpler tree should be used.

Age Group 1 Ternary Diagrams



Age Group 2 Ternary Diagrams



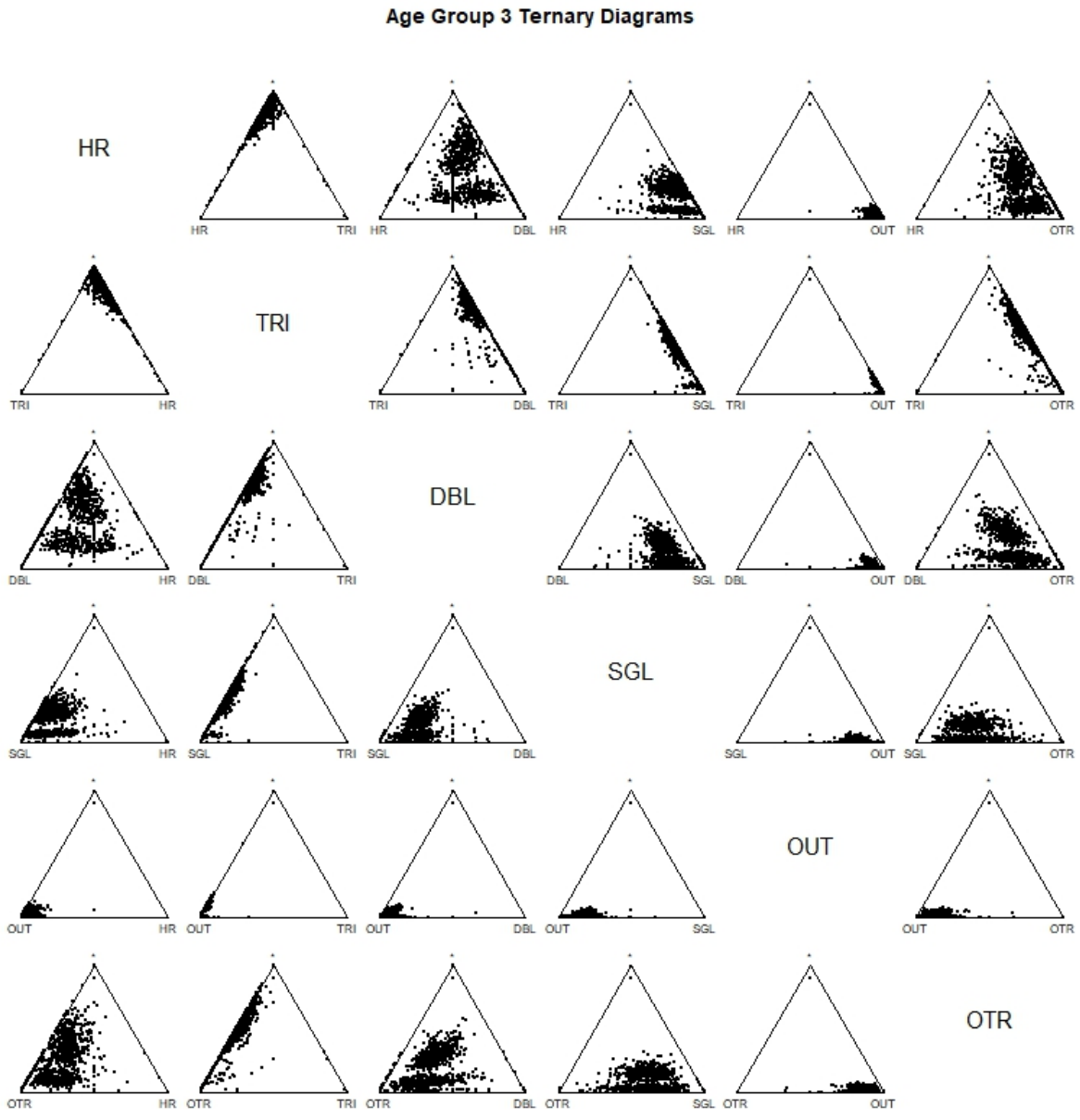


Figure 7.1: Ternary diagrams for the baseball dataset. Each subplot represents a different age group.

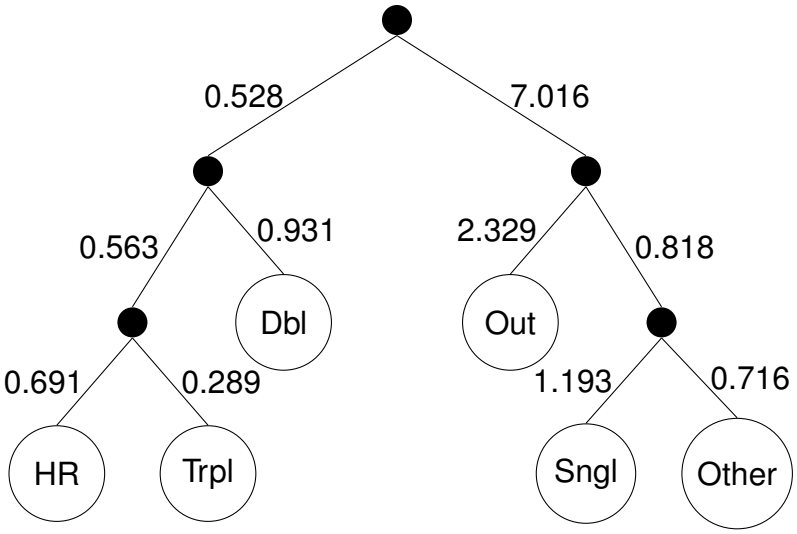


Figure 7.2: The binary nesting tree labeled with the MLEs of the α parameters when the baseball dataset is fed into the tree finding algorithm as one large group.

To check whether the tree produced by the algorithm was an adequate fit, first the α parameters associated with each branch were estimated. The MLEs of α parameters seen in Figure 7.2 were estimated using `dirichlet.mle` from the package `sirt` (Robitzsch, 2020). The entire dataset was used to estimate the alpha parameters instead of breaking the data down by age group. There are some data entries that are zero. A method for handling zeros is built into the function `dirichlet.mle`. Each x vlaue is transformed using the function in Equation 7.2. Next, to examine whether the tree in Figure 7.2 produces data similar to the real dataset, 10,000 vectors of simulated data were drawn using the tree model. The empirical correlation structure was computed from the simulated data. This was compared to the correlation structure generated by the real dataset. Table 7.3 shows the absolute difference between the correlation matrix for the real data and the simulated data. Table 7.4 displays the correlation matrix for the real dataset and Table 7.5 displays the correlation matrix for the simulated data.

There are many notes to make about the two correlation matrices. In the correlation matrix produced from the simulated data, the components in the left side of the tree are positively correlated with each other. These are the more favorable outcomes of home

	Home Run	Triple	Double	Single	Out	Other
Home Run	0	0.188	0.071	0.025	0.090	0.126
Triple		0	0.116	0.060	0.035	0.005
Double			0	0.086	0.064	0.066
Single				0	0.071	0.212
Out					0	0.066
Other						0

Table 7.3: This table shows the absolute differences between the correlations from the real data and the simulated data when individual age groups are not considered.

	Home Run	Triple	Double	Single	Out	Other
Home Run	1.000	-0.022	0.097	-0.070	-0.280	0.076
Triple	-0.022	1.000	0.014	0.025	-0.093	-0.034
Double	0.097	0.014	1.000	0.003	-0.324	-0.004
Single	-0.070	0.025	0.003	1.000	-0.677	-0.118
Out	-0.280	-0.093	-0.324	-0.677	1.000	-0.517
Other	0.076	-0.034	-0.004	-0.118	-0.517	1.000

Table 7.4: The sample correlation matrix generated by the real dataset. Positive Correlations are highlighted.

run, triple, and double. The components in the right side of the tree, the less favorable outcomes of out, single, and other, are negatively correlated with each other. The set of less favorable outcomes and the set of favorable outcomes are also negatively correlated with each other with one exception: that of single and other. This is why single and other are nested under a different internal node than out. There are differences between the correlation matrix of the actual data and the correlation matrix of the simulated data. These differences occur only when the magnitude of the correlation between the real components was small (< 0.2). The correlation structure between the components is preserved where it counts. Four correlations differ by a magnitude larger than 0.10 and only one differs by a magnitude larger than 0.20.

	Home Run	Triple	Double	Single	Out	Other
Home Run	1.000	0.166	0.168	-0.045	-0.190	-0.050
Triple	0.166	1.000	0.130	-0.035	-0.128	-0.039
Double	0.168	0.130	1.000	-0.083	-0.260	-0.070
Single	-0.045	-0.035	-0.083	1.000	-0.748	0.094
Out	-0.190	-0.128	-0.260	-0.748	1.000	-0.583
Other	-0.050	-0.039	-0.070	0.094	-0.583	1.000

Table 7.5: The correlation matrix of the simulated baseball data using the tree structure shown in Figure 7.2. The positive correlations are highlighted.

Table 7.6 lists the top four largest correlations by magnitude for the real data and the associated simulated data, as well as the absolute differences between them. These values differ by less than 0.09. The fact that many of the correlations seen in the real dataset are close to zero and the correlations with the largest magnitude are seen between the component out and another component suggests a simpler tree may work just as well.

Components	Data Correlation	Simulated Correlation	Absolute Difference
Out and Single	-0.677	-0.748	0.071
Out and Other	-0.517	-0.583	0.066
Out and Double	-0.324	-0.260	0.064
Out and Home Run	-0.280	-0.190	0.090

Table 7.6: The table displays the largest four correlations by magnitude in order for both the real dataset and the simulated dataset. The last column displays the absolute difference between the correlations of the real data and simulated data.

7.3.2. The Overall Test

The first test we conduct is the overall test for differences in mean vectors presented in Chapter 6. Let π_1 be the mean vector of age group 1, π_2 be the mean vector for age group 2, and π_3 be the mean vector for age group 3. The overall test is:

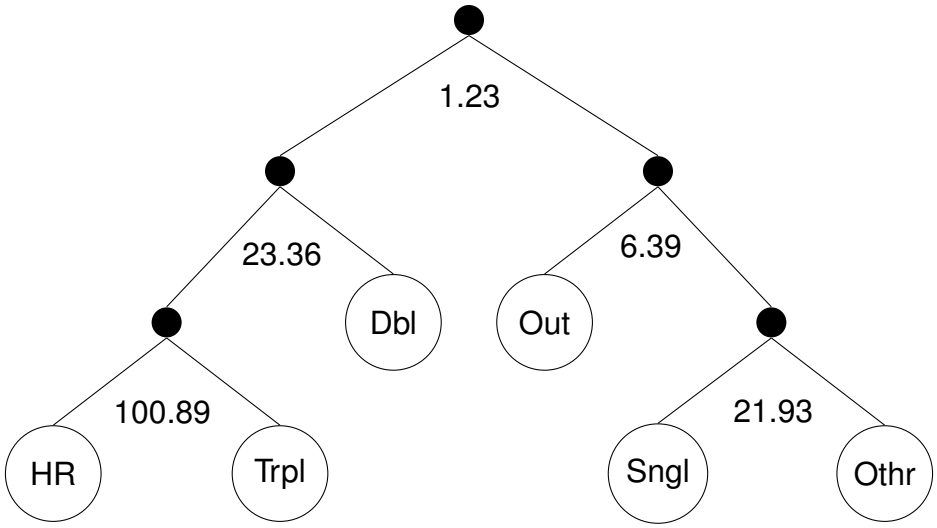


Figure 7.3: The value of the likelihood ratio test statistic at each layer of the nesting tree for the overall test applied to the baseball dataset.

$$\begin{aligned} &H_0 : \boldsymbol{\pi_1 = \pi_2 = \pi_3} \\ &H_a : \text{At least one pair of mean vectors differ} \end{aligned} \tag{7.3}$$

The likelihood ratio test statistic value at each layer of the tree is presented in Figure 7.3. The likelihood ratio test statistic for the overall test is the sum of these values, 153.80. The number of free parameters under the alternative hypothesis of different mean vectors for each group is $3(10) = 30$. The number of free parameters under the null hypothesis of the same mean vector for all three age groups is $5(3) + 5 = 20$. Thus, the likelihood ratio test statistic follows an approximately χ^2 distribution with $30 - 20 = 10$ degrees of freedom. The p-value associated with this test statistic is 6.5×10^{-28}. Clearly, the mean vectors between the age groups differ. The next step is to do pairwise comparisons tests between age groups. The p-value for the overall test is so small that null will be rejected no matter what adjustment is made for pairwise comparisons.

7.3.3. Pairwise Comparison Between Groups

The conclusion of the overall test is that at least one pair of mean vectors differ between groups. There are three follow-up hypothesis tests that can be done to determine between which two groups differences exist. The null hypotheses for these three tests are:

$$H_0 : \boldsymbol{\pi_1} = \boldsymbol{\pi_2} \tag{7.4}$$

$$H_0 : \boldsymbol{\pi_1} = \boldsymbol{\pi_3} \tag{7.5}$$

$$H_0 : \boldsymbol{\pi_2} = \boldsymbol{\pi_3}. \tag{7.6}$$

Using the tree structure in Figure 7.2, the pairwise overall tests were carried out. The likelihood ratio test statistic for each layer and each pairwise comparison is given in Table 7.7. The labeling for the test statistic is given in Figure 7.4. The number of free parameters for each pairwise test under the alternative hypothesis is $2(10) = 20$ and the number of free parameters under the null is $2(5) + 5 = 15$. Thus, the overall test statistic follows an approximate χ^2 distribution with 5 degrees of freedom.

LR Number	Group 1 vs Group 2	Group 1 vs Group 3	Group 2 vs Group 3
1	1.36	1.33	3.73
2	11.59	10.58	16.64
3	7.63	3.37	1.36
4	67.92	80.59	37.02
5	4.90	19.81	15.4
Overall Statistic	93.40	115.68	74.19
P-value	$1.3 \cdot 10^{-18}$	$2.6 \cdot 10^{-23}$	$1.4 \cdot 10^{-14}$

Table 7.7: The test statistics for the pairwise tests. The individual LR test statistics correspond to the labeling seen in Figure 7.4.

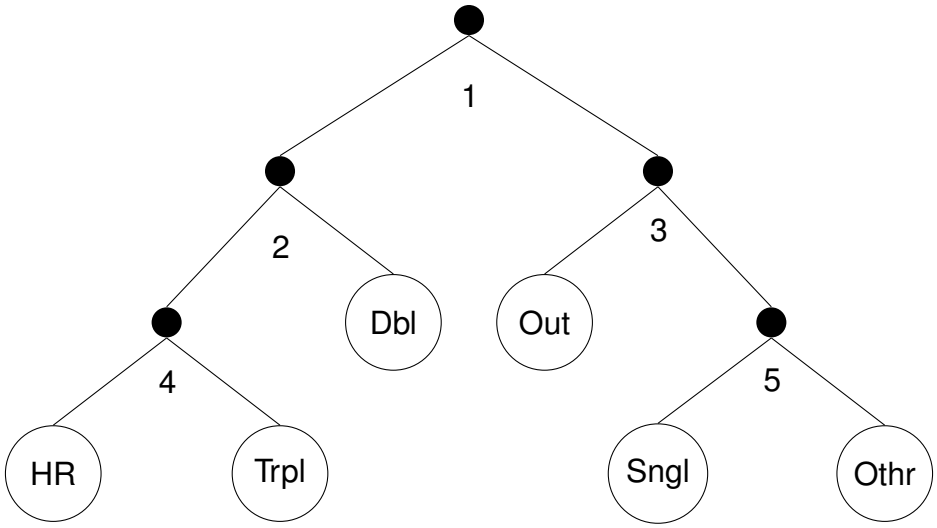


Figure 7.4: This figure shows the labeling of each layer of the tree. The numbers correspond with Table 7.7

The p-value for the two group comparisons are extremely small. Since the p-values are so small, we do not need to worry about adjusting our rejection regions for multiple tests. Based on the tests, each of the three groups have different mean vectors. The major contribution to the χ^2 statistic for all four tests comes from the subtree containing home runs and triples.

What if the tree structure used is not the correct one? Would the results be the same? In the next section, the overall test is completed with a single layer Dirichlet distribution to see if the conclusion remains the same.

7.4. Single-Layer Dirichlet Analysis

Suppose we analyze the data as a Dirichlet distribution with a single layer. This “tree" is shown in Figure 7.5. It can be argued that the single layer Dirichlet model is appropriate in this scenario since the positive correlations between components are small and may not be significantly different than zero.

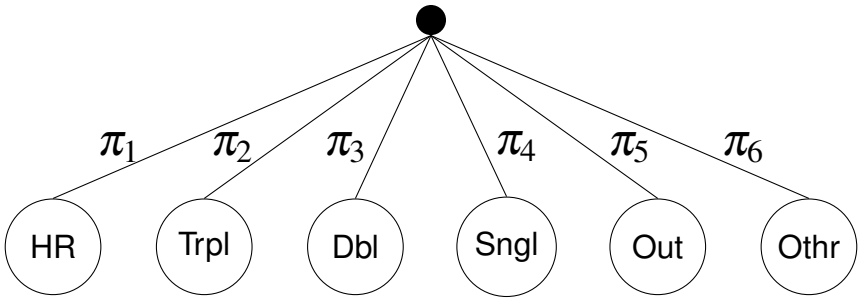


Figure 7.5: The single layer Dirichlet model for the baseball data.

Repeating the overall test for differences between the mean vector for the three groups, we get a test statistic of 13389.76. The number of free parameters under the null hypothesis is 8 and the number of free parameters under the alternative hypothesis is 3(6)=18. Thus, the test statistic follows an approximately χ^2 distribution with 10 degrees of freedom. The p-value here is approximately zero. Thus, we reach the same conclusion as we did using the multi-layered tree.

If we conduct pairwise comparison tests using a single layer tree, the results are just as un-shocking. For groups 1 and 2, the test statistic is 12052.9. For groups 1 and 3, it is 3181.4, and for groups 2 and 3, it is 11550.5. The p-value associated with the tests statistics is approximately zero. There are differences between the mean vectors between any two groups. This indicates that even if the algorithm did not pick a good fitting tree, we may get the same conclusion.

7.5. Testing if the Nesting Tree is Collapsible

The tree in Figure 7.2 is a binary tree. It may be possible to collapse the tree so that there are fewer layers and, in turn, more than two branches radiating from the internal nodes. This procedure was introduced in Chapter 3. There are four different internal nodes that may be removed.

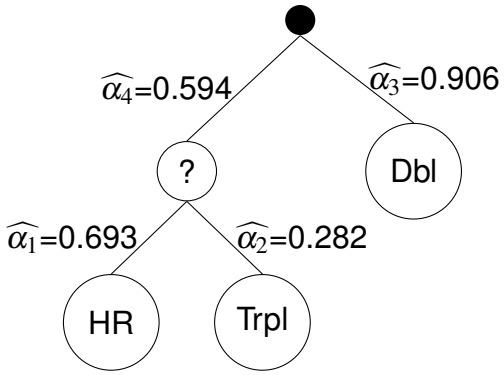


Figure 7.6: The first subtree we investigate to determine if the node labeled with a question mark can be removed. The values represent the values of the MLEs of the α parameters.

We start by considering the subtree in Figure 7.6. Note that the α estimates are slightly different. This is because the issue of zero values in the data was corrected before these estimates were made. The internal node labeled with the question mark is the one that is being considered for removal. We construct a confidence interval using Equation 3.3 where the variance is given by Equation 3.2. The 95% confidence interval for the quantity $\alpha_4 - \alpha_1 - \alpha_2$ is $(-0.382, -0.381)$. Since this confidence interval does not contain zero, we retain the node in question.

The next subtree we investigate is shown in Figure 7.7. The 95% CI for the parameter $\alpha_4 - \alpha_1 - \alpha_2$ is $(-1.056, -1.054)$. Thus, this internal node is also retained. The 95% CI for the parameter $\alpha_4 - \alpha_1 - \alpha_2$ for the subtree in Figure 7.8 is $(-0.887, -0.886)$. Again, since the confidence interval does not contain 0, the internal node is retained. Lastly, the 95% CI for the parameter $\alpha_4 - \alpha_1 - \alpha_2$ for the subtree in Figure 7.9 is $(4.651, 4.751)$; the node is retained.

Table 7.8 lists the four confidence intervals that were generated from the pruning method. The binary tree that was generated from the tree finding algorithm could not be collapsed at any of the internal nodes. These results are consistent with the naive approach of constructing 95% confidence intervals for the correlation coefficients in order

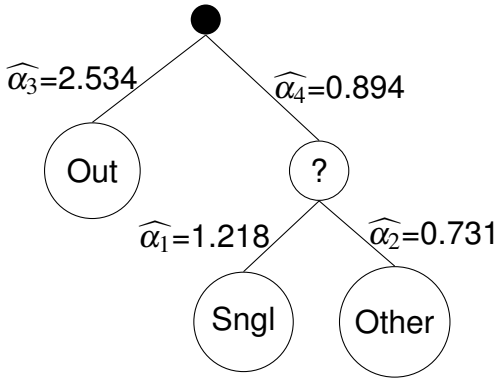


Figure 7.7: The second subtree we investigate to determine if the node labeled with a question mark can be removed.

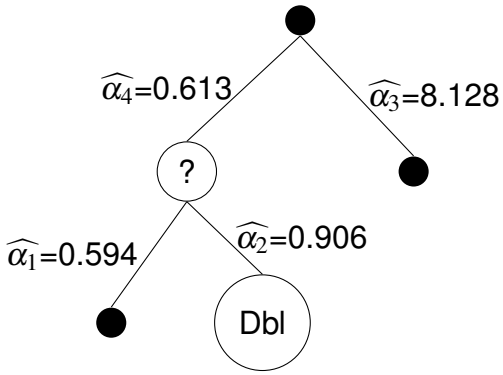


Figure 7.8: The third subtree we investigate to determine if the node labeled with a question mark can be removed.

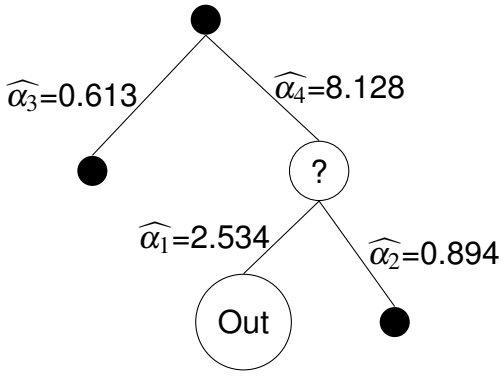


Figure 7.9: The fourth subtree we investigate to determine if the node labeled with a question mark can be removed.

to determine which are statistically different from zero. This leads more support for the fact that the full tree in Figure 7.2 fits the data well.

Subtree	Lower Endpoint	Upper Endpoint	Contains zero?
1	-0.382	-0.381	No
2	-1.056	-1.054	No
3	-0.887	-0.886	No
4	4.651	4.751	No

Table 7.8: The confidence intervals generated by applying the pruning method to each of the four subtrees of the baseball dataset. None of the intervals contain zero. Thus, the tree could not be collapsed.

7.6. Conclusion

Using the test generated in Chapter 6, differences in the composition of batting events have been shown between the three age groups. Whether the data was analyzed using a Dirichlet model with a single layer or a nested Dirichlet model, the conclusion that differences in the mean vectors exist between the three age groups was established. The tree finding algorithm presented in Turner (2013) provided an adequate nesting tree to model the data. When the pruning method developed in Chapter 3 was applied, none of the internal nodes could be removed as evidenced by Table 7.8.

	Group 1	Group 2	Group 3
HOME RUN	0.0179	0.0229	0.0233
TRIPLE	0.0059	0.0046	0.0036
DOUBLE	0.0429	0.0431	0.0433
SINGLE	0.1498	0.1487	0.1488
OUT	0.7062	0.6957	0.6835
OTHER	0.0773	0.0850	0.0975

Table 7.9: Sample mean vectors for the three groups using the baseball data.

The largest χ^2 test statistic occurred in the subtree that contained the nodes Home Run and Triple. This is likely where the biggest difference between groups exist. Table 7.9 shows the sample mean vectors for the three age groups. A small change in the proportion of home runs or triples results in a large difference between the groups since the relative change is great.

When conducting pairwise comparisons, the mean vectors were shown to be different for every pair of groups. Between groups 1 and 2, group 1 had fewer home runs but more triples. Between groups 1 and 3, the same comparison is true. The smallest percentage for home runs belongs to the youngest group probably due to a lack of experience and power. Muscle mass reaches a peak later in life. Between groups 2 and 3, the main difference is the percent of triples. The sample percentage of triples for group 2 was much greater than group 3. As players move past age 35, the combined percent of triples and home runs will slightly decrease. Perhaps the decrease in triples indicates less power, on average, as players age.

A possible problem with the analysis is that the subtrees that contribute the most to the value of the test statistic are the ones where the proportions are near zero. Home runs and triples have the smallest probability of occurring. Looking at subtree 4 in Figure 7.4 and the value of the test statistic associated with subtree 4 shown in 7.7, it is apparent that the variables home run and triple really drive up the value of the overall statistic.

There is not much difference between the results of the nested Dirichlet analysis and the single layer Dirichlet analysis. This is a demonstration of the robustness of the likelihood ratio test to the choice of the nesting tree. Even if the tree that is chosen is not the one that generated the data to begin with, the results from the test are still likely to be valid.

The results produced using the nested Dirichlet model in this chapter were surprisingly similar to the results presented in Null (2009). Null uses a more complicated nested

Dirichlet model with 14 event types. Using Figure 7.2, the unconditional probabilities of each of the six event types can be computed. The unconditional probabilities for the same six events using the model in Null (2009) can be calculated from a table presented in the paper. The unconditional probabilities using the two models are presented in Table 7.10. The models yield similar unconditional probabilities.

	Nested DD Model	Null Model
HOME RUN	0.019	0.025
TRIPLE	0.008	0.005
DOUBLE	0.044	0.048
SINGLE	0.150	0.157
OUT	0.688	0.677
OTHER	0.091	0.088

Table 7.10: Summary of the unconditional probabilities of the six event types using the nested Dirichlet model presented in this paper and the nested Dirichlet model in Null (2009).

To address the effects of aging, Null (2009) uses a fixed effects asymmetric quadratic additive model, a model more complicated than what is presented in this chapter. In Null's model, the composition of the 14 event types is given for each age by year from 20 to 42. To compare his results to ours, we averaged his results over the years for the three age groups and aggregated over the event types. Null gives the results as the percentage of increase or decrease for that event type based on age. The results after averaging and aggregating are shown in table 7.11. Based on this table, Null results are similar to what we presented here. Group 1 is the worst at hitting home runs but best at hitting triples. Group 3 is the worst at hitting triples, but the percentage of home runs is nearly the same when compared with group 2.

To better compare our results with those of Null, we ranked the three groups for each event type using our results and Null's results. The ranking is shown in Table 7.12. The rankings are the same for the events home run, triple, and double. There is a discrepancy

Age Group	Home Run	Triple	Double	Single	Out	Other
Group 1	-3.88%	2.01%	-0.86%	-2.10%	6.65%	-7.09%
Group 2	-0.04%	-0.46%	0.10%	0.54%	0.17%	-0.18%
Group 3	-0.66%	-3.07%	0.33%	3.95%	3.88%	-1.07%

Table 7.11: The percent of increase or decrease of each event type after averaging over relevant age in years and aggregating over event types using the results in Null (2009).

when it comes to predicting who hits the most singles. The ranking using both models is completely different. There is one reversal in ranking for outs and one reversal ranking for other. Overall, the results from the two models tended to corroborate each other.

	Home Run		Triple		Double		Single		Out		Other	
	ND	NL	ND	NL	ND	NL	ND	NL	ND	NL	ND	NL
Group 1	3	3	1	1	3	3	1	3	3	3	3	3
Group 2	1	1	2	2	2	2	3	2	2	1	2	1
Group 3	2	2	3	3	1	1	2	1	1	2	1	2

Table 7.12: The ranking for each age group and each event type using the nested Dirichlet model (ND) presented in this paper and Null's model (NL). The ranking is the same for three of the six events.

Chapter 8
Applications of Methods to Job Type dataset

No dataset perfectly matches the assumptions of a model. The baseball dataset was a mismatch for the methods developed in this dissertation for three reasons. Firstly, the dataset contained many zero entries that had to be dealt with before an analysis could be attempted. We did not develop any new methods for handling zeros, but instead used the transformation in Equation 7.2. Secondly, it contained sample proportions near zero, which we know greatly affects power and type I error rates based on the results in Chapter 5. However, even with sample proportions close to zero, the type I error rate will still be near the nominal error rate due to the large number of observations in each age group. The third issue, which was the main concern, was that the dataset contained positive sample correlations between components that were small in magnitude. The whole point in using a nested Dirichlet distribution is to account for positive correlations between components.

In our next example, we analyze a dataset with positive sample correlations that are large in magnitude. The drawback to using this new dataset is that it is comprised of only thirty-five observations. These thirty-five observations will be further broken down into three groups leaving roughly 10 - 12 observations per group. Even though the positive correlations are large in magnitude, because of the small sample size, they may not be statistically different from zero.

8.1. Metropolitan Jobs Dataset

The dataset analyzed in this chapter counts the number of jobs in twenty different industries in the largest thirty-five metro areas in the U.S. (Ogozaly, 2022). The twenty industries are listed in Figure 8.2 with the abbreviations used on heat map in Figure 8.1 in parentheses. The numbers in the list are used to label the terminal nodes in the tree diagrams throughout this chapter instead of using an abbreviation for the industries. The dataset was transformed from count data to proportions before any analysis was carried out. There were no combinations of metro areas and industries that resulted in counts of zero. Thus, no method of dealing with zeros was required in this case.

Before any analysis was conducted, the sample correlation between components for the full dataset was calculated. The correlation heat map in Figure 8.1 shows that there are many positive correlations larger than 0.3 between the different industry types. Here, abbreviations for each of the twenty industry types were used rather than numbers in Figure 8.2. For example, the correlation between the proportion of jobs in oil and gas extraction and the proportion of jobs in utilities is 0.37. It makes sense that the proportion of jobs in utilities would increase with the proportion of jobs in gas extraction. There is a positive correlation of 0.40 between manufacturing jobs and wholesale trade jobs. If more products are being manufactured, there needs to be more people to trade those products. Because of the large positive correlations between components, it is likely that conclusions using the nested model may be different than the standard model. The next step is to determine an appropriate nesting tree to use with the dataset.

8.2. The Nesting Tree

Feeding the data into the tree finding algorithm produced the tree in Figure 8.3. The numbered terminal nodes correspond to the list in Figure 8.2. The letters are used to

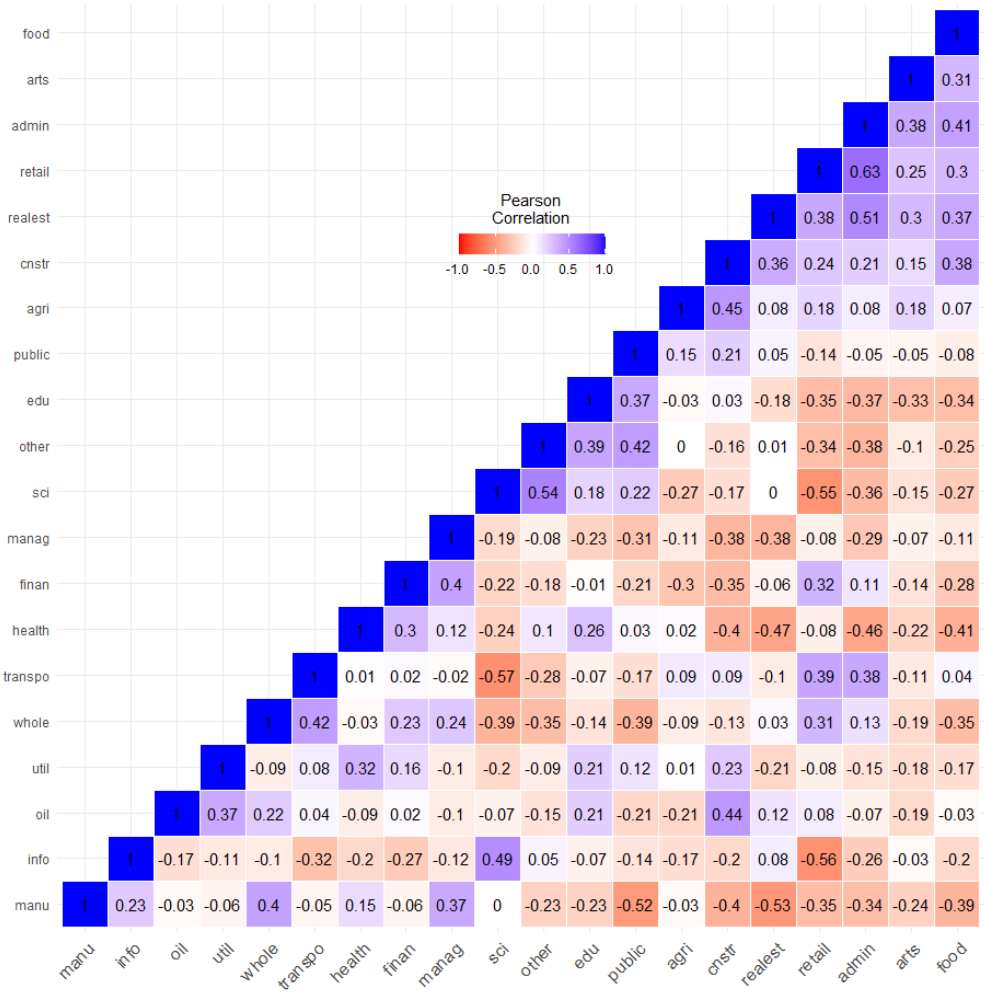


Figure 8.1: A heat map displaying the correlation between components of the job data. The deeper the blue, the closer the correlation is to positive one and the deeper the red the closer the correlation is to negative one. There is evidence of positive correlations that substantially differ from zero.

1. Agriculture, Forestry, Fishing and Hunting (agri)
2. Mining, Quarrying, and Oil and Gas Extraction (oil)
3. Utilities (util)
4. Construction (cnstr)
5. Manufacturing (manu)
6. Wholesale Trade (whole)
7. Retail Trade (retail)
8. Transportation, Warehousing (transpo)
9. Information (info)
10. Finance and Insurance (finan)
11. Real Estate and Rental and Leasing (realest)
12. Professional, Scientific, and Technical Services (sci)
13. Management of Companies and Enterprises (manag)
14. Administration and Support, Waste Management and Remediation (admin)
15. Educational Services (edu)
16. Health Care and Social Assistance (health)
17. Arts, Entertainment, Recreation (arts)
18. Accommodation and Food Services (food)
19. Other Services (other)
20. Public Administration (public)

Figure 8.2: A list of the types of industries in the jobs dataset.

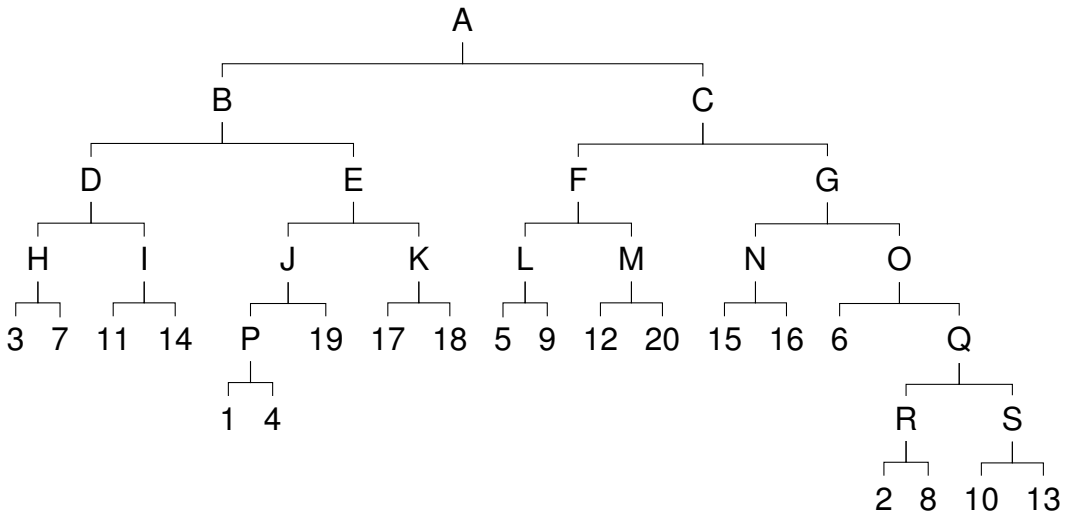


Figure 8.3: The jobs nesting tree. The list of jobs in Figure 8.2 gives a correspondence between the numbers and the types of industries. The letters represent nesting variables.

keep track of the layers. It should be the case that jobs that are positively correlated are nested under the same node and those that are negatively correlated are nested under a different node. Nested under node S are finance and management that have a positive correlation of 0.4. Nested under R are oil and transportation. Under node P is agriculture and construction that have a positive correlation of 0.45. Continuing the ad hoc check that the algorithm is working as expected, note that node 6 (wholesale trade) is on a separate branch than node Q. Either wholesaled trade should be negatively correlated with all the four variables under node Q or positively correlated with the four variables. The correlation between wholesale trade and oil production is 0.22, between wholesale trade and transportation is 0.42, between wholesale trade and finance is 0.23 and between wholesale trade and management is 0.24. This gives evidence that the nesting tree is appropriate for the dataset. After the nesting tree was constructed, 10,000 simulated observations were generated from the nested model. The sample correlation matrix of the simulated data was computed. The absolute differences between the simulated correlations and the real data correlations are shown in the heat map in Figure 8.4. Unlike what we saw with the baseball dataset, there are large discrepancies between the two correlation matrices. For

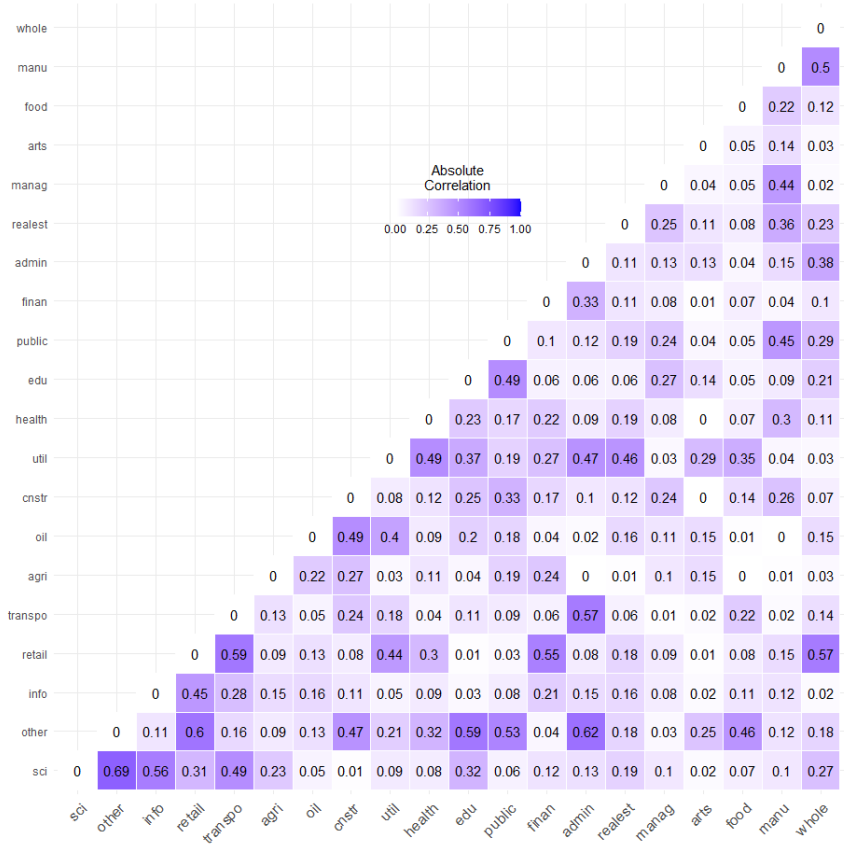


Figure 8.4: Heat map of differences in correlation between the sample data and the simulated data. The deeper the color blue, the larger the difference in the correlations between the sample data and the simulated data.

instance, there are 11 correlation differences that are greater than 0.5. However, there are 190 total correlations to consider; thus, the 11 large differences correspond to less than 6% of all correlations. Only 22% of the correlations are different than those seen in the data by 0.25 or more. This is pretty remarkable considering the tree was constructed from just thirty-five observations. Therefore, we conclude that the tree finding algorithm is doing an adequate job of capturing the correlation structure seen in the real dataset.

8.3. Testing Differences Between Groups

The goal in analyzing this dataset was to check if there are differences in the composition of industry types among the different regions in the United States. Since there are only thirty-five observations, we limited the number of regions to three. These were the East Coast, the West Coast, and Middle. The regions were based on the Petroleum Administration Defense Districts presented in Figure 8.5. Today these districts are used to analyze the movement of petroleum throughout the U.S. (U.S. Energy Information Administration, 2012). In the dataset, only one of the thirty-five cities was located in the Rocky Mountain Region (Denver) and just four were in the Gulf Coast Region (Houston, Dallas, San Antonio, and Austin). After combining the cities in the three middle regions, the East Coast group contained 11 observations, the Middle group had 14 observations, and the West Coast group had 10 observations. It is important for the analysis that each group included at least 10 observations. Based upon our power simulation studies, too small of a sample size can have a large impact on results.

Since there are twenty different components that make up each observation, showing a plot of the 190 ternary diagrams is not helpful. To get some picture of the data, we created a subcomposition with the first five job types as listed in Figure 8.2. Figure 8.6 shows the ternary diagrams for this subset. We should not draw any conclusion from the ternary diagram because only the first five components were used which means a lot of information is lost. The dimensionality in this dataset is too large to gain any sort of information from plots. As we saw with the baseball data, a formal test is needed to come to any conclusion. We will conduct a single layer overall test and a nested overall test and compare the results. We will then attempt to collapse the tree.

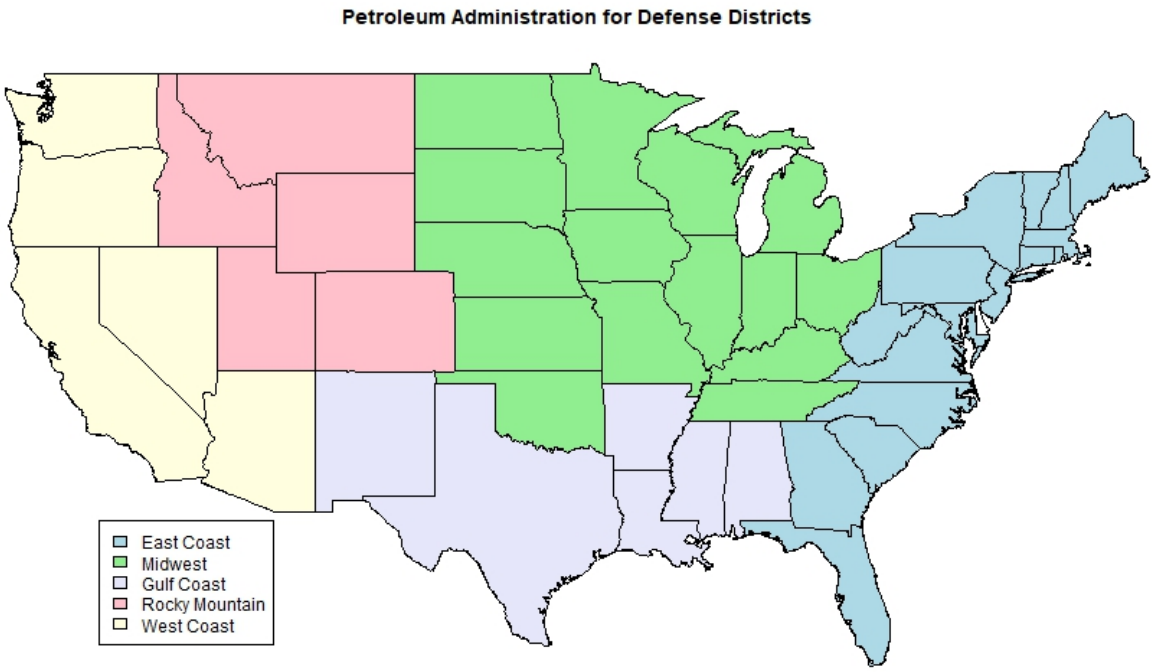


Figure 8.5: The jobs data was broken down into groups by region. The regions include the West Coast, East Coast, and Middle. The Middle group was formed by combining the Rocky Mountain, Midwest, and Gulf Coast Regions.

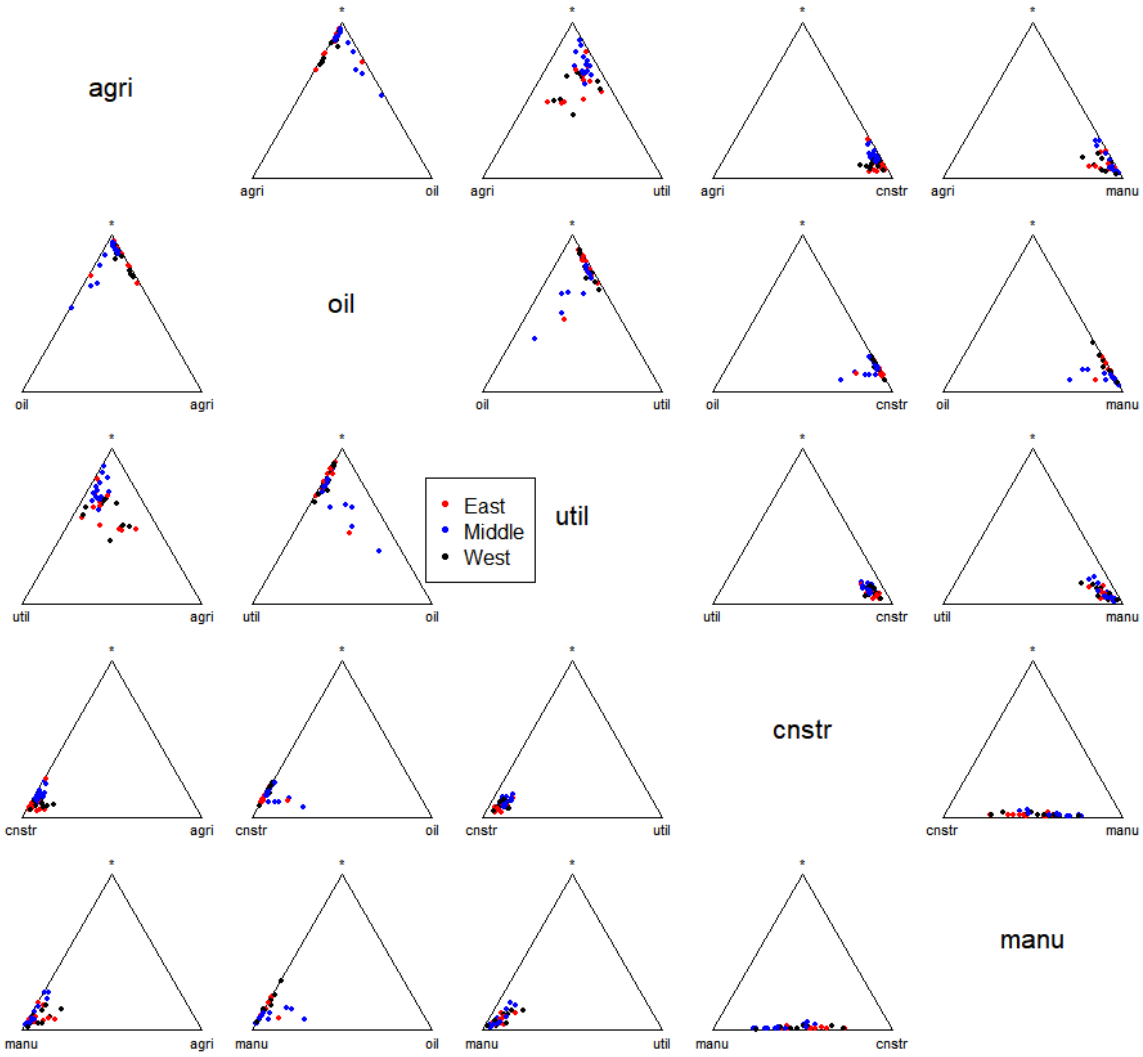


Figure 8.6: This figure displays ternary diagrams for the jobs dataset when only the first five components are taken into consideration. Extreme caution should be taken drawing any conclusion from this subset of data.

8.3.1. Single Layer Overall Test and Pairwise Comparisons

We conducted the hypothesis test in Equation 7.3 using the jobs dataset and a non-nested Dirichlet model. The test statistic for the single layer model is 99.5. The number of free parameters under the alternative hypothesis is 60 and the number of free parameters under the null hypothesis is 22. The overall test statistic follows an approximately χ^2 distribution with 38 degrees of freedom. The p-value associated with the test statistic of 99.5 is $2 \cdot 10^{-7}$. Thus, we conclude that there is strong evidence for a difference in job type composition between regions.

Next we conducted the three pairwise comparisons with null hypotheses given in Equations 7.4, 7.5, 7.6 for the jobs data using the single layer model. For the three tests, we obtained tests statistics of 66.8, 49.6, and 55.1 respectively. The degrees of freedom associated with each of these tests is 19. Thus, the p-values are $3.1 \cdot 10^{-7}, 0.0001$, and $2.0 \cdot 10^{5}$. The p-values are so small that we would still reject the null hypotheses using any multiple comparison adjustment. If we analyze the data using a non-nested Dirichlet model, we come to the conclusion that the three groups have different mean composition vectors and no pair of groups share a mean vector.

8.3.2. Nested Dirichlet Overall Test and Pairwise Comparisons

Using the tree diagram in Figure 8.3 we conducted an overall test for a common mean vector for the three groups. The overall likelihood ratio test statistic at each layer in the nested Dirichlet model is given in Table 8.1. The value of the overall test for the entire tree, summing over the value of the test statistics at each layer is 102.53. The degrees of freedom under the alternative hypothesis is 114. Under the null the degrees of freedom are 76. Thus, the test statistic follows an approximate χ^2 distribution with 38 degrees of freedom. The p-value associated with this test statistic is $7.7 \cdot 10^{-7}$. The largest contribu-

tion to the test statistic comes from the subtree with R as the parent node. This subtree has one branch for mining and gas and the other branch for transportation. We conclude from this test that there are differences among the mean vectors of the three groups.

Now that we have strong evidence for differences between the mean vectors of the three groups, we want to conduct all the possible pairwise comparisons between groups. Once again, we are conducting tests with the null hypotheses in Equations 7.4, 7.5, 7.6, this time with a nested model. The results for the pairwise comparisons at each level in the tree are in 8.1. The p-values are very small, indicating that the mean vector is different for each pair of groups. The comparison between groups 2 and 3 yielded the largest test statistic.

8.3.3. Pruning the Nesting Tree

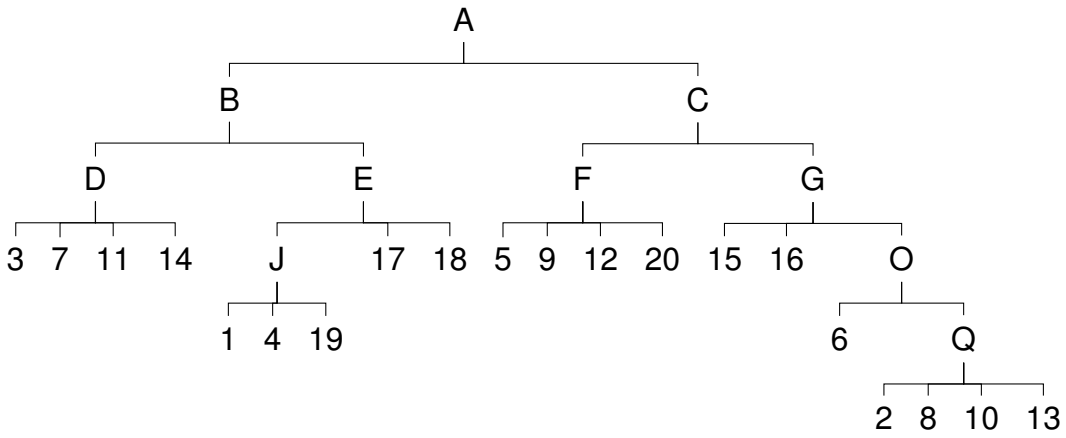


Figure 8.7: The reduced tree after round one of pruning. The nodes H, I, K, L, M, N, P, R, and S have been removed leaving subtrees that are no longer binary.

It is possible that the tree finding algorithm generated a tree that has more bifurcations than necessary. Using the method presented in Chapter 3, we checked whether each node was necessary using a 95% confidence interval for the difference in α parameters. Recall that if this interval contains zero, then the node is not needed and is removed to

Total	102.53	41.71	34.99	75.68
p-value	$7.7 \cdot 10^{-7}$	$1.9 \cdot 10^{-3}$	$1.4 \cdot 10^{-2}$	$1.02 \cdot 10^{-8}$
Node	All Groups	1 VS. 2	2 1 VS. 3	2 VS. 3
A	6.47	2.99	0.15	4.98
B	10.22	0.57	4.89	11
C	2.32	0.03	1.68	2.41
D	3.57	0.09	1.45	3.69
E	0.23	0.21	0.02	0.05
F	8.84	9.39	3.16	0.44
G	8.14	2.21	1.02	7.72
H	5.98	4.83	5.78	1.87
I	1.53	0.58	0.19	1.43
J	4.11	0.09	3.39	3.17
K	2.75	2.82	0.76	0.39
L	8.68	7.98	0.33	4.37
M	1.72	1.06	0.82	1.65
N	0.08	0.01	0.1	0.02
O	2	1.33	1.53	0.06
P	10	2.25	2.09	10.15
Q	2.12	0.5	2.22	0.73
R	21.61	4.69	3.15	20.43
S	2.16	0.08	2.26	1.12

Table 8.1: The test statistics for each layer for the nested Dirichlet model when testing for all three groups and doing pairwise comparisons.

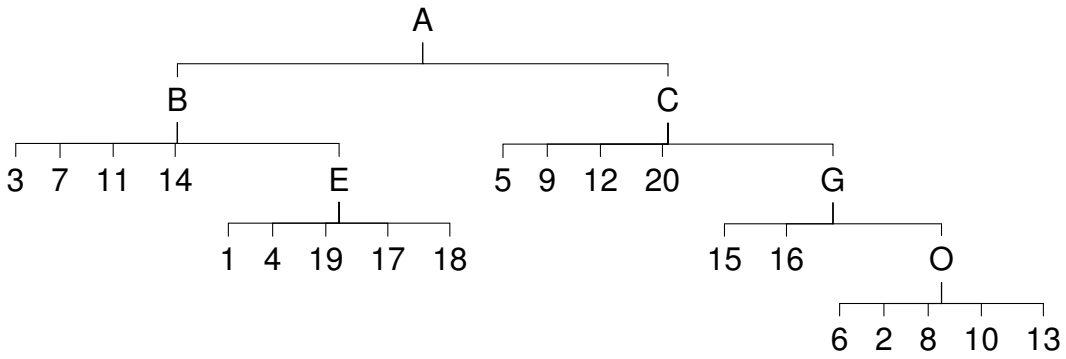


Figure 8.8: The reduced tree after a second round of pruning. The nodes D, J, F, and Q have been removed.

prune the tree. We started at the bottom of the tree and worked our way up. There are nine subtrees that make up the bottom layers of the tree. These are the subtrees with parent nodes H, I, P, K, L, M, N, R, S. In round one of pruning, we constructed the confidence intervals associated with each of these nodes. The complete set of confidence intervals are in Table 8.2. Each of the confidence intervals for nodes, H, I, P, K, L, M, N, R, and S contain zero; these nodes are not needed. Note that these intervals are not adjusted for multiple comparisons. Using a Bonferroni adjustment would increase the confidence level to $(100 - 0.05/19)\% = 99.73\%$. This would further increase the width of the confidence interval, ensuring that 0 is still in each one. The pruned tree after round one is shown in Figure 8.7.

After round one, there are four subtrees now at the bottom of the tree. The immediate parent nodes for the subtrees are D, J, F, and Q After computing the confidence intervals for these nodes, we once again come to the conclusion that all four nodes should be removed. The confidence intervals for all the nodes are in Table 8.2. The tree after round 2 of pruning is in Figure 8.8.

After round two, there are two subtrees that make up the bottom layer of the tree. The immediate parent nodes of the two subtrees are E and O. Sadly, these nodes did not survive the brutal culling and were chopped in this round of pruning, leaving the tree

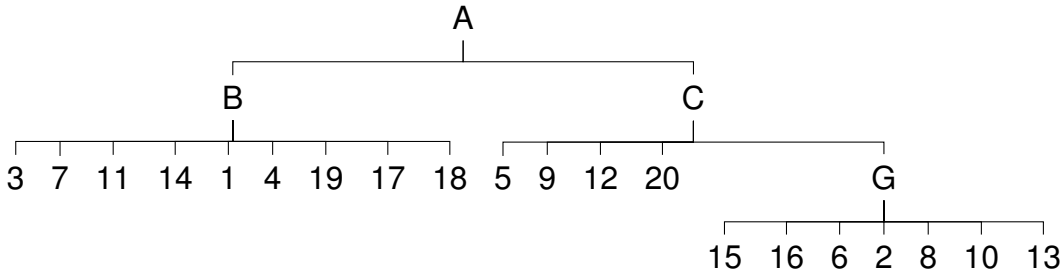


Figure 8.9: The tree after the third round of pruning. E and O have been removed.

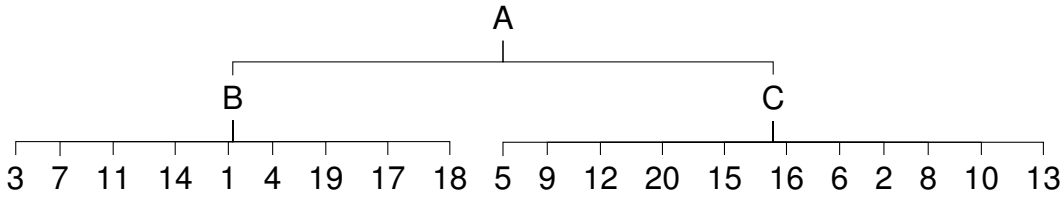


Figure 8.10: Pruning round 4

pictured in Figure 8.9. In the next round of pruning, we found that the subtrees labeled with node G was also not needed and we get the tree in Figure 8.10. This leaves us with just nodes B and C to check. The confidence intervals for nodes B and C both contained zero and these nodes were pruned as well. This leaves us with the single layer tree seen in 8.11.

8.4. Conclusion

It is both fortunate and unfortunate to return there and back again. It is fortunate in that there is no further analysis to do. If the reduced nesting tree had been anything but the

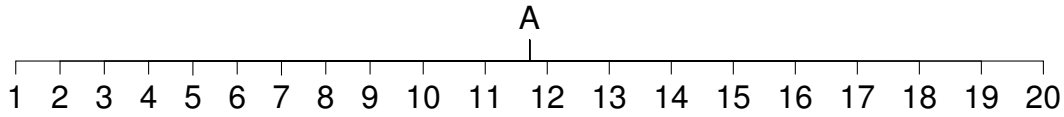


Figure 8.11: Pruning round 5. After the fifth round of pruning, we collapsed the tree all the way back to the single layer tree.

Node	Lower Endpoint	Upper Endpoint
B	-1690.7	1155.8
C	-486.3	384.9
D	-5864.1	5326.1
E	-1831.0	2039.4
F	-79.1	66.2
G	-362.9	228.7
H	-7440.3	7332.6
I	-6741.8	6575.2
J	-357.7	329.1
K	-1072.1	1000.6
L	-28.9	20.3
M	-86.5	60.1
N	-1696.6	1553.2
O	-118.0	136.6
P	-398.8	373.3
Q	-412.5	475.8
R	-34.0	25.0
S	-117.8	86.6

Table 8.2: The unadjusted confidence intervals used to decided the inclusion of each node in the nesting tree. Each interval contains zero which means that the node can be removed and the tree collapsed.

single layer tree, the new nesting tree test would have been used to investigate whether the conclusion using the reduced nesting tree was different than the conclusion reached using the other two models. It is unfortunate as it likely means that the pruning method was too aggressive in this case and the tree was not allowed to grow in a healthy organic way. The aggressiveness of the pruning method can be tempered by reducing the width of the confidence interval using a smaller confidence level. However, it is likely due to our small sample size and large number of components that the full tree recoiled back to the single layer tree. In this case, this is not a surprise since 95% confidence intervals for ρ using $r = 0.65$ and a sample of size 12 contains 0.

No matter which of the two models we used, we reached the same conclusion. The job compositions vary by region and no two regions share a common mean vector.

Chapter 9
Future Work

There are many open questions regarding the topic of nested Dirichlet models. Below are avenues for future research.

One future project is to determine how robust the ordinary Dirichlet model is to positive correlations between variables. By robust, we are referring to the outcomes of hypothesis tests of equal means are correct. Consider the case where one correlation among many negative correlations is positive. The fit of the Dirichlet distribution should be checked against the fit of the nested Dirichlet distribution in order to assess whether the nested Dirichlet is actually necessary. We saw with the baseball dataset that using the ordinary Dirichlet distribution to model the data led to the same conclusion as using a nested Dirichlet distribution. Further investigations would include altering the magnitude of the positive correlation while fitting both the non-nested and nested models until a different conclusion is reached. Another investigation would involve altering the number of positive correlations and repeating the same process.

The tree finding algorithm needs further refinement. First in the list is to find a better way to prune the trees. As we have shown, the current method of building MLE based confidence interval has low power. Another way to approach the problem would be to use Bayesian methods to generate confidence intervals. This may lead to a higher powered test. To my knowledge, no other paper has addressed how to build confidence intervals for differences in parameters in a nesting tree. Another project that has been left untouched is to look at how many observations are needed for the tree algorithm to give good results. The number of observations can be increased from a small starting amount until the

correct nesting tree is generated for both the standard tree finding method and the method with pruning. Along with increasing the number of observations, the number of nodes can be simultaneously increased to examine the relationship between the number of nodes in the tree and the necessary smallest sample size to generate the correct tree. Lastly, precision too could be altered to determine its impact on the necessary smallest sample size.

Power is correlated with the number of parameters estimated. As the number of nodes in a nesting tree increases, the number of estimated parameters increases. A simulation study to examine the impact on power as the number of nodes increases would be enlightening. There is likely a point where it is better, in the sense of a more powerful test, to use an inaccurate tree with fewer nodes than a more complicated tree that reflects reality.

Null (2009) modeled a set of baseball data using a nested Dirichlet distribution. With 14 different outcomes, there were too many possible nesting trees to find the best fitting one. Instead, he arbitrarily created two nesting variables: ground ball and fly ball events. Within this subset of trees, he searched for the best fitting one. The tree finding algorithm presented in Turner (2013) can be applied to Null's dataset to see if the same tree Null published is constructed. Since the tree finding algorithm was used successfully with the jobs dataset, which had 20 components, it seems likely that it could handle a dataset with 14 components. It is also worth investigating how dependent the analysis is on tree structure. For example, if Null's imposed tree structure is changed, the conclusion of the analysis may change along with it. In general, an investigation into how the nesting tree selected affects the results needs to be done.

In human microbiome studies, there are many sample taxa counts that are zero. These are likely not true zeros, but instead represent values that are below the detection limit. In most papers, taxa with zero counts are removed from the dataset. A better way of handling values that are below the detection limit should be explored. If we are modeling the data with a nested Dirichlet distribution, an analysis with an and without these taxa

should be run and the results compared to see how much of an impact deleting these taxa makes.

Lastly, the choice to analyze the baseball data using three independent age groups may not have been the best choice. Since each batter reappears in the dataset over many seasons, it is possible for one batter to be in two or more age groups. This means that these groups are not truly independent. Instead the design can be seen as repeated measures. Other designs besides independent group designs should be developed for nested Dirichlet models.

Bibliography

Aitchison, J. (1982). The statistical analysis of compositional data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 44(2):139–177.

Alenazi, A. (2021). Regression for compositional data with compositional data as predictor variables with or without zero values. *Journal of data science*, 17:219–237.

Andersen, C. R., Wolf, J., Jennings, K., Prough, D. S., and Hawkins, B. E. (2020). Accelerated failure time survival model to analyze Morris water maze latency data. *Journal of Neurotrauma*, 38:435 – 445.

Barnhart, C. D., Yang, D., and Lein, P. J. (2015). Using the morris water maze to assess spatial learning and memory in weanling mice. *PLoS ONE*, 10.

Byrd, R., Lu, P., Nocedal, J., and Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. *SIAM Journal of Scientific Computing*, 16:1190–1208.

Casella, G. and Berger, R. L. (2002). *Statistical Inference*, volume 2. Duxbury Pacific Grove, CA.

Connor, R. J. and Mosimann, J. E. (1969). Concepts of independence for proportions with a generalization of the dirichlet distribution. *Journal of the American Statistical Association*, 64:194–206.

Deacon, R. M. J. and Rawlins, J. N. P. (2006). T-maze alternation in the rodent. *Nature Protocols*, 1:7–12.

Dennis, S. Y. (1991). On the hyper-Dirichlet type I and hyper-Liouville distributions. *Communications in Statistics*, 20(12):4069–4081.

Egozcue, J. J., Pawlowsky-Glahn, V., Mateu-Figueras, G., and Barceló-Vidal, C. (2003). Isometric logratio transformations for compositional data analysis. *Mathematical Geology*, 35:279–300.

Gao, B., Chi, L., Zhu, Y., Shi, X., Tu, P., Li, B., Yin, J., Gao, N., Shen, W., and Schnabl, B. (2021). An introduction to next generation sequencing bioinformatic analysis in gut microbiome studies. *Biomolecules*, 11(4):530.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2003). *Bayesian Data Analysis*. Chapman and Hall, Boca Raton, Florida.

Gioia, V. and Pagui, E. C. K. (2021). Estimation of Dirichlet distribution parameters with bias-reducing adjusted score functions.

Gloor, G. B., Macklaim, J. M., Pawlowsky-Glahn, V., and Egozcue, J. J. (2017). Microbiome datasets are compositional: And this is not optional. *Frontiers in Microbiology*, 8.

Hickey, C., Kelly, S., Carroll, P., and O'connor, J. (2015). Prediction of forestry planned end products using Dirichlet regression and neural networks. *Forest Science*, 61(2):289–297.

Hijazi, R. H. and Jernigan, R. (2009). Modelling compositional data using Dirichlet regression models. *Journal of Applied Probability and Statistics*, 4:77–91.

Hron, K., Filzmoser, P., and Thompson, K. (2012). Linear regression with compositional explanatory variables. *Journal of Applied Statistics*, 39:1115 – 1128.

Human Microbiome Project Consortium (2012). A framework for human microbiome research. *Nature*, 486:215 – 221.

Koslovsky, M. D. and Vannucci, M. (2020). MicroBVS: Dirichlet-tree multinomial regression models with Bayesian variable selection-an R package. *BMC Bioinformatics*, 21(1):301 – 301.

Kraeuter, A. K., Guest, P. C., and Sarnyai, Z. (2019). The y-maze for assessment of spatial working and reference memory in mice. *Methods in molecular biology*, 1916:105–111.

La Rosa, P. S., Brooks, J. P., Deych, E., Boone, E. L., Edwards, D. J., Wang, Q., Sodergren, E., Weinstock, G., and Shannon, W. D. (2012). Hypothesis testing and power calculations for taxonomic-based human microbiome data. *PloS One*, 7(12):e52078–e52078.

La Rosa, P. S., Deych, E., Carter, S., Shands, B., Yang, D., and Shannon, W. D. (2019). *HMP: Hypothesis Testing and Power Calculations for Comparing Metagenomic Samples from HMP*. R package version 2.0.1.

Leising, K. J. and Blaisdell, A. P. (2009). Associative basis of landmark learning and integration in vertebrates. *Comparative cognition & behavior reviews*, 4:80–102.

Maugard, M., Doux, C., and Bonvento, G. (2019). A new statistical method to analyze Morris water maze data using Dirichlet distribution. *F1000 Research*, 38:1601 – .

McDonald, D., Birmingham, A., and Knight, R. (2015). Context and the human microbiome. *Microbiome*, 3(52):52 – 52.

McKinnon, K. (2018). Flow cytometry: An overview. *Current Protocols in Immunology*, 120:5.1.1 – 5.1.11.

Minka, T. (1999). The Dirichlet-tree distribution. https://tminka.github.io/papers/dirichlet/minka-dirtree.pdf. [Online; accessed 17-March-2021].

Minka, T. (2000). Estimating a Dirichlet distribution. https://tminka.github.io/papers/dirichlet/minka-dirichlet.pdf. [Online; accessed 17-March-2021].

Morris, R. G. (1981). Spatial localization does not require the presence of local cues. *Learning and Motivation*, 12(2):239–260.

Morris, R. G. M. (1984). Developments of a water-maze procedure for studying spatial learning in the rat. *Journal of Neuroscience Methods*, 11:47–60.

Ng, K. W., Tian, G.-L., and Tang, M.-L. (2011). *Dirichlet and Related Distributions Theory Methods and Applications*. John Wiley and Sons, London, England.

Null, B. (2008). The nested Dirichlet distribution: Properties and applications. Contact the author for a copy.

Null, B. (2009). Modeling baseball player ability with a nested Dirichlet distribution. *Journal of Quantitative Analysis in Sports*, 5(2):5–10.

Ogozaly, W. (2022). Jobs by industry, top 35 us metro areas. https://www.kaggle.com/datasets/walterogozaly/jobs-by-industry-top-35-us-metro-areas. [Online; accessed 18-September-2022].

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Reynolds, N. C., Zhong, J. Y., Clendinen, C. A., Moffat, S. D., and Magnusson, K. R. (2019). Age-related differences in brain activations during spatial memory formation in a well-learned virtual Morris water maze (vmwm) task. *NeuroImage*, 202.

Robitzsch, A. (2020). *sirt: Supplementary Item Response Theory Models*. R package version 3.9-4.

Simmons, J. P., Nelson, L. D., and Simonsohn, U. (2011). Anything as significant false-positive psychology : Undisclosed flexibility in data collection and analysis allows presenting.

Soares, P., Tomé, M., Skovsgaard, J. P., and Vanclay, J. K. (1995). Evaluating a growth model for forest management using continuous forest inventory data. *Forest Ecology and Management*, 71:251–265.

Stewart, C., Iverson, S., and Field, C. (2014). Testing for a difference in diet using fatty acid signatures. *Environmental and Ecological Statistics*, 21:775–792.

Tang, Y., Ma, L., and Niclolae, D. L. (2017). A phylogenetic scan test on Dirichlet-tree multinomial model for microbiome data.

Tian, H., Ding, N., Guo, M., Wang, S., Wang, Z.-D., Liu, H., Yang, J., Li, Y., Ren, J., Jiang, J., and Li, Z. (2019). Analysis of learning and memory ability in an Alzheimer's disease mouse model using the Morris water maze. *Journal of visualized experiments : JoVE*, 152.

Turner, J. (2013). *A novel approach to modeling immunology data derived from flow cytometry*. PhD thesis, Southern Methodist University.

U.S. Energy Information Administration (2012). Padd regions enable regional analysis of petroleum product supply and movements. https://www.eia.gov/todayinenergy/detail.php?id=4890#. Accessed: 2022-09-05.

van den Boogaart, K. G. and Tolosana-Delgado, R. (2013). *Analyzing Compositional Data with R*. Springer, New York, NY.

van den Boogaart, K. G., Tolosana-Delgado, R., and Bren, M. (2021). *compositions: Compositional Data Analysis*. R package version 2.0-1.

Vorhees, C. V. and Williams, M. . (2006). Morris water maze: procedures for assessing spatial and related forms of learning and memory. *Nature Protocols*, 1:848–858.

Vouros, A., Gehring, T. V., Szydlowska, K., Janusz, A., Tu, Z., Croucher, M., Lukasiuk, K., Konopka, W., Sandi, C., and Vasilaki, E. (2018). A generalised framework for detailed classification of swimming paths inside the Morris water maze. *Scientific Reports*, 8(1).

Wadsworth, W. D., Argiento, R., Guindani, M., Galloway-Peña, J. R., Shelburne, S. A., and Vannucci, M. (2017). An integrative Bayesian Dirichlet-multinomial regression model for the analysis of taxonomic abundances in microbiome data. *BMC Bioinformatics*, 18.

Wang, H., Shangguan, L., Wu, J., and Guan, R. (2013). Multiple linear regression modeling for compositional data. *Neurocomputing*, 122:490–500.

Wang, T. and Zhao, H. (2017). A Dirichlet-tree multinomial regression model for associating dietary nutrients with gut microorganisms. *Biometrics*, 73:792–801.

Warnes, G. R., Bolker, B., and Lumley, T. (2021). *gtools: Various R Programming Tools*. R package version 3.9.2.

Yang, D., Johnson, J., Zhou, X., Deych, E., Shands, B., Hanson, B., Sondergren, E., Weinstock, G., and Shannon, W. (2019). New statistical method identifies cytokines that distinguish stool microbiomes. *Scientific Reports*, 9:20082–11.

Zhang, Q. and Dao, T. (2020). A distance based multisample test for high-dimensional compositional data with applications to the human microbiome. *BMC Bioinformatics*, 21:205–205.

Zhong, J., Magnusson, K., Swarts, M., Clendinen, C., Reynolds, N., and Moffat, S. (2017). The application of a rodent-based Morris water maze (MWM) protocol to an investigation of age-related differences in human spatial learning. *Behavioral Neuroscience*, 131:470–482.