CORRELATION-BASED HIERARCHICAL MULTIPLE TESTING PROCEDURES

by

Priscilla Serwaa Bacino

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

of

Doctor of Philosophy

in

Statistics

MONTANA STATE UNIVERSITY
Bozeman, Montana

December 2023

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Mark Greenwood, for his continuous support, patience, and professional advice throughout my Ph.D. journey. His guidance has helped me grow as a professional, and his mentorship has been instrumental in shaping my career. It has been an immense privilege to conduct research under his supervision.

Secondly, I extend my thanks to my doctoral committee members: Dr. Shinjini Nandi, Dr. Andrew Hoegh, Dr. John Borkowski, and Dr. Ron June. Their expertise, insightful thoughts, and constructive feedback have been invaluable in refining my research and bringing this dissertation to fruition. I also wish to express gratitude to the members of the Mathematical Sciences department, especially Katie Sutich, for their consistent kindness and assistance.

I owe a deep debt of gratitude to my family and friends, especially my dad, Seth. Their unwavering love, encouragement, and understanding throughout the highs and lows of this rigorous journey have been a beacon of strength and inspiration.

Lastly, I must thank my loving husband, John, who has been a steadfast pillar of support throughout this process. His patience, love, and unwavering belief in me have been my anchor. I could not have completed this dissertation without him.

TABLE OF CONTENTS

TABLE OF CONTENTS – CONTINUED

TABLE OF CONTENTS – CONTINUED

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES – CONTINUED

LIST OF FIGURES – CONTINUED

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF ALGORITHMS

ABSTRACT

This dissertation explores multiple testing scenarios commonly encountered in biological sciences and elsewhere, where a large number of null hypotheses need to be tested, but typically, only a small fraction of them are actually false. In such situations, it becomes crucial to not only control the overall Type I error rate, such as the family-wise error rate, but also to maintain a sufficient level of statistical power to detect true signals. Traditional methods used for multiple testing adjustment in this context are often overly conservative, resulting in very few true detections, if any. Methods like the Bonferroni procedure exhibit this conservatism due to their assumption that all hypotheses are true nulls, which in practice, cannot be ruled out beforehand. To enhance statistical power, additional assumptions, domain-specific knowledge, or data-driven information are necessary. This work is motivated by applications in untargeted metabolomics, such as in the analysis of metabolite concentrations or peak intensities across groups of subjects, where metabolites share information that can typically be represented by their correlation coefficients. We propose generating a hierarchical structure based on correlation-based dissimilarities, to reflect this interdependence. This hierarchical structure allows us to organize hypotheses and identify the relationships among these hypotheses. Consequently, not all combinations of null hypotheses are valid within the hierarchical structure. These logical constraints can be leveraged when adjusting evidence obtained from the hypothesis tests to enhance the statistical power of multiple testing methods. Furthermore, visualizing the hierarchical structure can aid in understanding the dependency relationships among the hypotheses, facilitating result interpretation. In this dissertation, we present correlation-based hierarchical methods for controlling the family-wise error rate when dealing with correlated outcome variables. Within this framework, we propose methodologies to increase power when comparing means of outcome variables across different groups or treatments and conducting associated follow-up tests. These methodologies have been implemented in an R package to facilitate their practical application.

CHAPTER ONE

INTRODUCTION

Modern experiments in biological sciences and elsewhere often collect responses on subjects for many variables at once with the goal of investigating differences across interventions or groups of interest for each variable. In a typical example, one may be interested in assessing which genetic markers are associated with particular phenotype or risk of disease (Feldmann et al., 2021; Sesia et al., 2020). Similarly, one may be interested in which metabolite abundances are different across experimental treatments (Fortenbach et al., 2023; Jorge-Smeding et al., 2022; Veeramohan et al., 2023). In these and many similar settings, multiple testing is present.

Multiple testing refers to testing more than one hypothesis at once; if it is carried out without adjusting evidence from the tests, it can lead to an inflation of the Type I or false positive error rate. There are many existing methods that can be applied to control different types of error rates across a family or group of tests. Examples under the family-wise error rate (FWER) control framework include the Bonferroni (Dunn, 1961), the Bonferroni-Holm (Holm, 1979) and the Westfall-Young procedures (Thomas et al., 1994). These methods provide adjustments that can be applied to p-values so that the FWER, which is the probability of observing at least one false discovery across multiple tests, is bounded by a predetermined value. Traditional FWER methods, especially the Bonferroni, tend to be conservative as the number of hypothesis tests increases, leading to a smaller likelihood of a true detection, or lower statistical power. This is because methods like the Bonferroni assume the worst case scenario; that every hypothesis being tested is indeed true.

One way to improve the power of existing multiple testing procedures is to incorporate

structural information (Goeman & Finos, 2012; Hu et al., 2010; Meinshausen, 2008; Sankaran & Holmes, 2014). In many large-scale multiple testing situations, hypotheses possess a natural structure that can leveraged. The structure is often inherited from the composition of the underlying data and manifests in forms like groups, hierarchies, or networks. Even in cases where structure is present but unknown or unclear, it can be deduced from the data using methods like clustering. The structure provides additional information that can be employed to order hypotheses, prioritize hypotheses, or impose logical constraints, and reveal which combinations of true and false nulls are not possible. This information can be used to reduce the adjustment applied to the p-values while still maintaining the overall error rate at the predetermined value. The structure can also identify which hypotheses are related, thereby, driving a better narrative to conclusions and results of the analysis.

This dissertation highlights the advantages of incorporating structure into existing multiple testing procedures. In particular, it centers on leveraging hierarchical structures derived from performing hierarchical clustering on the response variables. This hierarchical framework dictates an order for conducting hypothesis tests at different resolutions of the data, termed hierarchy nodes. By adopting this structured approach, constraints are introduced that allow for a reduction in multiple testing adjustments at specific hierarchy nodes. As a result, there is enhanced statistical power while maintaining control over the family-wise error rate. The introduction section begins by setting forth commonly used notation throughout the chapters of this dissertation. It then delves into a fundamental overview of multiple testing, explores relevant literature associated with the hierarchical multiple testing procedures proposed in this work, and concludes with an outline of the dissertation.

1.1 Notation Used Throughout the Dissertation

Let y_{ijq} represent the ith observation, where $i = 1, \ldots, n$, from the jth group, with $j = 1, \ldots, J$, for the variable Y_q, where $q = 1, \ldots, Q$. Here, Q represents the total number

of response variables, n is the sample size, and J is the total number of groups, treatments, or interventions. The true mean response of variable Y_q in group j is denoted by μ_{jq}. The correlation coefficient between two response variables, Y_q and $Y_{q'}$, is represented as $r(Y_q, Y_{q'})$, and the resulting dissimilarity is $d(Y_q, Y_{q'})$.

Let C_k, for $k = 1, \ldots, K$ symbolize the cluster of variables formed at the kth step of an agglomerative hierarchical clustering algorithm. Any two clusters are either mutually exclusive or one is a subset of the other. The sequence C_k adheres to a depth-first search approach, commencing at the root (top) node. Hence, C_1 corresponds to the root node, with sequential numbering down the left-most branch to the terminal node. This sequence then shifts to the closest adjacent node and repeats. Refer to Figure 1.1 for an illustration.

The notations $\mathcal{H}_q$ and $\bar{\mathcal{H}}_q$ pertain to elementary or individual hypotheses, where $q = 1, \ldots, Q$, with each response variable having a corresponding null and alternative hypothesis. Associated p-values and test statistics are represented as p_q and t_q respectively. When ordered, they will be denoted as $p_{(1)}, \ldots, p_{(Q)}$ and $t_{(1)}, \ldots, t_{(Q)}$. The symbol H_k, for $k = 1, \ldots, K$, denotes the kth null hypothesis from the hierarchical family of hypotheses subjected to simultaneous testing, with $\bar{H}_k$ being its alternative counterpart. K is the sum total of null hypotheses in the hierarchy. In essence, H_k is the intersection of the elementary hypotheses tested at a node, for every Y_q within cluster C_k at that node i.e., $H_k = \bigcap_{Y_q \in C_k} \mathcal{H}_q$. Test statistics and p-values for these hypotheses are denoted by $\tau_1, \ldots, \tau_K$ and $\pi_1, \ldots, \pi_K$. When ordered, they are represented as $\tau_{(k)}$ and $\pi_{(k)}$, for the kth ordered test statistic and p-value.

The hierarchy of clusters $C_1, \ldots, C_K$ and the associated null hypotheses $H_1, \ldots, H_k$ is symbolized by $\mathfrak{T}$. Every node in the hierarchy contains a single null hypothesis. However, at each terminal node, where multiple follow-up tests such as pairwise comparisons of groups might be performed, a second of tier of hypotheses are introduced. In such scenarios, H_k is the primary tier null hypothesis at terminal node k, that may be associated with the testing of inter-group differences, while $H_{k,m}$, where $m = 1, \ldots, \binom{J}{2}$ and $J \geq 3$, symbolizes

the follow-up tier null hypotheses assessing pairwise differences.

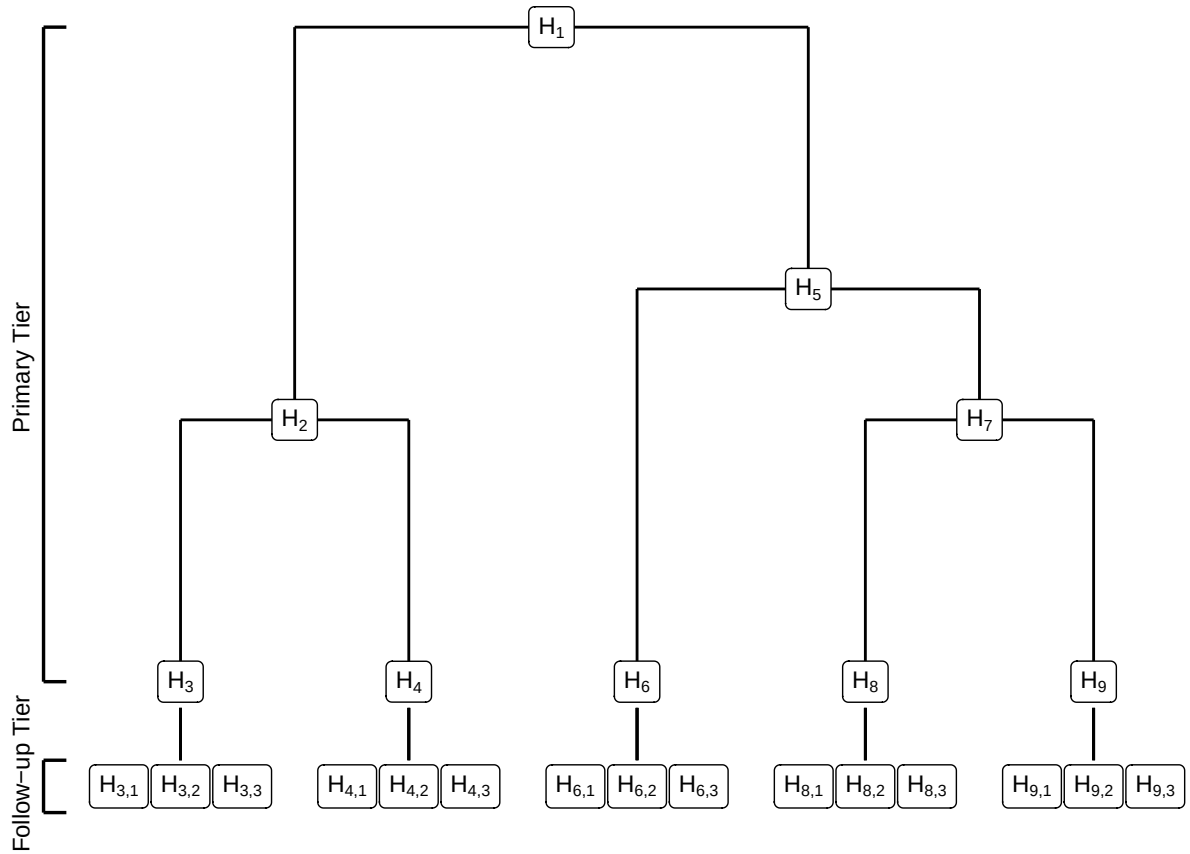


Figure 1.1: An example of the hierarchical structure showing the primary and follow-up tiers.

1.2 Hypothesis Testing

Hypothesis testing is a fundamental statistical technique for making inferences and conclusions about populations of interest in many areas of science. To answer a question about an unknown parameter, a researcher will often establish two competing hypotheses based on the research question: the null hypothesis denoted as H, and the alternative hypothesis denoted as $\bar{H}$. Typically, the null hypothesis is a statement that represents the conventional belief or status quo, such as H : there is no difference in the true mean of choline intensities between sick and healthy mice. The alternative hypothesis, on the other hand,

often represents the complement of the null hypothesis and may be the statement that the assumed status quo is false, which is what the researcher believes or hopes to be true. For example, in the above scenario, the alternative hypothesis could be $\bar{H}$: there is a difference in the true mean of choline intensities between sick and healthy mice.

Once the null and alternative hypotheses are formulated, a suitable test is selected to analyze the collected sample data. The test generates a test statistic and a p-value, which quantify the strength of evidence against the null hypothesis. Traditionally, a predetermined significance level is chosen, and if the p-value is below this level, the null hypothesis is rejected in favor of the alternative hypothesis. Conversely, if the p-value is not below the significance level, we fail to reject the null hypothesis and conclude that the observed data does not provide sufficient evidence to support the alternative hypothesis.

In light of the growing emphasis on research reproducibility, replicability, and the avoidance of p-value pitfalls such as p-hacking, researchers have undergone a shift in their approach to interpreting p-values, moving away from rigid decision cutoffs. This perspective has been emphasized by the American Statistical Association (ASA), stating as part of their six principles to address the misuse of p-values, the need to go beyond relying solely on p-value thresholds for drawing scientific conclusions (Wasserstein & Lazar, 2016).

Researchers now advocate for a more nuanced approach that involves interpreting p-values with graded evidence. Instead of categorizing results as either significant or non-significant based on a specific threshold, this approach acknowledges a continuum of evidence, ranging from no or little evidence to weak to strong, as the p-value progresses from 0 to 1. By considering this spectrum of evidence, researchers can achieve a more comprehensive evaluation of p-values. It also highlights the importance of considering other factors such as effect sizes, confidence intervals, as well as practical implications in the interpretation of results. This shift in perspective reflects a broader understanding that statistical significance alone is insufficient for making substantive claims about research findings.

However, there are still certain situations in statistical analysis and inference where the significance level remains important. For instance, when considering power analysis, sample size estimation, and multiple testing, the significance level plays a crucial role. These aspects of statistical analysis require a clear definition of the level of significance to determine the desired balance between Type I and Type II errors, to estimate the appropriate sample size, and to control the overall error rate. Thus, while the interpretation of p-values is evolving, the significance level continues to be a fundamental component in these specific areas of statistical investigation.

In this dissertation, we adopt a flexible approach, considering both the evolving interpretation of p-values and the importance of the significance level in certain contexts. For example, we specify a significance level to assess family-wise error rate control and power performance of the proposed methods. We also utilize decision thresholds in the proposed methods to determine whether to proceed further down the hierarchy when testing hierarchical families of hypotheses. This approach allows us to avoid testing a large number of hypotheses by focusing only on relevant nodes. However, we also provide the option to examine and evaluate all p-values at each node of the hierarchy for those interested in a more comprehensive analysis.

1.2.1 Errors in Hypothesis Testing

Two types of errors can occur when testing hypotheses. The Type I error, also known as a false positive or a false discovery, happens when the null hypothesis is rejected even though it is true. In other words, the test incorrectly concludes that there is a "significant effect" or "detectable difference" when in reality, one does not exist. The supremum of the probability of committing a Type I error is denoted as α and is typically set as the predetermined significance level. On the other hand, the Type II error, also known as a false negative, occurs when we fail to reject the null hypothesis even though it is false. The

probability of committing a Type II error is often denoted as β.

Avoiding these errors when testing hypotheses is indeed challenging due to the inherent uncertainty associated with sample data. Even with random sampling, there is a possibility that the collected sample data may not accurately reflect the true population and may fail to reveal a detectable difference, even if one truly exists in the population. As a result, researchers strive to minimize one or both types of errors, although achieving this balance can be difficult. The relationship between Type I and Type II errors involves a trade-off. Increasing the Type I error rate typically leads to a decrease in the Type II error rate for a given sample size and effect size. This trade-off occurs because by rejecting the null hypothesis more frequently, researchers are less likely to make the type of decision that results in a Type II error and vice versa.

To strike a balance between Type I and Type II errors, researchers commonly set a threshold for α, and employ statistical methods that control the Type I error rate at α while maximizing statistical power. Statistical power, represented as $1 - \beta$, refers to the probability of correctly rejecting a false null hypothesis. Throughout this work, the proposed methods will implement this strategy by specifying an α value and evaluating performance in terms of controlling α at this level while maximizing power.

Table 1.1: Type I and II errors

Decision	Null hypothesis is true	Null hypothesis is false
Reject null hypothesis	Type I error	Correct decision
Fail to reject null hypothesis	Correct decision	Type II error

1.3 Multiple Testing

In many data-rich applications, there is often a need to conduct numerous hypothesis tests simultaneously, often involving hundreds, thousands, or even millions of tests. When dealing with multiple hypothesis tests, it is important to consider an overall measure of the Type I error, as each individual test carries its own risk of committing a Type I error. In this subsection, we discuss one widely used measure of the accumulating risk called the family-wise error rate (FWER). Additionally, we describe multiple testing procedures commonly employed for controlling the family-wise error rate, including the Bonferroni that forms the basis of the proposed methods discussed in Chapters 3 and 4 of the dissertation. Lastly, we provide an overview of logical constraints hierarchical multiple testing procedures, which align with the research paradigm of this study.

1.3.1 Family-Wise Error Rate

The family-wise error rate (FWER) is the probability of making at least one false discovery when testing a family or a group of hypothesis tests. Specifically, when considering a family of K independent null hypotheses, each tested at α, the FWER is given by $1-(1-\alpha)^K$. As shown in Figure 1.2, as the number of tests grows, the FWER increases exponentially towards 1, making it challenging to maintain the desired level of α. This issue is known as the multiple testing problem or the problem of multiplicity.

To address the multiple testing problem and control the FWER, various adjustment procedures have been proposed in the literature. These procedures involve either modifying the individual prespecified α for each test or adjusting the evidence (p-value) from each test to maintain the desired FWER. Examples include the Bonferroni (Dunn, 1961), the Bonferroni-Holm (Holm, 1979) and the Westfall-Young procedures (Thomas et al., 1994).

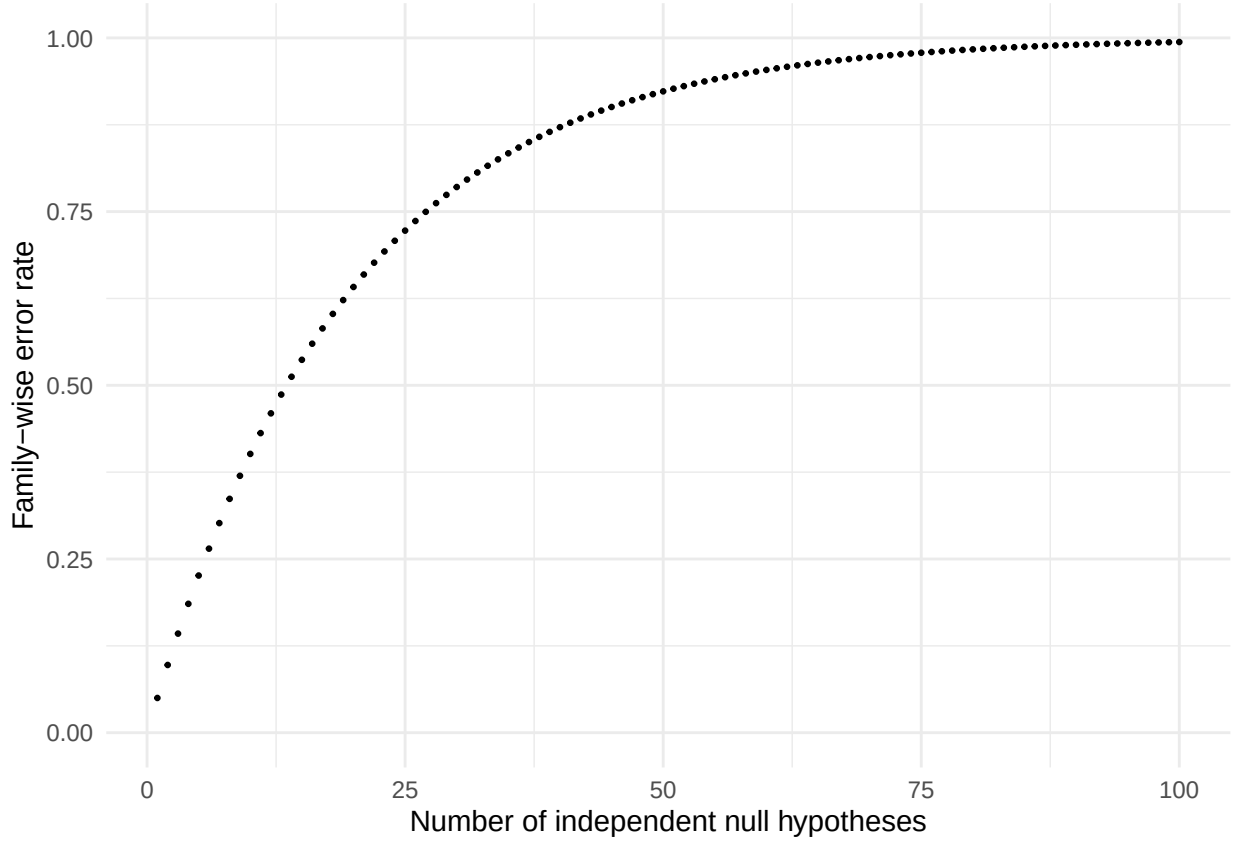


Figure 1.2: A plot of family-wise error rates against the number of hypotheses being tested simultaneously when $\alpha = 0.05$.

1.3.2 Multiple Testing Procedures

The Bonferroni procedure is perhaps the simplest multiple testing procedure in literature. To control multiplicity using this procedure for a family of tests, the α value for each test is divided by the number of total tests, K, or the p-value from each test is multiplied by K, *i.e.*, $\min(K\pi_k, 1)$ where π_k represents p-value obtained from testing null hypothesis H_k, $k = 1, \ldots, K$. The Bonferroni procedure guarantees strong control over the FWER. This means that regardless of which null hypotheses are true, the FWER is bounded by α when using the Bonferroni procedure. The procedure assumes independence between tests and can be overly conservative which may result in low statistical power especially for a large

number of tests or when the tests are positively correlated. Consider a family of 100 null hypotheses and an $\alpha = 0.05$. Applying the Bonferroni adjustment will result in an adjusted p-value of $\min(100\pi_k, 1)$ for $k = 1, \ldots, K$. In this case, an unadjusted p-value of 0.01, which would typically be considered strong evidence against the null hypothesis, is now adjusted to 1 indicating no evidence. Maintaining a similar level of statistical power when using the Bonferroni procedure would often necessitate a dramatic increase in sample size or only extremely large effect sizes will be detected. However, such increases in sample size can pose economic and logistic challenges, making it unfeasible or impractical in certain cases.

Holm's procedure (Holm, 1979), often referred to as the step-down Bonferroni method, offers a sequential approach to multiple testing adjustment. This procedure works by first sorting the p-values in ascending order. The smallest p-value is then multiplied by the total number of tests, K. The next smallest p-value is multiplied by $(K - 1)$, and this process continues until a p-value exceeds the adjusted threshold, with all subsequent p-values being deemed "non-significant" without further adjustment. Mathematically, for a sorted p-value $\pi_{(k)}$, the adjusted value is $\min(\pi_{(k)}(K - k + 1), 1)$ for $k = 1, \ldots, K$. Unlike the Bonferroni procedure, Holm's method increases power by leveraging the ordering of p-values. This means that if some tests yield very small p-values, other tests have a higher chance of being declared significant without being overly penalized. While it still offers strong control over the FWER, it does so in a less conservative manner than the Bonferroni procedure, especially when there are a few very small p-values or a large number of tests.

The Westfall-Young procedure (Thomas et al., 1994) aims to control the FWER while accounting for the correlation structure among tests, making it particularly suitable for multiple testing scenarios with dependent tests. At its core, the Westfall-Young procedure employs a permutation-based approach. For each permutation, p-values are computed, and the smallest among these permuted p-values is taken. The observed p-values are then compared to the distribution of these minimal permuted p-values to get adjusted values.

One of the primary advantages of the Westfall-Young procedure is that it provides a less conservative adjustment in the presence of positive dependence among tests. This means that in situations where tests are positively correlated, the Westfall-Young procedure can achieve higher power compared to methods that do not account for this correlation. However, the computational burden can be higher due to the permutation-based approach, especially with a large number of tests.

1.3.3 Logical Constraints in Multiple Testing

In many multiple testing scenarios, the hypotheses under consideration are often related, resulting in outcomes that are not mutually independent of one another. An example pointed out by Shaffer (1986) involves the testing three null hypotheses, H_1, H_2, and H_3. Each assesses the pairwise equality of three population means: $\mu_j, \mu_{j'}$, and $\mu_{j''}$, where $j \neq j' \neq j''$. Thus, we have $H_1 : \mu_j = \mu_{j'}, H_2 : \mu_j = \mu_{j''},$ and $H_3 : \mu_{j'} = \mu_{j''}$. The outcome of one null hypothesis imposes constraints on the outcomes of the others. Specifically, if one null hypothesis is indeed false, then at least one other must also be false. For example, if the null parameter domains of μ_j and $\mu_{j'}$ do not overlap, then the null domain of $\mu_{j''}$ cannot align with the other two simultaneously. Essentially, the null parameter domain of $\mu_{j''}$ must differ from at least one, thereby influencing the outcome of the associated hypotheses.

Similarly, situations where the null parameter domain of one hypothesis is encompassed by another can be considered. For instance, for the null hypotheses $H_1 : \mu_j = \mu_{j'}$ and $H_4 : \mu_j = \mu_{j'} = \mu_{j''}$, the null parameter values of H_4 are a subset of those for H_1. As a result, if H_4 is true, it implies that the values of μ_j and $\mu_{j''}$ also satisfy H_1, and thus H_1 must also be true. Figure 1.3 illustrates this relationship where the grey plane represents the domain where $H_1 : \mu_j = \mu_{j'}$ is true for values between 0 and 10, and the black dots symbolize the parameter values satisfying $H_4 : \mu_j = \mu_{j'} = \mu_{j''}$ within the same range. This is the situation encountered in a one-way analysis of variance setting with three groups, and potential interest

in comparing all pairs of groups.

Although there are many nuanced and specific relationships that can impose restrictions on hypotheses, this dissertation will focus primarily on the two constraints discussed here. These foundational concepts form the basis for the multiple testing procedures explored in this work.

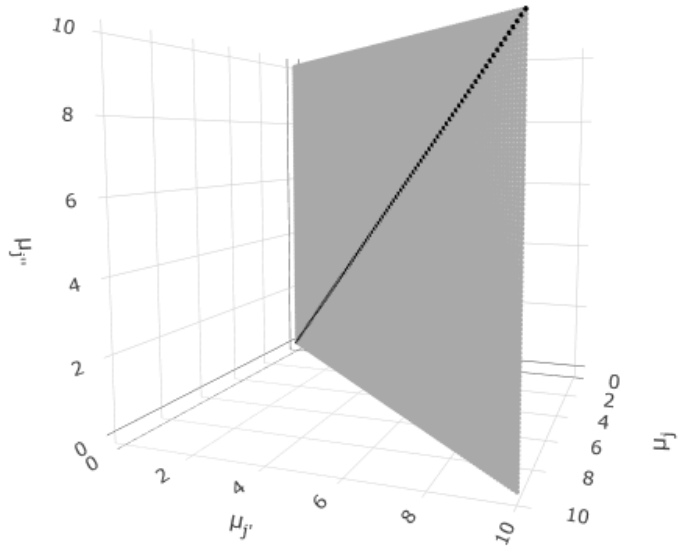


Figure 1.3: 3-D scatterplot visualizing the parameter space of two null hypotheses, H_1 and H_4. The grey plane represents the domain where $H_1 : \mu_j = \mu_{j'}$ is true for values between 0 and 10. The black dots on the other hand, symbolize the paramater values satisfying $H_4 : \mu_j = \mu_{j'} = \mu_{j''}$ within the same range. The validity of H_4 restricts the parameter domain of H_1, underscoring that if H_4 is true, then H_1 is true.

1.3.4 Hierarchical Multiple Testing Procedures

Consider a family of null hypotheses $H_1, H_2, \ldots, H_K$. $H_k : \theta_k \in \Omega_k$, where θ_k is an unknown parameter vector and Ω_k is its null parameter space. The null hypothesis H_k

logically implies $H_{k'}$ if and only if $\Omega_k \subseteq \Omega_{k'}$ and $H_1, H_2, \ldots, H_K$ is said to be a hierarchical family if there is an implication relationship between at least one pair of hypotheses, H_k and $H_{k'}$, $k, k' = 1, 2, \ldots, K$ within the family. An implication relationship between hypotheses in this context means that if H_k implies $H_{k'}$ and H_k is true, then $H_{k'}$ is also true. We refer to any multiple testing procedure that is suitable for testing families of this form as a hierarchical multiple testing procedure (HMTP).

In this work, we will only consider hierarchical families for which any pair of hypotheses fall into one of two categories: disjoint, where they do not share a parameter space, or nested, where one hypothesis implies the other. These hierarchical families can be visually represented using a tree-based structure, where the root hypothesis is positioned at the top and serves as an implication for all other hypotheses. An ancestor of a hypothesis is a hypothesis that implies it in the tree. For instance in Figure 1.1 , H_1 is the root hypothesis and it implies H_2 through H_9, making it an ancestor to all the other hypotheses. Conversely, H_2 to H_9 are descendants of H_1. The immediate ancestor of a hypothesis is called its parent, and any disjoint hypotheses that share a parent are siblings. In Figure 1.1, H_3 and H_4 are examples of siblings who have H_2 as their parent.

The HMTP testing strategy starts by evaluating the root hypothesis and then proceeds to consider its children in the hierarchical tree structure, but only if there is strong evidence against the root hypothesis. The procedure continues down the branches of the tree, testing each hypothesis, but only when there is substantial evidence against its parent hypothesis. In the remainder of this subsection, we will explore several HMTPs introduced in the literature that adhere to this general testing strategy.

Meinshausen (2008) introduced a hierarchical procedure based on the Bonferroni correction for selecting important variables in linear and generalized linear regression. This method offers an alternative approach to testing each variable individually, which is particularly advantageous when predictor variables are highly correlated. Testing variables

individually can lead to low statistical power, especially when multiple predictors are tested simultaneously. The hierarchical procedure addresses this issue by testing variables in clusters and identifying the set of predictors as the smallest clusters with detectable importance to the response. Meinshausen demonstrates that the method strongly controls the family-wise error rate (FWER) and proposes a Shaffer improvement (Shaffer, 1986) to enhance its power. Notably, when focusing on the terminal node, the method performs similarly to the Bonferroni-Holm procedure. However, this is largely influenced by the specific tests used to evaluate the importance of variable clusters, as well as the sample size of the data.

Meinshausen's procedure is not suited for high-dimensional scenarios, where the number of predictors exceeds the available sample size due to the the selected tests used to evaluate variable clusters. To address this limitation, Mandozzi & Bühlmann (2016b) proposed an extension specifically designed for high-dimensional datasets. Their approach involves multiple sample splitting, where the data is repeatedly divided into halves. In the first half, a Lasso model is fitted to identify the "active" set of predictors, which is then intersected with the cluster at each node to obtain a final set of screened predictors. The second half of the data is used to test the importance of these screened predictors, and the resulting p-values are aggregated over all iterations using empirical quantiles. Although the multiple sample splitting method used in this approach has sub-optimal power performance compared to similar methods, overall, the method effectively controls the family-wise error rate and outperforms individual variable testing.

The inheritance procedure, proposed by Goeman & Finos (2012), serves as a generalization and an enhancement to Meinshausen's method. This approach adopts a framework based on the sequential rejection principle (Goeman & Solari, 2010). Goeman & Finos (2012) generalize Meinshausen's testing procedure by introducing the concept of "inheritance", where the significance level, α, associated with the hypothesis at each node can be seen as wealth to be inherited by its descendants. When the hypothesis at a node is

rejected, its α is distributed proportionally to its children. The weight defaults to the number of variables considered within that cluster.

In contrast to the methods described by Meinshausen (2008) and Mandozzi & Bühlmann (2016b), where only children can inherit from parents, the inheritance procedure allows "not yet rejected" siblings, cousins (nodes that share a common ancestor two levels up), second cousins, and so on, to inherit α proportionally if the node has no children. The expanded inheritance framework introduces additional restricted combinations of true and false nulls in the tree, enabling the imposition of constraints at certain nodes to reduce the α adjustment, thereby uniformly increasing statistical power over Meinshausen's method. However, the algorithm involves a recursive method that can be computationally expensive especially as the number of hypotheses grows larger. Similar to Meinshausen's method, the inheritance procedure is not suitable for high-dimensional data. Mandozzi & Bühlmann (2016a) therefore employed similar approaches as described in Mandozzi & Bühlmann (2016b) to extend the method to accommodate high-dimensional settings.

Meijer & Goeman (2014) also provide an interesting generalization of Meinshausen's method for variable selection. The authors structure hypotheses in a directed acyclic graph (DAG), where each node represents a hypothesis, and the edges denote logical relationships between these hypotheses. Unlike Meinshausen's tree-based structure, where each node can have only one parent and hypotheses are either disjoint or fully overlapping, the DAG framework allows for multiple parent nodes and partially overlapping hypotheses. They argue that this flexibility is particularly advantageous in more complex data structures, specifically when testing the association of multiple gene sets of interest with the outcome variable.

The testing approach is similar to Meinshausen's top-down approach, where testing starts from the root of the DAG and proceeds to the children nodes. However, it is based on one of two proposed conditions: the rejection of at least one parent null hypothesis or all parent hypotheses. Multiple testing adjustments are based on the sequential rejection

principle (Goeman & Solari, 2010), just as in the inheritance procedure. The procedure is iterative, starting with an empty set of rejected null hypotheses and adding to the set at each iteration until no more hypotheses can be rejected. This, like the inheritance procedure, can be computationally expensive, especially when dealing with many hypotheses.

The HMTPs discussed by these authors primarily address the issue of variable selection in linear and generalized linear regression when the predictors are highly correlated. However, in many biological experiments, the focus may shift to estimating the association between response variables and different groups or investigating the effects of manipulated experimental conditions on the response, rather than identifying a set of meaningful predictors. In these cases, the problem is reversed from having Q predictors and a categorical response variable to being interested in the marginal associations between each of the Q response variables and the categorical explanatory variable(s). In this study, the aim is to examine the general hierarchical testing procedure under such settings, considering necessary modifications or substitutions to the tests considered and used by these authors. Additionally, the procedure will be extended to accommodate follow-up tests, which occur more frequently under this flipped scenario.

Several alternative approaches have been proposed to address the control of false discovery rate (FDR) in hierarchical multiple testing problems. FDR is the expected proportion of false discoveries among all discoveries, given that at least one discovery is made. Yekutieli (2008) introduced a multiple testing procedure that extends the widely used Benjamini-Hochberg (BH) procedure (Benjamini & Bogomolov, 2014) to cases where the hypotheses are arranged in a hierarchical tree structure. In this procedure, hypotheses at each level are tested simultaneously, taking into account the rejection decisions of their parent hypotheses. To control FDR, Yekutieli (2008) discusses three strategies: Full-tree FDR, Level-Restricted FDR, and Outer-nodes FDR.

The choice of strategy depends on how the total number of discoveries is counted to

control the false discovery rate (FDR) within the hierarchical tree structure. The full-tree strategy counts all discoveries in the entire tree, the level-restricted strategy counts discoveries at a specific level, and the outer-nodes strategy considers only the discoveries at the terminal nodes of the tree. This specification of discoveries is necessary because, under FDR control, a guaranteed rate of false discoveries for an entire family of hypotheses does not necessarily mean the FDR is also guaranteed at that α-level for a subset within the family. By explicitly incorporating the hierarchical order imposed by the tree, and accounting for the specific discoveries of interest, the procedure proposed by Yekutieli (2008) demonstrates improved power compared to the BH procedure without inflating the FDR, particularly in scenarios involving sparse testing problems.

Bogomolov et al. (2020) have also proposed extensions to the BH procedure that provide guaranteed control of the false discovery rate (FDR) for hypotheses arranged within a hierarchical tree. In contrast to the method proposed by Yekutieli (2008), which assumes independently distributed p-values, Bogomolov et al. (2020) offer FDR control under various specified dependency structures. In addition, Bogomolov et al. (2020) present an alternative version of their method that ensures FDR control under arbitrary dependency structures. However, this alternative version imposes more stringent adjustments, which can lead to decreased power compared to the original approach.

The method proposed by Bogomolov et al. (2020) also incorporates additional aspects of control by introducing a new error rate specifically for selected families of hypotheses at a particular level in the hierarchy. These selected families are determined by the order in which hypotheses are tested and the implication relationships that exist within the tree. In other words, for a given level in the hierarchy, only the error rates for families whose ancestors are rejected are controlled, while families that are not directly tested or have non-rejected ancestors are excluded from the control. It is shown that the testing strategies control the selected level-specific FDR at the targeted α, and perform comparably to similar methods in

literature.

Procedures that control the FDR, by design, yield less stringent multiple testing adjustments compared to methods that control the family-wise error rate (FWER), which can lead to power gains. However, when dealing with hierarchical families of hypotheses, FWER control offers a desirable property that is not present under FDR control. Under FWER control, a guarantee on the FWER for the entire tree implies a similar guarantee for any subgroup within the tree. This means that researchers have the freedom to choose which specific resolutions in the hierarchy to focus on for inference after even applying multiple testing adjustments. They can consider discoveries in the full tree or focus on specific subsets, while still maintaining control over the FWER (Finner & Roters, 2001; Goeman & Finos, 2012). For example, Meinshausen's method allows for simultaneous control over all resolutions of the hierarchy at level α, even though individual hypotheses can also be tested at that level. On the other hand, under FDR control, as discussed in studies such as those by Yekutieli (2008) and Bogomolov et al. (2020), the resolution of interest needs to be defined prior to adjustments in order to ensure appropriate control of the error rates. In the subsequent chapters of this work, our focus will be solely on FWER control.

1.4 Dissertation Outline

Chapter 2 reviews suitable clustering methods for grouping variables based on their pairwise sample correlations. We describe several different methods under hierarchical, and partitioning or optimization clustering, and illustrate the methods with packages from the statistical software program (R Core Team, 2023) using data collected from a clinical study to explore the bio-chemical and socio-geographical characteristics of 515 patients (Konopka et al., 2018). The focus is set on hierarchical clustering of variables since it is heavily utilized in the proposed hierarchical testing methods discussed in Chapters 3 and 4.

Chapter 3 introduces a multiple testing procedure for testing hierarchical families of

hypotheses for inter-group differences. The procedure is implemented in the `hiermt` R package. The motivation behind this approach stems from a Type I diabetes mellitus study conducted by Fahrmann et al. (2015) to detect differences in the plasma metabolome of hyperglycemic and normoglycemic mice. The multiple testing procedure starts at the highest node of the hierarchy by testing the global null hypothesis, which posits no difference in true mean abundances for all metabolites (response variables) between the two groups of mice. Subsequently, if strong evidence emerges against the global null hypothesis, a hypothesis for a smaller subset of metabolites is tested. This testing process continues until limited evidence against the null hypothesis is found for a particular subset of metabolites, prompting exploration to cease for that group. Alternatively, testing may reach the terminal node, focusing on group differences for a single metabolite. Furthermore, this chapter investigates the impact of different linkage criteria, which dictate how clusters are merged in hierarchical clustering, on family-wise error rates and statistical power. Finally, a comparative analysis is performed between the proposed procedure and existing methods, revealing enhanced statistical power and improved interpretability derived from considering the relationships within the hierarchical family of hypotheses.

Chapter 4 builds upon the findings of Chapter 3 by expanding the multiple testing procedure to accommodate scenarios involving more than two groups of observations. That is, if there is substantial evidence against the null hypothesis at a terminal node, pairwise comparisons are performed. Through simulations and theoretical analysis, we demonstrate that the testing procedure effectively controls the family-wise error rate (FWER) at all levels of the hierarchy, including terminal nodes with and without pairwise comparisons. Moreover, the logical constraints imposed by the hierarchical structure enable us to enhance the statistical power for detecting pairwise differences. We implement our procedure on data from a study by Petr et al. (2021) that investigated metabolomic changes in mice of various ages.

Chapter 5 of the dissertation serves as the concluding chapter, providing discussions on the findings presented in earlier chapters. In addition, this chapter offers insights into potential areas for methodological improvement, highlighting both the strengths and weaknesses of the proposed methods. The drawbacks of the proposed methods are identified and addressed, and recommendations for overcoming these limitations are presented. Additionally, this chapter also explores potential extensions of the proposed methods, offering new avenues for future research in the field.

CHAPTER TWO

CORRELATION-BASED VARIABLE CLUSTERING

Contribution of Authors and Co-authors

Author: Priscilla Bacino

Contributions: Responsible for majority of the writing

Co-Author: Dr. Mark Greenwood

Contributions: Provided feedback on statistical analysis and drafts of the manuscripts

Manuscript Information Page

Priscilla Bacino, Mark C. Greenwood

Status of Manuscript:
___x___ Prepared for submission to a peer-reviewed journal
_____ Officially submitted to a peer-reviewed journal
_____ Accepted by a peer-reviewed journal
_____ Published in a peer-reviewed journal

Abstract

Variable clustering is a statistical technique that groups variables based on their similarity or relationship with each other. In this review paper, we examine the methods used for variable clustering using correlation as the measure of similarity. Some of the methods are adapted from clustering techniques developed for observations and some are purpose-built for the problem of interest. We compare and contrast the different methods in detail. Our review provides insights into the strengths and limitations of each method and provides guidance on their appropriate use in different scenarios. To illustrate these concepts and methods, we use a real-world dataset containing biological attributes of 515 elderly patients.

2.1 Introduction

Clustering, or cluster analysis, refers to a broad set of techniques for identifying groups of homogeneous vectors, also known as clusters, within a dataset. These techniques are used to discover patterns and relationships in the data without using a predetermined response variable, and thus are considered as a type of unsupervised learning (Everitt, 2011; Hartigan, 1975; Landau et al., 2011; Maechler et al., 2022). In many applications of clustering, the vectors considered for grouping are observations such as customers of a product, or subjects within a study. In some settings, also, it is desirable to discover groups of similarly behaving variables.

For example, a common practice in modern research is to comprehensively collect data which often results in the measurement of many variables. Several of these variables may be highly correlated with others and thus could be considered redundant for the purposes of gathering information, learning from the data or creating groups of variables that might be of interest to elicit similar behavior to understand the system or process better. Clustering variables provides a way to assess collinearity and identify those similar groups and/or redundant variables. Each cluster could be represented with one variable from the cluster or a synthetic variable calculated from all the variables in the cluster, thereby reducing the dataset

to fewer variables that contain most of the information. This can be especially instrumental in high-dimensional settings because dimension reduction techniques allow the use of simpler techniques that are more suitable or possibly only applicable to lower-dimensional data (Maechler et al., 2022).

In other situations, incorporating the resulting cluster solution as supplementary information into can improve statistical power, predictions, or inference in an analysis. For example, in Chapters 3 and 4 of this dissertation, multiple testing procedures that use hierarchical clustering solutions to structure hypotheses are presented. It is shown that incorporating such additional information into multiple testing procedures improves signal detection and identification. Additionally, it is also shown that knowing which hypotheses are related can enhance interpretation of conclusions and results. Hierarchical clustering, for instance, plays an important role in the works of Meinshausen (2008), Hu et al. (2010), Sankaran & Holmes (2014), Mandozzi & Bühlmann (2016a) and Mandozzi & Bühlmann (2016b).

To cluster variables, a protocol must be established to determine similarity of variables and which variables should be grouped together. The protocols vary across techniques, resulting in a diverse range of methods that can be employed to achieve this goal. In this paper, we provide a review of several different methods that are either hierarchical, or partitioning (or optimization) clustering. While there are also clumping methods that permit variables to belong to multiple clusters, these will not be discussed in this paper. Additionally, this review will focus on the clustering of continuous variables, although many of the techniques discussed could be applied to mixed (continuous and categorical) data if suitable similarities or dissimilarities can be defined.

The structure of the paper is as follows: Section 2.2 presents various graphical methods for exploring the data to be clustered, providing insights into the expected patterns. Section 2.3 compares and contrasts popular measures of dissimilarity for clustering variables. Section

2.4 discusses some of the algorithms within the types of clustering that utilize correlation-based dissimilarities. Lastly, Section 2.5 provides a concluding section.

2.2 Exploring Clusters Graphically

Before clustering variables it can be helpful to visualize the data to get a sense of the structure, relationships, or outliers in the data. Visually exploring clustered data also gives clues about whether there are any apparent clusters which can help with deciding whether clustering is a good approach and, if so, what method to use. This section discusses a few different visualization techniques, and demonstrates how to apply them using the statistical software R (R Core Team, 2023).

For illustration, the dataset from a clinical study described in Konopka et al. (2018) where 44 biochemical and socio-geographical variables were collected on 515 patients will be used. Out of the 44 variables, 23 were continuous, and 21 were categorical or nominal. Considering the scope of our review, the focus will only be on the continuous variables in the dataset. Table 2.1 provides a list of the continuous variables and their descriptions as presented in Konopka et al. (2018). The study was approved by the Bio-ethical Committee of the Medical University of Silesia and informed written consent, including consent for genetic studies, was obtained from all of the subjects before testing.

2.2.1 Heatmaps

A heatmap is a two-dimensional graphical representation of data in which values are represented by colors. Heatmaps can be used to visualize clusters or space, but in this context, we will focus on cluster heatmaps. Cluster heatmaps are used to visualize large-scale multivariate data, as it allows for the representation of many variables in a single plot. They are constructed by arranging the data in a grid and using a color scale to encode the values.

Table 2.1: Continuous variables from the clinical datasets with their descriptions.

Variable Name	Description
AGE	Age (years)
HEIGHT	Height (cm)
WEIGHT	Weight(kg)
WAISTLINE	Waistline (cm)
HIP.GIRTH	Hip girth (cm)
BMI	Body mass index (kg/m^2)
FAT	Body fat as a percentage of body weight (%)
CHOL.HDL	Cholesterol serum level-High Density Lipoprotein (mg/dl)
CHOL.LDL	Cholesterol serum level-Low Density Lipoprotein (mg/dl)
CHOL.TOTAL	Total cholesterol level (mg/dl)
TGC	Triglycerides serum level (mg/dl)
GLUCOSE	Glucose serum level (mg/dl)
INS	Insulin serum level (muIU/ml)
TESTOSTERONE	Testosteron serum level (nmol/l)
ESTRADIOL	Estradiol serum level (pmol/l)
DHEA.S	Dehydroepiandrosteron serum level (ng/dl)
SHGB	Sex hormone binding globulin serum level (pmol/l)
FAI	Free Androgen Index (Ratio of total testosterone to SHBG x 100)
FEI	Free Estradiol Index (Ratio of total estradiol to SHBG x 100)
FSH	Follicle-simulating hormone serum level
ICTP	Carboxy-terminal cross-linked telopeptide of type I collagen serum level
OPG	Osteoprotegerin serum level
VITAMIN.D	Vitamin D serum level

The color scale may range from cool colors (such as blues and greens) to warm colors (such as reds and yellows), or may vary in intensity within a specific color to distinguish smaller values from larger ones. Note that heatmaps plot the original values from the data and as such do not provide any quantitative measure of relationships between variables.

There are several packages in R that can be used to create heatmaps. Static heatmap packages, such as `stats` (R Core Team, 2023), `pheatmap` (Kolde, 2019), `gplots` (Warnes et al., 2022), and `ggplot2` (Wickham, 2016) allow for the creation of traditional heatmap plots that can be saved and viewed offline. Interactive heatmap packages, such as `heatmaply`

(Galili et al., 2017), and `plotly` (Sievert, 2020) allow for the creation of interactive heatmaps that can be explored and manipulated online.

Additionally, several packages offer functions to create custom color palettes, including those that are color-blind friendly, which are particularly useful for heatmap plots. Given that the sole information in these plots regarding the magnitude of variable values is represented by color and hue, selecting color-blind friendly palettes becomes especially crucial. Packages in this regard include RColorBrewer (Neuwirth, 2022), viridis (Garnier et al., 2021), and grDevices (R Core Team, 2023).

The `heatmap` function in the `stats` packages comes with the base installation of R. It is simple to use, allows the specification of a color palette or color mapping function, hierarchical clustering and the display of resulting dendrograms with an additional line of code needed to add a legend. The `pheatmap` function from the `pheatmap` package adds the ability to annotate rows and columns, display a color key or legend with specified scaling, choose between k-means clustering and hierarchical clustering, and divide the rows or columns of the heatmap based on a specific number of resulting clusters. The `heatmap.2` function in the `gplots` package offers added functionality compared to the standard **`heatmap`** function, including the provision of a legend, density information, and trace lines. Here, "trace lines" refer to lines that can be added to a heatmap to demarcate the rows and/or columns. These lines enhance the visual separation of different parts or segments of the heatmap, aiding in readability and interpretation. The `geom_tile` function provides the most customization over `heatmap` along with all the many options available in `ggplot` for making plots, including control over individual tile appearance.

`Heatmaply` is designed specifically for creating interactive heatmaps online and offers a wide range of features. It provides control over the color scale, the ability to cluster rows and columns, and the ability to display data values upon hover. Additionally, `heatmaply` supports hierarchical clustering and the creation of heatmap dendrograms. It also offers the

ability to add annotations to the heatmap and to display multiple heatmaps side-by-side for comparison. One of the standout features of `heatmaply` is its interactivity. It is built using the `plotly` library and provides an interactive heatmap that allows users to hover over cells to see their values. `Plotly` provides a general-purpose visualization library that also offers its own heatmap functionality through the "heatmap" trace type. It provides control over the color scale and the ability to display data values upon hover. Like `heatmaply`, `plotly` also supports hierarchical clustering and annotations. However, compared to `heatmaply`, `plotly` may require additional work to create a heatmap because `plotly` is not specifically designed for heatmap visualization but it is designed to offer a wide range of visualizations with options for many data scenarios.

`RColorBrewer` is a color palette that offers a wide range of color schemes, including sequential, diverging, and qualitative palettes, with a focus on color accuracy and accessibility (such as color-blind friendly color palettes). `RColorBrewer` provides many built-in color palettes that can be easily used in R, as well as the ability to create custom color palettes. `Viridis` on the other hand provides colors that are easily distinguishable from one another, perceptually uniform, and colorblind friendly. The `viridis` color palette is ideal for use where it is important for the colors to preserve the overall order of the data being visualized. `ColorRampPalette` provides the ability to create custom color palettes. It takes as input a color ramp function and an optional number of colors, and returns a function that can be used to assign colors to data. This makes `colorRampPalette` a flexible option that can be used to create any type of color palette, including sequential, diverging, and qualitative palettes.

The heatmap shown in Figure 2.1 does not exhibit any apparent patterns when considering the magnitude of the data values. It is important to note that when heatmaps are used to plot raw magnitudes, they have a limitation in that they only plot the values of the data without providing information about the relationships between variables. As a

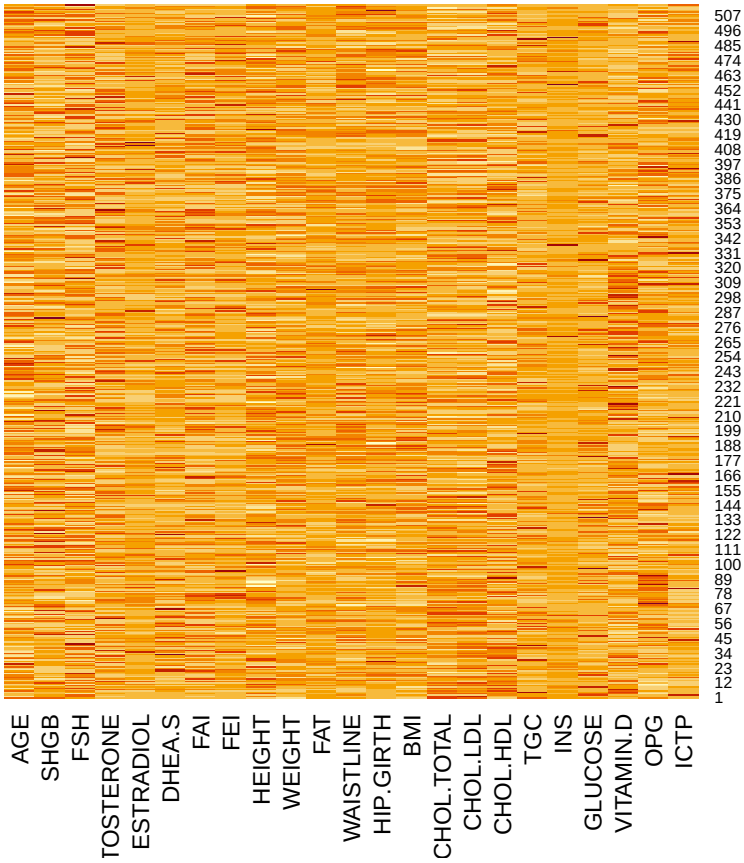


Figure 2.1: A heatmap of the standardized continuous attributes of the 515 clinical patients using the heatmap function from the stats package. Darker tiles represent higher attribute values and lighter indicate lower attribute values. These results are in the order of the rows and columns in the dataset.

result, the use of heatmaps in such scenarios may not be sufficient to reveal the underlying patterns and associations among variables. Sorting or rearranging variables and/or subjects can provide an enhanced ability to describe patterns in the data.

2.2.2 Correlogram

A correlogram is typically a square, symmetric grid that describes the associations among variables with colors. Each cell in the grid represents the relationship between two variables which is often quantified using the correlation coefficient. The correlation coefficient

is a measure of the degree and direction of association between two variables, mostly based on Pearson's r or Spearman's rank correlation. In the correlogram, the color of the grid and its intensity are based on the value of the correlation coefficient. Often more intense cool colors indicate stronger positive correlations and more intense warm colors indicate stronger negative correlations. The colors in the correlogram therefore highlight the variables that are most correlated in magnitude, making it easier to identify clusters and patterns in the variables.

A useful function for creating correlograms is the `corrplot` function, available in the `corrplot` (Wei & Simko, 2021) package. This function creates a graphical representation of a correlation matrix, where the relative size of certain glyph options indicate the strength of the correlation. The `corrplot` function also supports various visual encodings, such as color and text labels. It also offers options to order the variables and include correlation test p-values and confidence intervals on the graph. It is important to note that when the `corrplot` function orders variables based on a hierarchical cluster analysis, it uses a dissimilarity measure of $1 - r(Y_q, Y_{q'})$, where Y_q represents the qth variable in the dataset and $r(Y_q, Y_{q'}), q, q' \in 1, \ldots, Q$ represents the correlation coefficient between variables Y_q and $Y_{q'}$. Dissimilarities measure proximity or the level of "closeness" between pairs of variables, see Section 2.3 for more details.

This means that pairs of correlation coefficients that have equal strength but have opposite directions will have different distances. For example, a correlation coefficient of 0.5 will produce a distance of 0.5, while a correlation coefficient of -0.5 will produce a distance of 1.5. Consequently, negatively correlated variables will be treated as being more distant from each other, and may end up in different clusters. The `corrplot` function used throughout this work is modified to use the dissimilarity measure $\sqrt{2(1 - |r(Y_q, Y_{q'})|)}$ (James et al., 2013), which ensures that distances between equally strong but directionally different correlation coefficients are the same.

The `ggcorrplot` package (Kassambara, 2022) offers the `ggcorrplot` function, which serves as an alternative to the `corrplot` function within the `ggplot2` framework for creating correlograms. One advantage of using `ggcorrplot`, like other extensions of `ggplot2`, is the ability to leverage the extensive tools and options provided by `ggplot2` such as adding additional layers of plots, adjusting the aesthetics, and applying themes. However, it's important to note that the `ggcorrplot` function has limited functionality compared to `corrplot`. Specifically, it only allows for ordering variables using hierarchical clustering, whereas `corrplot` offers additional options such as alphabetical ordering, angular ordering of eigenvectors, and ordering based on the first principal components. Similarly to `corrplot`, caution should be exercised when ordering variables using hierarchical clustering, as the `ggcorrplot` function calculates distances using a dissimilarity measure of $(1 - r(Y_q, Y_{q'}))/2$, treating negative correlations differently from positive ones.

The correlogram displayed in Figure 2.2 provides valuable insights into the relationships between the measured attributes of the patients and suggests the presence of several distinct clusters within these attributes. One notable cluster is observed in the top left corner of the correlogram, indicating strong positive correlations among several variables. This cluster is meaningful and likely represents real associations since it encompasses various anthropometric measurements, such as hip girth, waistline, weight, and BMI. The positive correlations among these variables suggest that individuals with larger hip girths tend to have larger waistlines, higher weights, and higher BMIs which align with the expected patterns commonly observed in relation to these anthropometric measurements.

Additionally, Figure 2.2 reveals a strong positive correlation between the Free Estradiol Index (FEI) and estradiol serum levels, reflecting the relationship between overall estradiol in the serum and its bio-available fraction. Similarly, there is a strong positive correlation between the Free Androgen Index (FAI) and testosterone levels, indicating that as testosterone rises, its bio-available fraction also increases. Contrarily, both FAI and testosterone demonstrate

a strong negative correlation with FSH (Follicle-stimulating hormone) levels. This is consistent with the body's feedback mechanisms; elevated testosterone typically suppresses FSH production to maintain hormonal balance.

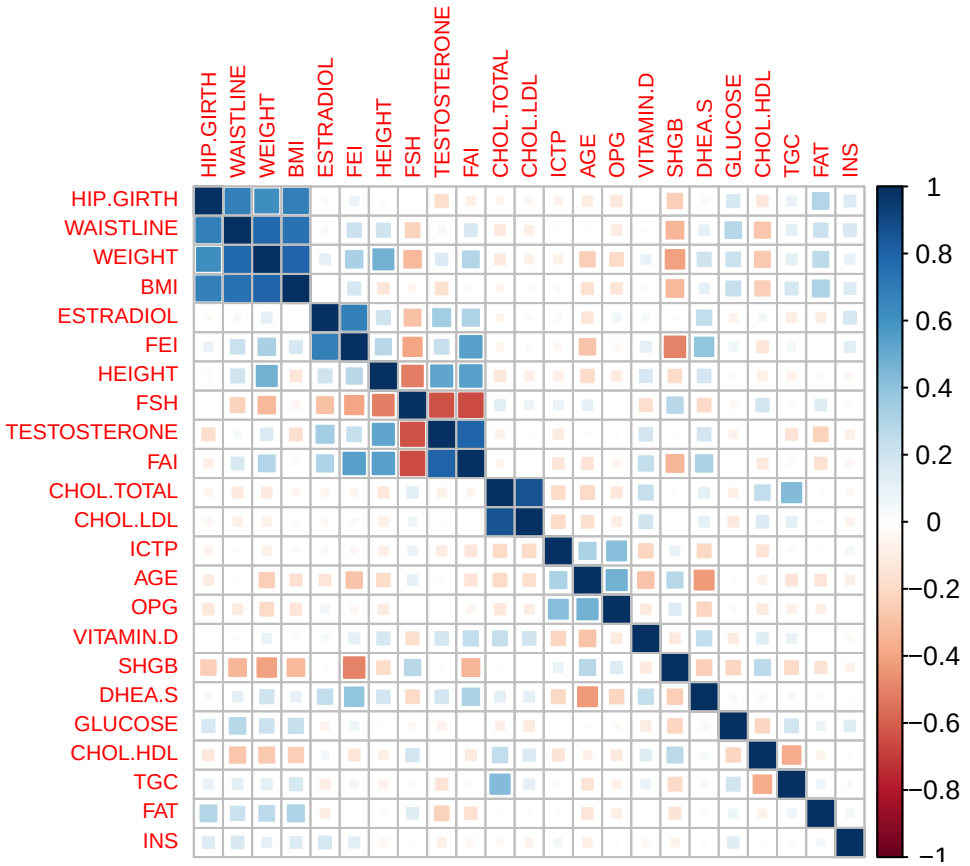


Figure 2.2: Correlogram of the standardized continuous attributes of the 515 patients from the clinical study using a modification of the corrplot function from the corrplot package. Darker or larger tiles represent stronger correlations and lighter or smaller tiles represent weaker correlations. Red and blue tiles represent negative and positive correlation values respectively.

2.2.3 Projecting into Lower Dimensions

Heatmaps and correlograms serve as convenient and effective tools for exploring the relationship between two variables in small to moderately large datasets before applying clustering techniques. However, in the case of large datasets with a high number of variables,

such as hundreds, thousands, or even millions, these methods can encounter computational inefficiencies and may become visually overwhelming. Additionally, the limited scope of heatmaps and correlograms in providing pairwise information may obscure complex relationships that may exist among multiple variables. Moreover, the ordering of variables on these plots can also have a demonstrable impact on the perceived pattern of correlations, potentially leading to biased interpretations.

To overcome these challenges, dimension reduction techniques can be employed. These techniques involve projecting and visualizing the original set of features in a lower-dimensional space while preserving most of the inherent structure. Examples of dimension-reducing techniques include multi-dimensional scaling (MDS) and principal component analysis (PCA); for more information on both methods see Härdle & Simar (2015). These techniques enable the identification of patterns within complex datasets without being overwhelmed by their size or complexity. Note, that these methods can be considered as standalone approaches for analyzing and comprehending the underlying structure present in the data, similar to variable clustering techniques. They can also serve as companion techniques to use along side clustering.

Classical multidimensional scaling (MDS) applied to a dissimilarity of variables aims to create a spatial representation of the differences or similarities among variables. It accomplishes this by placing points, from a dissimilarity or distance matrix (see Section 2.3 for more details), into a lower-dimensional Cartesian space. The goal is to find coordinates for each point that minimize the discrepancy between the original dissimilarity matrix and the distances in the lower-dimensional space, often referred to as stress. In the context of visualizing clusters, points are typically projected into a two-dimensional space. Points that are similar or have smaller dissimilarities in the original space are positioned closer together in the MDS plot, while points with larger dissimilarities are farther apart. Note that the choice of dissimilarity measure can impact the results, how the variables will be plotted, and

how they will be interpreted.

The base R function `cmdscale` from the `stats` package provides a straightforward implementation of classical multidimensional scaling. It requires arguments: the dissimilarity or distance matrix and the maximum dimension of the low-dimensional space used to represent the data. An example of an MDS plot is presented in Figure 2.3. In this, the cluster of anthropometric variables in the top right corner is prominently displayed, and is consistent with the findings from the correlogram in Figure 2.2. Furthermore, the plot suggests a strong similarity between the total cholesterol and low-density lipoprotein cholesterol levels of the patients. Other variables that exhibit similar information based on the plot include Testosterone and FSH, as well as FEI and Height.

Principal component analysis (PCA) is a powerful technique for reducing the dimensionality of data and can also aid in exploratory data analysis. PCA transforms the original variables into a new set of uncorrelated variables called principal components, while preserving the variance-covariance structure of the original variables. The first principal component represents the axis along which the data points exhibit the largest variation from the origin. The second principal component is orthogonal to the first and captures the next largest variability, and so on. Each subsequent principal component captures progressively less variability compared to the previous ones. The principal components are calculated as linear combinations of the original variables, using the eigenvectors of the variance-covariance matrix as the weights or coefficients. By retaining the first few principal components that capture the majority of the variability, the dimensionality of the data can be reduced while preserving most of the important information.

To visualize the relationships among variables using PCA, a biplot is commonly used. In a biplot, the variables are represented as arrows or vectors in a plot where axes are defined by the first two principal components. The direction and length of the arrows indicate the relationships and importance of the variables in the reduced-dimensional space. Variables

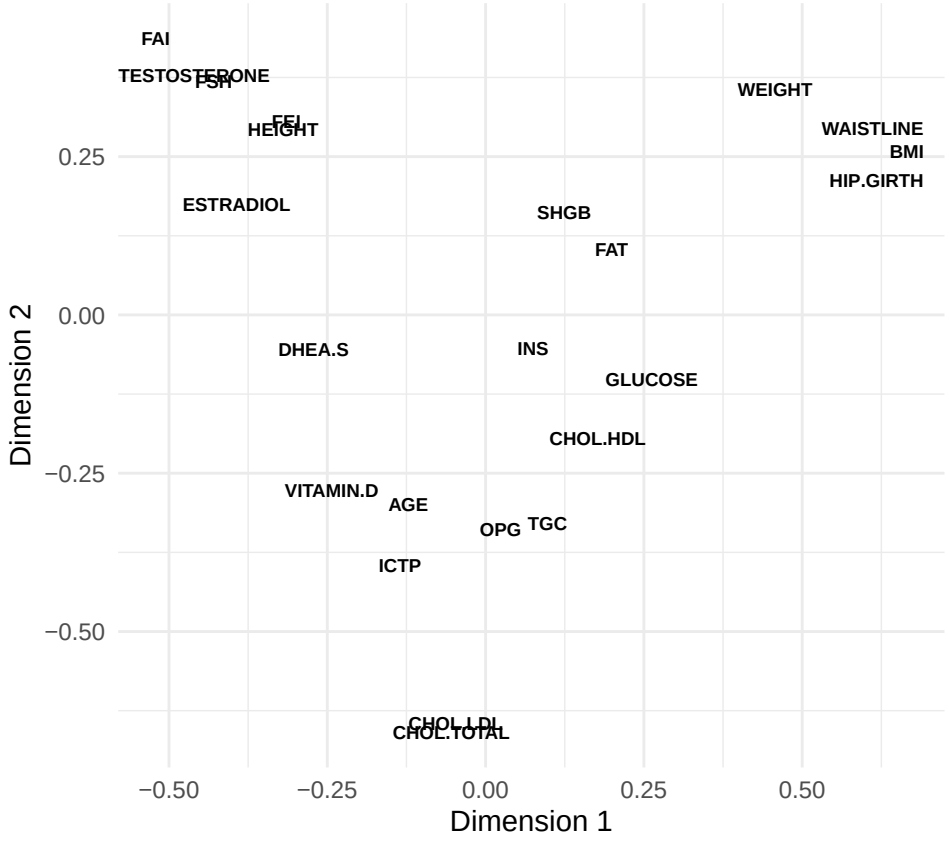


Figure 2.3: Multidimensional scaling of the standardized continuous attributes of the 515 clinical patients onto a 2-dimensional space.

that have similar directions or are close together in the biplot are considered to be related or have similar patterns of variation. In R, the `biplot` function from the `stats` package can be utilized to generate biplots of the variables in the reduced-dimensional space obtained from PCA. The resulting plot provides insights into the similarities and dissimilarities among the variables by the angle of the displayed variable vectors. That is, variables that lie on similarly oriented vectors are likely correlated and orthogonal vectors suggest less correlated variables. In Figure 2.4, for example, it can be observed that variables such as hip girth, BMI, waistline, weight, fat, glucose, and TGC are similar and share information, while they are dissimilar to variables like FEI, DEAS, vitamin D, testosterone, FAI, and estradiol, which

also exhibit similarities among themselves. This visualization aligns with the findings from the correlogram and the MDS plot.

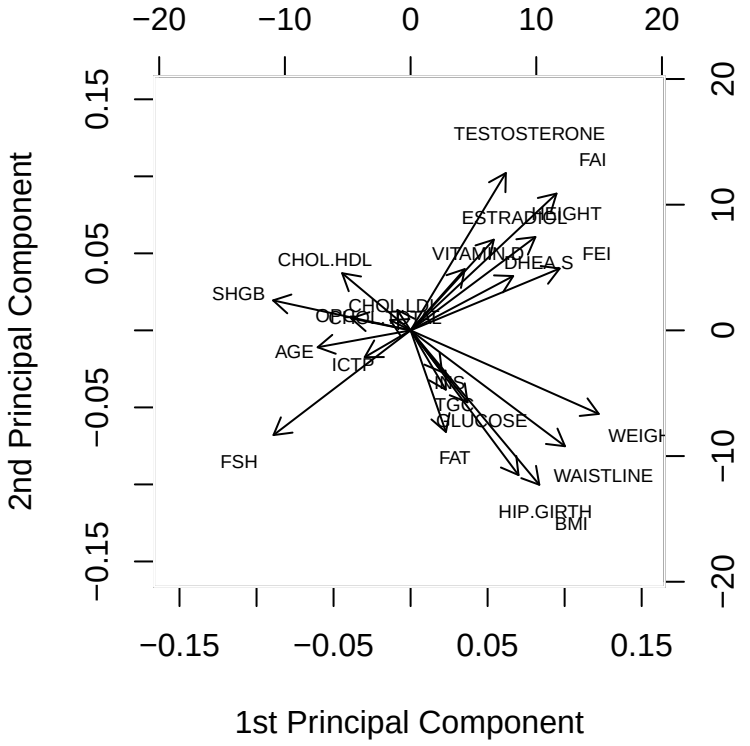


Figure 2.4: A biplot of the first and second principal components of the standardized continuous attributes from the clinical study dataset.

2.3 Measures of Dissimilarity

A key component in clustering is measuring the level of "closeness" between pairs of variables. This notion, often referred to as proximity, can be approached from two perspectives: similarity and dissimilarity. Measures of dissimilarity evaluate how far apart variables are from each other, whereas measures of similarity assess their level of agreement. While both perspectives aim to quantify the same underlying concept, clustering algorithms that rely on

proximity as an input often require a measure of dissimilarity instead of similarity.

For a given dataset with dimensions $n \times Q$, a dissimilarity matrix of size $Q \times Q$ can be computed, where each cell corresponds to the dissimilarity between a specific pair of variables. The dissimilarities could also arise directly from subjective distance scores or distances measured physically or from graphs. Let $d(Y_q, Y_{q'})$ denote the dissimilarity or distance between two variables Y_q and $Y_{q'}$. There exists a wide range of measures that can be used to calculate the dissimilarity between the variables, each with its own strengths and limitations. Different dissimilarity measures may be more suitable for specific types of data or clustering objectives, and they will be interpreted differently. However, smaller dissimilarity values generally indicate a higher degree of similarity or proximity between variables, while larger values signify greater dissimilarity.

When clustering observations, the most typical distance measure is the Euclidean distance. This measure calculates the "as-the-crow-flies" distance between two points in Euclidean space, which measures the length of a line between two points. Mathematically, the Euclidean distance between two variables is defined as $d_{\text{euclidean}}(Y_q, Y_{q'}) = ||Y_q - Y_{q'}|| = \sqrt{\sum_{i=1}^{n}(Y_{iq} - Y_{iq'})^2}$, where Y_q and $Y_{q'}$ are two distinct variables of sample size, n. The Euclidean distance considers two objects as similar if the observed values are closer to each other and dissimilar if the values are far apart. However, the Euclidean distance is unsuitable as a dissimilarity measure for clustering variables on different scales. This is because even if two variables share similar information but are on different scales or if the variables are negatively correlated, they will be deemed dissimilar by the Euclidean distance.

For example, we have observed from Figures 2.3 and 2.4 that BMI and waistline share similar information, and BMI and vitamin D exhibit apparent differences. However, the Euclidean distance between BMI and waistline is approximately 1599, while the distance between BMI and vitamin D is approximately 528. According to the Euclidean distance, it is implied that vitamin D and BMI are considered more similar, but this observation is solely

based on the proximity of the magnitude of their values. This finding contradicts the insights obtained through the various exploratory analyses conducted in Section 2.2.3.

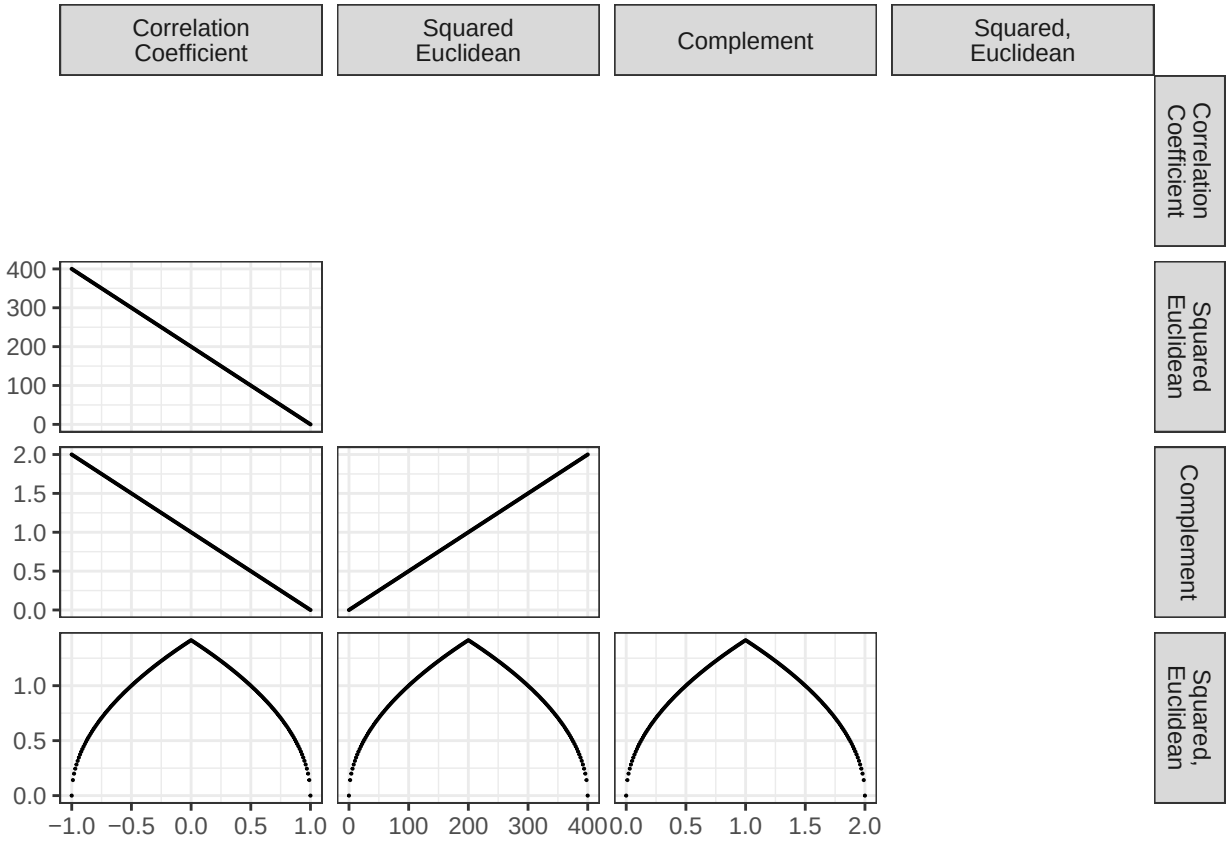


Figure 2.5: Scatterplot matrix illustrating the relationship between Pearson correlation coefficient (-1 to 1) and various distance measures, including the Euclidean distance and various correlation-based dissimilarities.

Even when variables are standardized to have means of 0 and standard deviations of 1, the Euclidean distance distinguishes negative differences from positive ones. This behavior mirrors the correlation-based dissimilarity, $1 - r(Y_q, Y_{q'})$, in how it addresses variables with positive and negative correlations. Upon standardization, the Euclidean distance directly relates to the correlation-based distance: $d^2_{\text{euclidean}}(Y_q, Y_{q'}) = 2n(1 - r(Y_q, Y_{q'}))$ (Legendre & Legendre, 2012). Figure 2.5 graphically represents this relationship, highlighting the one-to-one relationship between squared Euclidean distances of standardized variables and

the dissimilarities computed using $1 - r(Y_q, Y_{q'})$. Yet, an apparent deviation is seen when these distances are juxtaposed with those derived from $\sqrt{2(1 - |r(Y_q, Y_{q'})|)}$ which underscores that variables with negative correlations have reduced distances with $\sqrt{2(1 - |r(Y_q, Y_{q'})|)}$ but greater distances with the Euclidean measure.

Drawing from values in the clinical dataset, Figure 2.6, aligns with the theoretical presentation in Figure 2.5. It stresses that such patterns are not only theoretical propositions but tangible realities. This is particularly pertinent for platforms like the widely used metabolomics analysis tool, Metaboloanalyst 5.0 (Pang et al., 2021). Heatmaps are generated using Euclidean distances on this platform, and might unintentionally distinguish between positively and negatively correlated metabolites, a nuance many users might overlook or might not be aware of. Therefore, it is essential for users to be meticulous in understanding the underpinnings of the platforms they employ so that they are conversant with the inherent decisions made by such systems.

Other distance-based measures, such as the Manhattan distance (Larson & Sadiq, 1983), might exhibit characteristics similar to the Euclidean distance, even so, they can be more challenging to interpret, thus their use in gauging variable proximity is not advised. In the context of variable clustering, the absolute magnitudes of the variables are often of secondary importance since the primary objective is to identify clusters of variables that share similar information. Therefore, employing a correlation-based distance measure that makes use of the absolute pairwise sample correlations between the variables, such as $\sqrt{2(1 - |r(Y_q, Y_{q'})|)}$, will be a more appropriate choice as a dissimilarity measure under such scenarios.

A common choice for computing the correlation coefficient in variable clustering is often Pearson's r, given by $\widehat{\text{cov}}(Y_q, Y_{q'})/\hat{\sigma}_{Y_q}\hat{\sigma}_{Y_{q'}}$, where $\hat{\text{cov}}(Y_q, Y_{q'})$ is the estimated covariance between Y_q and $Y_{q'}$, and $\hat{\sigma}_{Y_q}$ and $\hat{\sigma}_{Y_{q'}}$ are their respective standard deviations. Another option is Spearman's correlation coefficient, given by $\widehat{\text{cov}}(R(Y_q), R(Y_{q'}))/\hat{\sigma}_{R(Y_q)}\hat{\sigma}_{R(Y_{q'})}$, where $R(.)$ represents the rank of a variable, from 1 to n (with some adjustments of ranks made for

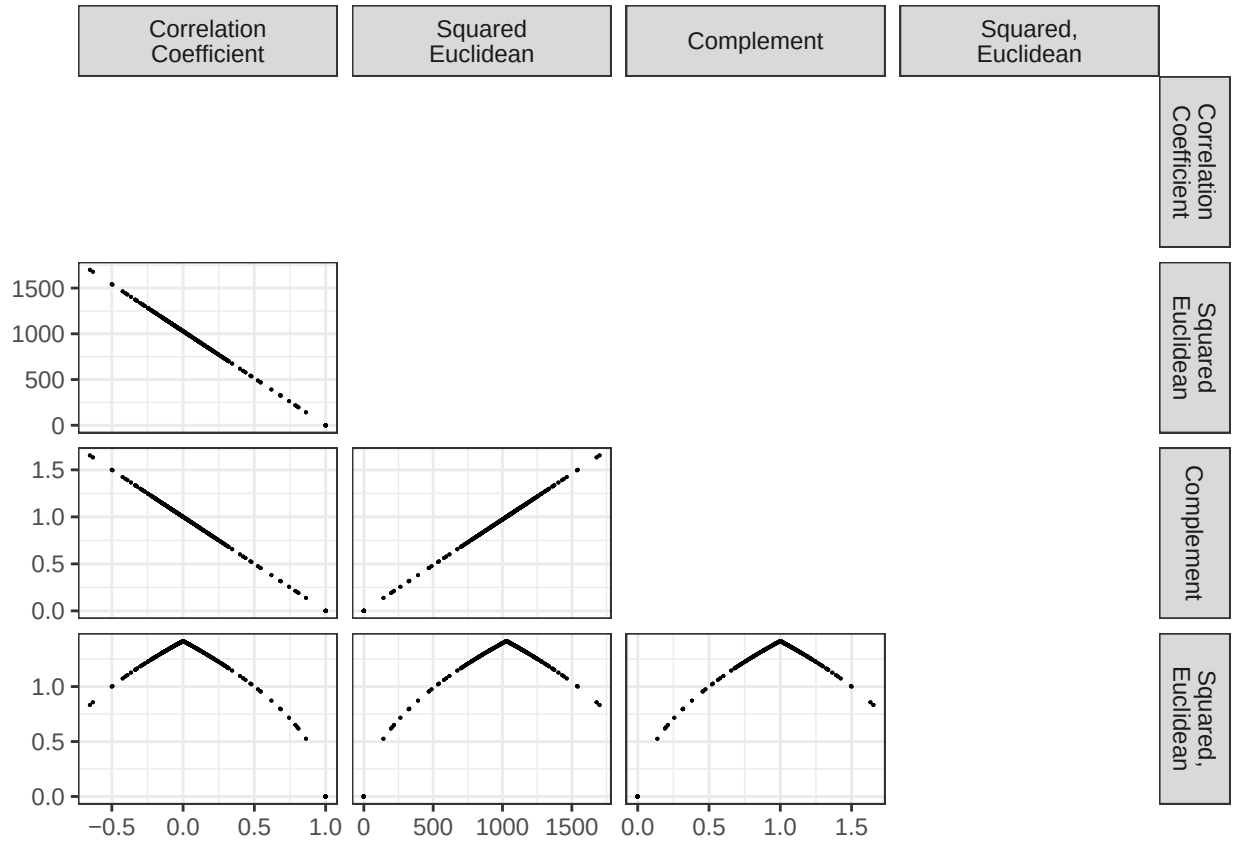


Figure 2.6: A comparison of the euclidean distance and the correlation-based dissimilarities of the standardized attributes from the clinical dataset.

tied observations). Pearson's correlation is sensitive to outliers. In many scenarios involving variable clustering, especially in high-dimensional datasets, it becomes crucial to consider the potential impact of outliers. In such cases, utilizing Spearman's correlation in the correlation-based dissimilarity measure can help mitigate the influence of outliers and provide a more resistance measure of association between variables. Euclidean distance of variables being equivalent to the Pearson correlation-based dissimilarity highlights the potential impact of outliers on the dissimilarity.

If all correlation coefficients are positive, meaning $r(Y_q, Y_{q'}) > 0$, then using the Euclidean distances or $1 - r(Y_q, Y_{q'})$ is typically not problematic. Yet, in real-world applications,

particularly within metabolomics, this ideal scenario is rare. Given the heavy reliance on heatmaps, correlograms, and clustering for inference and interpretation in this field, addressing these nuances is imperative, and a cautious approach is advised to researchers.

2.4 Methods for Variable Clustering

2.4.1 Hierarchical Clustering

Hierarchical clustering is a common class of clustering methods for grouping variables using proximity. Partitioning the set of variables into clusters using these methods may be agglomerative, or divisive. An agglomerative hierarchical clustering technique starts with each variable in its own cluster, and then iteratively merges the most similar clusters until all variables are in the same cluster. Divisive hierarchical clustering, on the other hand, starts with all variables in a single cluster, and then divides the cluster into smaller and smaller clusters until each variable ends in its own cluster.

Hierarchical clustering solutions are typically visualized using a two-dimensional diagram called a dendrogram. The dendrogram illustrates the fusions or divisions performed within the algorithm. An example of such a dendrogram is shown in Figure 2.7. The structure of a dendrogram resembles a binary tree, where the top-most cluster is located at the root and contains all the variables to be clustered. In divisive clustering, this top cluster serves as the starting point of the algorithm, while in agglomerative clustering, it represents the final cluster formed at the end of the algorithm, as seen in Figure 2.7. The height or distance between each node in the dendrogram represent the distances between the corresponding clusters.

2.4.1.1 Agglomerative Clustering In agglomerative clustering, the height or distance between each node is calculated using a linkage criterion. The linkage criterion specifies which two clusters should be fused at the next stage of the algorithm. Popular methods include

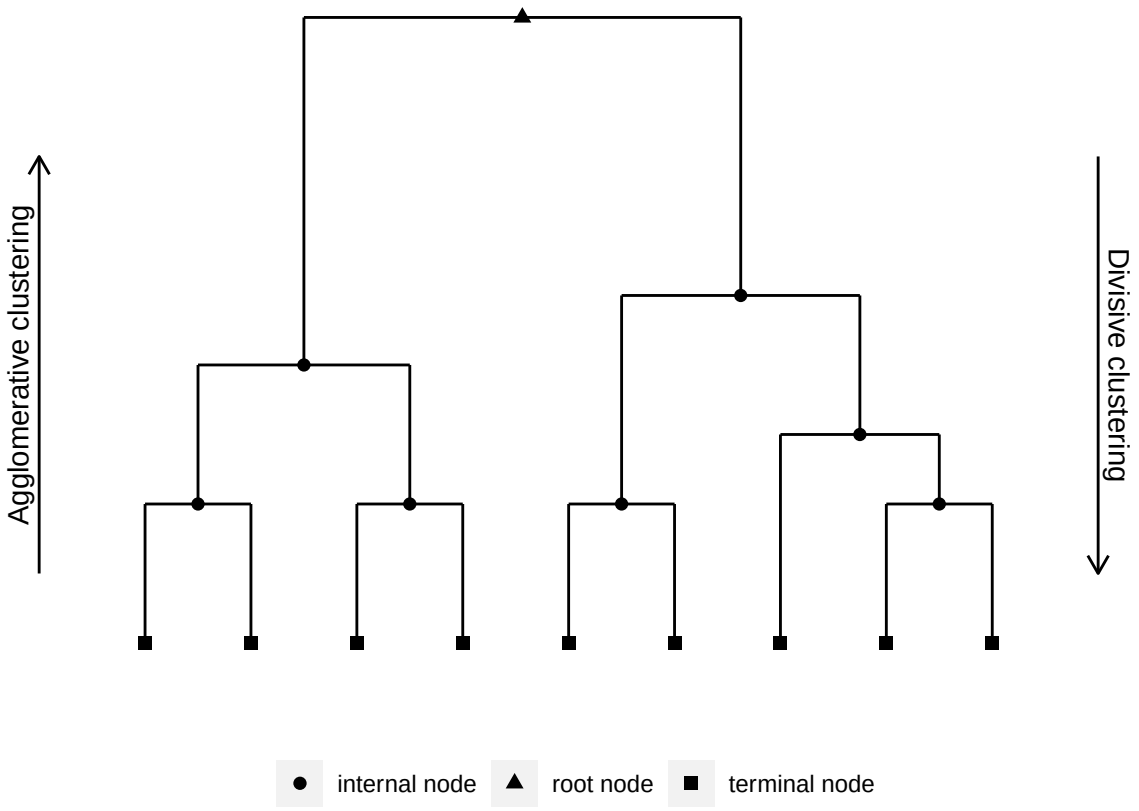


Figure 2.7: Hierarchical clustering illustrated in a dendrogram. The node with the triangle represents the root, circles represent branches, and squares represent terminal nodes or leaves.

single linkage, complete linkage, average linkage, and Ward's method. We denote $d(C_k, C_{k'})$ as the distance between two clusters C_k and $C_{k'}$ of variables. Single linkage, also known as nearest-neighbor linkage, defines the distance between two clusters as the minimum distance between pairs of their variables. Specifically, $d_{\text{single}}(C_k, Ck') = \min_{Y_q \in C_k, Y_{q'} \in C_{k'}} d(Y_q, Y_{q'})$. Single linkage is a simple but computationally inefficient method, and it can be prone to the "chaining" effect. The "chaining" effect occurs when outlier connections between two distinct groups of variables cause these groups to be erroneously merged into a single cluster. The chaining effect often leads to singleton clusters and elongated and skewed dendrograms, and is particularly prominent when dealing with a large number of variables. It is invariant

to monotonic transformations of the dissimilarity, thus for example, $\sqrt{2(1-|r(Y_q,Y_{q'})|)}$ and $1-|r(Y_q,Y_{q'})|$ will generate the same cluster solution.

Complete linkage, also referred to as the farthest-neighbor linkage, defines the distance between two clusters as the maximum distance between any pair of variables belonging to the respective clusters. Formally, $d_{\text{complete}}(C_k, C_{k'}) = \max_{Y_q \in C_k, Y_{q'} \in C_{k'}} d(Y_q, Y_{q'})$. This linkage approach exhibits reduced sensitivity to noise and outliers compared to single linkage, resulting in more compact and spherical clusters. However, the computational cost of complete linkage tends to be also high, especially when dealing with large datasets. Complete linkage is also invariant to monotonic transformations of the dissimilarities.

Average linkage (Sokal & Michener, 1958) calculates the distance between two clusters as the average of the distances between all pairs of variables, one variable from each cluster. This method is sometimes referred to as the unweighted pair-group average method (UPGMA). Mathematically, the linkage, $d_{\text{average}}(C_k, C_{k'}) = \sum_{Y_q \in C_k} \sum_{Y_{q'} \in C_{k'}} \frac{d(Y_q, Y_{q'})}{|C_k| \cdot |C_{k'}|}$. This method strikes a balance between single and complete linkage, resulting in more balanced clusters. Average linkage demonstrates robustness to noise and outliers, and it is often effective in handling clusters of varying shapes and sizes. However, it can be susceptible to the "pendant" effect, where clusters with small inter-cluster distances may be erroneously merged.

Ward's method, introduced by Ward (1963), presents a criterion for merging two clusters based on minimizing the change in the total within-cluster sum of squared distances. It aims to minimize the objective function, $\Sigma_k I(C_k)$. Here, $I(C_k) = \sum_{Y_q \in C_k} ||Y_q - \overline{Y}_{C_k}||^2$ and $\overline{Y}_{C_k}$ represents the mean of the variables within cluster C_k. According to Murtagh & Legendre (2014), there are two versions of the objective function for Ward's method. The first version, often denoted as `Ward.D`, uses the square root of the dissimilarities. This was the default option for Ward's method in R prior to version 3.03. The second version, referred to as `Ward.D2`, utilizes the squared dissimilarities. The `Ward.D2` objective function is considered preferable as it maintains the same objective function, $I(C_k)$, outlined by Ward (1963).

Interestingly, Murtagh & Legendre (2014) demonstrated that `Ward.D` can yield the same results as `Ward.D2` if the distances input into the objective function calculation for `Ward.D` are squared. Ward's method operates under the assumption that the variables can be represented in a Euclidean space, enabling a geometric interpretation of the clustering results. The clusters produced are often spherical and of comparable sizes (Landau et al., 2011), however, the method is prone to the influence of outliers.

There are multiple packages and functions available in R for performing agglomerative hierarchical clustering. The base R package, `stats`, includes the `hclust` function, which supports the various linkage methods discussed and other options like Mcquitty, median, and centroid linkages. Another package, `flashClust` (Langfelder & Horvath, 2012), offers a fast and efficient alternative to `hclust`, specifically designed for large datasets. This package is well-suited for situations that require computational efficiency. The `cluster` package (Maechler et al., 2022) provides additional clustering algorithms and functionalities for hierarchical clustering. An example is `agnes`, which offers the agglomerative nesting clustering algorithm and introduces a visualization called a banner.

In the banner display, white spaces indicate stages where variables are still in their own cluster. In the agglomerative banner display (Figure 2.9), the white space is on the left, while in the divisive banner display (Figure 2.10), the white space is on the right. Each red bar in the banner represents the stages or heights at which two variables merge, and the two ticks on either end of the bar indicate those variables. For example, total cholesterol and low-density lipoprotein cholesterol are the first to be merged at height, 0.5. Banner plots provide similar insights as the dendrogram but offer a more legible visualization, especially when there are many variables to be clustered. It also creates a multi-modal plot that gives a sense of the number of clusters that might be present within the variables.

The `dendextend` (Galili, 2015) package is a versatile tool for manipulating, visualizing, and comparing dendrograms. It provides functions for cutting, rotating, reordering, and

pruning dendrogram branches, allowing users to customize the structure. The package also supports the evaluation and comparison of different clustering solutions. For visualization, it offers options such as color-coding branches, adding labels, and highlighting specific clusters. Furthermore, it integrates with other R packages like `ggplot2` and `heatmaply`, expanding its capabilities. The package is also extensible, allowing users to develop custom functions. Most of the dendrograms created in this dissertation use a combination of `dendextend` tools and `ggplot2` to create custom dendrograms that display hierarchically adjusted p-values.

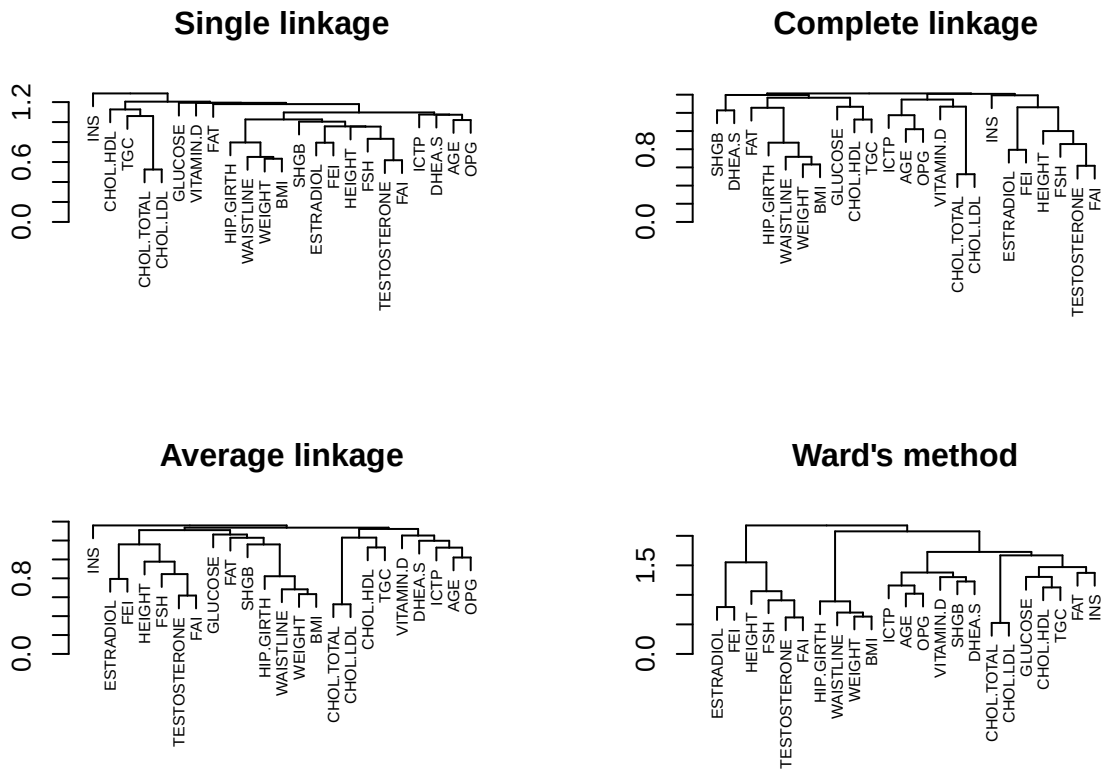


Figure 2.8: Agglomerative hierarchical clustering on the standardized attributes using the hclust function and four types of linkages: single, complete, average, and Ward's.

2.4.1.2 Divisive Clustering While many hierarchical clustering methods are agglomerative techniques, Divisive Analysis Clustering (diana), detailed in Chapter 6 of Kaufman &

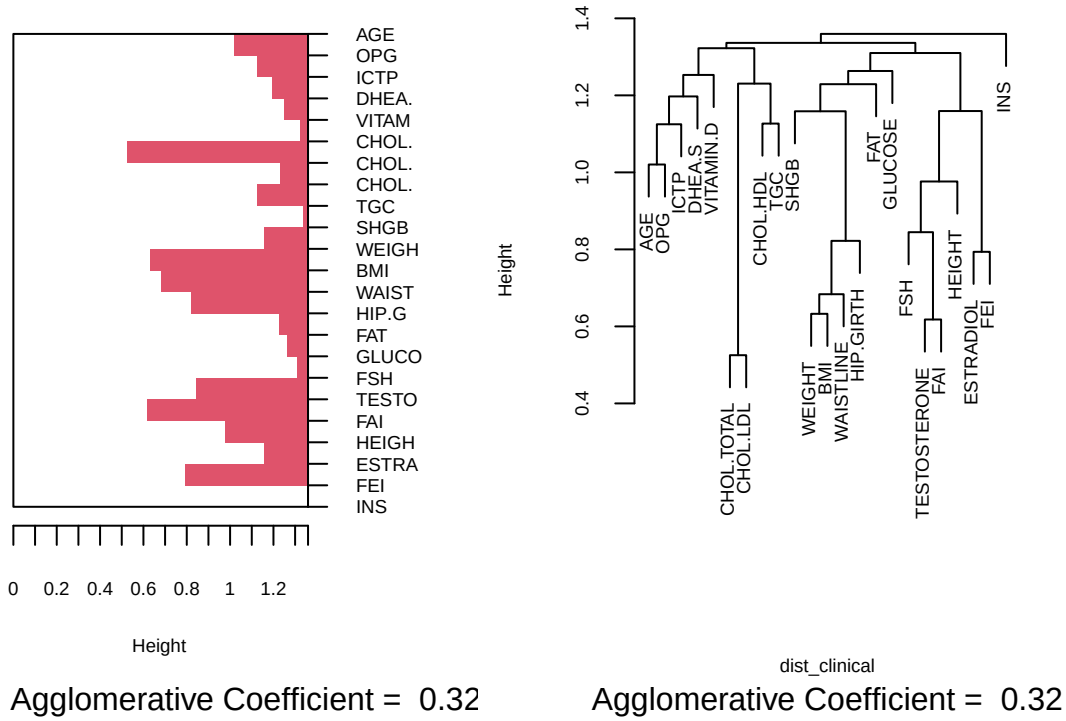


Figure 2.9: Agglomerative hierarchical clustering on the standardized attributes using the agnes function from the cluster package.

Rousseeuw (1990), employs a root to terminal node strategy. The algorithm commences by placing all Q variables into a single overarching cluster, which is then progressively partitioned at each iteration or hierarchy level until each variable ends alone in its own cluster. The cluster chosen for division at each iteration is the one with the widest spread, gauged by the maximum dissimilarity between any of its encompassing variables. The division process hinges on identifying the variable that, on average, is most dissimilar to its fellow members in the chosen cluster. This particular variable seeds the "splinter group", such that as the algorithm advances, it relocates variables to the "splinter group" if they exhibit more similarity to it than to the original cluster.

The `diana` function, available in the `cluster` package, facilitates divisive clustering. Similar to the `agnes` function from the same package, `diana` offers a divisive coefficient and a banner plot. The divisive coefficient is a value between 0 and 1 that captures the extent of the clustering structure identified. A divisive coefficient close to 1 suggests that there is a strong and clear clustering structure in the data, with the identified clusters being well-separated from each other. Conversely, a divisive coefficient close to 0 implies that the clusters are less distinct, indicating that the data might not inherently cluster well or that there are overlapping clusters. In the provided example, the divisive coefficient is 0.33, indicating a low to moderate level of clustering structure. This means the boundaries between these groupings are not very pronounced, making the clusters less distinguishable.

The banner plot visualizes the clustering solution. For instance, in the example plot in Figure 2.10, variables like `AGE` and `DHEA.S` appear to split from the main cluster at a higher height, suggesting they diverge earlier in the clustering process. In contrast, variables such as `GLUCOSE` and `FAT` remain part of the main cluster until lower heights, implying they are separated later. The white space on the plot indicates where each variable has ultimately been allocated to its individual cluster.

Monothetic clustering is another divisive hierarchical clustering algorithm that splits clusters based on the variable values thus making it not applicable to clustering variables (Chavent, 1998; Tran et al., 2021).

2.4.1.3 Considerations for Hierarchical Clustering Hierarchical clustering offers advantages such as its ability to provide a cluster solution without a pre-specified input of the number of clusters from the user and its ability to accommodate clusters of different sizes and shapes but it also comes with certain limitations. One limitation is its computational cost, particularly when applied to clustering a large number of features, which can make it time-consuming and resource-intensive. Additionally, interpreting the output of hierarchical

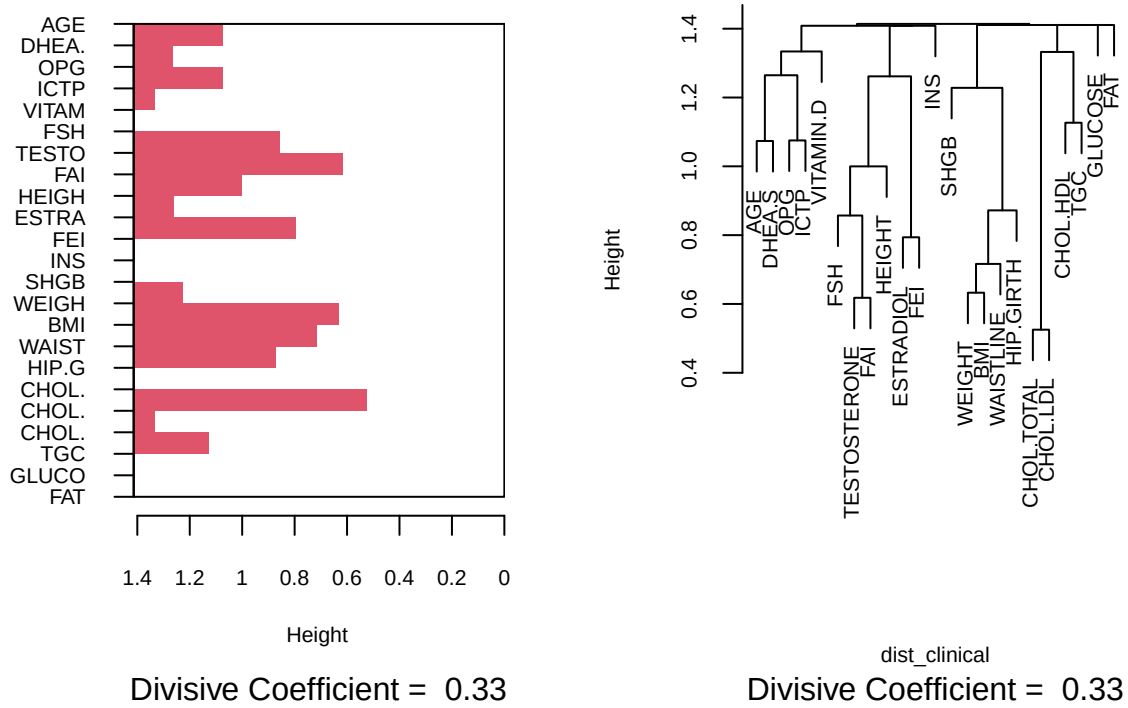


Figure 2.10: Divisive hierarchical clustering on the the standardized attributes using the diana function from the cluster package.

clustering can be challenging, especially when dealing with a large number of clusters. It is crucial to carefully assess and validate the clustering results to ensure their meaningfulness and utility. Another constraint of hierarchical clustering is that it does not allow for dynamic changes in the fusion or division decisions made during the algorithm. Once a cluster is formed, it remains fixed throughout the clustering process.

2.4.2 Partitioning (Optimization) Clustering

Partitioning-based clustering methods are a class of clustering algorithms that aim to divide the set of variables into a pre-specified number non-overlapping groups. Correlation-based partitioning methods of this kind iteratively divide variables into separate groups

based on the correlation coefficient. The objective of these methods is to minimize the sum of squared distances between the objects and their associated cluster centroids, medoids, or modes. The algorithms stop when convergence is reached, meaning that the cluster assignments no longer change.

2.4.2.1 K-means Clustering The k-means algorithm (MacQueen et al., 1967) is one of the most widely used partitioning methods for clustering, designed to divide objects into a predetermined number of clusters. This iterative algorithm updates cluster centroids and assigns objects to the nearest centroid until convergence is achieved. Traditionally, the k-means algorithm aims to partition observations into K pre-specified clusters by minimizing the squared Euclidean distances between each point and the mean of the points within each cluster. This objective function bears resemblance to that of Ward's method, which is commonly used in hierarchical cluster analysis.

However, applying the traditional k-means algorithm to cluster variables poses certain limitations due to its reliance on the Euclidean distance measure. The issue lies in the necessity of having the data points as inputs into the algorithm to calculate the centroids. Consequently, many existing methods and algorithms, including numerous functions in R, assume that the points are observations, leading them to primarily cluster the observations rather than the variables. Some researchers, as observed in the work of Hu et al. (2010), have sought to overcome this issue by transposing the data matrix and employing it as input for clustering. Despite this attempt, the approach has drawbacks. With transposed data, the Euclidean distances are still calculated, which, as discussed in Chapter 2.3, is often unsuitable as a dissimilarity measure for clustering variables.

A potential workaround involves converting dissimilarities into data points that can be represented in a Euclidean space. Multidimensional scaling (MDS) has emerged as a reliable method to achieve this transformation, generating points in a reduced-dimension

space that attempt preserve Euclidean distances. To retain the data structure effectively and avoid unnecessary dimension reduction, it is often advised to use the largest possible number of dimensions in the reduced space. The advantages of this strategy have been extensively explored in the work of Vera & Macías (2021), where the authors demonstrate that by using the points from multidimensional scaling, clustering results similar to performing traditional k-means on a data matrix can be achieved.

Relational k-means, introduced by Szalkai (2013b), offers another alternative to the traditional k-means clustering method, specifically designed to handle non-Euclidean distance scenarios. While conventional k-means requires actual data points as inputs, relational k-means can work with an arbitrary distance matrix. The algorithm refines an initial arbitrary cluster solution through iterative object reassignments to clusters. Each iteration involves reassigning objects to the cluster that minimizes the *clustering value* until convergence. This clustering value is determined by the sum of squared centroid distances of all variables that are in the assigned cluster. Mathematically, the clustering value is given by $\sum_{q=1}^{Q} u_{qk}$, where $u_{qk} = -\frac{1}{2} v_{qk}^{\intercal} \mathcal{D} v_{qk}$ represents the squared centroid distance. Here, $v_{qk'} = \sum_{q' \in C_k} \frac{e_{q'} - e_q}{|C_k|}$, where $\mathcal{D}$ is the $\mathbb{R}^{p \times p}$ distance or dissimilarity matrix, e_q is the qth standard basis vector, and $|C_k|$ is the cluster size. Importantly, if the squared centroid distances represent Euclidean distances, the objective function reduces to that of traditional k-means. Szalkai (2013a) provide an implementation of their proposed approach in C#. Presently, there is no known R package that implements this method, so this method is not explored further, but it provides an interesting avenue for future variable clustering using the modified correlation dissimilarity.

2.4.2.2 Partitioning Around Medoids Kaufman & Rousseeuw (1990) have proposed the use of actual objects, in this case variables, as representatives of clusters in the partitioning algorithm. In partitioning around medoids (PAM) clustering, a predetermined number of clusters, K, is required as input into the algorithm. The algorithm begins by selecting

an initial set of K variables also called medoids, with the aim of minimizing the average or sum of distances to other variables. Other variables are then considered as medoids iteratively until no further reduction in average or sum of distances can be achieved. This approach differs from the k-means algorithm, which uses centroids or mean values of points as representatives. By using actual variables as medoids, PAM exhibits increased robustness to outliers. Furthermore, it allows for the incorporation of dissimilarity measures beyond Euclidean distance, such as the correlation-based dissimilarity discussed previously which makes it suitable for variable clustering.

The `pam` function from the `cluster` package can be used to implement the partitioning around medoids (PAM) algorithm in R. The `pam` function performs k-medoids clustering and outputs the corresponding cluster assignments. For instance, Figure 2.11 illustrates the cluster solutions when the value of k is set to 4. These cluster solutions can be visualized by using colors in the MDS plot shown in Figure 2.3 to provide a visual representation of the clustering results.

2.4.2.3 Affinity Propagation The affinity propagation (Frey & Dueck, 2007) algorithm partitions objects into clusters based on their pairwise similarities. The algorithm identifies an initial object or *exemplar* for each cluster, similar to the idea behind PAM, and aims to find the most representative exemplars. However, unlike other partitioning clustering algorithms like k-means or PAM, Affinity Propagation (AP) does not require a predetermined number of clusters. It automatically determines the number of clusters through its message-passing algorithm, which begins by initializing two null matrices: the responsibility and availability matrices. The responsibility matrix contains values that quantify how suitable an object is to be an exemplar for another compared to others, and the availability matrix values indicate how much an object would prefer another to be its exemplar considering other options.

In each iteration, objects exchange messages, with each object sending responsibility

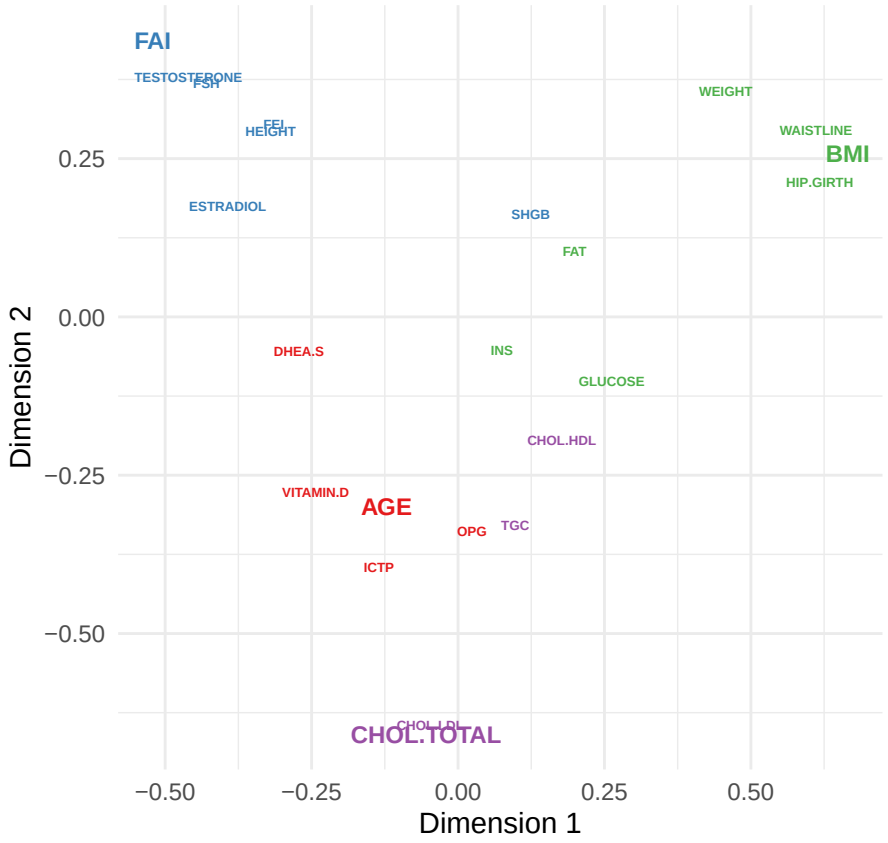


Figure 2.11: A four cluster solution using partioning around medoids superimposed on a 2-dimensional MDS plot. Medoids are indicated with a larger-sized text.

messages to others and availability messages to itself. The responsibility value, $l(q, q')$ is updated to $s(q, q') - \max_{q \neq q''} (a(q, q'') + s(q, q''))$ where $s(q, q')$ is the similarity value between variable Y_q and $Y_{q'}$, and $a(q, q')$ is their availability value. The availability value is updated to $\min(0, l(q, q') + \sum_{q'' \neq q, q'} \max(0, r(q', q''))$ on the off-diagonals and $\sum_{q \neq q'} \max(0, r(q, q'))$ on the diagonals. The iterative process continues until either the availability and responsibility values don't change, or a predetermined number of iterations is reached.

Exemplars for each cluster are identified based on the final values in the responsibility and availability matrices, and the number of clusters is determined by the number of exemplars. Each object is then assigned to the nearest exemplar to form the clusters. This automatic

determination of the number of clusters makes AP a valuable tool, as it allows the underlying structure and similarities between objects to dictate the clustering result without the need for pre-defining the number of clusters. The method is also robust to noise and is effective in handling irregularly shaped clusters. However, it may still need some tuning in the number of iterations if it produces fewer or more clusters than we find optimal. It may also suffer from high computational complexity, particularly for datasets with a large number of variables, due to the iterative process involving several large matrices.

In R, the `FCPS` package provides the function `APclustering`, which implements affinity propagation clustering. It takes input either as a matrix of pairwise dissimilarities between data points or the data matrix itself and transforms this into a similarity matrix. If supplied dissimilarities, the similarity matrix is calculated using $-(\mathcal{D}^2)$, where $\mathcal{D}$ is the dissimilarity matrix. If supplied a data matrix, it uses the Euclidean distance to calculate the dissimilarities before applying the same formula to transform them into similarities. The function returns the variables with their respective cluster memberships and the exemplars for each cluster. Figure 2.12 illustrates the cluster membership of each variable and the corresponding exemplars for the dataset from the clinical study. In this specific example, the clustering algorithm produces a five-cluster solution, as shown by color in the plot, and the exemplars are indicated with a larger sized text. Overall, we observe that the results are not very different from what was observed using PAM.

2.4.3 Hybrid Methods

In the literature, it is common to encounter new clustering techniques that combine or integrate established methods. This combination often leverages the advantages of one technique to offset the limitations of another, ensuring each compensates for where the other falls short. A well-known example is the integration of hierarchical clustering and the k-means algorithm.

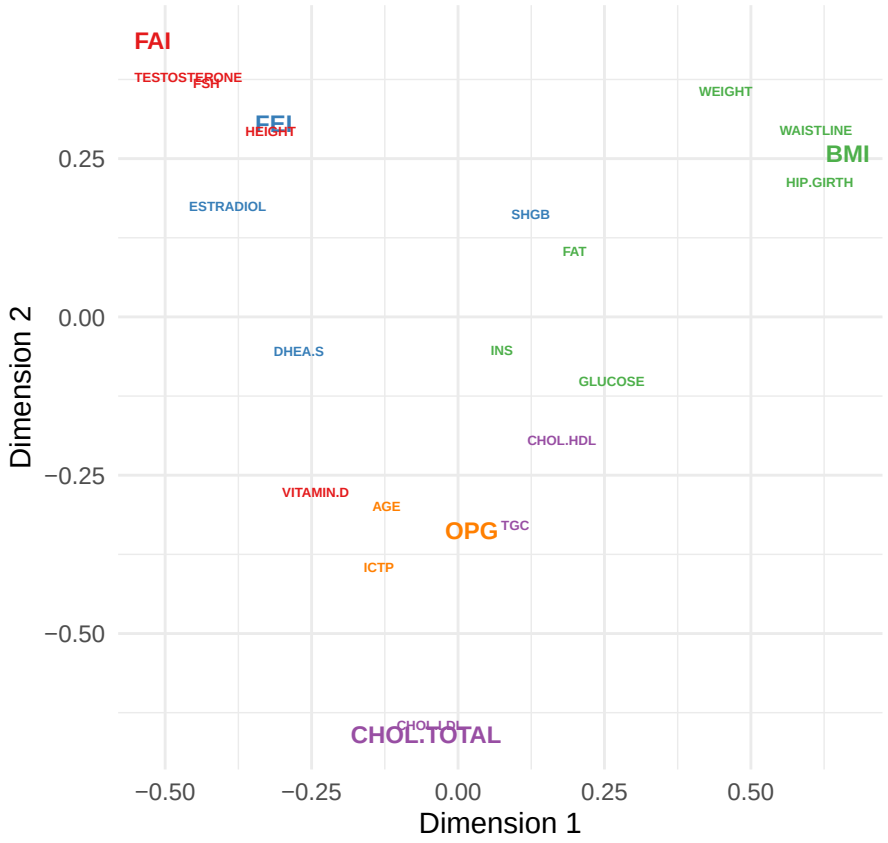


Figure 2.12: Affinity propagation clustering on the clinical data produces five clusters. Exemplars are indicated with larger-sized text, and clusters are indicated by color.

One limitation of hierarchical clustering lies in its rigidity, as it does not allow the reassignment of variables to a cluster once allocated. This is not the case with k-means, which possesses the flexibility to iterate cluster assignments. However, k-means is notably sensitive to the initial selection of cluster centers (Lu et al., 2008), potentially leading to a local minimum of the objective function. In cases where the hierarchical structure is not the priority, hierarchical clustering results can inform the initial set of centroids needed for the k-means algorithm. This integration enhances the k-means outcomes, making it easier to find initial cluster centers compared to a random choice. However, because both algorithms work in tandem this approach could be seen as excessive for a possible enhancement of cluster

results. This integration also doubles the number of parameters to consider and supply, including the dissimilarity and linkage method, along with a predetermined number of desired clusters.

Alternate approaches have been suggested that do not necessitate running both algorithms entirely but integrate one method with another. One such example is the hierarchical affinity propagation method proposed by Givoni et al. (2012). In this algorithm, the authors extend the principles of affinity propagation to hierarchical clustering using the message passing method of traditional affinity propagation. By considering all points as potential exemplars at different layers of the hierarchy and propagating information up and down the levels, the algorithm learns the relationships between clusters at various levels. This learning enables the identification of exemplars at each layer of the hierarchy, enhancing efficiency and robustness against outliers. Other notable examples of integrated methods include hierarchical k-means (Schubert, 2021), and hierarchical agglomerative clustering around medoids (Lamrous & Taileb, 2006).

To employ a hybrid approach for variable clustering, each clustering technique involved should not only be capable of clustering variables independently but also be compatible in terms of using the same form of dissimilarities.

2.5 Discussion

This review has outlined various graphical methods to visualize clustered variables, examined techniques to determine the proximity between two variables, and explored hierarchical and partitioning clustering methods to group similar variables together based on their correlations. A central insight permeates this review: no single solution is universally applicable. Each visualization, proximity measure, or clustering method offers its own unique advantages and disadvantages. The success of clustering is heavily contingent upon the chosen method, the specific data at hand, and the objectives of clustering. This underscores

the necessity for the evaluation of these factors to select appropriate clustering techniques to ensure meaningful results.

In that regard, and considering the general goals of variable clustering – dimension reduction or understanding the dependency structure among variables – it appears that the key focus shifts to finding variables that share similar information, rather than ones that have just similar directions or similar magnitudes. This shift entails using dissimilarities tailored for this purpose, a requirement that existing methods, including the default ones in R like the `k-means` algorithm, might not meet due to their dependence on Euclidean distance. The exploration of dissimilarities in this review highlights the limitations and possible inadequacies of employing Euclidean distances for variable clustering, and suggests alternatives, such as employing dissimilarities based on the absolute values of the correlation coefficients between variables, among other comparable strategies.

It is also essential that most traditional algorithms are designed with the clustering of observations in mind. A comprehensive understanding of the functions available in R are paramount to avoid their incorrect application when clustering variables. We advocate for the selection of dissimilarity methods that do not depend on the magnitudes of values for determining clusters. Essentially, any hierarchical or partitioning clustering method in R that permits the use of custom dissimilarity matrices can be effectively employed for variable clustering, given the method's compatibility with dissimilarity matrices centered on shared information rather than sizes. If a method exclusively accepts data, ensure the underlying algorithm employs suitable proximity methods for variable clustering.

CHAPTER THREE

A CORRELATION-BASED HIERARCHICAL MULTIPLE TESTING PROCEDURE

Contribution of Authors and Co-authors

Author: Priscilla Bacino

Contributions: Responsible for majority of the writing

Co-Author: Dr. Mark Greenwood

Contributions: Provided feedback on statistical analysis and drafts of the manuscripts.

Manuscript Information Page

Priscilla Bacino, Mark C. Greenwood

Status of Manuscript:
___x___ Prepared for submission to a peer-reviewed journal
_____ Officially submitted to a peer-reviewed journal
_____ Accepted by a peer-reviewed journal
_____ Published in a peer-reviewed journal

Abstract

Testing many hypotheses simultaneously without adjusting evidence for the array of tests considered increases the chance of false discoveries. Traditional methods for controlling the rate of false discoveries aim to obtain adjusted evidence from each hypothesis individually. This can lead to diminished power in large-scale testing situations, especially when the single effects may be too weak to be identified individually. A combination of signal identification of hypotheses and signal detection in the intersection of those individual hypotheses has the potential to improve statistical power and provide enhanced interpretation of results. An efficient way to test intersection hypotheses and individual hypotheses together is to begin testing with the global null hypothesis of no real effect, then proceed in a hierarchical manner to test smaller and smaller partitions until no more evidence is found or individual hypotheses are encountered. We consider a data-driven hierarchical multiple testing method of the manner described above. The method was initially proposed by Meinshausen (2008) for selecting correlated predictor variables in linear regression. We focus on applying the method to testing for multiple inter-group differences under the two-group model, motivated by a study of non-obese mice that assesses the response of plasma metabolites on Type I Diabetes Mellitus disease progression. The properties of the multiple testing procedure under marginal associations are also studied theoretically and using simulated data. Simulation and theoretical results show that the method provides family-wise error control. We also see in simulations that there is improvement in performance even at the individual variable level compared to traditional alternative methods.

3.1 Introduction

Advancements in biological research have led to studies that involve a large number of simultaneous hypothesis tests. For instance, researchers may examine thousands of metabolites to determine if their mean abundances vary across different experimental treatments (Barko & Williams, 2021; Ganguly, 2021; Hernandez et al., 2020). Similarly, in the field of neuroimaging, scientists may seek to identify the specific voxels in a brain image, numbering in the hundreds, thousands, or even millions, that are associated with particular tasks or activities (Kunas et al., 2022; Liu et al., 2022).

When multiple tests are performed simultaneously, there is an increased chance of making false discoveries. In statistics, a false discovery occurs when a test provides strong

evidence against the null hypothesis by random chance, even though the null hypothesis is true. Traditional methods, including the Bonferroni procedure (Dunn, 1961), and Bonferroni-Holm procedure (Holm, 1979), have been proposed to address this issue, and they are successful in guaranteeing that the probability of making at least one false discovery (also known as the family-wise error rate or FWER) does not exceed a pre-specified rate. However, they tend to be conservative, treat all tests as independent of one another, and do not provide a comprehensive story about what we can learn from the hypotheses.

In biological studies, the hypotheses tested are often related because the underlying variables are correlated. This could be due to genes or proteins corresponding to the same pathway, voxels being in close proximity, or metabolites functioning jointly in some biological process. In all these cases, there is an opportunity to increase statistical power and enhance the interpretation of results if the structure of dependence among the hypotheses is incorporated (Goeman & Finos, 2012; Hu et al., 2010). In this paper, we demonstrate the benefits of incorporating structural information with a correlation-based hierarchical approach to multiple testing.

Consider a study conducted to investigate the effects of Type 1 Diabetes Mellitus on the plasma metabolome, lipidome, and signaling lipids of non-obese diabetic (NOD) mice (Fahrmann et al., 2015). In the study, researchers measured blood glucose levels of 71 NOD mice at the time of their death, classifying them as either hyperglycemic or normoglycemic. Mice were diagnosed as hyperglycemic if their fasting (4hr) blood glucose levels at the time of sacrifice were at least 250 mg/dL; otherwise, they were labeled as normoglycemic. The study included 31 hyperglycemic and 40 normoglycemic mice. All samples were collected in a single batch. The researchers filtered the results to exclude any noisy or inconsistent peaks, making determinations based on criteria such as mass spectral matching, spectral purity, signal-to-noise ratio, and retention time. Only metabolites found in at least half of the samples were considered. For any missing values, they were inferred by examining the

extracted ion traces from the raw data after local background noise subtraction.

In total, for each mouse examined, the abundances of 181 known metabolites were quantified using gas chromatography time-of-flight (GC-TOF) mass spectrometry. These metabolite measurements were log-transformed and scaled to each have a mean of 0 and a standard deviation of 1. The primary objective is to determine if there is a difference in the true mean abundances between hyperglycemic and normoglycemic mice across the 181 metabolites.

A typical approach to address the research question of interest involves formulating individual hypotheses, also referred to as elementary hypotheses, for each metabolite to test the difference in true means. Equation (3.1) shows the null and alternative hypotheses for metabolite q, where $q = 1, \ldots, 181$. $\mathcal{H}$ and $\bar{\mathcal{H}}$ represent the null and alternative hypotheses, respectively, while $\mu_{hyper,q}$ and $\mu_{normo,q}$ denote the true mean abundances of hyperglycemic and normoglycemic mice for metabolite q.

$$\begin{aligned} \mathcal{H}_q &: \mu_{hyper,q} = \mu_{normo,q} \\ \bar{\mathcal{H}}_q &: \mu_{hyper,q} \neq \mu_{normo,q}. \end{aligned} \tag{3.1}$$

The results displayed in Figure 3.1 depict a volcano plot showcasing the $-\log_{10}$ adjusted p-values plotted against the $\log_2$ fold change between the group sample means for each metabolite. The $\log_2$ fold change is calculated as the difference between the mean expression levels of the two groups. The p-values are obtained by conducting multiple independent 2-sample Welch's t-tests (Welch, 1947) and are adjusted using the Bonferroni adjustment of $\min(181p_q, 1)$, where p_q is the p-value for the qth test, $q = 1, \ldots, 181$. While this approach identifies metabolites with detectable differences, it fails to provide insights into any relationships between hypotheses. Uncovering these relationships can be crucial, for example, in constructing a network to comprehend the molecular mechanisms associated with the development of a disease. Researchers often perform a separate correlation analysis to explore

connections among all variables or among just those variables with detected differences. However, we propose incorporating the structure of dependency into the multiple testing procedure. This can not only contribute a narrative to the results but also enhance the power of the testing procedure.

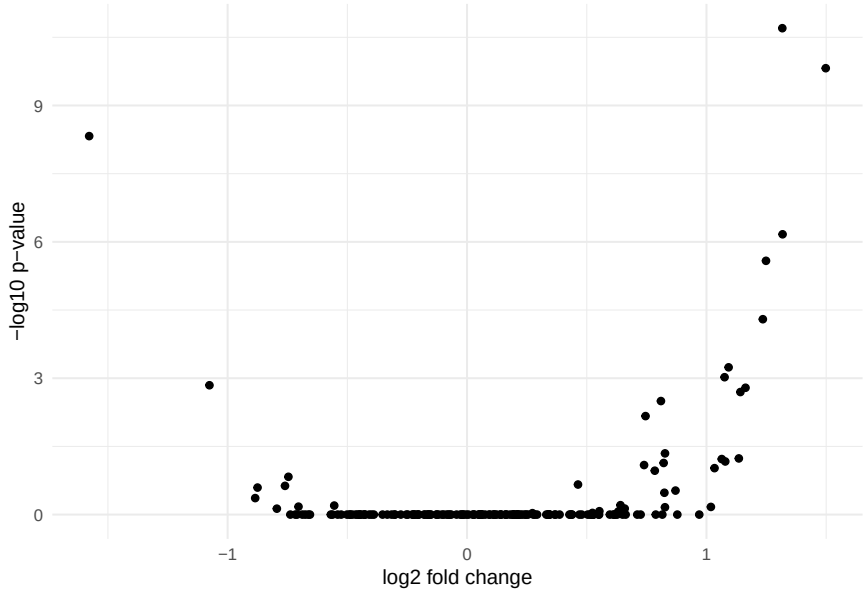


Figure 3.1: Volcano plot of the mice data. Results with a small p-value (large -log10 p-value) and large fold change are often considered most interesting to researchers.

We propose a hierarchical multiple testing approach involving K hypotheses arranged in a tree. Here, the testing starts at the top with the global null hypothesis, $H_{k=1}$, which assumes no difference between the mean abundances of the two groups of mice across all metabolites. Subsequently, if strong evidence emerges against the global null hypothesis, H_1, a null hypothesis for a smaller subset of metabolites is tested. This testing process continues until limited evidence against the null hypothesis is found for a particular subset of metabolites, prompting exploration to cease for that group. Alternatively, testing may reach the terminal node, focusing on group differences for a single metabolite. Although Meinshausen (2008) initially introduced this method for variable selection in regression, in this

paper, we tailor it to evaluate multiple inter-group differences and discuss the considerations required to apply the method to this new scenario.

The structure of this paper is as follows: In Section 3.2, we introduce the hierarchical testing procedure. We demonstrate that the method controls the family-wise error rate and we explain how the interrelationships among the K hypotheses contribute to enhanced power. In Section 3.3, we present two simulation studies. First, we evaluate the influence of the hierarchical structure on both the family-wise error rate (FWER) and power. Subsequently, we validate that the proposed approach maintains control over the FWER and juxtapose its efficacy against existing methods, specifically for elementary hypotheses or those at the terminal node. Our continued analysis of the mice dataset is detailed in Section 3.4. Lastly, in Section 3.5, we discuss our findings, delve into potential future extensions, and offer concluding remarks.

3.2 The Hierarchical Testing Procedure

The hierarchical testing procedure follows a top-down approach to test nested hypotheses. This method comprises three key components. The first step involves creating a hierarchy of nested tests, represented as a tree structure, where each node corresponds to a single hypothesis. Nodes have at most one parent and represent the intersection of their mutually exclusive children. For example, in a scenario with five response variables ($Q = 5$) and nine hypothesis tests ($K = 9$), refer to Figure 3.2. The second component entails conducting tests at each node of the hierarchy to calculate p-values. Finally, the third component involves applying a resolution-dependent adjustment to control the Family-Wise Error Rate (FWER) simultaneously at all nodes of the hierarchy. In this chapter, we will delve into these three components of the procedure.

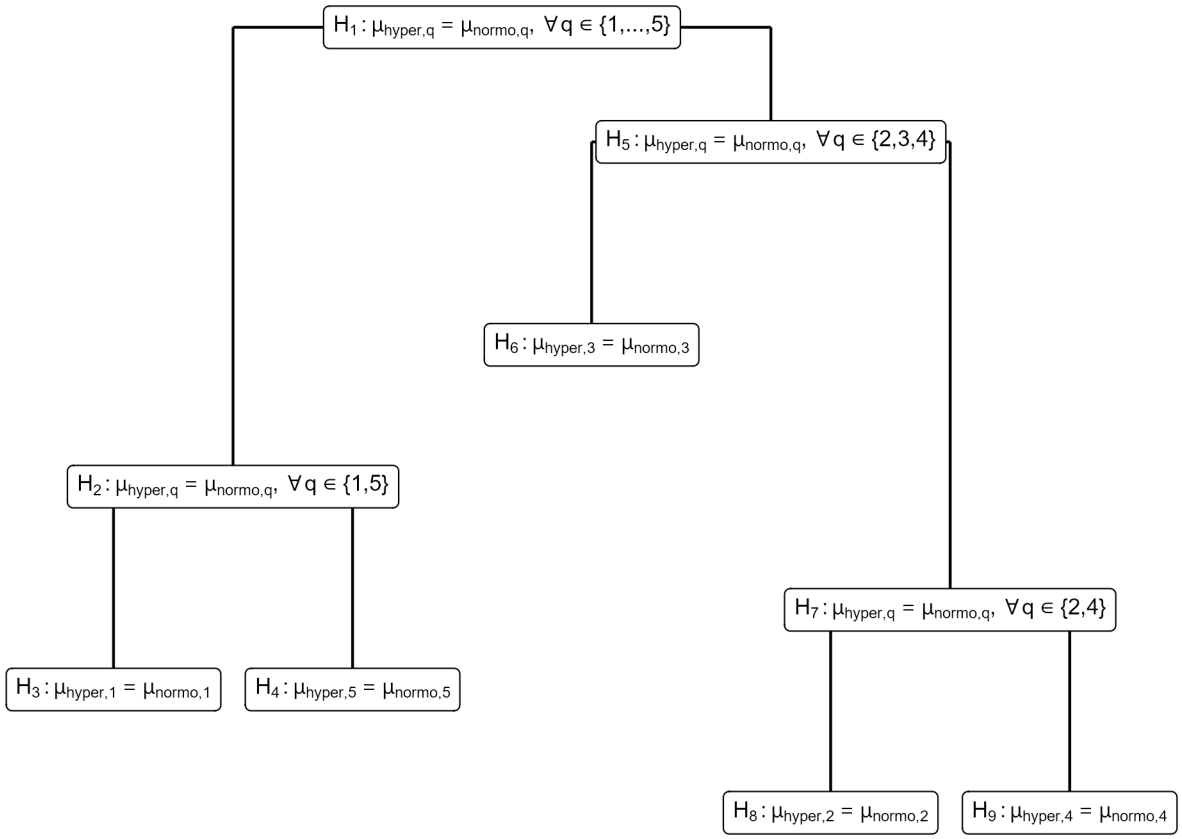


Figure 3.2: An example of a hierarchical testing structure for five metabolites represented in a tree. The root node contains the global null hypothesis testing for any group differences across the five metabolites and the hypotheses at the terminal nodes test for differences on single metabolites. The children of each node are mutually exclusive and have only one parent.

3.2.1 The Hierarchy

Incorporating structural information such as hierarchies with logical relationships into multiple testing procedures has become central to improving signal identification and interpretability (Benjamini & Bogomolov, 2014; Efron, 2007; Goeman & Finos, 2012; Nandi et al., 2021; Sun & Cai, 2009). Generally, hierarchies of the nature illustrated in Figure 3.2 may originate from two sources; from some external source or from the data.

Externally sourced hierarchies are often specified prior to data collection using domain

expertise or knowledge gained from a separate independent experiment. An example of a domain-specified hierarchy can be found in the study by Sankaran & Holmes (2014), where evolutionary trees based on microbial taxonomic families are used to test the effect of different environments on microbial abundance. On the other hand, data-generated hierarchies may be specified using inherent characteristics of the data or the underlying experiment. For instance, in the same study by Sankaran & Holmes (2014), global temperature data collected over time were divided to create a hierarchy of different resolutions. These resolutions ranged from the entire span of time covered by the dataset (1880-2009) to partitions containing decades of the temperature data.

When the data lack natural groupings or distinctive characteristics, or when researchers are uncertain about defining groups based on scientific knowledge, hierarchical clustering techniques (Hartigan, 1975) can be employed. Hierarchical clustering algorithms are a class of clustering methods that iteratively merge or divide variables to form hierarchical groups. For this study, we will utilize agglomerative hierarchical clustering to construct our hierarchies. This approach starts with each variable in its own cluster and then iteratively merges the most similar clusters until all variables are in the same cluster. The cluster solution obtained forms groups that correspond to the K hypotheses, indicating which group, denoted as C_k, is being tested for inter-group differences in the set of metabolites belonging to that cluster. In other words, the hypothesis H_k tests for inter-group differences among the metabolites in cluster C_k.

In hierarchical clustering, the choice of linkage criterion, which measures the distance between two clusters, and the pairwise distances between variables or metabolites, influences how these groups are formed. To calculate distances among metabolites and capture the dependency structure of the data in the hierarchy and multiple testing procedure, we will use their absolute pairwise sample correlations. The distance $d(Y_q, Y_{q'})$ between two variables Y_q and $Y_{q'}$ is given by $\sqrt{2(1 - |r(Y_q, Y_{q'})|)}$ (James et al., 2013), where $r(Y_q, Y_{q'})$ represents the

Spearman sample correlation coefficient. The choice of Spearman correlation coefficient is particularly beneficial in handling outliers and capturing monotonic relationships between variables that may not be linear. Also, the use of the absolute value ensures that variables are grouped based on the magnitude of correlation rather than the direction of the relationship.

In Section 3.3, we will compare and discuss three commonly used linkages in agglomerative hierarchical cluster analysis: Ward's, Complete, and Single linkages. This comparison aims to assess how different linkage choices and hierarchical structures influence error rates and statistical power of the hierarchical testing procedure.

3.2.2 Performing Cluster-Wide Tests

At each node, k, in the tree, a test is performed to assess whether any of the response variables in cluster, C_k, are associated with the experimental groups. These cluster-wide tests are often referred to as overall tests, and there are generally two ways to conduct such tests. One approach conducts tests based on the joint model containing all variables. Traditional parametric approaches include the overall or partial F-test (used by Meinshausen (2008) in the regression setting), Hotelling's T^2 test (Hotelling, 1931), and multivariate analysis of variance (MANOVA) tests such as Pillai's trace or Wilks' lambda (Warne, 2014).

The other approach involves combining the test-statistics or p-values obtained from testing elementary hypotheses for those variables within that cluster into a single p-value. For example, the p-value for the overall test H_1 can be obtained by combining the p-value obtained from all elementary tests, $\mathcal{H}_1, \ldots, \mathcal{H}_Q$. Examples of methods that utilize this approach include the Bonferroni test (Dunn, 1961), Simes' test (Simes, 1986), the minimum p-value test (Tippett et al., 1931), and the higher criticism test (Donoho & Jin, 2004).

Generally, tests that combine elementary p-values or test statistics offer more flexibility in model choices and can be optimally powerful when compared to methods that use the joint distribution of the variables. Moreover, traditional parametric tests (such as Hotelling's

T^2 or Wilks' lambda) are constrained to $Q \leq n$, but many applications of interest in this context have $Q > n$. Therefore, for the remainder of this work, we will focus on reviewing p-value combining tests. We are particularly interested in tests that control the Type I error rate under some form of dependency, as they align with our goal of intentionally grouping variables based on their correlations. Additionally, we seek tests that perform well in scenarios with sparse signals, where less than half of the hypotheses are false nulls (Arias-Castro & Ying, 2019). Our aim is not to review all tests that fall under this criteria; rather, we provide a range of tests suitable for various data scenarios commonly encountered in large-scale multiple testing. In the following subsection, we will discuss the Bonferroni global test, the Generalized Higher Criticism test (Barnett et al., 2017), and the Generalized Berk-Jones test (Sun & Lin, 2020). These tests have been selected due to their relevance in different contexts and their ability to handle various types of dependencies and sparse signals.

Suppose we have valid elementary p-values, $p_1, \ldots, p_Q$, or elementary test statistics, $t_1, \ldots, t_Q$, and we wish to test the null and alternative hypotheses, H_k and $\bar{H}_k$, defined in Equation (3.2).

$$\begin{aligned} H_k &: \mu_{hyper,q} = \mu_{normo,q} \text{ for all } q : Y_q \in C_k \text{ versus} \\ \bar{H}_k &: \mu_{hyper,q} \neq \mu_{normo,q} \text{ for at least one } q : Y_q \in C_k. \end{aligned} \tag{3.2}$$

The Bonferroni combined p-value, denoted as $\pi_{k,\text{Bon-adj}} = \min\{Qp_{(1)}, 1\}$ where $p_{(1)}$ is the first-ordered elementary p-value, provides stronger evidence against H_k for values closer to 0. The Bonferroni global test is simple, and it controls the Type I error rate under any form of dependence among all $\mathcal{H}_q : Y_q \in C_k$ (Guo, 2009). However, for cases where the size of the cluster, $|C_k|$ is large, and especially cases when dealing with weak and sparse effects, the test is often conservative. This is because the Bonferroni method is unable to combine information from tests that show weak evidence against many null hypotheses. Thus, making it challenging to obtain an adjusted p-value smaller than the chosen α.

In situations where the non-nulls are sparse and weak, two alternative tests, the Higher Criticism test (HC) (Donoho & Jin, 2004) and the Berk-Jones test (Berk & Jones, 1979), can be considered. The HC statistic can be described as the supremum of a standardized empirical process under H_k, while the Berk-Jones test is the maximum of a group of one-sided likelihood-based statistics. However, the original HC and Berk-Jones tests have limitations for our specific use case because they assume that the variables are independent. Additionally, the HC test is known to perform poorly under finite samples (Li & Siegmund, 2015). To address these limitations, we focus on extended versions of both tests: the Generalized Higher Criticism (GHC) test (Barnett et al., 2017) and the Generalized Berk-Jones (GBJ) test (Sun & Lin, 2020). These extended versions provide the optimal properties of the HC and Berk-Jones tests but with more desirable finite sample properties, making them suitable for correlated variables and better suited for situations with sparse and weak non-null effects. By utilizing the GHC and GBJ tests, the hierarchical testing procedure can benefit from more robust statistical power, particularly at finer resolutions where the number of hypotheses in the cluster is small.

Given that the elementary test statistics under the null hypothesis are normal, that is, $t_1, \ldots, t_Q = Z_1, \ldots, Z_Q \sim MVN(0_{|C_k|\times 1}, \mathbf{\Sigma}_{|C_k|\times|C_k|})$, and $|Z|_{(q)}$ is the absolute value of the qth order statistics, we define $C_k(u) = \sum_{q:Y_q \in C_k} I(|Z_q| \geq u)$ to be the number of marginal test statistics greater or equal to some threshold $u \geq 0$, $\bar{\Phi}(.)$ as the inverse of the standard normal CDF, and $|C_k|$ as the cardinality of C_k. The GHC test statistic is given by

$$\tau_{k,GHC} = \max_{u>0} \left[\frac{C_k(u) - 2|C_k|\bar{\Phi}_k(u)}{\sqrt{\widehat{\text{var}}(C_k(u))}} \right], \tag{3.3}$$

and the GBJ test statistic is given by

$$\tau_{k,GBJ} = \max_{1 \leq q \leq \frac{|C_k|}{2}} \left(\log \left[\frac{\Pr\left\{ S(|Z|_{(|C_k|-q+1)}) = q \middle| E(\mathbf{Z}) = q/|C_k|, \text{cov}(\mathbf{Z}) = \mathbf{\Sigma}_{|C_k|\times|C_k|} \right\}}{\Pr\left\{ S(|Z|_{(|C_k|-q+1)}) = q \middle| E(\mathbf{Z}) = 0, \text{cov}(\mathbf{Z}) = \mathbf{\Sigma}_{|C_k|\times|C_k|} \right\}} \right] \right) \cdot I\left\{ 2\bar{\Phi}(|Z|_{(|C_k|-q+1)}) < \frac{q}{|C_k|} \right\}. \tag{3.4}$$

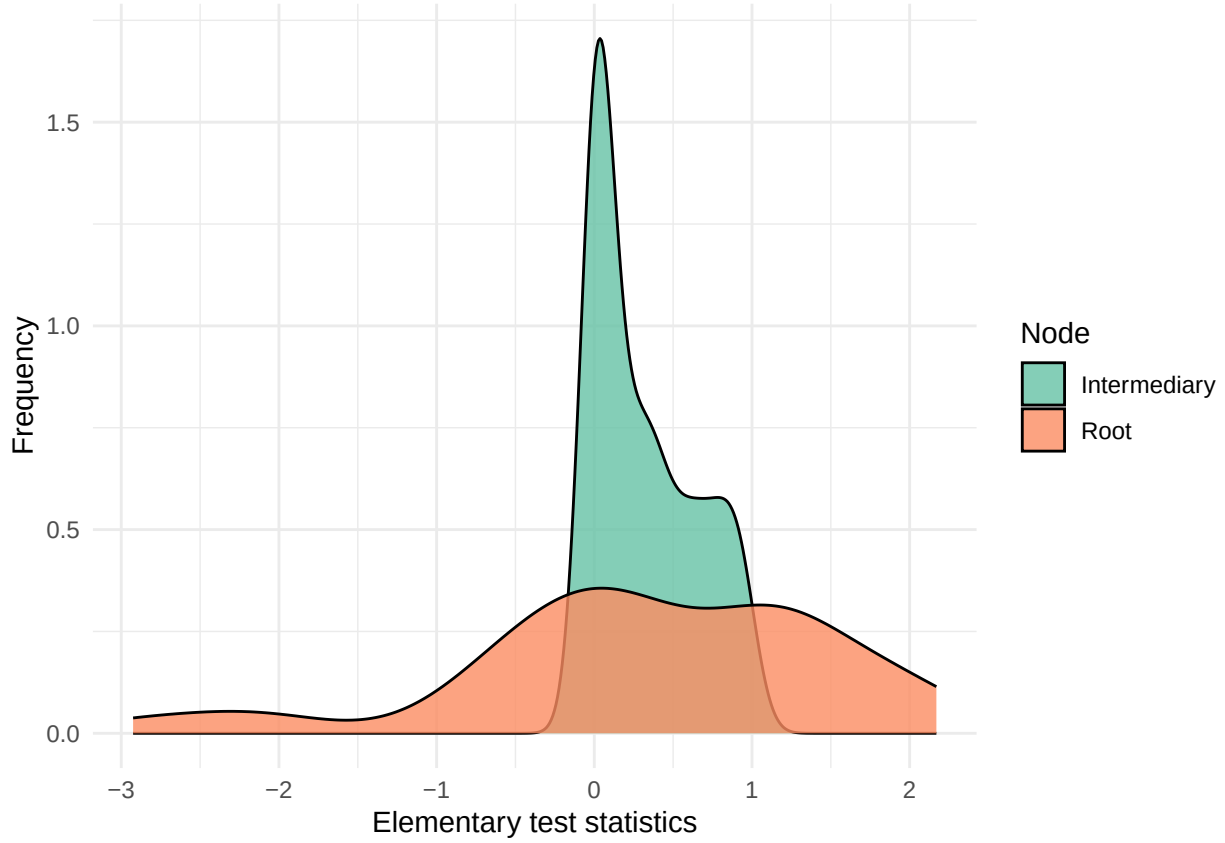


Figure 3.3: Two non-parametric density curves that show the distributions of elementary test statistics at the root node and at one of the intermediary nodes in the mice example. The root node comprises many large magnitude test statistics, while the intermediary node has only a few.

For example, consider Figure 3.3 that shows the distribution of elementary test statistics at the root node and at an intermediary node. We observe that, in general, the root node

contained many large magnitude test statistics, but the intermediary node had only a few extreme results. The combined p-value generated for the root node by GHC is tiny ($< .0001$) which signifies very strong evidence against the null hypothesis at H_1. In contrast, the GHC test generates a large p-value for the intermediary node, which is equal to 0.4838. Generally, larger magnitude elementary test statistics will produce smaller values of $\tau_{k,\text{GHC}}$ and $\tau_{k,\text{GBJ}}$ and provide more evidence against the null hypothesis H_k.

Both the GHC and GBJ tests extend their original versions by modeling $S(u)$ after beta-binomial-type distributions instead of a binomial distribution to accommodate correlated test statistics. In terms of performance, the GHC is better suited for extremely sparse situations, where the proportion of non-nulls is less than 0.25, while the GBJ is better suited for moderately sparse situations, where the proportion of non-nulls is between 0.25 and 0.5. We use the GHC and GBJ functions from the GBJ R package (Sun & Lin, 2020) to implement these two methods.

3.2.3 Adjusting Evidence from Cluster-Wide Tests for Multiplicity

Suppose K cluster-wide tests are conducted according to Equation (3.2) to obtain valid p-values, $\pi_1, \ldots, \pi_K$. At each node of the hierarchy, there is one p-value, indicating the evidence against the null hypothesis of no inter-group difference for metabolites in C_k. The clusters $C_1, \ldots, C_K$ satisfy the conditions $C_k \cap C_{k'} = \emptyset$ or $C_k \subset C_{k'}$ for some $k : k \neq k',\ k = 2, \ldots, K$. Note that C_1 is at the root node and contains all p-values. The numeration of the subscripts on C follows the depth-first search approach, which starts at the root node and sequentially assigns numbers down the left-most branch to its terminal nodes before moving to the next most adjacent node to repeat the downward assignment. This numeration approach ensures that in Algorithm 3.1, testing happens in a hierarchical manner, meaning that all ancestors of a node are tested before the node itself is tested. Here, an ancestor refers to any superset of set C_k. A tree structure with the depth-first search numeration is illustrated for C_k,

$k = 1, \ldots, 9$, in Figure 3.2.

Before describing the testing procedure, we will define some additional notation. Let par_k, desc_k, and si_k represent the parent, descendants, and the sibling(s) of C_k, respectively. Here, a parent cluster is defined as the ancestor with the lowest cardinality or size, descendants are any subsets of C_k, and C_k and si_k form a partition on par_k. Next, let

$$r_k = \begin{cases} \dfrac{Q}{|C_k| + |si_k|}, & \text{if si}_k \text{ is a terminal node} \\ \dfrac{Q}{|C_k|}, & \text{otherwise} \end{cases}, \tag{3.5}$$

where $|.|$ denotes the cardinality or size of the cluster. The summarized testing procedure is presented in Algorithm 3.1.

In algorithm 3.1, the p-value at node k is undergoes two adjustments; π_k is initially adjusted to account for the simultaneous execution of multiple cluster-wide tests to become π_k^{adj}, and then π_k^{adj} is adjusted hierarchically to $\pi_k^{\text{h-adj}}$ ensure hierarchy among the adjusted p-values. The adjusted p-value, denoted as π_k^{adj} , is determined by the formula $\pi_k r_k$, which is based on the Bonferroni adjustment. In the standard Bonferroni approach, each p-value is multiplied by the count of elementary hypotheses, which in this context equates to the number of response variables, Q; this is reflected in the numerator of the proposed adjustment. However, within each node of the hierarchy, we do not assess Q elementary hypotheses. Therefore, we modify the adjustment by dividing by the count of elementary hypotheses that are combined into a singular test. For instance, at the root node, H_1 incorporates all Q elementary hypotheses, permitting us to adjust the value to $Q/Q = 1$. This adjustment is logical, considering that at the root node, we would have only performed a singular test, obviating the need for multiplicity adjustments. Generally, this minimized adjustment translates to $\dfrac{Q}{|C_k|}$ (Meinshausen, 2008).

Following techniques proposed in Shaffer (1986) which leverage logical constraints

Algorithm 3.1: The hierarchical testing algorithm at level α

$\pi_k^{\text{h-adj}} = \texttt{NA}$, for all $k = 1, \ldots, K$

for $k = 1$ to K **do**

  if $\pi_k^{\text{h-adj}} = \texttt{NA}$ **then**

    Calculate:

      (1) Calculate π_k using any suitable cluster-wide test.

      (2) $\pi_k^{\text{adj}} = \pi_k r_k$

      (3) $\pi_k^{\text{h-adj}} = \max(\pi_k^{\text{adj}}, \pi_{\text{par}_k}^{\text{adj}})$

    if $\pi_k^{\text{h-adj}} \geq \alpha$ **then**

      $\pi_{\text{desc}_k}^{\text{adj}} = \pi_k^{\text{h-adj}}$

    end if

  else

    Skip k

  end if

end for

imposed by the hierarchical relationship, a reduction in this adjustment can be achieved by evolving the denominator $|C_k|$ to the piece-wise function, r_k, as presented in Equation (3.5). It is argued here that if node k has a terminal sibling, then the denominator can be reduced by one. The logical foundation for this is discussed below, and a mathematical proof is provided in Appendix A. Consider a scenario where H_1 is true, it follows that all other subsequent hypotheses $H_2, \ldots, H_K$ must also be true. To assess family wise error rate, which is the probability of incurring at least one Type I error in the tree, we can focus on H_1, because a Type I error anywhere in the tree implies a Type I error at H_1. This logical implication holds for any ancestor-descendant or parent-child relationship in the tree and

this means that we only have to check intermediate or terminal nodes for a Type I error only if they come from a generation of false hypotheses. Now, consider two sibling hypotheses, H_k and $H_{k'}$, who come from a generation of false hypotheses, it follows that at least one of them must be false. Let us assume that H_k is true, then $H_{k'}$ must be false and H_k will be the first in its generation that qualify to be checked for a Type I error. An advantage is thus gained by H_k because there is one false hypothesis we know about and thus one less true null hypothesis we have to account for in the tree. Consequently, for nodes that have a terminal sibling, we can further reduce the count of elementary tests by 1 and which leads to r_k.

Next in the algorithm, the hierarchical adjustment of π_k^{adj} to $\pi_k^{\text{h-adj}}$, preserves the hierarchical structure of the hypotheses. As previously mentioned, if H_k is true then all of its descendants are true, thus it follows that the p-value obtained for some descendant node is as small as that of all of its ancestors. Logically this makes sense since in the upper nodes of the tree, more response outcomes are being tested at each node, which opens the door to obtaining more evidence against the null hypothesis at that node. Lastly, in the algorithm, we include the "if $\pi_k^{\text{h-adj}}$" step for computational reasons, as once a parent has a p-value over some pre-specified threshold α, there is no need to search its descendant nodes. This condition can be bypassed easily by setting $\alpha = 1$, if one is interested in the full suite of hierarchically adjusted p-values.

Proposition 3.1. *Let $\mathfrak{T}$ be a hierarchy of tests arranged in a tree according to the discussions in Section 3.2.3 and consider the procedure described in Algorithm 3.1. Let $\mathfrak{T}_0$ be the set of all $H_k \in \mathfrak{T}, k = 1, \ldots, K$ where the truth is consistent with H_k. Then, $Pr(\exists\ H_k \in \mathfrak{T}_0 : \pi_k^{h\text{-}adj} \leq \alpha) \leq \alpha$, that is, the family-wise error rate is controlled at level $\alpha \in (0, 1)$.*

3.3 Simulations

In this section, we perform two simulation studies to explore the error rates and power properties of the hierarchical testing procedure under the two-group model. In the first simulation, we investigate the influence of different linkage criteria, or more generally, different hierarchical structures on the error rates and the power to detect cluster-wide signals. The second simulation study assesses the performance of the method at the terminal nodes, where individual hypotheses are tested, and compares them to some existing methods.

3.3.1 Simulation Study Setup

Let y_{ijq} represent the ith observation, $i = 1, \ldots, n_j$, from the jth group, $j = 1, \ldots, J$, of variable q, $q = 1, \ldots, Q$. $Q = 100$ is the total number of response variables, $n_j = 80$ is the sample size, and $J = 2$ is the total number of groups, treatments, or interventions. For each variable, sample $i = 1, \ldots, 40$ belongs to group $j = 1$, and sample $i = 41, \ldots, 80$ belongs to group $j = 2$. The distributions for the response variables in group j are given by $\mathbf{Y_j} = (Y_{j1}, \ldots, Y_{jQ})^T \sim MVN(\boldsymbol{\mu_j}, \boldsymbol{\Sigma})$. The mean vector for group 1, $\boldsymbol{\mu}_1 = \mathbf{0}$, and for group 2, $\boldsymbol{\mu}_2 = \{0, \beta\}^Q$, a vector of zeros and βs. $\boldsymbol{\mu_2}$ can also be described as the true difference in means between $\mathbf{Y_1}$ and $\mathbf{Y_2}$ since the mean vector for group 1 is $\mathbf{0}$. Five values of β are considered: $0.5, 0.75, 1, 1.25, 1.5$. The number of responses in group 2 with mean β, also known as sparsity, is varied over different numbers of variables: $10, 20$, and 50. The particular variables chosen to have a mean of β are done so randomly. For example, if the sparsity level is set to 10 and $\beta = 1$, then $\boldsymbol{\mu}_2$ is a vector of 90 zeros and 10 ones. As a result, in 10 randomly chosen variables out of the 100 variables, the truth is consistent with the alternative hypothesis in Equation (3.2).

Four different versions of the variance-covariance matrix, $\boldsymbol{\Sigma}$, are considered, with each version providing the opportunity to compare the method under different dependency settings:

independence, equal, exponential off-diagonal decay, and arbitrary variance covariance structures. We denote the variance of Y_{jq} by σ^2_{jq} which is set to be equal to 1, and the covariance between Y_{jq} and $Y_{jq'}$ by $\sigma_{jq,jq'}$. Under independence, the covariance, $\sigma_{jq,jq'}$ is 0 for $q \neq q'$. For the equal covariance structure, $\sigma_{jq,jq'} = 0.5$ when $q \neq q'$. For exponential off-diagonal decay, $\sigma_{jq,jq'} = 0.5^{|q-q'|}$ when $q \neq q'$. For the arbitrary covariance structure, we simulate a $Q \times Q$ correlation matrix using the `rcorrmatrix` function from the `clusterGeneration` package (Qiu & Joe., 2020) in R (R Core Team, 2023). Note that, since $\sigma^2_{jq} = 1$, the covariance matrix and the correlation matrix will be the same.

The Welch's two-sample t-test is used to generate a p-value for each set of hypotheses in Equation (3.2) under each case outlined above, and the procedure is repeated 1000 times. We use only the GHC test to calculate the cluster-wide p-values for these simulations, but we show in Appendix B that the results hold for the other tests discussed in Section 3.2.2. The empirical FWER is calculated as

$$FWER = \sum_{w=1}^{W} \frac{I(\pi_k^{\text{h-adj}} < \alpha \cap H_k \text{ is true }) \text{ at simulation w}}{\text{W}}, \tag{3.6}$$

and the average empirical power is calculated as

$$\frac{1}{W} \sum_{w=1}^{W} \left(\frac{\sum_{k=1}^{K} I(\pi_k^{\text{h-adj}} < \alpha \cap H_k \text{ is false }) \text{ at simulation w}}{\sum_{k=1}^{K} I(H_k \text{ is false}) \text{ at simulation w}} \right), \tag{3.7}$$

where W is the total number of simulations.

3.3.2 Comparing Across Linkages

In this subsection, we compare and discuss three commonly used linkage criteria in hierarchical clustering: the single linkage, the complete linkage, and Ward's criterion. The single linkage criterion measures cluster distance as the minimum distance between members of the clusters whereas the complete linkage uses the maximum distance. The Ward's criterion

measures the distance between two clusters by minimizing the error sum of squares (ESS) objective function. Murtagh & Legendre (2014) and Everitt (2011) have more information on Ward's method and hierarchical cluster analysis. All cluster analyses are performed using `hclust` in base R.

Figure 3.4 shows the empirical FWER compared across the different linkages, correlation versions, sparsity levels, and effect sizes. We see that generally the hierarchical testing procedure is more conservative when the number of non-nulls is less sparse. Also, it appears that the FWER for single linkage is the lowest compared to the other linkages when the variables are arbitrarily correlated or they are independent. In Figure 3.5, we observe that the powers for complete linkage and Ward's criterion tend to be comparable, and they are lower than that of the single linkage. This is as a result of the cluster-wide adjustment used on the node p-values. The adjustment is more lenient on clusters with a leaf sibling. Coincidentally, the single linkage criterion tends to produce chaining of long thin clusters of terminal nodes, thus creating more clusters that have a leaf sibling. For example, see Figure 3.6(b). It is important to note that the higher power seen with the single linkage is not an indication of an improved clustering ability. In fact, Figure 3.6(b) shows that the hierarchy produced by the single linkage tends to have no discernible clusters under equal covariance where it shows high power. Under the exponential off-diagonal decay in Figure 3.6(d), where chaining is less apparent, we see that in the power plot in Figure 3.5 that the power of the single linkage is not very different from the other linkages. Ultimately, care in the creation creation and assessment of the hierarchical cluster analysis seems to be related with how interpretable the hierarchy might be. Hierarchical structures that have long thin chains of variables may provide increased power, but this is at the expense of losing any meaningful clusters.

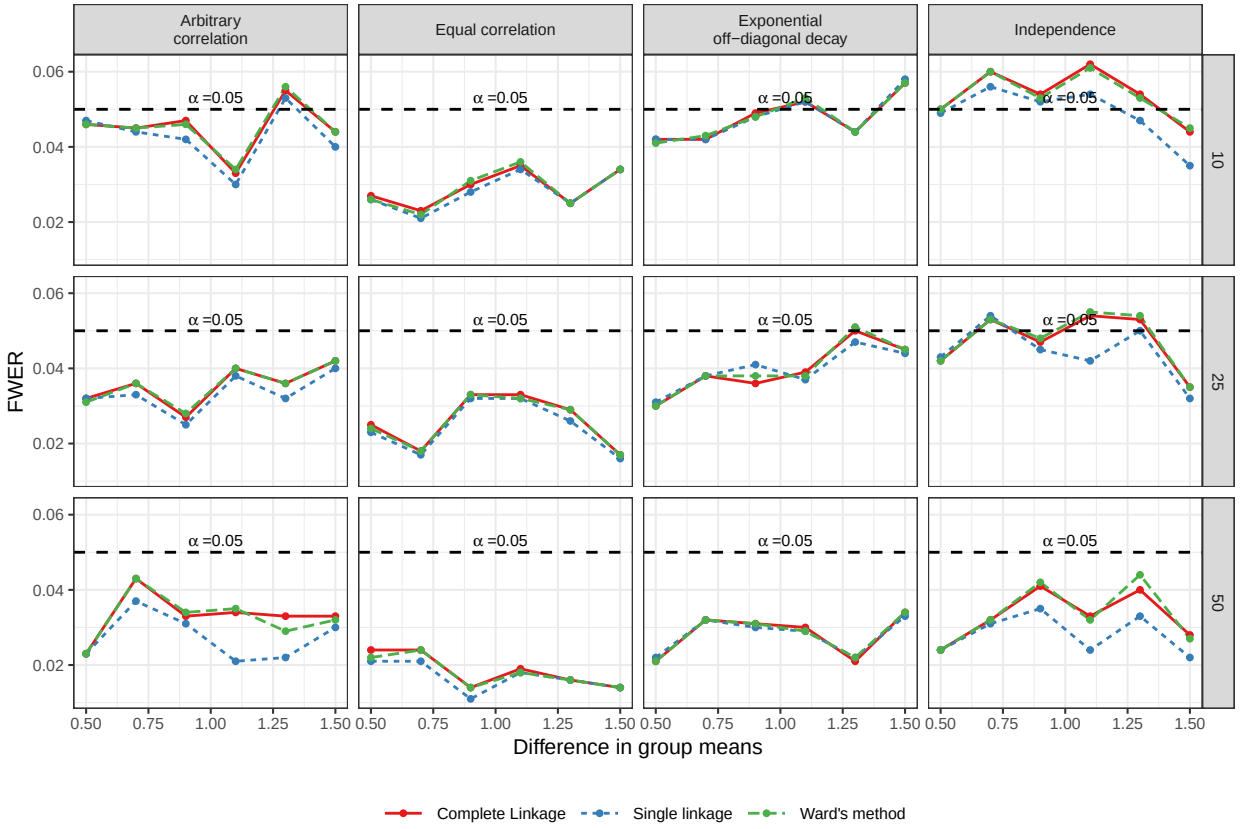


Figure 3.4: Empirical family-wise error rates of the hierarchical testing procedure compared across different correlations, sparsity levels, and linkage criteria ($\alpha = 0.05$) obtained with 1000 simulations on 100 variables for each setting.

3.3.3 Comparing to Existing FWER Methods

Traditional multiple testing controlling procedures aim at providing adjusted evidence for each elementary hypothesis test. Thus, we focus on the adjusted p-values obtained at the terminal nodes of the hierarchy to compare our adaption of the hierarchical testing procedure to existing traditional methods. The methods that are considered are the Bonferroni procedure, the Bonferroni-Holm procedure (Holm, 1979), and the Westfall-Young procedure (Thomas et al., 1994). The adjusted p-values of the Bonferroni method are given by $p_{q,\text{Bon-adj}} = \min(p_q Q, 1)$. The Bonferroni-Holm is a step-down procedure with adjusted

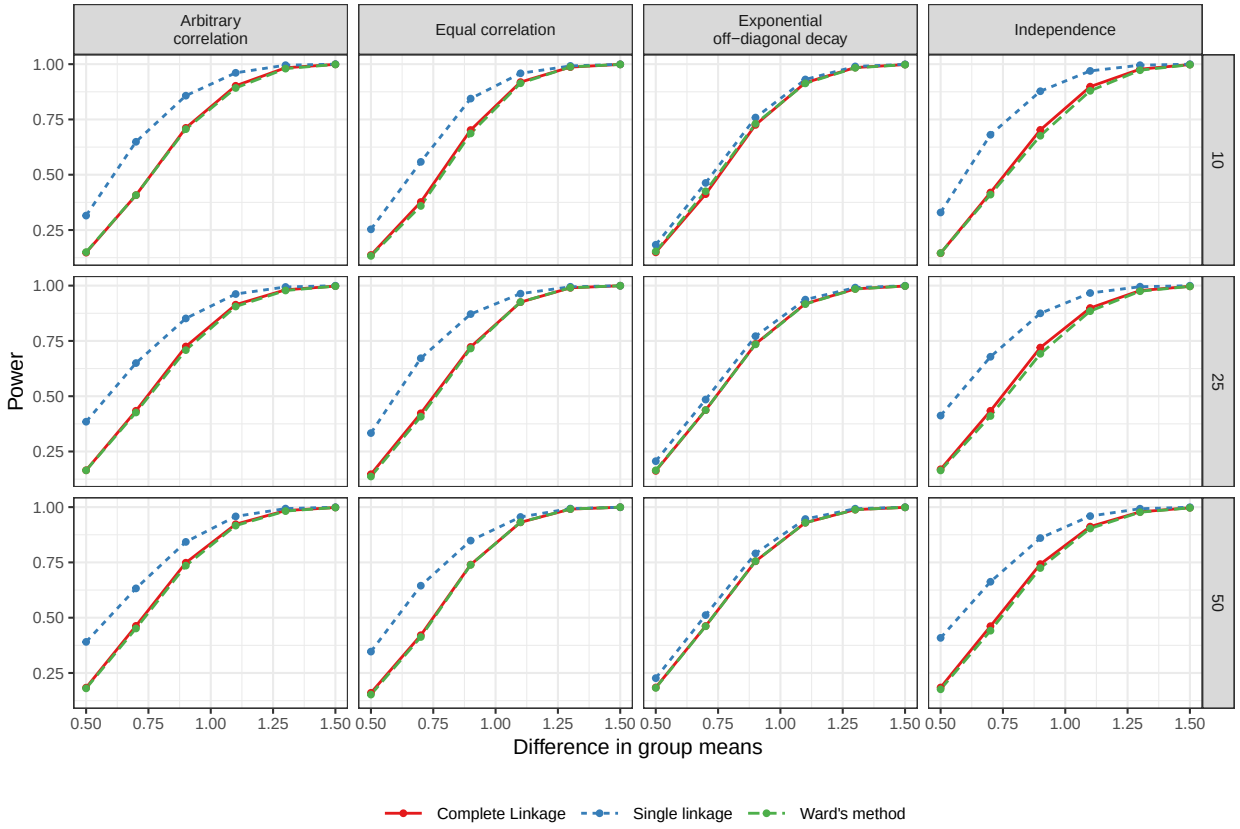


Figure 3.5: Average empirical powers of the hierarchical testing procedure compared across different correlations, sparsity levels, and linkage criteria (α= 0.05) obtained with 1000 simulations on 100 variables for each setting.

p-values $p_{(q),\text{Holm-adj}} = \min[p_{(q)}(Q - q + 1), 1]$, where $p_{(q)}$ is the qth ordered p-value. The Westfall-Young is a step-down procedure that uses a permutation-based approach to account for the correlation structure of the variables. In the simulations, we use the `mt.maxT` function from the `multtest` package (Pollard et al., 2005) to perform the Westfall-Young procedure with tests that are based on two-sided two-sample Welch t-statistics and the number of permutations set to 10000. See Thomas et al. (1994) Algorithm 2.8 for a more information on the procedure and algorithm choices.

Figure 3.7 suggests that at the terminal nodes of the hierarchy, the hierarchical testing procedure is only slightly more liberal than the Bonferroni procedure, and both methods are

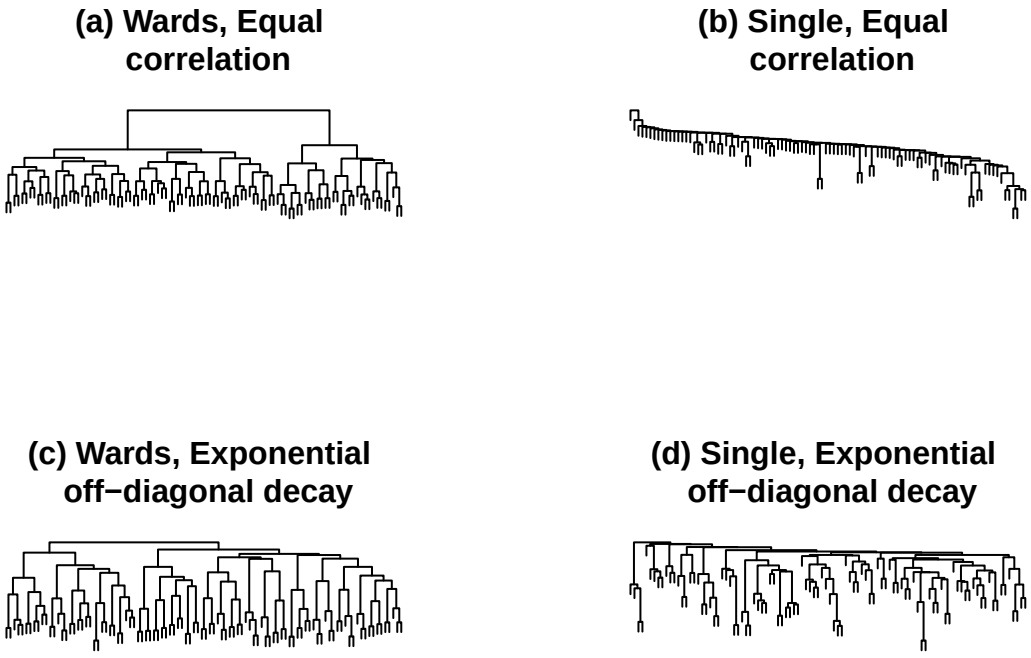


Figure 3.6: Examples of the hierarchical structure of the Ward's and Single linkage under equal covariance and exponential off-diagonal correlation decay.

more conservative when compared the Bonferroni-Holm and Westfall-Young approaches. This is unsurprising since the terminal nodes of the hierarchy receive a Bonferroni adjustment if its sibling is not a leaf node. When the sibling is a leaf node, the adjustment is reduced by a factor of 2 making it less stringent. This is also reflected in the power plots of Figure 3.8 where we see a minor improvement in the power of our method over the other methods. Note that the key advantage of our adaption of the hierarchical testing procedure is that it considers both adjusted combined evidence higher up in the hierarchy as well as adjusted marginal evidence at the terminal nodes of the hierarchy, taking advantage of logical relationships among the nodes to improve power at the terminal nodes of the hierarchy.

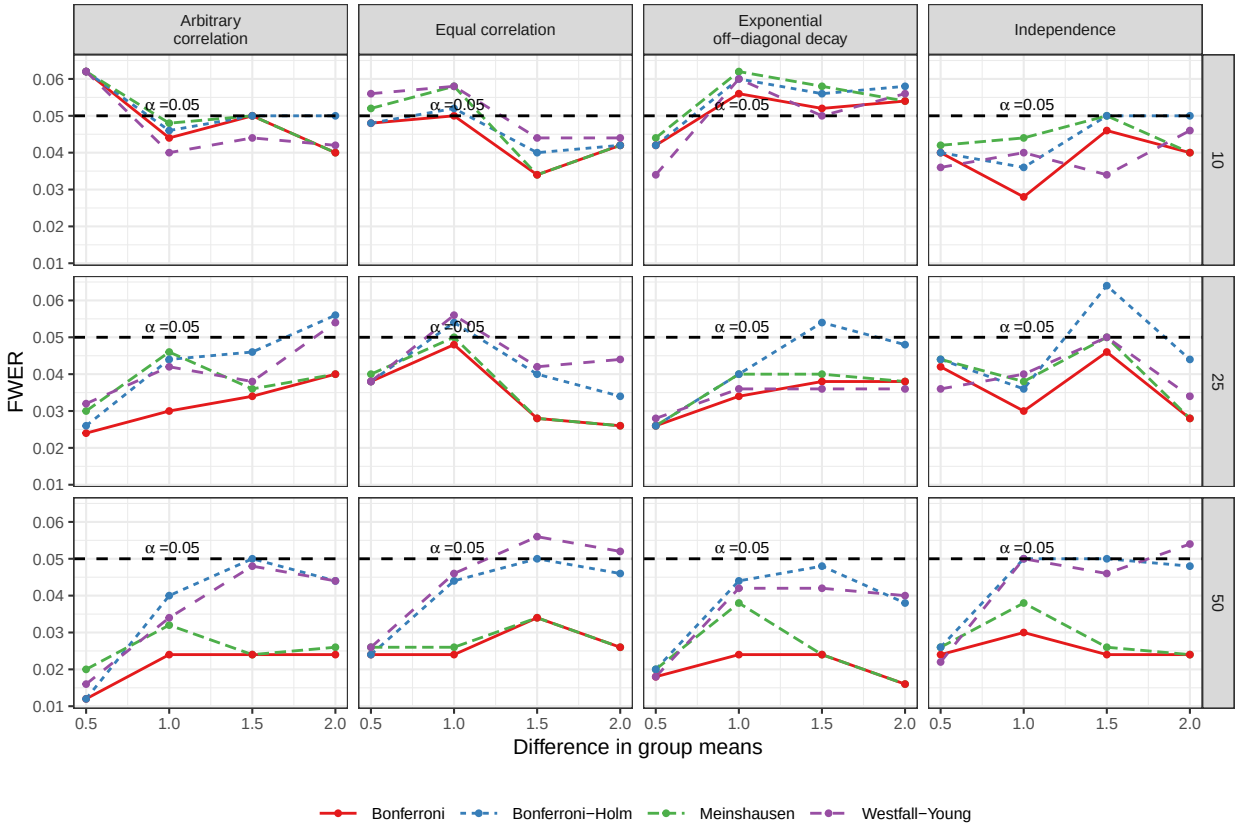


Figure 3.7: Empirical family-wise error rates at the leaf nodes compared to Bonferroni, Bonferroni-Holm, and Westfall-Young across different correlations and sparsity levels (α= 0.05) obtained with 1000 simulations on 100 variables for each setting.

3.4 Application to Mouse Data

Figure 3.9 provides a graphical illustration of the Spearman correlations of the $Q = 181$ metabolite abundances arranged using the Ward's hierarchical clustering order and a modified version of the `corrplot` (Wei & Simko, 2021). The plot demonstrates the dependency structure that exists among the metabolites. We see that the metabolites are generally positively correlated with a few that are negatively correlated. The correlated metabolites also appear clustered, indicating that there might be some interesting groupings of variables suitable for further investigation. We therefore use the Ward's criterion and the Spearman

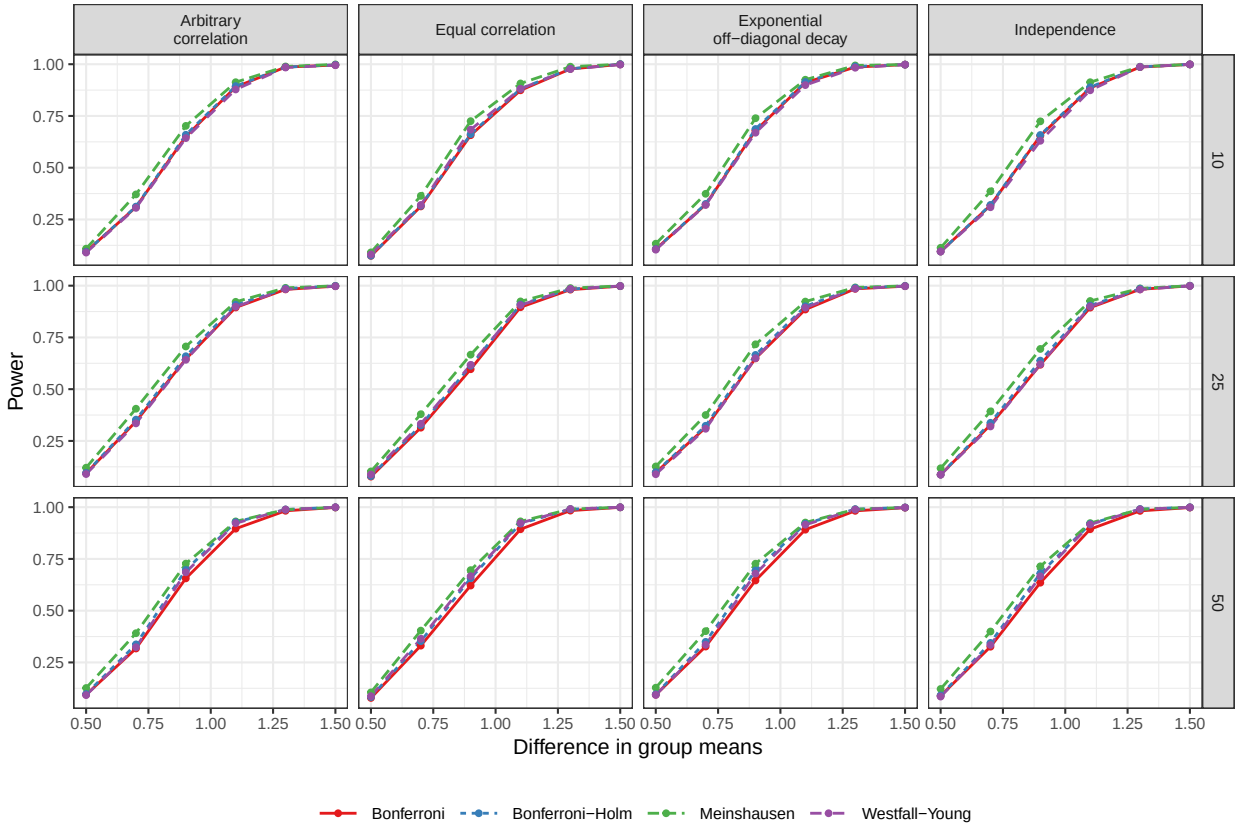


Figure 3.8: Average empirical powers at the leaf nodes compared to Bonferroni, Bonferroni-Holm, and Westfall-Young across different correlations and sparsity levels (α= 0.05) obtained with 1000 simulations on 100 variables for each setting.

correlation-based absolute value dissimilarity to create a hierarchical structure for testing in these data.

For each metabolite, we perform a two-sample Welch's two-sample t-test on the log-abundances of whether there is a difference between the true mean abundances of hyperglycemic and normoglycemic mice, and obtain an associated p-value. We then apply four multiple testing controlling procedures to the p-values: hierarchical testing with the GHC global test, Bonferroni, Bonferroni-Holm, and Westfall-Young to adjust for multiplicity. The p-values for the four methods along with the unadjusted p-values are presented in Figure 3.10. Note that in Figure 3.10, we only consider the adjusted p-values at the terminal nodes

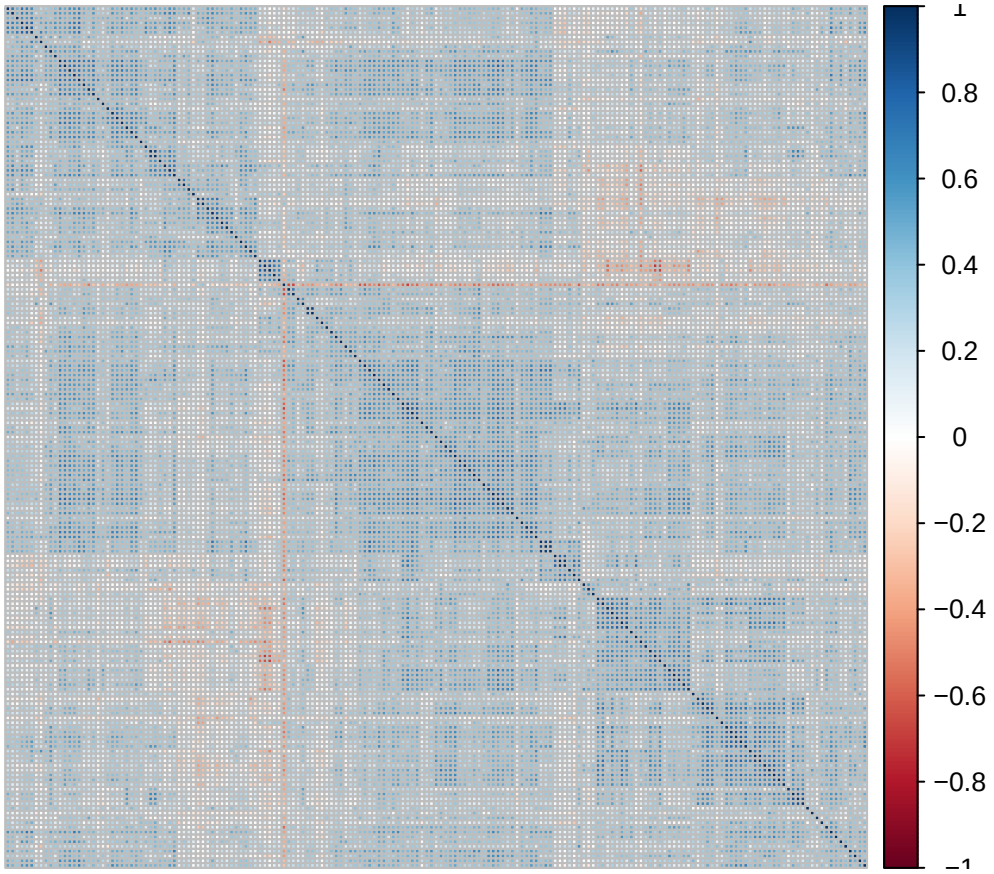


Figure 3.9: Spearman correlations of the 181 scaled and log-transformed metabolites found in the mouse study. Darker tiles represent higher correlations and lighter tiles represent lower correlations. Red and blue tiles represent negative and positive correlation values. Metabolites are sorted based on Wards hierarchical cluster analysis with the Spearman correlation absolute value dissimilarity.

of the hierarchical testing results for comparison.

In Figure 3.10, there is a one-to-one relationship between the hierarchical testing procedure and both the Bonferroni and the Bonferroni-Holm procedure, except on the occasions where the metabolite had a leaf sibling. In those cases where the metabolite had a leaf sibling, we see that as expected, the adjusted p-value is lower for the hierarchical testing method when compared to the other methods. It should be also noted that the hierarchically adjusted p-values are 1 for Westfall-Young adjusted p-values above 0.3. This is as a result of

how stringent the Bonferroni adjustment is at the terminal nodes of the hierarchy. A similar pattern can be seen in the plot of Bonferroni versus Westfall-Young, and Bonferroni-Holm versus Westfall-Young. However, this is not an issue since the strength of evidence needed for inference does not really change.

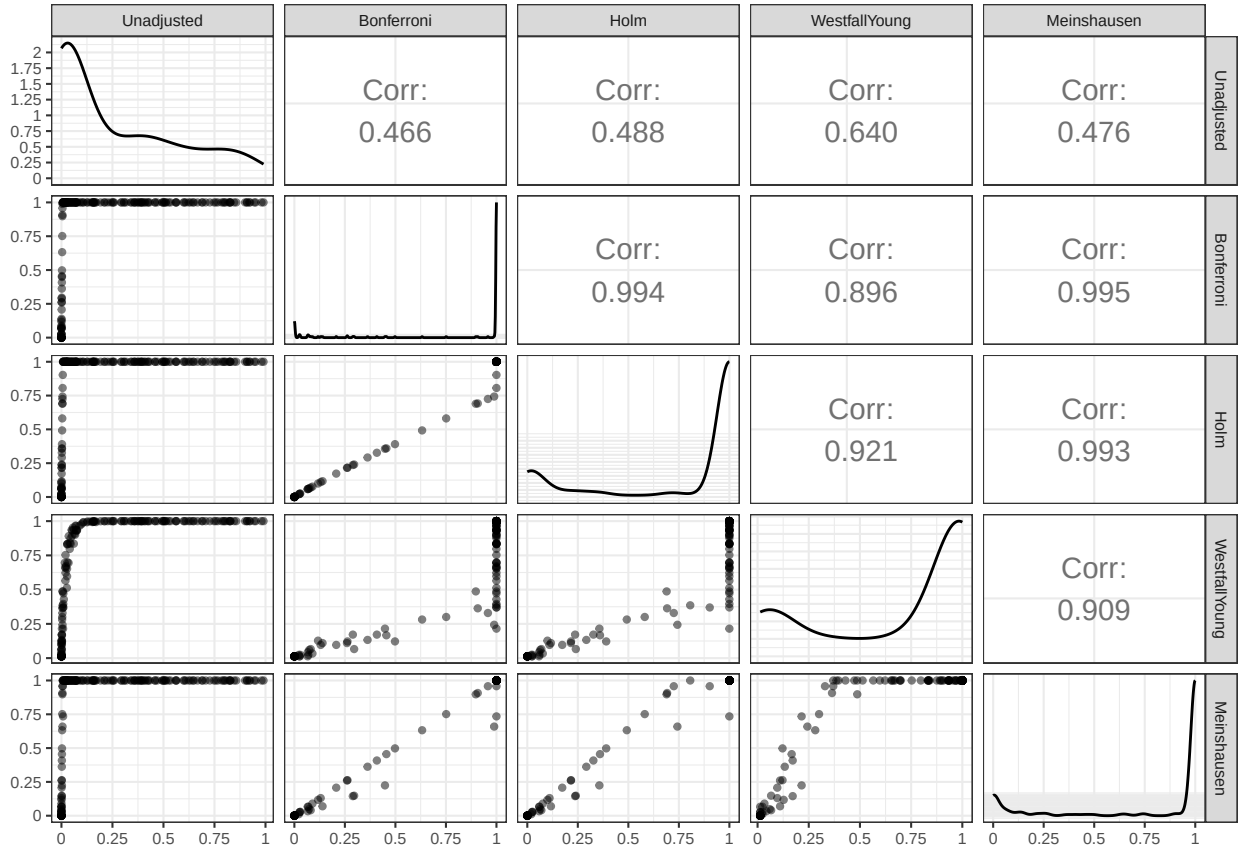


Figure 3.10: Plots of p-values for mouse data across the four adjustment methods along with the raw unadjusted p-values.

In Figure 3.11, we provide a dendrogram of the metabolites showing the Meinshausen adjusted p-values that are less than 0.05. There were 24 metabolites that were selected at the 0.05 threshold. This is compared to 21 by the Bonferroni, 21 by the Bonferroni-Holm, Westfall-Young with 25, and 62 unadjusted p-values less than 0.05. More interestingly, it can be observed from the dendrogram that the selected metabolites fall into two clusters,

this provides a logical next step for further investigation. It will be of interest to investigate how the selected metabolites in each cluster are related to each other. To do this, we cut the tree at the five cluster mark. The decision to pick five clusters is largely subjective, but one may employ any of the many methods available in literature for picking a suitable number of clusters, but by cutting the tree we can add cluster information to other visualizations.

The parallel coordinate (PC) plots (Schloerke et al., 2021) in Figure 3.12 show the standardized log-abundances of metabolites that belonged to one of the two clusters with at least one selection. Figure 1 from Fahrmann et al. (2015) suggests that Figure 3.12 (a) is generally made up of carbohydrates whereas Figure 3.12 (b) consists of amino acids. In both plots, the hyperglycemic mice tend to have increased abundances for carbohydrates and decreased abundances for amino acids compared to the normoglycemic mice. Overall, these general conclusions seem to be consistent with those from Fahrmann et al. (2015).

3.5 Discussion

In this paper, we implemented a hierarchical procedure to test multiple variables simultaneously for differences between two groups. The method used was initially proposed by Meinshausen (2008) to select variables in a linear regression. Through simulation studies, we examined how different hierarchical structures impact family-wise error rates and performance. We found out that hierarchies with long thin chained clusters may provide higher power because of the adjustment used but they may lead to un-interpretable and often singleton clusters in the hierarchy. We also compared the method to existing traditional methods and showed that even at the individual hypotheses level, the method improves power. However, this comes with a computational trade-off, given the extensive testing required across all nodes in the hierarchy. Although we implement decision rules to mitigate this issue, the sheer volume of outcomes may still pose challenges. Especially when dealing with thousands of outcomes, our method might run slower in comparison to other methods.

This work was motivated by an interest in accommodating variable dependence into multiple testing and inference. Frequently, clustering and multiple testing are treated and reported separately. However, we see that there are advantages of improving power and enhancing inference when they are considered together. The potential to focus cluster explorations on clusters that contain metabolites with detectable "signal" can allow a researcher to target specific related groups of features and ignore the other clusters that may not contain information of interest in the application.

The power of the method to detect important clusters and variables mainly hinges on the multiplicity adjustment used. The method currently uses a Bonferroni-based adjustment which assumes that all hypotheses tested are truly null, but this is not always so. This makes the method conservative at the level of individual variables. The Bonferroni adjustment also becomes more conservative when the hypotheses are strongly and positively correlated, yet this is a likely characteristic of the type of variables considered as seen in Figure 3.9. Thus, it may be advantageous in future extensions to consider modifications or changes to make the adjustment less stringent. Other extensions to the method could also be considered. For example, the method could be extended to accommodate an initial test for more than two groups that has a sub-hierarchy of tests for pairwise comparisons of all groups, or to test for interaction effects and then main effects in a two-way ANOVA design. Extensions to models where there is a suite of control variables could also be considered.

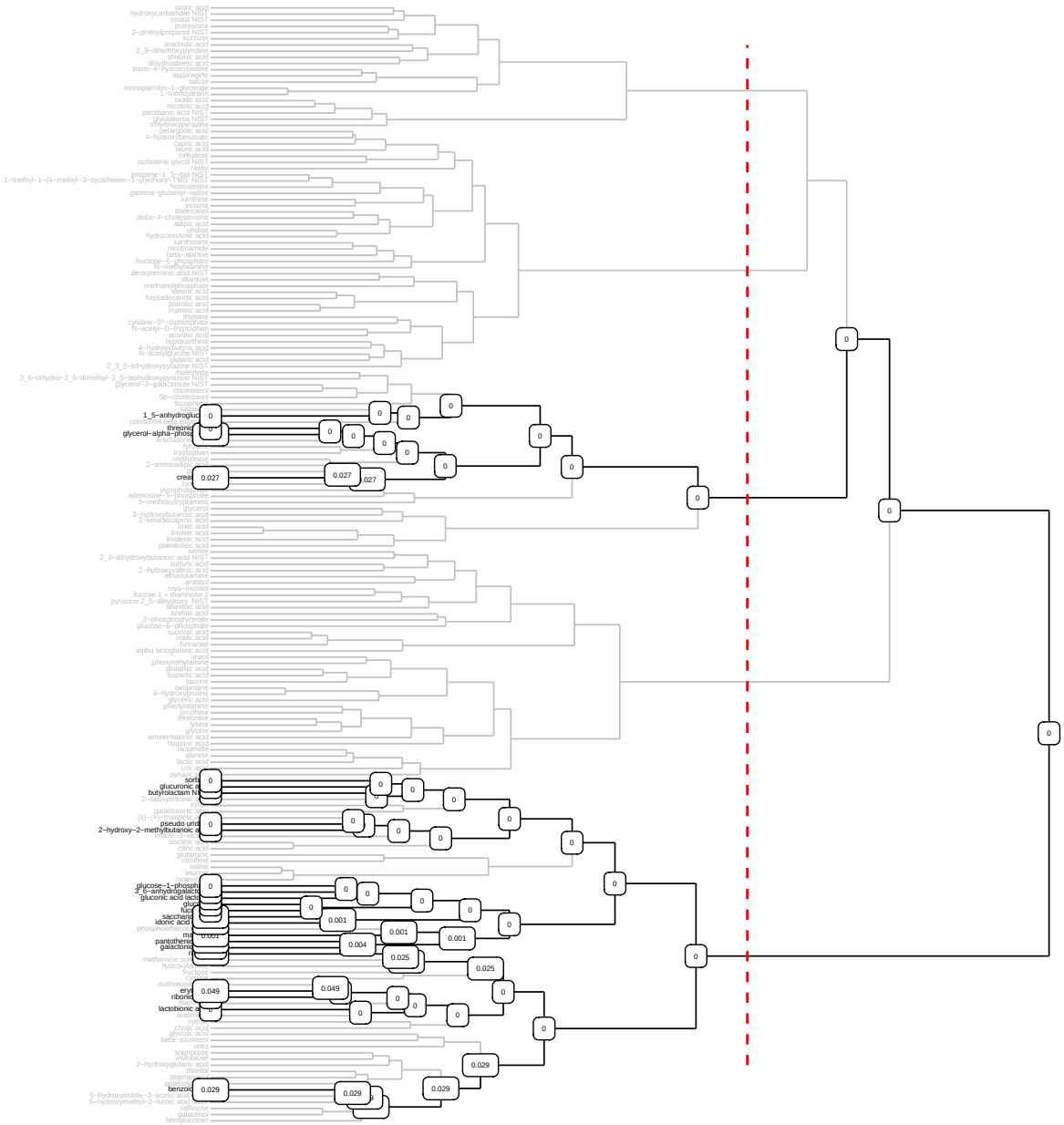


Figure 3.11: Dendrogram of metabolites with the Meinshausen and the adjusted p-values of the selected metabolites cutting tree at height with five clusters. Node branches and metabolite labels greyed out had p-values greater or equal to 0.05.

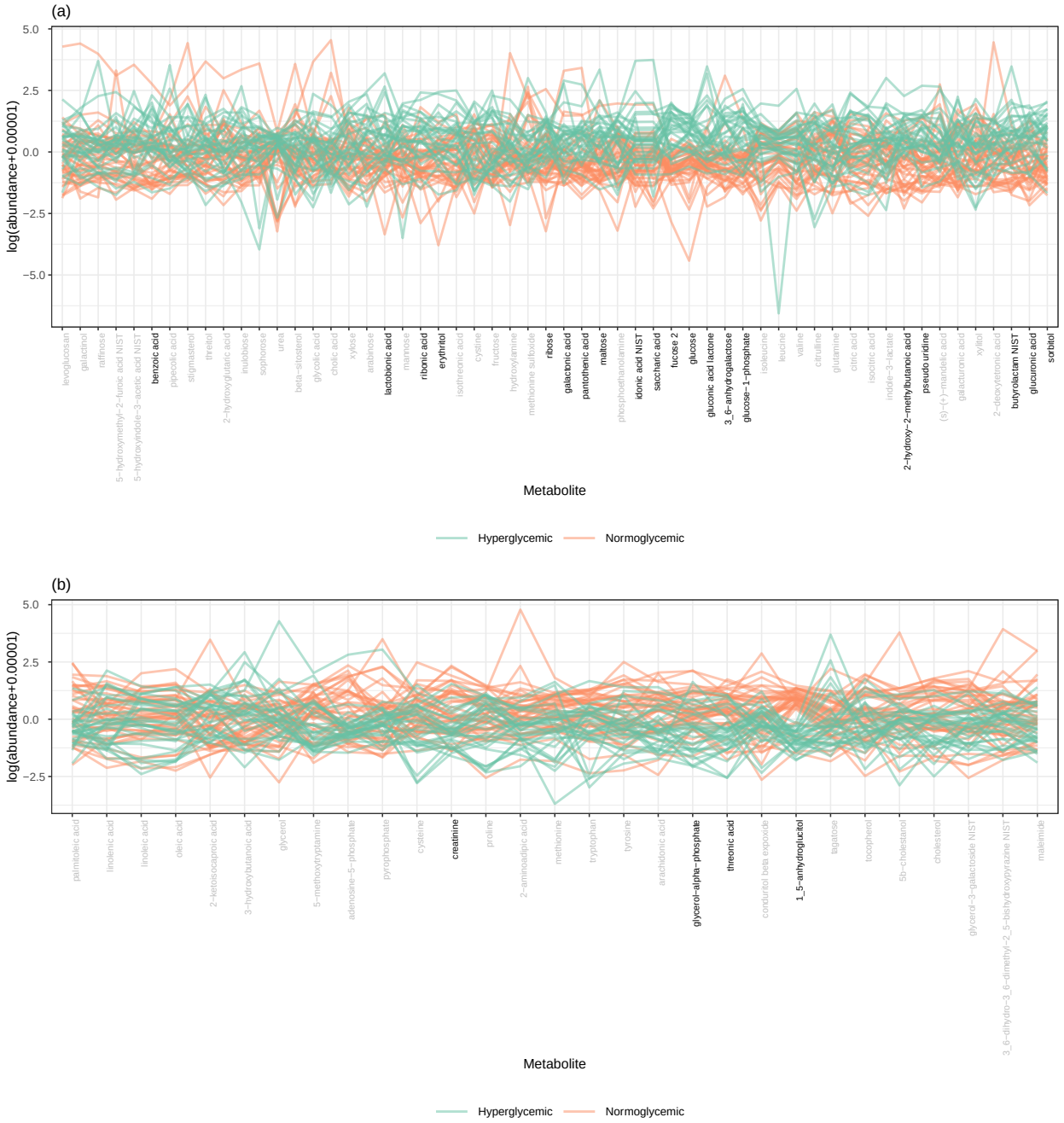


Figure 3.12: Parallel coordinate plots of two clusters from the dendrogram that contain at least one metabolite with Meinshausen adjusted p-value < 0.05. Metabolites with p-values < 0.05 in bold text.

CHAPTER FOUR

A TIERED HIERARCHICAL MULTIPLE TESTING PROCEDURE FOR FOLLOW-UP TESTS IN ANALYSIS OF VARIANCE

Contribution of Authors and Co-authors

Author: Priscilla Bacino

Contributions: Responsible for majority of the writing

Co-Author: Dr. Mark Greenwood

Contributions: Provided feedback on statistical analysis and drafts of the manuscripts

Manuscript Information Page

Priscilla Bacino, Mark C. Greenwood

Status of Manuscript:
___x___ Prepared for submission to a peer-reviewed journal
_____ Officially submitted to a peer-reviewed journal
_____ Accepted by a peer-reviewed journal
_____ Published in a peer-reviewed journal

Abstract

Investigating differences in mean outcomes among multiple groups, treatments, or experimental conditions is a common pursuit in biological research. However, when dealing with a multitude of outcomes, ranging from hundreds to millions, the number of hypotheses to be tested can be overwhelming. Traditional statistical methods, given the extensive number of hypotheses, may tend to be conservative. To address this challenge, we can harness structure in a multiple testing procedure to enhance statistical power and facilitate result interpretation. We introduce a tiered-hierarchical approach designed to control family-wise error rates. This method comprises two tiers: the primary tier, which assesses inter-group differences, and the follow-up tier, which examines specific group comparisons. The testing process is hierarchical, beginning at the top and moving downward, proceeding to the next level only when there is strong evidence against the hypothesis at each node. Additionally, we introduce an adaptive version of the method that enhances power based on the estimated number of true null hypotheses at the terminal nodes. Comparative evaluations demonstrate that this approach outperforms similar methods. To illustrate its practical application, we apply the method to both simulated and real datasets. The method is accessible through the hiermt package, currently available on Github.

4.1 Introduction

In biological experiments and studies, a frequent research objective is to investigate differences in the true mean of an outcome across various groups, treatments, or experimental settings. Examples include testing the efficacy of a drug across different groups of subjects, exploring the association between specific traits and a gene, or assessing how controlled environmental factors influence variables such as plant growth. Although an initial aim may be to detect if differences exist in the mean outcome across these groups, the ultimate goal often involves pinpointing the specific groups where these differences occur if more than two groups or are considered. To achieve this, studies commonly employ follow-up tests. These can be as straightforward as those comparing means between all pairs of groups or as intricate as analyzing contrasts between various treatment combinations that reflect the complexity of the study design. However, the complexity of these analyses can further

increase if multiple outcomes are examined concurrently, be it in the form of multiple genes, environmental factors, or endpoints that can range in number from hundreds to millions of outcomes. This introduces a challenge in simultaneous inference which is in managing the problem of multiplicity under two forms. The first stems from making numerous simultaneous comparisons for each outcome to localize group differences. The second emerges from the vast number of outcomes for which these comparisons are made. For example, with 100 outcomes and 5 groups, there are 1000 potential pairwise comparisons that could be considered.

Methods for multiple comparisons that control the family-wise error rate (FWER) for a single outcome have been well-established. These methods include tests specifically designed to evaluate contrasts and inherently adjust for multiple testing, such as the Tukey-Kramer (Kramer, 1956; Tukey, 1953), Scheffé (Scheffe, 1999), and Dunnett's test (Dunnett, 1955). The situation changes when considering multiple comparisons across multiple outcomes. Although the endeavor is prevalent in fields such as genomics, metabolomics, and proteomics (Hernandez et al., 2020; Maag et al., 2015; Sultan et al., 2022), there is no general consensus on an approach. Regardless, a pitfall to avoid is focusing on addressing only one of the multiplicity issues. For example, addressing multiplicity for comparing multiple groups within each outcome in isolation risks inflating the family-wise error rate (defined and explored in Section 4.5). General multiple testing procedures like the Bonferroni (Dunn, 1961) and the Bonferroni-Holm method (Holm, 1979) that aim to control the FWER across any family of tests may be applicable to the entire suite of tests comparing multiple groups for all outcomes, but the tests may be too stringent as the number of tests considered within this family can be large.

Some researchers have proposed multivariate techniques for multiple comparisons for (Nishiyama et al., 2014; Pietrzykowski & Zieliński, 2004). These methods allow for the simultaneous testing of differences across treatments, leveraging the joint distribution of outcomes to account for their inherent correlation structure. However, while these techniques

can be effective for detecting group differences that might be overlooked in single-outcome evaluations, they may sometimes lack necessary detail. Often, investigators aim to discern not only which groups differ but also the specific outcomes that drive these differences.

A two-step testing strategy, which can be conceptualized in two ways, can be adopted. The first approach begins with multivariate multiple comparisons to identify groups that exhibit differences, and then isolates the specific outcomes responsible for these differences. The second approach starts by pinpointing outcomes that manifest detectable group differences, and then refines these differences down to particular groups. The latter approach, often the natural choice, may also be more computationally efficient, as it focuses post-hoc testing efforts solely on outcomes with detectable group differences, making it more advantageous in high-dimensional settings or those involving many groups. For either strategy chosen, addressing both forms of multiplicity issues is essential.

In this paper, we introduce a two-stage multiple testing procedure set within the framework of controlling the family-wise error rate (FWER). This procedure adheres to the "outcomes first, groups later" approach previously discussed, adopting a hierarchical structure divided into two tiers. The primary tier focuses on testing hypotheses that assess potential group differences across outcomes, while the follow-up tier evaluates hypotheses for specific group differences. The procedure begins in the primary tier starting at the top of the hierarchy with a global hypothesis that tests for group differences across all outcomes, and continues down to the terminal nodes with hypotheses that test for differences for individual outcomes. If strong evidence is found against the null hypotheses at a terminal node for any specific outcome, the procedure advances to the follow-up tier to assess particular group differences for that outcome. The hierarchical structure, generated from correlation-based agglomerative hierarchical clustering, attempts to cluster hypotheses exhibiting a strong signal for potential differences together to facilitate signal detection and enhance interpretation. Additionally, the structure imposes logical constraints within the follow-up tier that can be used to gatekeep,

and dynamically adjust p-values, and improve statistical power.

The rest of this chapter is structured as follows. Section 4.2 delves into the tiered-hierarchical structure and Section 4.3 explores its associated testing procedure. Section 4.4 examines the theoretical aspects of controlling the family-wise error rate within the framework of our proposed method. Simulation-based performance in terms of statistical power, as well as error rates, are presented in Section 4.5. Finally, we analyze a real-world dataset in Section 4.6 and provide concluding discussions in Section 4.7.

4.2 The Tiered-Hierarchical Structure

To describe the tiered-hierarchical structure and its associated testing procedure, consider a scenario where the objective is to determine, among J groups, which pairs have differing mean outcomes, and the assessment is to be carried out for Q outcomes simultaneously. Let y_{ijq} denote the ith observation from the jth group for outcome Y_q, where $i = 1, \ldots, n$, $j = 1, \ldots, J$, and $q = 1, \ldots, Q$. The observation, y_{ijq}, can be expressed using a simple linear model as outlined in Equation (4.1). In this model, μ_{jq} represents the population mean of the jth group for outcome Y_q, and ε_{ijq} serves as the random error associated with observation, y_{ijq}. While no assumptions are made regarding the variance-covariance structure of the ε_{ijq}s, we do assume that these errors are normally distributed with a mean of 0, resulting in the

$$y_{ijq} = \mu_{jq} + \varepsilon_{ijq} \tag{4.1}$$

Now, consider a dendrogram that represents the results of an agglomerative hierarchical clustering of the Q outcomes (Hartigan, 1975). Agglomerative hierarchical clustering groups variables based on their proximity or how closely the variables are deemed to be related. These algorithms start with each variable as its own cluster, iteratively merge pairs of clusters that are most similar into larger clusters, and continues until all variables are contained

within a single cluster. Suppose K clusters are formed within the hierarchy and are numbered based on the depth-first search approach from C_1 at the root node to C_K at the furthest terminal node on the right.

The choice of linkage criterion, which measures the distance between two clusters, as well as the pairwise distances between outcomes both influence cluster formation. To calculate the distance between outcomes throughout this work, the absolute pairwise sample correlation coefficients will be used. The distance, $d(Y_q, Y_{q'})$, between two variables Y_q and $Y_{q'}$ will be calculated as $\sqrt{2(1 - |r(Y_q, Y_{q'})|)}$ (James et al., 2013), where $r(Y_q, Y_{q'})$ represents the Spearman sample correlation coefficient. The choice of Spearman correlation is beneficial for handling outliers and capturing monotonic relationships between variables that might not be linear. Furthermore, using the absolute value ensures that variables are grouped based on the magnitude of correlation rather than the direction of the relationship. The Ward's minimum variance method (Ward, 1963) that chooses the merging of clusters that minimizes the change in the total within-cluster sum of squared distances is used to determine the distance between two clusters.

The outcomes within each cluster, C_k, serve as the focal point for hypothesis testing at node k in the primary tier. At node k, we test a single null and alternative hypothesis for the outcomes jointly, examining whether there are any differences among the population means for the J groups, as depicted in Equation (4.2) to obtain the p-value, π_k. Any suitable overall test can be used to evaluate H_k and $H_{k'}$ and obtain the p-value. For this work, we will employ the Generalized Higher Criticism (GHC) proposed by (Barnett et al., 2017). This method does not directly consider the joint distribution of the outcomes but combines set of elementary test statistics into a single test statistic to obtain π_k. Each of the elementary test statistics is obtained from a hypothesis testing for group differences on each single outcome in C_k. This method has been chosen for its effectiveness in incorporating the underlying dependencies among outcomes and identifying sparse signals — situations where only a small

fraction of outcomes (less than 0.25) demonstrate detectable differences across groups. More details on this method are provided in Section 3.2.2.

$$\begin{aligned} &H_k : \mu_{1,q} = \cdots = \mu_{J,q} \text{ for all } q : Y_q \in C_k \text{ versus} \\ &\bar{H}_k : \mu_{j,q} \neq \mu_{j',q} \text{ for at least one } j \neq j' \text{ for any } q : Y_q \in C_k. \end{aligned} \tag{4.2}$$

In the follow-up tier, single outcomes situated at each terminal node of the primary tier become the focal point for hypothesis testing. There are $M = Q \times \binom{J}{2}$ nodes which corresponds to the count of all possible group pairings for a single outcome multiplied by the total number of outcomes. Each node contains a single null and alternative hypothesis, $H_{k,m}$ and $\bar{H}_{k,m}$ for the mth paired group, $m \in 1, \ldots, M = \binom{J}{2}$, for the outcome at terminal node k. The hypotheses assess whether there is a difference between the means of a specific group pair for a single outcome at node k of the primary tier. The numeration on m is defined alphabetically, for instance, with groups say, a, b and c, $m = 1$ would correspond to the pairing $a - b$, $m = 2$ to $a - c$ and so forth. Equation (4.3) outlines the corresponding null and alternative hypotheses when the populations means of groups j and j' are compared for a particular k and m:

$$\begin{aligned} &H_{k,m} : \mu_{j,q} = \mu_{j',q} \text{ for } Y_q \in C_k \text{ versus} \\ &\bar{H}_{k,m} : \mu_{j,q} \neq \mu_{j',q} \text{ where } j \neq j' \text{ for } Y_q \in C_k. \end{aligned} \tag{4.3}$$

.

Figure 4.1 provides an example of the testing architecture for five outcomes. Within the hierarchical structure, hypotheses at different nodes share relationships that are important for understanding the multiple testing adjustments proposed in the testing procedure detailed in Section 4.3. The hierarchy can be described as a generation of hypotheses, wherein a parent hypothesis is positioned one level above its child and has a direction connection to it. For instance, in Figure 4.1, the parent of H_7 is H_5. Meanwhile, the parent of $H_{3,1}, H_{3,2}$, and $H_{3,3}$

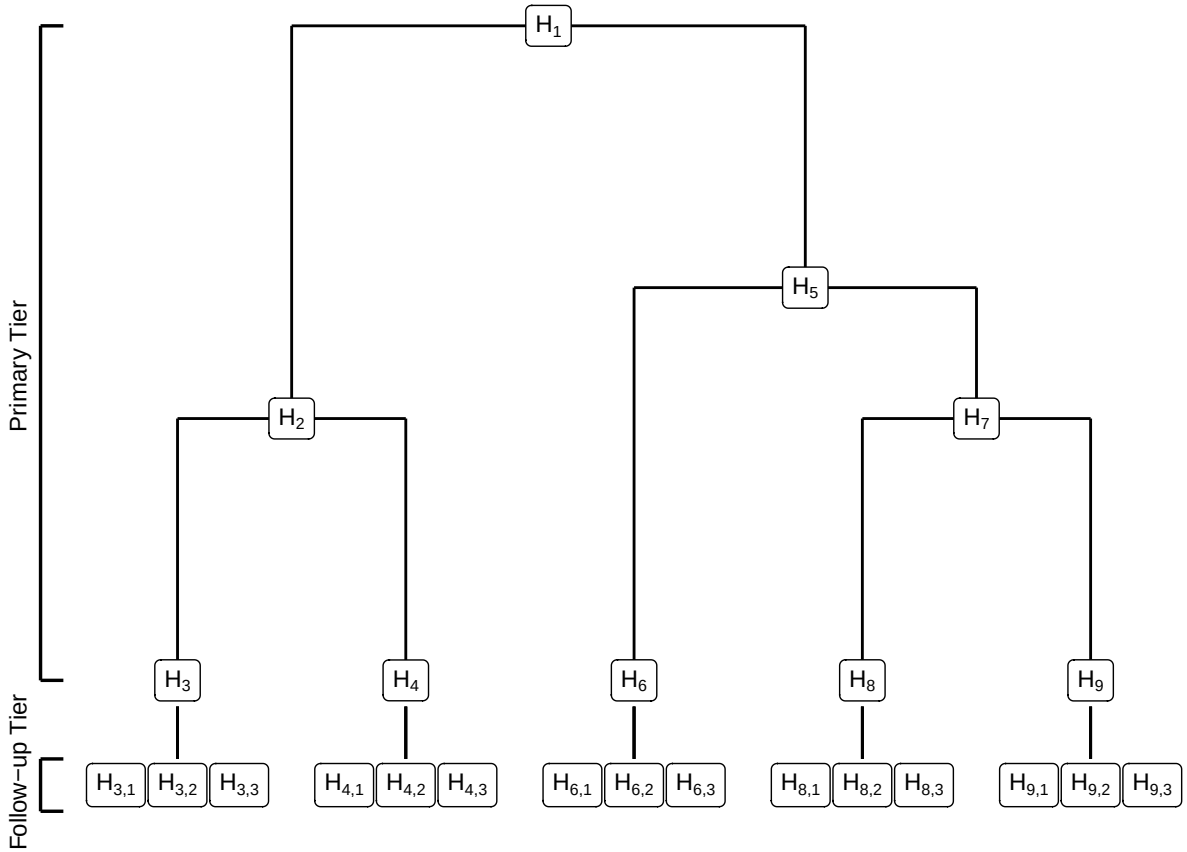


Figure 4.1: An illustration of the tiered-hierarchical structure. Hierarchical testing starts at the root with all outcomes to terminal nodes containing a single outcome. Tiered testing starts with the primary tier for any group differences and the follow-up tier for multiple comparisons.

is H_3. In notation, parents of H_k and $H_{k,m}$ will be denoted as H_{par_k} and $H_{\text{par}_{k,m}}$, respectively. In the primary tier, the outcomes clustered together at child nodes create a partition on the cluster of outcomes under consideration at the parent node. Conversely, in the follow-up tier, children are all the potential pairs of groups for a single outcome. Generally, a child hypothesis will probe into a narrower scope of the research problem compared to their parent hypothesis. An ancestor hypothesis refers to any hypothesis positioned above another in the lineage, not necessarily before it, while a descendant hypothesis is placed below in sequence. These are correspondingly denoted as H_{ansc_k} and H_{desc_k}.

The hypotheses within the tiered-hierarchical structure are logically related, such that not every combination of true and false null hypotheses is feasible. A true null hypothesis for a parent implies that its child hypothesis is also true. This is because the null parameter space of the parent is a subset of that of the child. Hence, any values that satisfy the null hypothesis of the parent must also satisfy that of its children. In the same vein, a false null parent hypothesis necessitates that at least one of its child hypotheses is also false. Furthermore, in the follow-up tier, not all combinations of true and false null hypotheses, when comparing pairs, are possible. A simple illustration, highlighted by Shaffer (1986), involves testing three null hypotheses: $H_{k,1}, H_{k,2}$, and $H_{k,3}$. Each one evaluates the pairwise equality of three population means: $\mu_j, \mu_{j'}$, and $\mu_{j''}$. Thus, the hypotheses are $H_{k,1} : \mu_j = \mu_{j'}, H_{k,2} : \mu_j = \mu_{j''}$, and $H_{k,3} : \mu_{j'} = \mu_{j''}$. The falsehood of any one null hypothesis sets limitations on the possible outcomes of the others. Specifically, if one hypothesis is false, then at least another must also be false. To illustrate, if the null parameter domains of μ_j and $\mu_{j'}$, for example, do not intersect, the null domain of $\mu_{j''}$ cannot align with both at the same time. This means the null parameter domain of $\mu_{j''}$ must differ from at least one of the two, impacting the outcomes of the associated hypotheses. Generally, when comparing the means of all possible pairs of J groups, at least $J - 1$ hypotheses will be false, if at least one of them is false.

In real world applications, reaching this expected outcome at a given decision threshold for evidence assessment might not always be possible. This may result from the failure of the procedure to detect genuinely false hypotheses at certain levels of effect size, or sample size. In general, the higher the power of a procedure, considering all other factors constant, the fewer instances there are of failing to detect genuinely false null hypotheses.

4.3 The Testing Procedure

The multiple testing procedure proposed is detailed in algorithm 4.2, which aims to evaluate hypotheses structured hierarchically, as described in Section 4.2. This procedure

navigates through the various nodes within the hierarchy and makes decisions on whether to proceed down the hierarchy based on a preset significance level, α, within the primary tier and when progressing to the follow-up tier. For a comprehensive evaluation of all tests across every node, one can set α to 1.

As a preparatory step, the hierarchically adjusted p-values, represented by $\pi_k^{\text{h-adj}}$ or $\pi_{k,m}^{\text{h-adj}}$, for node k or pairwise combination, m, is set to the value `NA` for all nodes. This serves as an indicator that the hierarchically adjusted p-values are not yet computed. The algorithm initiates at the root node starting with π_1 in the primary tier, and ends with $\pi_{K,M}$ in the furthest node on the right in the follow-up tier. Each p-value undergoes a double adjustment: initially, it is transformed from π to π^{adj} to accommodate the multiple hypotheses tested within the structure. Subsequently, it is adjusted to $\pi^{\text{h-adj}}$ to ensure that the hierarchical structure remains intact among these adjusted p-values. The hierarchically adjusted p-values are then compared to α. If $\pi^{\text{h-adj}} \geq \alpha$, then the hierarchically adjusted p-values at that node is assigned to all of its descendants. Once a value is assigned to $\pi^{\text{h-adj}}$, the algorithm skips that node in the iterative process.

A Bonferroni-type adjustment is applied to π to obtain π^{adj}. It is determined by the formula r_k described in Equation (4.4):

$$r_k = \begin{cases} \dfrac{Q}{|C_k|}, \text{ if } H_k \\ \dfrac{Q(J-1)(J-2)}{2}, \text{ if } H_{k,m} \end{cases} . \tag{4.4}$$

In the primary tier, r_k is the equivalent to $\dfrac{Q}{|C_k|}$ which is similar to the multiple testing adjustment proposed in Meinshausen (2008) for multiple testing when selecting important variables in linear and generalized linear regression settings. Typically, in the standard Bonferroni approach, each node p-value is multiplied by Q, which is the total number of elementary hypotheses under consideration. However, in this case, the adjustment is modified

to reflect that, at some nodes, those elementary hypotheses are considered jointly and tested in a single hypothesis. For instance, at the root node, H_1 incorporates all Q elementary hypotheses that may be tested under the standard Bonferroni approach, permitting us to adjust the value to $Q/Q = 1$. This adjustment is also logical, considering that at the root node, we would have only performed a singular test, obviating the need for multiplicity adjustments.

The standard Bonferroni adjustment applied to the follow-up tier would necessitate an adjustment of $Q\binom{J}{2} = \frac{QJ(J-1)}{2}$ to each $\pi_{k,m}$, given that there are $Q \times \binom{J}{2}$ pairwise hypotheses. However, leveraging the logical constraints offered by the hierarchical structure, we can reduce the adjustment applied to $\pi_{k,m}$. When assessing if at least one Type I error has occurred in the tree for the evaluation of the family-wise error rate, $H_{k,1}, \ldots, H_{k,M}$ become irrelevant if the hypothesis at the terminal node, H_k, is true. This is because if H_k is true, then all its associated pairwise hypotheses must also be true. Consequently, $H_{k,1}, \ldots, H_{k,M}$ would only require examination for a Type I error if H_k is false. Yet, if H_k is false, it implies that at least one among $H_{k,1}, \ldots, H_{k,M}$ is also false, which further suggests the falsehood of at least $J-1$ hypotheses, as discussed in Section 4.2. Hence, for the family-wise error rate in the follow-up tier, the emphasis can shift to the maximum number of true null pairwise hypotheses plausible when the parent H_k is false. This corresponds to $\binom{J}{2} - (J-1)$, leading to a reduced adjustment of $\frac{Q(J-1)(J-2)}{2}$ in the follow-up tier.

4.4 FWER Control

Proposition 4.1. *Let $\mathfrak{T}$ be a tiered-hierarchy of tests, H_k and $H_{k,m}, k = 1, \ldots, K, m = 1, \ldots, \binom{J}{2}$, arranged in a tree according to the discussions in Section 4.3 and consider the testing procedure described in Algorithm 4.2. Let $\mathfrak{T}_0$ be the set of all $H_k, , k = 1, \ldots, K$, and $H_{k,m}, m = 1, \ldots, \binom{J}{2} \in \mathfrak{T}$ where the truth is consistent with the respective nulls. Then,*

Algorithm 4.2: Proposed tiered-hierarchical multiple testing procedure evaluated at level α

$\pi_k^{\text{h-adj}}, \pi_{k,m}^{\text{h-adj}} = \texttt{NA}$, for all $k = 1, \ldots, K$, and $m = 1, \ldots, \binom{J}{2}$

for $k = 1$ to K **do**

if $\pi_k^{\text{h-adj}} = \texttt{NA}$ **then**

Calculate:

(1) Calculate π_k

(2) $\pi_k^{\text{adj}} = \pi_k r_k$

(3) $\pi_k^{\text{h-adj}} = \max(\pi_k^{\text{adj}}, \pi_{\text{par}_k}^{\text{adj}})$

if $\pi_k^{\text{h-adj}} \leq \alpha$ **then**

if $|C_k| = 1$ **then**

(1) Calculate $\pi_{k,m}$, for all $m = 1, \ldots, \binom{J}{2}$

(2) $\pi_{k,m}^{\text{adj}} = \pi_{k,m} r_k$

(3) $\pi_{k,m}^{\text{h-adj}} = \max(\pi_k^{\text{h-adj}}, \pi_{k,m}^{\text{adj}})$

end if

else

$\pi_{\text{desc}_k}^{\text{adj}} = \pi_k^{\text{h-adj}}$

end if

else

Skip k

end if

end for

Pr$(\exists\ H_k \in \mathfrak{T}_0 : \pi_k^{h\text{-}adj} \leq \alpha\ \cup\ \ H_{k,m} \in \mathfrak{T}_0 : \pi_{k,m}^{h\text{-}adj} \leq \alpha) \leq \alpha$*, that is, the family-wise error rate is controlled at level* $\alpha \in (0, 1)$.

Proof We define $\tilde{\mathfrak{T}}_0$ as it follows in Equation (4.5).

$$\tilde{\mathfrak{T}}_0 = \begin{cases} H_k \in \mathfrak{T}_0 : \forall\ H_{k'} \in \mathfrak{T}_0, C_k, k \neq k' \not\subset C_{k'} \\ H_{k,m} \in \mathfrak{T}_0 : H_k \notin \mathfrak{T}_0 \end{cases} \tag{4.5}$$

Simply put, from Equation (4.5), the set of hypotheses in $\tilde{\mathfrak{T}}_0$ are maximal. This means that if H_k is in $\tilde{\mathfrak{T}}_0$, then H_k can not be a descendant of any other null hypothesis in $\tilde{\mathfrak{T}}_0$. Similarly, $H_{k,m}$ can only be in $\tilde{\mathfrak{T}}_0$ if its associated H_k is not in $\mathfrak{T}_0$. The family-wise error rate is defined as

$$\Pr(\exists\ H_k \in \mathfrak{T}_0 : \pi_k^{\text{h-adj}} \leq \alpha\ \cup\ \ H_{k,m} \in \mathfrak{T}_0 : \pi_{k,m}^{\text{h-adj}} \leq \alpha) \tag{4.6}$$

which is equivalent to Equation (4.7) because for every H_k, or $H_{k,m} \in \mathfrak{T}_0/\tilde{\mathfrak{T}}_0$ there exists some ancestor, $H_{\text{ansc}_k} \in \tilde{\mathfrak{T}}_0$. That is, every true null hypothesis in the hierarchy is accounted for in $\tilde{\mathfrak{T}}_0$.

$$\Pr(\exists\ H_k \in \tilde{\mathfrak{T}}_0 : \pi_k^{\text{h-adj}} \leq \alpha\ \cup\ H_{k,m} \in \tilde{\mathfrak{T}}_0 : \pi_{k,m}^{\text{h-adj}} \leq \alpha) \tag{4.7}$$

It is easy to see that the two events in Equation (4.7) are mutually exclusive because if $H_k \in \tilde{\mathfrak{T}}_0$, then $H_{k,m}$ can not be in $\tilde{\mathfrak{T}}_0$, and vice versa. Thus, for $k \neq k'$,

$$= \Pr(\exists\ H_k \in \tilde{\mathfrak{T}}_0 : \pi_k^{\text{h-adj}} \leq \alpha)\ +\ \Pr(\exists\ H_{k',m} \in \tilde{\mathfrak{T}}_0 : \pi_{k',m}^{\text{h-adj}} \leq \alpha) \tag{4.8}$$

$$\leq \Pr(\exists\ H_k \in \tilde{\mathfrak{T}}_0 : \pi_k^{\text{adj}} \leq \alpha)\ +\ \Pr(\exists\ H_{k',m} \in \tilde{\mathfrak{T}}_0 : \pi_{k,m}^{\text{adj}} \leq \alpha) \tag{4.9}$$

$$= \Pr\left\{ \bigcup_{H_k \in \tilde{\mathfrak{T}}_0} \pi_k^{\text{adj}} \leq \alpha \right\} + \Pr\left\{ \bigcup_{H_{k',m} \in \tilde{\mathfrak{T}}_0} \pi_{k',m}^{\text{adj}} \leq \alpha \right\} \tag{4.10}$$

From Boole's inequality,

$$\leq \sum_{H_k \in \tilde{\mathfrak{T}}_0} \Pr(\pi_k^{\text{adj}} \leq \alpha) + \sum_{H_{k',m} \in \tilde{\mathfrak{T}}_0} \Pr(\pi_{k',m}^{\text{adj}} \leq \alpha) \tag{4.11}$$

$$= \sum_{H_k \in \tilde{\mathfrak{T}}_0} \Pr\left(\pi_k \leq \frac{\alpha}{r_k}\right) + \sum_{H_{k',m} \in \tilde{\mathfrak{T}}_0} \Pr\left(\pi_{k',m} \leq \frac{\alpha}{r_k}\right) \tag{4.12}$$

$$= \frac{\alpha}{Q}\left(\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| + \sum_{H_{k',m} \in \tilde{\mathfrak{T}}_0} \frac{2}{(J-1)(J-2)}\right) \tag{4.13}$$

Thus, show to family-wise error rate control, we must show that

$$\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| + \sum_{H_{k',m} \in \tilde{\mathfrak{T}}_0} \frac{2}{(J-1)(J-2)} \leq Q \tag{4.14}$$

We know that

$$\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| \leq Q \tag{4.15}$$

because if $H_1 \in \mathfrak{T}_0$, then $H_k,\ k = 2, \ldots, K$ can not be in $\tilde{\mathfrak{T}}_0$ by definition, and thus $\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| = |C_1| = Q$. However, if $H_1 \notin \mathfrak{T}_0$, then at least one of its children is also not in $\mathfrak{T}_0$ meaning $\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| < Q$. Hence, $\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| \leq Q$. Now, for every $H_{k',m} \in \tilde{\mathfrak{T}}_0$, all of its ancestors are false null hypotheses, and are therefore not in $\mathfrak{T}_0$ because if any of its ancestors are in $\mathfrak{T}_0$, they will be considered maximal which means $H_{k',m} \notin \tilde{\mathfrak{T}}_0$. Therefore by contradiction, we know that the $H_{\text{par}_{k',m}} = H_{k'}$ which is a terminal node, and has a cardinality equal to 1 is false. Thus,

$$\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| \leq Q - \sum_{H_{k'}: H_{k',m} \in \tilde{\mathfrak{T}}_0} 1 \tag{4.16}$$

For any given k', we know from the discussions in Section 4.3 that

$$\sum_{H_{k',m} \in \tilde{\mathfrak{T}}_0} 1 = \frac{(J-1)(J-2)}{2} \tag{4.17}$$

which means that for all k',

$$\sum_{H_{k',m} \in \tilde{\mathfrak{T}}_0} 1 = \frac{(J-1)(J-2)}{2} \sum_{H_{k'}: H_{k',m} \in \tilde{\mathfrak{T}}_0} 1 \tag{4.18}$$

$$\sum_{H_{k'}: H_{k',m} \in \tilde{\mathfrak{T}}_0} 1 = \sum_{H_{k',m} \in \tilde{\mathfrak{T}}_0} \frac{2}{(J-1)(J-2)} \tag{4.19}$$

Hence the proof.

4.4.1 An Adaptive Adjustment in the Follow-Up Tier to Improve Power

The number of true null hypotheses among a family of hypothesis tests can play a pivotal role in multiple testing, but it is rarely not known *a priori*. However, in certain cases, it can be estimated from available information learned in the process of testing. Multiple testing procedures where such an estimate can be obtained may leverage that information to enhance their efficiency and increase power. In our multiple testing scenario, such an estimate can be achieved and employed to further reduce the adjustment in the follow-up tier.

If limited evidence is found for the terminal node hypotheses, testing can be halted at that point, avoiding the need to examine pairwise differences. Instead, attention can be concentrated on nodes that exhibit a stronger signal for detecting differences. Consequently, the r_k adjustment at the subsequent tier can be adaptively reduced when considering the number of true null hypotheses at the terminal nodes. If Q_0 is the count of true nulls, such that $Q_0 \leq Q$, then it can be approximated by $\hat{Q}_0 = \sum I(\pi_{k,\text{terminal}} \geq \alpha)$. The adjustment in the follow-up tier then reduces to $\frac{(Q-\hat{Q}_0)(J-1)(J-2)}{2}$. This adaptive adjustment penalizes less compared to the one discussed in Section 4.3, therefore increasing the power to detect differences. We perform simulation studies in Section 4.5 to show that the family-wise error

rate is still controlled for the procedure with this revised adjustment, and we assess the performance of the proposed adjustment by comparing it to some existing methods.

4.5 Simulations

In this section, we conduct two simulation studies to explore both the error rates and the power properties of our tiered-hierarchical testing method. These simulations aim to empirically test the theoretical assertions made in Section 4.4 regarding the control of the family-wise error rate in the evaluation of group differences across multiple outcomes. The first simulation study focuses on exploring the error rates associated with various adjustment techniques. The second simulation study delves into assessing the power of the testing procedure compared to other procedures.

4.5.1 Simulation Study Setup

We employ a one-way ANOVA model for each outcome to explore various scenarios and assess their effects on family-wise error rates. Specifically, we consider varying three components: J, denoting the number of groups for each outcome; β, the size of the difference in mean between pairs of groups; and signal sparsity, which quantifies the number of outcomes for which a true difference exists between at least one pair of groups. For the number of groups J, we explore three settings: outcomes may have either three, four, or five groups. The specific means for each group are presented in Table 4.1. Each of these groups has a different sample size that is calculated based on the number of groups using the following R code, `round(seq(50, 75, length.out = group_no))` so that each is between 50 and 75, and the differences between any two sample sizes are equal. We consider six different different values for β, which are $0.5, 0.6, 0.7, 0.8, 0.9$, and 1. To generate the data for each group, the specified $Q = 100$ outcomes, $\mathbf{Y_j} = (Y_{j1}, \ldots, Y_{jQ})^T$, are drawn from a multivariate normal distribution with a mean vector, $\boldsymbol{\mu_j} = (\mu_{j1}, \ldots, \mu_j Q)^T$, and variance-covariance matrix, $\boldsymbol{\Sigma}$. The variance

covariance matrix, $\mathbf{\Sigma}$, is configured such that $\text{Cov}(Y_q, Y_{q'}) = 0.2$ and $Var(Y_q) = 1$.

Table 4.1: True mean values assigned to each group in the different group number settings considered

Number of groups	Means for each treatment group
Three groups	0, β, and β for j=1, 2, 3 respectively
Four groups	0, 0, β, and β for j=1, 2, 3, 4 respectively
Five groups	0, 0, β, β, and β for j=1, 2, 3, 4, 5 respectively

Each described scenario is executed 1000 times to estimate both the family-wise error rate and the average statistical power at each terminal node in the follow-up tier, where pairwise comparisons are conducted. To compute the empirical family-wise error rate, we average over 1000 simulation instances, the number of times for which at least one false pairwise detection occurs at any of the terminal nodes in the follow-up tier. Similarly, we compute the average power by summing up over 1000 simulations, the proportion of true detections out of the total number of all detections that are made within a single simulation instance. Both calculations are defined as:

$$\text{FWER} = \sum_{w=1}^{W} \frac{I(\pi_{k,m}^{\text{adj}} < \alpha \cap H_{k,m} \text{ is true}) \text{ at simulation w}}{\text{W}} \tag{4.20}$$

$$\text{Average Power} = \sum_{w=1}^{W} \frac{\sum_{k=1}^{K} \sum_{m=1}^{\binom{J}{2}} I(\pi_{k,m}^{\text{adj}} < \alpha \cap H_{k,m} \text{ is false}) \text{ at simulation w}}{\sum_{k=1}^{K} \sum_{m=1}^{\binom{J}{2}} I(H_{k,m} \text{ is false}) \text{ at simulation w}}. \tag{4.21}$$

4.5.2 FWE Rates

To assess the family-wise error rate and compare it to existing methods, we focus on the hypotheses in the follow-up tier of the hierarchical structure adjusted using the adaptive

tiered-hierarchical testing procedure. The FWER across the described settings is compared to those from a Bonferroni-only adjustment, Bonferroni coupled with Tukey's adjustment, Tukey-only adjustment, and no adjustment whatsoever.

The Bonferroni method adjusts each $\pi_{k,m}$ by multiplying it by $Q\binom{J}{2} = \frac{JQ(J-1)}{2}$, reflecting the total number of pairwise comparisons in the follow-up tier of the hierarchy. The Tukey-Kramer Honest Significant Difference (HSD) test is both a statistical test and a multiple comparison procedure that controls the family-wise error. It evaluates all possible pairs of the M group means for a single outcome using the studentized range distribution. For further information on this adjustment procedure, see Tukey (1949) and Hsu (1996). Although designed for a single outcome, this method can be adapted to handle multiple outcomes by combining it with a Bonferroni adjustment. This involves applying Tukey's HSD to the set of pairwise p-values for each of the Q outcomes and then multiplying each resulting p-value by Q to account for multiple outcomes. In the Tukey-only adjustment, only Tukey's test is performed, specifically for multiple pairwise comparisons within each single outcome, without any adjustment for considering Q outcomes. The no adjustment method maintains each $\pi_{k,m}$ in its original state, providing an uncorrected evaluation without any adjustments for multiple testing.

Figure 4.2 reveals that the family-wise error rate (FWER) for both the Tukey-only and no adjustment methods is severely inflated across all considered scenarios. As density of sparsity of signals decreases from 0.5 to 0.1, the Tukey-only method — which does not incorporate any across-outcome adjustments — closely approaches the no adjustment method which is equal to 1 in most scenarios. Essentially, employing only Tukey's adjustment for pairwise comparisons tests when there are multiple outcomes can be nearly as ineffective as making no adjustments at all. This highlights the vulnerability of addressing only one of the forms of multiplicity, as discussed in the introduction. This is especially pronounced in the context of sparse signals and suggests that additional correction mechanisms across outcomes

are needed to maintain the integrity of hypothesis testing in such situations. However, both the Bonferroni adjustment and the proposed tiered-hierarchical adjustment mitigate the inflation of the FWER, maintaining the desired level of α.

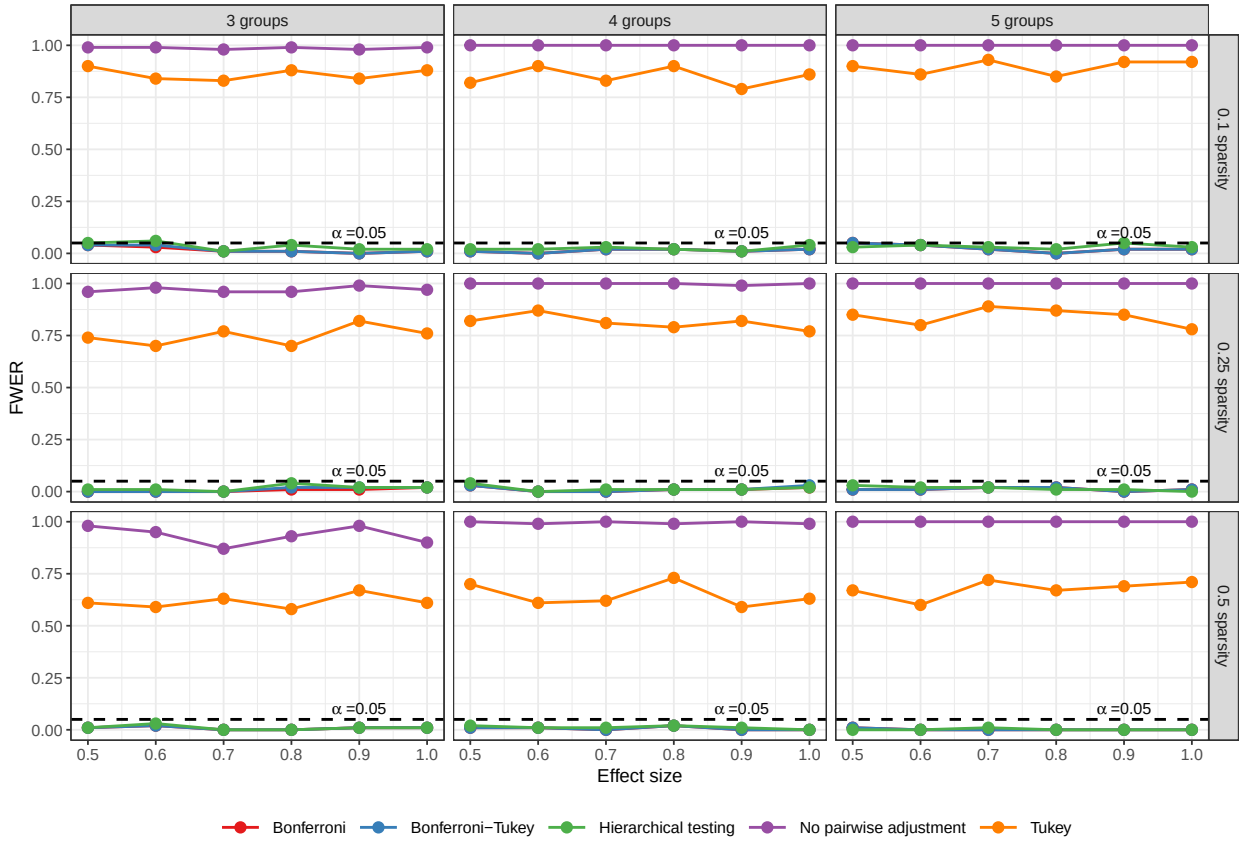


Figure 4.2: Empirical family-wise error rates for all pairwise comparisons perfomed using the Bonferroni, Bonferroni-Tukey, Hierarchical testing method, Tukey-only adjustement, and no adjustment obtained with 1000 simulation iterations on 100 variables for each setting ($\alpha = 0.05$).

4.5.3 Power

To assess the power performance of the adaptive tiered-hierarchical testing procedure, we compare it with the Bonferroni and Bonferroni with Tukey procedures. In Figure 4.3, as the effect size increases, the average power of all tests also increases. This is consistent

with the general understanding that a larger effect size results in greater power to detect differences. However, the proposed testing procedure exhibits better power performance relative to the other two methods. As sparsity increases, the power advantage of the adaptive tiered-hierarchical testing procedure over the other methods becomes even more pronounced, revealing a clear divergence suggesting that the tiered-hierarchical approach may be better equipped to handle sparse signals than the other methods.

Additionally, the plot highlights that as the number of groups or treatments grows, there is a general reduction in power. This trend aligns with the intuitive notion that with an increase in groups, there are more tests to perform, and, as such, stricter adjustments are applied which consequently reduce the power of the tests.

4.6 Dataset Application

In this section, we demonstrate the application of the tiered-hierarchical testing procedure for controlling the family-wise error rate. We consider two datasets, one simulated and the other a real dataset. For the simulated dataset, we examine 100 outcomes across four groups denoted as a, b, c, and d. In 10 randomly chosen outcomes, the groups have population means of 0, 0, 1, and 1, for the four groups respectively, otherwise the group means are 0,0,0,0. The sample sizes for each group are as follows: 50, 58, 67, and 75. The outcomes for each group are $\mathbf{Y_j} = (Y_{j1}, \ldots, Y_{jQ})^T$ and are drawn from a multivariate normal distribution with a mean vector, $\boldsymbol{\mu_j} = (\mu_{j1}, \ldots, \mu_{jQ})^T$, and an arbitrary variance-covariance matrix, $\boldsymbol{\Sigma}$, using the `rcorrmatrix` function from the `clusterGeneration` package (Qiu & Joe., 2020) in R (R Core Team, 2023).

In the second example, we analyze data from an untargeted brain metabolomics study by Petr et al. (2021), aimed at understanding the relationship between aging and various phenotypical, physiological, and functional changes leading to disease onset and mortality in mice. The mice were categorized into three age groups: 'young' (comprising 12 mice under

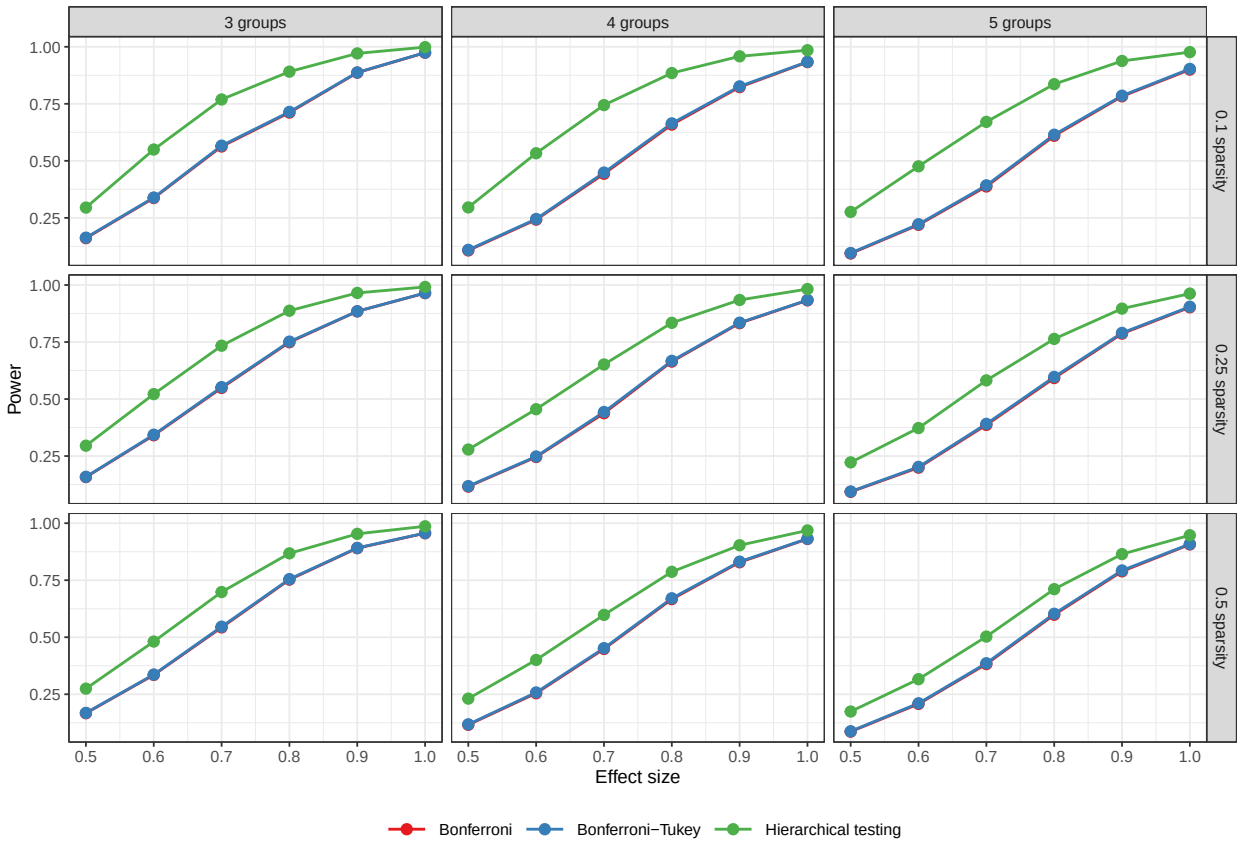


Figure 4.3: Average empirical powers at the follow-up tier for the tiered-hierarchical testing procedure compared to Bonferroni, and Bonferroni-Tukey procedures across different number of groups, difference sizes, and sparsity levels obtained with 1000 simulation iterations on 100 variables for each setting. Generally, the Bonferroni and Bonferroni-Tukey are hard to distinguish in the simulation results.

15 months old), 'mid-age' (consisting of 7 mice aged between 15 and 20 months), and 'old' (including 11 mice over 20 months old). For each mouse in the study, the concentrations of 113 named metabolites were quantified using gas chromatography time-of-flight (GC-TOF) mass spectrometry. The metabolite concentrations were log-transformed and scaled to achieve a mean of 0 and a standard deviation of 1. The primary objective of the study is to assess the differences in metabolite concentrations between pairs of age groups.

These data are available at the NIH Common Fund's National Metabolomics Data

Repository (NMDR) website, the Metabolomics Workbench, https://www.metabolomicswo rkbench.org, where it has been assigned Project ID PR001089. The data can be accessed directly via it's Project DOI: 10.21228/M8XT42 This work is supported by NIH grant, U2C-DK119886. In the study by Petr et al. (2021), an analysis of metabolomic profiles in the heart, liver, serum, and skeletal muscle was also conducted. The metabolites reported in their paper as having interesting insights across mice were determined on all five tissues. However, the exact methods used in their study were not clearly stated, making it challenging to directly compare our results with theirs.

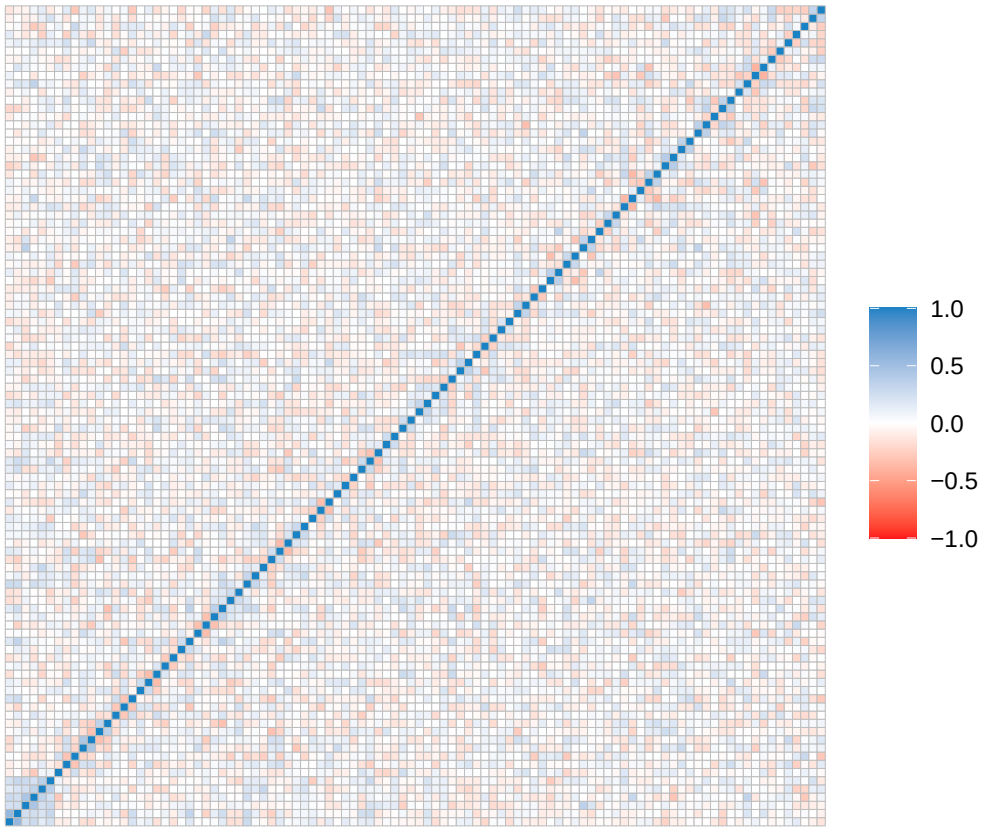


Figure 4.4: Correlogram of the outcome variables from the simulated study. Variables are sorted based on Wards hierarchical cluster analysis with the Spearman absolute value dissimilarity. Darker tiles represent stronger correlations and lighter tiles represent weaker correlations. Red and blue tiles represent negative and positive correlation values respectively.

The correlation plot in Figure 4.4 displays the Spearman correlation coefficients, arranged according to the order obtained from Ward's agglomerative hierarchical clustering of the simulated outcome variables. A distinct cluster of variables in the bottom left demonstrates a strong positive correlation. This pronounced correlation is expected and likely represents a signal of those outcomes to exhibit differences across the groups. Given the absence of other sources of variation and random source of variation, potential signals become evident in this visualization. Figure 4.5 exhibits strong positive correlations, with a few distinct clusters, as well as many negatively correlated metabolites.

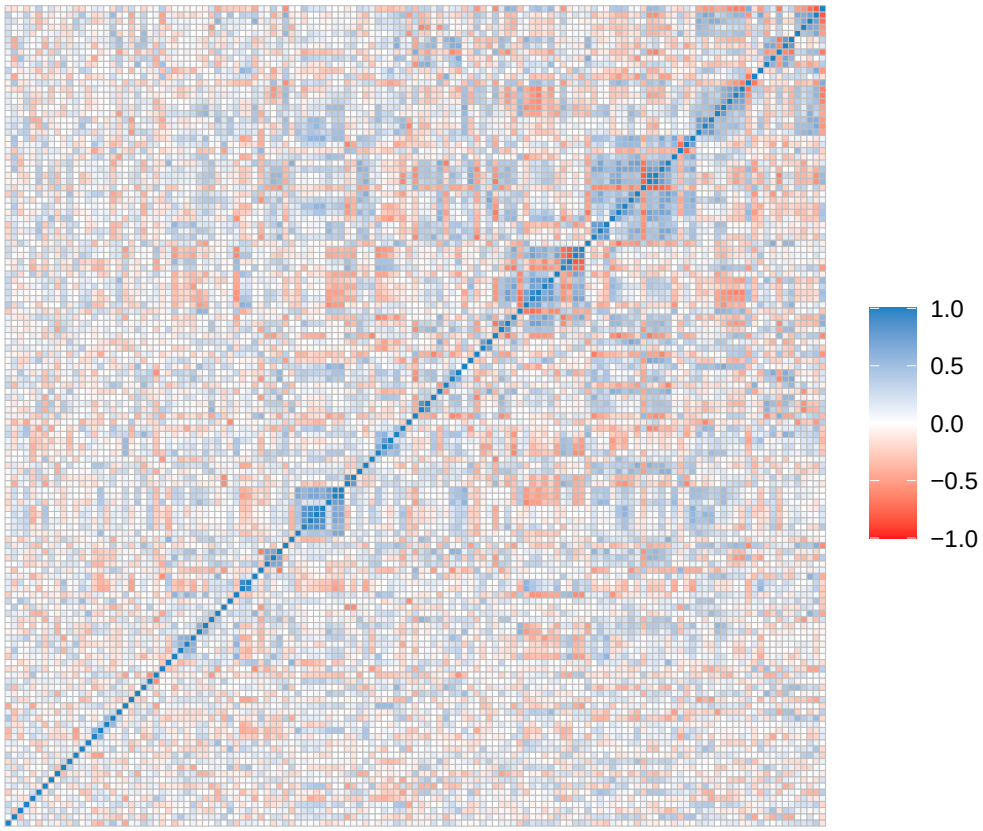


Figure 4.5: Correlogram of the log-transformed and standardized metabolite concentrations from the mice study. Metabolites are sorted based on Wards hierarchical cluster analysis with the Spearman absolute value dissimilarity. Darker tiles represent stronger correlations and lighter tiles represent weaker correlations. Red and blue tiles represent negative and positive correlation values respectively.

Figure 4.6 illustrates the results of the tiered-hierarchical testing procedure. At the top is a dendrogram, while a grid is situated below. The dendrogram depicts the p-values for hypotheses in the primary tier, and the grid showcases the p-values from the pairwise hypothesis tests of the follow-up tier. Nodes shaded in gray indicate p-values that were equal to or exceeded the threshold α, signifying areas where evidence was insufficient to proceed further testing downwards. In this diagram, we note that 10 variables made it through the entire testing process, revealing the outcomes that we simulated to exhibit differences between groups. For those variables we also see that the differences occurred between groups 1 and 3, 1 and 4, 2 and 3, and 2 and 4, aligning with our simulation design.

The age study shown in Figure 4.7 reveals that three metabolites, hexitol, dihydrocholesterol, and alpha-tocopherol, were detected as differing between middle-aged and young mice, as well as between old and young mice, suggesting that young mice are different from both middle-aged and old mice for those metabolites.

We compared the distribution of adjusted pairwise p-values from the hierarchical testing method to those obtained from the standard Bonferroni and another where the Bonferroni and Tukey procedures were used in conjunction. The results are illustrated in the plot shown in Figure 4.8, which displays the log10-transformed adjusted p-values for each method across different pairwise groups. Notably, the distribution of p-values across methods is not substantially varied. This is primarily because we do not have many outcomes with detectable differences; hence, the number of large p-values clouds any distinct pattern that might be gleaned from the data. More importantly, we observe that the -log10 adjusted p-values are higher for the hierarchical testing, signifying smaller p-values for this method. This highlights the enhanced power gained with the hierarchical testing method compared to the other methods under consideration.

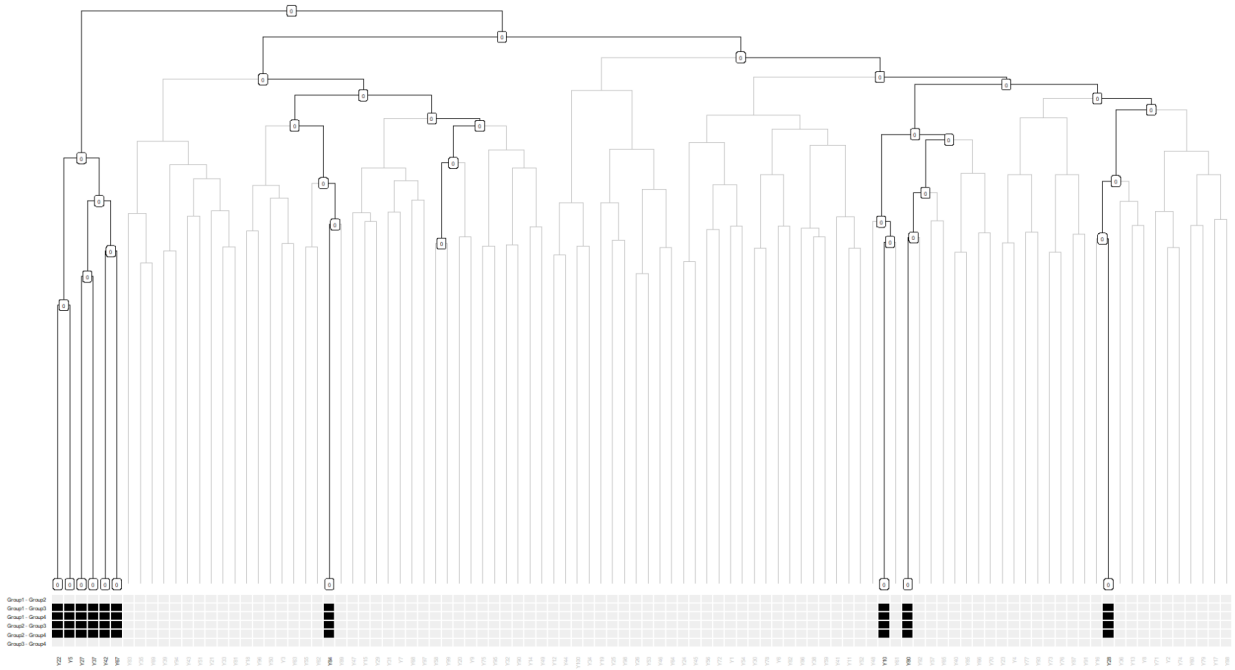


Figure 4.6: Tiered-hierarchy shows the hierarchically adjusted p-values from the simulated data. Greyed out nodes signify hierarchically adjusted p-values that were greater or equal to α

4.7 Discussion

In this paper, we have developed a multiple testing procedure tailored for hierarchically ordered families of hypotheses. Our hierarchical testing framework comprises two tiers: the primary tier and the follow-up tier. In the primary tier, we assess hypotheses concerning inter-group differences for a set of outcomes. This evaluation begins with all outcomes at the top level and progressively moves down to individual outcomes in the terminal nodes. Should strong evidence against the null hypothesis emerge at any terminal node, we proceed to the follow-up tier to assess pairwise differences. Our procedure capitalizes on two key elements: the correlations among outcomes and the logical constraints inherent to the hierarchical structure.

Leveraging outcome correlations allows us to group similar hypotheses, facilitating signal

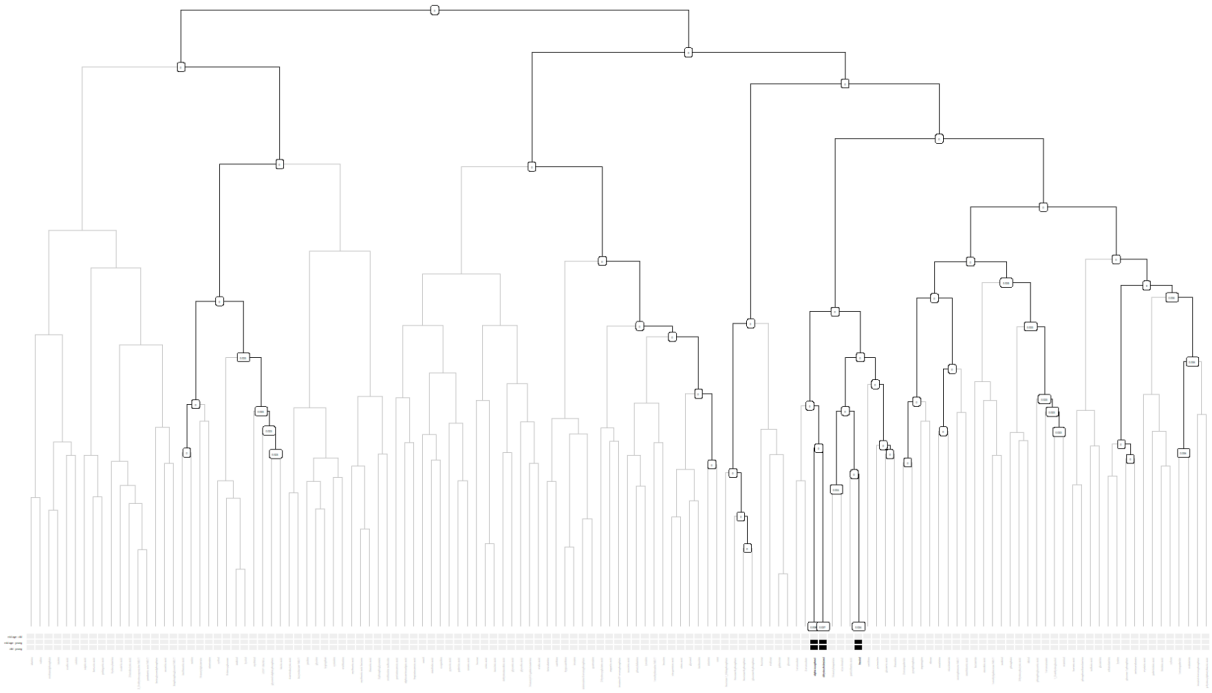


Figure 4.7: Tiered-hierarchy shows the hierarchically adjusted p-values from the mice dataset. Greyed out nodes signify hierarchically adjusted p-values that were greater or equal to α

detection and enhancing interpretability. Meanwhile, the logical constraints provide insights into hypotheses likely to be false, thus reducing the required adjustment for p-values and increasing statistical power. Theoretically, and through simulations, we demonstrate that our procedure controls the family-wise error rate across all nodes of the hierarchy simultaneously. Moreover, our method outperforms in terms of power when compared to existing methods that also control FWER. However, this comes with a computational trade-off, given the extensive testing required across all nodes in the hierarchy. Although we implement decision rules to mitigate this issue, the sheer volume of outcomes may still pose challenges. Especially when dealing with thousands of outcomes, our method might run slower in comparison to other methods.

It is important to note that the enhancements in our multiple testing procedure are based in a logical analysis of the relationships among hypotheses and remain independent

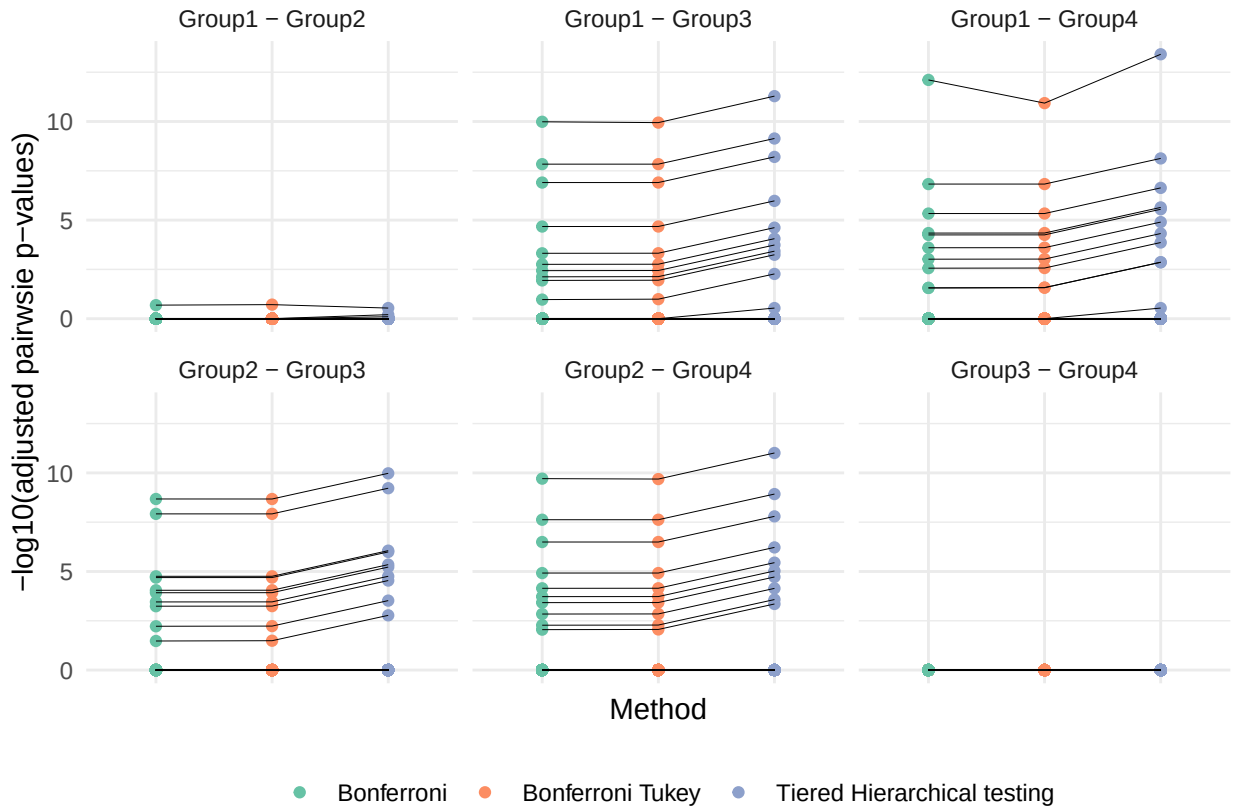


Figure 4.8: Spagetti plots showing the -log10 adjusted p-values using the Bonferroni, Bonferroni-Tukey, and the Tiered Hierarchical testing methods for each of the pairwise group differences.

of the specific test statistics employed, except for knowledge of their respective marginal distributions. Consequently, our methods offer flexibility and applicability in nonstandard scenarios. Other approaches to multiple testing leverage more robust techniques based on the joint distribution of test statistics, such as the Westfall-Young method (Thomas et al., 1994) can be combined with logical considerations to yield even more power. Additional logical constraints may be identified within various chosen structures, and it is possible that superior constraint frameworks exist that can be harnessed to further enhance statistical power.

Exploring other extensions to the method can enhance its applicability in practice. For instance, the method could be extended to handle tests involving multiple explanatory

variables, that allow for, say, the examination of interaction effects followed by main effects or all pairs of combinations of groups within the framework of a two-way ANOVA design. Furthermore, more complex models such as mixed models, which may incorporate random effects to account for additional sources of known variation, can be considered. In a broader sense, the general testing procedure is adaptable to accommodate a single "effect" per hierarchy, delving deeper into specific nodes to address the nuances of the research problem. Alternatively, multiple tiers can be employed to address the underlying structure of the study design.

There are computational gains from considering decision thresholds and making a decision at each node, typically this is with sparse signals, If there are only a few outcomes in the experiment that genuinely exhibit difference across outcomes, it not be wise to make.

CHAPTER FIVE

DISCUSSIONS AND EXTENSIONS

In this dissertation, we explore correlation-based hierarchical multiple testing procedures to control the family-wise error rate in settings where groups or treatments are compared across a multitude of outcomes. We propose procedures that leverage structural information in formulating, ordering, and testing logically related hypotheses to enhance power. These procedures are based on the Bonferroni method. The standard Bonferroni procedure anticipates the worst-case scenario involving all hypotheses being true, and offers a multiple testing adjustment aimed at controlling the family-wise error rate under this scenario. However, when we introduce structure, we can incorporate information about the relevant dependence between hypotheses. This helps identify non-feasible combinations of true and false null hypotheses, and target only those hypotheses that can be true when considering the family-wise error rate. This refined focus allows us to reduce the adjustment and thus increase power.

To introduce structure into the procedures, we choose to use agglomerative hierarchical variable clustering. The results of agglomerative hierarchical clustering depend on two primary factors: how variables are determined to be close, often quantified by a dissimilarity measure, and the criteria used to decide clusters to be joined together, determined with the linkage criteria. The choices of dissimilarity and linkage can considerably influence the clustering outcome, and provide a hierarchical structure that may lead to different multiple testing results. The clustering outcome is also crucial for how the multiple testing results are interpreted because outcomes variables can end up different clusters for different clustering input choices. Generally, there is no singular "correct" solution for clustering, so our goal is to choose a dissimilarity measure and a linkage criteria that produce clusters that can foster

practical insights.

In Chapter 2, we discuss options for dissimilarity measures, distinguishing between those that are suitable for variable clustering and those that are not. We advocate for measures that are both magnitude and directionally agnostic, ensuring that variables cluster based on shared information rather than magnitude or direction of change. Why is this important? Consider untargeted metabolomic studies as an illustrative example. In such studies, metabolites may manifest at different peak intensities, resulting in varying magnitudes. Magnitudes mainly reveal the quantities present, not necessarily how metabolites behave in relation to one another. Therefore, clustering based on magnitude alone might not yield meaningful or beneficial insights. Similarly, metabolites that exhibit similar behavior but in opposing directions should ideally cluster together. This is because their shared behavior pattern, regardless of the direction, is more informative than mere directional changes.

Additionally, Chapter 2 provides an exploration of various linkage criteria, outlining the characteristics and limitations of each. Further exploration of the impact of linkage criteria is explored in Chapter 3, where we evaluate how various criteria influence the family-wise error rates, and the power of the hierarchical testing procedure. Other clustering methods beyond hierarchical clustering algorithms that are suitable for correlation-based dissimilarities are discussed as well in Chapter 2.

The proposed hierarchical testing procedures are outlined in Chapters 3 and 4. We first examine the simple case of comparing two groups across numerous outcomes. Although the method discussed was initially discussed by Meinshausen (2008) in the context of linear regression, we adapt it to our group comparison context and delve into the necessary choices for the hierarchical testing approach in this scenario. We demonstrate that the family-wise error rate remains controlled, and this method provides enhanced power compared to certain existing techniques.

A logical next step is to extend this method to accommodate more than two groups. In

Chapter 4, we undertake this extension, and develop new multiple testing adjustments to fit the updated testing scenario. Broadly, the structure comprises two tiers: multiple group comparisons and multiple outcome testing. The first tier seeks to identify outcomes with detectable group differences, while the second pinpoints the specific groups manifesting these differences. Through simulations and a theoretical proof, we show that the family-wise error rate (FWER) remains controlled across all hierarchical levels, and highlight improved power performance relative to other methods.

5.1 Extensions to More Complex Study Designs

The study designs examined in this dissertation represent an introductory glimpse. While we have focused on tests involving a single factor, some studies embrace more complex factorial designs, often necessitating consideration of controlled factors or sources of known variation. In this section, we explore the potential for broadening the hierarchical testing framework into some of these scenarios.

We can consider designs involving two or more factors, each of which is of importance to the researcher. Typically, such research endeavors might be concerned with interaction effects. In these situations, we might consider consolidating the factors into a single factor which align with the scope of the procedures discussed in the dissertation. However, this approach might not always yield the desired insights for the researcher. At times, if there is little evidence of an interaction effect, the researcher may then wish to ascertain if some marginal effect exists. In such instances, the previously outlined procedure may fall short. A similar two-tier structure like the one described in Chapter 4 might be adapted to address this situation, where the first tier assesses interaction effects and the second tier evaluates main effects. However, new multiple testing adjustment will need to be considered for this scenario, and possible new logical constraints may arise that can be taken into consideration to enhance power.

As the sets of test-statistics or p-values of interest grows, a "flat" or optimization clustering technique—which does not require recursive partitioning of the outcome variables—might be better suited to evaluate the correlations between outcomes. It might be prudent to explore partitioning methods to reduce the number of nodes at which hypotheses are being tested. Although, any selected clustering approach must accommodate correlation-based dissimilarity measures, so this limits options for non-hierarchical methods. Opting for a flat structure can also offer computational benefits, reducing the complexity inherent to hierarchical structures.

Addressing additional sources of variation demands careful consideration of their impact on the correlations among outcomes. Since clustering aims to group outcomes based on these correlations by grouping signals with similar strength of evidence against their respective null hypotheses, extra sources of variation can complicate matters. This added complexity might result in clusters that do not accurately represent the signals we aim to detect. Although the family-wise error rate may remain controlled, there is a possibility of missing out on the complete narrative or interpretation. In such cases, one may explore the option to "purify" correlations among outcomes by separating out known sources of variation before attempting to use them in assessing proximity among outcomes. Although, our initial explorations did not show sufficiently promising results to pursue here, others may find some inspiration in the direction.

5.2 Looking Beyond the Bonferroni Method and the FWER

The enhancements in power that our proposed methods for multiple hypothesis testing offer are based on an understanding of the logical relationships among hypotheses and are not influenced by the specific nature of the test statistics used. More specifically, we only rely on knowledge about the distribution of the parameter(s) investigated at each node when assessing evidence but not when adjusting evidence. As a result, our methods are more

versatile compared to, for instance, re-sampling methods like the Westfall-Young procedure (Thomas et al., 1994), whose strategy incorporates distributional characteristics of the entire family of test statistics. However, it may be beneficial to combine logical analysis with a consideration of such distributional characteristics, as the resulting procedures could be superior in enhancing power.

Previously, we highlighted that the procedures introduced in this work draw from the Bonferroni method. To enhance power, it may be worth exploring alternative existing procedures for the foundational approach, such as a Holm-based approach, given that the Holm procedure consistently outperforms the standard Bonferroni procedure. Moreover, it may be advantageous to shift the focus from controlling the family-wise error rate to a more lenient overall measure, such as the false discovery rate. Both the Holm procedure and false discovery rate procedures, including the Benjamini-Hochberg method (Benjamini & Bogomolov, 2014), utilize a step-wise approach, organizing the p-values in a specific order. However, integrating these methods directly into a hierarchical framework can pose challenges, as the hypotheses within this framework are already structured within the hierarchy. A strategy that has been implemented in hierarchical testing under the false discovery rate involves treating each hierarchical level as a distinct family and applies these methods to ordered p-values at each level (Bogomolov et al., 2020; Yekutieli, 2008). In the context of our case, this approach would mean considering groups such as the first-generation offspring (children of the root node), the second generation (grandchildren), and so forth, as families. Future research may consider these directions to build on the work explored here.

REFERENCES CITED

Arias-Castro, E., & Ying, A. (2019). Detection of sparse mixtures: Higher criticism and scan statistic. *Electronic Journal of Statistics*, *13* (1).

Barko, P. C., & Williams, D. A. (2021). Untargeted analysis of the serum metabolome in cats with exocrine pancreatic insufficiency. *PLoS ONE*, *16* (9), 1–17. *https://doi.org/10.1371/journal.pone.0257856*

Barnett, I. et al. (2017). The generalized higher criticism for testing SNP-set effects in genetic association studies. *Journal of the American Statistical Association*, *112* (517), 64–76.

Benjamini, Y., & Bogomolov, M. (2014). Selective inference on multiple families of hypotheses. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, *76* (1), 297–318.

Berk, R. H., & Jones, D. H. (1979). Goodness-of-fit test statistics that dominate the kolmogorov statistics. *Z. Wahrscheinlichkeitstheorie Verw Gebiete*, *47*, 47–59.

Bogomolov, W. et al. (2020). Hypotheses on a tree: New error rates and testing strategies. *Biometrika. https://doi.org/10.1093/biomet/asaa086*

Chavent, M. (1998). A monothetic clustering method. *Pattern Recognition Letters*, *19* (11), 989–996. *https://doi.org/10.1016/s0167-8655(98)00087-7*

Donoho, D., & Jin, J. (2004). Higher criticism for detecting sparse heterogeneous mixtures. *The Annals of Statistics*, *32* (3), 962–994.

Dunn, O. J. (1961). Multiple Comparisons among Means. *Journal of the American Statistical Association*, *56* (293), 52–64. *https://doi.org/10.1080/01621459.1961.10482090*

Dunnett, C. W. (1955). A multiple comparison procedure for comparing several treatments with a control. *Journal of the American Statistical Association*, *50* (272), 1096–1121.

Efron, B. (2007). Correlation and large-scale simultaneous significance testing. *Journal of the American Statistical Association*, *102* (477), 93–103.

Eisner, R. et al. (2010). Learning to predict cancer-associated skeletal muscle wasting from 1H-NMR profiles of urinary metabolites. *Metabolomics*, *7* (1), 25–34. *https://doi.org/10.1007/s11306-010-0232-9*

Everitt, B. (2011). Cluster analysis (5th ed.). Chichester, West Sussex, U.K: Wiley.

Fahrmann, J. et al. (2015). Systemic alterations in the metabolome of diabetic NOD mice delineate increased oxidative stress accompanied by reduced inflammation and hypertriglyceremia. *American Journal of Physiology: Endocrinology and Metabolism*, *308 (11), E978–E989.*

Feldmann, M. J. et al. (2021). Average semivariance yields accurate estimates of the fraction of marker-associated genetic variance and heritability in complex trait analyses. *PLOS Genetics*, *17(8), e1009762. https://doi.org/10.1371/journal.pgen.1009762*

Finner, H., & Roters, M. (2001). On the false discovery rate and expected type i errors. *Biometrical Journal*, *43(8), 985–1005.*

Fortenbach, C. R. et al. (2023). Metabolic and proteomic indications of diabetes progression in human aqueous humor. *PLoS ONE*, *18 (1), e0280491. https://doi.org/10.1371/journal.pone.0280491*

Frey, B. J., & Dueck, D. (2007). Clustering by Passing Messages Between Data Points. *Science*, *315(5814), 972–976. https://doi.org/10.1126/science.1136800*

Galili et al. (2017). Heatmaply: An r package for creating interactive cluster heatmaps for online publishing. *Bioinformatics. https://doi.org/10.1093/bioinformatics/btx657*

Galili, T. (2015). Dendextend: An r package for visualizing, adjusting, and comparing trees of hierarchical clustering. *https://doi.org/10.1093/bioinformatics/btv428*

Ganguly, D. A. S., Samit AND Finkelstein. (2021). Metabolomic and transcriptomic analysis reveals endogenous substrates and metabolic adaptation in rats lacking Abcg2 and Abcb1a transporters. *PLoS ONE*, *16 (7), 1–33. https://doi.org/10.1371/journal.pone.0253852*

Garnier et al. (2021). viridis - colorblind-friendly color maps for r. *https://doi.org/10.5281/zenodo.4679424*

Givoni, I. et al. (2012). Hierarchical affinity propagation. *https://doi.org/10.48550/ARXIV.1202.3722*

Goeman, J. J., & Finos, L. (2012). The inheritance procedure: Multiple testing of tree structured hypotheses. *Statistical Applications in Genetics and Molecular Biology*, *11 (1). https://doi.org/10.1515/1544-6115.1554*

Goeman, J. J., & Solari, A. (2010). The sequential rejection principle of familywise error control. *The Annals of Statistics*, *38* (6). *https://doi.org/10.1214/10-aos829*

Guo, W. (2009). A note on adaptive Bonferroni and Holm procedures under dependence. *Biometrika*, *96* (4), 1012–1018. *https://doi.org/10.1093/biomet/asp048*

Härdle, W. K., & Simar, L. (2015). Applied multivariate statistical analysis. Springer Berlin Heidelberg. *https://doi.org/10.1007/978-3-662-45171-7*

Hartigan, J. A. (1975). Clustering algorithms. New York: Wiley.

Hernandez, G. V. et al. (2020). Dysregulated FXR-FGF19 signaling and choline metabolism are associated with gut dysbiosis and hyperplasia in a novel pig model of pediatric nash. *American Journal of Physiology-Gastrointestinal and Liver Physiology*, *318 (3). https://doi.org/10.1152/ajpgi.00344.2019*

Holm, S. (1979). A simple sequentially rejective multiple test procedure, *6 (2), 65–70.*

Hotelling, H. (1931). The generalization of student's ratio. *The Annals of Mathematical Statistics*, *2 (3), 360–378.*

Hsu, J. (1996). Multiple comparisons. *https://doi.org/10.1201/b15074*

Hu, J. X. et al. (2010). False discovery rate control with groups. *Journal of the American Statistical Association*, *105 (491), 1215–1227*.

James, G. et al. (2013). An introduction to statistical learning (Vol. 112). Springer.

Jorge-Smeding, E. et al. (2022). Untargeted metabolomics confirms the association between plasma branched chain amino acids and residual feed intake in beef heifers. *PLoS ONE*, *17 (11), e0277458. https://doi.org/10.1371/journal.pone.0277458*

Kassambara, A. (2022). Ggcorrplot: Visualization of a correlation matrix using 'ggplot2'. Retrieved from *https://CRAN.R-project.org/package=ggcorrplot*

Kaufman, L., & Rousseeuw, P. J. (1990). Finding groups in data. Wiley Series in Probability and Statistics. *https://doi.org/10.1002/9780470316801*

Kolde, R. (2019). Pheatmap: Pretty heatmaps. Retrieved from *https://CRAN.R-project.org/package=pheatmap*

Konopka, B. M. et al. (2018). Exploratory data analysis of a clinical study group: Development of a procedure for exploring multidimensional data. *PLoS ONE*, *13 (8), e0201950–e0201950*.

Kramer, C. Y. (1956). Extension of multiple range tests to group means with unequal numbers of replications. *Biometrics*, *12 (3), 307–310*.

Kunas, S. L. et al. (2022). Evidence for a hijacked brain reward system but no desensitized threat system in quitting-motivated smokers: An fMRI study. *Addiction*, *117 (3), 701–712*.

Lamrous, S., & Taileb, M. (2006). Divisive hierarchical k-means. 2006 International Conference on Computational Intelligence for Modelling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce (CIMCA'06). *https://doi.org/10.1109/cimca.2006.89*

Landau, S. et al. (2011). Cluster analysis. John Wiley & Sons.

Langfelder, P., & Horvath, S. (2012). Fast {r} functions for robust correlations and hierarchical clustering, 46. Retrieved from *http://www.jstatsoft.org/v46/i11/*

Larson, R. C., & Sadiq, G. (1983). Facility Locations with the Manhattan Metric in the Presence of Barriers to Travel. *Operations Research*, *31 (4), 652–669. https://doi.org/10.1287/opre.31.4.652*

Legendre, P., & Legendre, L. (2012). Numerical ecology. Elsevier.

Li, J., & Siegmund, D. (2015). Higher criticism: P-values and criticism. *The Annals of Statistics*, *43 (3), 1323–1350*.

Liu, N. et al. (2022). Bidirectional and parallel relationships in macaque face circuit revealed by fMRI and causal pharmacological inactivation. *Nature Communications*, *13 (1),*

1–16.

Lu, J. F. et al. (2008). Hierarchical initialization approach for K-Means clustering. *Pattern Recognition Letters*, *29 (6), 787–795. https://doi.org/10.1016/j.patrec.2007.12.009*

Maag, D. et al. (2015). Maize Domestication and Anti-Herbivore Defenses: Leaf-Specific Dynamics during Early Ontogeny of Maize and Its Wild Ancestors. *PLoS ONE*, *10 (8), e0135722. https://doi.org/10.1371/journal.pone.0135722*

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

Maechler, M. et al. (2022). Cluster: Cluster analysis basics and extensions. Retrieved from *https://CRAN.R-project.org/package=cluster*

Mandozzi, J., & Bühlmann, P. (2016a). A sequential rejection testing method for high dimensional regression with correlated variables. *The International Journal of Biostatistics*, *12 (1), 79–95.*

Mandozzi, J., & Bühlmann, P. (2016b). Hierarchical testing in the high-dimensional setting with correlated variables. *Journal of the American Statistical Association*, *111 (513), 331–343.*

Meijer, R. J., & Goeman, J. J. (2014). A multiple testing method for hypotheses structured in a directed acyclic graph. *Biometrical Journal*, *57 (1), 123–143. https://doi.org/10.1002/bimj.201300253*

Meinshausen, N. (2008). Hierarchical testing of variable importance. *Biometrika*, *95 (2), 265–278. https://doi.org/10.1093/biomet/asn007*

Metabolomics Workbench. (2020). PR001089. *https://doi.org/10.21228/M8XT42*

Murtagh, F., & Legendre, P. (2014). Ward's Hierarchical Agglomerative Clustering Method: Which Algorithms Implement Ward's Criterion? *Journal of Classification*, *31 (3), 274–295. https://doi.org/10.1007/s00357-014-9161-z*

Nandi, S. et al. (2021). Adapting to one- and two-way classified structures of hypotheses while controlling the false discovery rate. *Journal of Statistical Planning and Inference*, *215, 95–108. https://doi.org/10.1016/j.jspi.2021.02.006*

Neuwirth, E. (2022). RColorBrewer: ColorBrewer palettes. Retrieved from *https://CRAN.R-project.org/package=RColorBrewer*

Nishiyama, T. et al. (2014). Recent developments of multivariate multiple comparisons among mean vectors. *SUT Journal of Mathematics*, *50 (2). https://doi.org/10.55937/sut/1424793883*

Pang, Z. et al. (2021). MetaboAnalyst 5.0: narrowing the gap between raw spectra and functional insights. *Nucleic Acids Research*, *49 (W1), W388–W396. https://doi.org/10.1093/nar/gkab382*

Petr, M. A. et al. (2021). A cross-sectional study of functional and metabolic changes during aging through the lifespan in male mice. *eLife*, *10*. *https://doi.org/10.7554/elife.62952*

Pietrzykowski, R., & Zieliński, W. (2004). A new procedure of multivariate multiple comparisons.

Pollard, K. S. et al. (2005). Multiple testing procedures: R multtest package and applications to genomics, in bioinformatics and computational biology solutions using r and bioconductor.

Qiu, W., & Joe., H. (2020). clusterGeneration: Random cluster generation (with specified degree of separation). Retrieved from *https://CRAN.R-project.org/package=clusterGeneration*

R Core Team. (2023). R: A language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing. Retrieved from *https://www.R-project.org/*

Sankaran, K., & Holmes, S. (2014). structSSI: Simultaneous and selective inference for grouped or hierarchically structured data. *Journal of Statistical Software*, *59 (13), 1.*

Scheffe, H. (1999). The analysis of variance (Vol. 72). John Wiley & Sons.

Schloerke, B. et al. (2021). GGally: Extension to 'ggplot2'. Retrieved from *https://CRAN.R-project.org/package=GGally*

Schubert, E. (2021). HACAM: Hierarchical agglomerative clustering around medoids-and its limitations. In LWDA (pp. 191–204).

Sesia, M. et al. (2020). Multi-resolution localization of causal variants across the genome. *Nature Communications*, *11 (1). https://doi.org/10.1038/s41467-020-14791-2*

Shaffer, J. P. (1986). Modified sequentially rejective multiple test procedures. *Journal of the American Statistical Association*, *81 (395), 826–831.*

Sievert, C. (2020). Interactive web-based data visualization with r, plotly, and shiny. Retrieved from *https://plotly-r.com*

Simes, R. J. (1986). An improved bonferroni procedure for multiple tests of significance. *Biometrika*, *73 (3), 751–754.*

Sokal, R., & Michener, C. (1958). University of Kansas. A statistical method for evaluating systematic relationships. University of Kansas Science Bulletin. University of Kansas.

Sultan, F. et al. (2022). Temporal analysis of melanogenesis identifies fatty acid metabolism as key skin pigment regulator. *PLOS Biology*, *20 (5), e3001634. https://doi.org/10.1371/journal.pbio.3001634*

Sun, & Cai, T. T. (2009). Large-scale multiple testing under dependence. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, *71 (2), 393–424.*

Sun, & Lin, X. (2020). Genetic variant set-based tests using the generalized berk-jones statistic with application to a genome-wide association study of breast cancer. *Journal of the*

American Statistical Association, *115 (531), 1079–1091* .

Szalkai, B. (2013a). An implementation of the relational k-means algorithm. *https://doi.org/10.48550/ARXIV.1304.6899*

Szalkai, B. (2013b). Generalizing k-means for an arbitrary distance matrix. *https://doi.org/10.48550/ARXIV.1303.6001*

Thomas, G. E. et al. (1994). Resampling-based multiple testing: Examples and methods for p-value adjustment. *The Statistician*, *43 (2), 347. https://doi.org/10.2307/2348369*

Tippett, L. H. C. et al. (1931). The methods of statistics. The Methods of Statistics.

Tran, T. et al. (2021). monoClust: Perform monothetic clustering with extensions to circular data. Retrieved from *https://CRAN.R-project.org/package=monoClust*

Tukey, J. W. (1949). Comparing individual means in the analysis of variance. *Biometrics*, *5 (2), 99. https://doi.org/10.2307/3001913*

Tukey, J. W. (1953). The problem of multiple comparisons. Unpublished manuscript. In the collected works of john w. Tukey VIII., Multiple Comparisons: 1948-1983 1-300.

Veeramohan, R. et al. (2023). Comparative metabolomics analysis reveals alkaloid repertoires in young and mature Mitragyna speciosa (Korth.) Havil. Leaves. *PLoS ONE*, *18 (3), e0283147. https://doi.org/10.1371/journal.pone.0283147*

Vera, J. F., & Macías, R. (2021). On the Behaviour of K-Means Clustering of a Dissimilarity Matrix by Means of Full Multidimensional Scaling. *Psychometrika*, *86 (2), 489–513. https://doi.org/10.1007/s11336-021-09757-2*

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

Warne, R. (2014). A primer on multivariate analysis of variance (MANOVA) for behavioral scientists. *Practical Assessment, Research & Evaluation*, *19, 17–17.*

Warnes, G. R. et al. (2022). Gplots: Various r programming tools for plotting data. Retrieved from *https://CRAN.R-project.org/package=gplots*

Wasserstein, R. L., & Lazar, N. A. (2016). The ASA Statement on p-Values: Context, Process, and Purpose. *The American Statistician*, *70 (2), 129–133. https://doi.org/10.1080/00031305.2016.1154108*

Wei, T., & Simko, V. (2021). R package 'corrplot': Visualization of a correlation matrix. Retrieved from *https://github.com/taiyun/corrplot*

Welch, B. L. (1947). The generalization of 'Student's' problem when several population variances are involved. *Biometrika*, *34 (1-2), 28–35. https://doi.org/10.1093/biomet/34.1-2.28*

Wickham, H. (2016). ggplot2: Elegant graphics for data analysis. Retrieved from *https://ggplot2.tidyverse.org*

Yekutieli, D. (2008). Hierarchical false discovery rate-controlling methodology. *Journal of the American Statistical Association*, *103 (481), 309–316*.

APPENDICES

APPENDIX A

PRIMER FOR THE HIERMT PACKAGE

Introduction

In research it is common to encounter scenarios where many null hypotheses must be tested, but only a small fraction are genuinely false. In such contexts, controlling overall Type I error rates, such as the family-wise error rate, becomes paramount. At the same time, it is essential to maintain a sufficient level of statistical power to identify true signals. Traditional multiple testing adjustment methods can often be overly stringent, leading to few true detections, if any. For instance, the Bonferroni procedure assumes all hypotheses are true nulls, an assumption not always feasible to verify in advance. To enhance statistical power, it is often necessary to incorporate additional structure.

Motivated by applications in metabolomics, especially the analysis of metabolite concentrations or peak intensities across subject groups, we present the `hiermt` R package that introduces correlation-based hierarchical methods for family-wise error rate control with correlated outcomes. These methods leverage the pairwise absolute correlations among outcome variables to construct a hierarchical structure that reflects their interdependence. The method then categorizes hypotheses for various outcome clusters to delineate hypothesis relationships. The hierarchical structure imposes logical constraints on the combinations of possible true and false hypotheses, thereby making it so that the multiple testing adjustments can be reduced, and the power of multiple testing method can be improved. Additionally, visualizing this hierarchy can expose dependencies among hypotheses, making results more interpretable.

Package Overview

The `hiermt` package is currently available on GitHub, To install the package, the `remotes` package can be employed, which specializes in downloading R packages directly from GitHub. By using the `install_github` function from the `remotes` package, you can

easily obtain and install the `hiermt` package. Once installed, the package can be loaded into your session using the `library` function, as demonstrated in the following code example.

```
remotes::install_github("priscilla-bacino/hiermt")

library(hiermt)
```

This package offers two main functions. The principal function, **`hiermt`**, generates a 'hiermt object' responsible for performing multiple testing procedures and showcasing relevant attributes regarding the relationships in the hierarchy including the hierarchically adjusted p-values. An auxiliary function, **`plot.hiermt`**, produces a visual dendrogram that illustrates the hierarchical arrangement of the adjusted p-values. The syntax for the `hiermt()` function is presented in the following code snippet and the arguments for `hiermt` are as follows:

- `formula` parameter accepts a standard R formula object (R Core Team, 2023). For further details, refer to `??stats::formula()`. Within this formula, the left-hand side denotes the outcome(s), and the right-hand side identifies the explanatory variable(s). The period symbol "." can be employed on either side of the formula to encompass all other columns in the dataset not explicitly designated on the opposing side.

- `data` parameter is reserved for a data frame containing the variables mentioned in the formula. If this parameter is omitted, the function presumes that all requisite variables are explicitly defined in the formula.

- `global_test` parameter determines the global test method used to amalgamate marginal p-values within the hierarchical tree. Users have three options: "bonferroni," "ghc," and "gbj," with "ghc" set as the default.

- `alpha` parameter sets the significance level for hypothesis testing within the tree structure. While its default value is 0.05, users can modify it. To circumvent any significance testing decisions and just return adjusted p-values, set this parameter to 1.

- `linkage` parameter determines the agglomeration method employed in the `hclust` function to craft the dendrogram.

- `multcomp` is a logical parameter indicating whether to undertake multiple comparisons. Presently, the `hiermt` package exclusively supports pairwise comparisons.

```
hiermt(
  formula = .~x,
  data = df,
  global_test = "ghc",
  alpha = 0.05,
  linkage = "ward.D2",
  mult_comp = FALSE
)
```

Upon execution, the object of class `hiermt` is produced and it is a list with components:

- `hier_attr` object of class `data.table` or `data.frame` which provides information about each node in the dendrogram. Within this table, `node_counter` enumerates each node, `node` designates the response variable(s) present at each node, and `is_node_leaf` indicates if a node is a leaf or terminal. Relationships are defined through `descendants`, `ancestors`, `parent`, and `sibling`, which delineate the nodes below, the nodes directly leading above, the immediate node above, and nodes sharing the same parent, respectively. `is_sibling_leaf` specifies if a node's sibling is terminal, and adjustment outlines modifications to the `node_pvalue`, a raw p-value derived from either a t-test or ANOVA F-test. `adj_pvalue` represents the adjusted p-values, and `h_adj_pvalue` shows adjustments based on the hierarchical relationship.

- `grid_attr` data table contains details about multiple comparisons. If no comparison is performed, it returns the message: `[1] "No grid attributes, multiple comparisons were not performed."` However, when a comparison is executed, the table includes response_names, the labels for terminal nodes; `emmeanss`, the `emmeans` object for pairwise comparisons; `mult_comp_pvalues`, a vector of pairwise p-values; `adj_mult_comp_pvalues`, adjusted pairwise p-values; `h_adj_mult_comp_pvalues`, hierarchically adjusted p-values; and `mult_comp contrasts`, which stores contrasts compared for each in a vector.

- `dend` is the dendrogram object used in creating the hierarchy.

- `mult_comp` is a user-provided argument inquiring if multiple comparisons should be made.

- `alpha` user provided significance level.

- `call` represents the formula provided by the user.

Example Analysis

The `hiermt` package includes two datasets: `cachexia` (Eisner et al., 2010) and `age` (Metabolomics Workbench, 2020), which we will use to illustrate the functions within the package.

- `cachexia` consists of metabolite concentrations obtained from urine samples collected from 77 subjects from two groups; `cachexic` (patients with muscle loss) and `control`. The dataset contains 77 observations and 65 columns, with a column containing group information and the remaining columns containing the metabolite concentrations for the 64 named metabolites.

- `age` consists of metabolite concentrations obtain from brain samples collected from 29 mice of three age groups; `young` (mice younger than 15 months), `old` (mice between 15 and 20 months), `mid age` (mice older than 20 months). The dataset contains 29 observations and 134 columns , with a column containing age group information and the remaining columns containing the metabolite concentrations for the 133 named metabolites.

We will begin by applying the `hiermt` function to the cachexia example, as demonstrated in the code snippet below. Given that the cachexia dataset comprises only two groups, we deactivate multiple comparison adjustments by setting `mult_comp` to `FALSE`. For the global hypothesis test, we opt for the "ghc" option. Additionally, we choose "ward.D2" as our agglomeration linkage method, which represents Ward's method. The significance threshold, `alpha`, is set at 0.05. The formula is configured as `.~muscle_loss`, stating that all columns in the dataset, with the exception of `muscle_loss`, are to be considered outcome variables. The results of this operation is stored in the `hiermt_cachexia` object, specifically created for the cachexia dataset. After running this, if you inspect the `hiermt_cachexia` object, it should reveal the results of the hierarchical multiple testing procedure applied to the `cachexia` dataset with the specified parameters.

```
hiermt_cachexia <- hiermt(
  formula = .~muscle_loss,
  data = cachexia,
  global_test = "ghc",
  alpha = 0.05,
  linkage = "ward.D2",
  mult_comp = FALSE
)
```

```
hiermt_cachexia
```

```
## This is an object of class 'hiermt'.
```

The `hiermt_cachexia` possesses several attributes, the foremost being `hier_attr`, which displays various attributes for all nodes within the hierarchy, including the adjusted p-values. The subsequent attribute, `grid_attr`, generates a message stating, "No grid attributes, multiple comparisons were not performed." This message indicates the absence of pairwise comparisons in this analysis. When utilizing the `plot(hiermt_cachexia)` function, it presents a visual representation of the hierarchically adjusted p-values. Within this diagram, nodes colored in gray signify p-values that are equal to or surpass the designated threshold, α. These grayed nodes represent regions where the evidence was deemed insufficient to warrant further downward testing.

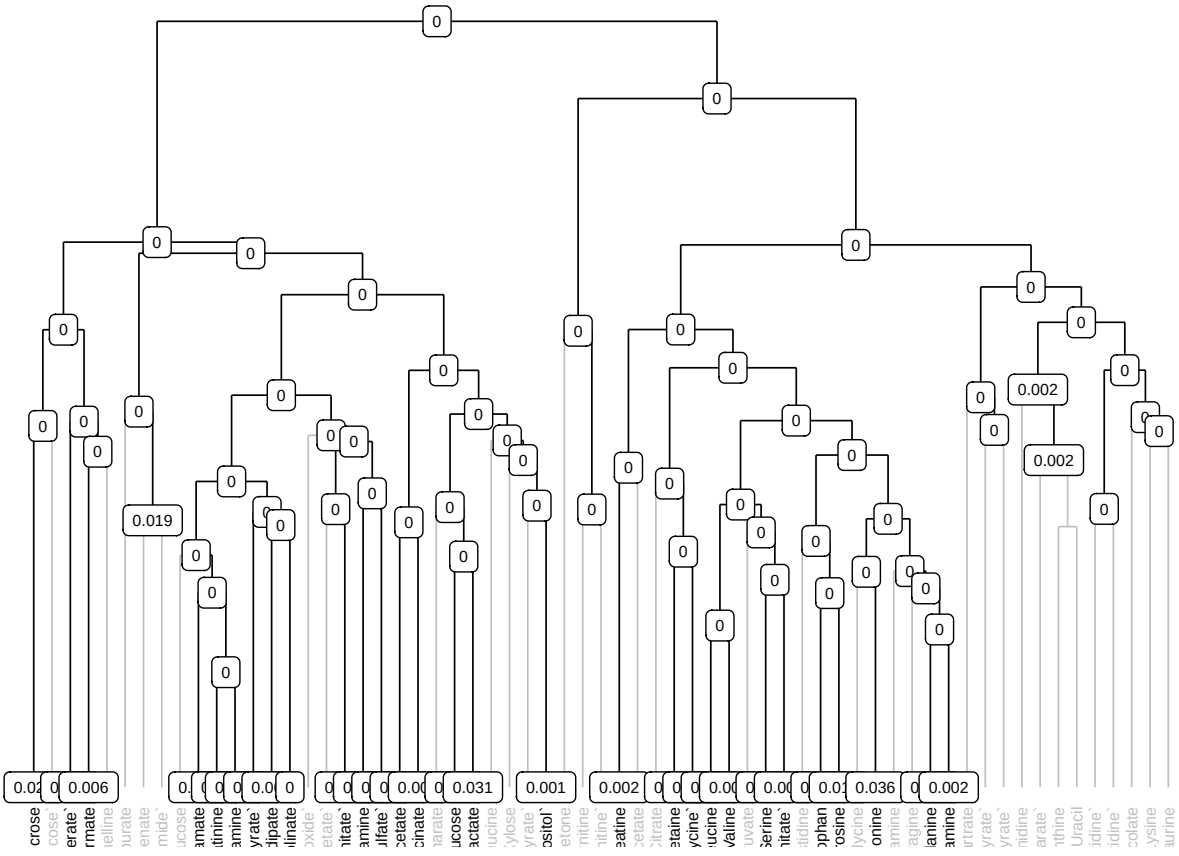
```
View(hiermt_cachexia$hier_attr)
```

	node_counter	node	is_node_leaf	descendants	ancestors	parent	sibling	is_sibling_leaf	adjustment	node_pvalue	adj_pvalue	h_adj_pvalue
1	1	c("Sucrose", "`1,6-Anhydro-beta-D-glucose`", "`3-H [...]	0	1:125	1	integer(0)	integer(0)	integer(0)	1.000000	1.398881e-14	1.398881e-14	1.398881e-14
2	2	c("Sucrose", "`1,6-Anhydro-beta-D-glucose`", "`3-H [...]	0	2:58	1:2	1	59	0	2.172414	6.439294e-15	1.398881e-14	1.398881e-14
3	3	c("Sucrose", "`1,6-Anhydro-beta-D-glucose`", "`3-H [...]	0	3:11	1:3	2	12	0	12.600000	4.063416e-13	5.119905e-12	5.119905e-12
4	4	c("Sucrose", "`1,6-Anhydro-beta-D-glucose`")	0	4:6	1:4	3	7	0	31.500000	1.698659e-04	5.350776e-03	5.350776e-03
5	5	Sucrose	1	5	1:5	4	6	1	31.500000	1.194325e-01	1.000000e+00	1.000000e+00
6	6	`1,6-Anhydro-beta-D-glucose`	1	6	c(1, 2, 3, 4, 6)	4	5	1	31.500000	5.079137e-02	1.000000e+00	1.000000e+00

```
hiermt_cachexia$grid_attr
```

```
## [1] "No grid attributes, multiple comparisons were not performed."
```

```
plot(hiermt_cachexia)
```



In the age example, due to the presence of more than two groups, we can enable the `mult_comp = TRUE` option to facilitate testing for pairwise differences. Activating this option leads to the generation of a dendrogram accompanied by a grid when the `plot(hiermt_age)` function is invoked. This grid is structured as a matrix, where the rows correspond to the number of metabolites (outcomes) and the columns correspond to the number of possible pairwise group comparisons. Each cell within this matrix represents an individual pairwise comparison for a particular metabolite. As with the dendrogram, squares that are shaded gray indicate instances where the test yielded a p-value greater than or equal to the threshold value α.

```
hiermt_age <- hiermt(
  formula = .~Factors,
  data = age,
  global_test = "ghc",
  alpha = 0.05,
  linkage = "ward.D2",
  mult_comp = TRUE
)

hiermt_age
```

```
## This is an object of class 'hiermt'.
```

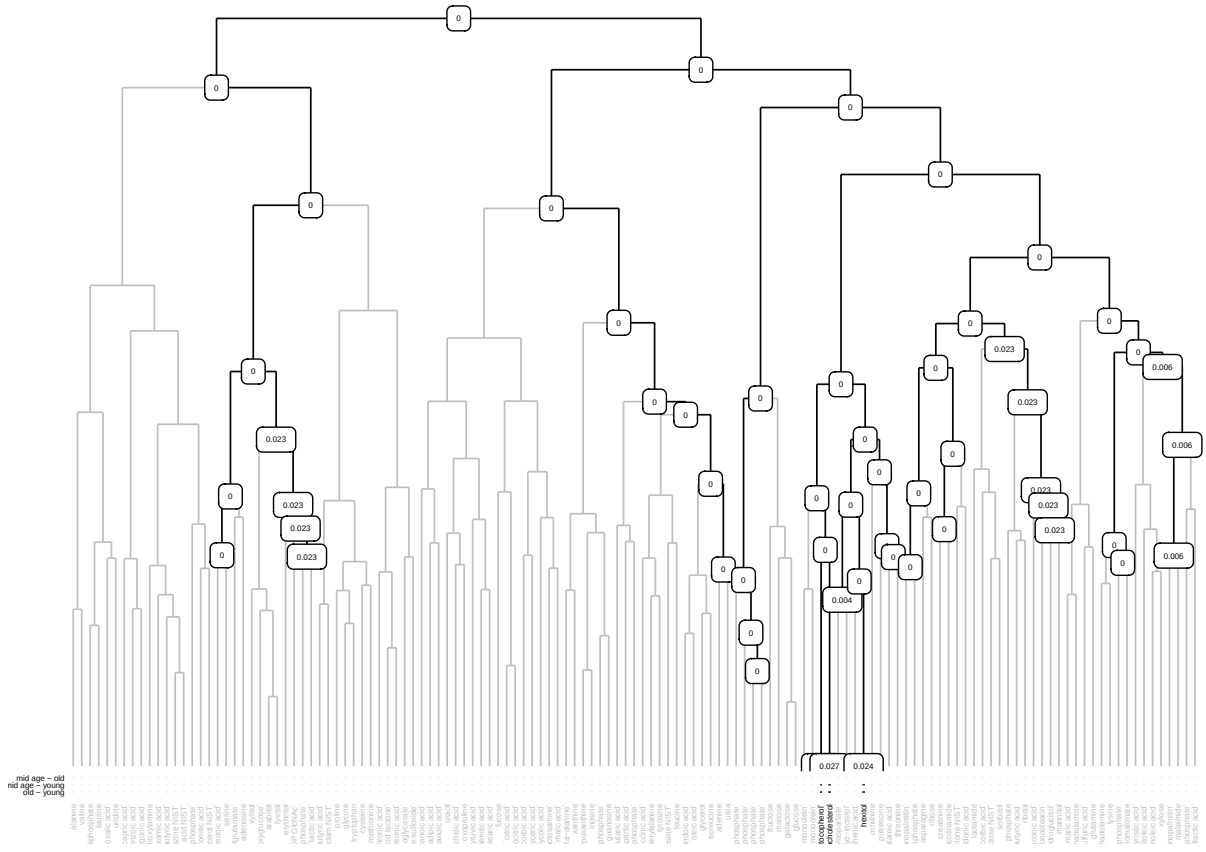
```
View(hiermt_age$hier_attr)
```

	node_counter	node	is_node_leaf	descendants	ancestors	parent	sibling	is_sibling_leaf	adjustment	node_pvalue	adj_pvalue	h_adj_pvalue
1	1	c("alanine", "valine", "methanolphosphate", "tauri [...]	0	1:265	1	integer(0)	integer(0)	integer(0)	1.000000	2.997602e-14	2.997602e-14	2.997602e-14
2	2	c("alanine", "valine", "methanolphosphate", "tauri [...]	0	2:82	1:2	1	83	0	3.243902	2.151890e-11	6.980520e-11	6.980520e-11
3	3	c("alanine", "valine", "methanolphosphate", "tauri [...]	0	3:35	1:3	2	36	0	7.823529	8.671667e-01	1.000000e+00	1.000000e+00
4	4	c("alanine", "valine", "methanolphosphate", "tauri [...]	0	4:14	1:4	3	15	0	22.166667	4.914145e-01	1.000000e+00	1.000000e+00
5	5	c("alanine", "valine")	0	5:7	1:5	4	8	0	66.500000	2.920032e-01	1.000000e+00	1.000000e+00
6	6	alanine	1	6	1:6	5	7	1	133.000000	4.821544e-01	1.000000e+00	1.000000e+00
7	7	valine	1	7	c(1, 2, 3, 4, 5, 7)	5	6	1	133.000000	2.101489e-01	1.000000e+00	1.000000e+00
8	8	c("methanolphosphate", "taurine", "`oxalic acid`", [...]	0	8:14	c(1, 2, 3, 4, 8)	4	5	0	33.250000	5.707246e-01	1.000000e+00	1.000000e+00

```
View(hiermt_age$grid_attr)
```

	response_names	emmeanss	mult_comp_pvalues	adj_mult_comp_pvalues	h_adj_mult_comp_pvalues	mult_comp_contrasts
1	`1-methylgalactose NIST`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.0281017661055282, 0.627911080138402, 0.0474283 [...]	c(0.0843052983165845, 1, 0.142285061077908)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
2	`1-methylinosine NIST`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.935617745004993, 0.945411054020267, 0.86564317 [...]	c(1, 1, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
3	`1-monoolein`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.462571427884097, 0.489995989730302, 0.11315922 [...]	c(1, 1, 0.339477661653381)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
4	`1-monopalmitin`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.384771325884623, 0.144466856217373, 0.48787954 [...]	c(1, 0.43340056865212, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
5	`1-monostearin`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.309708260760513, 0.117084069573344, 0.51301266 [...]	c(0.92912478228154, 0.351252208720031, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
6	`1_5-anhydroglucitol`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.0367906108141635, 0.0136308314065512, 0.618094 [...]	c(0.110371833244249, 0.0408924942196537, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
7	`2-hydroxyglutaric acid`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.293852317702442, 0.755797677345995, 0.12822858 [...]	c(0.881556953107327, 1, 0.38468576330326)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
8	`2-ketobutyric acid`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.479158520564299, 0.779904966120876, 0.62547170 [...]	c(1, 1, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
9	`2-monoolein`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.880463198988406, 0.227271227232326, 0.22970040 [...]	c(1, 0.681813681696977, 0.689101200837141)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
10	`2-monopalmitin`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.215856674891661, 0.00111058780501247, 0.011476 [...]	c(0.647570024674984, 0.00331763415153742, 0.034430 [...]	c(0.647570024674984, 0.36121813120751, 0.361218131 [...]	c("mid age - old", "mid age - young", "old - young [...]
11	`2_5-dihydroxypyrazine NIST`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.503764171313419, 0.830131128703463, 0.60536433 [...]	c(1, 1, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
12	`3-hydroxybutyric acid`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.615915587442141, 0.464685268809705, 0.79254642 [...]	c(1, 1, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
13	`3-phosphoglycerate`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.334964855794693, 0.156692669385444, 0.59954887 [...]	c(1, 0.476078008156333, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
14	`4-aminobutyric acid`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.778503914700122, 0.908077450846738, 0.85074331 [...]	c(1, 1, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
15	`6-deoxyglucose`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.468095503862587, 0.647634549406559, 0.75854547 [...]	c(1, 1, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
16	`N-acetyl-D-galactosamine`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.904869965337397, 0.449043878559226, 0.32256668 [...]	c(1, 1, 0.96770004553276)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
17	`N-acetylaspartate`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.705133384270404, 0.0773700478234219, 0.0182542 [...]	c(1, 0.232110143470266, 0.0547627952961825)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
18	`N-acetylglutamate`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.0602415424728546, 0.190164309461388, 0.4887914 [...]	c(0.180724627418564, 0.570492928384164, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]
19	`UDP GlcNAc`	list(emmeans = new("emmGrid", modelInfo = list(ca [...]	c(0.803873997945674, 0.743925722957535, 0.51578934 [...]	c(1, 1, 1)	c(1, 1, 1)	c("mid age - old", "mid age - young", "old - young [...]

```
plot(hiermt_age)
```



APPENDIX B

SUPPLEMENTAL MATERIALS FOR CHAPTER 3

Proof of Theorem 4.1

By definition of the hierarchical testing structure, we know that $H_1 : C_k \subset C_1$ for $k = 2, \ldots, K$ and $H_2, \ldots, H_k : C_k \cap C_{k'} = \emptyset$ or $C_k \subset C_{k'}, k \neq k'$. Let $\tilde{\mathfrak{T}}_0$ be such that

$$\tilde{\mathfrak{T}}_0 = H_k \in \mathfrak{T}_0 : C_{k'} \subset C_k, \forall\ H_{k'} \in \mathfrak{T}_0\, k = k'.$$

This means that if H_k is in $\tilde{\mathfrak{T}}_0$, then H_k can not be a descendant of any other null hypothesis in $\mathfrak{T}_0$. Consequently,

$$\Pr(\exists\ H_k \in \mathfrak{T}_0 : \pi_{k,\text{h-adj}} \leq \alpha) = \Pr(\exists\ H_k \in \tilde{\mathfrak{T}}_0 : \pi_{k,\text{h-adj}} \leq \alpha),$$

since $\forall\ H_k \in \mathfrak{T}_0,\ \exists\ H_{k'} \in \tilde{\mathfrak{T}}_0 : C_{k'} \subseteq C_k$. Now,

$$\Pr(\exists\ H_k \in \tilde{\mathfrak{T}}_0 : \pi_{k,\text{h-adj}} \leq \alpha) \leq \Pr(\exists\ H_k \in \tilde{\mathfrak{T}}_0 : \pi_{k,\text{adj}} \leq \alpha),$$

since $\pi_k^{\text{h-adj}} \leq \pi_k^{\text{adj}}$. From Boole's inequality,

$$\Pr\left\{ \bigcup_{H_k \in \tilde{\mathfrak{T}}_0} \pi_{k,\text{adj}} \leq \alpha \right\} \leq \sum_{H_k \in \tilde{\mathfrak{T}}_0} \Pr(\pi_{k,adj} \leq \alpha),$$

and it follows that

$$\sum_{H_k \in \tilde{\mathfrak{T}}_0} \Pr(\pi_{k,adj} \leq \alpha) = \sum_{H_k \in \tilde{\mathfrak{T}}_0} \Pr(\pi_k \leq \alpha r_k / Q) = \alpha / Q \sum_{H_k \in \tilde{\mathfrak{T}}_0} r_k.$$

Thus, to show family-wise error rate control, we must show that

$$\sum_{H_k \in \tilde{\mathfrak{T}}_0} r_k \leq Q$$

We know that

$$\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| \leq Q$$

.

because if $H_1 \in \mathfrak{T}_0$, then $H_k,\ k = 2, \ldots, K$ can not be in $\tilde{\mathfrak{T}}_0$ by definition, and thus $\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| = |C_1| = Q$. However, if $H_1 \notin \mathfrak{T}_0$, then at least one of its children is also not in $\mathfrak{T}_0$ meaning $\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| < Q$. Hence, $\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| \leq Q$. For every $H_k \in \tilde{\mathfrak{T}}_0$, all its ancestors are false null hypotheses. This is because if it has an ancestor that is true, then by definition that ancestor must be in $\tilde{\mathfrak{T}}_0$ as such, H_k will not be in $\tilde{\mathfrak{T}}_0$. Additionally, its sibling is false because if the parent of H_k is false, and H_k is true, then the sibling must be false because at least one of the children must be false. The falsehood of the sibling of H_k implies that it can not be $\tilde{\mathfrak{T}}_0$. Therefore,

$$\sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| \leq Q - |C_{\mathrm{si}_{k'}}|$$

Let $H_{k'}$ be a hypothesis in $\tilde{\mathfrak{T}}_0$ for which its sibling is a terminal node, then

$$\begin{aligned} \sum_{H_k \in \tilde{\mathfrak{T}}_0} r_k &= \sum_{H_k \in \tilde{\mathfrak{T}}_0 / H_{k'} \in \tilde{\mathfrak{T}}_0} |C_k| + \sum_{H_{k'} \in \tilde{\mathfrak{T}}_0} |C_k| + |C_{\mathrm{si}_{k'}}| \\ &= \sum_{H_k \in \tilde{\mathfrak{T}}_0} |C_k| + \sum_{H_{k'} \in \tilde{\mathfrak{T}}_0} |C_{\mathrm{si}_{k'}}| \end{aligned}$$

which we know is less or equal to Q. Hence the proof.

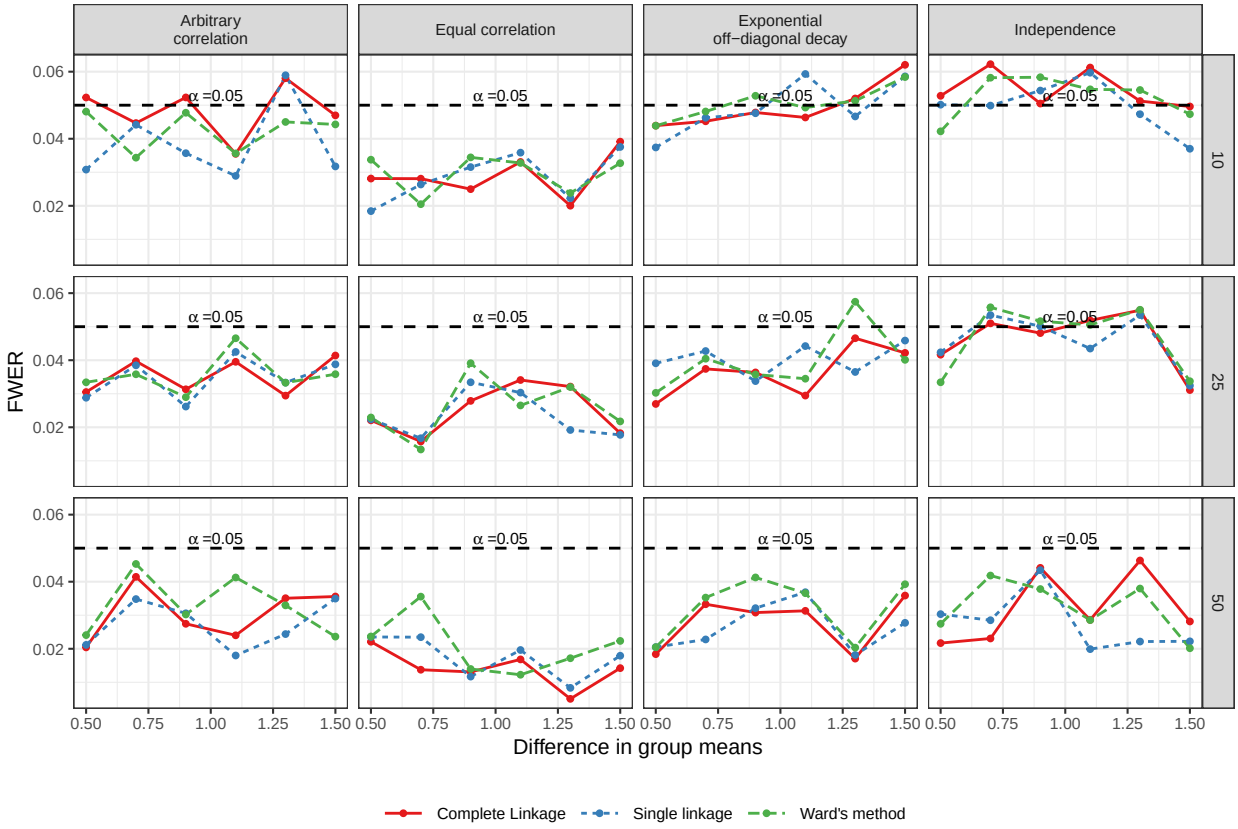


Figure B.1: Family wise error rate estimations using hierarachical testing with Bonferroni global test at the leaf nodes compared across different correlations, sparsity levels, and linkage criteria. These calculations are based on 1000 simulation iterations with 100 variables, considering scenarios with two groups, one with a sample size of 50 and the other with 75 (α= 0.05).

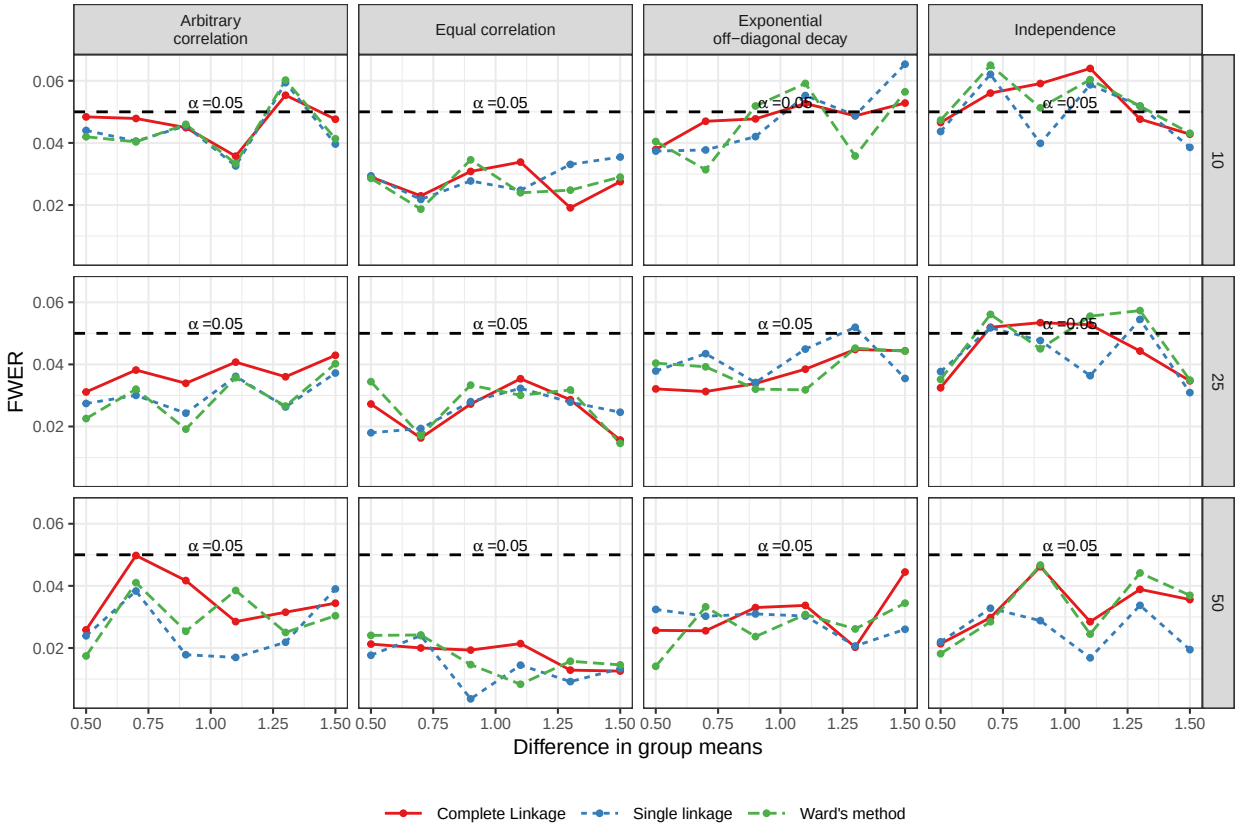


Figure B.2: Family wise error rate estimations using hierarachical testing with GBJ global test at the leaf nodes compared across different correlations, sparsity levels, and linkage criteria. These calculations are based on 1000 simulation iterations with 100 variables, considering scenarios with two groups, one with a sample size of 50 and the other with 75 (α= 0.05).

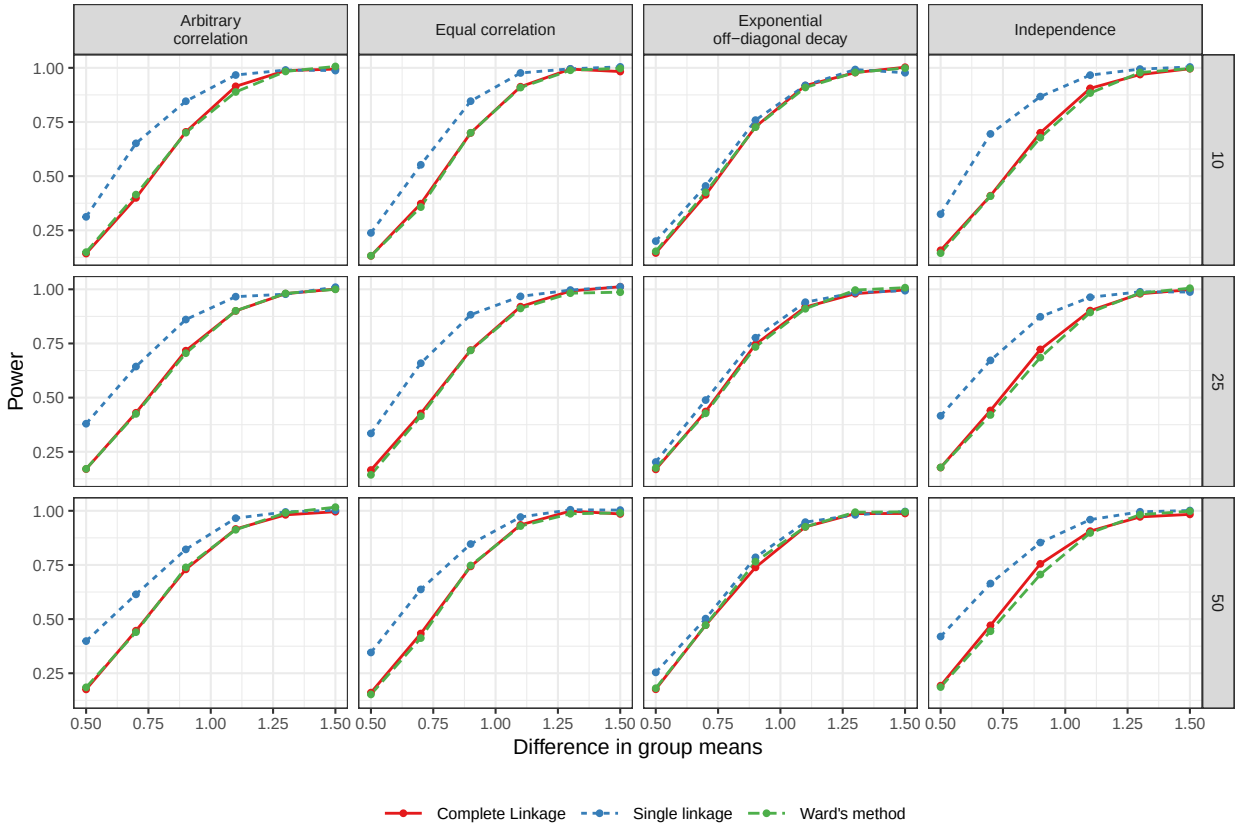


Figure B.3: Average power estimations using hierarachical testing with Bonferroni global test at the leaf nodes compared across different correlations, sparsity levels, and linkage criteria. These calculations are based on 1000 simulation iterations with 100 variables, considering scenarios with two groups, one with a sample size of 50 and the other with 75(α= 0.05).

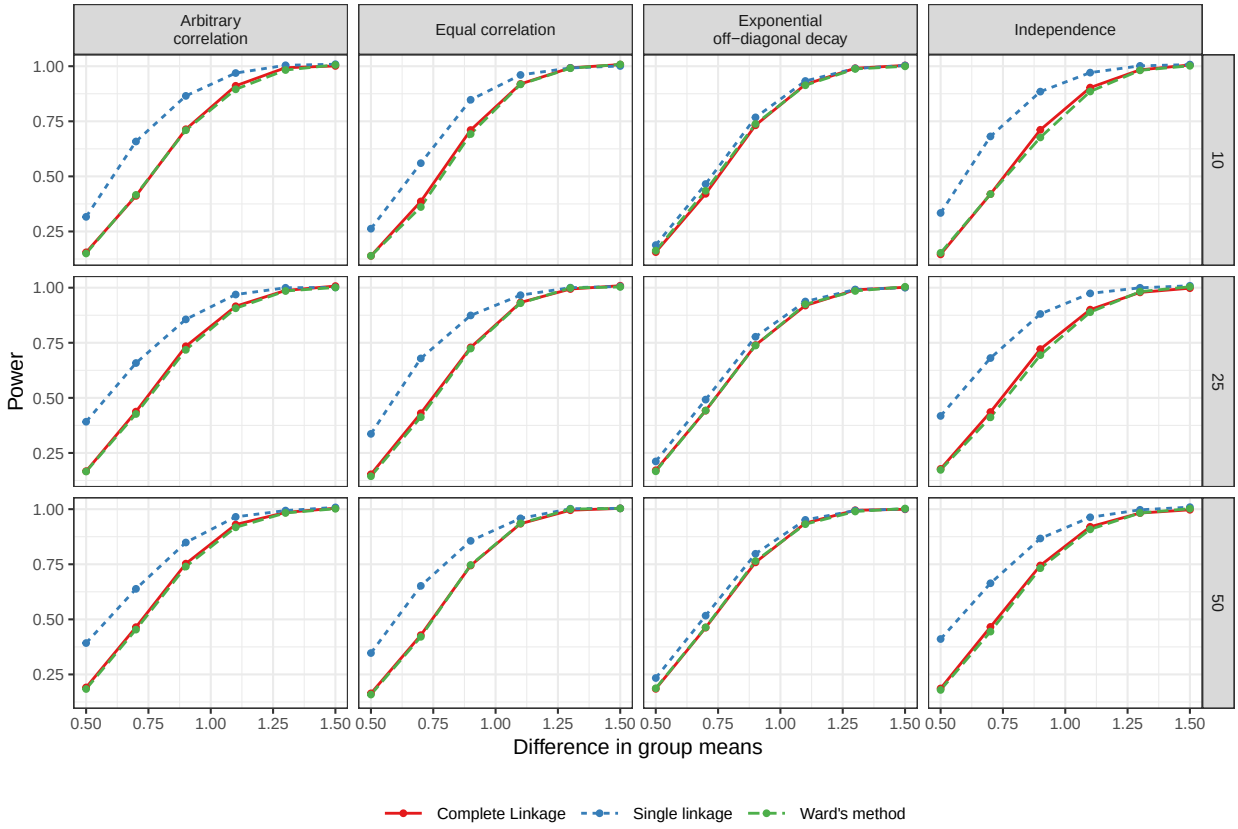


Figure B.4: Average power estimations using hierarachical testing with GBJ global test at the leaf nodes compared across different correlations, sparsity levels, and linkage criteria. These calculations are based on 1000 simulation iterations with 100 variables, considering scenarios with two groups, one with a sample size of 50 and the other with 75(α= 0.05).

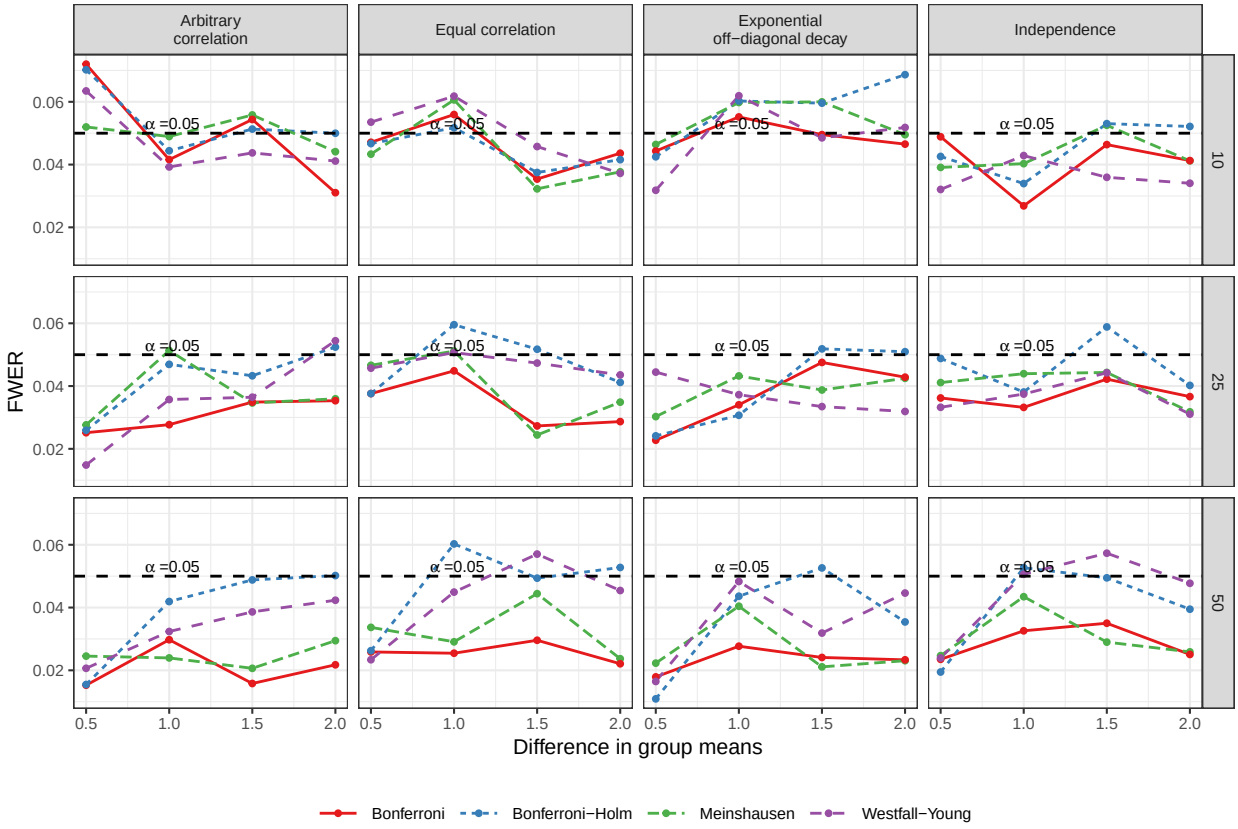


Figure B.5: family-wise error rates using hierarachical testing with Bonferroni global test at the leaf nodes compared to Bonferroni, Bonferroni-Holm, and Westfall-Young adjustments, evaluated across various correlation and sparsity levels. These calculations are based on 1000 simulation iterations with 100 variables, considering scenarios with two groups, one with a sample size of 50 and the other with 75(α= 0.05).

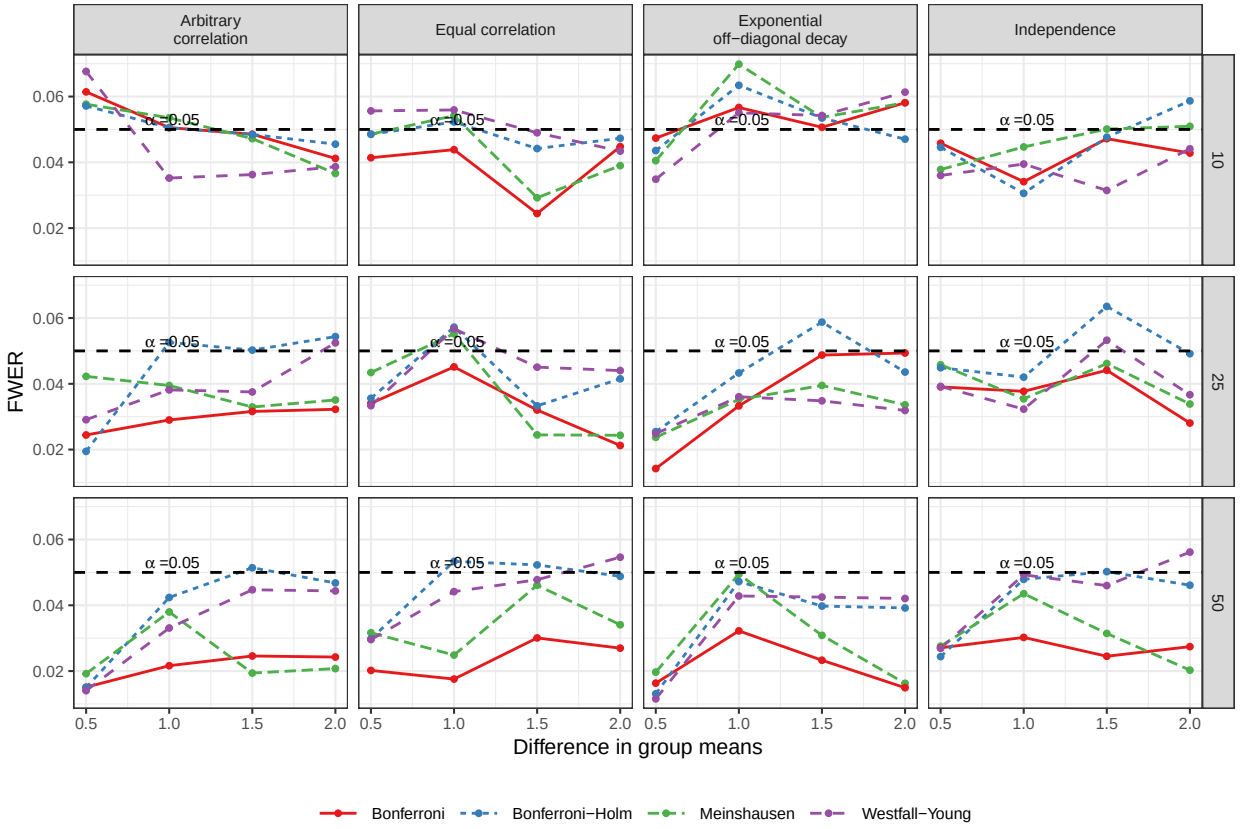


Figure B.6: family-wise error rates using hierarachical testing with GBJ global test at the leaf nodes compared to Bonferroni, Bonferroni-Holm, and Westfall-Young adjustments, evaluated across various correlation and sparsity levels. These calculations are based on 1000 simulation iterations with 100 variables, considering scenarios with two groups, one with a sample size of 50 and the other with 75(α= 0.05).

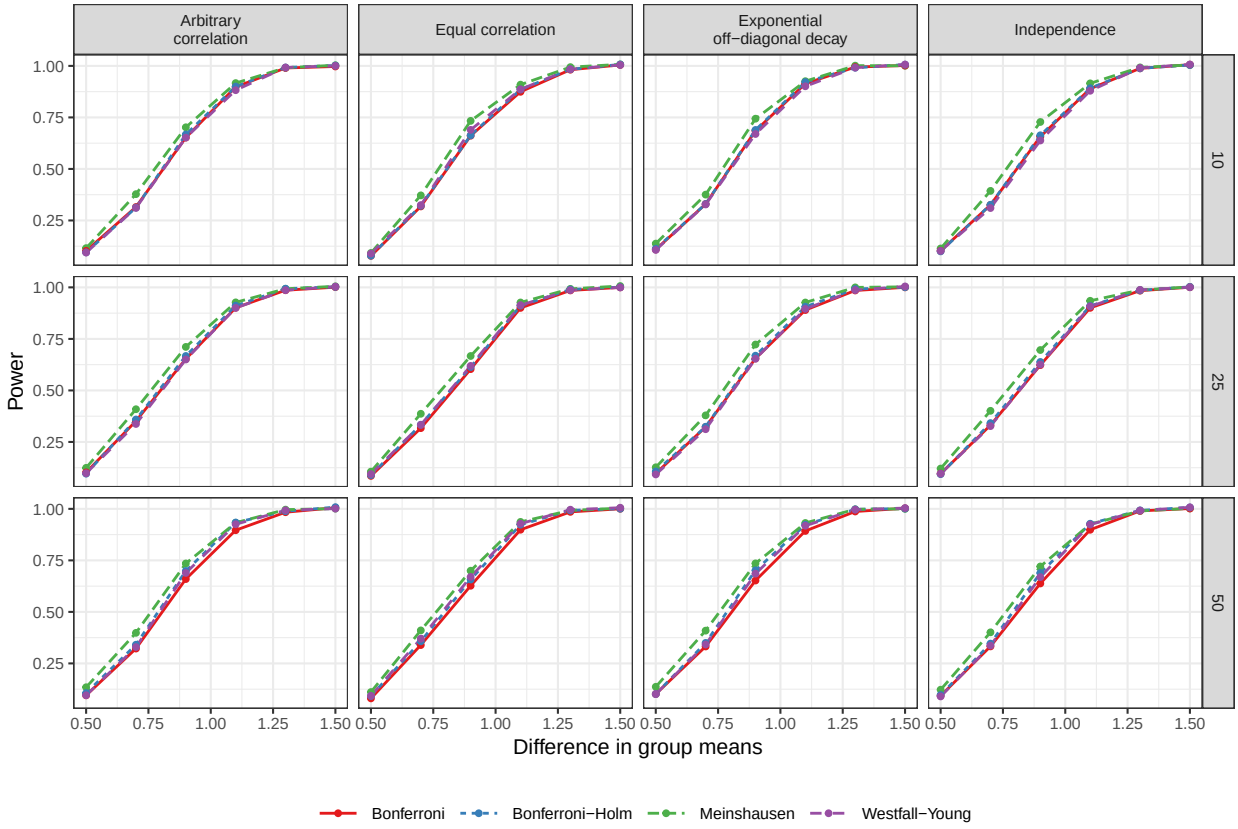


Figure B.7: Average power estimations using hierarachical testing with Bonferroni global test at the leaf nodes compared to Bonferroni, Bonferroni-Holm, and Westfall-Young adjustments, evaluated across various correlation and sparsity levels. These calculations are based on 1000 simulation iterations with 100 variables, considering scenarios with two groups, one with a sample size of 50 and the other with 75(α= 0.05).

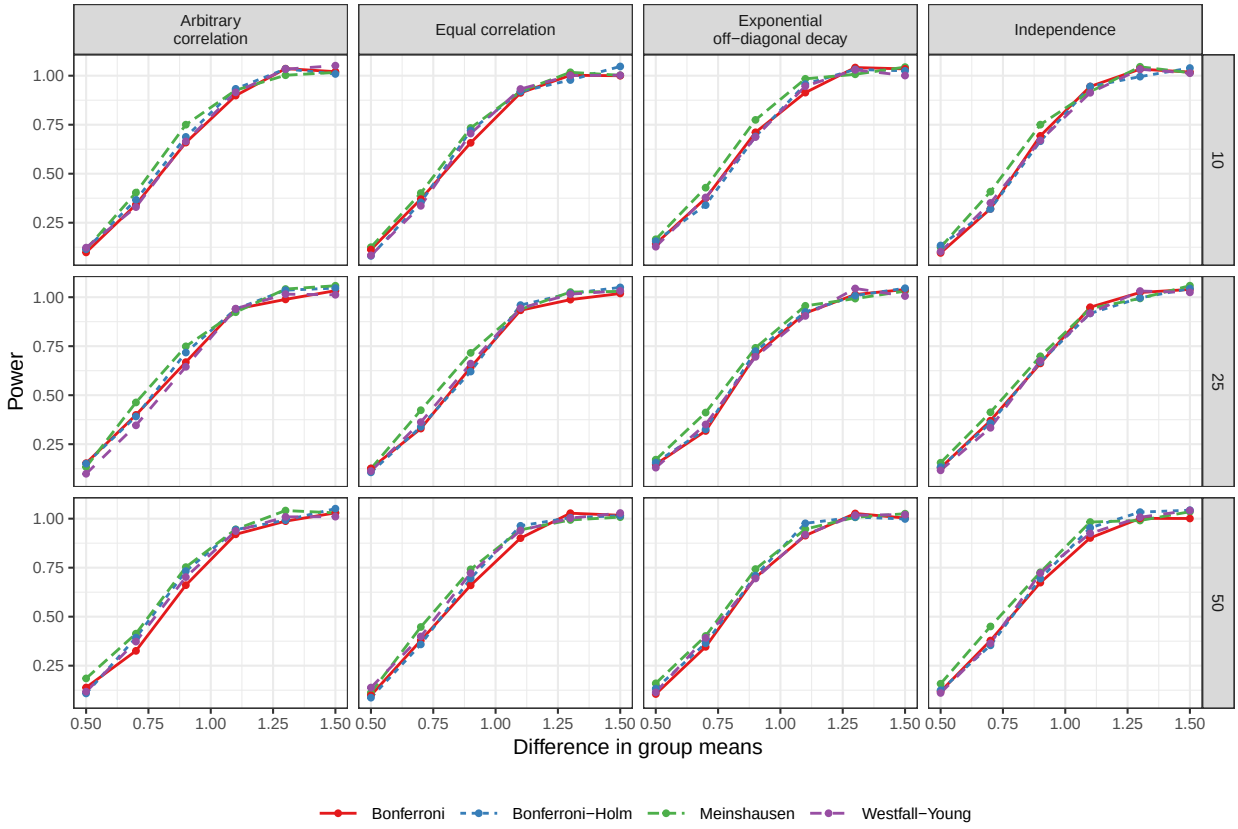


Figure B.8: Average power estimations using hierarachical testing with GBJ global test at the leaf nodes compared to Bonferroni, Bonferroni-Holm, and Westfall-Young adjustments, evaluated across various correlation and sparsity levels. These calculations are based on 1000 simulation iterations with 100 variables, considering scenarios with two groups, one with a sample size of 50 and the other with 75(α= 0.05).

APPENDIX C

SUPPLEMENTAL MATERIALS FOR CHAPTER 4

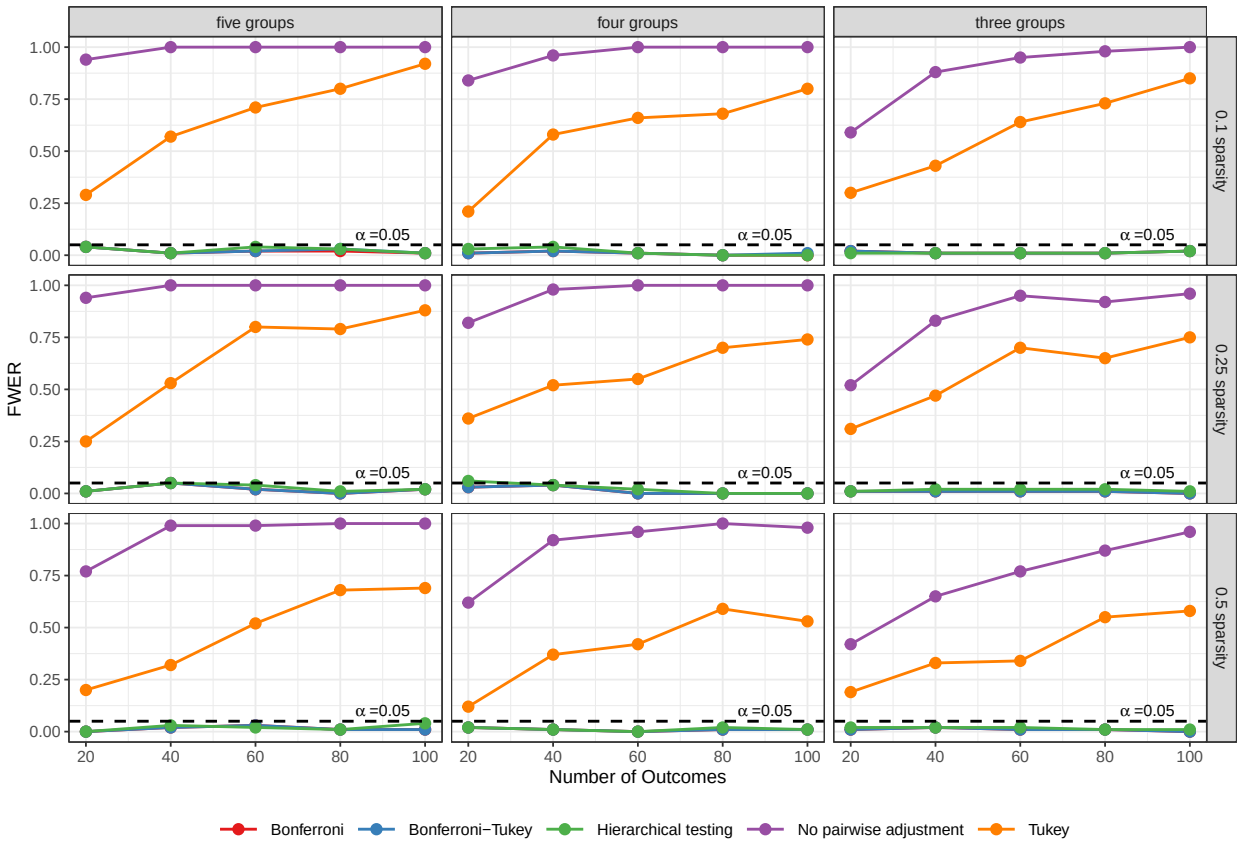


Figure C.1: Empirical family-wise error rates for all pairwise comparisons conducted using the Bonferroni, Bonferroni-Tukey, Hierarchical Testing Method, Tukey-only adjustment, and no adjustment. These rates were obtained from 1000 simulation iterations on 100 variables in each setting, with scenarios involving three, four, and five groups respectively, each group having sample sizes ranging between 50 and 75 ($\alpha = 0.05$).

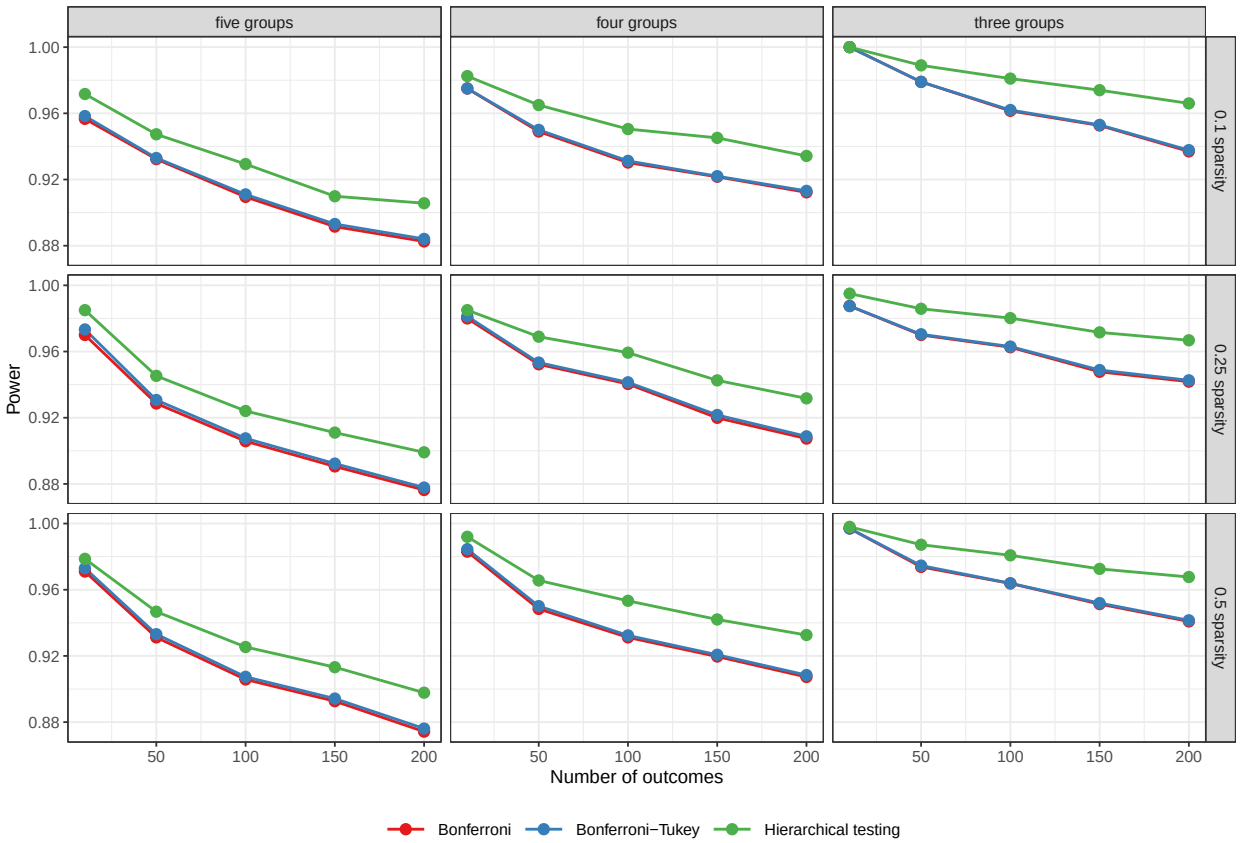


Figure C.2: Average empirical power estimations for all pairwise comparisons conducted using the Bonferroni, Bonferroni-Tukey, Hierarchical Testing Method, Tukey-only adjustment, and no adjustment. These estimations were derived from 1000 simulation iterations on 100 variables across settings, encompassing scenarios with three, four, and five groups respectively, each group maintaining sample sizes between 50 and 75 ($\alpha = 0.05$).